



# Early Proceedings of the AGACSE 2015 Conference

Edited by

Sebastià Xambó Descamps

Josep M. Parra Serra

Ramon González Calvet



UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH

Facultat d'Informàtica de Barcelona





Early Proceedings

of the

AGACSE 2015 Conference



APPLIED GEOMETRIC ALGEBRA  
IN  
COMPUTER SCIENCE AND ENGINEERING

July 29-31, 2015 Barcelona, Spain

**Early Proceedings**  
of the  
**AGACSE 2015 Conference**

FACULTAT D'INFORMÀTICA DE BARCELONA

Copyright © 2015  
FACULTAT D'INFORMÀTICA DE BARCELONA  
All rights reserved

ISBN: 978-84-606-9982-8

Printed in Barcelona (Spain)  
Composition: AGACSE 2015  
Local Organizing Committee

#### COPYRIGHT AND REPRINT PERMISSIONS

The intellectual property rights belong to the different authors. Abstracting is permitted with credit to the source.

For the copying, reprint, or republication permission, write to the authors in case of a single paper and to the editors otherwise.

# AGACSE 2015

## Early Proceedings Index

### Plenary lectures

David **HESTENES** (Keynote Lecture)

*Fifty Years with Geometric Algebra: a retrospective.*

Pierre **ANGLÈS**

*Geometric algebras and spinors*

Eduardo **BAYRO**-Corrochano

*Geometric Algebra for Cybernetics*

Leo **DORST**

*Projective Transformations as Versors*

Eckhard **HITZER**

*Fourier Transformations in Conformal Geometric Algebra*

Anthony **LASENBY**

*Geometric Algebra as a unifying language for Physics & Engineering  
and its use in the study of Gravity*

Waldyr A. **RODRIGUES, Jr.**

*Concept of the Lie Derivative of Spinor Fields. A Geometric Motivated Approach.*

Steve **SANGWINE**

*MATLAB toolbox for Clifford Algebras.*

Chris **DORAN** (pre-dinner talk)

*Game theory: From Black Holes to Battlefield 4.*

## Index of contributions

- A Only abstract
- | Plenary talk.
- || Parallel talk.
- PP Poster presentation.

The order is alphabetical of the **FAMILY NAME** of the presenter.

Timo **ALHO** (||)

*Coordinate-free evaluation of integrals in geometric calculus.*

Rafael **ALVES** and Carlile Lavor (A, ||)

*Clifford algebra applied to the molecular distance geometry problem.*

Murat **AN** and Chueng-Ryong Ji (||)

*Construction of Clifford representation of spin-1 spinors by component spinors and relation of them with left ideal spinors under Clifford and Grassmann basis.*

Linda Osuna, Humberto Caballero, Oscar Carbajal, Alexander Loukianov  
and Eduardo **BAYRO-CORROCHANO** (A, PP)

*Modeling, simulation and control for a bipedal robot using CGA.*

Luis Lechuga-Gutiérrez, Oscar Carbajal-Espinosa and Eduardo **Bayro Corrochano** (A, PP)

*A geometric approach for PID controller design.*

Gehová López-González, Eduardo **BAYRO-CORROCHANO** and Nancy Arana Daniel (||)

*Parallel Clifford support vector machines using the Gaussian kernel.*

Gehová López-González, Nancy Arana Daniel, Olivier Stasse, Mehdi Benallegue,  
Eduardo **BAYRO-CORROCHANO** (PP)

*Sphere-torus-patch bounding volumes using conformal geometric algebra.*

José Gerardo Soria-García, Gerardo Altamirano-Gómez, Susana Ortega-Cisneros  
and Eduardo **BAYRO-CORROCHANO** (PP)

*FPGA Implementation of a geometric voting scheme  
for the extraction of geometric entities from images.*

Peter **CAMERON** (||)

*Linking gauge theory gravity with quantized impedances.*

Leobardo **CAMPOS-MACÍAS**, Oscar Carbajal-Espinosa, Alexander Loukianov  
and Eduardo Bayro Corrochano (||)

*Inverse kinematics for a 6-DOF leg walking humanoid.*



Pablo **COLAPINTO** (| |)

*Composing surfaces with conformal rotors.*

Oliver **CONRADT** (A, | |)

*Comparing Grassmann and projective algebra.*

Claude **DAVIAU** (|)

*Three Clifford algebras for four kinds of interactions.*

Pierre-Philippe **DECHANT** (|)

*The  $E_8$  geometry from a Clifford perspective.*

Leo **DORST** (A, |)

*The construction of 3D conformal motions.*

Rodolfo **FIORINI** (|)

*GA and CICT for stronger arbitrary multi-scale biomedical and bioengineering solutions.*

Silvia **FRANCHINI**, Antonio Gentile, Filippo Sorbello, Giorgio Vassallo  
and Salvatore Vitabile (| |)

*A family of embedded coprocessors with native geometric algebra support.*

Ramon **GONZÁLEZ CALVET** (|)

*The geometric solution to the three-body problem  
and its application to colour image processing and quantum mechanics.*

Dietmar **HILDENBRAND**, Justin Albert and Patrick Charrier (| |)

*Geometric algebra computing for heterogeneous systems.*

Hongbo Li, Lei Dong, Changpeng Shao and Lei **HUANG** (Prize lecture)

*Elements of line geometry with geometric algebra.*

Rimvydas **KRASAUSKAS** (A, | |)

*Unifying theory of Pythagorean-normal surfaces.*

Peter **LEWINTAN** (A, | |)

*Soap films and the Gauss map.*

Hongbo **LI** (A, |)

*Fundamentals of 3D Clifford bracket algebra.*

Gene **McCLELLAN** (A, | |)

*Application of geometric algebra to the electroweak sector  
of the standard model of particle physics.*

E. Ulises **MOYA-SANCHEZ** and Marcela Bonell Manjarrez (|)

*Quaternion atomic phase magnification for 3D motion.*

Jaroslav Hrdina, Ales **NÁVRAT**, Peter Vasik and Radek Matousek (| |)

*Geometric control of the trident snake robot based on CGA.*

Margarita **PAPAEFTHYMIU**, George Papagiannakis, Andreas Aristidou and Marinos Ioannides (A, |)

*A conformal geometric algebra framework for mixed reality and mobile display.*

Alba **PÉREZ-GRACIA** and Federico Thomas (PP)

*Clifford algebra representation of grasping and manipulative hand actions for kinematic synthesis.*

Alessandro **PEROTTI** (|)

*Slice-regular functions over Clifford algebras and harmonic functions.*

Dimiter **PRODANOV** (PP)

*Clifford algebra support in MAXIMA.*

Rida T. Farouki, Graziano Gentili, Carlotta Giannelli, Alessandra Sestini and Caterina **STOPPATO** (A, |)

*Quaternionic polynomial problems for the construction of Pythagorean-hodograph curves.*

Murat **TANISLI** and Neslihan Sahin (A, PP)

*Electromagnetism-like Equations for Fluids in Higher Dimensions.*

Federico **THOMAS** and Alba Pérez-Gracia (|)

*On Cayley's factorization of 4D rotations and applications.*

Augusto Miss, Lino Resendis and Luis **TOVAR** (A, PP)

*Quaternionic  $F(p, q, s)$  Function Spaces.*

Jaroslav Hrdina, Ales Návrát, Peter **VASIK** and Radek Matousek (| |)

*Geometric control of robotic snakes based on CGA: (4,5,7,8-10) filtration.*

Terje **VOLD** (| |)

*Improved computational electromagnetism by least action.*

Samuel **WAINER**, Waldyr Rodrigues, Eduardo Notte Cuello, Rivera Tapia and Igor Kondrashuk (|)

*A Clifford bundle approach to the wave equation of a spin 1/2 fermion in the De Sitter manifold.*

Julio **ZAMORA-ESQUIVEL**, Alejandro Madrigal, Miguel Padilla, Allen Galaviz and Ana Paulina Cassale (| |)

*Line segments extraction from images using RBF in CGA.*

Václav **ZATLOUKAL** (| |)

*Classical field theories from Hamiltonian constraint: Canonical equations of motion and local Hamilton-Jacobi theory.*

## Foreword

This AGACSE-2015 Conference, devoted to **David Hestenes** for his masterly leadership in promoting, disseminating and teaching Clifford's Geometric Algebra, offers us the opportunity of instituting the AGACSE-2015 Prize as the first *David Hestenes Prize*.

It will be awarded to the best junior contribution to the AGACSE Conference in any of the several domains that gather together in the AGACSE community: mathematics, physics, engineering, computing science. They are all complementary and necessary from a universal perspective, faithful to the original Leibniz's dream of creating a *characteristica universalis*, a language able to provide a complete description and direct manipulation of geometrical and physical entities.

This Barcelona meeting is also an opportunity to remember and pay a due homage to the Mexican **Jaime Keller Torres**. Founder and lifetime editor of the *Advances in Applied Clifford Algebras*, he was until his untimely death one of the most active and prominent builders of the new area of Geometric Algebra. He was an extraordinary open minded scientist and a true humanist, and Barcelona, Wien and Toulouse (where he was awarded the Golden Medal of the Université Paul Sabatier) were loved and cherished cities in which he worked in and promoted the Clifford Algebra research.

Roger Boudet, Roy Chisholm, Gaston Casanova, Richard Delanghe, Artibano Micali, and especially Pertti Lounesto, are among those members of our community that would most likely have been very happy to participate in this first meeting of Clifford Geometric Algebra in Barcelona (and Spain).

In this Early Proceedings we have assembled the abstracts of invited plenary speakers and the contributions of the participants. The first abstract is that of Hestenes' keynote lecture, *Fifty Years with Geometric Algebra: a retrospective*, to which we have appended his lovely essay *Physics cum Mathematics for Modeling the Physical World*, as he will refer to it in his lecture.

As indicated in the index, the contributions of the participants are ordered according to the family name of the presenter and we have given equal consideration to the abstracts and papers stemming from a talks (plenary and in parallel) or from a poster presentation. This obeys to the wish to give equal opportunity to all participants with regard to the final proceedings. Let us take the occasion to thank the Scientific Committee for his refereeing job and for the advice: Pierre Anglès, Eduardo Bayro-Corrochano, Eckhard Hitzer, Anthony Lasenby, Hongbo Li, and Josep Manel Parra-Serra.

Our thanks to all the sponsors, and most especially the Universitat Politècnica de Catalunya, the Facultat d'Informàtica de Barcelona and the Departments of Applied Mathematics II and of Computed Architecture, which have provided all the facilities, human and material; the Catalan and Spanish mathematical societies, which in particular guaranteed that we could provide each participant a copy of the second edition of Hestenes' *Space-time Algebra*; the Facultat de Matemàtiques i Estadística of the UPC and the Centre de Recerca Matemàtica; and last, but not least, the participants, among which the some two thirds that have taken part in the AGACSE 2015 Summer School held in the preceding two days. Without the sum of all these factors this event would not have been possible. A thankful mention also to ARM (The Architecture of the Digital World), for sponsoring the special plenary lecture by Chris Doran and the MO Labs for sponsoring the glass sculpture for the Hestenes' Prize.

Finally our thanks to AGACSE, and most especially to Eduardo Bayro Corrochano, for having given us the chance to contribute to spread the word about the importance of Clifford Geometric Algebra in research and in designing better curricula for mathematics, physics and engineering studies.

The editors,

Sebastià Xambó Descamps

Josep Manel Parra Serra

Ramon González Calvet



## DAVID HESTENES

*The Genesis of Geometric Calculus* (keynote lecture)

### Summary

This will be a retrospective account of my personal journey in discovering, revitalizing and extending Geometric Algebra, with emphasis on the origin and influence of my book *Space-Time Algebra*. I will discuss guiding ideas, significant results and where they came from — with recollection of important events and people along the way. Lastly, I offer some lessons learned about life and science.

The following essay, published in these Early Proceedings for the first time, is included as the author will refer to it in his lecture.

### **Physics cum Mathematics for Modeling the Physical World**

*David Hestenes*

***Abstract:*** *Physics and mathematics have common ground in man's evolved ability to freely create mental models and use them to manage interactions with the natural world. But they differ in the ways they relate to experience.*

The *Copernican Revolution* in science culminated in Newton's *Principia* (1687), which integrated astronomy and terrestrial physics into a single science of motion. Immanuel Kant (1787) saw this as a striking union of mathematical theory with empirical fact that bridged the traditional divide between rationalism and empiricism. So he proposed a comparable "*Copernican Revolution*" in philosophy to account for it [1]. Just as Copernicus shifted the center of the universe from earth to sun, Kant shifted the focus of epistemology from structure of the external world to structure of mind. His revolutionary insight was that our perceptions and thoughts are shaped by inherent structure of our minds. He argued that the fundamental laws of nature, like the truths of mathematics, are knowable precisely because they do not describe the world as it really is but rather prescribe the structure of the world as we experience it.

Though the scientific revolution has expanded in spectacular fashion to integrate physics and astronomy with chemistry and biology, Kant's revolution in philosophy has hardly progressed. His profound influence on the epistemology of physics is evident in the writings of Einstein and Bohr as well as many other scientists and philosophers. However, continued debates on such topics as the

interpretation of quantum mechanics show no signs of consensus, and they have overlooked recent advances in cognitive science with high relevance to epistemology.

My purpose here is to open a new stage in Kant's revolution by explaining how findings of cognitive science can be marshaled to create a new "*science of mind*" with testable predictions and explanations as required of any "true" science. I begin with a restatement of Kant's primary question: *What does the structure of science and mathematics tell us about how the human mind works?* In searching for answers my working hypothesis will be: *The primary cognitive activities in science and mathematics involve making, validating and applying conceptual models!* In a word, science and mathematics are about MODELING — making and using models!

This essay argues for a "MODELING THEORY of MIND" to guide the multifarious branches of cognitive science in research on the nature of mind and brain, and the design of conceptual tools for science and mathematics. Core principles are explained and supporting evidence is sketched, but the brush is necessarily broad. More details are given in [2,3,4], especially for application to physics teaching and learning.

## I. NEWTON'S MODELING GAME

Newton did much more than provide the first mathematical formulation of a scientific theory in his *Principia*; he also demonstrated how to relate it to empirical fact. Though Kant recognized revolutionary implications for epistemology in this impressive feat, physicists have overlooked it. The issue has been thoroughly explicated in [5] by framing Newtonian theory in terms of models and modeling, so brief mention of key points is sufficient here. Newton could not make the crucial distinction between model and theory explicit in his original formulation, because the concept of model did not emerge in scientific discourse until the nineteenth century. But [5] shows that he made it implicitly. The point is that theoretical principles like Newton's Laws cannot be tested or applied except by incorporating them in models. Thus, models mediate between theory and experiment. And Newton's Laws can be regarded as a system of design principles for making models to describe, to predict, to explain and to control motions of material bodies.

Kant's insight can be explicated by noting that Newton linked up two distinct kinds of models: theoretical and empirical. A *theoretical model* derived from Newton's Laws predicts motions, while an *empirical model* derived from data describes a motion. A match between them explains a motion. In this way Newton explained Galileo's law of falling bodies and Kepler's three laws of planetary motion. Note the distinction between a theoretical Law (with a capital L) and an empirical law (with a lowercase l), also called an empirical model.

Comparison between theoretical and empirical models is such a standard practice of physicists since Newton that they seldom consider its profound epistemological implications. At its simplest, it involves creating an empirical model from data with a procedure often called "curve fitting." That's how Kepler's laws were derived. It is an important technique in the search for empirical regularities that are both quantifiable and reproducible. In high energy physics data analysis has become so complex that a new research specialty has emerged to handle it. That research, often called "phenomenology," is thus intermediate between theory and experiment.

For future analysis, it is worth noting that scientific work in all three domains is governed by definite but different rules; from mathematical rules for theorists, to measurement standards for experimentalists, to probability theory for phenomenologists. As Kant recognized, scientific objectivity requires strict adherence to rules. The question is: Where do the rules come from?

## II. FROM COMMON SENSE TO SCIENTIFIC THINKING

As we grow and learn through everyday experience, each of us develops a system of *common sense* (CS) concepts about how the world works. To evaluate introductory physics instruction, the *Force Concept Inventory* (FCI) was developed to detect differences (in student thinking) between CS concepts and Newtonian concepts about motion and its causes [6]. Results from applying the FCI were stunning from the get-go! First, the differences were huge before instruction. Second, the change was small after instruction. Third, results were independent of the instructor's experience, teaching method and peer evaluation. These results have been replicated thousands of times from high school to Harvard and in 25 different languages. The FCI is by far the most cited reference in the physics education literature, and it is widely used today to evaluate the effectiveness of teaching reforms.

Here we are interested in what the FCI tells us about human cognition. The FCI is based on a taxonomy of 35 CS concepts in 5 major categories [6]. These concepts are overlooked or summarily dismissed as misconceptions by most physicists. However, they are common outcomes from everyday experience, and they are quite serviceable for dealing with physical objects. Moreover, central CS concepts in the 5 categories have been clearly articulated and discussed by major intellects of the pre-Newtonian age, including Newton himself before the *Principia* [7]. So CS concepts should be regarded as alternative hypotheses about the physical world that, when clearly formulated, can be tested empirically.

For example, the CS concept: “a moving object implies existence of a force (a mover)” contravenes Newton's First Law. The Second Law is contravened by the concept that forces are due to “active agents” (usually living things), so there are no passive forces, although motion is deflected by passive objects called “barriers.” The Third Law is contravened by the common metaphorical notion that “interaction is like war” so in the “struggle between forces” “victory goes to the stronger.” In fact, CS thinking is shot through with metaphorical notions. One consequence of all this is that in a conventional physics course students systematically misinterpret what they hear and see in class, which goes a long way to account for the typical disastrous student performance on exams.

Ability to distinguish between CS concepts and scientific concepts in the FCI or elsewhere is not a matter of intelligence but of experience. It is acquired only by engagement with science itself, usually through academics. Remarkably, physicists seldom recall any event in their own transitions to Newtonian thinking. Typically, they presume that the world of classical physics is given directly by experience, in contrast to the subtlety and weirdness of quantum mechanics. They are blind to the subtle revolution in their own thinking that came from learning physics; for the FCI tells us that classical physics differs from common sense in almost every detail.

These facts suggest that the transition from common sense to scientific thinking is not a replacement of CS concepts with scientific concepts, but rather a realignment of intuition with experience. Science does not replace common sense. Rather, as Modeling Theory aims to show, science is a refinement of common sense differing in respect to:

- objectivity – with explicit rules & conventions for observer-independent inferences,
- precision – in measurement, in description and analysis,
- formalization – for mathematical modeling and analysis of complex systems,
- systematics – coherent, consistent & maximally integrated bodies of knowledge,
- reliability – critically tested & reproducible results,
- skepticism – about unsubstantiated claims.



### III. “What, precisely, is thinking?” — *Einstein*

Kant is unsurpassed in using introspection to analyze his own thinking. But introspection was dismissed as subjective and unreliable by behaviorists in the twentieth century, who claimed that scientific objectivity requires psychology to take its data from observable behavior under controlled conditions. However, the behaviorist straight jacket has been cast off in recent decades by the emergence of *cognitive science*, which draws its data and insights from many independent academic disciplines. Disciplinary barriers are crossed with increasing frequency, largely due to the speed and ease of electronic communication.

Human perception, memory and cognition are being studied in many different ways. The problem, as ever in science, is to identify reproducible patterns in the results. Here follows a sampling of approaches and results with high relevance to Modeling Theory.

The Learning Sciences: Research on teaching and learning is emerging as a coherent science with independent branches like physics education research (PER) devoted to a single discipline. The most robust finding in the field is that effective teaching requires matching the method to the subject matter, and that requires research embedded within each discipline.

An outstanding example of PER is Andy diSessa’s probing study into common sense notions of force [8]. He identifies a structure in common sense intuition that he calls Ohm’s p-prim. As he explains,

**Ohm’s p-prim** comprises “an agent that is the locus of an impetus that acts against a resistance to produce a result.”

Evidently this intuitive structure is abstracted from experience in pushing objects. It is an important characterization of the central Force-as-Action metaphor identified by the FCI. It also seems to be fundamental in the intuition of physicists, who often declare “A force is a push or a pull,” although they have extracted that from human action.

More generally, diSessa argues that this structure is fundamental to qualitative reasoning. He notes that the logic of Ohm’s p-prim is

the *qualitative proportion*: more effort  $\Rightarrow$  more result,  
and the *inverse proportion*: more resistance  $\Rightarrow$  less result.

This reasoning structure is often evoked for explanatory purposes in everyday experience.

As disclosed in Ohm’s p-prim, the concept of (causal) **agency** entails a basic

**Causal syntax**: agent  $\rightarrow$  (kind of action)  $\rightarrow$  on patient  $\rightarrow$  result.

DiSessa notes that this provides an interpretative framework for  $\mathbf{F} = ma$ , and he recommends exploiting it in teaching mechanics. However he does not recognize it as a basic aspectual schema for verb structure, which has been studied at length in *cognitive grammar* [9]. Aspectual concepts are generally about event structure, where events are changes of state and causes (or causal agents) induce events.

All this has direct bearing on Kant’s *Critique*. He said Hume woke him from his “dogmatic slumber” with his argument that no amount of empirical data can establish a cause-effect relation between events with certainty. Claiming that Newton’s Laws do establish causality with certainty, Kant argued that it must therefore be known prior to experience (“synthetic a priori”). One can argue instead that the fundamental Laws and Principles of science are discovered as general patterns in experience and simply adopted as postulates in our theories. But Ohm’s p-prim shows that causality

is imbedded in the way we think and so it may be a precondition for recognizing causal patterns. In this sense, at least, cognitive science supports Kant's view.

All this has bearing on other domains of cognitive science, for example, the psychology of perception. In particular, it provides more support for the view that there is no such thing as "passive perception." All perception is part of a "**perception-action cycle**." Even viewing a static visual scene is impossible without rapid movements of the eyeballs (saccades) to sample the visual input. In general, we learn about the world around us from our interactions (perception-actions) with it.

Note that the intuitive causal syntax discussed above can be construed (by metaphorical projection at least) as

**Operator syntax:** agent → (kind of action) → on patient → result,

where the action is on symbols (instead of material objects) to produce other symbols. When the symbols are words, this provides an intuitive base for verb structure expressing the action of mental agents on mental objects. The same idea has emerged independently in cognitive linguistics (see below). Also, the operator syntax provides an intuitive base for the mathematical concept of function (though probably not the only one).

Narrative Comprehension: Readers of stories construct mental models of the situation and characters described [10]. They infer causal connections relating characters' actions to their goals. They also focus attention on characters' movements, thereby activating nearby parts of the mental model. This activation is revealed in readers' faster answering of questions about such parts, with less facilitation the greater their distance from the focus. Recently visited as well as imagined locations are also activated for several seconds. These patterns of temporary activation facilitate comprehension.

Evolutionary Psychology [11] tells us that human brains evolved adaptively to enable navigation to find food and respond to threats. Successful hunting required a number of cognitive abilities: To create mental maps of the environment and plan actions, to design helpful tools, to "read" subtle clues in natural surroundings; and, finally to communicate and cooperate with other humans.

There can be little doubt that narrative emerged in human prehistory. The practice of storytelling is ancient, pre-dating not only the advent of writing, but of agriculture and permanent settlement as well. Language, an obvious prerequisite for storytelling, is likely to have emerged between 50,000 and 250,000 years ago. Cognitive linguistics (see below) aims to ascertain what language can tell us about evolved cognitive abilities.

Cognitive Psychology: Psychologist Philip Johnson-Laird [12] is a pioneer in studying human inference by manipulating mental models. His research supports the claim that *most human reasoning is inference from mental models*. We can distinguish several types of **model-based reasoning**:

- **Abductive**, to complete or extend a model, often guided by a semantic frame in which the model is embedded.
- **Deductive**, to extract substructure from a model.
- **Inductive**, to match models to experience.
- **Analogical**, to interpret or compare models.
- **Metaphorical**, to infuse structure into a model.
- **Synthesis**, to construct a model, perhaps by analogy or blending other models.
- **Analysis**, to profile or elaborate implicit structure in a model.

*Justification* of model-based reasoning requires translation from mental models to *inference from conceptual models* that can be publicly shared, like the scientific models discussed below.

In contrast, **formal reasoning** is computational, using axioms, production rules and other procedures. It is the foundation for rigorous proof in mathematics and formal logic. However, Modeling theory (see below) holds that mathematicians and even logicians reason mostly from mental models. Model-based reasoning is more general and powerful than propositional logic, as it integrates multiple representations of information (propositions, maps, diagrams, equations) into a coherently structured mental model. Rules and procedures are central to the formal concept of inference, but they can be understood as prescriptions for operations on mental models as well as on symbols.

Psychology of Spatial Perception: Everyone has imagination, the ability to conjure up an image of a situation from a description or memory. What can that tell us about mental models? Some people report images that are picture-like, similar to actual visual images. However, others deny such experience, and blind people are perfectly capable of imagination. Classical research in this domain found support for the view that *mental imagery is internalized perception*, but not without critics.

Barbara Tversky and collaborators [13] have tested the classical view by comparison to mental model alternatives. Among other things, they compared individual accounts of a visual scene generated from narrative with accounts generated from direct observation and found that they are *functionally equivalent*. A crucial difference is that perceptions have a fixed point of view, while mental models allow change in point of view. Furthermore, spatial mental models are more schematic and categorical than images, capturing some features of the object but not all and incorporating information about the world that is not purely perceptual. The general conclusion is that *mental models represent states of the world as conceived, not perceived*. To know a thing is to form a mental model of it.

Major characteristics of spatial mental models are summarized in the following list. The best fit to data is a *spatial framework model*, where each object has an *egocentric frame* consisting of mental extensions with three body axes.

### **Spatial MENTAL models**

- are *schematic*, representing only some features,
- are *structured*, consisting of *elements and relations*.
- **Elements are typically objects** (or reified things).
- **Object properties are idealized** (points, lines or paths).
- Object models are always *placed in a background* (context or **frame**).
- Individual objects are *modeled separately* from the frame, so they can move around in the frame.

The details in this list are abundantly supported by other lines of research, especially in cognitive linguistics, to which we now turn.

Cognitive Linguistics: The most extensive and coherent body of evidence comes from cognitive linguistics [14], supporting the **revolutionary thesis**: *Language does not refer directly to the world, but rather to mental models and components thereof! Words serve to activate, elaborate or modify mental models, as in comprehension of a narrative*.

This thesis rejects all previous versions of semantics, which located the referents of language outside the mind, in favor of **cognitive semantics**, which locates referents inside the mind. I see the evidence supporting cognitive semantics as overwhelming, but it must be admitted that some linguists are not convinced, and many research questions remain. Cognitive semantics can be regarded as a

culmination of Kant's revolution toward an epistemology grounded in science, though that is not often recognized by linguists.

Two pillars of cognitive linguistics deserve mention here. The first pillar is Eleanor Rosch's discovery that *natural categories* are determined by mental prototypes. For example, "birds" are classified by comparison to a prototypical bird, such as a robin. This should be contrasted with the classical concept of a *formal category* for which membership is determined by a set of defining properties, a noteworthy generalization of the container metaphor. This distinction between category types is supported by a mountain of empirical evidence on natural language use.

The second pillar is the notion of *image schema* introduced by Mark Johnson and George Lakoff. Image schemas are basic *structural units* (gestalts) that provide structure to natural language and presumably cognition. There are too many to discuss here. Many are discussed in [15] as structural elements in mathematical thinking, including four grounding metaphors for arithmetic.

Cognitive Neuroscience: Human brain structures have evolved to support perception, memory and movement, that is, all components needed to execute the perception-action cycle. But no distinct component for cognition has been identified. It seems reasonable, therefore, to conclude that cognition is executed by coopting drivers of the perception-action for internal planning and simulation.

Stanislas Dehaene reports [16]: "Mathematicians frequently evoke their "intuition" when they are able to quickly and automatically solve a problem, with little introspection into their own insight. Cognitive neuroscience research shows that "automaticity aspect" of mathematical intuition can be studied in the laboratory in reduced paradigms, and that relates to the availability of "core knowledge" associated with evolutionarily ancient and specialized cerebral subsystems." Subsystems involved in basic operations of arithmetic (such as number estimation, comparison, addition and subtraction) have been identified as genetically hardwired. The boundary between hardwired and learned mathematical abilities continues to be a rich area for further research.

The empirical research cited above supports an answer to Einstein's question: *Thinking* is a hardwired human ability to freely create mental models and use them for planning and controlling interactions with the physical world. To deepen this insight and coordinate empirical results, we need a scientific theory, to which we now turn.

#### IV. MODELING THEORY

Though Modeling Theory is proposed as a general theory of Mind embracing all aspects of cognitive science, we limit our attention here to cognition in physics and mathematics. We have seen above that the study of natural languages gives us rich information about the structure of mental models in common sense cognition. Given its greater precision and coherence, we can expect complementary and reinforcing results from studying the language of science, especially mathematics. Indeed, after spelling out the structure of **scientific models** in explicit detail below, we discuss its implications for cognition in physics and mathematics.

Our formulation of Modeling Theory rests on explication of two key concepts "model" and "morphism." We begin with the definition:

A **model** is a representation of structure in a given **system**.

A *system* is a set of related *objects*, which may be real or imaginary, physical or mental, simple or composite. The *structure* of a system is a set of relations among its objects. The system itself is called the *referent* of the model.

We often identify the model with its *representation* in a concrete inscription of words, symbols or figures (such as graphs, diagrams or sketches). But it must not be forgotten that the inscription is supplemented by a system of (mostly tacit) rules and conventions for encoding model structure.

From my experience as a scientist, I have concluded that **five types of structure suffice** to characterize *any scientific model*. Although my initial analysis was based on physics, I have concluded the classification is sufficient for all other sciences as well. As this seems to be an important empirical fact, a brief description of each type is in order here.

#### **Universal structures in scientific models [2, 17]:**

- **Systemic structure:** Its representation specifies (a) *composition* of the system (b) *links* among the parts (individual objects), (c) links to *external agents* (objects in the environment). A diagrammatic representation is usually best (with objects represented by nodes and links represented by connecting lines) because it provides a holistic image of the entire structure. Examples: electric circuit diagrams, organization charts, family trees.
- **Geometric structure:** specifies (a) *configuration* (geometric relations among the parts), (b) *location* (position with respect to a reference frame)
- **Object structure:** *intrinsic properties* of the parts. For example, mass and charge if the objects are material things, or *roles* if the objects are *agents* with complex behaviors. The objects may themselves be systems (such as atoms composed of electrons and nuclei), but their internal structure is not represented in the model, though it may be reflected in the attributed properties.
- **Interaction structure:** properties of the links (typically *causal* interactions). Usually represented as binary relations on object pairs. Examples of interactions: forces (momentum exchange), transport of materials in any form, information exchange.
- **Temporal (event) structure:** *temporal change in the state* of the system. Change in position (motion) is the most fundamental kind of change, as it provides the basic measure of time. Measurement theory specifies how to quantify the properties of a system into property variables. The state of a system is a set of values for its property variables (at a given time). Temporal change can be represented *descriptively* (as in graphs), or *dynamically* (by equations of motion or conservation laws).

Optimal precision in definition and analysis of structure is supplied by **mathematics, the science of structure.**

Both the model and its referent are structured objects, but they need not be distinct. Indeed, the usual notion of a *mathematical model* as a representation in terms of mathematical symbols does not specify any referent, so we say it is an *abstract model*. Of course, it is a perfect representation of itself. This suggests that we regard any structured object as an “abstract model.”

Our definition of “model” above is likewise abstract, because it does not specify the worlds (domains) in which the model and its referent exist as structured objects. To address this issue, Modeling Theory [3] posits three distinct worlds in which structured objects exist:

- World 1: The PHYSICAL WORLD of real things and events, including biological entities.
- World 2: The MENTAL WORLD of **mental models** generated by perception or intuition.
- World 3: The CULTURAL WORLD of human artifacts, including natural languages and mathematics in any form, written or spoken.

This helps us make a crucial distinction between mental models and conceptual models. **Mental models** are private constructions in the mind of an individual (World 2). They can be elevated to **conceptual models** by encoding model structure in symbols (World 3) that activate the individual's mental model and corresponding mental models in other minds. Thus, communication between individuals involves construction and use of shared *conceptual models*.

Note that a conceptual model establishes an *analogy* between a mental model and its symbolic representation. Mathematical models are symbolic structures, and to understand one is to create a mental model with analogous structure. Actually, *the structure is supplied by the mind not the symbols*, which are reduced to meaningless marks without a mind to interpret them.

An *analogy* is defined as a *mapping of structure* from one domain (*source*) to another (*target*) [18]. The mapping is always partial, which means that some structure is not mapped. Science sets up many kinds of analogy between and within the three worlds [3]. Thus, experimental testing or simply interpreting a scientific model (World 3) requires a mapping to a physical system (World 1) that I call a *referential analogy*. *Material analogies* relate structures of different physical objects in World 1 and this reduces to an *inductive analogy* when the objects are regarded as identical. And there are many more analogies with *computer models* (World 1).

There are other kinds of structure-preserving mappings such as *metaphors*, which Lakoff [15] defines as a projection of structure from one domain into another. I recommend formalizing all such concepts with the technical term **MORPHISM**. In mathematics a *morphism* is a *structure-preserving mapping*: Thus the terms *homomorphism* (preserves algebraic structure) and *homeomorphism* (preserves topological structure).

Now let us reconsider Kant's trenchant analysis of thinking in physics and mathematics. *Physical intuition* is accorded the same high regard by physicists that mathematicians accord to *mathematical intuition*. To quote unquestionable leaders in each field [2]: Einstein explains,

“The words or the language, as they are written or spoken, do not seem to play any role in my mechanism of thought. . . . The physical entities which seem to serve as elements in thought are certain signs and more or less clear images which can be voluntarily reproduced and combined. . . .”

Hilbert asserts,

“No more than any other science can mathematics be founded on logic alone; rather, as a condition for the use of logical inferences and the performance of logical operations, something must already be given to us in our faculty of representation, certain extralogical concrete objects that are intuitively present as immediate experience prior to all thought.”

Modeling theory asserts that physical and mathematical intuitions are merely two different ways to relate products of imagination to the external world. **Physical intuition** matches structure in mental models with structure in physical systems. **Mathematical intuition** matches mental structure with

symbolic structure. Thus, structure in imagination is common ground for both physical and mathematical intuition.

Kant reasoned in much the same way. He also took the physics and mathematics of his day as given and asked what makes them so special. His analysis is cogent even today, so key points are worth reconsidering. He began by identifying *construction in intuition* as *a means* for acquiring certain geometrical knowledge:

“Thus we think of a triangle as an **object**, in that we are conscious of the combination of the straight lines according to a **rule** by which such an **intuition** can always be **represented**. . . This representation of a universal procedure of imagination in providing an image for a concept, I entitle the **schema of this concept**.”

Kant did not stop there. Like any good scientist he anticipated objections to his hypothesis. Specifically, he noted that his intuitive image of a triangle is always a *particular triangle*. How, he asks, can construction of a concept by means of a single figure “express universal validity for all possible intuitions which fall under the same concept?” This is the general epistemological *problem of universality* for the case of Kant’s theory of geometrical proof. Kant’s notion of geometrical proof is by construction of figures, and he argues that such proofs have universal validity as long as the figures are “determined by certain universal conditions of construction.” In other words, construction in intuition is a *rule-governed activity* that makes it possible for geometry to discern “the universal in the particular.”

Kant’s argument is often dismissed because it led him to conclude that Euclidean geometry is *certain a priori*. But that is a red herring! Because we now know that non-Euclidean geometry can be associated with the same intuitive construction simply by changing the rules assigned to it. His essential point is that **mathematical inference from intuition is governed by subsumption under rules**. As mathematician Saunders MacLane [19] asserts, “Mathematics is not concerned with reality but with rule.”

## V. RULES AND TOOLS FOR THINKING AND DOING

Science and technology have coevolved with language and mathematics. The evolution is driven by invention of tools with increasing sophistication and power to shape and understand the physical world. The tools of science are of two kinds: instruments for detecting reproducible patterns in the material world, and symbolic systems to represent those patterns for contemplation in the mind.

The detection of patterns in nature began with direct observation using human sensory apparatus. Then the human perceptual range was extended by scientific instruments, such as telescopes and microscopes. Finally, Technology has replaced human sensory detectors with more sensitive instruments, and the data is processed by computers with no role for humans except to interpret the final results; even there the results may be fed to a robot to take action with no human participation at all.

Tool development in the cognitive domain began with the natural languages in spoken and then written form. Considering their *ad hoc* evolution, the coherence, flexibility and subtlety of the natural languages is truly astounding. More deliberate and systematic development of symbolic tools came with the emergence of science and mathematics. The next stage of enhancing human cognitive powers with computer tools is just beginning.

While science is a search for structure, mathematics is the science of structure. Every science develops specialized modeling tools to represent the structure it investigates. Witness the rich system of diagrams that chemists have developed to characterize atomic and molecular structure. Ultimately, though, these diagrams provide grist for mathematical models of greater explanatory power. What accounts for the ubiquitous applicability of mathematics to science? An answer is suggested by considering the coevolution of mathematics and physics from the perspective of modeling theory.

Tools of technology provide an obvious index of progress in human civilization, because their results are so tangible. A more subtle and informative index is the development of language and mathematics, which provide us with **tools to think with!** Though spoken language reaches back more than 150,000 years, written language is barely 5,000 years old, and printed books less than 700. With the invention of calculus by Newton and Leibniz in the seventeenth century, the development of mathematics and physics has accelerated to this day. *Kant put his finger on the source of this stunning revolution: the use of rules to harness the powers of human intuition.*

Precision in science requires precise standards and conventions, in short, precise rules in both empirical and theoretical domains. The coevolution of physics and mathematics has been driven by invention and application of new rules to shape human intuition and model the physical world. The tools of technology from simple hand tools to complex machines were obviously invented. Likewise the tools of mathematics were invented, not discovered; though it may be said that theorems derived from structures built with those tools are discovered.

The vicissitudes of mathematical invention are evident in the motley assortment of mathematical tools used by physicists today, from vectors and matrices to tensors, spinors and differential forms. Far from exhibiting the unity and richness of mathematics, these “tool kits” contribute redundancy, inefficiency and obscurity [21]. A more coherent and powerful system of mathematical tools explicitly designed to integrate algebra and geometry is already well developed with a huge range of applications. Few physicists and mathematicians know about it, so an introduction to the literature is appropriate here, especially as it supports the present thesis of *mathematics by design!*

Kant himself contributed to the rule-based developments in mathematics. He was the first to formulate the abstract commutative and associative rules for addition (published by his mathematician friend Johann Schultz). Within the next century, Hermann Grassmann and W. K. Clifford provided foundations for integrating geometry, algebra and calculus into a *universal geometric calculus* that is developing with renewed vigor today. A history of *geometric algebra and calculus* is given in [20]. Its implications for the design of mathematical tools to simplify and unify the physics and mathematics curriculum are discussed in [21]. Extension to modeling spacetime, quantum mechanics and gauge theory gravity is given in [22,23].

To the question: “*What is man?*”

Aristotle answered: “*Man is a rational animal.*”

Anthropologists observe: “*Man is a tool-making animal.*”

Modeling Theory suggests: “*Man is a modeling animal!*”

**Homo modelus!**



## REFERENCES

- [1] I. Kant (1781, 1787) *Critique of Pure Reason* (Palgrave Macmillan, 2003). German editions translated by Norman Kemp Smith (1929).
- [2] D. Hestenes, Modeling Theory for Math and Science Education, In R. Lesh, P. Galbraith, C. Hines, A. Hurford (eds.) *Modeling Students' Mathematical Competencies* (New York: Springer, 2010). <<http://geocalc.clas.asu.edu/html/Modeling.html>>
- [3] D. Hestenes, Notes for a Modeling Theory of Science, Cognition and Instruction, In E. van den Berg, A. Ellermeijer & O. Slooten (eds.) *Modelling in Physics and Physics Education*, (U. Amsterdam 2008).
- [4] D. Hestenes, Toward a Modeling Theory of Physics Instruction, *Am. J. Phys.* **55**: 440-454 (1987).
- [5] D. Hestenes, Modeling Games in the Newtonian World, *Am. J. Phys.* **60**: 732-748 (1992).
- [6] D. Hestenes, M. Wells, and G. Swackhamer, Force Concept Inventory, *The Physics Teacher* **30**: 141-158 (1992).
- [7] I. Halloun and D. Hestenes, Common Sense Concepts about Motion, *Am. J. Phys.* **53**, 1056-1065 (1985).
- [8] A. diSessa, Toward an Epistemology of Physics, *Cognition & Instruction* **10**: 105-225 (1993).
- [9] R. Langacker, *Cognitive Grammar*, (Oxford U. Press: New York, 2008)
- [10] S. Goldman, A. Graesser, & P. van den Broek, *Narrative Comprehension, Causality and Coherence*. (Lawrence Earlbaum: London, 2001).
- [11] M. Tomasello, *Origins of Human Communication* (MIT Press: Cambridge, 2008).
- [12] P. Johnson-Laird, *Mental Models* (Harvard U.P.: Cambridge, 1983).
- [13] Bryant, Tversky & Lanca (2000), Retrieving Spatial Relations from Observation and Memory. In van der Zee & Nikanne (eds.), *Cognitive Interfaces* (Oxford U. Press, Oxford).
- [14] W. Croft & D. Cruse, *Cognitive Linguistics*. (Cambridge U Press: Cambridge. 2004).
- [15] G. Lakoff & R. Núñez, *Where mathematics comes from*. (Basic Books: New York, 2000).
- [16] S. Dehaene. Origins of Mathematical Intuitions, The Case of Arithmetic. *Ann. N.Y. Acad. Sci.* **1156**: 232–259 (2009).
- [17] R. Giere, *SCIENCE without LAWS* (U. Chicago Press: Chicago, 1999).
- [18] D. Gentner, K. Holyoak & B. Kokinov (Eds.) *The Analogical Mind* (MIT Press: Cambridge, 2001)
- [19] S. MacLane, *Mathematics, Form and Function*. (Springer-Verlag: New York, 1986).
- [20] D. Hestenes, Grassmann's Legacy. In H-J. Petsche, A. Lewis, J. Liesen, S. Russ (Eds.) *From Past to Future: Grassmann's Work in Context* (Birkhäuser: Berlin, 2011). <<http://geocalc.clas.asu.edu/html/Overview.html>>
- [21] D. Hestenes, Oersted Medal Lecture 2002: Reforming the Mathematical Language of Physics. *Am. J. Phys.* **71**: 104-121 (2003). <<http://geocalc.clas.asu.edu/html/Overview.html>>
- [22] D. Hestenes, Spacetime Physics with Geometric Algebra, *Am. J. Phys.*: **71**: 691-714 (2003).
- [23] D. Hestenes, Gauge Theory Gravity with Geometric Calculus, *Foundations of Physics* **36**, 903-970 (2005).

## **EDUARDO JOSÉ BAYRO-CORROCHANO**

### *Geometric Computing for Cybernetics*

#### **Summary**

In this talk we present an advanced new mathematical framework, the conformal geometric algebra for applications in computer vision, graphics engineering, control engineering, robotics and machine learning. We will show that this mathematical system keeps our intuitions and insight of the geometry of the problem at hand and it helps us to reduce considerably the computational burden of the problems.

Surprisingly, as opposite to the standard projective geometry, in conformal geometric algebra we can deal simultaneously with incidence algebra operations (meet and join) and conformal transformations represented effectively using spinors (a kind of quaternions, dual quaternions, etc). In this regard this framework appears promising for dealing with representation, kinematics, dynamics, projective geometry and Riemann differential geometry problems without the need to abandon the mathematical system (as current approaches). We present some real tasks of perception and action treated in a very elegant and efficient way: body–eye calibration, 3D reconstruction and robot navigation and visually guided 3D object grasping, walking pattern generation and biped walking control making use of the directed distance and intersections of lines, planes and spheres both involving conformal transformations. For tracking, we use the Motor (dual quaternion) extended Kalman filter and for control problems we reformulate the differential geometry and the Jacobian based control rule for 6 D. O. F. robot arms using conformal geometric algebra. At the final part of the talk we will also present the design of an applications of geometric neural networks, geometric NGAS and the Multi-vector Support Vector Machines (generalization of MIMO SVMs) useful for learning in visual guided robotics and medical robotics. We will comment our current work on geometric spike neurons to system identification and control for humanoid behavior control.

The lecturer believes that the framework of geometric algebra can be in general of great advantage for applications in image processing, graphics engineering, stereo vision, compressed sensing, deep-learning, range data, laser, omnidirectional, laser and odometry based robotic systems (robot manipulators, mobile and humanoids), kinematics and dynamics of robot mechanisms, cognitive robotics, and advanced nonlinear control techniques.



## STEPHEN J. SANGWINE

### *MATLAB toolbox for Clifford Algebras*

#### **Summary**

MATLAB is a powerful numerical computing environment oriented towards manipulation of matrices and vectors (in the linear algebra sense, that is arrays of numbers).

The author (with Nicolas le Bihan) developed a quaternion toolbox for MATLAB in 2005 (QTFM), subsequently extended, and downloaded from Sourceforge over 10,000 times. This toolbox includes Fourier transforms, matrix decompositions including LU, QR, SVD, and the eigenvalue decomposition, as well as operations for signal and image processing (one of the original aims of the toolbox).

In 2013 development started on a Clifford toolbox along similar lines. The new toolbox can be initialised to any Clifford algebra with signature  $(p, q, r)$  where  $p, q, r$  are the numbers of basis elements that square to  $+1, -1, 0$  respectively. A very powerful feature of the toolbox is the ability to iterate over Clifford algebras by re-initialising the toolbox signature inside a loop. This does not destroy existing variables, but checks are performed to ensure that variables are not used unless their signature matches the current signature. It is thus possible to explore whether particular algorithms will work in all algebras, or only in a subset. As with the QTFM toolbox, it is designed to work just like Matlab, so that the powerful colon notation and square bracket concatenation works with multivector arrays. Overloading of MATLAB function names means that operations on multivectors are carried out with the same function names as operations on real or complex arrays.

The fundamental structure of the toolbox is now complete, and some steps have started on implemented high-level algorithms, beginning with the LU decomposition and the matrix inverse. The toolbox inherently treats matrices of multivectors, and the components of the multivectors may be complex (this follows from the fact that ordinary numbers in MATLAB are by default complex). Since the underlying numeric operations are implemented by MATLAB, calculations are vectorised (that is implemented using processor-level parallelism), giving fast performance. Care has been taken to represent multivectors internally so

that zero components are stored as empty arrays – avoiding needless multiplications by zero and wasted storage (important if large matrices are manipulated in a high-dimensional algebra).

The talk will include live demonstrations of the toolbox capabilities.

## LEO DORST

### *Projective Transformations as Versors*

#### Summary

Whenever I introduce people in machine vision and computer graphics to the wonders of conformal geometric algebra (CGA) to describe the Euclidean rigid body motions and similarities, they always ask: ‘Can you also do projective transformations?’ In those fields, projective transformations expressed in  $4 \times 4$  homogeneous coordinate matrices are the standard, giving a nice integration of pinhole projective imaging and Euclidean motions. The naturally transforming conics are the usual primitives in many geometric modeling packages, and used to understand the real world data. CGA, with its spheres and circles, feels much too restrictive to them.

Our answer should be ‘*Yes, we can!*’, because we are of course convinced that geometric algebra can do all of geometry, elegantly, compactly, and advantageously. So, it is time to get specific, or we will lose credibility. Currently, there appear to be two proposals:

- Goldman [3] uses  $\mathbb{R}^{4,4}$  to transcribe the  $4 \times 4$  matrices into rotor form, following the general framework of [1]. He gives the bivector generators of all standard projective transformations, but has to employ some unusual constructions to incorporate projections. The 28 degrees of freedom in the rotors are not all used, and the geometrical meaning of the blades and in this model is unclear.
- One can explore the accidental group isomorphism between the groups  $SL(4)$  and  $Spin(3;3)$  to represent the homogeneous coordinate matrices. Klawitter [4] has recently shown how to convert a versor from  $\mathbb{R}^{3,3}$  to a  $4 \times 4$  matrix, and vice versa. His approach is coordinate based, and the bivector generators of standard projective transformations are unfortunately not made explicit. He points out that the blades are line complexes (and thus unfortunately not conics).

My presentation exposes the  $\mathbb{R}^{3,3}$ -bivectors of the canonical projective transformations, thus bringing the  $\mathbb{R}^{3,3}$  model of Klawitter closer to the practical flavor exhibited Goldman’s work on  $\mathbb{R}^{4,4}$ . While doing so, we develop some geometrical and contextual insights in the  $\mathbb{R}^{3,3}$  model (the ‘space of lines’), showing how the cross ratio and duality are represented [2]. We briefly treat the

geometrical meaning of its blades, and how this affects modeling reality. Conics do appear briefly and strangely.

As I started to investigate  $\mathbb{R}^{3,3}$ , I was hoping that it might give us an ‘oriented projective geometry’ [5] in which we can compute consistently with directed lines, a capability that would be very useful in machine vision and computer graphics, but currently lacking. I will show why this unfortunately fails: the odd versors that would make reflections explicit actually do something else (and rather useless).

Summarizing, neither  $\mathbb{R}^{4,4}$  nor  $\mathbb{R}^{3,3}$  are quite what we would like to present to the practitioners.

More work is needed, and soon!

## References

- [1] C. Doran, D. Hestenes, F. Sommen, and N. Van Acker. Lie groups as spin groups. *J. Math. Phys.*, 34(8):3642–3669, 1993.
- [2] Leo Dorst. Spin(3,3) bivector characterization of elementary 3d projective transformations (working title), 2015. In preparation for submission to AACA.
- [3] R. Goldman and S. Mann.  $R^{4,4}$  as a computational framework for 3-dimensional computer graphics. *Advances in Applied Clifford Algebra*, 2014. Approved, to appear 2014.
- [4] D. Klawitter. A Clifford algebraic approach to line geometry. *Advances in Applied Clifford Algebra*, 2014. arXiv:1311.0131v4.
- [5] Jorge Stolfi. *Oriented Projective Geometry*. Academic Press, 1991.

## **ANTHONY LASENBY**

*Geometric Algebra as a unifying language for Physics and Engineering  
and its use in the study of Gravity*

### **Summary**

The aim of the first part of this lecture is to give an overview of the ability of Geometric Algebra (GA) to provide a unifying mathematical language for Physics and Engineering. Examples from several different fields will be given, including electromagnetism, rigid body dynamics, quantum mechanics and signal processing, and it will be shown how the availability of this common language aids in being able to work on advanced problems in multiple areas, emphasising common aspects and aiding geometrical intuition in each.

Then in the second half of the lecture, we specialise to the area of gauge theories and gravity, with the aim of showing that when equipped with an understanding of GA, hitherto apparently difficult subjects such as General Relativity, can become quickly accessible using the same GA techniques needed in engineering applications. In particular, we show how General Relativity can be regarded as a gauge theory in flat space, and suggest how an extension of the symmetries which are gauged can lead to a novel theory of gravity, with potential implications across a wide range of phenomena.





## DAVID EELBODE, ECKHARD HITZER

### *Fourier Transformations in Conformal Geometric Algebra*

#### Summary

Conformal geometric algebra is very popular in applications of geometric algebra [1]. In recent years research and application of Clifford Fourier transformations is also flourishing [2]. Therefore the natural question is asked for the appropriate form of Fourier transformations in conformal geometric algebra. It seems suitable in a first step to select the bivector square roots of minus one [3,4] in the kernel construction. Moreover the question is for the proper scalar function of the position and frequency vectors in an exponential kernel. We basically suggest to use the inner product of conformal points, and investigate the consequences of this choice. Interesting strong relations to linear canonical transformations [5], such as the Fresnel transformation are explained. Then several standard properties of the thus established conformal geometric algebra Fourier transform (confFT) are shown. We further intend to discuss eigenfunctions, carefully taking the non-commutativity of conformal object signal functions and the confFT kernel into account.

#### References

- [1] E. Hitzer, T. Nitta, Y. Kuroe, *Applications of Clifford's Geometric Algebra*, Adv. Appl. Clifford Alg., Vol. 23, Online First, March 2013, (2013), pp. 377-404. DOI: 10.1007/s00006-013-0378-4. Preprint: <http://arxiv.org/abs/1305.5663>.
- [2] E. Hitzer, S.J. Sangwine (eds.), "Quaternion and Clifford Fourier transforms and wavelets", Trends in Mathematics 27, Birkhauser, Basel, 2013.
- [3] E. Hitzer, J. Helmstetter, R. Ablamowicz, *Square roots of  $-1$  in real Clifford algebras*, in E. Hitzer, S.J. Sangwine (eds.), "Quaternion and Clifford Fourier transforms and wavelets", Trends in Mathematics 27, Birkhauser, Basel, 2013, pp. 123-153. DOI: 10.1007/978-3-0348-0603-9\_7. Preprints: <http://arxiv.org/abs/1204.4576>, [http://www.tntech.edu/files/math/reports/TR\\_2012\\_3.pdf](http://www.tntech.edu/files/math/reports/TR_2012_3.pdf). First published in K. Guerlebeck (ed.), Proc. of the 9th Int. Conf. on Clifford Algebras and their Applications, (2011).

[4] E. Hitzer, *The quest for conformal geometric algebra Fourier transformations*, In T. Simos, G. Psihoyios and C. Tsitouras (eds.), Numerical Analysis and Applied Mathematics ICNAAM 2013, AIP Conf. Proc. 1558, pp. 30-33 (2013). DOI: 10.1063/1.4825413, Preprint: <http://vixra.org/abs/1310.0248>

[5] K. B. Wolf, *Integral Transforms in Science and Engineering*, Mathematical Concepts and Methods in Science and Engineering, Vol. 11, Springer, 1979, chapter 9. Online text: <http://www.fis.unam.mx/~bwolf/integral.html>

**RAFAEL F. LEÃO, WALDYR A. RODRIGUES JR., SAMUEL A. WAINER**

*Concept of Lie Derivative of Spinor Fields. A Geometric Motivated Approach*

## **Summary**

Using the Clifford bundle  $(\mathcal{C}\ell(M, \mathbf{g}))$  and spin-Clifford bundle  $(\mathcal{C}\ell_{\text{Spin}_{1,3}^e}(M, \mathbf{g}))$  formalisms, which permit to give a meaningful representative of a Dirac-Hestenes spinor field (even section of  $\mathcal{C}\ell_{\text{Spin}_{1,3}^e}(M, \mathbf{g})$ ) in the Clifford bundle, in this lecture we give a geometrically motivated definition for the Lie derivative of spinor fields in a Lorentzian structure  $(M, \mathbf{g})$ , where  $M$  is a manifold such that  $\dim M = 4$  and  $\mathbf{g}$  is Lorentzian of signature  $(1, 3)$ . Our Lie derivative, called the spinor Lie derivative (and denoted  $\overset{s}{\mathcal{L}}_{\xi}$ ) is given by nice formulas when applied to Clifford and spinor fields, and moreover  $\overset{s}{\mathcal{L}}_{\xi}\mathbf{g} = 0$  for any vector field  $\xi$ . We compare our definitions and results with the many others appearing in the literature on the subject.



## PIERRE ANGLÈS

### *Geometric algebras and spinors*

(Dedicated to the memories of Pertti Lounesto, Jaime Keller and Artibano Micali)

In these days the angel of topology and the devil of abstract algebra fight for the soul of each individual mathematical domain.

Hermann Weyl, *Invariants*, 1939, *Gesammelte Werke*, Band III, page 681.

### **Summary**

The lecture will be divided into three parts.

The first one is devoted to the structure of the Clifford algebra and will deal with all the main mathematical content of Clifford algebras.

After a short presentation of the history of Clifford algebras, we will analyze the concept of an algebra. As pointed out by Nicolas Bourbaki, W.K. Clifford introduced the algebras known as Clifford algebras and proved that they are tensor products of quaternion algebras or of quaternion algebras by a quad-ratic extension.

Elementary properties of quaternion algebras will be recalled. Basic definitions and properties of Clifford algebras will be given. The construction of a basis will be studied. Standard classical properties will be given. The covering groups for the standard orthogonal group, the special orthogonal group, the connected component of the identity of the orthogonal group of a regular quadratic finite dimensional space over  $\mathbb{R}$  or  $\mathbb{C}$  will be given. Clifford algebras for standard pseudo Euclidean spaces will be studied.

The second part deals with spinors in contemporary physics. We give the terminology and define Dirac Spinors, Weyl Spinors, Majorana Spinors, Weyl Majorana Spinors. The language of physicists concerning spin groups, spin bundles and spin connections will be studied. The links between spinors and the fundamental interactions will be analyzed.

The third part is devoted to a few comments on Spinors in Minkowski space-time. Spinorial coordinates, the Weyl Spinor space, dotted spinors, spinor tensors, Dirac spinors, Dirac Matrices, Chirality, Charge conjugation and Majorana Spinors are successively presented.

The conclusion will be a look to the algebraic foundation of the theory of twistors and a short analysis on the epistemological level of the importance of Geometric Algebras.

## **CHRIS DORAN**

*Game theory: From Black Holes to Battlefield 4*

### **Summary**

In 2005 Dr Doran cut short his academic career studying applications of geometry algebra to take on the challenge of building a company. The initial hope was to commercialise geometric algebra in some form, though that proved to be challenging. Instead the company he formed, Geomerics, focused attention on solving problems in real-time graphics with emphasis on lighting. Geomerics technology is now used in some of the most popular games on release, as well as being the default technology in the most popular game engine in use today. In late 2013 Geomerics was acquired by the chip-design company ARM, completing the journey from spin-out through growth to exit.

In this talk we discuss the practicalities of building a company based on mathematical know-how; the challenges of turning this into practical products; considerations around funding and investors; and working with a team of commercial software engineers. The talk covers some of the recent history of the games industry and highlights some of the key topics in graphics research today where geometric algebra should have an impact. The talk ends with a discussion of what the geometric algebra community needs to do to increase its impact in the wider world, and how we can take advantage of modern media techniques to achieve this.





# COORDINATE FREE INTEGRALS IN GEOMETRIC CALCULUS

T. Alho<sup>a</sup>

<sup>a</sup> Science Institute

University of Iceland, Reykjavik, Iceland  
alho@hi.is [presenter, corresponding]

ABSTRACT. We introduce a method for evaluating integrals in geometric calculus without introducing coordinates, based on using the fundamental theorem of calculus repeatedly and cutting the resulting manifolds so as to guarantee the existence of a boundary and an antiderivative at each step. The method is a direct generalization of the usual method of integration on  $\mathbb{R}$ . It may lead to both practical applications and help unveil new connections to various fields of mathematics.

## 1. INTRODUCTION

One of the main selling points for Geometric Algebra and Calculus [1–10] is the claim that it allows carrying out computations in inner product spaces without resorting to coordinates. Indeed, there exist well developed methods for simplifying algebraic statements and solving equations, computing the vector derivative and the multivector derivative, and finally for developing a theory of directed integration, all in a coordinate free manner. However, when it comes to actually computing the value of an integral, a coordinate system is invariably introduced [4, 5, 11]. This paper takes key steps towards remedying this.

In calculus on  $\mathbb{R}$ , definite integration is usually carried out by finding an antiderivative or an indefinite integral of the function to be integrated, and then applying the fundamental theorem of calculus to obtain the desired definite integral. The fundamental theorem of geometric calculus [4, 12], a version of which can be expressed as

$$(1) \quad \int_M d^m x \partial_M F = \int_{\partial M} d^{m-1} x F,$$

where  $\partial_M$  is the vector derivative on the manifold  $M$ , provides a tool to do the same in any number of dimensions, for functions with values in the geometric algebra.

Let us briefly recall the main elements in (1). In a directed integral, the integration measure  $d^m x$  is an  $m$ -vector valued element of the tangent algebra of  $M$ , analogous to the volume form in the theory of differential forms. When  $M$  is embedded in a higher dimensional manifold, the directed integral therefore carries more information than the usual integral with a scalar valued measure, including information about the orientation of the manifold that the integral is over, weighted by the integrand.

The vector derivative on a manifold,  $\partial_M$ , is a vector-valued derivative operator, and so in addition to taking derivatives it acts algebraically as a vector. On a manifold, it only considers differences along the manifold, but note that the result of the derivation can take values in the full geometric algebra, so it is distinct from the covariant derivative. In coordinates, one can define  $\partial_M = \sum_i p_{M_x}(e^i) \frac{\partial}{\partial x^i}$ , where  $p_{M_x}(a)$  is the projection of the vector  $a$  to the tangent space of the manifold at point  $x$ . In what follows, we usually suppress  $x$  in the notation, and also  $M$  where the manifold is clear from context.

For example, assume we are integrating a function  $f(x)$  over a  $d$ -dimensional subset  $M$  of  $\mathbb{R}^d$ , which is sufficiently smooth to satisfy the assumptions of the fundamental theorem and has a finite number of connected components. The first step is to find an

antiderivative  $F_1(x)$  of  $f(x)$ , i.e.  $\partial_M F_1(x) = f(x)$  for all  $x \in M$ . Now we get, according to (1), an integral over the  $d - 1$  dimensional boundary  $\partial M$  of  $M$ . We'd like to use the fundamental theorem again, and so we look for an antiderivative  $F_2(x)$  of  $F_1(x)$  on the boundary  $\partial M$  with respect to the derivative  $\partial_{\partial M}$  on the boundary. Given an antiderivative  $F_2(x)$  we run into the problem that the boundary of the boundary of a set is always empty. We move forward by making an incision of the boundary, i.e. we choose a set  $E_2$  such that  $\partial M \setminus E_2$  has a smooth boundary  $\partial_{E_2} \partial M := \partial(\partial M \setminus E_2)$ , and  $\text{vol}(E_2) < \varepsilon_2$ . Now the integral

$$(2) \quad \int_{\partial_{E_2} \partial M} d^{d-2} x F_2(x)$$

differs from our desired integral by at most  $\text{vol}(E_2) \sup_{x \in E_2} \|F_2(x)\|$ . Notice that we have to choose  $F_2$  and  $E_2$  such that  $F_2$  is continuous in  $\partial M \setminus E_2$ , in order to justify our use of the fundamental theorem. This requirement is actually crucial, since any finite value of the integral as we shrink  $\varepsilon$  to zero comes from what are essentially branch cut discontinuities in the antiderivative. Indeed, due to the presence of branch cuts, we could not have found  $F_2(x)$  on the whole manifold, giving a second reason why the incision is necessary.

We then simply repeat the same construction  $d$  times, at each step requiring that for incision  $E_n$  the volume  $\text{vol}(E_n) < \varepsilon_n$  and that each antiderivative is continuous in the integration set. In the final step, the integration will be over a one-dimensional manifold, which simply has a finite number of points as a boundary, leaving us with a finite sum of values of the  $d$ th antiderivative. Then as we let all of the  $\varepsilon_n$ 's go to zero, we get our final result.

As will be shown via examples, this method allows computing integrals without invoking a coordinate system. However, we will find in all practical examples that we do need to invoke reference vectors or multivectors, and the expectation is indeed that this will turn out to be generic, as the reference multivectors provide a mechanism for choosing a specific antiderivative.

We expect that this method of integration will open up new possibilities in analyzing any integral or differential systems in  $n$ -dimensions. This includes the theory of partial differential equations<sup>1</sup>, numerical estimation methods for integrals, and also connections to algebraic geometry, since it becomes possible, at least in principle, to handle all aspects of surfaces expressible as algebraic equations in a coordinate independent manner.

In this paper, we first prove that when the requisite antiderivatives and submanifolds exist and satisfy a number of reasonable properties, the above construction indeed gives the desired result. We then give some examples of elementary integrals worked out according to the method. Finally, we elaborate on possible implications and directions for further research.

## 2. INTEGRATION BY ANTIDERIVATIVES

Let us briefly recall some definitions and establish some notation.

Our basic notation follows that used by [13]. We use the left- and right contractions  $\lfloor$  and  $\rfloor$  instead of the single dot product, our scalar product contains the reverse,  $A * B = \langle A \tilde{B} \rangle$ , and our dual is a right multiplication by the pseudoscalar. The norm on a geometric algebra is defined as  $\|A\|^2 = A * A = \langle A \tilde{A} \rangle$ . Although our method generalizes

<sup>1</sup>When such equations are expressed in geometric calculus, we follow [4] in considering this a misnomer, and prefer the term vector differential equation.

easily to the case of mixed signatures, we will for simplicity consider here only spaces where the inner product is positive definite, and so the multivector norm defines a well-behaved concept of convergence.

Since the directed integral of a multivector function can always be expanded in a multivector basis in terms of scalar coefficient functions, we can import the concept of integrability from scalar valued integrals:

**Definition 1.** A function  $f : M \rightarrow \mathcal{G}_M(x)$  is L-integrable in the sense of the directed integral on  $M$  if each of the scalar functions  $a^I(x) * (\frac{d^m x}{\|d^m x\|} f(x))$  are L-integrable on  $M$  with the measure  $\|d^m x\|$ , where  $a^I(x)$  is a multivector basis [4] of  $\mathcal{G}_M(x)$ , and L is a definition of integrability for scalar valued functions, such as Riemann or Lebesgue.

In what follows, we will simply refer to integrability, and by that mean integrability in the sense of the directed integral based on a suitable definition of scalar integrability. For all the theorems and examples in this paper, the Riemann integral will be sufficient.

We write  $\text{vol}(M)$  for the volume of a manifold in the appropriate dimension, i.e. for  $\dim(M) = 2$  the volume is the area, and so on.

For completeness, let us recall the definition of the tangent algebra and the vector derivative [4]:

**Definition 2.** Let  $M$  be a Euclidean vector manifold [4]. Then the *tangent algebra* of  $M$  at  $x \in M$ , denoted by  $\mathcal{G}_M(x)$  is the geometric algebra, i.e. real Clifford algebra, generated by the tangent space  $T_x M$ .

**Definition 3.** Given a vector derivative  $\partial_M$  on an orientable vector manifold  $M$  and an orientable submanifold  $N \subseteq M$  and a unit pseudoscalar of  $N$ ,  $I_N(x) \in \mathcal{G}_M(x)$ , for each  $x \in N$ , the projected derivative  $\partial_N$  is given by [4]

$$(3) \quad \partial_N = p_{N_x}(\partial) = \sum_i p_{N_x}(e_i) e_i \cdot \partial_M = \sum_i p_{N_x}(e_i) \frac{\partial}{\partial x_i},$$

where  $p_{N_x}(a) = I_N(x)^{-1}(I_N(x) \lrcorner a)$  is the projection of a vector  $a$  to the tangent algebra of the manifold  $N$  at  $x \in N$ , and  $e_i$  is a basis of the tangent space  $T_x M$ .

Note that the partial derivative operator does not operate on the pseudoscalar  $I_N(x)$ , and also that the projected derivative can take values in the full tangent algebra of  $M$ , not just  $N$ . Then one version of the fundamental theorem of calculus can be expressed as [4, 12]

**Theorem 1** (Fundamental theorem of calculus). *Let  $M$  be an oriented  $m$ -dimensional vector manifold with a boundary  $\partial M$  that is a vector manifold,  $f$  a differentiable function  $f : M \rightarrow \mathcal{G}_M(x)$ , and  $\partial_M$  the vector derivative on  $M$ . Then*

$$(4) \quad \int_M d^m x \partial_M f(x) = \int_{\partial M} d^{m-1} x f(x),$$

We will only consider this form, with the pseudoscalar measure to the left of the integrand, in this paper. Let us prove a simple lemma:

**Lemma 2.** *Let  $M$  be an oriented  $m$ -dimensional vector manifold and  $f : M \rightarrow \mathcal{G}_M(x)$  be an integrable function from the manifold to the algebra. Given a bounded submanifold  $E \subset M$  such that  $f$  is bounded in  $E$ , then*

$$(5) \quad \left\| \int_{M \setminus E} d^m x f(x) - \int_M d^m x f(x) \right\| \leq \text{vol}(E) \sup_{x \in E} \|f(x)\|$$

*Proof.* Direct calculation using the triangle inequality:

$$(6) \quad \left\| \int_{M \setminus E} \mathbf{d}^n x f(x) - \int_M \mathbf{d}^n x f(x) \right\| = \left\| \int_E \mathbf{d}^n x f(x) \right\| \leq \int_E \|\mathbf{d}^n x f(x)\| \\ = \int_E \|\mathbf{d}^n x\| \|f(x)\| \leq \int_E \|\mathbf{d}^n x\| \sup_{x \in E} \|f(x)\| = \text{vol}(E) \sup_{x \in E} \|f(x)\|.$$

Note that the supremum exists and is finite since  $E$  is bounded and  $f$  is bounded on  $E$ .  $\square$

The point of Lemma 2 is that it allows us to cut out a part of the manifold in order to guarantee that it has a boundary, and still keep control of the error we're making. Also, we will find out that usually functions on manifolds without boundary do not have single valued antiderivatives, and the lemma allows us to exclude a branch cut, since the existence of the antiderivative is only necessary on the part of the manifold that is not cut.

**Definition 4.** Let  $M$  be a vector manifold and  $f : M \rightarrow \mathcal{G}_M(x)$  be a function on the manifold. If  $f$  has an antiderivative  $F$  on  $M$ , we write  $F =: \partial_M^{-1} f$ . If  $\partial_M^{-1} f$  again has an antiderivative on  $N \subseteq M$ , we denote that by  $\partial_{MN}^{-2}$ , and in general we write  $\partial_{M_1 M_2 \dots M_n}^{-n} f$  for the  $n$ th antiderivative of  $f$  on the manifold  $M_n$ , if it exists, with  $M_1 \subseteq M_2 \subseteq \dots \subseteq M_n$ .

Note that due to the projection operator in the derivative on a manifold, the antiderivative in general depends on the manifold in which it is defined. In other words an antiderivative on a submanifold is not necessarily just the restriction of some antiderivative on the full manifold. Also, in the above definition the antiderivative is ambiguous, so when using the notation we have to either define how to choose a specific antiderivative, or show that our results don't depend on the choice.

Now we get to the main result:

**Theorem 3.** *Let  $M$  be an  $m$ -dimensional orientable vector manifold, and  $f : M \rightarrow \mathcal{G}_M(x)$  an integrable function. If there exists a sequence of orientable manifolds  $N_0 \subset N_1 \subset \dots \subset N_m = M$  and a sequence of bounded sets  $E_i$  such that*

- *if  $\partial N_{i+1} \neq \emptyset$ ,  $N_i = \partial N_{i+1}$ , otherwise  $N_i = \partial(N_{i+1} \setminus E_{i+1})$ , where  $E_{i+1}$  is a bounded set such that the boundary  $\partial(N_{i+1} \setminus E_{i+1})$  is a non-empty vector manifold, and  $\partial_{N_{i+1} \dots N_m}^{-m+i+1} f$  is integrable and bounded on  $E_{i+1}$ .*
- *there exists an antiderivative  $\partial_{N_i \dots N_m}^{-m+i} f$  on  $N_i$ , which is bounded.*
- *$N_0$  is a finite set*

*then the integral of  $f$  over  $M$  can be computed by evaluating the  $n$ th antiderivative on  $N_0$ :*

$$(7) \quad \left\| \int_M \mathbf{d}^n x f(x) - \sum_{x_i \in N_0} \partial_{N_0 \dots N_m}^{-m} s_i f(x_i) \right\| \leq \varepsilon,$$

*where  $\varepsilon = \max_i \sup_{x \in N_i} \|\partial_{N_i \dots N_m}^{-m+i} f(x)\| \sum_i \delta_i$ , and the signs  $s_i \in \{-1, 1\}$  are determined by the orientations of the boundaries at each step.*

Before proving the theorem, we make a few remarks. We basically forced the theorem to be true by sticking all the difficult parts into the assumptions. Note however that the local existence of an antiderivative is guaranteed for a differentiable function [11, 12, 14], and also that the set  $N_0$  is automatically discrete since it's the boundary of a 1-dimensional manifold, and with very mild assumptions on  $M$  the  $E_i$  can be chosen

such that  $N_0$  is a finite set. In essence these assumptions allows us to prove the theorem without getting mixed up in topological complications, and for most practical applications the natural choice of the sets  $N_i$  will anyway fulfill these assumptions, which is why we're not interested in sharpening the theorem at this point.<sup>2</sup>

*Proof of theorem 3.* First note that since the integral of a bounded function over a bounded set is finite, each of the supremums in the expression for  $\varepsilon$  exist. The only part left to prove is the inequality. Using the fundamental theorem, Lemma 2 and the triangle inequality, we first compute

$$\begin{aligned}
 & \left\| \int_M \mathbf{d}^n x f(x) - \sum_{x_i \in N_0} \partial_{N_0 \dots N_m}^{-m} s_i f(x_i) \right\| \\
 &= \left\| \int_M \mathbf{d}^n x f(x) - \int_{N_1 \setminus E_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) \right\| \\
 &= \left\| \int_M \mathbf{d}^n x f(x) - \int_{N_1 \setminus E_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) \right. \\
 &\quad \left. + \int_{N_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) - \int_{N_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) \right\| \\
 &\leq \left\| \int_M \mathbf{d}^n x f(x) - \int_{N_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) \right\| \\
 &\quad + \left\| \int_{N_1 \setminus E_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) - \int_{N_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) \right\| \\
 &\leq \left\| \int_M \mathbf{d}^n x f(x) - \int_{N_1} \mathbf{d}x \partial_{N_1 \dots N_m}^{-m+1} f(x) \right\| \\
 (8) \quad &+ \text{vol}(E_1) \sup_{x \in E_1} \left\| \partial_{N_1 \dots N_m}^{-m+1} f(x) \right\|.
 \end{aligned}$$

Note that since two antiderivatives differ at most by a monogenic function  $\psi$  for which  $\partial_{N_0} \psi(x) = 0$  [4], this result is independent of the choice of antiderivative.

We can then continue using similar steps, each of which produces an approximation error  $\text{vol}(E_i) \sup_{x \in E_i} \left\| \partial_{N_i \dots N_m}^{-m+1} \right\|$ , until finally at the  $m$ 'th step, we get

$$(9) \quad \left\| \int_M \mathbf{d}^n x f(x) - \int_{N_m} \mathbf{d}x \partial_{N_m}^0 f(x) \right\| + \sum_i \text{vol}(E_i) \sup_{x \in E_i} \left\| \partial_{N_i \dots N_m}^{-m+i} f(x) \right\|,$$

where  $N_m = M$  and  $\partial_M^0 f(x)$  is the function itself, and so the integral term is zero. Approximating the the supremums by their maximum concludes the proof.  $\square$

There is a simple corollary;

**Corollary 4.** *Let  $M$  be a vector manifold without a boundary, and  $f : M \rightarrow \mathcal{G}_M(x)$  be a bounded integrable function such that its integral over  $M$  is non-zero. Then any anti-derivative  $\partial_M^{-1} f$  of  $f$  must have a branch cut discontinuity which divides the manifold into at least two parts with non-zero volumes.*

<sup>2</sup>Since in many applications there may be a branch cut that goes to infinity, relaxing the assumption about  $E_i$ 's being bounded would be beneficial, allowing to compute also such integrals when they are finite. This would entail finding a sufficient set of assumptions to guarantee that  $\int_{E_i} \partial_{N_i \dots N_m}^{-m+i} f$  goes to zero as the set  $E_i$  shrinks to zero. In specific cases this should not be difficult.

*Proof.* Assume the opposite, that is, that there exists an antiderivative of  $f$  on the whole of  $M$ . Then we can make a cut according to theorem 3, and let its volume shrink to zero. Since the antiderivative of a bounded function is bounded (which can be seen, for example, by considering the scalar components and applying the usual theorems of integration), this means that the result of the integration is zero. This is a contradiction.  $\square$

In particular, this means that the norm of the volume form on a manifold without boundary cannot have an antiderivative everywhere. Also, since every function on a manifold is an antiderivative of it's own derivative, this corollary may have some links to the hairy ball theorem.

Note also that even though the method is phrased in terms of the directed integral, it is immediately applicable to the usual integral with a scalar measure. We simply write  $\|\mathbf{d}^m x\| f(x) = \mathbf{d}^m x I(x) f(x)$ , where  $I(x)$  is the unit pseudoscalar of the manifold at  $x$ .

In order to do a specific calculation, we find the necessary antiderivatives and sets to cut out by any means we like, and then using theorem 3, we can rest assured that as we let the volume of the incisions  $E_i$  go to zero we get the exact value of the integral. Note that since the errors are additive, the order of the limits for the various sets doesn't matter (unless their construction dictates a specific order). Of course, we have only proven that if this construction can be made, then we can do the coordinate free integral. Let us next present some examples to show that such constructions indeed do exist.

### 3. EXAMPLES

Next we compute examples of applying this method of integration. Since these quite trivial examples already show many of the features we expect to encounter in more generic cases, we work them out in detail. The algebra and rules for computing derivatives needed in this section are contained, for example, in [4, 5, 13, 15].

**3.1. The area of a disk.** As the first example of application of the method, we calculate the area of a disk of radius  $r$  in  $\mathbb{R}^2$ . The integral we intend to compute is

$$(10) \quad A_{B_r} = \int_{B_r} \mathbf{d}^2 x,$$

where  $B_r = \{x \in \mathbb{R}^2 : \|x\| < r\}$ . Note that since the directed volume element  $\mathbf{d}^2 x$  is a bivector, we expect to get the result as a bivector. The first step is to find the antiderivative of the constant function 1. This is by inspection  $\frac{1}{2}x$ , since in general the derivative  $\partial_M x$  is  $mx$ , where  $m$  is the dimension of the manifold [4, 15]. Therefore, the integral is reduced to

$$(11) \quad \frac{1}{2} \int_{S^1} dx x,$$

where  $dx$  is the vector-valued measure on the circle. Now the projection of a vector  $a$  to  $S^1$  at point  $x$  is  $p_{S^1}(a) = x^{-1}(x \wedge a)$ . Intuitively, we see that the integral to calculate measures distance along the circle, i.e. the angle. So does the complex logarithm, and so we are led to the try the function  $\log(xx_0)$ , where  $x_0$  is an arbitrary constant vector in  $\mathcal{G}(\mathbb{R}^2)$ , and since  $xx_0$  is in the even subalgebra of  $\mathcal{G}(\mathbb{R}^2)$  which is isomorphic to the complex numbers with the unit pseudoscalar  $I_2$  acting as the imaginary unit, the logarithm may be defined analogously to the complex logarithm.

In order to compute the projected derivative, we observe that in general  $\partial_M f(x) = \dot{\partial}_M(a] \partial_x) f(x)$ , where  $\partial_x$  is the full vector derivative without the projection, and the overdot denotes that derivative  $\dot{\partial}_M$  acts only on  $a$ . Then, using the chain rule and the

fact that the derivative  $(xx_0) * \partial_z$  reduces to the directed derivative in the direction  $xx_0$  [15], which further reduces to the complex derivative times  $xx_0$  since the direction commutes with the argument, we can further calculate

$$\begin{aligned} \partial_{S^1} \log(xx_0) &= \dot{\partial}_{S^1} (\dot{xx}_0) * \partial_z \log z|_{z=xx_0} = \dot{\partial}_{S^1} (\dot{xx}_0) z^{-1}|_{z=xx_0} \\ (12) \qquad \qquad \qquad &= x_0 \frac{x_0 x}{\|xx_0\|^2} = x^{-1}, \end{aligned}$$

where the overdot limits the scope of the derivative to the dotted objects, as in [4]. We observe that  $\partial_{S^1} x^2 = 0$ , as expected, and therefore deduce immediately that  $\partial_{S^1} \frac{1}{2} x^2 \log(xx_0) = \frac{1}{2} x$ , which is our antiderivative. The boundary of  $S^1$  is empty, but according to our method we cut a small segment, for example the part where  $\frac{|x \cdot x_0|}{\|xx_0\|} > \cos \varepsilon$  which is the part at an angle less than  $\varepsilon$  to  $x_0$ . The complex logarithm function is bounded away from zero, and our incision is bounded, so the assumptions of theorem 3 are satisfied and we calculate

$$(13) \qquad \int_{S^1} dx \frac{1}{2} x = \frac{1}{2} x^2 \log(xx_0)|_{x \in \partial(S^1 \setminus \{x: \|x-x_0\| < \varepsilon\})}.$$

Let us choose the branch of the complex logarithm such that  $\log(xx_0)|_{x=x_0} = \log\|xx_0\| + 0I_2$ . We observe that since the antiderivative must be continuous inside the set where we made the cut, we must then allow the logarithm to approach the value  $\log\|xx_0\| + 2\pi I_2$  on the other side of the cut. Therefore the subtraction results in  $A_{B_r} = \pi r^2 I_2$ , as expected.

**3.2. The volume of a cylinder.** Let us do an example in three dimensions. Let  $M$  be the cylinder defined by the equations

$$(14) \qquad \qquad \qquad I_3 \wedge x = 0$$

$$(15) \qquad \qquad \qquad (\omega \lrcorner x)^2 \leq r^2$$

$$(16) \qquad \qquad \qquad 0 \leq (\omega^{-1}(\omega \wedge x)) \lrcorner (\omega I_3) \leq h,$$

where  $\omega$  is a unit bivector determining the plane orthogonal to the axis of the cylinder,  $r$  and  $h$  are positive real numbers, and  $I_3$  is the pseudoscalar of the 3D space in which the cylinder lies. Eq. (14) guarantees that the cylinder is in the space determined by  $I_3$  and effectively reduces the problem to three dimensions, whereas Eq. (15) sets the radius of the cylinder. Eq. (16) sets the height of the cylinder.

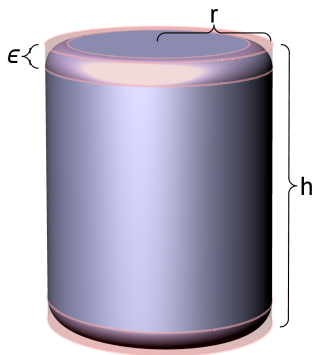


FIGURE 1. The cylinder to be integrated. The red translucent part is the chamfer which we cut away before the first integration. Note that while it's surface does not have a pseudoscalar defined everywhere, the volume itself does. The bottom and top of the cylinder are in the plane defined by the bivector  $\omega$ .

In this case, the cylinder has a sharp edge, which would, after the first integration, contradict the assumption that the pseudoscalar of the surface exists everywhere. Let us therefore this time use lemma 2 to cut a circular chamfer of radius  $\varepsilon$  to the edges, such that the remaining manifold is smooth. The chamfer has a volume proportional to  $\varepsilon^2$ . Note that in all three parts the 3D pseudoscalar is well defined everywhere.

The first integral is again trivial, resulting in  $\frac{1}{3}x$ , since the cylinder is lying in a flat space, and we are integrating the constant function. After this, we again use lemma 2



to ignore the surface of the chamfer, and only concern ourselves with the flat parts of the surface integral. For the surface integral along the sides, we first observe that, with  $f(x) = (\omega \rfloor x)^2$  being the function which's constant value surface  $f(x) = r^2$  defines the side of the cylinder, and given a point  $x$  on the side, the projection of a vector  $a$  to the tangent space is given by

$$(17) \quad p_{\text{side}}(a) = (\partial f(x)I_3)^{-1}(\partial f(x)I_3) \rfloor a = r_\omega(a) + p_{x \rfloor \omega}(a),$$

where

$$(18) \quad r_\omega(a) = \omega^{-1}(\omega \wedge x) \quad \text{and} \quad p_{x \rfloor \omega}(a) = (x \rfloor \omega)^{-1}(x \rfloor \omega) \rfloor a$$

are the rejection from, i.e. part orthogonal to,  $\omega$ , and the projection to the direction of the vector  $x \rfloor \omega$ , which lies in the plane of omega and orthogonal to  $x$ , respectively.

We find the antiderivative  $\partial_{\text{side}}^{-1}x = xr_\omega(x)$ . This can be verified by taking the derivative and using the facts that  $x = p_\omega(x) + r_\omega(x)$ , where  $p_\omega(x)$  is the projection to  $\omega$ , the fact that since the projection to the tangent space splits as in Eq. (17) then also the derivatives split in the same way, and finally that  $\partial_p x \wedge a = d_p a - p(a)$ , where  $\partial_p$  is the derivative projected with the projection  $p$  and  $d_p$  is the dimension of the subspace projected to.

In order to do the final integral for the side along the boundary left by the chamfer cut, which is a circle in the plane  $\omega$ , and at height  $h - \varepsilon$  above the origin, we note that  $r_\omega(x)$  is simply the constant vector height along the circle and therefore also constant with respect to the derivative on that circle, so we are left with integrating  $x = p_\omega(x) + r_\omega(x)$  on the circle. Now  $p_\omega(x)$  is on the plane of the circle, and therefore we know from the disk example that the integral of the  $p_\omega(x)$  -part will be  $2\pi \|p_\omega(x)\|^2 I_2$  with  $I_2 = \omega$  and  $\|p_\omega(x)\|^2 = r^2$ . Integrating the constant produces  $x$  times the constant, and since  $x$  is regular on the whole circle, the subtraction will produce 0. The other boundary component is the circle along the bottom, where the calculation is identical expect that now  $\|r_\omega(x)\| = \varepsilon$ , and the sign is opposite since the orientation of the boundary is opposite. The integral along the sides then total  $\frac{2\pi}{3}r^2(h - 2\varepsilon)I_3$ , where the pseudoscalar  $I_3$  comes from the product of the bivector  $\omega$  and vector  $r_\omega(x)$ .

The other boundary components are the caps on the top and the bottom. The projection to the tangent plane is simply  $p_\omega$ , and therefore splitting again  $x = p_\omega(x) + r_\omega(x)$ , we find the antiderivative

$$(19) \quad \partial_\omega^{-1}x = \frac{1}{2}p_\omega(x)^2 + \frac{1}{2}p_\omega(x)r_\omega(x).$$

The half on the second term comes from the fact that the projection is two-dimensional. We have to integrate this on the boundary of the cap, which is the circle at radius  $r - \varepsilon$  (since we cut the chamfer off the edge). The first term again integrates to zero, since on the circle  $p_\omega(x)$  is a constant, whereas the second term again reduces to the case of the disk, and therefore produces  $\frac{\pi}{3}(r - \varepsilon)^2 \omega r_\omega(x)$ , where we have inserted the 1/3 from the first integral. The cap on the bottom is again the same, with this time  $\|r_\omega(x)\| = \varepsilon$ , and so putting the caps and the side together and letting  $\varepsilon \rightarrow 0$  we get the final result

$$(20) \quad \int_{\text{cylinder}} d^3x = \pi r^2 h I_3$$

as expected.

#### 4. CONCLUSIONS AND OUTLOOK

We have presented a method for computing integrals in  $m$ -dimensions without using coordinates. Naturally, the level of freedom from using coordinates depends on how

the manifold and the integrand are defined. One purely coordinate free way is to define the manifold by solutions of  $m(x) = 0$ , where  $m(x)$  is a function of the vector  $x$  constructed from geometric products of  $x$  with itself and some (possibly infinite) set of constant multivectors  $A_i$ , where the geometric relations between  $A_i$  and  $x$  are known in sufficient detail to allow carrying out all the necessary algebraic manipulations without coordinates. Both of our examples are in this form.

In the examples, we integrate the constant function on two manifolds in order to compute their volumes. The actual computations in these examples are not complicated when compared to the same computation in coordinates, which for a fair comparison needs to take into account the derivation of the Jacobian in polar or cylindrical coordinates. Further development of our method will indeed require building a comprehensive toolbox of systematic methods for finding antiderivatives of multivector valued functions of vector variables on vector manifolds. While this program is still in its infancy, we have found some rules with some level of generality: for example, an antiderivative of  $f(\|x\|)$  in  $d$ -dimensions is simply  $\frac{x}{\|x\|^d} \int ds s^{d-1} f(s)$ , where  $f(s)$  is a scalar valued function of a scalar, and so the remaining integral is an ordinary scalar integral. This rule is of course equivalent to integrating in a spherical coordinate system, expressed in a coordinate-free way.

As an interesting note, in some examples which we have worked out but not reported here, such as the volume of  $B^3$ , it is not necessary to actually find an antiderivative, but rather one can find a function whose derivative differs from the desired one by a function which can be seen to integrate to zero. We can then use such a function instead of the antiderivative to still get the correct result. However, we will not comment on this further before we understand the phenomenon in more detail. It may turn out to be only a fortunate coincidence occurring in a limited number of cases, rather than something that can be included in a general toolbox.

Let us indulge in some speculation concerning possible applications of the method to more than just evaluating integrals in the few special cases where antiderivatives can be explicitly found. Consider a function  $f(x)$  on a manifold  $M$  defined by  $m(x) = 0$  for some multivector valued function  $m(x)$  and with  $x$  in  $\mathbb{R}^d$ . In order to calculate the integral of  $f(x)$  over  $M$ , the method involves finding the  $d$ -fold antiderivative of  $f$  with respect to derivatives projected on  $M$ , and evaluating it on a discrete set of points on the manifold. Therefore, at least in the final step, we only really need to know some topological facts about the manifold in order to choose the points such that they are all on the same branch of the antiderivative. Of course, the manifold also enters into the calculation via the projections of the derivative operator. For the first integration in the case where  $m(x)$  is scalar-valued the projected derivative is given simply by  $(\partial m(x) I_d)^{-1} (\partial m(x) I_d) \lrcorner \partial$ , where the first two  $\partial$ 's affect only the  $m(x)$ 's immediately following them. Similar formulas can be worked out for more general  $m(x)$ . Now, we can use the Taylor series approximation for multivector functions [15] and approximate both functions  $f(x)$  and  $m(x)$  by their Taylor series. If the antiderivatives of all the monomial terms<sup>3</sup> can be explicitly constructed, then this should in principle allow for a systematic series expansion for the values of integrals on a large class of manifolds, in terms of integrals of the monomials. The theoretical connections to algebraic geometry and topology should prove interesting.

---

<sup>3</sup>We need to also expand the inverse appearing in the projection, or to integrate a rational function of multivectors, which cannot be done in quite closed form even for the real numbers, as the roots of the polynomials need to be found in the partial fraction expansion.

For (vector) differential equations the very same rules for finding antiderivatives that are crucial for our method will be useful in finding closed form solutions in a coordinate invariant way. In addition, similar series expansion methods as those outlined above should pave the way to finding series expansions for solutions of vector differential equations, and may even aid in their numerical evaluation.

On a philosophical level, our method represents a further step into the direction of establishing multivectors as geometric numbers, which can indeed be constructed, manipulated and interpreted in a wholly coordinate-free way. Beyond that, we believe that these speculations do not exhaust the potential applications of the method.

#### ACKNOWLEDGEMENTS

We thank A. Lewandowski and L. Thorlacius for very helpful comments and proof-reading of the manuscript. The author is supported in part by Icelandic Research Fund grant 130131-053 and by a grant from the University of Iceland Research Fund.

#### REFERENCES

- [1] D. Hestenes, “Oersted medal lecture 2002: reforming the mathematical language of physics”, *Am. J. Phys* **71**, 104–121 (2003).
- [2] D. Hestenes, “Spacetime physics with geometric algebra”, *Am. J. Phys* **71**, 691–714 (2003).
- [3] D. Hestenes, *New foundations for classical mechanics: fundamental theories of physics (fundamental theories of physics)* (Kluwer Academic Publishers; 2nd edition, 1999).
- [4] D. Hestenes and G. Sobczyk, *Clifford algebra to geometric calculus: a unified language for mathematics and physics (fundamental theories of physics)* (Kluwer Academic Publishers, 1987).
- [5] C. Doran and A. Lasenby, *Geometric algebra for physicists* (Cambridge University Press, 2007).
- [6] S. Gull, A. Lasenby, and Doran, “Imaginary numbers are not real - the geometric algebra of spacetime”, *Found. Phys.* **23**, 1175–1201 (1993).
- [7] A. Lasenby, C. Doran, and S. Gull, “A Multivector derivative approach to Lagrangian field theory”, *Found.Phys.* **23**, 1295–1327 (1993).
- [8] C. Doran, A. Lasenby, and S. Gull, “States and operators in the spacetime algebra”, *Found Phys.* **23**, 1239–1264 (1993).
- [9] A. Lasenby, C. Doran, and S. Gull, “Gravity, gauge theories and geometric algebra”, *Phil.Trans.Roy.Soc.Lond.* **A356**, 487–582 (1998), arXiv:gr-qc/0405033 [gr-qc].
- [10] J. Lasenby, A. N. Lasenby, and C. J. L. Doran, “A unified mathematical language for physics and engineering in the 21st century”, *Philosophical Transactions of The Royal Society B: Biological Sciences* **358** (2000) 10 . 1098 / rsta.2000.0517.
- [11] A. Macdonald, *Vector and geometric calculus* (CreateSpace Independent Publishing Platform, 2012).
- [12] G. Sobczyk and O. Sánchez, “Fundamental theorem of calculus”, English, *Advances in Applied Clifford Algebras* **21**, 221–231 (2011).
- [13] E. Chisolm, “Geometric algebra”, (2012), eprint: 1205.5935.
- [14] A. Macdonald, “A Survey of Geometric Algebra and Geometric Calculus”, (2015).
- [15] E. Hitzer, “Multivector Differential Calculus”, ArXiv e-prints (2013), arXiv:1306.2278 [math.HO].

# CLIFFORD ALGEBRA APPLIED TO THE MOLECULAR DISTANCE GEOMETRY PROBLEM

Rafael Alves<sup>a</sup> and Carlile Lavor<sup>b</sup>

<sup>a</sup> Department of Applied Mathematics  
University of Campinas (Unicamp), Brazil  
rsantos@ime.unicamp.br [presenter, corresponding]

<sup>b</sup> Department of Applied Mathematics  
University of Campinas (Unicamp), Brazil  
clavor@ime.unicamp.br

One of the experimental procedures used to determine the tridimensional structure of a protein molecule is the nuclear magnetic resonance (NMR). This technique provides a set of distances between nearby atoms, which can be used to obtain the cartesian coordinates of all atoms. This problem is known as the Molecular Distance Geometry Problem (MDGP) [5]. Differently from the continuous optimization approach, often applied to MDGP, where the search for a single solution demands a high computational effort, our proposal is to develop methods based on a combinatorial approach. In this case, the search domain can be reduced to a discrete space having the structure of a tree [4]. We refer to this class of problems as the Discretizable Molecular Distances Geometry Problem (DMDGP). An advantage of this approach is the possibility of finding multiple solutions. When the distance information provided by the NMR experiments is given through a list of lower and upper bounds, that is, inexact distances, the problem is also referred as the *interval* DMDGP (*i*DMDGP). An *interval* Branch-and-Prune (*i*BP) algorithm has been developed over the past years, and is able to find multiple solutions for a given instance [2].

A difficulty that arises from the discrete approach is the choice of the number of times we split a given interval. A small number can lead us to an empty set of solutions and a large number could lead to huge number of solutions, where many of them are very similar to each other. For this reason, we decided to take advantage of the geometric interpretation of the *i*DMDGP, as a search for sphere intersections, and apply Clifford algebra tools to reduce the interval distances. Preliminary results can be seen in [1] and in the recent work [3] we also show how Clifford algebras can be useful in interval reduction.

## REFERENCES

- [1] Alves, R., Cassioli, A., Mucherino, A., Lavor, C. and Liberti, L., Adaptive Branching in *i*BP with Clifford Algebra. In *Proceedings of the Workshop on Distance Geometry and Applications*, pp. 1-4, 2013.
- [2] Cassioli, A., Bardiaux, B., Bouvier, G., Mucherino, A., Alves, R., Liberti, L., Nilges, M., Lavor, C. and Malliavin, T., An algorithm to enumerate all possible protein conformations verifying a set of distance constraints. In *BMC Bioinformatics*, v. 16, 2015.
- [3] Lavor, C., Alves, R., Figueiredo, W., Petraglia, A. and Maculan, N., Clifford Algebra and the Discretizable Molecular Distance Geometry Problem. In *Advances in Applied Clifford Algebras*, Online First, 2015.
- [4] Lavor, C., Liberti, L., Maculan, N. and Mucherino, A., Recent advances on the discretizable molecular distance geometry problem. In *European Journal of Operational Research*, v. 219, pp. 698-706, 2012.
- [5] Liberti, L., Lavor, C., Maculan, N. and Mucherino, A., Euclidean distance geometry and applications. In *SIAM Review*, v. 56, pp. 3-69, 2014.



# THE CONSTRUCTION OF CLIFFORD REPRESENTATION OF SPIN-1 SPINORS AND THE RELATION BETWEEN THE CLIFFORD-BASIS REPRESENTATION AND THE GRASSMANN-BASIS REPRESENTATION

**Murat An<sup>a</sup> and Chueng-Ryong Ji<sup>b</sup>**

Department of Physics  
North Carolina State University, Raleigh, NC, USA  
<sup>a</sup> man@ncsu.edu, <sup>b</sup> crji@ncsu.edu

**ABSTRACT.** We derive a spinor representation of spin-1 for Clifford algebra (1,3) (or  $Cl_{1,3}$ ) by using isomorphism  $Cl_{1,3} \approx SO^+(1,3)$  for both  $(1/2, 1/2)$  and  $(1, 0) \oplus (0, 1)$  Lorentz group representations. The relationship between the two representations is found to be analogous to the relationship between the electromagnetic gauge field  $A^\mu$  and the electromagnetic field strength tensor  $F^{\mu\nu}$ . Similar to the relationship between  $A^\mu$  and  $F^{\mu\nu}$ , the two representations can be unified by one formula  $u(p, \lambda) = \not{\epsilon} \wedge \not{p}/m$ . We observe that there is a correlation between these spinors and Hestenes Dirac spinors for spin-1/2 particles in  $Cl_{1,3}$ . We also discuss that the Grassmann basis obtained from spin representation of Clifford algebra provides more convenient basis for spin-1 spinors especially for  $(1, 0) \oplus (0, 1)$  Lorentz group spinor representation, although the Clifford basis is better for spin-1/2 and  $(1/2, 1/2)$  spinor representations of Clifford algebra. The work by Winnberg for a superfield of spinors of Clifford algebra offers a physical meaning to Grassmann variables.

## 1. INTRODUCTION

We begin with Hestenes's description of Dirac spinors with projection method [1] in projective spin-1/2 representation group  $(1/2, 0) \oplus (0, 1/2)$ . We use the isomorphism between  $Cl_{1,3}$  and  $SO^+(1,3)$  groups in order to write down Polarization vectors ( $(1/2, 1/2)$  spinors) with Clifford numbers and show that these spinors can be expressed in terms of Hestenes's Dirac spinors. We find that  $(1, 0) \oplus (0, 1)$  spinors can be given by  $u(p, \lambda) = \not{\epsilon}(p, \lambda) \wedge \not{p}/m$  as the vector potential field  $A_\mu$  and the field strength tensor  $F_{\mu\nu}$  are related to each other. The expressions of these spinors in Standard representation of  $SO(1,3)$  and  $SU(2)$  group for spin-1 with Weyl representation are presented in Appendices (A) and (B), respectively. We discuss that the Grassmann basis or Witt basis [2] may be the better choice of basis for the spinors with chiral representation with coordinates in spherical harmonics form in comparison with the Clifford basis. We also present the relation of polarization vector and  $(1, 0) \oplus (0, 1)$  spinor with gauge fields and discuss how they are connected with local coordinate transformations.

## 2. CLIFFORD ALGEBRA

This section is about the definition and some properties of Clifford algebra. The Clifford algebra  $Cl_{l,n}$  over  $\mathbb{R}$  is an associative algebra having  $l+n$  generators  $\gamma_1, \gamma_2, \dots, \gamma_l, \gamma_{l+1}, \dots, \gamma_{l+n}$  such that

$$\gamma_i^2 = \dots = \gamma_l^2 = 1, \gamma_{l+1}^2 = \dots = \gamma_{l+n}^2 = -1, \quad \gamma_i \gamma_j = -\gamma_j \gamma_i \quad (i \neq j).$$

The interior product and the exterior (wedge) products of  $a$  and  $b$  are

$$a \cdot b = \frac{ab + ba}{2}$$

and

$$a \wedge b = \frac{ab - ba}{2}$$

from adding and subtracting these two equation, we find that Clifford product is

$$ab = a.b + a \wedge b$$

and

$$ba = a.b - a \wedge b.$$

The volume element for the  $\{\gamma_i\}$  frame is

$$(1) \quad I_n = \gamma_1 \wedge \gamma_2 \wedge \dots \wedge \gamma_n.$$

We also have a reciprocal frame  $\{\gamma^i\}$  defined by

$$(2) \quad \gamma^i \cdot \gamma_j = \delta_j^i, \quad \forall i, j = 1, \dots, n.$$

The reciprocal frame can also be constructed as

$$(3) \quad \gamma^j = (-1)^{j-1} \gamma_1 \wedge \gamma_2 \wedge \dots \wedge \check{\gamma}_j \wedge \dots \wedge \gamma_n I_n^{-1},$$

where the check on  $\check{\gamma}_j$  states  $\gamma_j$  is missing form the expression.

**2.1. Clifford Algebra (1,3) and Grades.** Clifford algebra (1,3) is basically a Clifford algebra with Minkowski metric  $\gamma_0^2 = 1, \gamma_1^2 = \gamma_2^2 = \gamma_3^2 = -1$  and  $\gamma_i \gamma_j = -\gamma_j \gamma_i$ . It is represented as  $Cl_{1,3}$  shortly and our metric is defined as  $g^{\mu\nu} = \gamma^\mu \cdot \gamma^\nu$ . Because of its geometrical properties,  $Cl_{1,3}$  is also called as Geometric algebra and each Clifford number has been given a geometric meaning. Moreover, these numbers can be used as position frame for a spacetime point. We have different vector spaces which map to Real vector spaces as  $\Lambda^k \mathbb{R}$  with  $k = 0, 1, \dots, n$  when  $n = p + q$  in  $Cl_{p,q}$ . These different vector spaces make Clifford algebra a graded algebra as  $A = \bigoplus_{nN} A_n$ . The Clifford algebra  $Cl_{1,3}$  is of 16 dimensions and has the basis consisted of

1	scalar	$\Lambda^0 \mathbb{R}$
$\gamma_0, \gamma_1, \gamma_2, \gamma_3$	vector	$\Lambda^1 \mathbb{R}^4$
$\gamma_{01}, \gamma_{02}, \gamma_{03}, \gamma_{23}, \gamma_{31}, \gamma_{12}$	bivector	$\Lambda^2 \mathbb{R}^4$
$\gamma_{123}, \gamma_{023}, \gamma_{031}, \gamma_{012}$	trivector	$\Lambda^3 \mathbb{R}^4$
$\gamma_{0123}$	volume element	$\Lambda^4 \mathbb{R}^4$ .

The whole Clifford Algebra(1,3) in terms of all grades or multivector structure is

$$Cl_{1,3} = \mathbb{R} \oplus \mathbb{R}^4 \oplus \Lambda^2 \mathbb{R}^4 \oplus \Lambda^3 \mathbb{R}^4 \oplus \Lambda^4 \mathbb{R}^4.$$

General form of Poincare isomorphism can be written as  $M^{\mu\nu} \simeq i\gamma^\mu \wedge \gamma^\nu / 2$ . For the rest of paper, we write the wedge product of two Clifford numbers  $\gamma^\mu \wedge \gamma^\nu$  in shortly as  $\gamma^{\mu\nu}$ . In Clifford algebra, the Lorentz transformations can be performed by rotors which are written as  $R = e^{\frac{B}{2}}$ , where B is a bivector and a short-hand notation for  $\gamma^{\mu\nu}$ . It transforms a vector  $v$  as

$$(4) \quad v' = RvR^{-1},$$

where  $R^{-1}$  is  $e^{-\frac{B}{2}}$ . This also can be applied to any multivector A

$$(5) \quad A' = RAR^{-1}.$$

For rotations, the bivector B will be  $B = \gamma_{ij}$  where  $i, j = 1, 2, 3$  with  $i \neq j$  and for boosts it is  $B = -\gamma_0i$ .

### 3. SPINORS

In our work, we will begin with Hestenes' construction of Dirac spinors which can be also seen as part of left minimal ideals. Then, we will construct spin-1 from component spinor approach by associating each component of vectors with a Clifford number. We define the rest frame spinors of  $(1/2, 1/2)$  as vectors of  $Cl_{1,3}$  in cartesian representation of their spins and the  $(1, 0) \oplus (0, 1)$  spinors are derived from polarization vectors such as  $u(p, \lambda) = \not{\epsilon} \wedge \not{p}/m$ .

**3.1. Primitive Idempotents and Minimal left Ideals.** The mathematical definition of the left minimal ideals is that, if  $I$  is a subset of  $S$  semigroup,  $I$  is a left minimal ideal of  $S$  if  $SI \subset I$  and  $I \neq \{\}$ . In [5], it states that there is a non-scalar element  $\gamma_i$  and  $\gamma_i^2 = 1$  in the standard basis  $Cl_{p,q}$ . We could set two left ideals as  $e = \frac{1}{2}(1 + \gamma_i)$  and  $f = \frac{1}{2}(1 - \gamma_i)$ , where  $e + f = 1$  and  $ef = fe = 0$ . Thus,  $Cl_{p,q}$  separates into two left ideals  $Cl_{p,q} = Cl_{p,q}e \oplus Cl_{p,q}f$ .

Theorem: There are  $k = q - r_{q-p}$  non-scalar elements  $\gamma_i^2 = 1$ , such that  $\gamma_i \gamma_j = \gamma_j \gamma_i$  in the basis of  $Cl_{p,q}$ . They have order of  $2^k$  and

$$(6) \quad f = \frac{1}{2}(1 + \gamma_1) \frac{1}{2}(1 + \gamma_2) \dots \frac{1}{2}(1 + \gamma_k).$$

This product of idempotents is primitive in  $Cl_{p,q}$ . Therefore, the left ideal  $S = Cl_{p,q}f$  is minimal in  $Cl_{p,q}$ . In this theorem,  $r_{q-p}$  is the Radon-Hurtwitz number and is given by

$i$	0	1	2	3	4	5	6	7
$r_i$	0	1	2	2	3	3	3	3

and the recursion formula in the Radon-Hurtwitz number is  $r_{i+8} = r_i + 4$ .

**3.2. Spin-1/2 Spinors and Hestences Projection Method.** One of the methods that construct spinors with Clifford algebra is using columns components of matrices as vectors (primitive idempotent). The projection operators can be used separate these components as In [3], Hestenes uses  $U = \frac{1}{4}(1 + \gamma_0)(1 + \sigma_3)$  projection for spinors. In this equation the factors  $\frac{1}{2}(1 + \gamma_0)$  and  $\frac{1}{2}(1 + \sigma_3)$  are energy and spin projection operators. However, in our notation we express our spin operator as  $i\gamma_{12}$  instead of  $\sigma^3$ . With this change our spinor of projective spin representation of  $(1/2, 0) \oplus (0, 1/2)$  would be

$$(7) \quad u = \frac{1}{2}(1 + \gamma_0) \frac{1}{2}(1 + i\gamma_{12}).$$

$u$  represents a positive energy and positive helicity spinor. So, the other variant of spinors would be

$$(8) \quad u_1 = \frac{1}{4}(1 + \gamma_0)(1 + i\gamma_{12}), \quad u_2 = \frac{1}{4}(1 + \gamma_0)(1 - i\gamma_{12}), \quad u_3 = \frac{1}{4}(1 - \gamma_0)(1 + i\gamma_{12}), \quad u_4 = \frac{1}{4}(1 - \gamma_0)(1 - i\gamma_{12}).$$

Note that

$$(9) \quad \gamma_0 u_1 = u_1, \quad \gamma_0 u_2 = u_2, \quad \gamma_0 u_3 = -u_3, \quad \gamma_0 u_4 = -u_4$$

and

$$(10) \quad i\gamma_{12} u_1 = u_1, \quad i\gamma_{12} u_2 = -u_2, \quad i\gamma_{12} u_3 = u_3, \quad i\gamma_{12} u_4 = -u_4.$$

Here,  $u_1, u_2, u_3,$  and  $u_4$  are not the same spinors as the standard Dirac algebra, since each of them are different projections and left minimal ideal operators only operates on between rows. So we need to apply some operators on our original spinor  $u$  or  $u_1$ . We will define two operators to achieve other spinors; charge and spin raising and lowering operators:

$$(11) \quad Q^\mp = \frac{1}{2}(\mp \gamma_1 - i\gamma_2)$$

$$(12) \quad S^\mp = \frac{1}{2}(i\gamma_{23} \pm \gamma_{31}).$$

These operators are also in moving frame so  $Q^\mp = RQ^\mp(0)R^{-1}$  and  $S^\mp = RS^\mp(0)R^{-1}$ . We can simplify them as

$$(13) \quad Q = Q^+ + Q^-, \quad S = S^+ + S^-.$$



Since we only have two states, particle or anti-particle or up or down spins, one of the  $\pm$  always eliminate the one state and change the other state. Using these operators, we can obtain all Dirac spinor representations from  $u_1$  as

$$(14) \quad u^1 = u_1, \quad u^2 = Su_1, \quad v^1 = SQ u_1, \quad v^2 = Qu_1.$$

Dirac spinors for moving frame are given by

$$(15) \quad u^{(i)}(p) = Ru^{(i)}(0).$$

Here, rotor  $R$  describes a boost and we choose rotor as Dirac boost that is a single boost to any direction  $R = e^{-\gamma_0 \phi^i / 2}$ :

$$(16) \quad R = \cos \frac{\phi}{2} - \gamma_0 \frac{\phi^i}{\phi} \sin \frac{\phi}{2} = \sqrt{\frac{E+m}{2m}} \left( 1 + \gamma_0 \frac{p^i}{E+m} \right),$$

where  $\cosh(\frac{\phi}{2}) = \sqrt{\frac{E+m}{2m}}$  and  $\sinh(\frac{\phi}{2}) \frac{\phi^i}{\phi} = \frac{p^i}{\sqrt{2m(E+m)}}$  with  $\phi = \sqrt{(\phi^1)^2 + (\phi^2)^2 + (\phi^3)^2}$ .

If we look at matrix representation of Clifford number, we can see that it is the same spinors in standard representation but in  $4 \times 4$  matrices.

The spinor  $u^{(1)}$  is given by

$$(17) \quad u^{(1)}(0) \cong \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad u^{(1)}(p) = Ru^{(1)} \cong \sqrt{\frac{E+m}{2m}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{p_3}{E+m} & 0 & 0 & 0 \\ \frac{p_1+i p_2}{E+m} & 0 & 0 & 0 \end{pmatrix},$$

The conjugate of spinors are  $\bar{u}^{(i)}(p) = u^{*(i)}(0)R^{-1}$  and in matrix representation it is transpose of spinors. In Standard model, antiparticle spinors are actually charge conjugation of particle spinors so that instead of  $u^{(3)}$  and  $u^{(4)}$ , they are  $v^{(1)} = -i\gamma_2 u^{(1)*}$  and  $v^{(2)} = -i\gamma_2 u^{(2)*}$  in matrix representation or negative energy solutions (Feynman-Stckelburg interpretation)  $v^{(1)}(p) = u^{(4)}(-p)$ ,  $v^{(2)}(p) = u^{(3)}(-p)$  in terms of Clifford numbers  $v^{(1)} = Su^2 = -i\gamma_2 u^2$  and  $v^{(2)} = Su^1 = -i\gamma_2 u^1$  in rest frame.

We could put Dirac spinors from Eq. (14) in terms of Clifford number as

$$(18) \quad u^1(p) = R(1 + \gamma_0)(1 + i\gamma_{12})/4$$

$$(19) \quad u^2(p) = Ri\gamma_{23}(1 + \gamma_0)(1 + i\gamma_{12})/4$$

$$(20) \quad v^1(p) = R\gamma_3(1 + \gamma_0)(1 + i\gamma_{12})/4$$

$$(21) \quad v^2(p) = R(-i\gamma_2)(1 + \gamma_0)(1 + i\gamma_{12})/4.$$

### 3.3. Spin-1 Spinors.

3.3.1. *Polarization Vectors.* In order to construct the polarization vectors in Clifford algebra, we could use the isomorphism between Clifford and  $SO(1,3)$  such that  $Cl_{1,3} \approx SL(2, \mathbb{C}) \approx SO^+(1,3)$  since the standard polarization vectors are already derived from this group.

We can begin with describing our vector as  $v = \gamma_\mu x^\mu$  instead of  $v = \{t, x, y, z\} = e_\mu x^\mu$  where  $e_\mu$  are column vectors for  $\mu = 0, 1, 2, 3$  in  $SO(1,3)$  so that we could use Clifford numbers instead of columns vectors. Then we define the 0 helicity as the z direction vector as  $\varepsilon(0,0) = \gamma_3$  for  $\varepsilon(p, \lambda)$  in  $Cl_{1,3}$ . Using raising and lowering operators,  $J^+ = (i\gamma_{23} - \gamma_{31})/\sqrt{2}$  and  $J^- = (i\gamma_{23} + \gamma_{31})/\sqrt{2}$ , we can get spin +1 and -1 polarization vectors and the whole polarization

vectors will look like

$$(22) \quad \varepsilon(0, +) = -(\gamma_1 + i\gamma_2)/\sqrt{2}$$

$$(23) \quad \varepsilon(0, 0) = \gamma_3$$

$$(24) \quad \varepsilon(0, -) = -(-\gamma_1 + i\gamma_2)/\sqrt{2}.$$

As we know, we could make direct connection between polarization vectors and spherical harmonics as

$$(25) \quad \varepsilon(0, +) \sim Y_1^1, \quad \varepsilon(0, 0) \sim Y_1^0, \quad \varepsilon(0, -) \sim Y_1^{-1}.$$

We could define the spin operator in terms of spinors

$$(26) \quad J_z = \varepsilon(+)\varepsilon^*(+) - \varepsilon(-)\varepsilon^*(-) = i\gamma_{12}$$

and the corresponding function of operator in  $Cl$  is interior product. One can see that in Clifford algebra of  $Cl_{1,3}$  the operation  $i\gamma^{12} \cdot v = v'$  corresponds to  $J_z v = v'$  in  $SO(1,3)$  space.

In Clifford algebra, we use rotors to make a boost in a system. Contrary to Spin-1/2 spinors, for Spin-1 spinors we need two-sided rotors as in Lorentz transformations. The systems makes sense if we consider spin-1 particles as mediator of spin-1/2 particles. In this case, as an example we will show polarization vectors in momentum space with Dirac spinor that is  $R$  is a boost to any direction  $R = e^{-\gamma_0 \phi^i / 2}$  and polarization vectors will be

$$(27) \quad \begin{aligned} \varepsilon(p, +) &= \frac{1}{\sqrt{2}} R(-\gamma_1 - i\gamma_2) R^{-1} \\ &= \frac{1}{m} (\gamma_0 p^R + \gamma_1 \frac{m(p^0 + m)/\sqrt{2} + p^1 p^R}{p^0 + m} + \gamma_2 \frac{im(p^0 + m)/\sqrt{2} + p^2 p^R}{p^0 + m} + \gamma_3 \frac{p^3 p^R}{p^0 + m}) \end{aligned}$$

$$(28) \quad \begin{aligned} \varepsilon(p, 0) &= R\gamma_3 R^{-1} \\ &= \frac{1}{m} (\gamma_0 p^3 + \gamma_1 \frac{p^1 p^3}{m + p^0} + \gamma_2 \frac{p^2 p^3}{m + p^0} + \gamma_3 \frac{m^2 + mp^0 + (p^3)^2}{m + p^0}) \end{aligned}$$

$$(29) \quad \begin{aligned} \varepsilon(p, -) &= \frac{1}{\sqrt{2}} R(\gamma_1 - i\gamma_2) R^{-1} \\ &= \frac{1}{m} (\gamma_0 p^L + \gamma_1 \frac{m(p^0 + m)/\sqrt{2} + p^1 p^L}{p^0 + m} + \gamma_2 \frac{-im(p^0 + m)/\sqrt{2} + p^2 p^L}{p^0 + m} + \gamma_3 \frac{p^3 p^L}{p^0 + m}), \end{aligned}$$

where  $p^R = (p^1 + ip^2)/\sqrt{2}$  and  $p^L = (p^1 - ip^2)/\sqrt{2}$ .

When we compare these polarization vectors with standard ones in (65), we can observe one-to-one correspondance by finding each component by  $\varepsilon^\mu(p, \lambda) = \gamma^\mu \cdot \varepsilon(p, \lambda)$ .

Using the notation from [4], we could construct spin-1 state from spin-1/2 states as

$$(30) \quad A_{m-m'} = \Psi_m \Psi_{m'}^* \Gamma.$$

We could acquire similar expression for Polarization vectors

$$(31) \quad \varepsilon(p, m - m') = u^m(p) \bar{u}^{m'}(p) \Gamma(p) + v^m(p) \bar{v}^{m'}(p) \Gamma(p).$$

Here, we could define  $\Gamma(p) = RSQR^{-1}$  with  $S$  and  $Q$  from (13). It makes us enable to construct polarization vectors from Dirac spinors in Clifford algebra in Sec.(III-B) :

$$(32) \quad \varepsilon(p, +) = (u^1(p) \bar{v}^2(p) - v^1(p) \bar{u}^2(p))/\sqrt{2} = R(-\gamma_1 - i\gamma_2) R^{-1} / \sqrt{2}$$

$$(33) \quad \varepsilon(p, 0) = (u^1(p) \bar{v}^1(p) + u^2(p) \bar{v}^2(p) - v^1(p) \bar{u}^1(p) - v^2(p) \bar{u}^2(p))/2 = R\gamma_3 R^{-1}$$

$$(34) \quad \varepsilon(p, -) = (u^2(p) \bar{v}^1(p) + v^2(p) \bar{u}^1(p))/\sqrt{2} = R(\gamma_1 - i\gamma_2) R^{-1} / \sqrt{2}.$$

3.3.2.  $(1,0) \oplus (0,1)$  Lorentz Group Spinors. Since we defined the spinors of  $(1/2, 1/2)$  Lorentz group as  $\gamma^\mu$ , we could set a isomorphism to spin  $(1,0) \oplus (0,1)$  in terms of bivectors by using the relation similar convention of electric and magnetic field related with four potential as  $E + iB \in (1,0)$  and  $E - iB \in (0,1)$  as these parts are related as  $\gamma_{\mu\nu} F^{\mu\nu} = \nabla \wedge A$  where  $\nabla = \gamma^\mu \partial_\mu$  and  $A = \gamma^\mu A_\mu$ .

$(1/2, 1/2)$		$(1,0)$	$(0,1)$
$\gamma_1$	$\leftrightarrow$	$(\gamma_{01} - i\gamma_{23})$	$(\gamma_{01} + i\gamma_{23})$
$\gamma_2$	$\leftrightarrow$	$(\gamma_{02} - i\gamma_{31})$	$(\gamma_{02} + i\gamma_{31})$
$\gamma_3$	$\leftrightarrow$	$(\gamma_{03} - i\gamma_{12})$	$(\gamma_{03} + i\gamma_{12})$

Because of the similarity with  $F^{\mu\nu}$ , the spinor  $u(p) \sim \varepsilon_i \wedge p/m$  with derivative change into momentum and for the rest frame  $u(0) \sim \gamma_i \wedge \gamma_0$ . In this situation our spinors on  $(1,0) \oplus (0,1)$  representation will look like

$$(35) \quad u(0, +) = -\gamma_{01}(1 - i\gamma_{12})/\sqrt{2}$$

$$(36) \quad u(0, -) = \gamma_{01}(1 + i\gamma_{12})/\sqrt{2}$$

$$(37) \quad u(0, 0) = \gamma_{03}$$

so that we can make connection with  $u(0, \lambda)$  vectors as

$$(38) \quad -(\gamma_{01} - i\gamma_{23})(1 - i\gamma_{12})/2 \leftrightarrow (1, 0, 0, 0, 0, 0)$$

$$(39) \quad (\gamma_{01} - i\gamma_{23})(1 + i\gamma_{12})/2 \leftrightarrow (0, 0, 1, 0, 0, 0)$$

$$(40) \quad (\gamma_{03} - i\gamma_{12})/\sqrt{2} \leftrightarrow (0, 1, 0, 0, 0, 0)$$

$$(41) \quad -(\gamma_{01} + i\gamma_{23})(1 - i\gamma_{12})/2 \leftrightarrow (0, 0, 0, 1, 0, 0)$$

$$(42) \quad (\gamma_{01} + i\gamma_{23})(1 + i\gamma_{12})/2 \leftrightarrow (0, 0, 0, 0, 0, 1)$$

$$(43) \quad (\gamma_{03} + i\gamma_{12})/\sqrt{2} \leftrightarrow (0, 0, 0, 0, 1, 0).$$

Lets make the same Lorentz transformation of boost for the spin  $(1,0) \oplus (0,1)$  representation with similar way of polarization vectors  $u(p, \lambda) = Ru(0, \lambda)R^{-1}$ :

(44)

$$u(p, +) = N \left\{ -\frac{(\gamma_{01} - i\gamma_{23})(1 - i\gamma_{12})}{2} (p^0 + m + p^3)^2 + \frac{(\gamma_{03} - i\gamma_{12})}{\sqrt{2}} 2(p^0 + m + p^3)p^R + \frac{(\gamma_{01} - i\gamma_{23})(1 + i\gamma_{12})}{2} 2(p^R)^2 \right. \\ \left. - \frac{(\gamma_{01} + i\gamma_{23})(1 - i\gamma_{12})}{2} (p^0 + m - p^3)^2 - \frac{(\gamma_{03} + i\gamma_{12})}{\sqrt{2}} \sqrt{2}\sqrt{2}(p^0 + m - p^3)p^R + \frac{(\gamma_{01} + i\gamma_{23})(1 + i\gamma_{12})}{2} 2(p^R)^2 \right\}$$

$$(45) \quad u(p, 0) = N \left\{ -\frac{(\gamma_{01} - i\gamma_{23})(1 - i\gamma_{12})}{2} \sqrt{2}(p^0 + m + p^3)p^L + \frac{(\gamma_{03} - i\gamma_{12})}{\sqrt{2}} \sqrt{2}(p^0(p^0 + m) - (p^3)^2) \right. \\ \left. + \frac{(\gamma_{01} - i\gamma_{23})(1 + i\gamma_{12})}{2} \sqrt{2}(p^0 + m - p^3)p^R + \frac{(\gamma_{01} + i\gamma_{23})(1 - i\gamma_{12})}{2} \sqrt{2}(p^0 + m - p^3)p^L \right. \\ \left. + \frac{(\gamma_{03} + i\gamma_{12})}{\sqrt{2}} \sqrt{2}(p^0(p^0 + m) - (p^3)^2) - \frac{(\gamma_{01} + i\gamma_{23})(1 + i\gamma_{12})}{2} \sqrt{2}(p^0 + m + p^3)p^R \right\}$$

(46)

$$u(p, -) = N \left\{ -\frac{(\gamma_{01} - i\gamma_{23})(1 - i\gamma_{12})}{2} 2(p^L)^2 + \frac{(\gamma_{03} - i\gamma_{12})}{\sqrt{2}} 2(p^0 + m - p^3)p^L - \frac{(\gamma_{01} - i\gamma_{23})(1 + i\gamma_{12})}{2} (p^0 + m - p^3)^2 \right. \\ \left. - \frac{(\gamma_{01} + i\gamma_{23})(1 - i\gamma_{12})}{2} 2(p^L)^2 + \frac{(\gamma_{03} + i\gamma_{12})}{\sqrt{2}} - 2(p^0 + m + p^3)p^L - \frac{(\gamma_{01} + i\gamma_{23})(1 + i\gamma_{12})}{2} (p^0 + m + p^3)^2 \right\},$$

where  $N = 1/(\sqrt{2}m(p_0 + m))$ . We can see one-to-one component correspondence to standard spinors in (69), (70), and (71) and Clifford spinors from the expanded form Clifford spinors.

Here, we are comparing our  $(1,0) \oplus (0,1)$  field spinors with spinors which are expressed in terms of spin  $\pm$  states components rather than cartesian states  $(x, y, z)$  as well as in chiral basis

(Left and Right handed spin 1  $SU(2)$  groups). We require to define a new basis which is written in terms of handedness and spin states. In the next section, we will see that this expression will look better when we change our basis into the Grassmann or Witt basis.

**3.4. Grassmann algebra and Clifford Spinors.** J. Winnberg showed that [6] there is an equivalence between the spin representation of a Clifford algebra and a Grassmann algebra which is the main algebra of supersymmetry and the relation to the orthogonal group.

We use a little different notation and redefine our Clifford basis as

$$(47) \quad \theta_1 = (\gamma_0 + \gamma_3)/\sqrt{2}, \quad \theta_2 = (-\gamma_1 - i\gamma_2)/\sqrt{2}$$

$$(48) \quad \bar{\theta}_1 = (\gamma_0 - \gamma_3)/\sqrt{2}, \quad \bar{\theta}_2 = (\gamma_1 - i\gamma_2)/\sqrt{2}$$

then we see that it satisfies the properties of Grassmann algebra with Clifford spinors.

$$(49) \quad (\theta_1)^2 = (\bar{\theta}_1)^2 = (\theta_2)^2 = (\bar{\theta}_2)^2 = 0 \quad \text{and} \quad \theta_i \theta_j = -\theta_j \theta_i \text{ for } i \neq j$$

So we could rewrite our polarization vector with this new basis as well:

$$(50) \quad \varepsilon(p, \lambda) = \gamma_\mu \varepsilon^\mu = \theta_1 \varepsilon^+ - \theta_2 \varepsilon^L + \bar{\theta}_2 \varepsilon^R + \bar{\theta}_1 \varepsilon^-,$$

where  $\varepsilon^+ = (\varepsilon^0 + \varepsilon^3)/\sqrt{2}$ ,  $\varepsilon^- = (\varepsilon^0 - \varepsilon^3)/\sqrt{2}$ ,  $\varepsilon^R = (\varepsilon^1 + i\varepsilon^2)/\sqrt{2}$ ,  $\varepsilon^L = (\varepsilon^1 - i\varepsilon^2)/\sqrt{2}$ .

$$(51) \quad u(p, \lambda) = \theta_1 \theta_2 u^1 + (\bar{\theta}_1 \theta_1 - \theta_1 \bar{\theta}_1 + \bar{\theta}_2 \theta_2 - \theta_2 \bar{\theta}_2) u^2 + \bar{\theta}_1 \bar{\theta}_2 u^3 \\ + \bar{\theta}_1 \theta_2 u^4 + (\bar{\theta}_1 \theta_1 - \theta_1 \bar{\theta}_1 - \bar{\theta}_2 \theta_2 + \theta_2 \bar{\theta}_2) u^5 + \theta_1 \bar{\theta}_2 u^6.$$

When we compare the Grassmann basis with Clifford one, you could see that the previous expression looks much nicer than the one in Clifford basis in (44), (45), and (46). We also see how they are related that  $\theta_2$  and  $\bar{\theta}_2$  are for  $\pm$  polarizations and  $\theta_1$  and  $\bar{\theta}_1$  are for right and left handed basis, although  $\theta_1$  are not exactly  $R$  or  $L$  systems but combination of them and each components of  $u(p, \lambda)$  in  $Cl_{1,3}$  (38), (40), (39), (41), (43), (42) would be the components in (51) respectively.

The relation between polarization vector and  $u$  spinor can be expressed as

$$(52) \quad u(p, \lambda) = \not{\varepsilon}(p, \lambda) \wedge \not{p}/m.$$

It is also possible to express  $\not{p}$  like  $\not{p} = \theta_1 p^+ - \theta_2 p^L + \bar{\theta}_2 p^R + \bar{\theta}_1 p^-$ .

So, we are able to write down spinors as a function of each others like

$$(53) \quad u^1(p, \lambda) = -(p^L \varepsilon^+ - p^+ \varepsilon^L)/m$$

$$(54) \quad u^2(p, \lambda) = -(p^- \varepsilon^+ - p^+ \varepsilon^- + p^R \varepsilon^L - p^L \varepsilon^R)/\sqrt{2}m$$

$$(55) \quad u^3(p, \lambda) = (p^- \varepsilon^R + p^R \varepsilon^-)/m$$

$$(56) \quad u^4(p, \lambda) = -(p^+ \varepsilon^R - p^R \varepsilon^+)/m$$

$$(57) \quad u^5(p, \lambda) = -(p^+ \varepsilon^- - p^- \varepsilon^+ + p^L \varepsilon^R - p^R \varepsilon^L)/\sqrt{2}m$$

$$(58) \quad u^6(p, \lambda) = (p^- \varepsilon^L - p^L \varepsilon^+)/m$$

#### 4. POLARIZATION VECTORS AND GAUGE THEORY

A gauge field provides connection between two local fields. A spin 1 field can be considered as a gauge field between spin 1/2 fields and this connection gives us two sided left minimal ideal field as we see in (31). There are similar studies about consideration of two-sided equivalence transformations of spinors [8]. Gauge field can be illustrated with a set of internal dimension called fiber. In this view, the polarization vectors in Sec (3.3) can be seen as local transformations of left minimal ideal spinors similar to  $S$  operator in (13) which transforms particle and anti particle spinors to each other as  $v^{2,1} = S u^{1,2}$  and effected by moving frame

by two-sided  $v^{2,1}(p) = RSR^{-1}Ru^{1,2}$ . One can observe that the time-fibre of left minimal ideal with a boost will be momentum  $\gamma^\mu p_\mu = mR\gamma_0R^{-1}$  and space-fiber will be the vector potential  $q\gamma^\mu A_\mu^i = mR\gamma_iR^{-1}$  because of the relationship between polarization vectors and vector potentials

$$(59) \quad A_\mu(p) = \int \frac{e^{ipx}}{(2\pi)^4} \varepsilon_\mu(x) dx.$$

We could write the displacement between two point on a manifold with wave function as

$$(60) \quad d\psi(x) = \psi(x+dx) - \psi(x) \cong i\gamma^\mu (p_\mu + qA_\mu) dx$$

where  $\gamma^\mu dx = dx^\mu$  and the wave function with these displacement will describe as a phase transformation

$$(61) \quad \psi'(x) = e^{iq \int A_\mu dx^\mu} e^{ipx} = e^{i\alpha(x)} \psi(x).$$

When we expand our new wave function in differential form as

$$(62) \quad \psi'(x+dx) = \psi'(x) + \gamma^\mu \partial_\mu \psi'(x) dx + \gamma^\mu \gamma^\nu \partial_\mu \partial_\nu \psi'(x) \frac{dx^2}{2} + \dots$$

We can rewrite this expansion with momentum space  $p_\mu = -i\partial_\mu$  and separate into different grades:

$$(63) \quad \begin{aligned} \psi'(x+dx) &= \psi'(x) + i\gamma^\mu (p_\mu + qA_\mu) \psi'(x) dx \\ &- \left\{ p^2 + \gamma^{\mu\nu} \{ i(\partial_\mu p_\nu) + i(\partial_\mu A_\nu) + A_\mu p_\nu + p_\mu A_\nu \} \right\} \psi'(x) \frac{dx^2}{2} + \dots, \end{aligned}$$

where  $\gamma^{\mu\nu} p_\mu p_\nu = p \wedge p$  and  $\partial_\mu p^\mu$  terms vanish. We also can observe that the two  $(1,0) \oplus (0,1)$  spinor fields emerge in the second order of gauge field expansion in bivector part while polarization vector appeared in vector part.

#### APPENDIX A. SPIN-1 VECTOR SPINOR IN $SO(1,3)$ GROUP

The boost operators for polarization vectors in standard representation are elements of  $SO(1,3)$  group

$$(64) \quad K_1 = \begin{pmatrix} 0 & i & 0 & 0 \\ i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad K_2 = \begin{pmatrix} 0 & 0 & i & 0 \\ 0 & 0 & 0 & 0 \\ i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad K_3 = \begin{pmatrix} 0 & 0 & 0 & i \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ i & 0 & 0 & 0 \end{pmatrix}$$

and our boost operator defined as  $B = e^{-iK_i \phi^i}$  and our vectors are  $\varepsilon^\mu(0,+) = \{0, -1, -i, 0\}$ ,  $\varepsilon^\mu(0,0) = \{0, 0, 0, 1\}$ , and  $\varepsilon^\mu(0,-) = \{0, 1, -i, 0\}$ . Then the polarization vectors in momentum space would be

and in terms of polarization states

$$(65) \quad \varepsilon^\mu(p,+) = N \begin{pmatrix} (p^0+m)p^R \\ (p^0+m)m/\sqrt{2} + p^1 p^R \\ i(p^0+m)m/\sqrt{2} + p^2 p^R \\ p^3 p^R \end{pmatrix}, \quad \varepsilon^\mu(p,-) = N \begin{pmatrix} (p^0+m)p^L \\ (p^0+m)m/\sqrt{2} + p^1 p^L \\ -i(p^0+m)m/\sqrt{2} + p^2 p^L \\ p^3 p^L \end{pmatrix}$$

$$\varepsilon^\mu(p,0) = N \begin{pmatrix} (p^0+m)p^3 \\ p^1 p^3 \\ p^2 p^3 \\ (p^0+m)m + (p^3)^2 \end{pmatrix} \quad \text{where } N = 1/(m(p^0+m))$$

APPENDIX B.  $(1, 0) \oplus (0, 1)$  LORENTZ GROUP SPINORS OF SPIN-1

Similar to spin 1/2 spinors, we could express  $(1, 0) \oplus (0, 1)$  in two separate spaces, right-handed and left-handed, instead of pauli matrices we could use  $SU(2)$  matrices for spin-1

$$(66) \quad J_1 = \frac{1}{2} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad J_2 = \frac{i}{2} \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}, \quad J_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

In this case, the rest frame spinors in chiral representation are

$$(67) \quad u(0, +) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad u(0, 0) = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad u(0, -) = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

The boost operators are :  $(K_1)_R = iJ_1$ ,  $(K_2)_R = iJ_2$ ,  $(K_3)_R = iJ_3$ ;  $(K_1)_L = -iJ_1$ ,  $(K_2)_L = -iJ_2$ ,  $(K_3)_L = -iJ_3$

and the complete boost operator is

$$(68) \quad K = \begin{pmatrix} K_R & 0 \\ 0 & K_L \end{pmatrix}$$

with boosted frame will be  $u(p, \lambda) = e^{-iK_i \phi^i} u(0, \lambda)$  and spinors in momentum space are

$$(69) \quad u(p, +) = \frac{1}{2\sqrt{2}m(p^0 + m)} \begin{pmatrix} (p^0 + m + p^3)^2 \\ 2(p^0 + m + p^3)p^R \\ 2(p^R)^2 \\ (p^0 + m - p^3)^2 \\ -2(p^0 + m - p^3)p^R \\ 2(p^R)^2 \end{pmatrix}$$

$$(70) \quad u(p, 0) = \frac{1}{\sqrt{2}m(p^0 + m)} \begin{pmatrix} (p^0 + m + p^3)p^L \\ p^0(p^0 + m) - (p^3)^2 \\ (p^0 + m - p^3)p^R \\ -(p^0 + m - p^3)p^L \\ p^0(p^0 + m) - (p^3)^2 \\ -(p^0 + m + p^3)p^R \end{pmatrix}$$

$$(71) \quad u(p, -) = \frac{1}{2\sqrt{2}m(p^0 + m)} \begin{pmatrix} 2(p^L)^2 \\ 2(p^0 + m - p^3)p^L \\ (p^0 + m - p^3)^2 \\ 2(p^L)^2 \\ -2(p^0 + m + p^3)p^L \\ (p^0 + m + p^3)^2 \end{pmatrix}$$

## APPENDIX C. WIGNER-D MATRIX

In general the Wigner D-matrix is described as

$$(72) \quad D_{m'm}^j(\phi', \theta, \phi) = \langle jm' | \mathcal{R}(\phi', \theta, \phi) | jm \rangle = e^{-im'\phi'} d_{m'm}^j e^{-im\phi}$$

and the wigner (small) d-matrix with general element is

$$(73) \quad d_{m'm}^j(\theta) = \langle jm' | e^{-iJ_y \theta} | jm \rangle.$$

In this Appendix, we show that there is another way to get these matrix elements by using Clifford algebra. As we know, Pauli matrices already describe spin 1/2 matrix elements. Yet, our spinors (or eigen vectors) are a little different from the classical one. There are two sets of spinors for up and down states each for particle and anti-particle states. But, we will only look at particle case of  $\bar{u}^1 u^1$ .

When we use the Dirac spinors in (18) and (19), we will have

$$(74) \quad d_{1/2,1/2}^{1/2} \sim \bar{u}^1 e^{\gamma_3 \theta/2} u^1 = \cos \frac{\theta}{2} \bar{u}^1 u^1$$

$$(75) \quad d_{1/2,-1/2}^{1/2} \sim \bar{u}^1 e^{\gamma_3 \theta/2} u^2 = -\sin \frac{\theta}{2} \bar{u}^1 u^1$$

$$(76) \quad d_{-1/2,1/2}^{1/2} \sim \bar{u}^2 e^{\gamma_3 \theta/2} u^1 = \sin \frac{\theta}{2} \bar{u}^1 u^1$$

$$(77) \quad d_{-1/2,-1/2}^{1/2} \sim \bar{u}^2 e^{\gamma_3 \theta/2} u^2 = \cos \frac{\theta}{2} \bar{u}^1 u^1.$$

The only difference in spin 1/2 case is instead of one, we have two cases for two basis. We could achieve higher cases also by using the relation

$$(78) \quad d_{(m'+m),2m'}^{2j} = \sum_{\text{all possible } N} \frac{1}{\sqrt{N}} \langle jm' | e^{\gamma_3 \theta/2} | jm'' \rangle \langle jm'' | e^{-\gamma_3 \theta/2} | jm \rangle.$$

Then, we could also derive spin 1 wigner d-matrix elements from Pauli algebra as

$$(79) \quad d_{1,1}^1 = \bar{u}^1 e^{\gamma_3 \theta/2} u^1 \bar{u}^1 e^{-\gamma_3 \theta/2} u^1 = \cos^2 \frac{\theta}{2} u^1 = \frac{1 + \cos \theta}{2} \bar{u}^1 u^1$$

$$(80) \quad d_{0,1}^1 = \{ \bar{u}^1 e^{\gamma_3 \theta/2} u^1 \bar{u}^1 e^{-\gamma_3 \theta/2} u^2 + \bar{u}^2 e^{\gamma_3 \theta/2} u^1 \bar{u}^1 e^{-\gamma_3 \theta/2} u^1 \} / \sqrt{2} = \frac{\sin \theta}{\sqrt{2}} \bar{u}^1 u^1$$

$$(81) \quad d_{-1,1}^1 = \bar{u}^2 e^{\gamma_3 \theta/2} u^1 \bar{u}^1 e^{-\gamma_3 \theta/2} u^2 = \sin^2 \frac{\theta}{2} u^1 = \frac{1 - \cos \theta}{2} \bar{u}^1 u^1$$

$$(82) \quad d_{0,0}^1 = d_{1,1}^1 - d_{-1,-1}^1 = \bar{u}^1 e^{\gamma_3 \theta/2} (u^1 \bar{u}^1 - u^2 \bar{u}^2) e^{-\gamma_3 \theta/2} u^1 = \cos \theta \bar{u}^1 u^1.$$

## REFERENCES

- [1] D. Hestenes. *Observables, operators, and complex numbers in the Dirac theory*. J. Math. Phys., **16(3):556**, (1975).
- [2] M. Pavšič. *Space inversion of spinors revisited: A possible explanation of chiral behavior in weak interactions*. Phy. Lett. B, **692(3)** (2010) 212-217.
- [3] D. Hestenes. *Clifford algebra and the interpretation of quantum mechanics*. In J. S. R. Chisholm and A.K. Common, editors, Clifford Algebras and their Applications in Mathematical Physics, page ...D. Reidel, 1986.
- [4] M. A. J. Ashdown, S. S. Somaroo, S. F. Gull, C. J. L. Doran and A. N. Lasenby. *Multilinear Representations of Rotation Groups within Geometric Algebra*. J. Math. Phys. **39(3)** (1998), 1566-1588.
- [5] Lounesto, Pertti. *Clifford Algebras and Spinors*. Cambridge University Press, Cambridge, 1997.
- [6] J. O. Winnberg. *Superfields as an extension of the spin representation of the orthogonal group*. J. Math. Phys. **18**, 625 (1977).
- [7] R. Brauer, H. Weyl *Spinors in n dimensions*. Am. J. Math. **57**, 425-449 (1935).
- [8] J. S. R. Chisholm, R. S. Farwell. *Gauge transformations of spinor within a Clifford algebra*. J. Math. Phys. **32**, 2805-2823 (1991).

# MODELING, SIMULATION AND CONTROL FOR A BIPEDAL ROBOT USING CGA

**L. Osuna<sup>a</sup>, H. Caballero, O. Carbajal, A. Loukianov and E. Bayro-Corrochano**

<sup>a</sup> Automatic Control Department  
CINVESTAV-Gdl, Zapopan, Jalisco, Mexico  
linda.osuna@gdl.cinvestav.mx [presenter, corresponding]

Due to the amount of mathematical work and calculations required to obtain the dynamic model for a bipedal robot, different methods have been proposed. The main two methods orientated to obtain the dynamic model for a robot are the Newton-Euler and the Euler-Lagrange formulations, nevertheless, both methods work in the Euclidian Space. Once the models are obtained, a representation is needed to simulate and implement the control laws online, and for this case the virtual work method or the differential kinematic are used (see [1]).

The problem when trying to implement the model obtained via Euler-Lagrange or Newton-Euler, is the large extension of these representations, and the high computational cost that its solving demands. To counter this, the Conformal Geometric Algebra (CGA) improves the calculation and implementation time due to its nature, since the Euclidian Space is extended, the representation of the rigid transformations resulting from the connection between links are simplified (see [2]).

## 1. MODELING ROBOTS VIA CGA

Making use of the CGA  $G_{4,1}$ , rigid transformations can be expressed in conformal geometry carrying out reflections between planes:

- Translation: The translation is the product of two reflections between parallel planes

$$(1) \quad T = 1 + \frac{1}{2}ae_{\infty} = e^{-\frac{a}{2}e_{\infty}},$$

here  $a$  represent the translational vector, any entity can be translated doing  $x' = Tx\tilde{T}$ .

- Rotation: The rotation is the product of two reflections between nonparallel planes:

$$(2) \quad R = \cos\left(\frac{\theta}{2}\right) - \sin\left(\frac{\theta}{2}\right)l = e^{-\frac{\theta}{2}l},$$

here  $l$  denotes the rotation axis.

The screw motion *called motor*  $M = TR\tilde{T}$  represents the rotation related to an arbitrary axis  $L$

$$(3) \quad M = e^{-\frac{q}{2}L},$$

where  $L$  is a line, and  $q$  represents the rotation angle or the translation in case of  $L$  at infinity. Similarly, any geometric entity can be rotated doing  $x' = Mx\tilde{M}$ .



1.1. **Kinematics.** The direct kinematics for any serial robot can be represented as a succession of motors and its valid points, lines, planes, circles and spheres, as developed in [3]

$$(4) \quad x'_j = \prod_{i=1}^j M_i x_j \prod_{i=1}^j \tilde{M}_{j-i+1},$$

similarly,  $L'$  is defined in terms of  $L$  as follows

$$(5) \quad L'_j = \prod_{i=1}^{j-1} M_i L_j \prod_{i=1}^{j-1} \tilde{M}_{j-i}.$$

Differential kinematics equation:

$$(6) \quad \dot{x}'_j = \sum_{i=1}^j [x'_j \cdot L'_i] \dot{q}_i$$

is explained in [4].

1.2. **Dynamics.** The dynamic equation for a serial robot with  $n$  DOF can be written as follows:

$$(7) \quad M\ddot{q} + C\dot{q} + G = \tau,$$

where  $q$  are the angles of the  $n$  DOF,  $M$  is the inertia matrix,  $C$  is the coriolis and centrifugal forces matrix,  $G$  is the vector of gravitational forces, and  $\tau$  is the torque input. Now, (7) can be written in terms of CGA resulting in

$$(8) \quad \delta I \ddot{q} + V^T m (V \ddot{q} + \dot{V} \dot{q} + a) = \tau,$$

where  $a$  is the acceleration of the gravity force,  $\delta I = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 0 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} I_1 & 0 & \dots & 0 \\ I_2 & I_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I_n & I_n & \dots & I_n \end{bmatrix}$  with  $I_i$

as the inertia of the link  $i$ ,  $m = \begin{bmatrix} m_1 & 0 & \dots & 0 \\ 0 & m_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & m_n \end{bmatrix}$ , with  $m_i$  as the mass of each link, and

$V = \begin{bmatrix} x'_1 \cdot L'_1 & 0 & \dots & 0 \\ x'_2 \cdot L'_1 & x'_2 \cdot L'_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x'_n \cdot L'_1 & x'_n \cdot L'_2 & \dots & x'_n \cdot L'_n \end{bmatrix}$  with  $x_i$  as the mass center of the link  $i$  in its initial position

and  $x'_i$  as the mass center in function of joints variables; similarly, the joints axis  $i$  is denoted as  $L_i$  and the joints axis  $i$  in function of the joints variables as  $L'_i$ . The development of the dynamic model can be seen in [5].

Good is to remark that  $m$ ,  $I$  and  $a$  are constant matrices and can be known. Only  $V$  and  $\dot{V}$  changes through time and they are computed in a parallel way, using a thread for each matrix's component. In this way and having  $n^2$  threads it is possible to get the  $V$  in  $O(\text{Log}_2(n))$ .

## 2. CONTROL ALGORITHMS

There exist several kinds of control algorithms applicable to robotic systems (so call mechanical systems). These algorithms could be of classical control, as shown in [6], mainly formed by proportional controls via state feedback (P, PD, or PID). Besides the classical control, modern control algorithms can be applied, a representative technique is the Sliding Mode Control (SMC), developed in [7]. By means of SMC the system acquires robustness against external disturbances and parametric variations.

Of the proportional controllers, the PD (Proportional-Derivative) is the more popular among the robotic systems. The form of this controller is as follows

$$(9) \quad \tau = -K_p \varepsilon - K_v \dot{q},$$

where  $K_p \in \mathbb{R}^+$  is the proportional gain,  $K_v \in \mathbb{R}^+$  is the derivative gain,  $\varepsilon = q - q_{ref}$  is the tracking error and  $\dot{q}$  is the velocities of the joints.

On the other hand, to apply a control law using SMC the form of  $\tau$  results

$$(10) \quad \tau = -K_p S + K_s \frac{S}{\|S\|}$$

where  $S = c\varepsilon + \dot{q}$ , with  $c \in \mathbb{R}^+$ , is the sliding surface for the SMC,  $K_s \in \mathbb{R}^+$  is the gain for the sliding mode, and  $\frac{S}{\|S\|}$  is the approximation of the sign function when using vectors.

### 3. SIMULATION RESULTS

The dynamic model was obtained for a 6 DOF bipedal robot as the one from the Fig. 1 according to the Section 1.2. Once having the model, a simulation using MATLAB software was done, applying the control laws from Eq. (9) and Eq. (10) separately in order to follow the references for taking a step.

The results of the tracking control are shown in Figures 2 and 3. As well, a simulation of the walking was done and in Fig. 4 the performance is shown.

### 4. REMARKS

In this work, a method to obtain the dynamic model of serial robots was used to obtain the model from a 6 DOF bipedal robot. As was shown, different control techniques were applied to the CGA dynamic model, moreover, simulations were done, exhibiting one of the main advantages of modeling using CGA, because the same model could be used to analyze, control law design and simulation of the robot.

Is also important to remark that the application of different control techniques using CGA is possible because the control techniques use matrices, and even though the elements of the matrices are CGA entities, the theory of classic and modern control is equally applicable independently of the space in which is being worked.

### REFERENCES

- [1] Høifødt, H., Dynamic Modeling and Simulation of Robot Manipulators, M. Sc. Thesis, *Norwegian University of Science and Technology*, 2011.
- [2] Bayro-Corrochano, E., Geometric Computing: for Wavelet Transforms, Robot Vision, Learning, Control and Action. *Springer*, 2010.
- [3] Bayro-Corrochano, E. and Kähler, D., Motor Algebra Approach for Computing the kinematics of robot manipulators. In *Journal of Robotics Systems*, volume 17(9), pages 495-516, 2000.
- [4] Zamora-Esquivel, J. and Bayro-Corrochano, E., Kinematics and Differential Kinematics of Binocular Robot Heads. In *Proceedings of ICRA*, 2006.
- [5] Zamora-Esquivel, J. and Bayro-Corrochano, E., Parallel Forward Dynamics: a geometric approach. In *International Conference on Intelligent Robots and Systems*, pages 2377-2382, 2010.
- [6] Kelly, R. and Santibañez, V., Control de Movimiento de Robots Manipuladores. *Pearson Prentice Hall*, 2013.
- [7] Utkin, V., Guldner, J. and Shi, J., Sliding Mode Control in Electro-Mechanical Systems. *Taylor & Francis Group*, 2009.

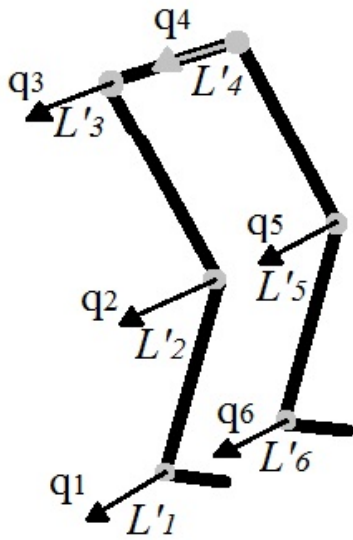


FIGURE 1. Scheme for 6 DOF bipedal robot.

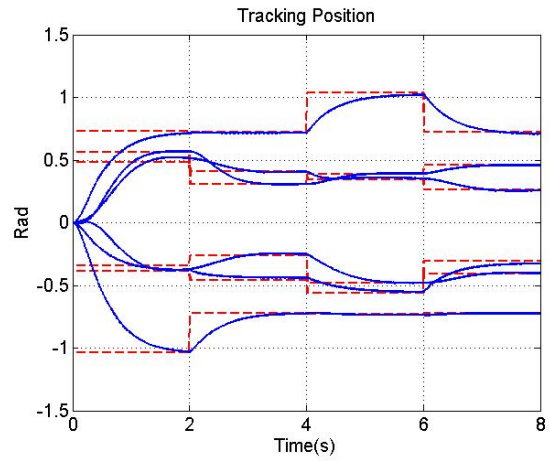


FIGURE 3. Tracking Reference for the angles  $q$  of the bipedal robot using SMC controller from Eq. (10).

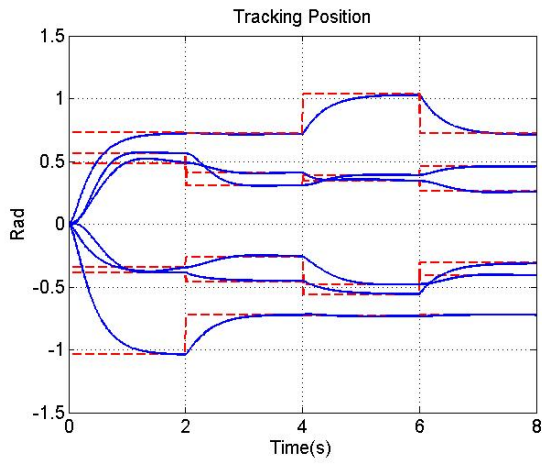


FIGURE 2. Tracking Reference for the angles  $q$  of the bipedal robot using PD controller from Eq. (9).

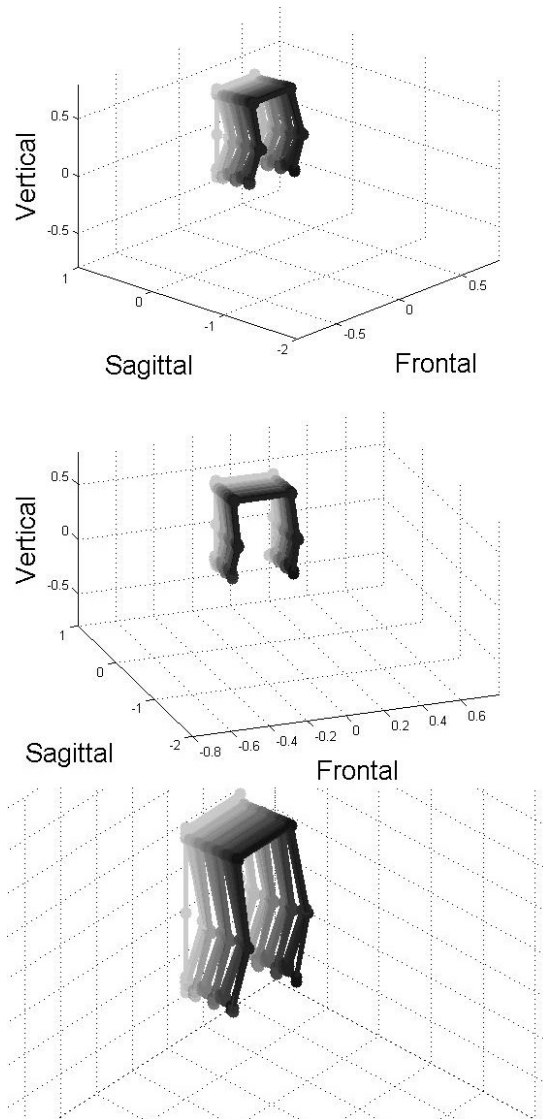


FIGURE 4. Walking simulation.

# A GEOMETRIC APPROACH FOR PID CONTROLLER DESIGN

L. Lechuga-Gutiérrez<sup>a</sup> O. Carbajal-Espinosa<sup>b</sup> and E. Bayro-Corrochano<sup>c</sup>

<sup>a,b,c</sup> Department of Electrical Engineering and Computer Science  
 Centro de Investigación y de Estudios Avanzados del I.P.N., Guadalajara, Jalisco, México  
 lrlechuga@gdl.cinvestav.mx  
 ocarbajal@gdl.cinvestav.mx,  
 edb@gdl.cinvestav.mx

## 1. INTRODUCTION

The PID controller (Proportional, Integral and Derivative) is a well know algorithm used in many applications, as chemical processes, temperature control, robotics and others. From the point of view of Geometric Algebra, the PID controller is an incomplete technique, because the final control law is designed only with partial information, taking in account only a scalar value obtained by the dot product between a gain vector  $K = [k_p, k_i, k_d]^T$ , and an error vector given by the error, the integral of this error and its derivative, this is  $E_r = [e(t), \int e(t)dt, \frac{de(t)}{dt}]^T$  leaving aside the properties and advantages offered by the geometric product between them. In this work is presented a novel approach to associate the geometric product of the gain and error vectors with the geometric entity called *motor* in order to improve the performance of the PID controller. The PID controller was designed as the sum of the product of error elements and the gain elements. The complet control law is defined as the error by a proportional gain, plus an integral gain by the integral of the error plus the derivative of the error signal by its respective derivative gain. There are methods to define the values of the gains, as the Ziegler and Nichols method [1], and the controller can be defined as

$$(1) \quad k_p \left( e(t) + \frac{1}{k_i} \int_0^t e(\tau) d\tau + k_d \frac{de(t)}{dt} \right)$$

Now, defining  $K_i = \frac{k_p}{k_i}$  and  $K_d = k_p k_d$ , the PID becomes

$$(2) \quad K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

this can be expressed as the dot product between two vectors as:

$$(3) \quad K \cdot E_r = [K_p, K_i, K_d] \cdot \left[ e(t) + \int_0^t e(\tau) d\tau + \frac{de(t)}{dt} \right]$$

## 2. PID DEFINITION IN GEOMETRIC ALGEBRA

We will use the geometric algebra of the Euclidean space  $\mathbb{G}_{3,0,0}$  as the mathematical framework with all its properties and the vector basis  $e_1, e_2, e_3$  to define the PID control law. Using the geometric product of two vector, we obtain the scalar part of the PID controller and a bivector part with the wedge product  $K \wedge E_r$ , this is show in the figure 1.

$$(4) \quad K \wedge E_r = e_{23} \left( K_i \frac{de(t)}{dt} - K_d \int_0^t e(\tau) d\tau \right) + e_{31} \left( K_d e(t) - K_p \frac{de(t)}{dt} \right) + e_{12} \left( K_p \int_0^t e(\tau) d\tau - K_i e(t) \right)$$

Then, the PID controller is defined now as the sum of a scalar part (dot product) and a bivector part (wedge product), and it is easy to see that the geometric product creates a PID control that depends on the magnitude of both vectors (vector error ( $E_r$ ) and the gain's vector ( $K$ )) and the angle between them. Now, using a *motor* [5] it is possible to define a new gain vector  $K'$  as

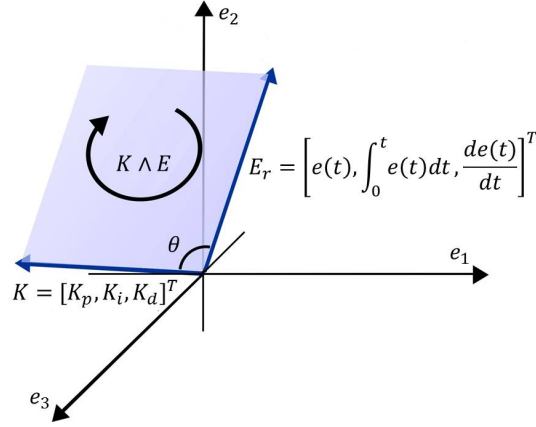


FIGURE 1. PID controller defined in  $\mathbb{G}_{3,0,0}$

$K' = M(\theta)K_0\tilde{M}(\theta)$ , where  $K_0$  is the initial gain vector. In order to modify our control law it is only required to modify the angle  $\theta$  of the motor, and simultaneously modify the magnitude of the control law. The new gain vector  $K'$  is given by

$$(5) \quad K' = M(\theta)K_0\tilde{M}(\theta) = 1 + I(R_K K_0 \tilde{R}_K + K_0 \cdot E_r)$$

where  $R_K = e^{(-\frac{1}{2}l_K(\theta_0 + \theta))}$ ,  $\theta_0 = \cos^{-1}\left(\frac{K_0 \cdot E_r}{\|K_0\| \|E_r\|}\right)$ ,  $\theta$  is the actualization parameter and  $l_K$  is defined as

$$(6) \quad l_K = (K' \wedge E_r + I((K' \wedge E_r) \wedge K' \cdot E_r))$$

this means that the gain vector  $K'$  rotates in the plane formed between itself and the error's vector  $E_r$  once the parameter  $\theta$  is actualized, furthermore is scaled in a factor equals to  $(K_0)$  without altering how to obtain the control law, since it is still the dot product between them.

Finally the control law  $U$  created by the geometric PID' ( $PID_G$ ) is expressed by

$$(7) \quad U = E_r \cdot MK' \tilde{M}$$

### 3. SIMULATION RESULTS

To evaluate the efficiency of the proposed geometric PID a linear second-order plant was simulated:

$$(8) \quad m\ddot{X} = b\dot{X} + kX - U(t)$$

where  $m=1\text{kg}$ ,  $k=.1\text{Nm}$ ,  $b=.7\text{Pa}\cdot\text{s}$ , the initial vector's of gains  $K_0$ , was calculated with the method of Ziegler Nichols ( $K_0=[2 \ 0.07 \ 10]$ ) and the desired reference is  $.7\text{m}$ . The results of the control  $PID_G$  was compared to the classic PID Figure 2

The control law  $U$  for a classic PID is expressed by.

$$(9) \quad U = E_r \cdot K_0$$

and the control law  $U_G$  generated by the  $PID_G$

$$(10) \quad U = E_r \cdot M(\theta)K_0\tilde{M}(\theta)$$

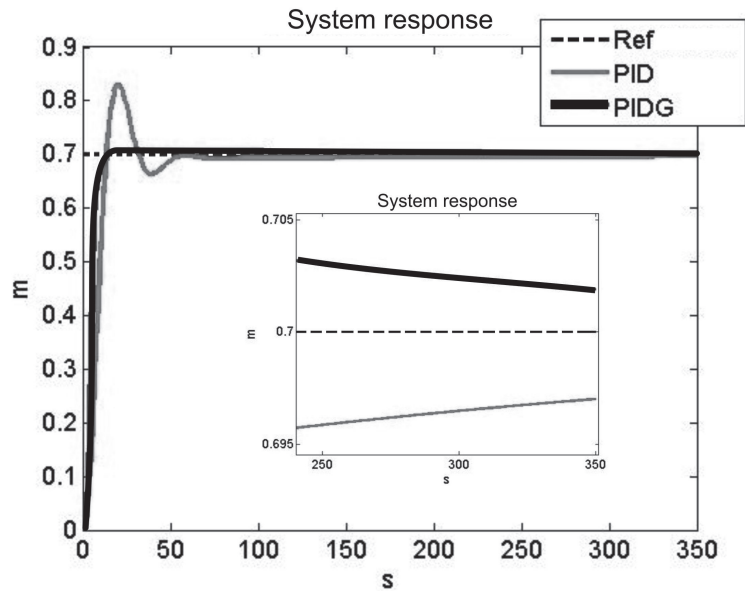


FIGURE 2. Classic PID controller vs geometric PID

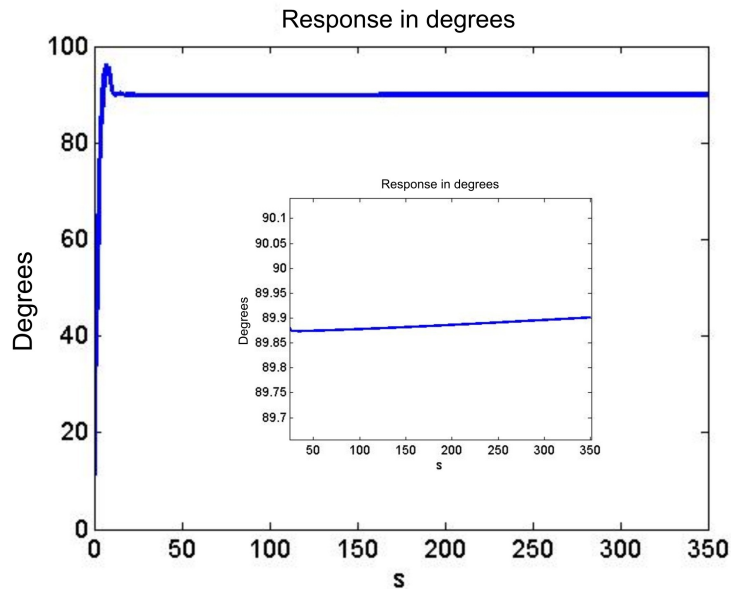


FIGURE 3. Angle evolution

The response of the angle  $\theta$  is shown in figure 3, it can be seen that while the state converges to the desired reference the angle converges to  $90^\circ$ . The motor changes the angle of the vector  $K'$  respect to the error vector  $E_r$ , the evolution of the gain vector is shown in figure 4.

Finally in figure 5 is presented the dynamic of the error, its derivative and integral, both the PID control as  $PID_G$  PLC as 3D vectors.

#### 4. CONCLUSION

In this paper it is shown a novel method using geometric algebra to redefine the classic PID control. Instead of change the three values of the gains of the controller, we only need to change the value of an angle in order to modify the performance of the controller.

Dynamic of the gains vector

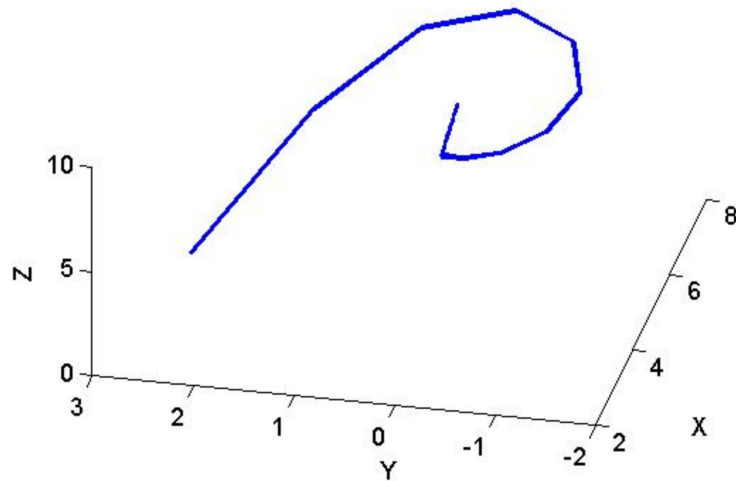


FIGURE 4. Evolución del vector de ganancias

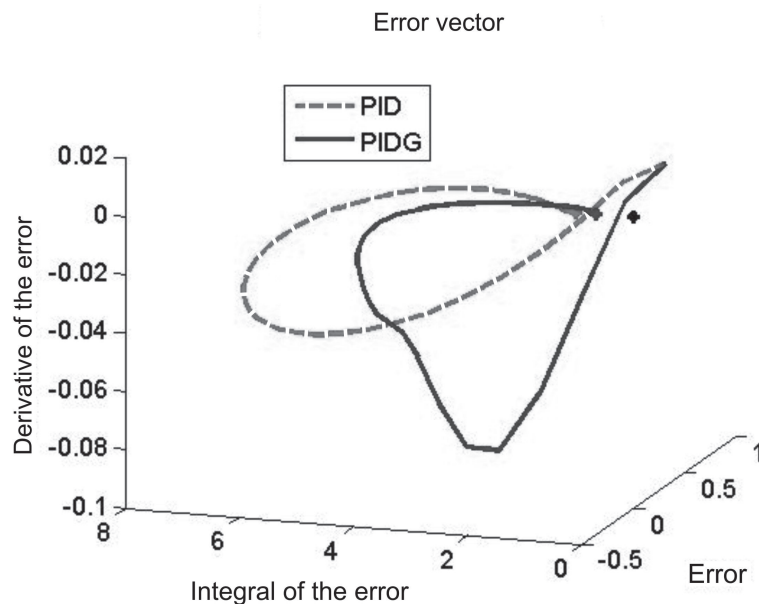


FIGURE 5. Classic PID error vs Geometric PID error

## REFERENCES

- [1] K. Ogata, Modern Control Engineering, Fifth Edition. *Prentice Hall* 2010.
- [2] C. Perwass, Geometric Algebra with Applications in Engineering. In *Springer-Verlag*, Berlin, 2009.
- [3] K. Aström and T. Hägglund, Advanced PID Control. In *The Instrumentation, Systems, and Automation Society (ISA)*, EE.UU, 2006.
- [4] K. Aström, Computer-Controlled Systems, Theory and Design. In *Prentice-Hall*, New York, 1997.
- [5] E. Bayro Corrochano, Geometric Computing, for wavelet transformations, robot vision, learning, control an action. In *Springer-Verlag*, London, 2010.
- [6] E. Bayro Corrochano, Geometric Computing for Perception Action Systems: Concepts, Algorithms, and Scientific Applications. In *Springer-Verlag*, London, 2001.
- [7] J. Vince, Geometric Algebra for Computer Graphics. In *Springer-Verlag*, London, 2008.
- [8] D. Hestenes, New Foundations for Classical Mechanics. In *Springer-Verlag*, New York, 1999.
- [9] L. Doran, Geometric Algebra and its Applications to Mathematical Physics. In *Ph.D. Dissertation, Univ. Cambridge*, Cambridge, U.K., 1994.
- [10] W. Clifford, Preliminary Sketch of Bi-quaternions. In *Proc. London Math. Soc.*, 4:381-395, London, 1873.

# PARALLEL CLIFFORD SUPPORT VECTOR MACHINES USING THE GAUSSIAN KERNEL

G. López-González<sup>a</sup>, N. G. Arana-Daniel<sup>b</sup>, and E. J. Bayro-Corrochano<sup>c</sup>

Department of Electrical Engineering and Computer Science  
Center for Research and Advanced Studies (Cinvestav), Guadalajara, Mexico  
gelopez@gdl.cinvestav.mx<sup>a</sup> [corresponding], edb@gdl.cinvestav.mx<sup>c</sup> [presenter]

Department of Electronics and Computation  
Center of Exact Sciences and Engineering (CUCEI), Guadalajara, Mexico  
nancyaranad@gmail.com<sup>b</sup>

**ABSTRACT.** This work presents a parallelization method for the Clifford Support Vector Machines, based in two characteristics of the Gaussian Kernel. The pure real-valued result and its commutativity allows us to separate the multivector data in its defining subspaces. These subspaces are independent from each other, so we can solve the problem using parallelism. The motivation is to present an easy approach that can be explained using the more common known concepts of complex numbers and quaternions, because in general there exists a lack of familiarity with geometric algebra.

## 1. INTRODUCTION

There exist empirical and theoretical evidence that Support Vector Machines (SVM) [17, 18, 4] give good generalization performance on a wide variety of problems [9].

The approach presented in this work can be applied to SVMs that use the kernel trick, originally proposed in [1]. This trick, in the case of SVMs, is derived from the dual problem, that consists in the quadratic maximization of Lagrange multipliers

$$(1) \quad L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} y_i y_j \alpha_i \alpha_j x_i^T x_j,$$

where  $\alpha_i$  represents the Lagrange multiplier for each input vector  $x_i$ .  $y_i$  and  $y_j$  are the labels of the input vectors  $x_i$  and  $x_j$ , respectively. We can replace  $x_i^T x_j$  with the dot product between two explicit mapped input vectors  $\phi(x_i), \phi(x_j)$ . The explicit mapping  $\phi(\cdot)$  transforms the original data into a higher dimensional Reproducing Kernel Hilbert Space (RKHS). The kernel trick consist in doing this mapping in an implicit way, reducing the computational cost

$$(2) \quad K(x_i, x_j) \simeq \phi(x_i) \cdot \phi(x_j).$$

In recent years extensions of the real-valued SVM have been proposed, including those based on complex numbers [20], quaternions [10] and Clifford algebras [3]. These extensions aim for two things: first, to classify problems beyond the binary one with only one machine and second, to work with data codified in those complex and hyper-complex spaces.

---

The first author is supported by CONACYT-Mexico with the Ph.D. scholarship with number 235318, and the project funding CB 178222.



In this work we present a parallelization method for these hyper-complex SVM. This new approach takes advantage of two characteristics of the Gaussian Kernel. The pure real-valued result and its commutative nature. We can use this fact to separate or to keep apart the subspaces which define the multivector input data. Also they help us to reduce the dual Lagrangian. We also did a deduction based in the analytic method used in Sequential Minimal Optimization (SMO). The deduction proves that the subspaces are independent from each other. Therefore we can solve the problem using parallelism.

The paper is organized as follows:

Section 2 is a brief introduction to geometric algebra, with a concise explanation of the operations used in this work. Section 3 is an introduction to Clifford SVMs, this presents an explanation of the CSVM decision function and its dual Lagrangian. Section 4 is an explication of the derivation of the analytic part of the Sequential Minimal Optimization algorithm. Section 5 is dedicated to explain the properties of the Gauss kernel which allow us to design the parallelized algorithm presented in this work.. Section 6 presents some experimental results. And finally, section 7 is devoted to the conclusions.

## 2. GEOMETRIC ALGEBRA

Geometric algebra, also called Clifford algebra [8, 2, 12], is a coordinate-free approach to geometry based on the algebras of Grassmann and Clifford.

Let  $\mathbb{V}$  be a vector space of dimension  $n$ . We are going to define and generate an algebra  $\mathbb{G}_n$ , called a geometric algebra where  $e_1, e_2, \dots, e_n$  is a set of basis vectors of  $\mathbb{V}^n$  called canonical vector basis.

The basis vectors square to  $1, -1$ ; this means that there are positive integers,  $p$  and  $q$ , such that  $n = p + q$  and

$$(3) \quad \begin{aligned} e_i^2 &= 1, i = 1 \dots p \\ e_i^2 &= -1, i = p + 1, \dots, p + q. \end{aligned}$$

The algebraic product, called the geometric product or Clifford product, of two basis vectors is anti-commutative:

$$(4) \quad e_j e_k = -e_k e_j, \forall j \neq k.$$

The product of two basis vectors will simply be denoted by juxtaposition as

$$(5) \quad e_j e_k = e_{jk}.$$

This introduces the concept of grade. The product of  $n$  non-equal basis vectors results in a new generator of grade  $n$ . An algebra with  $n$  basis vectors has  $2^n$  generators  $\{1, e_1, \dots, e_n, e_{12}, \dots, e_{(n-1)n}, \dots, e_{1\dots n}\}$ . The generator of grade  $n$  is called the unit pseudoscalar of the algebra, and is generally denoted by the letter "I". The algebra  $\mathbb{G}_n$  is obtained with the direct sum of the linear subspaces of grades  $0, 1, 2, \dots, n$ .

$$(6) \quad \mathbb{G}_n = \bigwedge^0 \mathbb{V}_n \oplus \bigwedge^1 \mathbb{V}_n \oplus \bigwedge^2 \mathbb{V}_n \oplus \dots \oplus \bigwedge^n \mathbb{V}_n.$$

A multivector  $A \subset \mathbb{G}_n$  can be separated in its generators as

$$(7) \quad A = A_1 + A_{e_1} + A_{e_2} + \dots + A_I.$$

Geometric algebras isomorphic to complex numbers and quaternions. A complex number has the form

$$(8) \quad z = a + bi,$$

where  $i^2 = -1$ . In this way, it is easy to see that the algebra  $\mathbb{G}_{0,1}$  with the generators  $\{1, e_1\}$  is isomorphic with the complex numbers.

In a similar way if we observe the fundamental formula for quaternion multiplication

$$(9) \quad i^2 = j^2 = k^2 = ijk = -1,$$

and the algebra  $\mathbb{G}_{0,2}$  with the generators  $\{1, e_1, e_2, I\}$  we can get the same behaviour and get the same result with the product of the vectors as follows:

$$(10) \quad e_1^2 = e_2^2 = I^2 = e_1 e_2 I = e_{1212} = -1.$$

**Geometric Algebra Product and Norm.** Now we provide a simplified version of the operations involved in this work, the Clifford product and the norm of a multivector.

The Clifford product of two multivectors  $A, B$  is denoted in an implicit way as

$$(11) \quad C = AB,$$

and it can be interpreted as the result of each element of  $A$  multiplied by each element of  $B$ . This is similar to the multiplication of two polynomials. Therefore if we multiply a scalar  $\lambda$  and a multivector  $A$  the result is

$$(12) \quad \lambda A = \sum_{i=1}^{i=I} \lambda A_i i,$$

where  $i$  is the corresponding generator.

The Clifford product is associative and distributive with respect to the addition, but it is not commutative

$$(13) \quad AB \neq BA.$$

To obtain the norm of a multivector we must multiply it with its conjugate, just like with complex numbers

$$(14) \quad |A| = AA^\dagger,$$

instead of defining the conjugate of a multivector, we are going to provide another interpretation to keep the simplicity of the definitions. The result of the equation 14 is a summation of the square of each element, without taking into account the corresponding generator

$$(15) \quad |A| = \sum_{i=1}^{i=I} A_i^2.$$

### 3. CLIFFORD SVM

The Clifford SVM [3] is the most general extension of the real-valued support vector machine.

The dual Lagrangian for the optimization problem is

$$(16) \quad L_D = \frac{1}{2} \sum_{i,j=1}^l \sum_{m,n} [(y_i)_m (\alpha_i)_m K(x_i, x_j)_{mn} (y_j)_n (\alpha_j)_n] - \sum_{i=1}^l \sum_m (\alpha_i)_m,$$

where  $l$  is the number of training vectors. Both  $m$  and  $n$  represent the corresponding generators, e.g.  $m_1$  is the real part,  $m_2$  is the part corresponding to the  $e_1$  generator, etc.  $K$  is the kernel function. The subindex  $mn$  represents a Clifford product between those generators and means that we only use that part of the function result, e.g. if  $m = e_1$  and  $n = e_{12}$ , then we only use the coefficient of  $e_2$ .

The decision function for the Clifford SVM is written as

$$(17) \quad y = g \text{sign}_v \left[ \sum_{j=1}^l (\alpha_j \circ y_j) (k(x_j, x) + b) \right],$$

where  $v$  is the state of valency, corresponding to the number of classes that the Clifford SVM will classify. The Clifford SVM can classify up to  $v = (2^n)^2$  classes, where  $n$  is the number of basis vectors of the Clifford algebra in which the input data are defined. The  $\circ$  operator is defined as

$$(18) \quad \alpha_j \circ y_j = \sum_m m (\alpha_j)_m (y_j)_m$$

where  $m$  stands for the corresponding generator.

### 4. SEQUENTIAL MINIMAL OPTIMIZATION

As stated before, the training of a dual SVM requires the maximization of a very large and complex quadratic programming optimization problem (QP). There have been some proposal to solve the QP without the need to use the whole training set at the same time. Chunking algorithm [16] consists in selecting the worst  $M$  training vectors that violate the KKT conditions and solve the QP with these. We repeat this step adding the next  $M$  violators to the previous result until the whole QP problem is solved. Osuna [11] showed that the QP can be divided in a series of smaller QP problems, and also proposes to maintain the size of the sub-problem, to allow the training of arbitrarily sized data sets.

Sequential Minimal Optimization [13], SMO, is a proposal to solve the minimum sub-problem, consisting of two Lagrange multipliers that have to obey a linear equality constraint. The advantage of the SMO lies in the fact that solving for two Lagrange multipliers can be done analytically, therefore an entire inner iteration due to numerical QP optimization is avoided.

SMO has three parts, an analytic method to solve for the two Lagrange multipliers, heuristics to choosing which Lagrange multipliers to optimize and a method for computing the bias. We will only see the analytic method because it provides, in combination with the properties of the Gaussian kernel, the mathematical proof of the validity of this work.

The derivation of the analytic method [14, 19] is as follows:

Due the constraint

$$(19) \quad \sum_{i=1}^l y_i \alpha_i = 0,$$

we have that, without loss of generality, we can select two Lagrange multipliers  $\alpha_1$  and  $\alpha_2$  for optimization, these have to meet the next constraint

$$(20) \quad y_1 \alpha_1 + y_2 \alpha_2 = y_1 \alpha_1^{old} + y_2 \alpha_2^{old} = const,$$

and giving the next definitions

$$(21) \quad \begin{aligned} s &= y_1 y_2 \\ \gamma &= \alpha_1 + s \alpha_2, \end{aligned}$$

we can rewrite the dual Lagrangian, equation 1, in terms of  $\alpha_1$  and  $\alpha_2$ , by discarding all the parts that do not depend of those terms as

$$(22) \quad \begin{aligned} L_D &= \alpha_1 + \alpha_2 + const - \frac{1}{2}(y_1 y_1 (x_1^T x_1) \alpha_1^2) + \\ & y_2 y_2 (x_2^T x_2) \alpha_2^2 + 2y_1 y_2 (x_1^T x_2) \alpha_1 \alpha_2 + 2 \left( \sum_{i=3}^N \alpha_i y_i x_i^T \right) (y_1 x_1 \alpha_1 + y_2 x_2 \alpha_2) + const, \end{aligned}$$

if we have  $K(x_1, x_1) = x_1^T x_1$ ,  $K(x_2, x_2) = x_2^T x_2$ ,  $K(x_1, x_2) = x_1^T x_2$  and

$$(23) \quad \begin{aligned} v_j &= \sum_{i=3}^N \alpha_i y_i x_i^T x_j \\ &= x_j^T w^{old} - \alpha_1^{old} y_1 x_1^T x_j - \alpha_2^{old} y_2 x_2^T x_j \\ &= (x_j^T w^{old} - b^{old}) + b^{old} - \alpha_1^{old} - \alpha_1^{old} y_1 x_1^T x_j - \alpha_2^{old} y_2 x_2^T x_j \\ &= u_j^{old} + b^{old} - \alpha_1^{old} y_1 x_1^T x_j - \alpha_2^{old} y_2 x_2^T x_j, \end{aligned}$$

where  $u_j$  is the output of  $x_j$  under the old parameters. With these conditions the equation 22 becomes

$$(24) \quad \begin{aligned} L_D &= \alpha_1 + \alpha_2 - \frac{1}{2}(K(x_1, x_1) \alpha_1^2 + \\ & K(x_2, x_2) \alpha_2^2 + 2sK(x_1, x_2) \alpha_1 \alpha_2) + 2y_1 v_1 \alpha_1 + 2y_2 v_2 \alpha_2 + const, \end{aligned}$$

reducing the problem we have

$$(25) \quad L_D = \frac{1}{2} \eta \alpha_2^2 + (y_2 (E_1^{old} - E_2^{old}) - \eta \alpha_2^{old}) \alpha_2 + const,$$

where  $\eta = 2K(x_1, x_2) - K(x_1, x_1) - K(x_2, x_2)$  and the variable  $E_j^{old} = u_j^{old} - y_j$  is the estimation error. The first derivative of equation 25 is

$$(26) \quad \frac{\partial L_D}{\partial \alpha_2} = \eta \alpha_2 + (y_2 (E_1^{old} - E_2^{old}) - \eta \alpha_2^{old}),$$

if  $\frac{\partial L_D}{\partial \alpha_2} = 0$ , we have that

$$(27) \quad \alpha_2^{new} = \alpha_2^{old} + \frac{y_2(E_1^{old} - E_2^{old})}{\eta},$$

by setting the bounds if  $y_1 \neq y_2$

$$(28) \quad \begin{aligned} L &= \max(0, \alpha_2 - \alpha_1) \\ H &= \min(C, C + \alpha_2 - \alpha_1) \end{aligned}$$

or if  $y_1 = y_2$

$$(29) \quad \begin{aligned} L &= \max(0, \alpha_2 + \alpha_1 - C) \\ H &= \min(C, \alpha_2 + \alpha_1) \end{aligned}$$

we have

$$(30) \quad \begin{aligned} &H \quad \text{if} \quad \alpha_2^{new} \geq H \\ \alpha_2^{new,clipped} &= \alpha_2^{new} \quad \text{if} \quad L < \alpha_2^{new} < H \\ &L \quad \text{if} \quad \alpha_2^{new} \leq L, \end{aligned}$$

and the new value for  $\alpha_1$  can be computed as

$$(31) \quad \alpha_1^{new} = \alpha_1^{old} + s(\alpha_2^{old} - \alpha_2^{new,clipped})$$

## 5. GAUSSIAN KERNEL

The Gaussian kernel is based in a radial basis function, RBF, and has different definitions such as

$$(32) \quad g(x, y) = \exp\left(-\frac{|x-y|}{2\sigma^2}\right)$$

and

$$(33) \quad g(x, y) = \frac{1}{\sqrt{2\pi\rho}} \exp\left(-\frac{|x-y|}{2\pi^2}\right),$$

as we can observe, the Gaussian kernel has two very interesting properties. The first one is that the mapping is not based in the Clifford product, but in the norm of a subtraction of vectors. Equation 13 tell us that a kernel based in the Clifford product is not commutative. Now if we observe equation 15 we can see that the result of this kernel is indeed commutative.

In equation 15 is observed that the result of the kernel is purely real, regardless of the nature of the entry vectors.

Based on this properties and the equation 12, we can rewrite the decision function in equation 17, as a combination of  $m$  subspaces, each one for every generator used to label the data

**Algorithm 1** Gaussian Kernel

---

**Input:** vector data  $x$ , vector data  $y$   
Initialize  $sum = 0$ .  
**for**  $k = 1$  **to**  $vectorSize$  **do**  
  **for**  $l = 1$  **to**  $dimensionSize$  **do**  
     $sum += (x_{k,l} - y_{k,l})^2$ .  
  **end for**  
**end for**  
 $g = rbf(sum)$   
**Return:**  $g$

---

$$(34) \quad y_m = \text{sign} \left[ \sum_{j=1}^l ((\alpha_j)_m (y_j)_m) g(x_j, x) + b_m \right]$$

and the dual Lagrangian in equation 16 as

$$(35) \quad L_D = \frac{1}{2} \sum_{i,j=1}^l \sum_m [(y_i)_m (\alpha_i)_m g(x_i, x_j) (y_j)_m (\alpha_j)_m] - \sum_{i=1}^l \sum_m (\alpha_i)_m.$$

If we take this new dual Lagrangian and apply the same steps as in the analytic method of SMO we get

$$(36) \quad \begin{aligned} L_D = & (\alpha_1)_m + (\alpha_2)_m + \text{const} - \frac{1}{2} ((y_1)_m (y_1)_m (x_1^T x_1) \alpha_1^2) \\ & + (y_2)_m (y_2)_m (x_2^T x_2) \alpha_2^2 + 2 (y_1)_m (y_2)_m (x_1^T x_2) \alpha_1 \alpha_2 \\ & + 2 \left( \sum_{i=3}^N (\alpha_i)_m (y_i)_m x_i^T \right) ((y_1)_m x_1 (\alpha_1)_m \\ & + (y_2)_m x_2 (\alpha_2)_m + \text{const}), \end{aligned}$$

which in practical terms is identical to the real-valued one, equation 22. Now, the parameter of the decision functions is not real, but thanks to the equation 34 we can separate this in multiple real valued coefficients. If we take this in consideration and apply the same steps to obtain  $v_j$  in equation 23, we have a partial result of the output, equivalent to the  $m$  part of the total. These conditions mean that we have a virtually identical problem to the real-valued one and that each subspace  $m$  can be computed independently, so we can use parallelism to compute these solutions.

Thanks to the fact that SMO is a derivation of the same dual Lagrangian used in others training methods, we can also use these results and incorporate them to those methods.

One last thing to consider about the Gaussian kernel is its implementation. Based in equation 15 we can program  $|x - y|$  with two nested loops as shown in algorithm 1, so we do not have the need of a special library or tool to work with geometric algebra.

## 6. EXPERIMENTAL RESULTS

In this section we presents some experiments performed in order to to prove the idea behind this work. We did not use classical experiments, like the multidimensional XOR, because they are binary and a test with this method will be equal to using a real-valued SVM.

**Algorithm 2** Generate labelled spirals

---

```

for  $i = 1$  to 100 do
   $s = i/100$ .
   $t = s\pi$ .
   $s1 = +5s \cos(t) + 5s \sin(t)e_1, \quad y = +1 + e_1$ 
   $s2 = -5s \cos(t) - 5s \sin(t)e_1, \quad y = -1 - e_1$ 
   $s3 = -5s \sin(t) + 5s \cos(t)e_1, \quad y = +1 - e_1$ 
   $s4 = +5s \sin(t) - 5s \cos(t)e_1, \quad y = -1 + e_1$ 
end for

```

---

**2D Spirals.** The objectives of this experiment are to simulate a complex-valued SVM and use existing libraries and tools. We choose Scilab [15] using LIBSVM [5] because they are free and well known. LIBSVM implements an SMO-type algorithm [6]. The decision to present this problems using complex numbers, was done in order to make the example easily to test. With the two pseudocodes provided we believe that anyone familiar with the chosen tools can reproduce the result shown, see how easy it was to implement and then apply the algorithm to its own data.

First we generate four spirals with labels in  $\mathbb{G}_{0,1}$ , as shown in algorithm 2. The result can be observed in figure 1.

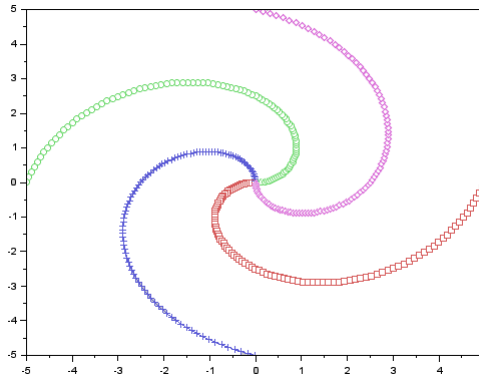


FIGURE 1. The four labelled spirals generated by the algorithm 2.

Then we proceed to combine two SVM, one for the real part and another for the complex part, to get the classification of the four classes. The training was done using the equation 32. The result can be seen in figure 2. We can observe that the four spaces were separated correctly according to the given data, as the red class with label  $y = -1 - e_1$ , is assigned to the "unclassified space".

**Circles intersections.** This experiment was done in C++ using the kernel-adatron algorithm [7] as a training method. The objectives are to simulate a pure quaternion-valued SVM and use a training method different from SMO.

The problem consist in having three overlapping circles, as shown in figure 3, determining our classes.

Each component of the label  $y = \pm e_1 \pm e_2 \pm I$  represents the area of a circle, assigning the positive label to a point inside the circle and the negative to one outside. This is an interesting case, because sometimes there exist common characteristics between different classes, so a Clifford SVM is a very efficient classifier to solve this problem because it classifies these common characteristics as the same in one space and then separates the classes as different in another. The

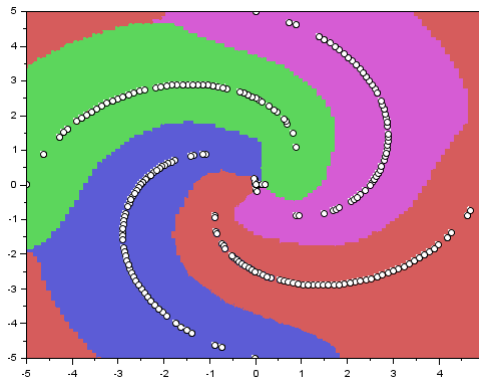


FIGURE 2. Classification of the spirals, the little circles represent the support vectors.

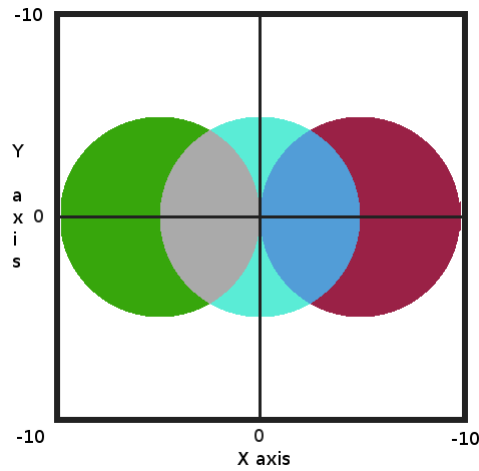


FIGURE 3. The three circles used as templates for our classes. The way the circles intersect with each other gave us six classes, white, green, gray, cyan, blue, and magenta.

training was done using the function in equation 33. We were able to approximate the form given by the circles, as show in figure 4.

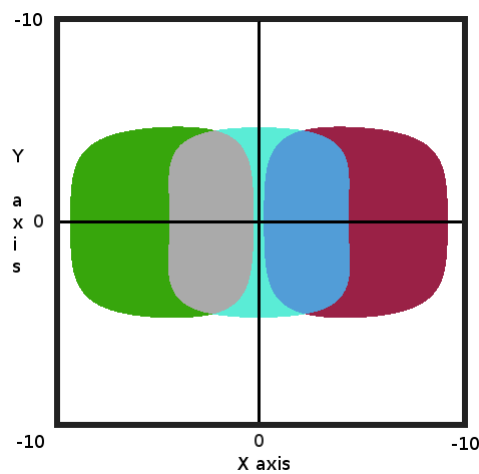


FIGURE 4. Approximation of the six classes using the SVM.



## 7. CONCLUSIONS

We have proved that the approach to Clifford SVMs presented in this paper can be easily implemented using existing libraries, is compatible with different training methods, is easy to use parallelism and requires almost zero knowledge of geometric algebra.

Future work will be focused in studying the derivation of SMO or another similar method that allows the use of any kernel, because there are classification problems that yield better results using different kernels, like the polynomial. Another area of possibly study is the SVMs based in the primal problem, because they do not use a kernel.

## REFERENCES

- [1] M. A. Aizerman, E. A. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. In *Automation and Remote Control*, number 25, pages 821–837, 1964.
- [2] E. Bayro Corrochano. *Geometric computing : for wavelet transforms, robot vision, learning, control and action*. Springer, London, New York, 2010.
- [3] E. Bayro-Corrochano and N. Arana-Daniel. Clifford support vector machines for classification, regression, and recurrence. *IEEE Transactions on Neural Networks*, 21(11):1731–1746, 2010.
- [4] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.*, 2(2):121–167, June 1998.
- [5] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [6] R.-E. Fan, P.-H. Chen, and C.-J. Lin. Working set selection using second order information for training support vector machines. *J. Mach. Learn. Res.*, 6:1889–1918, Dec. 2005.
- [7] T.-T. Frieß, N. Cristianini, and C. Campbell. The kernel-adatron algorithm: a fast and simple learning procedure for support vector machines. In *Machine Learning: Proceedings of the Fifteenth International Conference*. Morgan Kaufmann Publishers, 1998.
- [8] D. Hestenes and G. Sobczyk. *Clifford Algebra to Geometric Calculus : A Unified Language for Mathematics and Physics*, volume 5 of *Fundamental Theories of Physics*. Kluwer Academic Publishers, Dordrecht, 1987.
- [9] B. Hyeran and S.-W. Lee. Applications of support vector machines for pattern recognition: A survey. *Pattern Recognition with Support Vector Machines: First International Workshop*, 2388.
- [10] G. López-González, N. Arana-Daniel, and E. Bayro-Corrochano. Quaternion support vector classifier. In E. Bayro-Corrochano and E. Hancock, editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, volume 8827 of *Lecture Notes in Computer Science*, pages 722–729. Springer International Publishing, 2014.
- [11] E. Osuna, R. Freund, and F. Girosi. An improved training algorithm for support vector machines. pages 276–285. IEEE, 1997.
- [12] C. B. U. C. B. U. Perwass. *Geometric algebra with applications in engineering*. Geometry and computing. Springer, Berlin, 2009.
- [13] J. C. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. Technical report, ADVANCES IN KERNEL METHODS - SUPPORT VECTOR LEARNING, 1998.
- [14] J. C. Platt. Advances in kernel methods. chapter Fast Training of Support Vector Machines Using Sequential Minimal Optimization, pages 185–208. MIT Press, Cambridge, MA, USA, 1999.
- [15] Scilab Enterprises. *Scilab: Free and Open Source software for numerical computation*. Scilab Enterprises, Orsay, France, 2012.
- [16] V. Vapnik. *Estimation of Dependences Based on Empirical Data: Springer Series in Statistics (Springer Series in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1982.
- [17] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.
- [18] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.
- [19] H. Yu and S. Kim. SVM tutorial: classification, regression, and ranking. *Handbook of Natural Computing*, p479-506, Springer, 2012.
- [20] L. Zhang, W. Zhou, and L. Jiao. Complex-valued support vector classifiers. *Digital Signal Processing*, 20(3):944 – 955, 2010.

# SPHERE-TORUS-PATCH BOUNDING VOLUMES USING CONFORMAL GEOMETRIC ALGEBRA

G. López-González<sup>a</sup>, N. G. Arana-Daniel<sup>b</sup>, O. Stasse<sup>c</sup>,  
M. Benallegue<sup>d</sup> and E. J. Bayro-Corrochano<sup>e</sup>

Department of Electrical Engineering and Computer Science  
Center for Research and Advanced Studies (Cinvestav), Guadalajara, Mexico  
gelopez@gdl.cinvestav.mx<sup>a</sup> [corresponding], edb@gdl.cinvestav.mx<sup>e</sup> [presenter]

Department of Electronics and Computation  
Center of Exact Sciences and Engineering (CUCEI), Guadalajara, Mexico  
nancy.arana@cucei.udg.mx<sup>b</sup>

Gepetto Team  
Laboratory for Analysis and Architecture of Systems (LAAS-CNRS), Toulouse, France  
ostasse@laas.fr<sup>c</sup>, mehdi.benallegue@laas.fr<sup>d</sup>

**ABSTRACT.** In this work we use conformal geometric algebra to implement the method called Sphere-Torus-Patch Bounding Volumes. The goal of this method is to bulge a convex shape, creating a strictly convex shape. This allows us to use continuous differentiable distance functions. A function with these characteristics is necessary to use a smooth solver to approach collision related tasks. Furthermore, this method is automated, tunable, efficient and has a good volume ratio.

## 1. INTRODUCTION

The method presented here is based on the seminal work of Escande, Miossec, Benallegue and Kheddar [1]. They presented a method, called Sphere-Torus-Patch Bounding Volumes (STP-BV), for the automation of the bulging and its extension to the 3-D case. The STP-BV is a robust geometric operator that constructs a strictly convex shape of a given geometric model. In other words, it is a method to bulge (i.e., round or curve) a polyhedral convex hull into a strictly convex hull/shape. This generated shape is useful for collision tasks.

Collision related tasks are important in fields such as robotics and computer graphics. One efficient way to do them is to have a function that give us the distance between two objects. If the distance function is continuous differentiable we can use a smooth solver.

So, as discussed earlier, the objective of the STP-BV is to have a strictly convex hull, due to the theorem first proposed in [2], then rediscovered and differently in [3]:

**Theorem 1.1.** The distance function for a pair of objects is not continuous differentiable unless one of the objects is convex and the other is strictly convex.

To explain this we must provide two definitions

**Definition 1.1. Convex object:** An object is convex if for every pair of points within the object, every point on the straight line segment that joins the pair of points is also within the object.

---

The first author is supported by CONACYT-Mexico with the Ph.D. scholarship with number 235318, and the project funding CB 178222.

**Definition 1.2. Strictly convex object:** An object is strictly convex if for every pair of points within the object, every point on the straight line segment that joins the pair of points is also within the object, excluding the boundary of the object.

To clarify these definitions we provide an example in the Figure 1. We can observe that the square is a convex object, but not a strictly convex one, because the line segment between two points on the same side is on its sides. Meanwhile the circle is a strictly convex object, because all the points that form any line segments between two of its points are inside it.

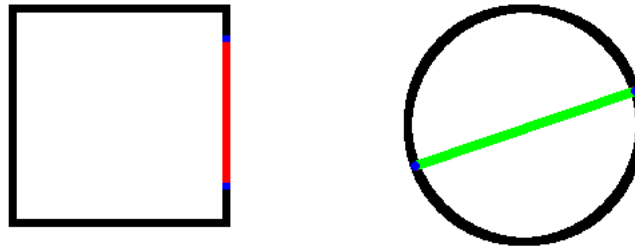


FIGURE 1. The red segment between the two blue points in the square indicates that it is not a strictly convex object.

As explained in [4], non-differentiability of the distance function can be a problem when used in smooth optimization, the optimization solver might not converge in some configurations, as observed in Figure 2.

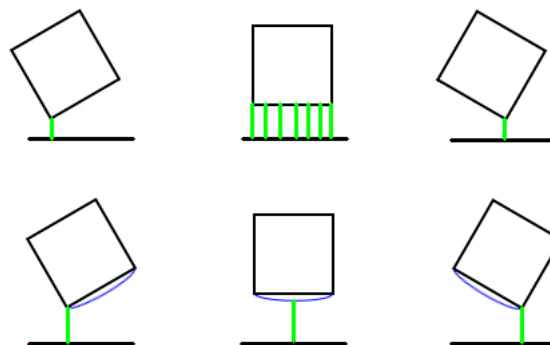


FIGURE 2. When two convex objects have a parallel position, we have a discontinuity in the difference function. This does not happen when one object is strictly convex.

Then, Why not use non-smooth optimization solvers? This is an option. However, with respect to smooth optimization routines, non-smooth ones are less easily available, less complete, and slower. Furthermore, it is worth adding distance constraints in existing schemes that already rely on smooth optimization routines (robotic models are generally smooth), or directly on function derivatives.

The STP-BV uses geometric entities such as spheres and circles, and takes advantage of constructs and operations between them. This makes it a perfect candidate for an implementation using conformal geometric algebra (CGA).

The paper is organized as follows: Section 2 presents previous approaches. Section 3 explains the basics of conformal geometric algebra(CGA) used in this work. Section 4 explains all the elements related to the STP-BV, including the implementation details using CGA. In section 5 we show some results. Finally section 6 is devoted to the conclusions.

## 2. PREVIOUS APPROACHES

Some of the previous works about in the subject use a geometric entity per link. This presents different problems depending on the shape of the selected entity. If the shape is strictly convex, they can work as bounding volumes (covering at best the robot's links). This shapes would guarantee the differentiability of the distance function.

A single sphere or ellipsoid per link exhibits a weak volume ratio fitting, particularly in the case of thin and long robot links.

Superellipsoids and hyperquadrics have a better volume ratio. But computing distance from these shapes to others is too time-consuming [5] for tasks in control.

Furthermore, it is impossible to make the volume ratio fitting for a general convex shape arbitrarily close to 1.

The use of parts of these shapes to better fit each link increases the number of constraints added to the optimization problem. By combining spheres, k-IOS [6] can fit the object tighter, yet the ratio might be no better than 0.5 in pathological cases.

Few other papers addressed the non-differentiability of the distance between convex bodies. Rusaw in [7] used non-smooth analysis in the context of sensory-based planning.

Patel et al. [8] used cylinders as BVs to cover a robot and recognized problems in the control when pairs of cylinders become parallel. They proposed solving this issue by thresholding the output speed of the robot.

Many authors acknowledged this problem when using capsules [9] or convex hulls in optimization problems [10].

## 3. CONFORMAL GEOMETRIC ALGEBRA

Conformal geometric algebra is a representation of the vector space as presented in [11]. A more detailed explanation can be find in [12] The objective is to expand the vector space  $\mathbb{R}^n$  with the Minkowski space  $\mathbb{R}^{1,1}$ .

Let  $\mathbb{R}^{n+1,1}$  be a real vector space, which has associated a Geometric Algebra  $\mathbb{G}_{n+1,1}$ , then its vector basis satisfy:

$$(1) \quad \begin{aligned} e_+^2 &= 1, \\ e_-^2 &= -1, \\ e_i^2 &= 1, \text{ for } i = 1, \dots, n. \end{aligned}$$

In addition, the following properties are satisfied:

$$(2) \quad \begin{aligned} e_+ \cdot e_- &= 0, \\ e_i \cdot e_+ &= 0, \\ e_i \cdot e_- &= 0, \text{ for } i = 1, \dots, n. \end{aligned}$$

So that, we define two null basis:

$$(3) \quad \begin{aligned} e_\infty &= e_- + e_+, \\ e_0 &= \frac{e_- - e_+}{2}, \end{aligned}$$

with the properties:  $e_0^2 = e_\infty^2 = 0$ , and  $e_\infty \cdot e_0 = -1$ .

The set of all null vectors in  $\mathbb{R}^{n+1,1}$  is called the *null cone*, and its intersection with an hyperplane with normal  $e_\infty$ , and containing point  $e_0$ , is a surface called *horosphere*, defined as:

$$(4) \quad \mathbf{N}_e^n = \{x_c \in \mathbb{R}^{n+1,1} : x_c^2 = 0, x_c \cdot e_\infty = -1\}.$$

Now, all points that lie on the horosphere are called *conformal points*, represented by:

$$(5) \quad x_c = x_e + \frac{x_e^2 e_\infty}{2} + e_0,$$

where,  $x_e \in \mathbb{R}^n$ . In addition, three unit pseudoscalars are defined:  $I_e$  for  $\mathcal{G}_n$ ,  $E$  that represents the Minkowski plane, and  $I$  for  $\mathcal{G}_{n+1,1}$ :

$$(6) \quad \begin{aligned} I_e &= e_1 e_2 \cdots e_n, \\ E &= e_\infty \wedge e_0, \\ I &= I_e \wedge E. \end{aligned}$$

Finally, Table 1 summarizes the representation of geometric entities in  $\mathcal{G}_{4,1}$ , where  $\mathbb{I}PNS$  stands for inner-product null space, and  $\mathbb{O}PNS$  stands for outer-product null space.

TABLE 1. Representation of three-dimensional geometric entities in CGA.

Entity	$\mathbb{I}PNS$	$\mathbb{O}PNS$
Circle	$Z = S \wedge \pi = S \wedge S$	$Z^* = \bigwedge_{i=1}^3 x_{ci}$
Line	$L = n_e I_e + m_e e_\infty$	$L^* = e_\infty \wedge \bigwedge_{i=1}^2 x_{ci}$
Sphere	$S = c_c - \frac{\rho^2}{2} e_\infty$	$Z^* = \bigwedge_{i=1}^4 x_{ci}$
Plane	$\pi = n_e + d e_\infty$	$\pi^* = e_\infty \wedge \bigwedge_{i=1}^3 x_{ci}$
Point Pair	$PP = \bigwedge_{i=1}^2 S_i$	$PP^* = \bigwedge_{i=1}^2 x_{ci}$

#### 4. STP-BV

Consider a point cloud  $\mathcal{P}$ , and for  $r \leq 0$ , the set  $\mathcal{P}_r$  of (closed) balls  $\mathfrak{B}(P, r)$  centred at each  $P \in \mathcal{P}$  and with radius  $r$  ( $\mathcal{P}_0 \equiv \mathcal{P}$ ). Let  $\mathfrak{B}_{R,r}(\mathcal{P})$  be the set of all balls with radius  $R$  that encloses every ball of  $\mathcal{P}_r$  ( $R - r$  must be at least the radius of the smallest sphere containing  $\mathcal{P}_r$ ).

**Definition 4.1. STP-BV:** The STP-BV of the set  $\mathcal{P}$  with the raddi  $R$  and  $r$  is the intersection of all the balls of radius  $R$  containing the balls of radius  $r$  centred at each point of  $\mathcal{P}$

$$(7) \quad STP_{R,r} \mathcal{P} = \bigcap_{B \in \mathfrak{B}_{R,r}(\mathcal{P})} B.$$

Henceforth, we ignore the special case where there is a single point in  $\mathcal{P}$  and  $r = 0$ , therefore the STP-BV has a non-empty interior.

4.1. **Geometric Constructions.** In order to create the bounding volume, we must know how to bulge the faces and vertices.

- Faces

Every face composed of 3 points delimiting a triangle, if a face has more that 3 points is subdivided, is curved to a spherical cap.

In order to obtain the sphere needed we took the three conformal points of the face and convert them into spheres as

$$(8) \quad S = x_c - \frac{R^2}{2} e_\infty,$$

and intersect them to get a point pair

$$(9) \quad PP = x_{c1} \wedge x_{c2} \wedge x_{c3},$$

The points in  $PP$  correspond to the two possible centres of the sphere, see Figure 3. To get the correct point we get the normalized line and sphere that form  $PP$  and do

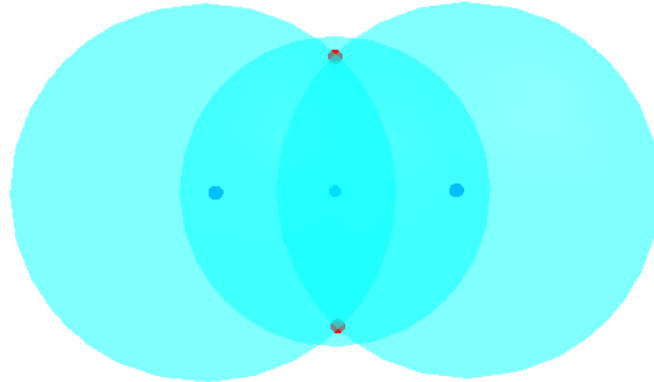


FIGURE 3. The red point pair is the intersection of the 3 spheres.

$$(10) \quad x_E = c_E - n_E r_s,$$

where  $c_E$  is the euclidean center of the sphere,  $r_s$  is the radius and  $n_E$  is the vectorial direction of the line. Another option, if we are using GAALOP, is the provided macro *ExtractSecondPoint* to obtain the point.

- Edge

For the edge, the authors use a torus and found a clever and efficient way to represent it, that can be translated to CGA entities.

The torus is represented by the construct of a circle, that can be decomposed into its center, norm and radius and a sphere which center always lays on the circle.

**Algorithm 1** STP-BV construction

---

**Input:** point cloud  $\mathcal{P}$ , value of  $R$   
**Result:** the STP-BV polyhedron  $\mathcal{H}_R \mathcal{P}$   
 $v, v_1, v_2$  and  $v_3$  are vertices  
 $c, c_1$  and  $c_2$  are sphere centers  
*next* : is a list of sets  $(v_1, v_2, v_3, (v, a))$  sorted by increasing value of  $a$ .  
*output* : a structure describing  $\mathcal{H}_R \mathcal{P}$   
**BuildVolume**( $\mathcal{P}, R$ ) *init*(*next*, *output*)  
**while** no *next*.isEmpty() **do**  
     $(v_1, v_2, v_3, (v, a)) \leftarrow$  *next*.pop()  
    *output*.addFace( $v_1, v_2, v_3$ )  
    **if** *next*.contains( $v_1, v$ ) **then**  
        *next*.delete( $v_1, v$ )  
    **else**  
        *next*.insert( $v_1, v, v_2, \text{nextPoint}(v_1, v, v_2)$ );  
    **end if**  
    **if** *next*.contains( $v, v_2$ ) **then**  
        *next*.delete( $v, v_2$ )  
    **else**  
        *next*.insert( $v, v_2, v_1, \text{nextPoint}(v, v_2, v_1)$ )  
    **end if**  
**end while**  
**Return:** *output*

---

4.2. **Constructing the STP-BV.** We have done all the modifications using CGA in individual functions, returning the same result or an equal one. This means that the general steps of algorithm remain the same.

There exist by definition a one to one ratio between the spheres and tori of the  $STP_{R,r}(\mathcal{P})$  and the faces and edges of  $\mathcal{H}_R(\mathcal{P})$ .

The Algorithm 1 describes the construction of the STP-BV. It is for the most part a gift wrapping algorithm, with spheres instead of planes. We find an initial sphere which associates spheres contain  $\mathcal{P}$ . We then rotate the sphere around the edges of the face until it reaches new points of  $\mathcal{P}$ . This defines new faces and new edges around which to rotate.

Algorithm 2 describes the intermediate functions. The *init* function finds an initial face. Since we can perform only one rotation at a time, we maintain a list of rotations to be completed. Maintaining this list *next* is the main task of Algorithm 1.

Each element of *next* is a set  $(v_1, v_2, v_3, (v_4, a))$  encoding a positive rotation with angle  $a$  of the sphere associated with the face  $v_1 v_2 v_3$ . The rotation around the edge  $v_1, v_2$  must warrant that the face reaches the point  $v_4$ .  $a$  is the angle between  $IC_1$  and  $IC_2$ .

The list *next* is ordered according to the angles  $a$ . It supports the classical operations *pop* and *push*. The operation *contains*( $v_1, v_2$ ) return true if any element of *next* begins with  $(v_1, v_2)$  or  $(v_2, v_1)$ . *delete*( $v_1, v_2$ ) eliminates such an element. *output* is a structure describing a polyhedron (set of vertices, edges and faces). The function *addFace*( $v_1, v_2, v_3$ ) uploads it by adding the face  $v_1 v_2 v_3$  (described counter-clockwise) and the corresponding edges and faces, if not already present. *sphere*( $v_1, v_2, v_3, R$ ) returns the center of the sphere as constructed early. *angle*( $c_1, c_2, v_1, v_2$ ) returns the angle between  $ic_1$  and  $ic_2$ , where  $i$  is the middle point between  $v_1$  and  $v_2$ . *nextPoint*( $v_1, v_2, v_3$ ) looks for the first point  $v$  encountered while rotating the sphere associated with  $v_1 v_2 v_3$  around the edge  $V_1 v_2$ , and returns it along with the angle  $a$  needed to reach it.  $v$  denotes the only point for which the sphere associated with  $v_1 v v_2$  contains  $\mathcal{P}$ , but when there are co-spherical points, any point will do. We discuss the special case  $v = v_3$  later.

**Algorithm 2** intermediate functions

---

```

init(next,out put)
for each  $(v_1, v_2, v_3) \in \mathcal{P}^3$  with  $v_i \neq v_j$  do
   $c \leftarrow \text{sphere}(v_1, v_2, v_3, R)$ 
  if  $\mathcal{P} \subseteq \mathfrak{B}(c, R)$  then
    out put.addFace( $v_1, v_2, v_3$ )
    next.insert( $v_1, v_2, v_3, \text{nextPoint}(v_1, v_2, v_3)$ )
    next.insert( $v_2, v_3, v_1, \text{nextPoint}(v_2, v_3, v_1)$ )
    next.insert( $v_3, v_1, v_2, \text{nextPoint}(v_3, v_1, v_2)$ )
    Return: SUCCESS
  end if
  Return: FAIL
end for

nextPoint( $v_1, v_2, v_3$ )
for each  $v \in \mathcal{P} - \{v_1, v_2\}$  do
   $c_2 \leftarrow \text{sphere}(v_1, v, v_2, R)$ 
  if  $\mathcal{P} \subseteq \mathfrak{B}(c_2, R)$  then
    if  $v = v_3$  then
      Return:  $(v, \pi)$ 
    else
       $c_1 \leftarrow \text{sphere}(v_1, v_2, v_3, R)$ 
      Return:  $(v, \text{angle}(c_1, c_2, v_1, v_2))$ 
    end if
  end if
end for

```

---

At each iteration, Algorithm 1 processes a new face  $v_1v_2$  obtained by turning around  $v_1v_2$ . For the two edges  $v_1v$  and  $vv_2$  we can observe two cases:

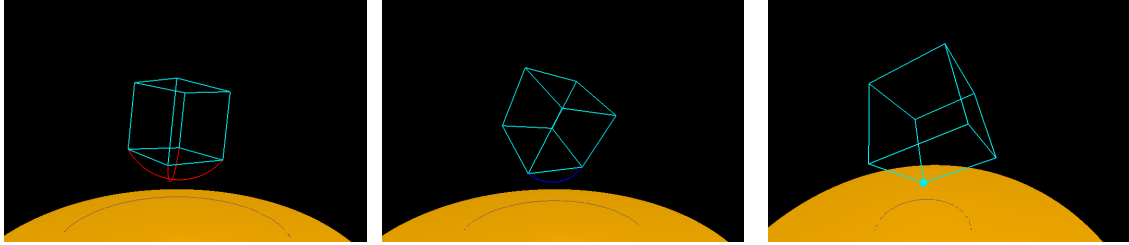
- 1) We encountered the edge before when processing a face  $f$ . There is no need to rotate around this edge anymore because rotating the sphere associated with  $f$  around the edge would give  $v_1v_2$  and vice versa. We remove the rotation from *next*.
- 2) We did not encounter the edge before. We need to perform the rotation around it later so we add it to *next*.

The output of Algorithm 1 is  $\mathcal{H}_R\mathcal{P}$  from which, with the computations described in the previous section, we can build the boundary of  $STP_{R,0}\mathcal{P}$  (spheres and tori with their limits) and thus any  $STP_{R+r}\mathcal{P}$ .

The reason for choosing the smallest rotations first is to ensure the robustness of the algorithm with numerical errors in the case of polygonal faces whose vertices are on the same sphere (co-spherical). In this case, all vertices should be reached at the same time when turning around an edge of this face, and thus, we select the vertex with the lowest index. Yet, this is sometimes not the case because of numerical rounding errors.

Thus, different vertices can be chosen when we reach the face by turning around different edges, resulting most of the times in overlapping triangles. This leads to an algorithm failure. To prevent this, we force the algorithm to finish covering a polygonal face that it already began, by choosing to turn around the edge with the lowest rotation. This avoids us setting a threshold that defines when points are co-spherical and searching for co-spherical points each time there is a rotation around an edge.





(A) A face, the two arcs represent its sphere. (B) A edge, the arc represents its tori. (C) A vertex, the cyan sphere is its bulge.

FIGURE 4. Closest part of the cube to the ground.

In special cases, it can happen that the rotation of the face  $v_1v_2v_3$  around  $v_1v_2$  stops at  $v = v_3$ . For example, this occurs when  $\mathcal{P}$  has only three points, but it can happen with more. In this case, after turning around  $v_1v_2$ , the algorithm will remove the rotations around  $v_1v_3$  and  $v_2v_3$  from next, while one of them might still be needed. To avoid this, *nextPoint* forces the angle to  $\pi$  when detecting the case, to ensure that the corresponding rotation will be the last one processed.

Finally, *init* fails in two cases:

- 1) If  $R$  is smaller than the radius of  $\mathcal{P}$  ( $STP_{R,0}(\mathcal{P})$  does not exist)
- 2) In the case of thin long clouds, where no ball of  $\mathfrak{B}_{R,0}(\mathcal{P})$  is tangent to three points of  $\mathcal{P}$ . In the latter case,  $STP_{R,0}(\mathcal{P})$  is the torus where the two furthest points of  $\mathcal{P}$  define its axis.

## 5. RESULTS

**The cube in CLUCalc.** This experiment is the example of the cube done in CLUCalc [13, 14]. In this exercise we get the closest part of the cube to the floor, represented by a yellow plane. In this example we do a simulation of the construction of the STP-BV to know the exact behaviour of the conformal entities. The aim of this experiment was to know how to replicate the results of the original algorithm using CGA. We did not program a proper distance function for this experiment. Instead we use the planes of the faces as geometric constraints to delimit the valid distance of the spheres to the floor.

The Figure 4 contain the results of this experiments, showing a face, edge, and vertex as the closest part of the cube to the plane.

**The sch library in ROS.** This experiment consist in the modification of the libraries provided by the authors [15] and the integration with ROS. The aim is to use it for robotic simulations using r-viz and gazebo, and eventually test with real robots. The efficient code for the CGA was obtained using GAALOP [16].

The first part of this integration was to visualize the results of the algorithm using rviz. In the Figure 5 we observe two spheres and two boxes, with the line segments representing the witness points. We must clarify that since the box is not a conformal entity, we did not alter the function to create one.

The second part of the integration consist in taking information of the positions of the links from gazebo (these later can be replaced with information from a robot). In Figure 6 we can observe the integration using the files for the robot Reem-C [17].

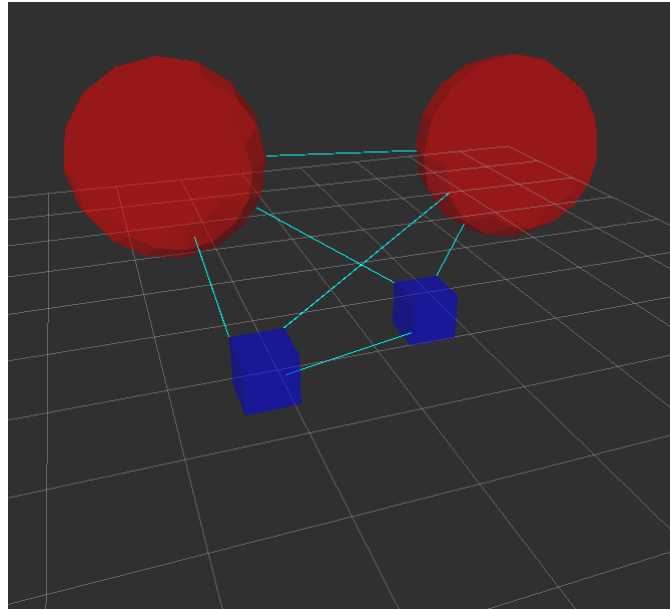


FIGURE 5. Visualization of the algorithm using rviz.

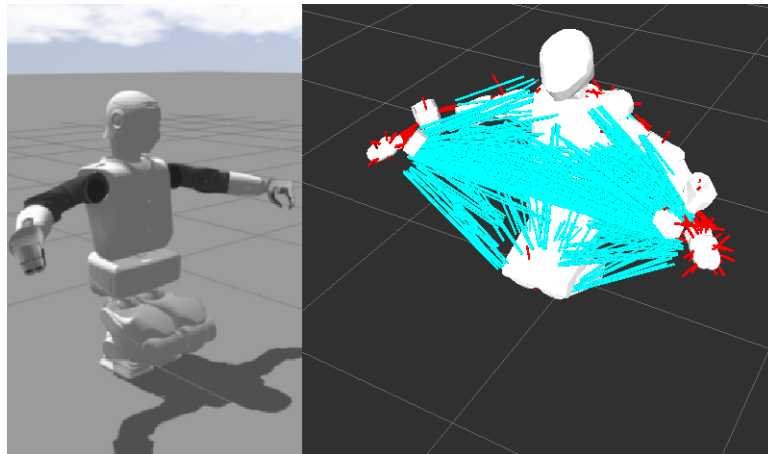


FIGURE 6. In the left we observe the robot simulation in gazebo. In the right the visualization of the algorithm in rviz, where the read lines represent contact between the parts.

## 6. CONCLUSIONS

In this work we presented an implementation of the STP-BV using CGA.

Future work will be focused in finding ways to improve the method using CGA and its integration with other algorithms using CGA.

## REFERENCES

- [1] A. Escande, S. Miossec, M. Benallegue, and A. Kheddar, A Strictly Convex Hull for Computing Proximity Distances With Continuous Gradients. In *IEEE Transactions on Robotics*, volume 30, pages 666-678, 2014.
- [2] E. G. Gilbert and D. W. Johnson, Distance functions and their application to robot path planning in the presence of obstacles. In *IEEE Journal of Robotics and Automation*, volume RA-1, no. 1, pages 21-30, 1985.
- [3] A. Escande, S. Miossec, and A. Kheddar, Continuous gradient proximity distances for humanoids free-collision optimized-postures generation. In *IEEE-RAS International Conference on Humanoid Robots*, volume 1, pages 188-195, 2007.

- [4] F. Bonnans, C. Gilbert, C. Lemaréchal and C. A. Sagastizábal, Numerical Optimization: Theoretical and Practical Aspects. *Springer-Verlag*, 2002.
- [5] N. Chakraborty, J. Peng, S. Akella, and J. Mitchell, Proximity queries between convex objects: An interior point approach for implicit surfaces. *IEEE International Conference on Robotics and Automation*, pages 1910–1916, 2006.
- [6] X. Zhang and Y. J. Kim, k-IOS: Intersection of spheres for efficient proximity query, *IEEE International Conference on Robotics and Automation*, pages 354-359, 2008.
- [7] S. Rusaw, Sensor-based motion planning in SE(2) and SE(3) via non-smooth analysis. *Oxford University Computing Laboratory, Oxford, U.K.* Tech. Rep. PRG-RR-01-13, Sep. 27, 2001.
- [8] R. Patel, F. Shadpey, F. Ranjbaran, and J. Angeles, A collision-avoidance scheme for redundant manipulators: Theory and experiments, *Journal of Robotics Systems* volume 22, no. 12, pages 737–757, 2005.
- [9] A. Dietrich, T. Wimbock, A. Albu-Schaffer, and G. Hirzinger, Integration of reactive, torque-based self-collision avoidance into a task hierarchy *IEEE Transactions on Robotics* volume 28, no. 6, pages 1278–1293, 2012.
- [10] R. Lampariello, D. Nguyen-Tuong, C. Castellini, G. Hirzinger, and J. Peters, Trajectory planning for optimal robot catching in real-time. *IEEE International Conference on Robotics and Automation*, pages 3719–3726, 2011.
- [11] H. Li, D. Hestenes, and A. Rockwood, Generalized homogeneous coordinates for computational geometry. *Geometric Computing with Clifford Algebras*, 2001.
- [12] E. J. Bayro-Corrochano *Geometric Computing: for Wavelet Transforms, Robot Vision, Learning, Control and Action.* Springer, 2010.
- [13] C. Perwass, *Geometric algebra with applications in engineering.* Springer, 2009.
- [14] C. Perwass, CLUCalc. 2006, available at <http://www.clucalc.info/>
- [15] SCH-Core. 2014, available at <https://github.com/jrl-umi3218/sch-core/>.
- [16] C. Schwinn, D. Hildenbrand, F. Stock and A. Koch, Gaalop 2.0 – A Geometric Algebra Algorithm Compiler. *Proceedings GraVisMa Workshop*, 2010.
- [17] Reem-C. 2014, available at <http://wiki.ros.org/Robots/REEM-C>

# FPGA IMPLEMENTATION OF A GEOMETRIC VOTING SCHEME FOR THE EXTRACTION OF GEOMETRIC ENTITIES FROM IMAGES

G. Soria-García , G. Altamirano-Gómez, S. Ortega-Cisneros, and E. Bayro-Corrochano<sup>b</sup>

Department of Electrical Engineering and Computer Science  
CINVESTAV, Campus Guadalajara, Jalisco, México  
{jgsoria, galtamira, sortega }@gdl.cinvestav.mx

<sup>b</sup> edb@gdl.cinvestav.mx[presenter, corresponding]

**ABSTRACT.** In this work, we present a Field Programmable Gate Array (FPGA) implementation of a Conformal Geometric Algebra Voting Scheme for detection of circles and lines in images with real objects. All geometric operations, such as meet, join, and transformation of entities, are computed in hardware using an FPGA.

The voting scheme consists in two stages; in this hardware version, we have implemented the first stage of the algorithm that operates over a neighborhood in the image. The top design consists of five main hardware sub-modules that interact between them using intermediate memories.

The design has been validated comparing the results of the FPGA with the results previously obtained in a software reference application using real image data. We were able to observe that by using separated memories for intermediate results, the main units of computation can operate in a kind of pipeline mode over separated groups of data. This characteristic allows a better performance in the algorithm execution.

## 1. INTRODUCTION

When humans look at a scene, our visual system is capable of detecting prominent features and finding meaningful relationships between them. The process of structuring visual information into coherent units is called perceptual organization, and its principles for the human visual system are stated in Gestalt psychology theory [1, 2].

From the computer science perspective, traditional methods for detection of geometric entities resort to voting schemes such as the Hough transform [3] and Tensor Voting [4]. Using Conformal Geometric Algebra (CGA), a generalization of these methods was proposed by Altamirano and Bayro [5]; such a generalization is made via an inner product of a set of tokens,  $t_i$ , with respect to a flag,  $F$ :

$$\frac{1}{N} \sum_{i=1}^N W(a_1, \dots, a_m, t_i) \frac{(F \cdot t_i)^2}{|F|^2}, \quad (1)$$

and  $W$  is a function that maps a set of parameters  $\{a_1, \dots, a_m, t_i\}$  to a scalar value, and codifies perceptual properties according to Gestalt principles. Thus, the role of a voting scheme is to find the geometric structure  $F$  that minimizes Equation 1; thus,  $F$  is supported by a set of tokens  $\{t_1, t_2, \dots, t_N\}$ , and at the same time satisfies perceptual constraints codified in  $W$ .

For computing  $F$ , Altamirano and Bayro [5] propose a two stage methodology: a local voting process, which extracts salient geometric entities supported in a local neighborhood, and a global voting process, which clusters the output obtained by the local voting process.

Since the local voting step is performed in each edge-pixel of the image, this method is highly computationally demanding for this reason we propose a hardware implementation using FPGA.

The top design consists of five main hardware sub-modules. The first sub-module called *voting module* computes one vote for each neighbor in the current neighborhood. The *Intersection*

*module* computes all possible intersections between pairs of lines obtained by the *voting module*. Then, those intersections are processed by the *DBSCAN module* in order to be classified in clusters using integer labels. The *Cluster mean module* takes those elements classified in clusters and computes the mean point of the cluster as well as the overall density. Finally, the *Cluster select module* returns those clusters with maximum density. All these modules interchange data using intermediate on-chip memories.

This implementation is targeted to a ZC206 evaluation board featured with the new Xilinx FPGA Zynq 7000 based on a SoC (system on chip) architecture running at 200 MHz. The design has been validated by comparing the results of the FPGA with the results previously obtained in a software reference application using real images data.

The organization of the paper is as follows: Section 2 presents an introduction to CGA, the Voting Method is described in Section 3.1, the FPGA implementation is presented in Section 4, and experimental results are shown in Sections 5 and 6, and finally, Section 7 is devoted to conclusions and future work.

## 2. CONFORMAL GEOMETRIC ALGEBRA

Conformal Geometric Algebra (CGA) allows the representation of geometric entities and their properties by embedding a euclidean space  $\mathbb{R}^n$  in a higher dimensional vector space  $\mathbb{R}^{n+1,1}$ . Here, we summarize the construction of CGA; for a detailed study see [6].

Let  $\mathbb{R}^{n+1,1}$  be a real vector space, which is associated with geometric algebra  $\mathcal{G}_{n+1,1}$ , then its vector bases satisfy:  $e_+^2 = 1$ ,  $e_-^2 = -1$ , and  $e_i^2 = 1$ , for  $i = 1, \dots, n$ . In addition, the following properties are satisfied:  $e_+ \cdot e_- = 0$ ,  $e_i \cdot e_+ = 0$ , and  $e_i \cdot e_- = 0$ , for  $i = 1, \dots, n$ .

Then, we define two null bases:  $e_\infty = e_- + e_+$ , and  $e_0 = 0.5(e_- - e_+)$ , with the properties:  $e_0^2 = e_\infty^2 = 0$ , and  $e_\infty \cdot e_0 = -1$ .

The set of all null vectors in  $\mathbb{R}^{n+1,1}$  is called the *null cone*, and its intersection with an hyperplane with normal  $e_\infty$ , and containing point  $e_0$ , is a surface called *horosphere*, defined as:

$$\mathbf{N}_e^n = \{x_c \in \mathbb{R}^{n+1,1} : x_c^2 = 0, x_c \cdot e_\infty = -1\} . \quad (2)$$

Now, all points that lie on the horosphere are called *conformal points*, represented by:

$$x_c = x_e + 0.5x_e^2 e_\infty + e_0 , \quad (3)$$

where,  $x_e \in \mathbb{R}^n$ . In addition, three unit pseudoscalars are defined:  $I_e$  for  $\mathcal{G}_n$ ,  $E$ , which represents the Minkowski plane, and  $I$  for  $\mathcal{G}_{n+1,1}$ :

$$I_e = e_1 e_2 \dots e_n; E = e_\infty \wedge e_0; I = I_e \wedge E . \quad (4)$$

Finally, Table 1 summarizes the representation of geometric entities in CGA  $\mathcal{G}_{3,1}$ , where IPNS stands for inner-product null space, and OPNS stands for outer-product null space.

TABLE 1. Representation of geometric entities in CGA  $\mathcal{G}_{3,1}$ .

Entity	IPNS	OPNS
Circle	$Z = c_c - \frac{\rho^2}{2} e_\infty$	$Z^* = \bigwedge_{i=1}^3 x_{ci}$
Line	$L = n_e + d e_\infty$	$L^* = e_\infty \wedge \bigwedge_{i=1}^2 x_{ci}$
Point Pair	$PP = \bigwedge_{i=1}^2 Z_i$	$PP^* = \bigwedge_{i=1}^2 x_{ci}$

## 3. CONFORMAL GEOMETRIC ALGEBRA VOTING SCHEME

The essential components of this approach are summarized in two methodologies: to represent information using CGA, and to communicate information via a voting process.

**3.1. Representation of Information Using CGA.** For applications with images, we consider them as a vector space  $\mathbb{R}^2$ . Then, a pixel  $p_i$  is represented as a conformal point of CGA, and perceptual structures such as circles and lines in the image are represented as elements of CGA  $\mathcal{G}_{3,1}$ , according to Table 1. Then, a pixel  $p_i$  lying on a circle or line,  $S$ , satisfies:  $S \cdot p_i = 0$ . Consequently, for a set of pixels  $\{p_1, p_2, \dots, p_N\}$ , the circle or line that satisfies  $S \cdot p_i = 0$ , for  $i = 1, 2, \dots, N$  define a minimum for:

$$\frac{1}{N} \sum_{i=1}^N W(a_1, \dots, a_m, t_i) \frac{(S \cdot p_i)^2}{|S|^2}, \quad (5)$$

where  $W$  is a function used to codify perceptual properties according to the Gestalt principles of proximity, co-curvilinearity, and constancy of curvature, and is defined as [4]:

$$W(s, \rho, c, \sigma) = \exp\left(-\frac{s^2 + c\rho^2}{\sigma^2}\right), \quad (6)$$

where  $c$  controls the degree of decay with curvature,  $\sigma$  determines the neighborhood size of the voting,  $s$  represents the arc length, and  $\rho$  the curvature. The values of  $c$  and  $\sigma$  are taken as input parameters, while  $s$  and  $\rho$  are computed as follows:

$$s = \frac{\theta d}{\sin \theta}, \quad \rho = \frac{2 \sin \theta}{d}, \quad (7)$$

where  $d$  represents the Euclidean distance between  $p_0$  and  $p_i$ , and  $\theta$  represents the angle between the tangent of the circle at  $p_0$  and the line defined by  $p_0$  and  $p_i$ , given by the equation:

$$\theta = \arctan\left(\frac{v_i}{u_i}\right) - \arctan\left(-\frac{c_x}{c_y}\right). \quad (8)$$

**3.2. Communication of information.** This stage has two parts: a local voting process, which extract salient geometric entities supported in a local neighborhood, and a global voting process, which clusters the output obtained by the local voting process.

**3.2.1. Local Voting.** In this stage, we took an edge image, selected an edge pixel, denoted by  $p_0$ , and defined a neighborhood, denoted by  $P_0$ . Without loss of generality, we set  $p_0$  as origin of the coordinate system; then, a pixel  $p_i$  has image coordinates  $(u_i, v_i)$ .

Next, each pixel in the neighborhood  $p_0$  casts a vote in the form of a line:

$$l_i = -\frac{u_i}{|p_{ie}|}e_1 - \frac{v_i}{|p_{ie}|}e_2 + \frac{|p_{ie}|}{2}e_\infty, \quad (9)$$

After that, we map each point  $p_i$  in  $P_0$  to a line with density, denoted by  $l_i$ , and obtain a set of lines denoted by  $L_0$ . The intersection of each pair of lines in  $L_0$ , is the pair of points:

$$c_{0ij} \wedge e_\infty = (l_i \wedge l_j)^*, \quad (10)$$

and from this result we extract  $c_{0ij}$ . If the result is zero, then the lines meet at a point at infinity, hence  $p_0$ ,  $p_i$  and  $p_j$  are collinear. In this case, the vote is the line passing through  $p_0$ ,  $p_i$  and  $p_j$ :

$$n_{0ij} = (p_0 \wedge p_i \wedge e_\infty)^*. \quad (11)$$

In the voting space, each  $c_{0ij}$  or  $n_{0ij}$  is a point, and its saliency value is given by:

$$W_{0ij} = W(s_i, \rho_i, c, \sigma) + W(s_j, \rho_j, c, \sigma). \quad (12)$$

Once we have computed all possible intersection of lines in set  $L_0$ , we cluster resulting points using DBSCAN algorithm [7]. The next step is to compute the perceptual saliency of each cluster:

$$\bar{W} = \sum_i W_i. \quad (13)$$

Finally, we select clusters that surpass a threshold value and compute the weighted mean of points in the selected cluster, for circles:

$$\bar{c}_{0ij} = \frac{1}{\bar{W}} \sum_i \sum_j (W_{0ij} c_{0ij}). \quad (14)$$

and the radius is given by the magnitude of  $\bar{c}_{0ij}$ . On the other hand, if the winner of the voting process is a line, we compute:

$$\bar{n}_{0ij} = \frac{1}{\bar{W}} \sum_i \sum_j (W_{0ij} n_{0ij}). \quad (15)$$

and the Hesse distance or foot is zero.

3.2.2. *Global Voting.* Let  $\hat{S}$  and  $\hat{L}$ , be the output of the local voting process, i.e. a set of circles and lines, respectively; then, we apply DBSCAN algorithm to separate into clusters: circles with the same center and radius, and lines with the same normal and Hesse distance. Thus,  $\hat{S}$  is partitioned into subsets  $\hat{S}_1, \dots, \hat{S}_q$ , where each  $\hat{S}_k$  is a cluster of circles obtained by DBSCAN. In the same way,  $\hat{L}$  is partitioned into subsets  $\hat{L}_1, \dots, \hat{L}_q$ , where each  $\hat{L}_k$  is a cluster of lines, obtained by DBSCAN. Finally, we compute the weighted mean of each cluster  $\hat{S}_1, \dots, \hat{S}_q, \hat{L}_1, \dots, \hat{L}_q$  and select those entities that surpass a threshold value.

#### 4. HARDWARE IMPLEMENTATION

Field Programmable Gate arrays (FPGAs) are electronic devices based on a matrix array of logic blocks whose function or behavior and their interconnections can be programmed to emulate almost every digital circuit or system. In addition to these logic blocks, FPGAs also contain other dedicated blocks, such as: on-chip memories and optimized blocks for arithmetic operations, mainly used in digital signal processing (DSP) applications, as well as blocks dedicated to the good management of the clock signals through all the device. Finally, the I/O blocks are located at the device boundaries and are responsible for the real world interaction. They can manage different voltage standards as well as high impedance states [8]. A general FPGA architecture is depicted in Figure 1.

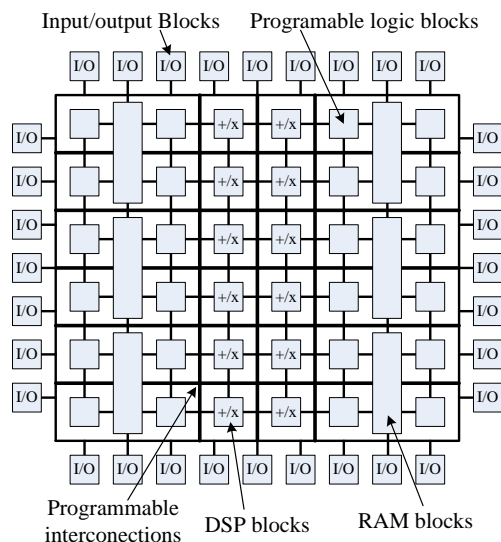


FIGURE 1. Generic architecture of an FPGA.

The above features allow FPGAs to host multiple instances (copies) of a module. This way, we can process multiple sets of data in parallel, obtaining run times which are generally lower

those for the same process executed in a PC. This is the main motivation for implementing the geometric voting scheme in an FPGA.

In the following subsections we present details of the implementation.

**4.1. Methodology.** Traditionally, FPGAs are "programmed" using hardware description languages (HDLs). The most important HDLs used in the industry and education are VHDL and Verilog. In order to program FPGAs, it is necessary to have knowledge about digital circuit design in order to use any of these languages, this is a problem for software programmers interested in implementing their algorithms in FPGAs. For that reason, many tools that convert high level code in HDL were developed, this process is known as high level synthesis (HLS). Practically all modern computer assisted design (CAD) tools for FPGAs have this capability. HLS has considerably decreased the development time for implementing complex digital systems in FPGAs. For these reasons, we have chosen HLS to implement our algorithm.

**4.2. Developing platform.** Our design is planned to be implemented in the ZC706 developing board from Xilinx Inc. This board is featured with the new FPGA family based on a system on chip (SoC) architecture. In addition to the programmable logic blocks included by the typical FPGAs, this SoC family has two ARM cores in the same silicon die, allowing a better integration in projects mixing software and hardware design.

Xilinx Inc provides the Vivado HLS tool for high level synthesis in its products. This tool admits C, C++, and SystemC as high level languages. It is important to consider that only a subset of the languages is supported for synthesis. There are features that cannot be implemented in hardware, for instance, memory dynamic allocation is not supported. The size of all arrays must be known in compilation time, as the arrays are implemented as embedded memories with fixed size. Another point to consider in HLS code is the size of the data type used. It is important to use variables with the minimum necessary length in bits to store values in order to avoid wasting valuable logical resources in the FPGA. HLS tools provide data types with arbitrary length in bits to guarantee optimal resource utilization.

**4.3. Architecture.** In order to minimize memory utilization in our design, all real values are stored using simple precision floating point format (32 bits), while integers are stored using arbitrary lengths, noted above. The algorithm description is designed to process one neighborhood at a time, allowing multiple neighborhoods to be processed in parallel for future implementations using multiple instances of the module presented here. The architecture results in 5 main sub-modules and 5 memory arrays, as shown in Figure 2.

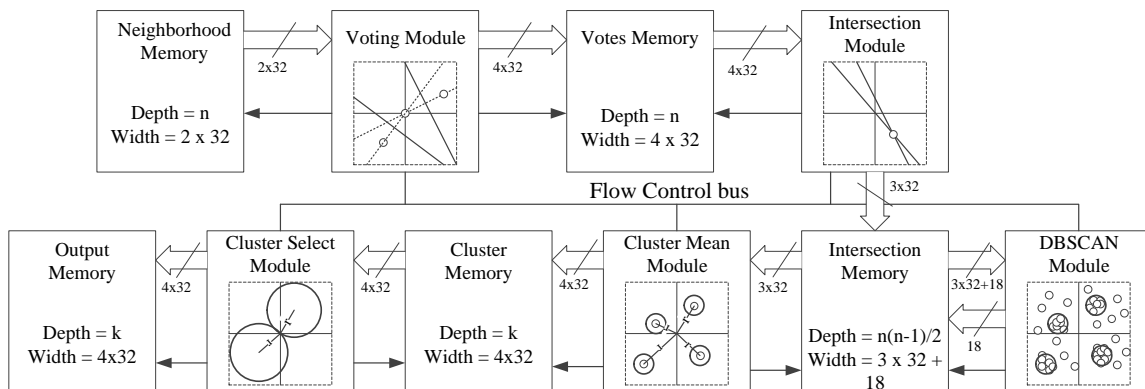


FIGURE 2. Illustrative block diagram of the local voting unit architecture in FPGA.

At the input, there is a memory *neighborhood memory* where the coordinates of the elements in the neighborhood are stored in pairs (x,y). Floating point values are used to store these



coordinates, as shown in Figure 3. The coordinate pair corresponding to  $p_0$ , is stored in the first address of the *neighborhood memory* (address 0), and the rest of points are stored with coordinates relative to  $p_0$  due to the origin shifting to the center of the neighborhood. This memory has a depth of  $n$ , where  $n$  is the maximum amount of neighbors per neighborhood supported by the module, and the width is  $2 \times 32$  bits.

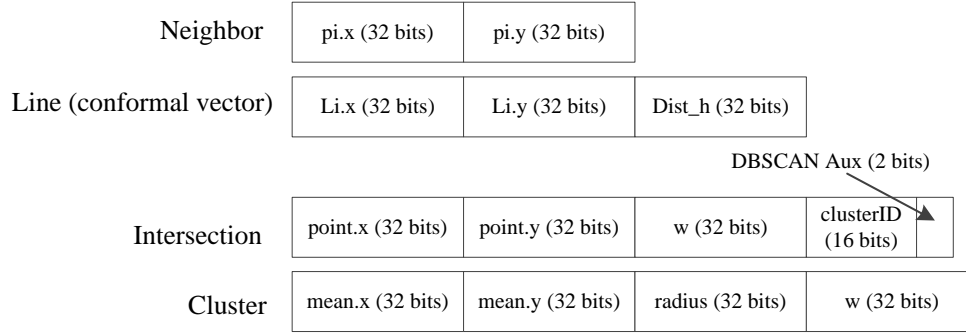


FIGURE 3. Storage format for data types used in the FPGA implementation.

The first sub-module then takes each element in *neighborhood memory* and generates the four coefficients of the conformal vector corresponding to the vote line according to Equation (9). The results are stored in *votes memory*, whose depth is also equal to  $n$  but with a width of  $3 \times 32$  bits since  $e_0$  coefficient is always equal to zero; therefore, it is not necessary to store that value. The second sub-module takes the lines in pairs and computes every possible intersection between them using Equation (10). Furthermore, in this step the perceptual prominence is also computed using Equation (12). The equations of the coefficients for the resulting bivector are computed as follows: let  $l_1$  and  $l_2$  conformal vectors describing lines in  $\mathcal{G}_{3,1}$ :

$$\begin{aligned} l_1 &= nx_1 e_1 + ny_1 e_2 + dh_1 e_\infty + 0e_0, \\ l_2 &= nx_2 e_1 + ny_2 e_2 + dh_2 e_\infty + 0e_0 \end{aligned} \quad (16)$$

Then applying wedge product between  $l_1$  and  $l_2$  results in the bivector:

$$l_1 \wedge l_2 = a_1 e_{12} + a_2 e_{1\infty} + a_3 e_{10} + a_4 e_{2\infty} + a_5 e_{20} + a_6 e_{\infty 0}, \quad (17)$$

where:

$$\begin{aligned} a_1 &= (nx_1 \times ny_2) - (ny_1 \times nx_2), & a_2 &= (nx_1 \times dh_2) - (dh_1 \times nx_2), \\ a_3 &= (nx_1 \times 0) - (0 \times nx_2) = 0, & a_4 &= (ny_1 \times dh_2) - (dh_1 \times ny_2), \\ a_5 &= (ny_1 \times 0) - (0 \times ny_2) = 0, & a_6 &= (dh_1 \times 0) - (0 \times dh_2) = 0, \end{aligned} \quad (18)$$

The result of applying dualization to Equation (17) is:

$$(l_1 \wedge l_2)^* = b_1 e_{12} + b_2 e_{1\infty} + b_3 e_{10} + b_4 e_{2\infty} + b_5 e_{20} + b_6 e_{\infty 0}, \quad (19)$$

where:

$$\begin{aligned} b_1 &= -a_6 = 0, & b_2 &= a_4, & b_3 &= -a_5 = 0, \\ b_4 &= -a_2, & b_5 &= a_3 = 0, & b_6 &= a_1. \end{aligned} \quad (20)$$

To extract  $c_{0ij}$  from Equation (19) according to Equation (10), we need only the following coefficients:  $b_2$ ,  $b_4$  and  $b_6$ . Once the center of the circle has been computed, the result is stored in *intersection memory*, using the format for intersections shown in Figure 3. This element

stores the center of the circle, the perceptual prominence  $w$ , as well as the cluster it belongs to and 2 extra boolean auxiliary fields to DBSCAN algorithm. Thus, the depth of Memory 3 is equal to  $\frac{n \times (n-1)}{2}$  in order to be able to deal with the quantity of intersections, and the width is equal to  $(3 \times 32 + 16 + 2 \times 1)$  bits according to Figure 3.

The next sub-module executes the DBSCAN algorithm over the data in *intersection memory*; this block modifies the clusterID field of each element in order to classify the intersections in clusters. When a point is considered noise, the block labels that intersection as -1. The sub-module called *means* computes the overall density of each cluster using Equation (13) as well as the weighted mean center of each cluster using Equation (14), and the results are stored in *cluster memory* using the format shown in Figure 3. An element of type cluster stores the weighted mean center of a circle, and its density as well as the radius that is calculated using the euclidean distance from the center to  $p_0$ . Thus, the depth of *cluster memory* is  $k$ , where  $k$  is the maximum number of clusters supported by the design and is assigned by the developer. The width is equal to  $(4 \times 32)$  bits according to the format shown in Figure 3.

Finally, the last sub-module finds the cluster with maximum density and selects all the clusters that surpass this value with a tolerance of 1 unit. The clusters fulfilling this criteria are stored in the output memory, which has the same geometry as *cluster memory*. The clusters stored in the *output memory* are then the winners of the local voting process in the current neighborhood.

## 5. EXPERIMENTAL ANALYSIS

In this section, we present an experimental comparison between the output of the FPGA implementation of the algorithm, and a C-language version implemented in a personal computer (PC).

For this first work, the FPGA implementation was executed in the Vivado simulator provided by Xilinx Inc, using the ZC706 board as the target device running at 100Mhz. On the other hand, the C-code implementation, was executed on a laptop computer with a processor Intel Core i7-3630QM (3rd generation), 16GB of RAM DDR@1600Mhz, and a Linux Mint 17 operating system with kernel version 3.13.0-24-generic.

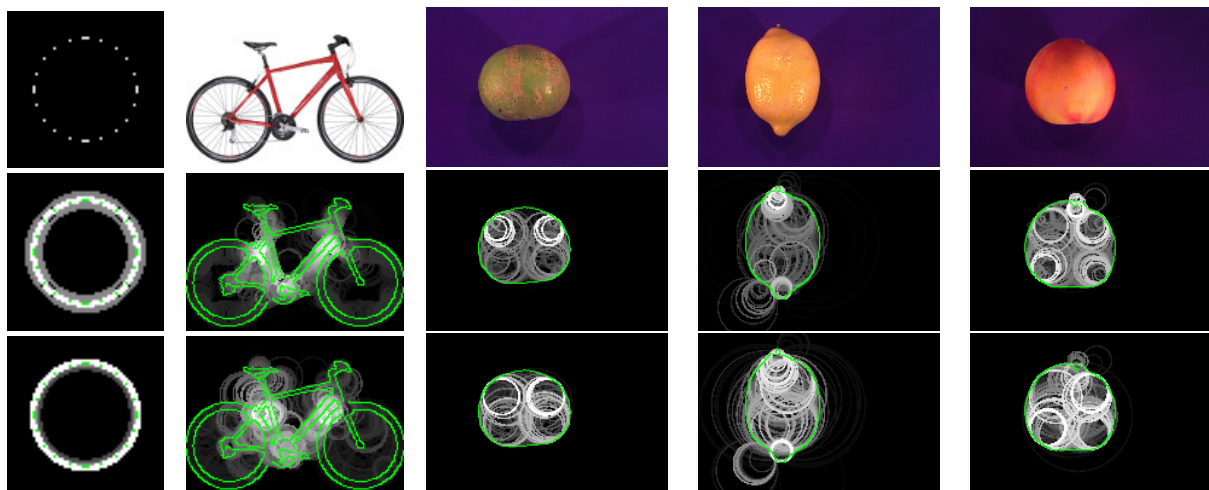


FIGURE 4. Experimental results. Input images in the first row, the output obtained by the FPGA in the second row, and the output obtained by the PC in the third row

Figure 4 shows: the input images in the first row, the output obtained by the FPGA in the second row, and the output obtained by the PC algorithm in the third row. Each column shows the result for different input images. In the case of RGB input images, they were pre-processed using a mean filter and a Canny edge detector [9]. In the case of output images, green pixels

represent edge-pixels of the input image, and grayscale pixels represent the output of the algorithm. Pixels in white represent the circles with maximum saliency, obtained by the local voting algorithm.

Although the FPGA implementation uses data types half as precise as the PC implementation (32bits float vs 64 bits double), Figure 4 shows that both implementations obtain similar outputs. In particular, the first experiment took approximately half the time in the FPGA implementation.

## 6. FPGA RESOURCE OCCUPATION

Table 2 shows the resource occupation in the FPGA for the implementation of the voting scheme algorithm.

TABLE 2. Resource occupation in FPGA

Resource	BRAM.18K	DSP48E	FF	LUT
Expression	-	-	0	238
Instance	226	214	53029	119293
Memory	659	-	0	0
Multiplexer	-	-	-	561
Register	-	-	603	-
Total	885	214	53632	120092
Available	1090	900	437200	218600
Utilization (%)	81	23	12	54

As we can see, the most used resource is memory, this is expected due to all the memory arrays shown in Figure 2, for future implementations that will include lines detection and global voting, internal resources will not be sufficient, therefore we will need to use external resources as 1GB DDR memory mounted in the development board.

## 7. CONCLUSIONS AND FUTURE WORK

We have presented a work in progress regarding an FPGA implementation of a voting method for feature extraction in images. All geometric operators such as meet, join, and conformal transformations are implemented in hardware in an application-driven device. Of particular interest is the output of the local voting, which is a set of local descriptors of shape; these features are useful in robotic and computer vision applications such as 3D-reconstruction, object manipulation and grasping, and SLAM, among others.

Experiments show that the FPGA implementation can use data types with the half the precision without compromising accuracy in the computed entities. This is an important aspect due to the restrictions of resources in FPGAs.

Future work will focus on the following aspects: integrating lines detection for local voting stage, implementing the global voting stage for both circles and lines, downloading the algorithm to the FPGA and performing the experiments in the board, and modifying the architecture for parallel neighborhood processing.

## REFERENCES

- [1] K. L. Boyer and S. Sarkar, "Guest editors' introduction: Perceptual organization in computer vision: Status, challenges, and potential," *Computer Vision and Image Understanding*, vol. 76, no. 1, pp. 1–5, 1999.
- [2] J. Wagemans, J. H. Elder, M. Kubovy, S. E. Palmer, M. A. Peterson, M. Singh, and R. von der Heydt, "A century of gestalt psychology in visual perception: I. perceptual grouping and figure-ground organization," *Psychological bulletin*, vol. 138, no. 6, p. 1172, 2012.

- [3] R. O. Duda and P. E. Hart, "Pattern recognition and scene analysis," 1973.
- [4] P. Mordohai and G. Medioni, "Tensor voting: a perceptual organization approach to computer vision and machine learning," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 2, no. 1, pp. 1–136, 2006.
- [5] G. E. Altamirano-Gómez and E. Bayro-Corrochano, "Conformal geometric method for voting," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pp. 802–809, Springer, 2014.
- [6] H. Li, D. Hestenes, and A. Rockwood, "Generalized homogeneous coordinates for computational geometry," in *Geometric Computing with Clifford Algebras*, pp. 27–59, Springer, 2001.
- [7] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise.," in *Kdd*, vol. 96, pp. 226–231, 1996.
- [8] I. Kuon, R. Tessier, and J. Rose, "Fpga architecture: Survey and challenges," *Foundations and Trends in Electronic Design Automation*, vol. 2, no. 2, pp. 135–253, 2008.
- [9] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-8, pp. 679–698, Nov 1986.



# QUANTIZING GAUGE THEORY GRAVITY

**P. Cameron**

Strongarm Studios  
Mattituck, NY USA  
peter.cameron@protonmail.ch

ABSTRACT. The shared background independence of spacetime algebra and the impedance approach to quantization, coupled with the natural gauge invariance of phase shifts introduced by quantum impedances, opens the possibility that identifying the geometric objects of the impedance model with those of spacetime algebra will permit a more intuitive understanding of the equivalence of gauge theory gravity in flat space with general relativity in curved space.

## INTRODUCTION

In the preface to the newly published second edition of his seminal text[1], Professor Hestenes makes four “bold and explicit... claims for innovation” in SpaceTime Algebra:

- STA enables a unified, **co-ordinate free** formulation for all of relativistic physics, including the Dirac equation, Maxwell’s equation, and General Relativity.
- Pauli and Dirac matrices are represented in STA as **basis vectors** in space and spacetime respectively, with no necessary connection to spin.
- STA reveals that the **unit imaginary** in quantum mechanics has its origin in spacetime geometry.
- STA reduces the mathematical divide between classical, quantum, and relativistic physics, especially in the use of **rotors** for rotational dynamics and gauge transformations.

The preface encourages making such claims, lest the innovations be overlooked. “Modestly presenting evidence and arguing a case is seldom sufficient.”[1] In this spirit, the following five bold and explicit claims are made for the Impedance Approach to quantization:

- IA is **background independent** - This fundamental connection with STA goes deep, to the **co-ordinate free** formulation essential for quantum gravity[2, 3, 4]. In STA, motion is described with respect to the object in question rather than an external coordinate system. Similarly, impedances are calculated from Mach’s principle applied to the two body problem[5, 6]. Motion is described with respect to one of the two bodies. IA is background independent. There is no third body, no independent observer to whom rotations can be referenced, only spin.
- IA contains **gravity** - Matching quantized impedances at the Planck scale reveals an exact identity between electromagnetism and gravity [7]. By far the most imprecise of the fundamental constants, the gravitational constant  $G$  cancels out in the calculation.
- IA is **gauge invariant** - Impedances shift phase. Quantum impedances shift quantum phase. In gauge theories phase coherence is maintained by covariant derivatives. In IA coherent phase shifts are introduced by the impedances. IA is gauge invariant.
- IA is **finite** - In IA the quantization scale is taken to be the electron Compton wavelength. Low and high energy impedance mismatches provide natural cutoffs as one moves away from the quantization length. No need to renormalize. IA is finite.
- IA is **confined** - Reflections from the natural cutoffs of the impedance mismatches provide confinement to the vicinity of the quantization length.

The presence of gravity in IA in conjunction with the coordinate-free background independence common to STA and IA invites the conjecture[8] that scale dependent impedances (Coulomb,

dipole, scalar Lorentz,...) of IA can be associated with the translation gauge field of gauge theory gravity [9, 10, 11, 12, 13, 14] and scale invariant impedances (quantum Hall/vector Lorentz, chiral, centrifugal, Coriolis, three body,...) with the rotation gauge field.

Let us see how this comes about, and what consequences might follow.

### HISTORICAL PERSPECTIVE ON THE IMPEDANCE APPROACH

Given the practical everyday utility of the impedance concept in technical applications, it is not surprising that one finds the most helpful historical introductions and expositions not in the academic literature, but rather in that of technologically advanced industries, where proper application of the concept is *essential for economic success* [15, 16, 17, 18].

This inadvertent divorce of theoretical from practical has profound consequences for quantum field theory (QFT), where the Hamiltonian and Lagrangian formalisms focus upon conservation of energy and its flow between potential and kinetic, rather than upon that which governs the flow, the impedances.

The most rudimentary example can be found at the foundation of quantum electrodynamics (QED), in the photon-electron interaction. The formidable breadth of the crack through which the impedance concept has fallen becomes apparent when one considers that the near field photon impedances [19] shown in figure 1 cannot be found in the physics textbooks of electricity and magnetism, QED, or QFT [20].

*What governs the flow of energy in photon-electron interactions is explicitly absent from the formal education of the PhD physicist.*

The significance can be seen by examining energy flow between a 13.6 eV photon and the quantum Hall impedance of the electron. The figure illustrates the scale-dependent photon near-field dipole impedance that permits energy to flow without reflection between Rydberg and Bohr, between photon and hydrogen atom. However, what is lacking in the impedance match is the corresponding scale dependent electron dipole impedance.

The force operative in the quantum Hall effect is the vector Lorentz force. Impedance quantization is a possibility for all forces [6]. Quantizing with electromagnetic forces only and taking the quantization length to be the electron Compton wavelength gives the impedance network of figure 2, where the electron ‘external dipole’ impedance match to the photon is represented by the large blue diamonds. The nodes of the network are strongly correlated with the unstable particle coherence lengths [21, 22], suggesting that, as in the hydrogen atom, energy flows to and from the unstable particle spectrum via this network of electron impedances.

If impedance quantization is both a fact of nature and a powerful theoretical tool (as explicated later in this paper), how is it not already present in the Standard Model? One might suggest that the absence is simply an historical accident, a consequence of the order in which experimentalists revealed relevant phenomena [20]. The scaffolding of QFT was erected on experimental discoveries of the first half of the twentieth century, on the foundation of QED, which was set long before the Nobel prize discovery of the scale invariant quantum Hall impedance in 1980 [23].

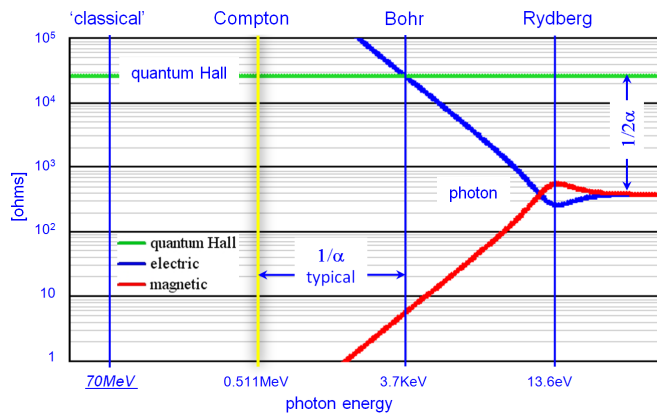


FIGURE 1. Electron quantum Hall and photon near and far field impedances vs. photon energy [19]

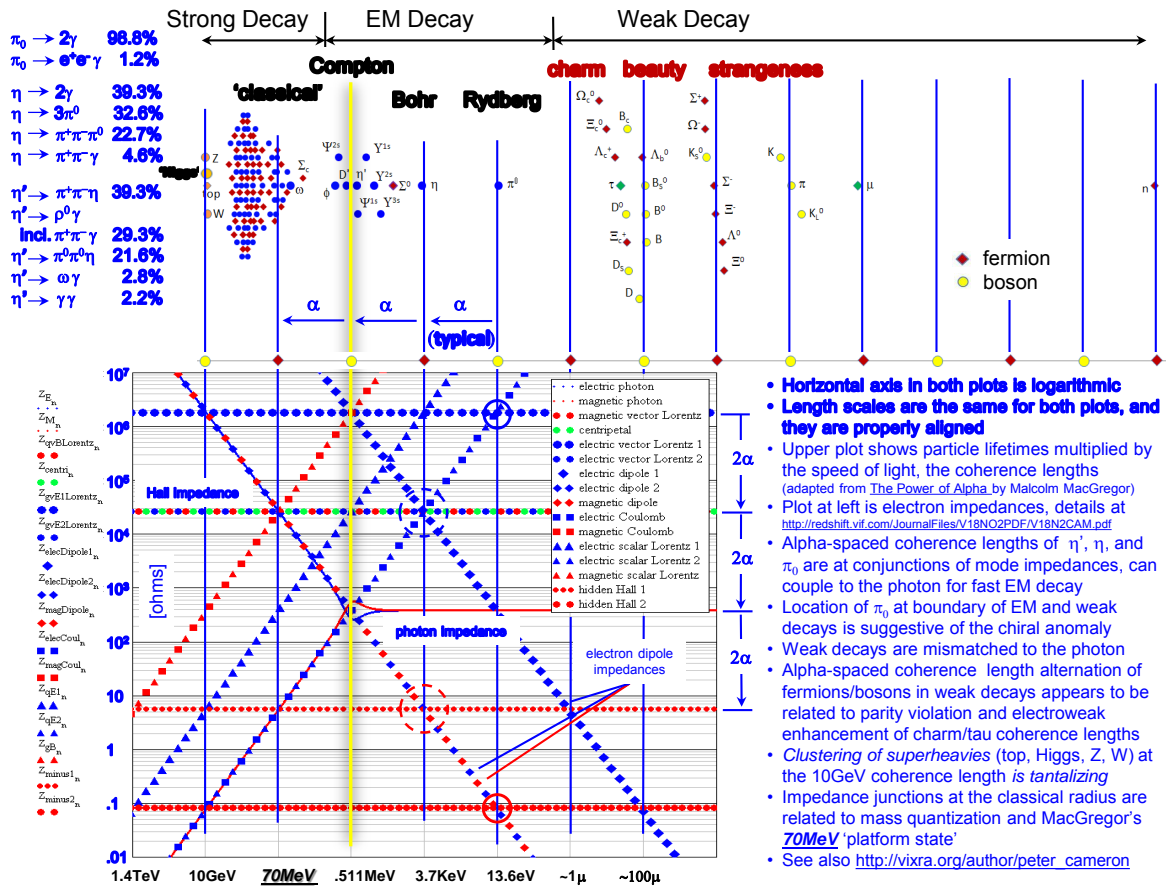


FIGURE 2. The 'One Slide' [35]

The discovery of exact impedance quantization was greatly facilitated by the scale invariance. This classically peculiar impedance is topological, the measured impedance being independent of the size or shape of the Hall bar, independent of the size or shape of the resistor that governs the flow of current. Prior to that discovery, impedance quantization was more implied than explicit in the literature [24, 25, 26, 27, 28, 29, 30]. Early mentions include the 1955 paper of Jackson and Yovits [24] and the 1957 paper of Landauer [25].

The 1959 thesis of Bjorken [26] presents an approach summarized [27] as "...an analogy between Feynman diagrams and electrical circuits, with Feynman parameters playing the role of resistance, external momenta as current sources, and coordinate differences as voltage drops. Some of that found its way into section 18.4 of..." the canonical text [28]. As presented there, the units of the Feynman parameter are [sec/kg], the units of mechanical *conductance* [31].

It is not difficult to understand what led Bjorken astray, as well as those (including the present author) who have made more recent similar attempts [5, 32, 33, 34]. The units of mechanical impedance are [kg/sec]. One would think that more [kg/sec] would mean more mass flow. However, the physical reality is more [kg/sec] means more impedance and *less* mass flow. This is one of many interwoven mechanical, electromagnetic, and topological paradoxes [35] to be found in the SI system of units, which ironically were developed with the intent that they "...would facilitate relating the standard units of mechanics to electromagnetism." [36].

With the confusion that resulted from misinterpreting conductance as resistance and lacking the concept of quantized impedance, the anticipated intuitive advantage [28] of the circuit analogy was lost. The possibility of the jump from a well-considered analogy to a photon-electron impedance model was not realized at that time.

Like the first Rochester Conference on Coherence and Quantum Optics in 1960, the 1963 paper/thesis by Feynman and Vernon [29] on the "Interaction of Systems" was motivated by the



invention of the maser. The authors devoted a thesis to concepts needed for impedance matching to the maser. Lacking again was the explicit concept of *quantized* impedance.

While the 1970 paper by Landauer [30] somewhat clarified his earlier work, the explicit concept of impedance quantization remained obscure.

Quantization of mechanical impedance in the hydrogen atom was introduced in a 1975 unpublished note [5]. However, the quantity with units [kg/sec] was interpreted as mass flow in the deBroglie wave, with confusion arising again due to the inversion in the SI system of units.

Had impedance quantization been discovered in 1950 rather than 1980, one wonders whether the concept might have found its way into the foundation of QED at that time, before it was set in the bedrock. As it now stands, the inevitable reconciliation of practical and theoretical, the incorporation of impedances into the foundations of quantum theory, is paradigm-changing.

### THE IMPEDANCE MODEL

Given the experimental evidence of quantization in the photon and quantum Hall impedances and the realization that mechanical impedances can be calculated from Mach's principle applied to the two body problem [5], it is a short step to introduce the (inverse square of) line charge density needed to convert mechanical impedances [6] to electrical, where techniques for calculating electromagnetic interactions between the objects of Geometric Algebra are known.

With electromagnetic fields only, taking maximal symmetry between electric and magnetic, and taking the simplest geometric objects needed for a realistic model [6] gives

- quantization of magnetic and electric flux, charge, and dipole moment
- three objects - flux quantum (no singularity), monopole (one), and dipole (two)
- confinement to a fundamental length, taken to be the electron Compton wavelength
- the photon

In seeking to link IA to STA, one possibility is to explore the correspondence between the geometric objects of the two approaches, as shown in figure 3:

The calculated coupling impedances of the interactions between these geometric objects[6, 21, 37], the coupling impedances of the modes of the model, are shown in figure 2. Of immediate interest in terms of defining the components of the Dirac wavefunction are the modes intersecting at the electron Compton wavelength, including those of the .511 MeV photon. The energy of a photon whose wavelength is the electron Compton wavelength equals the .511 MeV rest mass of the electron.

The modes at the .511 MeV node that is matched to the  $377 \Omega$  photon impedance fall into one of three categories:

- self-interaction between the electric and magnetic flux quanta of the photon
- interaction between the flux quanta of the photon and the electron modes
- self-interaction of the excited electron modes

The three categories are stages in the transfer of energy from photon to electron. Results of the geometric products that describe these interactions are shown in figure 4.






item	marker	symbol	GA object
electric flux quantum		$\phi_E$	bivector
magnetic flux quantum		$\phi_B$	bivector
electric dipole		$d_E$	vector
magnetic dipole		$\mu_B$	vector
electric monopole		$q_E$	scalar

FIGURE 3. Possible linking of a subset of the objects of IA with STA

In the **first stage** the coupled magnetic and electric flux quanta of the photon are propagating in free space [38]. The geometric product of the two flux bivectors delivers a pseudoscalar and a scalar.

In the **second stage**, which describes excitation of the electron by the photon, we have four interactions, each between one of the flux quanta of the photon and one of the geometric objects of the impedance model as shown in figure 3.

Here the two flux quanta of the free space photon start to decohere due to the opposing phase shifts of the capacitive and inductive impedances of the electron. Keller summarizes a possible interpretation of this process in the preface to his treatise on quantum theory in near-field electrodynamics [39].

*“Matter-attached fields are unavoidably present in the near-field... and in the covariant notation their quantization leads to the scalar and longitudinal photons, and then by a certain unitary transformation to gauge and near-field photons.”*

In near-field electrodynamics *“The longitudinal electric field is always of crucial importance...this field involves the difference between the longitudinal and scalar photons”*. [40]

The ‘certain unitary transformation’

$\begin{pmatrix} i & -i \\ 1 & 1 \end{pmatrix}$  is complex. Applying this transformation to the scalar and longitudinal photon wave functions delivers their ‘real’ sum (the gauge photon) and ‘imaginary’ difference (the near-field photon) [40]. The gauge photon carries the phase information (not a single measurement observable) that permits the instantaneous non-local projection of entangled photons into complementary eigenstates [22]. In the impedance model the associated impedance is scale invariant.

Assigning the experimental reality of non-local state reduction of entangled photons to the gauge photon implies the reality of the near-field photon in the excited states of the electron.

Keller’s treatment doesn’t employ STA. Presumably the geometric aspects of  $i$  in the transformation matrix are not yet understood, and certainly not by the present author. With that in mind, it should be noted that the above interpretation assumes that Keller’s longitudinal and near-field photons can be identified with the corresponding pseudoscalars of STA shown in figure 4.

In the **third stage** we have the self-interacting modes of the electron model that were excited by the impedance matched photon. These modes comprise an even sub-algebra of STA. The complete algebra appears only in the photon-electron interaction of the second stage.

In the impedance approach the ‘electron’ is a coupled mode family obeying linear superposition. The correlation of the network nodes with the coherence lengths shown in figure 2 suggests that the elementary particle spectrum consists of excited modes of the impedance network, that the network comprises the ‘structure of the vacuum’ as cited earlier [38]. Any of them, when taken as components of the Dirac wave function, should deliver meaningful results.

mode	interaction	geometric product	resulting STA grades
photon	self	$\phi_E \phi_B$	pseudoscalar + scalar
photon-electron	mutual	$\phi_E \phi_B$	pseudoscalar + scalar
photon-electron	mutual	$d_E \phi_E$	pseudovector + vector
photon-electron	mutual	$\mu_B \phi_E$	pseudovector + vector
photon-electron	mutual	$q_E \phi_B$	bivector + scalar
stationary photon	self	$\phi_E \phi_B$	pseudoscalar + scalar
electric dipole	self	$d_E d_E$	bivector + scalar
magnetic dipole	self	$\mu_B \mu_B$	bivector + scalar
electromagnetic dipole	self	$d_E \mu_B$	bivector + scalar
scalar Lorentz	self	$q_E \phi_E$	bivector
Coulomb	self	$q_E q_E$	scalar

FIGURE 4. Grades of the photon-electron interaction at .511 MeV

The initial conjecture [8] relating IA and STA was based upon the distinction between scale invariant (rotation gauge field) and scale dependent (translation gauge field) impedances. With the one known exception of the massless photon, which is unique in having both scale invariant far-field and scale dependent near-field impedances, the invariant impedances cannot communicate energy/information, only quantum phase. This distinction plays a fundamental role in entanglement, non-locality, and state reduction [22], the black hole information paradox [41], the chiral anomaly [42], time asymmetry [43], the extreme early Big Bang [44] and at the foundational level in interpretations of quantum mechanics [45].

The centrifugal impedance shown in figures 2 (green dots) and 5 (green line) is scale invariant. Scale invariant impedances cannot be shielded [43]. The vector Lorentz impedance of the Aharonov-Bohm effect is one example. The question here is what role invariant impedances might play in gravitation. The equivalence principle as stated by Heisenberg [46] reads

*“...gravitational forces can be put on the same level as centrifugal or other forces that arise as a reaction of the inertia...”*

### THE PLANCK PARTICLE

Just as the energy of a photon whose wavelength is the electron Compton wavelength equals the electron rest mass, the energy of a photon whose wavelength is the Planck particle Compton wavelength is the rest mass of the Planck particle and its associated event horizon. This is the ‘electromagnetic black hole’, the simplest Planck particle eigenstate. A more detailed model can be had by taking the quantization length to be not the electron Compton wavelength, but rather the Planck length, resulting in the network of figure 5.

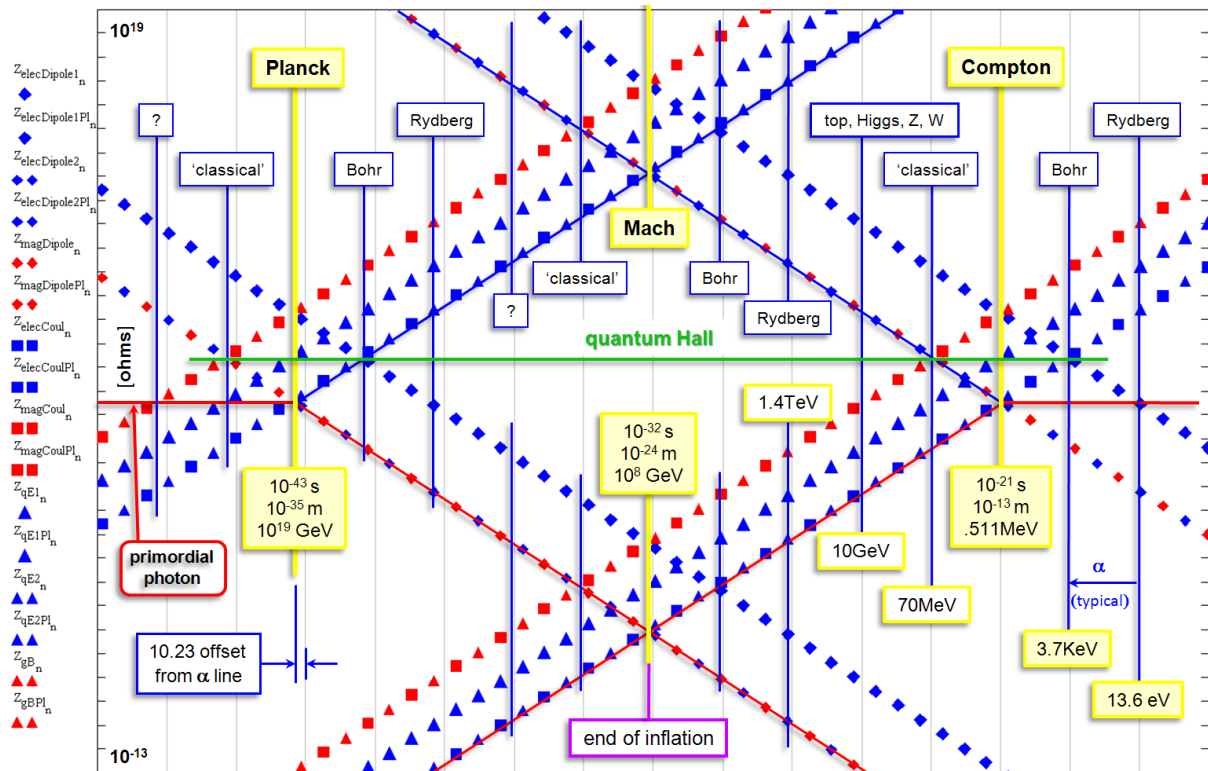


FIGURE 5. An impedance template for the Big Bang - a subset of the electron and Planck particle impedance networks, showing a .511 Mev photon entering from the right and the ‘primordial photon’ from the left. The green line represents both quantum Hall and centrifugal impedances [7, 41, 44].

Calculating the impedance mismatch between electron and Planck particle gives an identity between electromagnetism and gravity [7, 41]. The calculation proceeds in the same manner as the

impedance match of the 13.6 eV photon near-field impedance to the quantum Hall impedance at the Bohr radius, by attempting to match the .511 MeV photon near-field impedance to the quantum Hall impedance at the Planck length. Similar calculations can be done with any of the coupled impedances of figure 5.

The gravitational force between these two particles is equal to the impedance mismatched electromagnetic force they share. This result suggests that both gravity and rest mass are of electromagnetic origin. While strong classical arguments have been advanced against electromagnetic theories of gravity [47], preliminary examination suggests that such arguments fail when the full consequences of quantum phase coherence are taken into consideration.

IA delivers exact results at the Planck particle event horizon (and beyond to the singularity, completely decoupled by the infinite mismatch to the dimensionless point). Relativistic curvature corrections are unneeded. The impedance model is flat space.

## MASS

At the nine digit limit of experimental accuracy, the exact identity between gravity and electromagnetism that was found by impedance matching to the Planck particle [7] limits the energy transfer between these two particles to the rest mass of the electron. In this sense the electromagnetic interaction with the Planck particle can be considered a route to the ‘origin of mass’, and the Planck particle almost but not quite virtual. The Casimir effect comes to mind.

The impedance model offers a simple second route to mass. The model is comprised of self-interacting electromagnetic fields in flat space, configured as geometric objects in the flat spacetime algebra of gauge theory gravity, and confined by impedance mismatches as one moves away from the quantization scale. The mode impedances of the self-interacting geometric objects are shown in figures 2 and 5.

The second ‘origin of mass’ in the impedance model is the stored energy of the electromagnetic fields. Calculating that energy [48] at the relevant quantization scales gives the electron mass at the limit of experimental accuracy, the muon mass at one part per thousand, the pion at two parts in ten thousand, and the nucleon at seven parts in one hundred thousand. The pion and muon calculations invoke a supersymmetry of sorts. The nucleon calculation is admittedly a bit of a kludge, but interesting none-the-less.

## GRAVITY

The relatively recent discovery that Gauge Theory Gravity in flat space is equivalent to General Relativity in curved space [9, 10, 11, 12, 13, 14] is both astounding and a paradigm shift of itself. Why work in curved space all these years if one can work so much more simply in flat space? How did it get this way?

Like the absence of impedances from QED, this is another historical accident. It arose because the geometric algebra of Grassman and Clifford was lost with the early death of Clifford and the ascendancy of the simpler Gibbs’ vector formalism in the late 19th century. Clifford algebra persisted in various forms without geometric insight until rediscovered and expanded by Hestenes starting in the 1950s. Einstein and company did not have that tool at hand, worked with tensor calculus (which is a subset of geometric algebra/calculus, as is the Dirac algebra of quantum mechanics) in curved space. Whether one describes gravity as the effect of mass curving space or quantum phase shifts, the claim here is that they yield equivalent results.

Just as mass is of electromagnetic origin in the impedance approach, so must be gravity. What then of the graviton? Which of Keller’s photons [49] is the graviton?

Some guidance comes from two essential characteristics of gravity that, upon first consideration, would seem to rule out an electromagnetic origin [47]. First, unlike electromagnetic forces, it appears that gravity cannot be shielded. However, as mentioned earlier in the context of both the centrifugal force and the Aharonov-Bohm effect of the vector Lorentz force, scale invariant impedances cannot be shielded [22, 43]. And second, gravity appears to have only one sign. We observe only attractive gravitational forces. However again it seems these impedances have a particular characteristic that is relevant here. These impedances are DC. As such, they can account for the attractive-only character of gravity. In the case of observables it seems that they act by retarding the phases [50], or the space bending if you will. In the case of the ‘dark matter/energy’ of the impedance model [6] the possibility exists that either or both the phase or/and its effect upon such matter/energy would be repulsive rather than attractive.

Given that linear superposition applies to this quantum network of nonlinear coupled modes (!), it would seem that any of the scale invariant impedances would be equal to the tasks outlined above, as appropriate for a given set of initial conditions. The phase shifts of Gauge Theory Gravity could be communicated by any of the scale invariant impedances. Yet a paradox remains, apparently topological and one of many. The scale invariant impedances can do no work, can only communicate quantum phase. And we all know gravity can do work.

Given that gravity is of electromagnetic origin and that Keller’s three-stage formalism gives a reasonable approximation of the near field photon-electron interaction, we return to the question of which of Keller’s photons, the ‘gravity photon’, corresponds to the graviton of quantum GR.

For the interaction between two electrons, gravity is about forty-two orders of magnitude weaker than the Coulomb force. If we take a characteristic length to be the electron Compton wavelength (about  $10^{-12}$  meters), or equivalently the wavelength of a .511MeV photon, then the wavelength of the ‘gravity photon’ will be about forty-two orders of magnitude greater, or about  $10^{30}$  meters. The radius of the observable universe is about  $10^{26}$  meters.

The point is that our material existence appears to be in the extreme near field of the ‘gravity photons’ of *almost* all of the mass in the universe. The *almost* arises due to the  $\pi$  phase shift of those ‘gravity photons’ whose average energy is above a few GeV. The phase shift due to field oscillation reverses the effective direction at around the present age of the universe. The high energy portion of matter becomes repulsive on the scale of the universe.

It can be argued that in the extreme near field the scale dependent impedances appear scale invariant, due to the flatness of the phase as the amplitude goes to zero. One might conjecture that this is what permits the scale dependent ‘Coulomb’ impedance of the monopole mass to appear to have the ‘cannot be shielded’ property of the scale invariant impedances. Hopefully the topological character of geometric algebra as informed by quantized impedances will provide a proper formalism for such a conjecture.

## CONCLUSION

Trusting and following the rigorous logics of both the geometric algebra of Grassman and Clifford and the foundations of Mach’s intuitions regarding the origins of mass have led most unexpectedly to an electromagnetic model that offers the possibility of a formalism bringing together electromagnetism, nuclear forces, and gravity. One hopes that this possibility will be recognized, and eagerly awaits the work by mathematical physicists with geometric algebra as informed by impedance quantization. Please. If it exists, show us that formalism.

Equally or more promising is the integration of impedance quantization in the toolbox of the nano-engineer, the quantum chemist, the biologist,... The economic future of impedance quantization appears to be in AMO/condensed matter physics [52].

## ACKNOWLEDGEMENTS

The author thanks Michael Suisse for helpful discussions and innumerable literature searches, and family and friends for unfailing support and encouragement.

In closing, it seems appropriate to repeat a quote attributed [51] to Einstein:

*“To understand the electron would be enough.”*

## REFERENCES

- [1] D.O. Hestenes, *Space-Time Algebra*, 2nd ed., Birkhauser (2015)
- [2] L. Smolin, *The Case for Background Independence* (2005) <http://arxiv.org/abs/hep-th/0507235>
- [3] M. Rozali, *Comments on Background Independence and Gauge Redundancies*, *Adv.Sci.Lett.* 2 p.244-250 (2009) <http://arxiv.org/abs/0809.3962>
- [4] E. Leader and C. Lorce, *The angular momentum controversy: What’s it all about and does it matter?*, *Physics Reports* 541 3 p.163–248 (20Aug2014) <http://arxiv.org/pdf/1309.4235v1.pdf>
- [5] P. Cameron, *The Two Body Problem and Mach’s Principle*, submitted to *Am. Jour. Phys* (1975), in revision. The unrevised version was published as an appendix to the Electron Impedances note [6].
- [6] P. Cameron, *Electron Impedances*, *Apeiron* 18 2 222-253 (2011) <http://redshift.vif.com/JournalFiles/V18NO2PDF/V18N2CAM.pdf>
- [7] P. Cameron, *Background Independent Relations Between Gravity and Electromagnetism* (2012) <http://vixra.org/abs/1211.0052>
- [8] P. Cameron, *Impedance Quantization in Gauge Theory Gravity*, Gravity Research Foundation Essay Competition (2015) <http://vixra.org/abs/1503.0262>
- [9] A. Lasenby, C. Doran, and S. Gull, *Astrophysical and Cosmological Consequences of a Gauge Theory of Gravity*, in N. Sanchez and A. Zichichi (ed.), *Current Topics in Astrofundamental Physics*: Erice, 1994, p.359, World Scientific, Singapore (1995).
- [10] D. Hestenes, *Spacetime Calculus for Gravitation Theory* (1996) <http://libra.msra.cn/Publication/4860262/spacetime-calculus-for-gravitation-theory>
- [11] A. Lasenby et.al, *Gravity, gauge theories and geometric algebra*, *Phil. Trans. R. Lond. A* 356 487–582 (1997) <http://arxiv.org/abs/gr-qc/0405033>
- [12] J. Lasenby, A. Lasenby and C. Doran, *A unified mathematical language for physics and engineering in the 21st century*, *Phil. Trans. R. Soc. Lond. A* 358, 21-39 (2000) <http://geometry.mrao.cam.ac.uk/2000/01/a-unified-mathematical-language-for-physics-and-engineering-in-the-21st-century/>
- [13] C. Doran and A. Lasenby, *Geometric Algebra for Physicists*, Cambridge University Press, p.20 (2003)
- [14] D. Hestenes, *Gauge Theory Gravity with Geometric Calculus*, *Found. Phys.* 35 (6) 903-970 (2005) <http://geocalc.clas.asu.edu/pdf/GTG.w.GC.FP.pdf>
- [15] H. Hall, *A History of Impedance Measurements*, historical archives of the General Radio Company (1992) [http://www.ietlabs.com/pdf/GenRad History/A History of Z Measurement.pdf](http://www.ietlabs.com/pdf/GenRad%20History/A%20History%20of%20Z%20Measurement.pdf)
- [16] S. Schelkunoff, *The Impedance Concept and its Application to Problems of Reflection, Refraction, Shielding, and Power Absorption*, *Bell System Technical Journal*, 17, 17-48 (1938) <http://www3.alcatel-lucent.com/bstj/vol17-1938/articles/bstj17-1-17.pdf>
- [17] *Impedance Measurement Handbook - A guide to measurement technology and techniques*, Agilent Technologies <http://cp.literature.agilent.com/litweb/pdf/5950-3000.pdf>
- [18] *Understanding the Fundamental Principles of Vector Network Analysis*, Agilent Technologies (2012) <http://cp.literature.agilent.com/litweb/pdf/5965-7707E.pdf>
- [19] C. Capps, *Near Field or Far Field?*, *Electronic Design News*, p.95 (16 Aug 2001) <http://edn.com/design/communications-networking/4340588/Near-field-or-far-field->
- [20] for a partial bibliographic listing of grad school physics texts lacking the photon near field impedance, see P. Cameron, *Historical Perspective on the Impedance Approach to Quantum Field Theory* (2014) <http://vixra.org/abs/1408.0109>  
In my searching I’ve not yet found even one physics text that has the near field photon impedance.
- [21] P. Cameron, *Generalized Quantum Impedances: A Background Independent Model for the Unstable Particles* (2012) <http://arxiv.org/abs/1108.3603>

- [22] P. Cameron, *Quantum Impedances, Entanglement, and State Reduction* (Mar 2013)  
<http://vixra.org/abs/1303.0039>
- [23] K. von Klitzing et.al, *New method for high-accuracy determination of the fine-structure constant based on quantized Hall resistance*, PRL **45** 6 494-497 (1980)
- [24] J.L. Jackson and M. Yovits, *Properties of the Quantum Statistical Impedance*, Phys. Rev. **96** 15 (1954)  
<http://www.deepdyve.com/lp/american-physical-society-aps/properties-of-the-quantum-statistical-impedance-8qnQk0mK33>
- [25] R. Landauer, *Spatial Variation of Currents and Fields Due to Localized Scatterers in Metallic Conduction*, IBM J. Res. Dev. **1** 223 (1957)
- [26] J. Bjorken, *Experimental tests of Quantum electrodynamics and spectral representations of Green's functions in perturbation theory*, Thesis, Dept. of Physics, Stanford University (1959)  
<http://searchworks.stanford.edu/view/2001021>
- [27] J. Bjorken, private communication (2014)
- [28] J. Bjorken, and S. Drell, *Relativistic Quantum Fields*, McGraw-Hill, section 18.4 (1965)
- [29] R. Feynman and F. Vernon, *The Theory of a General Quantum System Interacting with a Linear Dissipative System*, Annals of Physics **24** 118-173 (1963)  
<http://isis.roma1.infn.it/presilla/teaching/mqm/feynman.vernon.1963.pdf>
- [30] R. Landauer, *Electrical Resistance of Disordered One-dimensional Lattices*, Philos. Mag. **21** 86 (1970)
- [31] N. Flertcher and T. Rossing, *The Physics of Musical Instruments*, 2nd ed., Springer (1998)
- [32] C. Lam, *Navigating around the algebraic jungle of QCD: efficient evaluation of loop helicity amplitudes*, Nuc. Phys. B **397**, (1–2) 143–172 (1993)
- [33] C. Bogner, *Mathematical aspects of Feynman Integrals*, PhD thesis, Mainz (2009)
- [34] D. Huang, *Consistency and Advantage of Loop Regularization Method Merging with Bjorken-Drell's Analogy Between Feynman Diagrams and Electrical Circuits*, EJP C, (2012) <http://arxiv.org/abs/1108.3603>
- [35] P. Cameron, *The 'One Slide' Introduction to Generalized Quantum Impedances*, p.39-43  
<http://vixra.org/abs/1406.0146>
- [36] M. Tobar, *Global Representation of the Fine Structure Constant and its variation*, Meteorologia **42** 129-133 (2005)
- [37] The mathcad file that generates the impedance plots is available from the author.
- [38] M. Urban etal, *The Quantum Vacuum as the Origin of the Speed of Light*, Eur.Phys.J. D (2013) 67:58  
<http://arxiv.org/abs/1302.6165>
- [39] O. Keller, *Quantum Theory of Near-Field Electrodynamics*, Springer Series in Nano-Optics and Nanophotonics (2011)
- [40] *ibid*, chapters 16 and 22
- [41] P. Cameron, *Possible Resolution of the Black Hole Information Paradox*, Rochester Conference on Quantum Optics, Information and Measurement (2013). <http://www.opticsinfobase.org/abstract.cfm?URI=QIM-2013-W6.01>
- [42] P. Cameron, *An Impedance Approach to the Chiral Anomaly* (Feb 2014) <http://vixra.org/abs/1402.0064>
- [43] P. Cameron, *Delayed Choice and Weak Measurement in the Nested Mach-Zehnder Interferometer*, accepted for presentation at the Berlin Conference on Quantum Information and Measurement (2014)  
<http://vixra.org/abs/1310.0043>
- [44] P. Cameron, *The First Zeptoseconds: An Impedance Template for the Big Bang* (2015)  
<http://vixra.org/abs/1501.0208>
- [45] M. Suisse and P. Cameron, *Quantum Interpretation of the Impedance Model*, accepted for presentation at the Berlin Conference on Quantum Information and Measurement (2014) <http://vixra.org/abs/1311.0143>
- [46] W. Heisenberg, *Physics and Philosophy: The Revolution in Modern Science*, p.95, George Allen and Unwin (1958)
- [47] J. Wheeler and I. Cuifolini, *Gravitation and Inertia*, p.391, Princeton (1995)
- [48] P. Cameron, *Electric and Magnetic Flux Quanta: The Pion Mass*, Apeiron **18** 1 29-42 (2011)  
<http://redshift.vif.com/JournalFiles/V18NO1PDF/V18N1CAM.pdf>
- [49] O. Keller, *Light: The Physics of the Photon*, Springer Series in Nano-Optics and Nanophotonics (2014)
- [50] Pound, R. and Rebka, G., *Apparent Weight of Photons*, Phys. Rev. Lett. **4**, 337 (1960)
- [51] J. Talman, private communication (2011)
- [52] extensive references to the role of impedances in AMO and condensed matter can be found in the bibliography of [20]

# INVERSE KINEMATICS FOR A 6-DOF LEG WALKING HUMANOID.

**L. Campos-Macias<sup>a</sup>, O. Carbajal-Espinosa<sup>a</sup>,  
A. Loukianov<sup>a</sup>, E. Bayro-Corrochano<sup>a</sup>**

<sup>a</sup> CINEVESTAV del IPN-Unidad Guadalajara,  
Av. del Bosque 1145. Col. El Bajío,  
CP. 45019. Zapopan, Jalisco, Mexico.  
lecampos@gdl.cinvestav.mx , ocarbajal@gdl.cinvestav.mx ,  
louk@gdl.cinvestav.mx , edb@gdl.cinvestav.mx

**ABSTRACT.** This paper describes a novel method to solve the inverse kinematics of a humanoid robot anthropomorphically configured of 6 degrees of freedom using conformal geometric algebra. Creating different geometric entities such as lines, planes and spheres in order to achieve the desired position and orientation of the body and the foot, re-configuring individually the amount of rotation for each joint. Taking into account the avoidance of obstacles and preventing self collision. The effectiveness of the proposed algorithm is proved and tested via practical experiments. Results indicate that the proposed algorithm achieves the expected behavior.

## 1. INTRODUCTION

Humanoid robotics have been actively researched in the last twenty years. This is because they are intended to be used in human societies and therefore are required to have high mobility. The motivation of that is the suitability of the biped structure for tasks in the human environment, and the goal of the studies in this area is to reach the human walking dexterity, efficiency, stability, effectiveness, and flexibility. There are many problems that involve the manipulation of a humanoid robot, this paper focuses in explain a new method to obtain the required angles of a six degrees of freedom humanoid leg anthropomorphically designed given a desired stance of the hip and foot using all the benefices that the geometric algebra can provide, creating geometric entities as conformal points, spheres, lines and planes which are transformed via rotors, translators and motors defined in this algebra. Because the solution of the system is open, there exists an infinity amount of solutions that can be found for a given stance and those solutions can end in a unreachable configuration for the humanoid robot physical possibilities including for instance a self collision, that is the main contribution of this proposal, using the advantages that the conformal geometric algebra provides, some geometric restrictions are imposed to the movement of the humanoid leg resembling the physical possibilities of a human leg and re-configuring one or more joints of the humanoid leg without changing the entire attitude of the leg.

This paper is organized as follows. A brief introduction to geometric conformal algebra, the description and creation of geometric entities as well as the rotors, translators and rotors are described in Section 2. The procedure followed to obtain the equations that link the desired pose with the solution of the inverse kinematics is given in section 3. Practical results with the proposed method are given in Section 4. Finally, conclusions are included in Section 5.

## 2. CONFORMAL GEOMETRIC ALGEBRA

One of the main characteristics of the Clifford algebra (Geometric Algebra), is that allows to represent entities of higher order with a compact symbology with lineal operations. The lines, planes or spheres are examples of entities of higher order and they are represented as a unique elements of the Clifford Algebra. In this section, the most basic concepts of Geometric Algebra are overviewed, for further information refer to [1] [2].



The geometric algebra of an Euclidean 3D space  $\mathcal{G}_3$  has a base in points, ergo, it works with vector which represents points in the space, the motor algebra  $\mathcal{G}_{3,0,1}^+$  works with a base of lines. Using the Minkowski plane [1] to generate *null vectors*, the Euclidean space  $\mathbb{R}^n$  can be expanded to  $\mathbb{R}^{n+1,1} = \mathbb{R}^n \oplus \mathbb{R}^{1,1}$ . This expansion of the Euclidean space results in the geometric conformal algebra  $\mathcal{G}_{n+1,1}$ . This new algebra called *Conformal Geometric Algebra*, has the characteristic of taking the sphere as unit of calculation, which allows work with other geometric primitives (lines, points, planes, circles, etc.) In general an Euclidean point  $\mathbf{x}(\mathbb{R}^n)$  is represented in  $\mathbb{R}^{n+1}$  as

$$(1) \quad \mathbf{x}_c = \mathbf{x} + \alpha e_0 + \beta e_\infty$$

where  $\alpha$  and  $\beta$  are scalars and  $e_0, e_\infty$  denotes a null base [1]. The conformal vector space derived of  $\mathbb{R}^3$  is denoted as  $\mathbb{R}^{4,1}$ . This algebra corresponding to  $\mathcal{G}_{4,1}$  has a vector base given by and has  $2^5$  elements.

The geometric entities of the geometric conformal algebra are presented in Table 1.

Entity	IPNS	G	OPNS (dual)	G
Sphere	$s = \mathbf{p} + \frac{1}{2}(\mathbf{p}^2 - \rho^2)e_\infty + e_0$	1	$s^* = a \wedge b \wedge c \wedge d$	4
Point	$x = \mathbf{x} + \frac{1}{2}\mathbf{x}^2 e_\infty + e_0$	1	$x^* = (-E\mathbf{x} - \frac{1}{2}\mathbf{x}^2 e_\infty + e_0)I_E$	4
Plane	$\pi = \mathbf{N}I_E - d e_\infty$ $\mathbf{N} = (\mathbf{a} - \mathbf{b}) \wedge (\mathbf{a} - \mathbf{c})$ $d = (\mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c})I_E$	1	$\pi^* = e_\infty \wedge a \wedge b \wedge c$	4
Line	$L = P_1 \wedge P_2$ $= \mathbf{r}I_E + e_\infty \mathbf{M}I_E$ $\mathbf{r} = \mathbf{a} - \mathbf{b}$ $\mathbf{M} = \mathbf{a} \wedge \mathbf{b}$	2	$L^* = e_\infty \wedge a \wedge b$	3
Circle	$z = s_1 \wedge s_1$ $s_z = (e_\infty \cdot z)z$ $\rho_z = \frac{z^2}{(e_\infty \wedge z)^2}$	2	$z^* = a \wedge b \wedge c$	3
Point pair	$PP = s_1 \wedge s_2 \wedge s_3$	3	$PP^* = a \wedge b, X^* = e_\infty \wedge x$	2

TABLE 1. Representation of entities in conformal geometric algebra.

Conformal transformation of geometric entities it is called to that in which preserves its form. The transformation does not affect the angles of the figure y preserves the straight lines and circles. Any conformal transformation in  $\mathbb{R}^n$  is expressed as a composition of inversions in spheres and reflections in hyperplanes. In general a conformal transformation is defined as

$$(2) \quad g(\mathbf{x}) = G\mathbf{x}\widehat{G}^{-1} = \sigma\mathbf{x}'$$

where  $\mathbf{x} \in \mathbb{R}^{n+1,1}$ ,  $G$  is a versor and  $\sigma$  a scalar.

The conformal transformations are shown in Table 2.

Type	$g(\mathbf{x})$ in $\mathbb{R}^n$	Versor in $\mathcal{G}_{n+1,1}$	$\sigma(x)$
Inversion	$\frac{\rho^2}{\mathbf{x} - \mathbf{p}} + \mathbf{p}$	$s = \mathbf{p} - \frac{1}{2}\rho^2 e_\infty$	$\left(\frac{\mathbf{x} - \mathbf{p}}{\rho}\right)^2$
Reflection	$-\mathbf{nxn} + 2\mathbf{x}\delta$	$P = \mathbf{n} + \delta e_\infty$	1
Translation	$\mathbf{x} - \mathbf{t}$	$\mathbf{T} = 1 + \frac{1}{2}\mathbf{t}e_\infty$	1
Rotation	$R_\theta \lambda \widetilde{R}_\theta$	$R_\theta = \exp(-\frac{\theta}{2}l)$	1

TABLE 2. Conformal transformations.

### 3. INVERSE KINEMATICS

The kinematic layout of the lower limbs is the basic determiner in order to imitate the human movements. The leg is created of two segments which determine the possibilities of the humanoid robot such as the length of a step or how far can the foot reach. The human articulation of the hip is a perfect ball-and-socket joint, it is reproduced in this bipedal humanoid robot with tree concurrent revolutes joints which are successively orthogonal. The human knee essentially performs the simple kinematic function of flexion-extension, then the use of a single transversal axis joint is therefore the easiest solution. Finally, the rotations in a human ankle and foot are complex but it is to be noted that two rotational axes are functionally required in the ankle, then a universal joint is used. This configuration is basic to emulate the human movements [3] and is the used in this paper.

To explain the inverse kinematics of a humanoid robot, a 6 degrees of freedom (DoF) model,

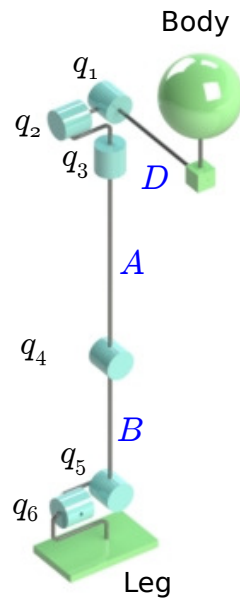


FIGURE 1. Structure of a 6 DoF biped robot leg.

all actuated, of one Leg anthropomorphically designed shown in Fig. 1 is used. The other leg can be treated in the same manner. Usually, a desired trajectory of the body and the feet of a humanoid robot considering stability are given to satisfy the movement of the humanoid to achieve some task, for example walking, squats, lying down, crawling and getting up etc. Then, the amount of joint rotation of the hip, knee and ankle must be found with this information. There are different forms to solve this problem, i.e. the analytically and numerical methods [4], but singularities, position of global and local coordinates, or the reconfiguration of the entire leg are the main problems of those approaches. For that, a new method using conformal geometric algebra is proposed.

The desired position and attitude of the body are given of the form  $x_b, y_b, z_b, \alpha_b, \beta_b, \gamma_b$  for the  $x$ ,  $y$  and  $z$  axis, the pitch, yaw and roll with respect to a world frame coordinates. And the leg,  $x_L, y_L, z_L, \alpha_L, \beta_L, \gamma_L$  for the  $x$ ,  $y$  and  $z$  axis, the pitch, yaw and roll respectively. It is supposed that the position and orientation of the leg are given in terms of a reference fixed coordinate system in the body. Then, in order to simplify the equations is defined  $D$  as the distance between the body and the hip joint,  $A$  and  $B$  are defined as the upper and lower leg length as is showed in Fig. 1. Three rotors describing the desired pitch, yaw and roll of the leg are created of the

form:

$$(3) \quad \begin{aligned} R_{x_L} &= e^{-\frac{1}{2}\alpha_L e_{23}} \\ R_{y_L} &= e^{-\frac{1}{2}\beta_L e_{31}} \\ R_{z_L} &= e^{-\frac{1}{2}\gamma_L e_{12}} \end{aligned}$$

and a general rotor describing the entire desired attitude of the leg is given of the form:

$$(4) \quad R_L = R_{x_L} \cdot R_{y_L} \cdot R_{z_L}.$$

Then, with the desired euclidean coordinates, a translator is formed:

$$(5) \quad T_L = e^{-\frac{1}{2}(x_L e_1 + y_L e_2 + z_L e_3) e_\infty}.$$

Using Eqs. (4) and (5), a motor describing the desired pose of the leg, in terms of the body frame, is given of the form

$$(6) \quad M_L = T_L \cdot R_L.$$

A conformal point that describes the fixed frame in the center of the body, another which describes the initial position of the hip joint using the known length  $D$ , and one last that shows the initial position of the ankle joint using  $A$  and  $B$  constant, are created

$$(7) \quad \begin{aligned} x_w &= 0e_1 + 0e_2 + 0e_3 + \frac{1}{2}(0e_1 + 0e_2 + 0e_3)^2 e_\infty + e_0 \\ x_h &= -De_1 + 0e_2 + 0e_3 + \frac{1}{2}(-De_1 + 0e_2 + 0e_3)^2 e_\infty + e_0 \\ x_{f_w} &= -De_1 - (A+B)e_2 + 0e_3 + \frac{1}{2}(-De_1 - (A+B)e_2 + 0e_3)^2 e_\infty + e_0. \end{aligned}$$

In order to find the pose of the ankle joint, the motor of the leg is applied to Eq. (7) of the form

$$(8) \quad x_f = M_L \cdot (x_{f_w}) \cdot \tilde{M}_L.$$

Two spheres are defined with center at the points of Eqs (7) and (8)

$$(9) \quad \begin{aligned} s_1 &= x_h - \frac{1}{2}A^2 e_\infty \\ s_2 &= x_f - \frac{1}{2}B^2 e_\infty, \end{aligned}$$

and the wedge product is applied between spheres of Eq. (9). As is shown in Fig. 2 this yields an intersection circle,

$$(10) \quad Z = s_1 \wedge s_2.$$

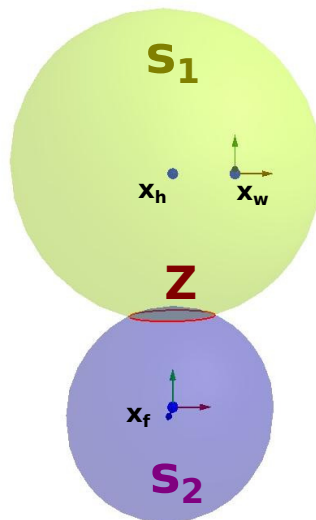


FIGURE 2. Intersection of spheres  $s_1$  and  $s_2$ .

A line is created in the  $e_3$  coordinate and transformed with the motor of the leg, in order to find the desired ankle joint,

$$(11) \quad L_3 = M_L \cdot (e_0 \wedge e_3 \wedge e_\infty) \cdot \tilde{M}_L.$$

A line normal to the circle of Eq. (10) is generated in the form

$$(12) \quad L_c = Z \wedge e_\infty,$$

then, a plane which intersects the circle of Eq. (10) in the largest distance from the center to the edge is formed using Eqs. (11) and (12)

$$(13) \quad P = (L_c \cdot (e_0 \wedge e_\infty)) \wedge L_3,$$

this creates two intersection points that represents the intersection of the two geometric entities of Eqs. (10) and (13) which are mathematically given by

$$(14) \quad pp = Z \wedge (P - I).$$

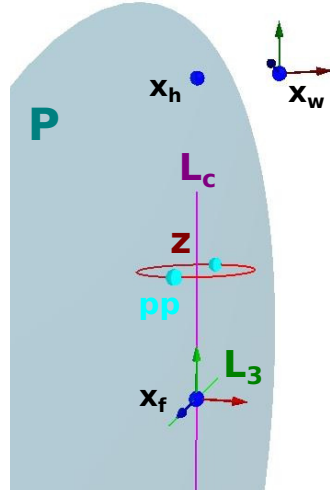


FIGURE 3. Pair of points.

Hence, two possible configurations for the knee joint pose are found as the Figure 3 shows. In order to achieve the anthropomorphic design, the restriction of taking the positive point is set. Given the knee point described by

$$(15) \quad pp_d = pp - I$$

$$x_k = \frac{pp_d + \sqrt{pp_d \cdot pp_d}}{pp_d \cdot e_\infty}.$$

Hitherto the three important points  $x_h$ ,  $x_k$  and  $x_f$  of the desired configuration given the pose of the body and the foot have been found. With those, three lines representing the links distance between each joint are created of the form using Eqs. (7) and (15)

$$(16) \quad \begin{aligned} Link_D &= e_\infty \wedge x_w \wedge x_h \\ Link_A &= e_\infty \wedge x_h \wedge x_k \\ Link_B &= e_\infty \wedge x_k \wedge x_f, \end{aligned}$$

this is observed in Fig. 4.

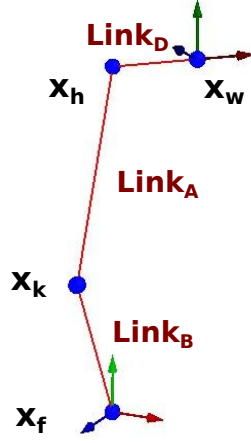


FIGURE 4. Creation of lines representing the links distance.

Finally with this information, the amount of rotation of each joint is easily calculated. The angle between lines  $Link_A$  and  $Link_B$  of Eq. (16) shows the *roll* of the knee

$$(17) \quad q_4 = \arccos \left( \frac{\sqrt{Link_A \cdot Link_B}}{|Link_A| \cdot |Link_B|} \right).$$

To find the *roll* of the spherical joint at the hip, a reference plane is created of the following form

$$(18) \quad P_{1_r} = e_3 \wedge x_h \wedge x_k \wedge e_\infty,$$

and a plane parallel to the frontal plane of the humanoid robot which moves as the spheres move is formed by

$$(19) \quad P_{1_f} = x_w \wedge x_h \wedge x_k \wedge e_\infty.$$

With Eqs. (18) and (19) the amount of rotation is described of the form

$$(20) \quad q_1 = \frac{\pi}{2} - \arccos \left( \frac{\sqrt{P_{1_r} \cdot P_{1_f}}}{|P_{1_r}| \cdot |P_{1_f}|} \right).$$

To find the *yaw* of the same spherical joint, the angle between the links described by the lines  $Link_D$  and  $Link_A$  of Eq. (16) is simply calculated by

$$(21) \quad q_2 = \frac{\pi}{2} - \arccos \left( \frac{\sqrt{Link_D \cdot Link_A}}{|Link_D| \cdot |Link_A|} \right).$$

Concerning the universal joint at the ankle, the angle of rotation in the *yaw* direction is given using the lines of Eqs. (11) and (16)

$$(22) \quad q_5 = \frac{\pi}{2} - \arccos \left( \frac{\sqrt{Link_B \cdot L_3}}{|Link_B| \cdot |L_3|} \right).$$

Then, on the occasion to find the amount of rotation at the *roll* direction of the ankle joint, a line is created parallel to the sagittal plane of the humanoid robot and transformed by the motor created in Eq. (6)

$$(23) \quad L_{3_1} = M_L \cdot (e_0 \wedge e_1 \wedge e_\infty) \cdot \tilde{M}_L,$$

using this, the angle is given of the form

$$(24) \quad q_6 = \arccos \left( \frac{\sqrt{Link_B \cdot L_{3_1}}}{|Link_B| \cdot |L_{3_1}|} \right) - \frac{\pi}{2}.$$

Finally, the last angle which describes the amount of rotation in the *pitch* direction of the spherical joint at the hip is calculated. First, several rotors and translators are created using the

initial conditions of the analysis and the information obtained so far. This will give the virtual position of the ankle. Those are given of the form

$$(25) \quad \begin{aligned} T_{h_{31}} &= e^{-\frac{1}{2}(x_h)e_\infty} \\ T_{h_{32}} &= e^{-\frac{1}{2}(De_1 - Ae_2 + 0e_3)e_\infty} \\ R_{h_{31}} &= e^{-\frac{1}{2}(q_1)e_{23}} \\ R_{h_{32}} &= e^{-\frac{1}{2}(q_2)e_{31}} \\ R_{h_{33}} &= e^{-\frac{1}{2}(0)e_{12}} \\ R_{h_{34}} &= e^{-\frac{1}{2}(q_4)e_{31}}. \end{aligned}$$

With Eq. (25), a general motor to describe the virtual pose of the ankle is formed by

$$(26) \quad M_h = (T_{h_{31}} \cdot (R_{h_{31}} \cdot R_{h_{32}} \cdot R_{h_{33}}) \cdot \tilde{T}_{h_{31}}) \cdot (T_{h_{32}} (R_{h_{34}}) \tilde{T}_{h_{32}}).$$

then, with the initial condition of the ankle of Eq. (7), the virtual point which gives the pose of the ankle is calculated with

$$(27) \quad x_{v_f} = M_h \cdot (x_{f_w}) \cdot \tilde{M}_h.$$

Furthermore, and in order to make more accurate the calculation, a virtual plane is also formed by

$$(28) \quad P_{v_f} = e_\infty \wedge x_w \wedge x_{v_f} \wedge x_k.$$

Also, a plane is created with the purpose of impose a geometric constraint

$$(29) \quad P_{c_f} = e_\infty \wedge x_h \wedge x_k \wedge x_f.$$

Then if the distance, calculated as

$$(30) \quad (x_{v_f} \wedge P_{c_f}) / \mathbb{I}_E,$$

between the virtual point of the ankle described in Eq.(28) and the restriction plane in Eq. (29) is grater or equal to zero, the amount of rotation is given by

$$(31) \quad q_3 = \arccos \left( \frac{\sqrt{(Link_B \wedge x_h) \cdot P_{v_f}}}{|(Link_B \wedge x_h)| \cdot |P_{v_f}|} \right).$$

else is given by

$$(32) \quad q_3 = -\arccos \left( \frac{\sqrt{(Link_B \wedge x_h) \cdot P_{v_f}}}{|(Link_B \wedge x_h)| \cdot |P_{v_f}|} \right).$$

The planes and lines defined in the previous equations can be oriented and translated depending in the needs of the humanoid robot, defining them with a different pose to achieve a different configuration, this helps to avoid obstacles. In regard to prevent self collisions, the geometric restrictions imposed at Eq. (15) and (30) helps to restrict the movement of the links, moreover, the radio of spheres (9) facilitates obtaining configurations achievable within the space of work of the biped humanoid robot.

This equations can only be applied to a robot that have the same configuration that Fig. 1, i.e. if the robot does not have three joint axis intersecting at one point, a different algorithm will be needed.

#### 4. APLICACIONES

The verification of the proposed method to solve the inverse kinematics of a biped robot is tested via simulations and real time. It was applied to the humanoid biped robot prototype MexOne at the Laboratory of Automatic Control at The Center for Research and Advanced Studies of the National Polytechnic Institute - Guadalajara Unit shown in Fig. 5. The Figure (6) shows the constants in meters of the humanoid used to test the inverse kinematics algorithm.

The references trajectories are created for a two steps walking with 5 centimeters length in  $e_3$  axis and 2 centimeters of height in  $e_2$  axis. In Fig. 7 can be seen  $x_b, z_b, z_L$  and  $y_L$ . The reference  $y_b$ , is considered to be zero,  $x_L$  is considered as the  $D$  constant and  $\alpha_b, \beta_b, \gamma_b, \alpha_L, \beta_L, \gamma_L$  are fixed in zero, for the left leg.

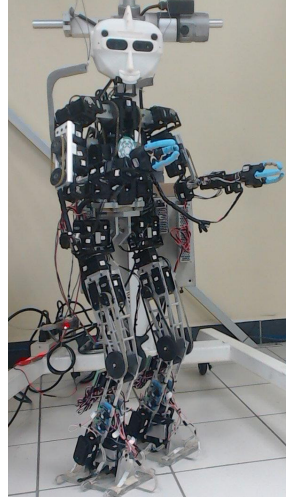


FIGURE 5. Humanoid biped robot prototype MexOne.

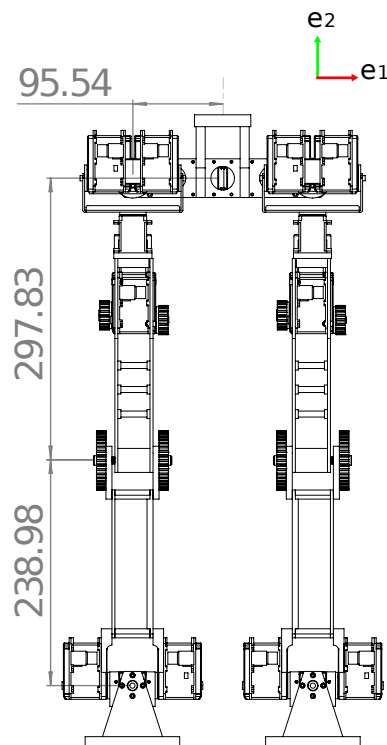


FIGURE 6. Humanoid biped robot prototype MexOne constants.

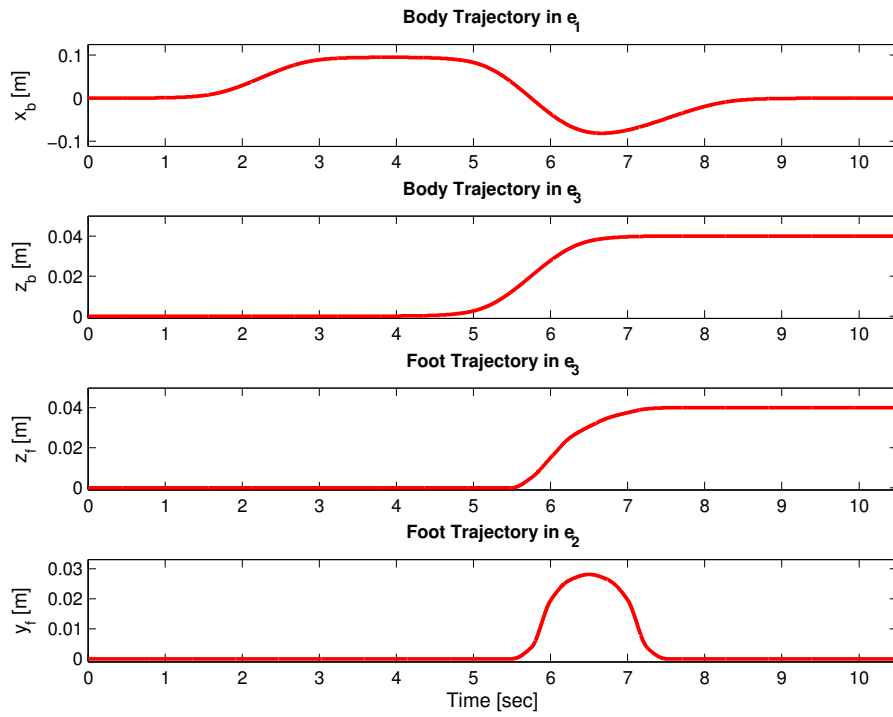


FIGURE 7. Reference trajectories.

In Fig. 8 can be observed the amount of rotation for each joint, in order to follow the body and leg trajectories. Finally in Fig. 9 a real time implementation of the experiment is shown.

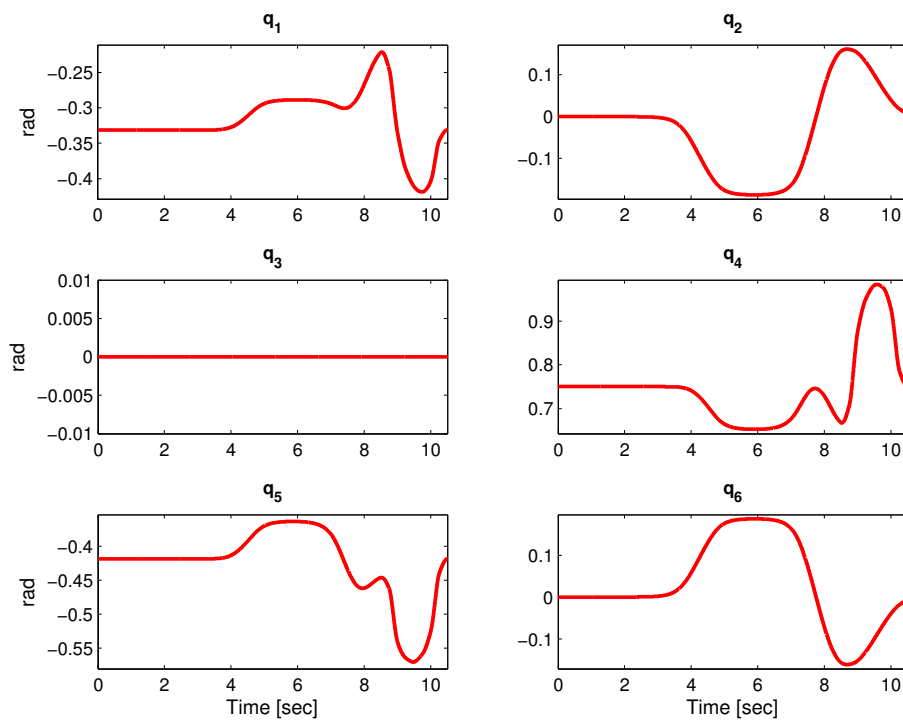


FIGURE 8. Joint trajectories left leg.



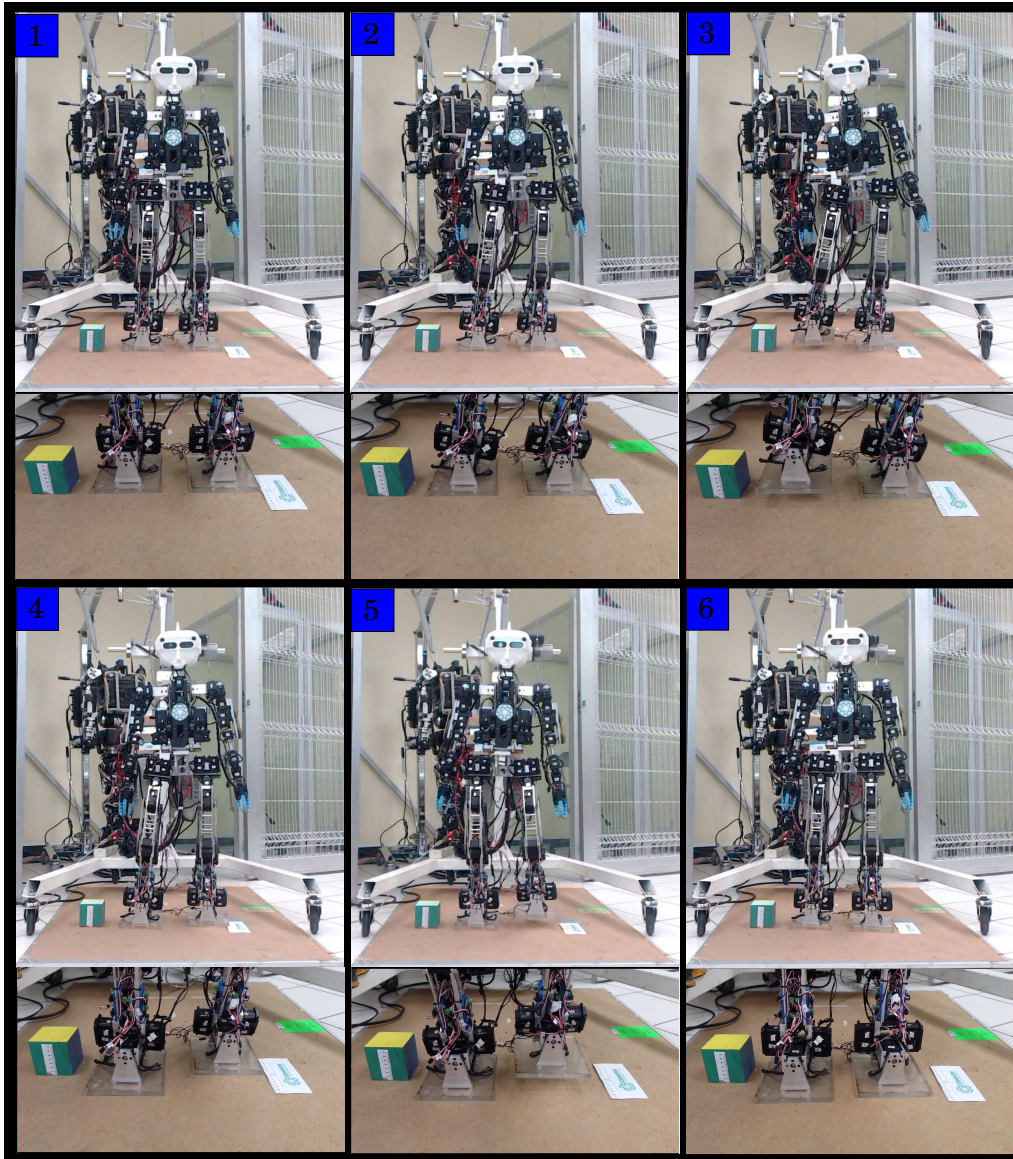


FIGURE 9. Real time experiment.

## 5. CONCLUSIONS

A novel algorithm for inverse kinematics using conformal geometric algebra was showed. Proving the benefices of using geometric entities such as planes and spheres in order to modify the amount of rotation of each joint by separate without re-configuring the entire leg of the humanoid, making easier to avoid obstacles and prevent self collision. The algorithm, was proved with a simple routine of two steps with a humanoid robot prototype.

## REFERENCES

- [1] Bayro-Corrochano, Eduardo. *Geometric Computing: For Wavelet Transforms, Robot Vision, Learning, Control and Action*. Springer Publishing Company, Incorporated, 2010.
- [2] Hestenes, David, and Garret Sobczyk. *Clifford algebra to geometric calculus: a unified language for mathematics and physics. Vol. 5*. Springer Science and Business Media, 1987.
- [3] Chevallereau, Christine and Bessonnet, Guy and Abba, Gabriel and Aoustin, Yannick. *Bipedal Robots: Modeling, Design and Walking Synthesis*. John Wiley & Sons, 2013.
- [4] Kajita, Shuuji, et al. *Introduction to humanoid robotics*. Springer, 2014.

# COMPOSING SURFACES WITH CONFORMAL ROTORS

**Pablo Colapinto**

Allosphere Research Facility  
Media Arts and Technology Program  
University of California, Santa Barbara  
wolftype@gmail.com

**ABSTRACT.** Conformal immersions are harmonic and numerically stable surfaces whose tangents scale isometrically, providing many elegant geometric properties of use in design. This paper maps these forms within the context of discrete differential geometry in order to outline an approach to synthesizing curved surfaces with potential application in architectural geometry and computer graphics. Using Dorst and Valkenburg’s “Square Root and Logarithm of Rotors” (2011), we reformulate the rationalization of *cyclidic nets*, piecewise smooth surfaces characterized and controlled directly by their tangents.

**Keywords:** Conformal Geometric Algebra, Cyclidic Nets, Rational Splines, Architectural Geometry, Discrete Differential Geometry

## 1. INTRODUCTION

In a remarkable paper from 2011 [11], Dorst and Valkenburg rigorously *decompose* general rotations in the 3D conformal model of geometric algebra into commuting point pair generators, revealing much of the beauty and logic in the algebraic machinery that generates any structure-preserving 3D motion (see also Leo Dorst’s contribution to these proceedings [9]). To explore some of the implications of that work, we outline a parametric approach to design which *composes* surfaces from out of that decomposition, suggesting the central role of conformal rotors in the establishment of a discrete calculus using the geometric algebra mechanism.

The benefits of conformal mappings in design are known. In architectural geometry, discretization of conformal mappings allows the design of doubly-curved forms with torque-free nodes and consistent offsets [2]. In computer graphics, conformal mappings have been used to deform meshes intuitively and in a way which preserves details and textures [6, 5].

Leveraging Dorst and Valkenburg’s analysis of general conformal rotors from orthogonal generators, we reformulate the discrete cyclidic nets described by Bobenko and Huhnen-Venedey in [3], who use Lie Sphere Geometry to build on the original rationalization of Martin in [22] based on Dupin’s cyclides [13]. Our main finding is that the conformal rotor construction provides a straightforward mechanism for linearly rationalizing these beneficial surfaces within the already rich context of conformal geometric algebra.

**1.1. Background.** *Cyclidic nets*, a method of computer-assisted design introduced by R. Martin in his 1983 thesis, enable the digital designer to construct surfaces by specifying curvature directly at tangents. Motivated by a desire to develop a technique to form firmly grounded in geometry, Martin showed that these biquadratic meshes are piecewise smooth patches of Dupin cyclides, and thereby possess a simple algebraic representation. Built with families of generalized circles, discrete differential geometry over these surfaces exhibits excellent convergence to the continuous case. As a result, rationalization and efficient evaluation of these surfaces – and their general form as Darboux cyclides – has been the subject of much research [14, 23]. Some modern techniques, such as developed by Bobenko and Huhnen in [3], rely on *Lie sphere geometry*, a projective model of *contact-based* geometric constructions closely related to the conformal model of geometric algebra. Others, such as Krasauskas and Zubé in [20], use geometric algebra to construct generic Darboux cyclidic patches using bilinear interpolation of quaternion-weighted Bézier arcs.

In other work specific to GA and CGA, Doran explores circle and sphere blending through direct linear interpolation of the primitive elements (that is, without rotors, conformal maps, or logarithms) [7], and Druoton et al explore constructions of Dupin cyclides by generating the full families of tangent spheres in [12].

---

With support from the Robert W. Deutsch Foundation.

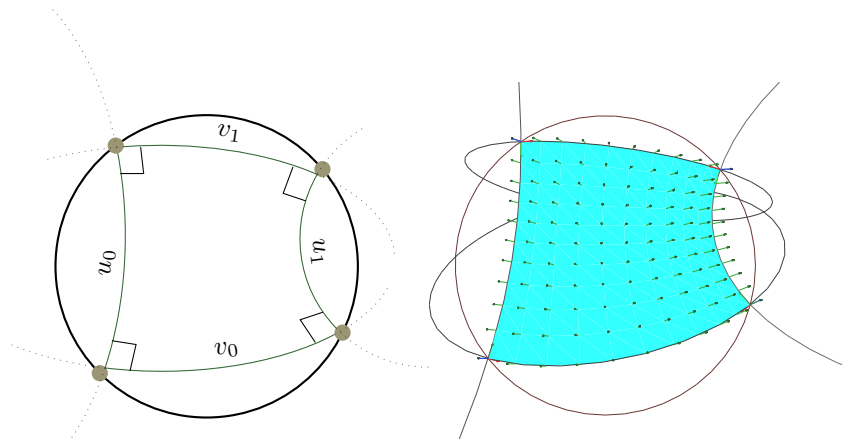


FIGURE 2.1. A four-sided principal patch is constructed from four cocircular points with orthogonal corners. Principal contact edges lie along circles.

Our current exploration differs from these others in its reliance on orthogonally-composed conformal rotors and their logarithms to rationalize blends between coordinate surface elements. We emphasize the use of sphere-to-sphere conformal transformations to discretize these coordinate surfaces, allowing ratios of basic geometric primitives to guide our rationalization. We provide a more controllable and general result than our experiments in [4], where we examined the use of interpolated null tangents to create “boosted surfaces”.

We find our technique is firmly grounded in a geometric (spherical) interpretation of curvilinear coordinate systems, and that within the conformal model our mappings apply as easily to tangents and normals as they do to points, representing a novel approach to constructing a general operator on a surface.

**1.2. Notation.** We do not provide an introduction to the mechanics of conformal geometric algebra here. The references [1, 4, 10, 21] more than suffice as an introduction.

We rely on the 2010 Amsterdam convention for notation of the conformal geometric algebra of  $\mathbb{R}^{4,1}$ , with three Euclidean basis blades  $\{e_i\}$ , homogenizing origin  $n_o = .5(e_- - e_+)$ , and infinity  $n_\infty = e_+ + e_-$ , and null points  $p = n_o + x + \frac{1}{2}x^2n_\infty$ . Note that  $n_o$  and  $n_\infty$  are sometimes identified as  $e_o$  and  $e_\infty$  or just  $o$  and  $\infty$  in other texts. Brackets  $\langle \mathcal{R} \rangle_k$  around a multivector signify only the  $k$ -graded elements of  $\mathcal{R}$ . The absence of a subscript (as in  $\langle \mathcal{R} \rangle$ ) signifies only the scalar component,  $\langle \mathcal{R} \rangle_0$ . We use the Hestenes left-contraction inner product  $\rfloor$  throughout as our generic dot product.

Following the style of [10], lower case greek letters refer to the dual (*inner-product null space*) representations of geometric elements, and upper case greek letters the direct (*outer-product null space*) representations. Thus where possible in our algorithms we use  $\sigma$  and  $\Sigma$  for dual and direct spheres,  $\lambda$  and  $\Lambda$  for dual and direct lines,  $\pi$  and  $\Pi$  for dual and direct planes,  $\tau$  for point pairs and null tangent vectors, and  $\kappa$  for circles and null tangent bivectors.

$I$  is the 5-blade pseudoscalar of  $\mathbb{R}^{4,1}$ , with duality in the conformal space by multiplication with  $I^{-1}$  is notated with a star, as in  $\sigma = \Sigma^*$ . Involution of an element  $\Sigma$  is indicated with the hat symbol  $\hat{\Sigma}$ . Finally, we specify rotors,  $\mathcal{R}$ , in calligraphic font and their application to a generic element  $X$  via the ‘sandwich’ product as  $\mathcal{R}[X] = \mathcal{R}X/\mathcal{R}$ .

## 2. PRINCIPAL PATCHES

Throughout the literature on parametric surface design with Dupin cyclides, a *principal patch* is created through bilinear rationalization of four points on a circle (Figures 2.1 and 2.2). Given these four cocircular points and the orientation of a frame on one of them, the patch surface is determined. Surfaces constructed out of such patches are called *cyclidic nets*. We seek here to develop a system for constructing and discretizing these patches.

**2.1. Outline of the Algorithm.** To rationalize a principal patch – that is, to evaluate the position of a point  $p$  at coordinate  $(u, v)$  – we seek a particular conformal mapping of a 2-manifold into 3D  $f : M \rightarrow \mathbb{R}^3$ . In what follows, we will formulate this map in terms of two orthogonal transformation generators,  $\tau_u$  and  $\tau_v$ , each one a 2-blade *point pair* in the conformal geometric algebra of  $\mathbb{R}^{4,1}$ .

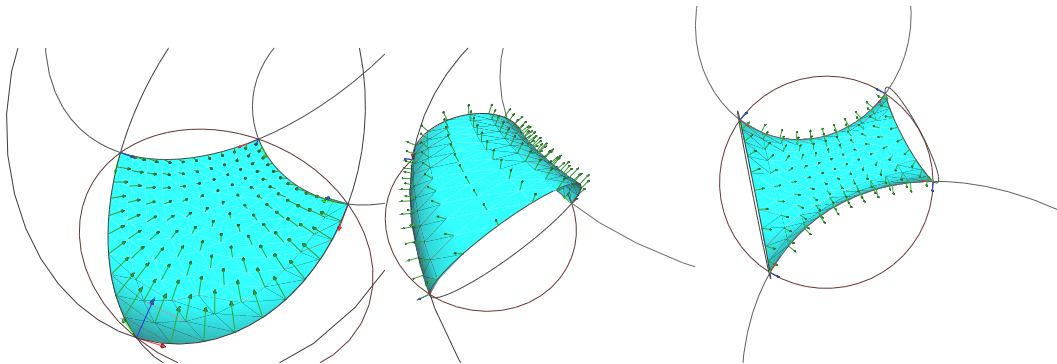


FIGURE 2.2. Given four points on a circle, one frame of orientation at one point (i.e. 3 degrees of freedom) is sufficient to fully characterize the patch.

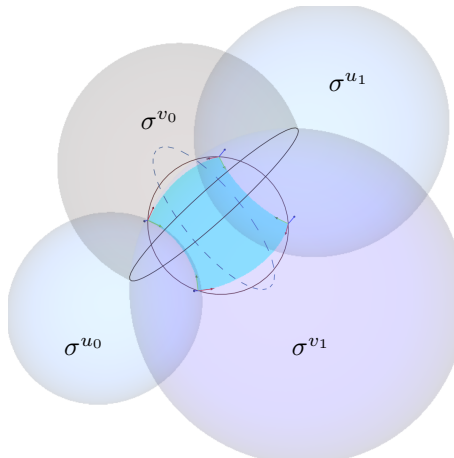


FIGURE 2.3. To construct a cyclidic patch, we generate rotors from spheres tangent to the circular edges of the patch. Individual principal 'simple' rotors  $\mathcal{R}_u$  and  $\mathcal{R}_v$  in the  $u$  and  $v$  directions of a patch are calculated as the *square root of the normalized ratio of coordinate surface spheres*. The logarithms of these rotors are point pairs. Here we render their dual representation as circles, both real (in the case the principal coordinate surface spheres intersect) and imaginary (when they do not).

Labelling each side of the patch  $u_0$ ,  $u_1$ ,  $v_0$ , and  $v_1$  (Figure 2.1) maps our edges to a 2-dimensional  $(u, v)$  - coordinate system in the range of  $[0, 1]$  where the subscript denotes a constant coordinate value, e.g.  $u_0 = 0$ . One transformation generator,  $\tau_u$ , will be responsible for interpolating  $u_0$  to  $u_1$  and the other,  $\tau_v$ , will be responsible for interpolating  $v_0$  to  $v_1$ .

Each edge is part of a coordinate curve *circle*  $\kappa^{u_0}, \kappa^{u_1}, \kappa^{v_0}$ , and  $\kappa^{v_1}$  which, as we will see, are themselves each part of larger coordinate surface *spheres*  $\sigma^{u_0}, \sigma^{u_1}, \sigma^{v_0}$ , and  $\sigma^{v_1}$ . We define  $\tau_u$  and  $\tau_v$  as the principal logarithms of conformal rotors  $\mathcal{R}_u$  and  $\mathcal{R}_v$ , which we define as the square root of the normalized ratio of coordinate surface spheres. Thus,  $\mathcal{R}_u = \sqrt{\frac{\sigma^{u_1}/\sigma^{u_0}}{\|\sigma^{u_1}/\sigma^{u_0}\|}}$  and  $\mathcal{R}_v = \sqrt{\frac{\sigma^{v_1}/\sigma^{v_0}}{\|\sigma^{v_1}/\sigma^{v_0}\|}}$ . These rotors encode the transformation that take one constant-coordinate surface ( $\sigma^{u_0}$ ) to another ( $\sigma^{u_1}$ ).

The logarithm of these rotors are 2-blade point pairs which can be linearly weighted to interpolate the transformation. In Figure 2.3 we draw the 2-blade's dual representation as circles. The geometric constraints on the cocircular system ensures that the point pair generators are orthogonal and commute, a condition which allows us to compose them into a *general conformal rotor*:  $K_{u_t v_t} = \mathcal{R}_{v_t} \mathcal{R}_{u_t}$ , with  $u_t$  and  $v_t$  each evaluated in the range of  $[0, 1]$  and  $\mathcal{R}_{u_t} = e^{\tau_{u_t}}$ . These rotors are applied to one corner of the patch,  $p$ , at  $(u_0, v_0)$  in order to transform it to a point at  $(u_t, v_t)$ . Thus,

$$(2.1) \quad f(u_t, v_t) = \mathcal{K}_{u_t v_t}[p]$$

**2.2. Constructing Tangents.** To better seat our treatment within the larger construction of a geometric calculus, we will emphasize the direct use of *tangent* elements accessible in the conformal geometric algebra. This also enables us to leverage the findings of Bobenko and Huhnen-Venedey in their use of Lie geometry, based as it is on *contact* elements. We therefore make explicit use of several basic operations in CGA relating tangent elements to round elements.<sup>1</sup> The first,

$$(2.2) \quad \kappa = p \rfloor \hat{\Sigma}$$

is the extraction of the tangent of a direct spheret  $\Sigma$  at a point  $p$  where the hat  $\hat{\cdot}$  signifies an *involution* (on a  $k$ -graded element:  $\hat{X}_k = (-1)^k X_k$ ). For a sphere  $\Sigma$ , equation 2.2 returns a null-valued (zero radius) circle. For a circle  $\kappa$  the same expression returns a null-valued point pair:

$$(2.3) \quad \tau = p \rfloor \hat{\kappa}$$

As a corollary we can build round elements from a tangent and a point:

$$(2.4) \quad \kappa = p \wedge \tau$$

which is the direct construction of a circle from a point  $p$  on it and tangent vector element  $\tau$  along it. Similarly,

$$(2.5) \quad \Sigma = p \wedge \kappa$$

is the direct construction of a sphere from a point  $p$  and tangent bivector element  $\kappa$ . We also find it a useful to remember that homogenous tangent vector elements  $\tau$  and  $\kappa$  are easily constructed by translation of a tangent vector and bivector at the origin, and that geometrically speaking they are null point pairs and null circles, possessing a weight and an orientation but no radius.

**2.3. Triply Orthogonal Coordinates.** As Figure 2.4 demonstrates, we can begin our synthetic construction with the notion of a triply orthogonal local frame  $\{e_k | k \in 1, 2, 3\}$  at  $p$ . To emphasize the fact that our frame is a *null tangent frame* at a point in space, we write  $\{\tau_k\}$  to signify the tangent vectors  $\{e_k\}$  have been translated to point  $p$ :

$$(2.6) \quad \tau_i = \mathcal{T}_p[n_o v_i]$$

where  $\mathcal{T}_p$  is a translation rotor and  $v_i$  is the Euclidean vector of our local  $e_i$ .

We label  $\sigma_j^i$  the principal contact sphere with normal  $e_i$  at  $p$  along direction  $e_j$ , and should be understood as representing a constant coordinate surface along the  $e_j$  direction. The raising of the index associates our construction with Hestenes' notation for curvilinear coordinate systems in [16]. In the case that it is unambiguous in which direction we are moving, we drop the lower index.<sup>2</sup> In our notation for cyclidic nets of the previous section, we omit the subscript since it is clear in which direction we are moving, and instead specify only the coordinate that is held constant. The full notation for  $\sigma^{u_0}$  would read  $\sigma_v^{u_0}$  to signify a surface of constant coordinate ( $u = 0$ ) as we move the in the  $v$  direction along the patch.

The coordinate curve in the  $e_i$  direction is defined as the union of constant coordinate surfaces  $\wedge \sigma_i^j, j \neq i$ . Note that in this sense each surface  $\sigma_i^j$  encodes a partial derivative  $\frac{\partial j}{\partial i}$  and each curve  $\kappa_i$  can be thought of as the exterior product of partial derivatives which contribute to it's definition (see footnote 2).

<sup>1</sup>Many components of these algorithms can be found in the essential table 14.1 on page 407 of [10], and in section 15.2 of that text.

<sup>2</sup>In [16], Hestenes proposes the "tangential derivative as the most fundamental of all concepts of derivative", and explains that the relation between a tangential frame  $\{e_k\}$  on a manifold and its inverse *reciprocal* frame  $\{e^k\}$  reveals that all "the coordinate curves are intersections of coordinate surfaces" (p.34, emphasis in original). Intriguingly, this relationship is precisely Dupin's theorem, and suggests to us that the rationalization of cyclidic transformations using the mappings of orthogonal conformal rotors could provide clues as to how to explore differentiable manifolds in general in the conformal model – namely, by treating *reciprocal tangent frames as normals to some dual sphere coordinate surface*.

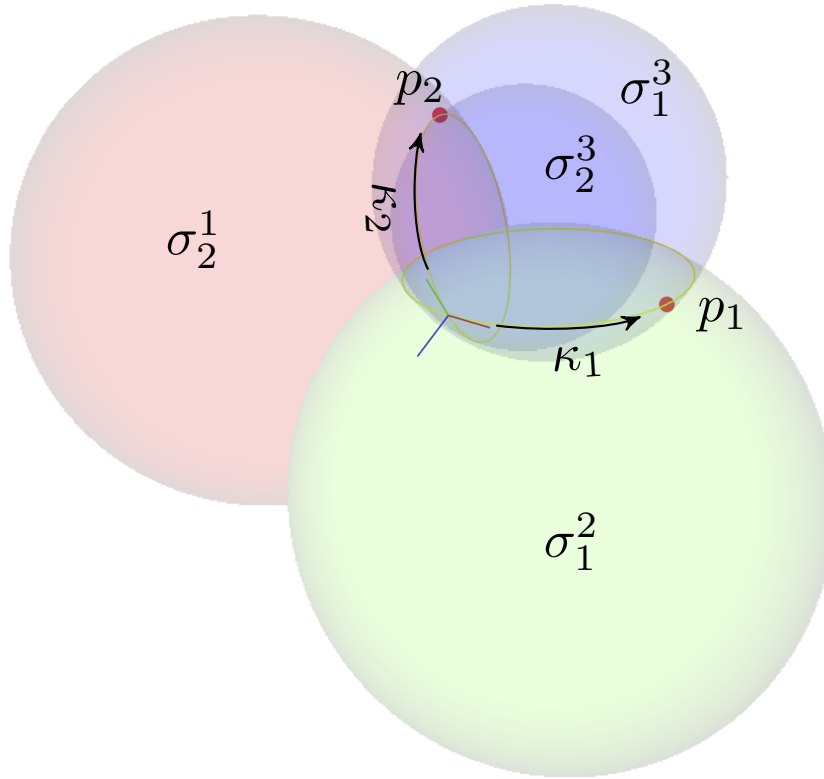


FIGURE 2.4. To construct *triply orthogonal curvilinear coordinate surfaces* we first consider an orthogonal tangent frame  $\{e_k\}$  at  $p$  with  $k = 1, 2, 3$ . *Coordinate surfaces*  $\{\sigma_j^k\}$  are spheres tangent to the frame. We use superscripts to denote the coordinate that is held constant, and subscripts to denote the tangent direction along which we travel. To create *coordinate curves*, we specify two surfaces (in blue) with  $e_3$  as a normal, each one corresponding to a different tangent direction. The coordinate curves  $\kappa_1$  and  $\kappa_2$  are circle intersections of these contact surfaces, a relation known as Dupin's theorem. Picking two points  $p_1$  and  $p_2$  along these curves defines a circle with  $p$ .

**2.4. Null Tangents as Coordinate Surface Generators.** One can create coordinate surfaces by applying a transversion (or *boost*) at  $p$  in the  $e_k$  direction to a dual plane through  $p$  with  $e_k$  normal, thereby bending it into a dual sphere through  $p$ . As we discussed in [4], such a curvature generator is constructed by translating a tangent vector from the origin to  $p$ , thereby creating a null point pair  $\tau$  which squares to 0. Our coordinate surface rotor becomes

$$(2.7) \quad \mathcal{R} = 1 - \frac{\tau}{2},$$

a simple result which stems directly from following Dorst and Valkenburg's rules for exponentiation of a 2-blade:

$$(2.8) \quad \mathcal{R} = e^{-\tau/2} = \cosh(\tau/2) - \sinh(\tau/2)$$

where

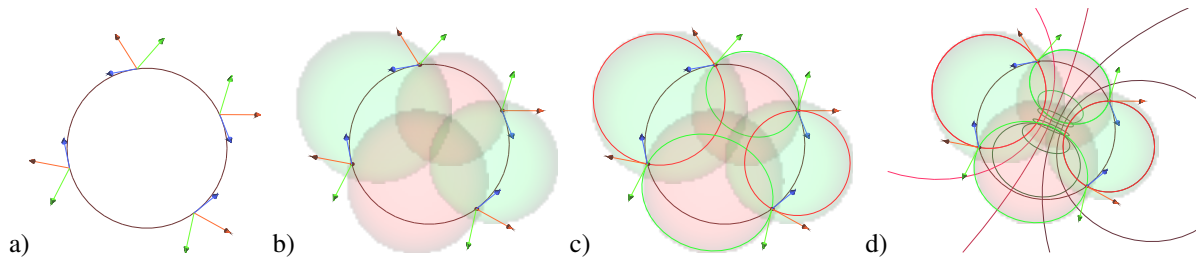


FIGURE 2.5. a) Given four cocircular points and one frame at one point, all other frames are calculated by applying equation 2.12 counter-clockwise from the known frame. We thereby extract tangent bivectors from triply orthogonal coordinate surfaces generated at each point. Here we draw the normalized dual of the resulting tangent bivectors. Note that consecutive tangent frames are reflections of each other. b) To generate a patch, we will interpolate between two orthogonal coordinate surfaces bilinearly. c) Coordinate curves along the edges of our patch are circles. d) The rotor ratio of coordinate surfaces applied directly to the circular edge curves begins to suggest a surface.

$$(2.9) \quad \sinh(X) = \begin{cases} \frac{\sinh(\sqrt{X^2})}{\sqrt{X^2}} X & \text{if } X^2 > 0 \\ X & \text{if } X^2 = 0 \\ \frac{\sin(\sqrt{-X^2})}{\sqrt{-X^2}} X & \text{if } X^2 < 0 \end{cases}$$

and

$$(2.10) \quad \cosh(X) = \begin{cases} \cosh(\sqrt{X^2}) & \text{if } X^2 > 0 \\ 1 & \text{if } X^2 = 0 \\ \cos(\sqrt{-X^2}) & \text{if } X^2 < 0. \end{cases}$$

**2.5. Coordinate spheres at four points on a circle.** Of course if the  $p_1$  and  $p_2$  along the coordinate curve are already known, then the coordinate surfaces can be constructed directly by application of equation 2.5. This happens in practice when we are given four cocircular points  $\{p_{(i,j)}\}$  and a starting frame at one of them. Figures 2.1, 2.2 and 2.9 were in fact generated this way.  $\tau_2$  of tangent frame  $\{\tau_k\}$  at  $p_{(0,0)}$ , notated  $\tau_{(0,0)}^2$ , is dual to a tangent bivector  $\kappa_{(0,0)}^2$  which wedged with the point  $p_{(1,0)}$ , creates a sphere  $\sigma$  whose own tangent vector  $\tau_{(1,0)}^2$  at  $p_{(1,0)}$  we can extract by application of equation 2.2:

$$\tau_{(1,0)}^2 = (p_{(1,0)} \rfloor ((\tau_{(0,0)}^2)^* \wedge p_{(1,0)}))^*.$$

The entire tangent frame at  $p_{(1,0)}$  can be similarly derived,

$$\tau_{(1,0)}^i = (p_{(1,0)} \rfloor ((\tau_{(0,0)}^i)^* \wedge p_{(1,0)}))^*$$

as can subsequent points  $p_{(1,1)}$  and  $p_{(0,1)}$ . If the points are numbered sequentially counterclockwise around the circle we have

$$(2.11) \quad \tau_j^i = (p_j \rfloor ((\tau_{j-1}^i)^* \wedge p_j))^*$$

More directly we can manipulate the tangent bivectors  $\kappa_j^i$  themselves,

$$(2.12) \quad \kappa_j^i = p_j \rfloor (\kappa_{j-1}^i \wedge p_j).$$

We visualize their dual representation as tangent vectors in figure 2.5.

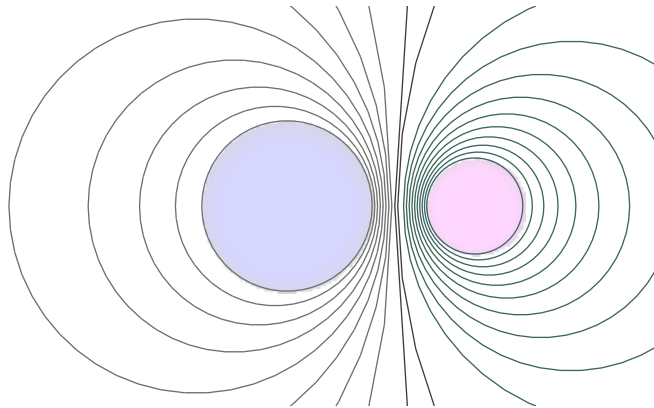


FIGURE 2.6. Taking a sphere to a sphere by exponentiation of the linearly weighted log of a simple rotor. Here we draw the circle cross section of the sphere during its transformation.

**2.6. Rationalization of Coordinate Surfaces with Simple Rotors.** With the system of principal contact elements in place, let us explore their *rationalization*, a method of discretizing the transformation of curvatures. In the intuitive logic of geometric algebra, we must calculate ratios of elements before we can rationalize their transformation. Thus we first calculate the ratio of spheres, for example to rationalize the surface in the  $u$  direction we find the ratio across it:

$$(2.13) \quad \mathcal{R}_s = \frac{\sigma^{u_1}}{\sigma^{u_0}}$$

which gives us a scalar and point pair bivector. We normalize this product by dividing out the reverse norm:  $\mathcal{R}_n = \mathcal{R}_s / \|\mathcal{R}_s\|$ , where the reverse norm is defined as

$$(2.14) \quad \|\mathcal{R}_s\| = \begin{cases} \sqrt{\mathcal{R}_s \cdot \tilde{\mathcal{R}}_s} & \text{if } \mathcal{R}_s \cdot \tilde{\mathcal{R}}_s > 0 \\ -\sqrt{-(\mathcal{R}_s \cdot \tilde{\mathcal{R}}_s)} & \text{if } \mathcal{R}_s \cdot \tilde{\mathcal{R}}_s < 0. \end{cases}$$

Our normalized rotor  $\mathcal{R}_n$  represents *twice* the transformation that takes  $\sigma^{u_0}$  to  $\sigma^{u_1}$  thus we must calculate the square root using Dorst and Valkenburg's method (equation 5.2 in [11]):

$$(2.15) \quad \mathcal{R}_u = \sqrt{\mathcal{R}_n} = \frac{1 + \mathcal{R}_n}{\sqrt{2(1 + \langle \mathcal{R}_n \rangle)}}.$$

$\mathcal{R}_u$  is the *linearizable* transformation from  $\sigma^{u_0}$  to  $\sigma^{u_1}$  and is the square root of normalized ratios – in short  $\mathcal{R}_u = \sqrt{\text{normalized}(\frac{\sigma^{u_1}}{\sigma^{u_0}})}$ . To actually linearize this rotor, we calculate its logarithm, which is the 2-blade point pair exponent  $-\tau_u/2$  in the expression  $\mathcal{R}_u = e^{-\tau_u/2}$ , and which we can weigh linearly and then re-exponentiate using equation 2.8. The logarithm itself is detailed in equations 5.20 and 5.21 of Dorst and Valkenburg's text, and is replicated here for convenience:

$$(2.16) \quad \log(\mathcal{R}) = \text{atanh2}(\langle \mathcal{R} \rangle_2, \langle \mathcal{R} \rangle)$$

where

$$(2.17) \quad \text{atanh2}(s, c) = \begin{cases} \frac{\text{asinh}(\sqrt{s^2})}{\sqrt{s^2}} s & \text{if } s^2 > 0 \\ s & \text{if } s^2 = 0 \\ \frac{\text{atan2}(\sqrt{-s^2}, c)}{\sqrt{-s^2}} s & \text{if } -1 \leq s^2 < 0. \end{cases}$$



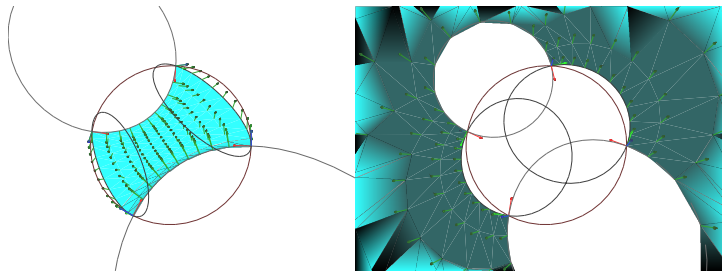


FIGURE 2.7. Two directions of interpolation of a conformal rotor are possible. Additional precautions are needed to determine the correct direction and to determine the full interval of a given direction.

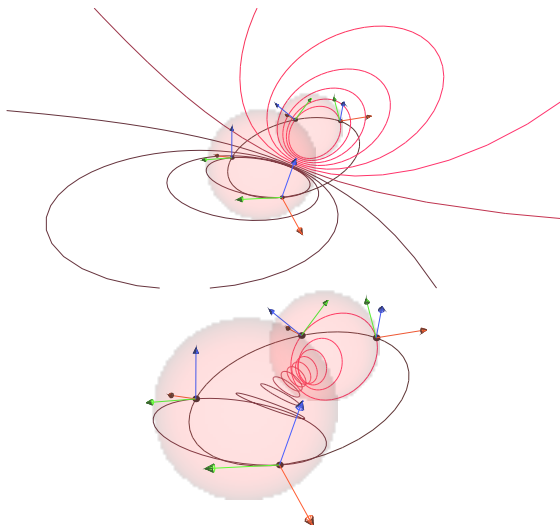


FIGURE 2.8. Given two intersecting spheres, there are two directions that one sphere can conformally transform into the other. Here we render circles undergoing transformation in both directions.

We calculate  $\tau_v$  the same way, and now have the components necessary to plug into equation 2.1. Below we make a simple modification to equation 2.17 to account for two choices of direction of motion around the orbit.

**2.7. The Direction of Interpolation.** Given two *intersecting* spheres, the conformal transformation between them can occur clockwise or counterclockwise. As in any rotation, the direction in which the transformation occurs determines the points that are evaluated along the way. Figures 2.7 and 2.8 illustrate this difference, and present the need for additional measures be taken to ensure the correct orientation of rationalization.

The direction is determined by the log function. The default direction uses

$$(2.18) \quad \frac{\text{atan2}(\sqrt{-s^2}, c)}{\sqrt{-s^2}} s$$

whereas an alternative uses a modified weighting in the opposite direction:

$$(2.19) \quad \frac{-(\pi - \text{atan2}(\sqrt{-s^2}, c))}{\sqrt{-s^2}} s$$

We find that using equation 2.19 is necessary for calculating  $\tau_u$  (resp.,  $\tau_v$ ) precisely when our initial corner point  $p_{(0,0)}$  has a negative dot product with the opposing coordinate surface sphere  $\sigma^{u_1}$  (resp.,  $\sigma^{v_1}$ ). Calling our alternative  $\log_{alt}$  we have

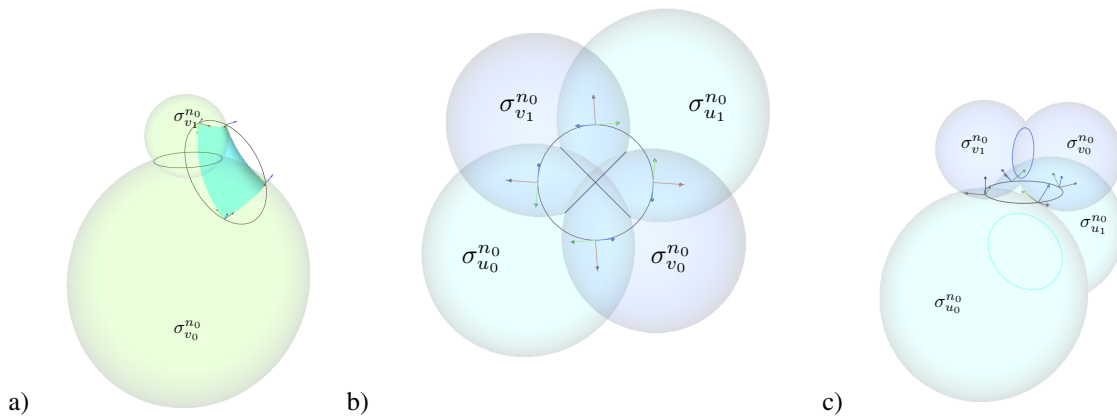


FIGURE 2.9. a) A patch is a  $C^1$ -smooth blend between coordinate surfaces of constant coordinate  $n_0$ , that is as a constant along the *normal* axis. Here  $\sigma_{v_0}^{n_0}$  (resp.  $\sigma_{v_1}^{n_0}$ ) represents the principal contact sphere change in the normal along the  $v_0$  (resp.  $v_1$ ) edges. b) and c) Two views of principal contact spheres tangent to the surface frames.

$$(2.20) \quad \tau_{u,v} = \begin{cases} \log_{\text{alt}}(\mathcal{R}_{u,v}) & \text{if } p_{(0,0)} \cdot \sigma^{u_1,v_1} < 0 \\ \log(\mathcal{R}_{u,v}) & \text{otherwise.} \end{cases}$$

**2.8. Surface Blending.** Thus far (with the exception of Figure 2.4) we have primarily visualized coordinate surfaces in two directions:  $\sigma_v^u$  and  $\sigma_u^v$ . Here we discuss the third coordinate surface direction –  $\sigma^n$  – which can be constructed with respect to both  $u$  and  $v$  as  $\sigma_u^n$  and  $\sigma_v^n$ . These surfaces are the *osculating* spheres tangent to the  $uv$  surface. We use the notation  $\sigma^{n_0}$  on these surfaces to indicate that  $uv$  surface itself lies on the constant ( $n_0 = 0$ ) in the direction normal to the surface. Given two spheres and a circle through both is sufficient to characterize an orthogonal curvilinear coordinate system at each point on the circle from which a patch can be constructed (Figure 2.9a).

Not *any* circle will suffice, however, and here we detail a simple construction for blending between two contact spheres. Figure 2.10 illustrates the method, and Figure 2.11 depicts various results.

- (1) Pick a point  $p$  on one of the spheres  $\sigma_{v_1}$ .
- (2) Find the plunge  $\kappa$  through  $p$  orthogonal to both spheres:  $\kappa = p \wedge \sigma_{v_0} \wedge \sigma_{v_1}$ .
- (3) Find the first intersection point of  $\kappa$  with the second sphere  $\sigma_{v_0}$ .
- (4) Repeat for another point on  $\sigma_{v_1}$ .

These four points will be cocircular and it only remains to define  $\{\tau_k\}$  for each frame.  $\tau_3$  is given by the point positions on the contact spheres. Defining some rotation about that axis for point  $p$  will determine the positions of all four frames.

### 3. DISCUSSION AND FUTURE WORK

We have introduced a method of rationalizing cyclidic nets by composing conformal transformations from two orthogonal coordinate surface pairs. Our technique emerges as a direct result of the orthogonal decomposition of general conformal transformations uncovered by Dorst and Valkenburg in [11].

A good next step in our formulation will be to analyze a given *simplicial* surface and try to find its closed conformal representation through piecewise integration of cyclidic patches. In the literature, generating a circle packing on the triangularized surface is often the first step towards such a construction. Meshing of *point cloud* data could also stand to benefit from our conformal immersions. Using CGA, Hildenbrand has explored sphere fitting of point clouds in [19] and with Seibert and Becker in [24], a topic also tackled by Dorst in [8]. Another interesting avenue will be to see whether our methods can extend to blending via the more general Darboux cyclides.

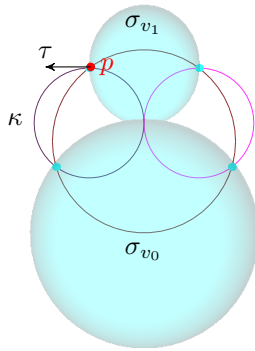


FIGURE 2.10. Construction of a circle net to blend between two spheres. Given a point  $p$  on sphere  $\sigma_{v_1}$ , we calculate the *orthogonal plunge*  $\kappa$  by wedging  $p$  with both spheres:  $\kappa = p \wedge \sigma_{v_0} \wedge \sigma_{v_1}$ . We then find the intersection of  $\kappa$  with  $\sigma_{v_0}$ . We repeat for a second point on  $\sigma_{v_1}$ . All four points thus defined will be cocircular.  $\tau$  is the tangent normal to the surface patch at  $p$ .

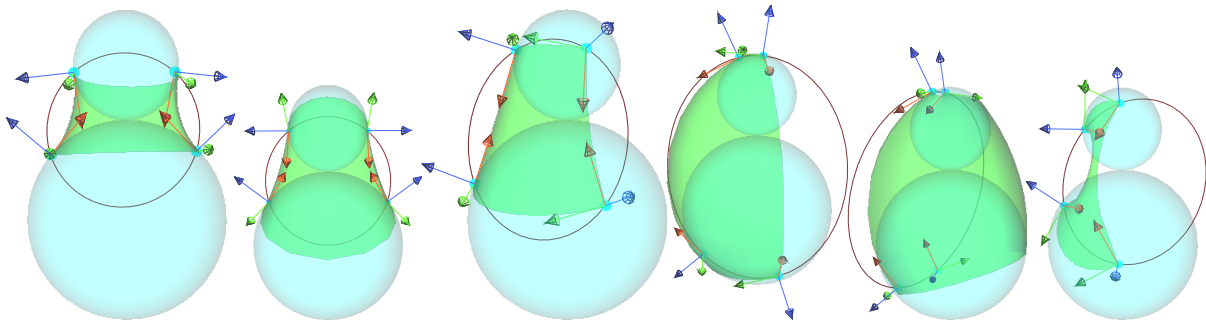


FIGURE 2.11. Various patches blending the same two spheres.

Because of the structure-preserving nature of conformal transformations within the CGA mechanism, our composed transformations are as capable of operating on tangents and normals as well as points, greatly simplifying basic calculations in differential geometry. We suspect that approaching differential geometry by careful study of integration of conformal mappings will help in developing the discrete geometric calculus within the conformal model. In our treatment, our frames are already orthogonal – a condition which can be applied to non-orthogonal frames using the reciprocal construction of geometric calculus. Hestenes’ writings are, as usual, a good place to start this mapping [15, 17, 18] as is Sobczyk’s simplicial calculus [25].

Finally, we would like to more carefully consider the relationship between the rotors that transforms these spheres across a surface patch and the *shape tensor* and *shape bivector* or *curl*, to better pin-point the pair generators that most clearly and generally match Hestenes’ definition of the shape bivector as the “angular velocity of the pseudoscalar as it slides along the manifold” [18]. Explicating such relationships will give more space for a discrete calculus to form.

#### 4. ACKNOWLEDGMENTS

This work was funded by the Robert W. Deutsch Foundation, through the Allosphere Research Facility led by JoAnn Kuchera-Morin at UCSB’s Media Arts and Technology Program, as well as by the Systemics Lab led by Marko Peljhan. The author would also like to thank Leo Dorst, whose suggestions, solutions, and hints have proved invaluable in developing – indeed in even considering – this formulation. In addition, this paper has directly benefitted from communications with Wesley Smith, Garret Sobczyk, Dietmar Hildenbrand, and Ian Bell. Of course any errors or omissions are to be attributed to the author alone.

## REFERENCES

- [1] Eduardo Bayro-Corrochano and Luis Eduardo Falcón. Geometric algebra of points, lines, planes and spheres for computer vision and robotics. *Robotica*, 23(06):755–770, 11 2005.
- [2] Pengbo Bo, Helmut Pottmann, Martin Kilian, Wenping Wang, and Johannes Wallner. Circular arc structures. *ACM Trans. Graph.*, 30(4):101:1–101:12, July 2011.
- [3] Alexander I. Bobenko and Emanuel Huhnen-Venedey. Curvature line parametrized surfaces and orthogonal coordinate systems: discretization with dupin cyclides. *Geometriae Dedicata*, 159(1):207–237, 2012.
- [4] Pablo Colapinto. Boosted surfaces: Synthesis of meshes using point pair generators as curvature operators in the 3d conformal model. *Advances in Applied Clifford Algebras*, 24(1):71–88, 2014.
- [5] Keenan Crane, Ulrich Pinkall, and Peter Schröder. Spin transformations of discrete surfaces. *ACM Trans. Graph.*, 30(4):104, 2011.
- [6] Keenan Crane, Ulrich Pinkall, and Peter Schröder. Robust fairing via conformal curvature flow. *ACM Trans. Graph.*, 32(4):61, 2013.
- [7] Chris Doran. Circle and sphere blending with conformal geometric algebra, 2003. cite arxiv:cs/0310017Comment: 20 pages, 13 figures.
- [8] Leo Dorst. Total least squares fitting of k-spheres in n-d euclidean space using an (n+2)-d isometric representation. *Journal of Mathematical Imaging and Vision*, 50(3):214–234, 2014.
- [9] Leo Dorst. The construction of 3d conformal motions. In *Proceedings of Applications of Geometric Algebra in Computer Science and Engineering*, 2015.
- [10] Leo Dorst and Stephen Mann. *Geometric algebra for computer science an object-oriented approach to geometry*. Elsevier ; Morgan Kaufmann, Amsterdam; San Francisco, 2007.
- [11] Leo Dorst and Robert Valkenburg. Square root and logarithm of rotors in 3d conformal geometric algebra using polar decomposition. In Leo Dorst and Joan Lasenby, editors, *Guide to Geometric Algebra in Practice*, pages 81–104. Springer, 2011.
- [12] Lucie Druoton, Laurent Fuchs, Lionel Garnier, and Rémi Langevin. The non-degenerate dupin cyclides in the space of spheres using geometric algebra. *Advances in Applied Clifford Algebras*, 24(2):515–532, 2014.
- [13] Charles Dupin. *Applications de géométrie de mécanique à la marine, aux ponts et chaussées; pour faire suite aux développemens de géométrie*. 1822.
- [14] Sebti Foufou and Lionel Garnier. Dupin cyclide blends between quadric surfaces for shape modeling. *Comput. Graph. Forum*, 23(3):321–330, 2004.
- [15] David Hestenes. Differential forms in geometric calculus. In F. Brackx, R. Delanghe, and H. Serras, editors, *Clifford Algebras and their Applications in Mathematical Physics*, volume 55 of *Fundamental Theories of Physics*, pages 269–285. Springer Netherlands, 1993.
- [16] David Hestenes. Geometric calculus, 1998. Available at <http://geocalc.clas.asu.edu/pdf/NFMPchapt2.pdf>.
- [17] David Hestenes. The shape of differential geometry in geometric calculus. In Leo Dorst and Joan Lasenby, editors, *Guide to Geometric Algebra in Practice*, pages 393–410. Springer, 2011.
- [18] David Hestenes and Garret Sobczyk. *Clifford algebra to geometric calculus : a unified language for mathematics and physics*. D. Reidel ; Distributed in the U.S.A. and Canada by Kluwer Academic Publishers, Dordrecht; Boston; Hingham, MA, U.S.A., 1984.
- [19] Dietmar Hildenbrand. Geometric computing in computer graphics using conformal geometric algebra. *Computers and Graphics*, 29(5):795 – 803, 2005.
- [20] Rimvydas Krasauskas and Severinas Zube. Bezier-like parametrizations of spheres and cyclides using geometric algebra. In K Guerlebeck, editor, *Proceedings of 9th International Conference on Clifford Algebras and their Applications in Mathematical Physics*, Weimar, Germany, 2011.
- [21] Hongbo Li, David Hestenes, and Alyn Rockwood. Generalized homogeneous coordinates for computational geometry. In Gerald Sommer, editor, *Geometric Computing with Clifford Algebras*, pages 27–59. Springer Berlin Heidelberg, 2001.
- [22] Ralph Robert Martin. *Principal Patches for Computational Geometry*. PhD thesis, Pembroke College, Cambridge University, 1983.
- [23] Helmut Pottmann, Ling Shi, and Mikhail Skopenkov. Darboux cyclides and webs from circles. *Computer Aided Geometric Design*, 29(1):77 – 97, 2012. Geometric Constraints and Reasoning.
- [24] Helmut Seibert, Dietmar Hildenbrand, Meike Becker, and et al. *Estimation of curvatures in point sets based on geometric algebra*. 2010.
- [25] G.E. Sobczyk. Simplicial calculus with geometric algebra. In A. Micali, R. Boudet, and J. Helmstetter, editors, *Clifford Algebras and their Applications in Mathematical Physics*, volume 47 of *Fundamental Theories of Physics*, pages 279–292. Springer Netherlands, 1992.



# THREE CLIFFORD ALGEBRAS FOR FOUR KINDS OF INTERACTIONS

C. Daviau<sup>a</sup> and J. Bertrand<sup>b</sup>

<sup>a</sup> Le Moulin de la Lande, 44522 Pouillé-les-coteaux, France  
claude.daviau@nordnet.fr [presenter, corresponding]

<sup>b</sup> 15 Avenue Danielle Casanova, 95210 St Gratien, France  
bertrandjacques-m@orange.fr

ABSTRACT. Three Clifford algebras are sufficient to describe all interactions of modern physics: The Clifford algebra of the usual space is enough to describe all aspects of electromagnetism, including the quantum wave of the electron. The Clifford algebra of space-time is enough to describe electro-weak interactions. To get the gauge group of the standard model, with electro-weak and strong interactions, a third algebra is sufficient, with only two more dimensions of space. This enlarged frame allows to include also gravitation.

## 1. WHY SPACE ALGEBRA ?

Why Clifford algebras are important in physics? Physics uses waves and the Fourier theorem says that any periodic function may be decomposed in a sum of sin and cos functions. These functions are more easily studied with the complex exponential function. Moreover the exponential function is the main tool in Lie groups. The exponential function needs products, so we must also know how to multiply numbers and vectors and products of vectors: we need an algebra.

Why space algebra ? The first reason is simply that our physical space is 3-dimensional. The second reason is the spin 1/2 which implies the use of  $SL(2, \mathbb{C})$  that is a subgroup of  $Cl_3^* = GL(2, \mathbb{C})$ , true group of form invariance of the electromagnetism, wave of the electron included. We have previously shown, first in ICCA7 in Toulouse, how this group of all invertible elements in the space algebra  $Cl_3$  acts. Firstly time and space take naturally place in the auto-adjoint part of this algebra: With

$$(1) \quad x^0 = ct \ ; \ \vec{x} = x^1 \sigma_1 + x^2 \sigma_2 + x^3 \sigma_3 \ ; \ \partial_\mu = \frac{\partial}{\partial x^\mu}$$

we let [1][2][3][4][5][6][7][11][12][13]

$$(2) \quad x = x^0 + \vec{x} = \begin{pmatrix} x^0 + x^3 & x^1 - ix^2 \\ x^1 + ix^2 & x^0 - x^3 \end{pmatrix}.$$

Space-time is made of the auto-adjoint part of the space algebra

$$(3) \quad \hat{x} = \bar{x} = x^0 - \vec{x} \ ; \ x^\dagger = x$$

$$(4) \quad \det(x) = x\hat{x} = x \cdot x = (x^0)^2 - \vec{x}^2 = (x^0)^2 - (x^1)^2 - (x^2)^2 - (x^3)^2$$

This allows to read the Dirac wave of the electron and its wave equation in the space algebra. The link with the old complex formalism is simple if we use the left and right Weyl spinors  $\eta$  and  $\xi$  by letting:

$$(5) \quad \hat{\phi} = \sqrt{2}(\eta \ -i\sigma_2 \xi^*) = \sqrt{2} \begin{pmatrix} \eta_1 & -\xi_2^* \\ \eta_2 & \xi_1^* \end{pmatrix} \ ; \ \phi = \sqrt{2}(\xi \ -i\sigma_2 \eta^*) = \sqrt{2} \begin{pmatrix} \xi_1 & -\eta_2^* \\ \xi_2 & \eta_1^* \end{pmatrix}$$

---

Date: June 5, 2015.

The conjugation  $h : \phi \mapsto \widehat{\phi}$  is the main automorphism of  $Cl_3$ .  $M^\dagger$  being the transposed conjugate of  $M$  and using

$$(6) \quad \overline{\phi} = \widehat{\phi}^\dagger$$

the homogeneous non-linear wave equation of the electron, which has the Dirac equation as linear approximation, reads:

$$(7) \quad \overline{\phi}(\nabla\widehat{\phi})\sigma_{21} + \overline{\phi}qA\widehat{\phi} + m\rho = 0 ; \quad \nabla = \sigma^\mu \partial_\mu ; \quad \sigma_{21} = \sigma_2\sigma_1 = -i\sigma_3 ; \quad \rho = |\det(\phi)|.$$

where  $q = e/\hbar c$ ,  $m = m_0c/\hbar$ ,  $\sigma^0 = \sigma_0$ ,  $\sigma^j = -\sigma_j$ ,  $j = 1, 2, 3$ . This equation is invariant under any transformation  $R$  defined by an element  $M$  of the Lie group  $Cl_3^*$ :

$$(8) \quad x' = R(x) = MxM^\dagger$$

$$(9) \quad \phi'(x') = M\phi(x)$$

$$(10) \quad \nabla = \overline{M}\nabla'\widehat{M} ; \quad qA = \overline{M}q'A'\widehat{M}$$

$$(11) \quad m\rho = m'\rho' ; \quad \rho' = |\det(\phi')|.$$

Only one  $M$  term is present in (9) when two  $M$  terms are present in (8) and (10): consequently the wave turns with a  $\theta$  angle when the space turns with a  $2\theta$  angle. The application

$$(12) \quad f : M \mapsto R$$

is not an isomorphism, but only an homomorphism from  $Cl_3^*$  into a group of geometric transformations that I named Lorentz dilations. They are the product, in any order, of an element of the restricted Lorentz group  $\mathcal{L}_+^\uparrow$  by a homothety with ratio  $r = |\det(M)|$ . The kernel of  $f$  is the chiral group generated by  $i = \sigma_1\sigma_2\sigma_3$  which orients the space. Consequently the chirality is essential in quantum mechanics, it is present in the transformation of the wave (9), and this chirality disappears in the geometry of space-time since it is absent in (8). Next, space-time vectors are said contravariant if they satisfy (8) and covariant if they satisfy (10). In these equalities  $\nabla$  and  $qA$  are similarly transformed and this allows the gauge invariance under the electric gauge transformation:

$$(13) \quad \phi \mapsto \phi' = \phi e^{ia\sigma_3} ; \quad A_\mu \mapsto A_\mu' = A_\mu - \frac{1}{q}\partial_\mu a$$

The existence here of the 3 index is the reason of the existence of three generations of fundamental fermions. The invariance of the  $m\rho$  product instead of the invariance of  $m$  and  $\rho$  separately is the reason of the existence of the Planck constant [7][11][12]. The space algebra  $Cl_3$  is 8-dimensional on  $\mathbb{R}$ , therefore the wave equation (7) is equivalent to a system of 8 numeric equations with partial derivatives. With the basis  $(1, \sigma_1, \sigma_2, \sigma_3, i\sigma_1, i\sigma_2, i\sigma_3, i)$  of  $Cl_3$  the equation corresponding respectively to 1 and  $i\sigma_3$  are [9]

$$(14) \quad \mathcal{L} = 0$$

$$(15) \quad \partial_\mu J^\mu = 0 ; \quad J = J^\mu \sigma_\mu = \phi \phi^\dagger$$

where  $J$  is the current of density of probability and  $\mathcal{L}$  is the Lagrangian density. Then the law of conservation of the  $J$  current is a part of the wave equation and similarly (14) is a part of the wave equation. The wave equation comes from a Lagrangian mechanism and (this is new) the wave equation contains  $\mathcal{L} = 0$ . This explains why there is a principle of extremum in quantum mechanics. We have previously explained how classical electromagnetism, without or with magnetic monopoles, reads in space algebra (chapter 4 of [11] and [12]).

## 2. THE ELECTRO-WEAK GAUGE IN SPACE-TIME ALGEBRA

Two non equivalent homomorphisms exist from  $Cl_3^*$  into the group of dilations on the space-time. This is the origin of the existence of left and right waves, that are fundamental for weak interactions. We have obtained [9] in the  $Cl_{1,3}$  algebra a wave equation with mass term for

the pair electron+neutrino, both form invariant (then relativistic invariant) and gauge invariant under the  $U(1) \times SU(2)$  gauge group of electro-weak interactions. The wave  $\Psi_l$  of the electron+neutrino is a function of space and time with value in the space-time algebra. It reads:

$$(16) \quad \Psi = \begin{pmatrix} \phi_e & \hat{\phi}_n \\ \hat{\phi}_a \sigma_1 & \hat{\phi}_p \sigma_1 \end{pmatrix} = \begin{pmatrix} \phi_e & \hat{\phi}_n \\ \hat{\phi}_n & \hat{\phi}_e \end{pmatrix}$$

where indexes  $e, n, p$  and  $a$  indicate the respective waves of the electron, the electronic neutrino, the positron and the electronic anti-neutrino. They satisfy:

$$(17) \quad \phi_e = \sqrt{2}(\xi_e - i\sigma_2 \eta_e^*) ; \hat{\phi}_e = \sqrt{2}(\eta_e - i\sigma_2 \xi_e^*)$$

$$(18) \quad \phi_n = \sqrt{2}(\xi_n - i\sigma_2 \eta_n^*) ; \hat{\phi}_n = \sqrt{2}(\eta_n - i\sigma_2 \xi_n^*)$$

$$(19) \quad \hat{\phi}_p = \hat{\phi}_e \sigma_1 ; \hat{\phi}_a = \hat{\phi}_n \sigma_1.$$

The Weinberg-Salam model [14] has no use for the right wave  $\xi_n$  of the neutrino and cancels this term. This gives

$$(20) \quad \phi_n = \sqrt{2}(0 - i\sigma_2 \eta_n^*) ; \hat{\phi}_n = \sqrt{2}(\eta_n \ 0)$$

$$(21) \quad \hat{\phi}_p = \sqrt{2}(\eta_p - i\sigma_2 \xi_p^*) ; \phi_p = \sqrt{2}(\xi_p - i\sigma_2 \eta_p^*)$$

$$(22) \quad \hat{\phi}_a = \sqrt{2}(0 - i\sigma_2 \xi_a^*) ; \phi_a = \sqrt{2}(\xi_a \ 0)$$

$$(23) \quad \xi_{1p} = \eta_{2e}^*, \ \xi_{2p} = -\eta_{1e}^*; \ \eta_{1p} = -\xi_{2e}^*; \ \eta_{2p} = \xi_{1e}^*$$

$$(23) \quad \xi_{1a} = \eta_{2n}^*, \ \xi_{2a} = -\eta_{1n}^*.$$

The charge conjugation is the same as in the standard model of quantum physics. Instead of the  $36 = 8 \times 9/2$  tensorial densities built from the wave of the electron we get now  $78 = 12 \times 13/2$  tensorial densities built from the  $12 = 8 + 4$  parameters of the wave of electron+neutrino. We let

$$(24) \quad a_1 = \det(\phi_e) = \rho e^{i\beta} = 2(\xi_{1e} \eta_{1e}^* + \xi_{2e} \eta_{2e}^*)$$

$$(25) \quad a_2 = 2(\eta_{1e}^* \eta_{2n}^* - \eta_{2e}^* \eta_{1n}^*)$$

$$(26) \quad a_3 = 2(\xi_{1e} \eta_{1n}^* + \xi_{2e} \eta_{2n}^*)$$

$$(27) \quad \rho = \sqrt{a_1 a_1^* + a_2 a_2^* + a_3 a_3^*}.$$

The invariant wave equation of electron+neutrino reads

$$(28) \quad \tilde{\Psi}_l(\mathbf{D}\Psi_l)\gamma_{012} + m\rho\tilde{\Psi}_l\chi_l = 0$$

where  $\tilde{\Psi}_l$  is the reverse of  $\Psi_l$  and where  $\chi_l$  reads:

$$(29) \quad \chi_l = \frac{1}{\rho^2} \begin{pmatrix} a_1^* \phi_e + a_2^* \phi_n \sigma_1 + a_3^* \phi_n & -a_2^* \phi_{eL} \sigma_1 + a_3^* \phi_{eR} \\ a_2 \hat{\phi}_{eL} \sigma_1 + a_3 \hat{\phi}_{eR} & a_1 \hat{\phi}_e - a_2 \hat{\phi}_n \sigma_1 + a_3 \hat{\phi}_n \end{pmatrix}$$

with  $\phi_{eR} = \phi_e(1 + \sigma_3)/2$ , right part of  $\phi_e$ ,  $\phi_{eL} = \phi_e(1 - \sigma_3)/2$  left part of  $\phi_e$ . The covariant derivative  $\mathbf{D}$  uses four operators  $P_\mu$  which form a basis of the Lie algebra of  $U(1) \times SU(2)$  and reads:

$$(30) \quad P_\pm(\Psi_l) = \frac{1}{2}(\Psi_l \pm \mathbf{i}\Psi_l \gamma_{21}) ; \mathbf{i} = \gamma_{0123}$$

$$(31) \quad P_0(\Psi_l) = \Psi_l \gamma_{21} + \frac{1}{2}\Psi_l \mathbf{i} + \frac{1}{2}\mathbf{i}\Psi_l \gamma_{30} = \Psi_l \gamma_{21} + P_-(\Psi_l) \mathbf{i}$$

$$(32) \quad P_1(\Psi_l) = \frac{1}{2}(\mathbf{i}\Psi_l \gamma_0 + \Psi_l \gamma_{012}) = P_+(\Psi_l) \gamma_3 \mathbf{i}$$

$$(33) \quad P_2(\Psi_l) = \frac{1}{2}(\Psi_l \gamma_3 - \mathbf{i}\Psi_l \gamma_{123}) = P_+(\Psi_l) \gamma_3$$

$$(34) \quad P_3(\Psi_l) = \frac{1}{2}(-\Psi_l \mathbf{i} + \mathbf{i}\Psi_l \gamma_{30}) = P_+(\Psi_l)(-\mathbf{i}).$$



The covariant derivative uses also:

$$(35) \quad D_\mu = \partial_\mu - ig_1 \frac{Y}{2} B_\mu - ig_2 T_j W_\mu^j$$

with  $T_j = \frac{\tau_j}{2}$  for a doublet of left-handed particles and  $T_j = 0$  for a singlet of right-handed particle.  $Y$  is the weak hypercharge,  $Y_L = -1$ ,  $Y_R = -2$  for the electron. To transpose into space-time algebra, we let

$$(36) \quad D = \sigma^\mu D_\mu; \quad \mathbf{D} = \gamma^\mu D_\mu = \begin{pmatrix} 0 & D \\ \widehat{D} & 0 \end{pmatrix}; \quad \gamma^0 = \gamma_0 = \begin{pmatrix} 0 & I_2 \\ I_2 & 0 \end{pmatrix}$$

$$(37) \quad B = \sigma^\mu B_\mu; \quad \mathbf{B} = \gamma^\mu B_\mu = \begin{pmatrix} 0 & B \\ \widehat{B} & 0 \end{pmatrix}; \quad \gamma^j = -\gamma_j = \begin{pmatrix} 0 & -\sigma_j \\ \sigma_j & 0 \end{pmatrix}, \quad j = 1, 2, 3$$

$$(38) \quad W^j = \sigma^\mu W_\mu^j; \quad \mathbf{W}^j = \gamma^\mu W_\mu^j = \begin{pmatrix} 0 & W^j \\ \widehat{W}^j & 0 \end{pmatrix}$$

We have proved [7][11][12] that the covariant derivative (35) of the Weinberg-Salam model is equivalent to

$$(39) \quad \mathbf{D} = \boldsymbol{\partial} + \frac{g_1}{2} \mathbf{B} P_0 + \frac{g_2}{2} (\mathbf{W}^1 P_1 + \mathbf{W}^2 P_2 + \mathbf{W}^3 P_3); \quad \boldsymbol{\partial} = \gamma^\mu \partial_\mu.$$

The wave equation (28) is invariant under the  $R$  transformation generated by any  $M$  element of  $Cl_3^*$ , this implies the relativistic invariance of the wave. It is also gauge invariant under the gauge transformation:

$$(40) \quad \Psi'_l = [\exp(a^\mu P_\mu)](\Psi_l)$$

where  $D_\mu \Psi_l$  is replaced by  $D'_\mu \Psi'_l$ :

$$(41) \quad D'_\mu \Psi'_l = \exp(a^\mu P_\mu) D_\mu \Psi_l$$

$$(42) \quad B'_\mu = B_\mu - \frac{2}{g_1} \partial_\mu a^0$$

$$(43) \quad W'^j P_j = \left[ \exp(S) W_\mu^j P_j - \frac{2}{g_2} \partial_\mu [\exp(S)] \right] \exp(-S); \quad S = a^1 P_1 + a^2 P_2 + a^3 P_3.$$

Contrarily to the Weinberg-Salam model, which was unable to manage an electron with a proper mass linking the right and left waves of the electron, our wave equation has a mass term and is fully gauge invariant. Therefore the complicated mechanism of spontaneous broken symmetry needed to rebuild a mass term is useless. The Higgs boson exists, but it shall not give a way to compute the proper mass of particles.

The proper mass in (28) is the proper mass of the electron alone, then the wave equation of electron+neutrino is reduced to the invariant wave equation (7) when the wave of the electronic neutrino is canceled. If the wave of the electron is canceled the mass term is canceled and we get for the neutrino the Dirac equation without mass:

$$(44) \quad \nabla \widehat{\phi}_n = 0$$

The real part of the wave equation is also  $\mathcal{L} = 0$ : the Lagrangian formalism is a consequence of the wave equation. There is also a conservative current, that is now

$$(45) \quad J = \phi_e \phi_e^\dagger + \phi_n \phi_n^\dagger; \quad \partial_\mu J^\mu = 0.$$

### 3. THE GAUGE GROUP OF THE STANDARD MODEL IN $Cl_{1,5}$

To get a wave function including electron+neutrino plus two quarks  $u$  and  $d$  with three states of color each the space-time algebra is not enough: even if quarks have only left waves these 9 spinor wave (two spinors for the electron) necessitates 36 parameters and  $36 < 2^n$  if  $n$  is equal or greater than 6. We have proved [10][12] that  $Cl_{1,5}$  is enough to get the complete wave as a function of space-time in the Clifford algebra of this extended space-time. Three states of

“color” are named r, g, b (red, green, blue). So we build a wave with all fermions of the first generation as

$$(46) \quad \Psi = \begin{pmatrix} \Psi_l & \Psi_r \\ \Psi_g & \Psi_b \end{pmatrix}$$

where  $\Psi_l$  is defined by (16) and  $\Psi_r, \Psi_g, \Psi_b$  are defined on the same model:

$$(47) \quad \Psi_r = \begin{pmatrix} \phi_{dr} & \phi_{ur} \\ \widehat{\phi}_{\bar{u}r} \sigma_1 & \widehat{\phi}_{\bar{d}r} \sigma_1 \end{pmatrix} = \begin{pmatrix} \phi_{dr} & \phi_{ur} \\ \widehat{\phi}_{ur} & \widehat{\phi}_{dr} \end{pmatrix}$$

$$(48) \quad \Psi_g = \begin{pmatrix} \phi_{dg} & \phi_{ug} \\ \widehat{\phi}_{\bar{u}g} \sigma_1 & \widehat{\phi}_{\bar{d}g} \sigma_1 \end{pmatrix} = \begin{pmatrix} \phi_{dg} & \phi_{ug} \\ \widehat{\phi}_{ug} & \widehat{\phi}_{dg} \end{pmatrix}$$

$$(49) \quad \Psi_b = \begin{pmatrix} \phi_{db} & \phi_{ub} \\ \widehat{\phi}_{\bar{u}b} \sigma_1 & \widehat{\phi}_{\bar{d}b} \sigma_1 \end{pmatrix} = \begin{pmatrix} \phi_{db} & \phi_{ub} \\ \widehat{\phi}_{ub} & \widehat{\phi}_{db} \end{pmatrix}.$$

These definitions use the matrix representation:

$$(50) \quad L_\mu = \begin{pmatrix} 0 & \gamma_\mu \\ \gamma_\mu & 0 \end{pmatrix}, \quad \mu = 0, 1, 2, 3; \quad L_4 = \begin{pmatrix} 0 & -I_4 \\ I_4 & 0 \end{pmatrix}; \quad L_5 = \begin{pmatrix} 0 & \mathbf{i} \\ \mathbf{i} & 0 \end{pmatrix},$$

The wave contains not only all particles of the first generation, but also all anti-particles of these objects. We let

$$(51) \quad \underline{W}^j = L^\mu W_\mu^j, \quad j = 1, 2, 3; \quad \underline{D} = L^\mu D_\mu; \quad L^0 = L_0; \quad L^j = -L_j$$

for  $j = 1, 2, 3$ . The covariant derivative of electro-weak interaction reads now

$$(52) \quad \underline{D}(\Psi) = \underline{\partial}(\Psi) + \frac{g_1}{2} \underline{B} \underline{P}_0(\Psi) + \frac{g_2}{2} \underline{W}^j \underline{P}_j(\Psi).$$

We use two projectors  $\underline{P}_\pm$  satisfying

$$(53) \quad \underline{P}_\pm(\Psi) = \frac{1}{2}(\Psi \pm \mathbf{i}\Psi L_{21}); \quad \mathbf{i} = L_{0123}$$

Three operators act on the quark sector as on the lepton sector :

$$(54) \quad \underline{P}_1(\Psi) = \underline{P}_+(\Psi) L_{35}$$

$$(55) \quad \underline{P}_2(\Psi) = \underline{P}_+(\Psi) L_{5012}$$

$$(56) \quad \underline{P}_3(\Psi) = \underline{P}_+(\Psi) L_{0132}.$$

The fourth operator acts differently on the lepton wave and on the sector:

$$(57) \quad \underline{P}_0(\Psi) = \begin{pmatrix} P_0(\Psi_l) & P'_0(\Psi_r) \\ P'_0(\Psi_g) & P'_0(\Psi_b) \end{pmatrix}$$

$$(58) \quad P_0(\Psi_l) = \Psi_l \gamma_{21} + P_-(\Psi_l) \mathbf{i} = \Psi_l \gamma_{21} + \frac{1}{2}(\Psi_l \mathbf{i} + \mathbf{i} \Psi_l \gamma_{30})$$

$$(59) \quad P'_0(\Psi_r) = -\frac{1}{3} \Psi_r \gamma_{21} + P_-(\Psi_r) \mathbf{i} = -\frac{1}{3} \Psi_r \gamma_{21} + \frac{1}{2}(\Psi_r \mathbf{i} + \mathbf{i} \Psi_r \gamma_{30})$$

The value  $-1/3$  gives the four correct values of the charges of quarks and antiquarks (see [12] 6.1). To get the gauge group of chromodynamics we need the projectors

$$(60) \quad P^+ = \frac{1}{2}(I_8 + L_{012345}) = \begin{pmatrix} I_4 & 0 \\ 0 & 0 \end{pmatrix}$$

$$P^- = \frac{1}{2}(I_8 - L_{012345}) = \begin{pmatrix} 0 & 0 \\ 0 & I_4 \end{pmatrix}.$$

The operators corresponding to the eight generators of  $SU(3)$  read ( $\Psi_c$  is abbreviated as  $c$ ,  $c = r, g, b$ ):

$$(61) \quad \Gamma_1(\Psi) = \frac{1}{2}(L_4\Psi L_4 + L_{01235}\Psi L_{01235}) = \begin{pmatrix} 0 & g \\ r & 0 \end{pmatrix}$$

$$(62) \quad \Gamma_2(\Psi) = \frac{1}{2}(L_5\Psi L_4 - L_{01234}\Psi L_{01235}) = \begin{pmatrix} 0 & -ig \\ ir & 0 \end{pmatrix}$$

$$(63) \quad \Gamma_3(\Psi) = P^+\Psi P^- - P^-\Psi P^+ = \begin{pmatrix} 0 & r \\ -g & 0 \end{pmatrix}$$

$$(64) \quad \Gamma_4(\Psi) = L_{01253}\Psi P^- = \begin{pmatrix} 0 & b \\ 0 & r \end{pmatrix}; \Gamma_5(\Psi) = L_{01234}\Psi P^- = \begin{pmatrix} 0 & -ib \\ 0 & ir \end{pmatrix}$$

$$(65) \quad \Gamma_6(\Psi) = P^-\Psi L_{01253} = \begin{pmatrix} 0 & 0 \\ b & g \end{pmatrix}; \Gamma_7(\Psi) = -iP^-\Psi L_4 = \begin{pmatrix} 0 & 0 \\ -ib & ig \end{pmatrix}$$

$$(66) \quad \Gamma_8(\Psi) = \frac{1}{\sqrt{3}}(P^-\Psi L_{012345} + L_{012345}\Psi P^-) = \frac{1}{\sqrt{3}} \begin{pmatrix} 0 & r \\ g & -2b \end{pmatrix}.$$

We can extend the covariant derivative of electro-weak interactions (39):

$$(67) \quad \underline{D}(\Psi) = \underline{\partial}(\Psi) + \frac{g_1}{2}\underline{B}P_0(\Psi) + \frac{g_2}{2}\underline{W}^j P_j(\Psi) + \frac{g_3}{2}\underline{G}^k \underline{i}\Gamma_k(\Psi).$$

where  $g_3$  is another constant and  $\underline{G}^k = L^\mu G_\mu^k$  are eight terms called ‘‘gluons’’. Since  $L_4$  commute with any element of  $Cl_{1,3}$  and since  $P_j(i\Psi_{ind}) = iP_j(\Psi_{ind})$  for  $j = 0, 1, 2, 3$  and  $ind = l, r, g, b$  each operator  $\underline{i}\Gamma_k$  commutes with all operators  $\underline{P}_j$ . Using 12 real numbers  $a^0, a^j, j = 1, 2, 3, b^k, k = 1, 2, \dots, 8$ , we let

$$(68) \quad S_1 = \sum_{j=1}^3 a^j \underline{P}_j; \quad S_2 = \sum_{k=1}^8 b^k \underline{i}\Gamma_k$$

and we get, using exponentiation (see [12] 6.2)

$$(69) \quad \exp(a^0 \underline{P}_0 + S_1 + S_2) = \exp(a^0 \underline{P}_0) \exp(S_1) \exp(S_2)$$

The set of these operators is a  $U(1) \times SU(2) \times SU(3)$  Lie group. Only difference with the standard model the structure of this group is not postulated but calculated. The invariance under  $Cl_3^*$  (and particularly the relativistic invariance) of this covariant derivative comes from (9). The gauge invariance reads

$$(70) \quad \Psi' = [\exp(a^0 \underline{P}_0 + S_1 + S_2)](\Psi); \quad \underline{D} = L^\mu \underline{D}_\mu; \quad \underline{D}' = L^\mu \underline{D}'_\mu$$

$$(71) \quad \underline{D}'_\mu \Psi' = \exp(a^0 \underline{P}_0 + S_1 + S_2) \underline{D}_\mu \Psi$$

$$(72) \quad B'_\mu = B_\mu - \frac{2}{g_1} \partial_\mu a^0$$

$$(73) \quad W'^j_\mu P_j = \left[ \exp(S_1) W^j_\mu P_j - \frac{2}{g_2} \partial_\mu [\exp(S_1)] \right] \exp(-S_1)$$

$$(74) \quad \underline{G}'^k_\mu \underline{i}\Gamma_k = \left[ \exp(S_2) \underline{G}^k_\mu \underline{i}\Gamma_k - \frac{2}{g_3} \partial_\mu [\exp(S_2)] \right] \exp(-S_2).$$

The  $SU(3)$  group of chromodynamics, generated by the  $\Gamma_k$  operators projecting the wave on the quark sector acts only on this sector of the wave:

$$(75) \quad P^+ [\exp(b^k \underline{i}\Gamma_k)](\Psi) P^+ = P^+ \Psi P^+ = \begin{pmatrix} \Psi_l & 0 \\ 0 & 0 \end{pmatrix}$$

We get then a  $U(1) \times SU(2) \times SU(3)$  gauge group for a wave including all fermions of the first generation. This group acts on the lepton sector only by its  $U(1) \times SU(2)$  part. The physical translation is: leptons do not strongly interact, they have only electromagnetic and

weak interactions. This is fully satisfied in experiments. The novelty here is that this comes from the structure itself of the quantum wave. Since it is independent on the energy scale, we understand why great unified theories do not work.

The wave equation for the complete wave reads:

$$(76) \quad 0 = (\underline{D}\Psi)L_{012} + \underline{\mathbf{M}}.$$

The mass term

$$(77) \quad \underline{\mathbf{M}} = \begin{pmatrix} m_2\rho_2\chi_b & m_2\rho_2\chi_g \\ m_2\rho_2\chi_r & m_1\rho_1\chi_l \end{pmatrix}$$

contains  $a_j$  defined in (24) to (26) and  $s_j$  in [12](B.168) to (B.182) and

$$(78) \quad \rho_1^2 = a_1a_1^* + a_2a_2^* + a_3a_3^*; \quad \rho_2^2 = \sum_{j=1}^{j=15} s_j s_j^*.$$

Since the only  $U(1) \times SU(2)$  part of the gauge group acts on electron+neutrino the wave equation acts separately in a lepton part and a quark part:

$$(79) \quad 0 = (\underline{D}\Psi^l)L_{012} + m_1\rho_1 \begin{pmatrix} 0 & 0 \\ 0 & \chi_l \end{pmatrix}; \quad \Psi^l = \begin{pmatrix} \Psi_l & 0 \\ 0 & 0 \end{pmatrix}$$

$$(80) \quad 0 = (\underline{D}\Psi^c)L_{012} + m_2\rho_2\chi^c; \quad \chi^c = \begin{pmatrix} \chi_b & \chi_g \\ \chi_r & 0 \end{pmatrix}; \quad \Psi^c = \begin{pmatrix} 0 & \Psi_r \\ \Psi_g & \Psi_b \end{pmatrix}.$$

The  $\chi_c$ ,  $c = r, g, b$  are defined in [12](B.184) to (B.186). The wave equation (79) is equivalent to

$$(81) \quad \mathbf{D}\Psi_l\gamma_{012} + m_1\rho_1\chi_l = 0; \quad \gamma_{012} = \gamma_0\gamma_1\gamma_2$$

which is the equation (28) with  $m_1 = m$ ,  $\rho_1 = \rho$ . This wave equation is equivalent to the invariant form:

$$(82) \quad \tilde{\Psi}_l(\mathbf{D}\Psi_l)\gamma_{012} + m_1\rho_1\tilde{\Psi}_l\chi_l = 0; \quad \tilde{\Psi}_l = \begin{pmatrix} \bar{\phi}_e & \phi_n^\dagger \\ \bar{\phi}_n & \phi_e^\dagger \end{pmatrix}.$$

The double link between the Lagrangian density and the wave equation exists also for the complete wave equation (76): the real part of the invariant equation is the sum of the lepton term and the corresponding term for the quark part of the wave equation. This one is equivalent to the invariant form:

$$(83) \quad 0 = \tilde{\Psi}^c(\underline{D}\Psi^c)L_{012} + m_2\rho_2\tilde{\Psi}^c\chi^c$$

$$(84) \quad \tilde{\Psi}^c = \begin{pmatrix} \tilde{\Psi}_b & \tilde{\Psi}_r \\ \tilde{\Psi}_g & 0 \end{pmatrix}; \quad \chi^c = \begin{pmatrix} \chi_b & \chi_g \\ \chi_r & 0 \end{pmatrix}$$

Like in the lepton case, the real part of the wave equation is simply the equality (see [12] 7.6)

$$(85) \quad \mathcal{L} = 0.$$

This link between the wave equation and the Lagrangian density is very strong on the mathematical point of view, since it comes from an algebraic calculation, similar to take the real part of a complex number. The way going from the Lagrangian density, by the variational calculus and an integration by parts, is very dubious on the physical point of view for propagating waves. This method is nevertheless always available on the mathematical point of view. Similarly to (45) only one of the numeric equations equivalent to (83) is simple, the law of conservation of the total current:

$$(86) \quad \partial_\mu J_t^\mu; \quad J_t = \phi_{dr}\phi_{dr}^\dagger + \phi_{ur}\phi_{ur}^\dagger + \phi_{dg}\phi_{dg}^\dagger + \phi_{ug}\phi_{ug}^\dagger + \phi_{db}\phi_{db}^\dagger + \phi_{ub}\phi_{ub}^\dagger.$$

The wave equation (76) is form invariant under the group  $Cl_3^*$  defined by (8) and (9). The  $\Psi$  wave satisfies:

$$(87) \quad \Psi'(x') = \underline{N}\Psi(x) ; \quad \underline{N} = \begin{pmatrix} N & 0 \\ 0 & N \end{pmatrix} ; \quad N = \begin{pmatrix} M & 0 \\ 0 & \widehat{M} \end{pmatrix} ; \quad m'_1\rho'_1 = m_1\rho_1 ; \quad m'_2\rho'_2 = m_2\rho_2.$$

The wave equation (76) is gauge invariant under the  $U(1) \times SU(2) \times SU(3)$  group of transformations defined by (70) to (74) (see [10] and [12] B.4 and B.5). The lepton part and the quark part of the complete wave may be separated, because the lepton part is insensitive to the  $SU(3)$  part of the gauge group. If the quark part is canceled the wave is reduced to the lepton wave (28).

#### 4. GRAVITATION

Quantum physics has been unable to account for gravitation until now because gravitation necessitates a relativistic formulation. And the relativistic part of quantum physics used in the standard model has suppressed the mass term of the electron to get the electro-weak gauge invariance. Now we have a relativistic wave equation with mass term able to get the electro-weak and strong interactions with gauge transformation. Mass terms present in our wave equations are able to account for both aspects of the mass : inertial mass and gravitational mass.

**4.1. Inertia.** In the precedent section the form invariance was obtained for the  $R$  transformations defined from  $M$  terms that were the same for the whole space-time. Such constant  $M$  terms produce dilations which link an inertial frame to another one. Wave equations of quantum mechanics are implicitly written in such inertial frames. An inertial frame is a frame which has no move of rotation and none acceleration if we compare with very far stars. For instance we consider a frame at the surface of a sphere, with a fixed third axis that is parallel to the axis  $\sigma_3$  of the rotation which is also an axis of the sphere. At the point P the  $\vec{n}$  vector is normal to the sphere. The first axis is supposed orthogonal to the axis of rotation and in the plane  $(\sigma_3, \vec{n})$ . We suppose that  $(\sigma_1, \sigma_2, \sigma_3)$  is an orthonormal direct basis. The move of the frame at P is made of a move of translation in the direction  $\sigma_2$  and a move of rotation with axis  $\sigma_3$ . We name  $R$  the distance of P to the axis of rotation and  $\omega$  the angular velocity. The velocity of the move of translation is  $v = \omega R$ . We let

$$(88) \quad M_1 = e^{\delta\sigma_2} ; \quad M_2 = e^{\omega dx^0 i\sigma_3/2c} ; \quad M = M_2 M_1 ; \quad x' = M x M^\dagger$$

$$(89) \quad \delta = \frac{1}{2} \operatorname{atanh}\left(\frac{\omega R}{c}\right)$$

And we get

$$(90) \quad x'^0 = \cosh(2\delta)x^0 + \sinh(2\delta)x^2 ; \quad x'^3 = x^3$$

$$(91) \quad x'^1 = x^1 + \frac{\omega}{c} [\sinh(2\delta)x^0 + \cosh(2\delta)x^2] dx^0$$

$$(92) \quad x'^2 = [\sinh(2\delta)x^0 + \cosh(2\delta)x^2] - \frac{\omega}{c} x^1 dx^0.$$

This gives

$$(93) \quad \frac{d^2 x'^1}{dt^2} = \omega c \sinh(2\delta) \approx \omega^2 R.$$

which is the centrifugal acceleration.

Any variable  $M = M(x)$  satisfying, in the vicinity of a space-time point  $\underline{x}$  reads

$$M = M_1 M_2 ; M_1 = M(\underline{x})$$

$$(94) \quad M_2 = M_1^{-1} M = 1 + \frac{dx^\mu}{2} (p_\mu + f_\mu \sigma_1 + l_\mu \sigma_2 + a_\mu \sigma_3 + h_\mu i \sigma_1 + g_\mu i \sigma_2 + b_\mu i \sigma_3 + iq_\mu)$$

$$(95) \quad M_2^\dagger = 1 + \frac{dx^\mu}{2} (p_\mu + f_\mu \sigma_1 + l_\mu \sigma_2 + a_\mu \sigma_3 - h_\mu i \sigma_1 - g_\mu i \sigma_2 - b_\mu i \sigma_3 - iq_\mu)$$

where  $a_\mu, b_\mu, \dots, q_\mu$ , are 32 numeric functions of  $x$  and  $dx^\mu$  are the infinitesimal increments of  $x^\mu$  coordinates. This gives, for  $x' = M_2 x M_2^\dagger$ :

$$(96) \quad x'^0 = x^0 + (p_\mu x^0 + f_\mu x^1 + l_\mu x^2 + a_\mu x^3) dx^\mu$$

$$(97) \quad x'^1 = x^1 + (f_\mu x^0 + p_\mu x^1 + b_\mu x^2 - g_\mu x^3) dx^\mu$$

$$(98) \quad x'^2 = x^2 + (l_\mu x^0 - b_\mu x^1 + p_\mu x^2 + h_\mu x^3) dx^\mu$$

$$(99) \quad x'^3 = x^3 + (a_\mu x^0 + g_\mu x^1 - h_\mu x^2 + p_\mu x^3) dx^\mu$$

Christoffel's symbols  $\Gamma_{\beta\gamma}^\alpha$  being usually defined as

$$(100) \quad x'^\alpha = x^\alpha + \Gamma_{\beta\gamma}^\alpha x^\beta dx^\gamma$$

this gives

$$(101) \quad \Gamma_{0\mu}^0 = \Gamma_{1\mu}^1 = \Gamma_{2\mu}^2 = \Gamma_{3\mu}^3 = p_\mu$$

$$(102) \quad \Gamma_{0\mu}^1 = \Gamma_{1\mu}^0 = f_\mu ; \Gamma_{0\mu}^2 = \Gamma_{2\mu}^0 = l_\mu ; \Gamma_{0\mu}^3 = \Gamma_{3\mu}^0 = a_\mu$$

$$(103) \quad \Gamma_{3\mu}^2 = -\Gamma_{2\mu}^3 = h_\mu ; \Gamma_{1\mu}^3 = -\Gamma_{3\mu}^1 = g_\mu ; \Gamma_{2\mu}^1 = -\Gamma_{1\mu}^2 = b_\mu$$

Only 28 amongst the 32 numeric functions are present in the Christoffel's symbols: the four  $q_\mu$  present in (94) are not in the geometry, because the kernel of the group homomorphism  $M \mapsto R$  is the  $U(1)$  group generated by  $i$  [3][7]. Einstein had very early the intuition that something was lacking in quantum physics. Effectively the  $q_\mu$  are lacking, not for the spinors of quantum physics, but for vectors and tensors: the  $q_\mu$  are present in the  $M_2$  term multiplying the spinors of the wave, and are lacking in the transformation of contravariant vectors (100). They are also lacking in the transformation of covariant vectors: we have

$$(104) \quad \nabla = \sigma^\mu \partial_\mu = \overline{M} \sigma^\mu \widehat{M} \partial'_\mu$$

with the same  $\sigma^\mu$ . This gives

$$(105) \quad \nabla' = \sigma^\nu \partial'_\nu = \overline{M}^{-1} \sigma^\nu \widehat{M}^{-1} \partial_\nu = \sigma^\nu (\partial_\nu - dx^\mu \Gamma_{\nu\mu}^\rho \partial_\rho)$$

Therefore we get for covariant vectors the usual

$$(106) \quad \partial'_\nu = \partial_\nu - dx^\mu \Gamma_{\nu\mu}^\rho \partial_\rho.$$

This gives

$$(107) \quad \begin{aligned} \overline{\phi}' \nabla' \widehat{\phi}' &= (\overline{M\phi}) \overline{M}^{-1} \sigma^\mu \widehat{M}^{-1} \partial_\mu (\widehat{M\phi}) \\ &= \overline{\phi} \overline{M} \overline{M}^{-1} \sigma^\mu \widehat{M}^{-1} [(\partial_\mu \widehat{M}) \widehat{\phi} + \widehat{M} (\partial_\mu \widehat{\phi})] \end{aligned}$$

$$(108) \quad \begin{aligned} &= \overline{\phi} \sigma^\mu \widehat{M}^{-1} (\partial_\mu \widehat{M}) \widehat{\phi} + \overline{\phi} \sigma^\mu \partial_\mu \widehat{\phi} \\ &= \overline{\phi} \sigma^\mu [ -(\partial_\mu \widehat{M}^{-1}) \widehat{M} ] \widehat{\phi} + \overline{\phi} \nabla \widehat{\phi} \\ &= \overline{\phi} [\nabla - (\nabla \widehat{M}^{-1}) \widehat{M}] \widehat{\phi}. \end{aligned}$$

$$(109) \quad \overline{\phi}' \nabla' \widehat{\phi}' = \overline{\phi} D \widehat{\phi}$$

where we have let

$$(110) \quad D = \nabla - (\nabla \widehat{M}^{-1}) \widehat{M}$$

$$(111) \quad = \sigma^\mu [\partial_\mu + \frac{1}{2}(p_\mu - f_\mu \sigma_1 - l_\mu \sigma_2 - a_\mu \sigma_3 + h_\mu i \sigma_1 + g_\mu i \sigma_2 + b_\mu i \sigma_3 - iq_\mu)]$$

This introduces 8 space-time vectors that we name ‘‘potentials of inertia’’:

$$(112) \quad p = \sigma^\mu p_\mu = \sigma^\mu \Gamma_{0\mu}^0; \quad f = \sigma^\mu f_\mu = \sigma^\mu \Gamma_{0\mu}^1; \quad l = \sigma^\mu l_\mu = \sigma^\mu \Gamma_{0\mu}^2$$

$$(113) \quad a = \sigma^\mu a_\mu = \sigma^\mu \Gamma_{0\mu}^3; \quad h = \sigma^\mu h_\mu = \sigma^\mu \Gamma_{3\mu}^2; \quad g = \sigma^\mu g_\mu = \sigma^\mu \Gamma_{1\mu}^3$$

$$(114) \quad b = \sigma^\mu b_\mu = \sigma^\mu \Gamma_{2\mu}^1; \quad q = \sigma^\mu q_\mu$$

$$(115) \quad D = \nabla + \frac{1}{2}(p - f\sigma_1 - l\sigma_2 - a\sigma_3 + hi\sigma_1 + gi\sigma_2 + bi\sigma_3 - iq).$$

These eight potentials become under a dilation  $R$  induced by a constant  $M$

$$(116) \quad D = \overline{M} D' \widehat{M}; \quad \nabla = \overline{M} \nabla' \widehat{M}; \quad p = \overline{M} p' \widehat{M}; \quad q = \overline{M} q' \widehat{M}.$$

In space-time algebra we shall need

$$(117) \quad \begin{aligned} \widehat{D} &= \widehat{\nabla} - (\widehat{\nabla} M^{-1}) M \\ &= \widehat{\nabla} + \frac{1}{2}(\widehat{p} + \widehat{f}\sigma_1 + \widehat{l}\sigma_2 + \widehat{a}\sigma_3 + \widehat{h}i\sigma_1 + \widehat{g}i\sigma_2 + \widehat{b}i\sigma_3 + i\widehat{q}) \end{aligned}$$

$$(118) \quad \mathbf{D} = \begin{pmatrix} 0 & D \\ \widehat{D} & 0 \end{pmatrix}; \quad \underline{\mathbf{D}} = \begin{pmatrix} 0 & \mathbf{D} \\ \mathbf{D} & 0 \end{pmatrix}.$$

And the covariant derivative unifying inertia to gauge interactions becomes

$$(119) \quad \underline{D} = \underline{\mathbf{D}} + \frac{g_1}{2} \underline{B} P_0 + \frac{g_2}{2} \underline{W}^j P_j + \frac{g_3}{2} \underline{G}^k \mathbf{i} \Gamma_k$$

Contrarily to all other terms that contains projectors, the term of inertia acts on the whole wave. This universality is a characteristic of inertia.

**4.2. Wave normalization.** The invariance of the Lagrangian under all translations, as with the linear Dirac theory, induces the existence of a conservative impulse-energy tensor, the Tetrode’s tensor :

$$(120) \quad T_\nu^\mu = \partial_\nu \overline{\psi} \frac{\partial \mathcal{L}}{\partial (\partial_\mu \overline{\psi})} + \frac{\partial \mathcal{L}}{\partial (\partial_\mu \psi)} \partial_\nu \psi - \delta_\nu^\mu \mathcal{L}.$$

Since the wave equation is homogeneous, the Lagrangian is null and we get:

$$(121) \quad T_\nu^\mu = \frac{i}{2} (-\overline{\psi} \gamma^\mu \partial_\nu \psi + \partial_\nu \overline{\psi} \gamma^\mu \psi).$$

For a stationary state of an electron with energy  $E$  we have:

$$(122) \quad \psi = e^{iEt/\hbar} \psi(x); \quad \overline{\psi} = e^{-iEt/\hbar} \overline{\psi}(x); \quad \partial_0 \psi = i \frac{E}{\hbar c} \psi; \quad \partial_0 \overline{\psi} = -i \frac{E}{\hbar c} \overline{\psi}.$$

So we get:

$$(123) \quad T_0^0 = \frac{i}{2} (-\overline{\psi} \gamma^0 (i \frac{E}{\hbar c}) \psi - i \frac{E}{\hbar c} \overline{\psi} \gamma^0 \psi) = E \frac{J^0}{\hbar c}.$$

The condition normalizing the wave function of the electron must then be replaced by

$$(124) \quad \iiint \frac{J^0}{\hbar c} dv = 1$$

that is equivalent, for a bound state, to

$$(125) \quad \iiint T_0^0 dv = E.$$

This means that the quantum precept of the normalization of the wave is not arbitrary but is a consequence of the principle of equivalence between inertial and gravitational mass-energy, basis of the General Relativity.

**4.3. Gravitation.** The global energy  $E$  of the electron is the temporal component of a space-time vector, the energy-momentum vector. Since the integration has been made in a frame where this momentum is null, this vector reads  $(E, 0, 0, 0, )$  but it will be seen by other observers moving as  $(p_0, p_1, p_2, p_3)$ . General relativity considers all particles of the universe as giving each such an energy-momentum space-time vector, and if there is no pressure the density of this fluid of particles  $(d_0, d_1, d_2, d_3)$  constitutes the material contribution to the symmetric tensor of energy  $T_\mu^\nu = d_\mu d^\nu$ . Einstein has linked this material tensor to the curvature of the space-time manifold:

$$(126) \quad \frac{1}{\chi} [R_\mu^\rho - \frac{1}{2} \delta_\mu^\rho (R - 2\Lambda)] = T_\mu^\rho$$

where  $\Lambda$  is the cosmological constant and  $\chi$  is the constant of gravitation. We have placed this constant on the left side, then (126) is invariant under  $Cl_3^*$ . The density of mass per unit volume  $\mu_0$  used in the Newtonian law of gravitation

$$(127) \quad \Delta U = -4\pi G \mu_0 ; \quad \chi = 8\pi \frac{G}{c^4}$$

gives, in the case of matter without pressure:

$$(128) \quad T_\mu^\rho = \mu_0 c^2 u_\mu u^\rho ; \quad u^\mu = \frac{dx^\mu}{ds} ; \quad ds^2 = g_{\mu\nu} dx^\mu dx^\nu .$$

The standard model of quantum physics is therefore compatible with general relativity.

## 5. CONCLUDING REMARKS

The use of these three Clifford algebras presents many advantages:

**We reunite** the frame of classical physics, which was the space-time and vectors or tensors built on space-time, with the frame of quantum mechanics, since all interactions are described with real Clifford algebras.

Quantum waves used here are not so different from other waves of classical physics. They are well-defined functions of space and time with value in algebras that are also linear spaces on the real field.

Since the geometric algebra of the physical space is isomorphic to the complex linear space of  $2 \times 2$  matrices (Pauli algebra) linear spaces of complex functions have invaded quantum physics. Nevertheless this isomorphism is not an isomorphism of complex algebraic structures, but an isomorphism of algebras on the real field. Then complex structures are not fundamental, the true frame of quantum physics is the geometric algebra. Hermiticity and unitarity are not fundamental: the true conjugation is the reversion, in each algebra (see (7), (28), (83)). Reverse and adjoint are identical in  $Cl_3$  and only there.

Waves, for fermions and antifermions but also for systems of particles [4] are well defined functions of space and time with value in a real Clifford algebra: now we are not disturbed by the difference between space coordinates and time coordinate coming from the non-relativistic quantum wave of systems.

**We integrate** in a classical frame all novelties coming from quantum mechanics:

The fermionic wave is made of spinors, objects that are part of the geometric algebra, with the peculiarity that they turn on a half angle in a spacial rotation.



The charge conjugation is a pure quantum transformation that changes only the differential part of the quantum wave (see [12] 3.4).

The group of form invariance accounts for the spin 1/2 and it is then necessarily greater than the Lorentz group. The fundamental group of invariance of all physical interaction is  $Cl_3^*$ , group of all invertible elements in the geometric algebra. Physics, computer science and engineering are reunited !

Because there are two inequivalent homomorphisms of  $Cl_3^*$  in the group of Lorentz dilation there are left and right waves. They turn differently in a Lorentz dilation: the physical space is oriented, and this is experimentally well known in weak interactions.

The invariance group is compatible with an oriented space and with an oriented time.

The integration of two quarks in each generation, with three color states each, is made by adding two and only two dimensions to the usual space. This is compatible with the group of invariance  $Cl_3^*$  in a way that separates automatically the three usual dimensions from the two supplementary dimensions (see[12] 7.4.1). This enlarged geometric frame allows a wave equation for all particles and antiparticles of each generation. The wave equation is gauge invariant under a gauge group that is exactly the  $U(1) \times SU(2) \times SU(3)$  group of the standard model of quantum physics. This group is not postulated but is a consequence of the geometric structure of the enlarged space-time. The mass term of the wave equation is compatible with the form invariance under  $Cl_3^*$  that generalizes the relativistic invariance, and it is compatible with the gauge invariance: no need of symmetry breaking.

### **We justify:**

Three and only three generations and a fourth neutrino [8].

The existence of the Planck constant (see [12] 3.3).

The normalization of the wave and the existence of a density of probability.

The insensitivity of leptons to strong interactions.

The strict conservation of the baryonic number, linked to this insensitivity.

The existence of a Lagrangian formalism.

**We comfort** the standard model by diminishing the too numerous free parameters:

Only two proper masses in each generation.

One number gives the four values of charges of quarks and antiquarks.

The value of the electric charge is determined by the existence of magnetic monopoles (see [12] 8.3.2).

### **We integrate gravitation:**

Inertial frames are also frames where a double link exists between the wave equation and the Lagrangian density.

Non-inertial frames are frames coming from the use of variable terms in  $Cl_3^*$ .

The normalization of the wave and the existence of a density of probability are consequences of the principle of equivalence between inertial and gravitational mass-energy.

After integration on all physical space, the impulse-energy tensor of the quantum wave becomes the symmetric tensor of General Relativity.

## REFERENCES

- [1] W. E. Baylis. *Clifford (Geometric) Algebras*, chapter The Paravector Model of Spacetime, Birkhauser, Boston, 1996, pp. 237–296.
- [2] C. Daviau. *Interprétation cinématique de l'onde de l'électron*, Ann. Fond. L. de Broglie, **30**(3-4), 2005, pp. 409–428.
- [3] C. Daviau. *L'espace-temps double*, JePublie, Pouillé-les-coteaux, 2011.
- [4] C. Daviau.  *$Cl_3^*$  invariance of the Dirac equation and of electromagnetism*, Adv. Appl. Clifford Algebras, **22**(3):611–623.
- [5] C. Daviau. *Double Space-Time and more*, JePublie, Pouillé-les-coteaux, 2012.
- [6] C. Daviau. *Nonlinear Dirac Equation, Magnetic Monopoles and Double Space-Time*, CISP, Cambridge UK, 2012.
- [7] C. Daviau. *Invariant quantum wave equations and double space-time*, Adv. in Imaging and Electron Physics, **179**, chapter 1:1–137, 2013.
- [8] C. Daviau and J. Bertrand. *A lepton Dirac equation with additional mass term and a wave equation for a fourth neutrino*, Ann. Fond. Louis de Broglie, **38**, 2013, pp. 57–82.
- [9] C. Daviau and J. Bertrand. *Relativistic gauge invariant wave equation of the electron-neutrino*, Journal of Modern Physics, **5**, pp.1001–1022, <http://dx.doi.org/10.4236/jmp.2014.511102>, 2014.
- [10] C. Daviau and J. Bertrand. *A wave equation including leptons and quarks for the standard model of quantum physics in Clifford algebra* JMP, **5**, pp. 2149–2173, <http://dx.doi.org/10.4236/jmp.2014.518210>, 2014.
- [11] C. Daviau and J. Bertrand. *New Insights in the Standard Model of Quantum Physics in Clifford Algebra*, Je Publie, Pouillé-les-coteaux, 2014 and <http://hal.archives-ouvertes.fr/hal-00907848>.
- [12] C. Daviau and J. Bertrand. *Additional Insights in the Standard Model of Quantum Physics in Clifford Algebra*, World Scientific Publishing, New-York, Singapour, 2015, accepted for publication
- [13] M.A. Naïmark. *Les représentations linéaires du groupe de Lorentz*, Dunod, Paris, 1962.
- [14] S. Weinberg. *A model of leptons*, Phys. Rev. Lett, **19**, pp. 1264–1266, 1967.



# THE $E_8$ GEOMETRY FROM A CLIFFORD PERSPECTIVE

**Pierre-Philippe Dechant**

Departments of Mathematics and Biology  
York Centre for Complex Systems Analysis  
University of York, Heslington YO10 5GE, United Kingdom  
ppd22@cantab.net [presenter, corresponding]

ABSTRACT. At the last AGACSE in 2012, I presented a new link between the geometries of three and four dimensions; in particular, I have shown that any symmetry group in three dimensions induces a corresponding symmetry group in four dimensions, via a new Clifford spinor construction. This connection had been overlooked for centuries (usually one assumes the larger groups are more fundamental) but the new construction derives all the exceptional phenomena in 4D –  $D_4$ ,  $F_4$  and  $H_4$  – via induction from the 3D symmetry groups of the Platonic solids  $A_3$ ,  $B_3$  and  $H_3$ . This spinor construction also explains the unusual 4D automorphism groups. The 4D groups in fact do not contain anything that is not already present in the 3D groups they are induced from.

David Hestenes invited me to stay with him in Phoenix for half a year, and in La Rochelle set me the challenge of deriving the exceptional group  $E_8$ , the holy grail of mathematics and physics, in Geometric Algebra too – he even gave me the  $E_8$  T-shirt that he himself got from Garrett Lisi.

I am pleased to report that I have now solved his challenge. I have found a construction that constructs the  $E_8$  root system in analogy to the above construction going from 3D to 4D. The previous construction worked along the following lines: each 3D root system – which generates the corresponding reflection symmetry group via the reflections in the hyperplanes orthogonal to the root vectors – allows one to form a group of spinors by multiplying together even numbers of the reflection generating root vectors in the Clifford algebra; these spinors have four components (the usual 1 scalar and 3 bivector components) and one can endow these with a 4D Euclidean metric. One can then show that the resulting set of spinors reinterpreted as 4D vectors satisfies the axioms for a 4D root system, thereby generating a symmetry group in 4D.

For instance, starting with the root system  $H_3$  which generates the symmetries of the icosahedron, one generates a group  $2I$  of 120 spinors via multiplication in the Clifford algebra. These are precisely the 120 root vectors of the 4D root system  $H_4$  once reinterpreted using the 4D Euclidean metric.  $H_4$  is exceptional and the largest non-crystallographic Coxeter group; it also has the exceptional automorphism group  $2I \times 2I$ . However, this is trivial to see in terms of the spinor group  $2I$ , as it must be trivially closed under left and right multiplication via group closure.

Taking the full set of pinors (i.e. not restricting to even products of root vectors) of the icosahedron, one generates a group of 240 pinors with 8 components, as befits a group of multivectors in 3D. I have now been able to show that these 240 pinors are precisely the 240 roots of  $E_8$  via a new inner product, thereby generating the exceptional group  $E_8$ .

It is extraordinary that the exceptional geometry  $E_8$  has been hiding in the shadows of icosahedral geometry for millennia, without anyone noticing. As with the 4D induction construction, this discovery was only possible in Clifford algebra – there is much prejudice against the usefulness of Clifford algebras (since they have matrix representations) and usually matrix methods are equivalent if less insightful – but the 4D and 8D induction constructions are to my knowledge the only results that *require* Clifford algebra and were completely invisible to standard matrix methods.

I will also talk about a GA treatment of the  $E_8$  geometry more generally, in particular the geometry of the Coxeter plane and partial Dynkin diagram foldings. I will also review the McKay correspondence and trinitities; icosahedral symmetry and  $E_8$  are indirectly related through these, but the direct connection via Clifford methods had again so far been overlooked.

## 1. INTRODUCTION

Lie groups are a central subject of 20th century mathematics as well as physics. In particular, the largest exceptional Lie group  $E_8$  is central to String Theory and Grand Unified Theories and

is thus arguably the single most important symmetry group in modern theoretical physics. Lie groups are continuous groups but they are closely related to their corresponding Lie algebras whose interesting part in turn is described by a root system: a collection of reflection generating vectors called roots, which generate a reflection symmetry group (called a crystallographic Coxeter group or Weyl group). It is easy to move between those four related concepts and we will usually not make a distinction – with the exception of non-crystallographic root systems such as  $H_3$  (which generates icosahedral symmetry) and its 4D analogue  $H_4$ , since their non-crystallographic nature means that there is no associated Lie algebra. The  $E_8$  root system is thus commonly thought of as an exceptional (i.e. there are no corresponding symmetry groups in arbitrary dimensions) phenomenon of eight dimensional geometry, very far removed from our usual experience with 3D geometry. It is the purpose of this article to show that the eight dimensions of 3D Clifford algebra actually allow  $E_8$  to be unveiled as a 3D geometric phenomenon in disguise; likewise all 4D exceptional root systems arise within 3D geometry. This opens up a revolutionary way of viewing exceptional higher-dimensional phenomena in terms of 3D spinorial geometry. The non-exceptional Lie groups have been realised in Geometric Algebra by David Hestenes et al [10] as spin groups, and Lie algebras as bivector algebras; here we offer a Clifford geometric construction of all the exceptional phenomena via their root systems – of course,  $H_4$  does not even have an associated Lie algebra and Lie group.

This paper is structured in the following way. After some preliminary definitions and background in Section 2, we show in general that any 3D symmetry group induces a 4D symmetry group, via their root systems (Section 3). In particular, the Platonic root systems ( $A_3, B_3, H_3$ ) induce all the exceptional 4D root systems ( $D_4, F_4, H_4$ ) in terms of 3D spinors which also explains their unusual symmetry groups. We concretely explain the case of icosahedral symmetry  $H_3$  inducing the exceptional largest non-crystallographic Coxeter group  $H_4$  from a spinor group (the binary icosahedral group  $2I$ ) that describes the 60 icosahedral rotations in terms of 120 spinors doubly covering the rotations. The above collection of Platonic root systems ( $A_3, B_3, H_3$ ) in fact forms a trinity that is related to the trinity of exceptional Lie groups ( $E_6, E_7, E_8$ ) via various intermediate trinities and also via my new spinor construction in combination with the McKay correspondence. This is the first hint that the icosahedron may be indirectly related with  $E_8$ . Section 4 makes a completely new, direct connection between them by concretely constructing the 240 roots of the  $E_8$  root system from the 240 spinors that doubly cover the 120 icosahedral reflections and rotations in the Clifford algebra of 3D. Thus **all** the exceptional root systems can in fact be seen as induced from the polyhedral symmetries and the Clifford algebra of 3D, which offers a completely new way of better understanding these in terms of spinorial geometry with potential for a wide range of profound consequences. Section 5 discusses  $H_4$  as a rotational subgroup of  $E_8$ , and thus affords a second way of viewing  $H_4$  more naturally as a group of rotations rather than reflections (as is the standard view in the Coxeter picture). This construction of  $H_4$  from  $E_8$  is in fact a partial folding of the  $E_8$  diagram, whilst a complete folding leads to the Coxeter plane. We therefore finish by discussing the geometry of the Coxeter plane of various root systems, notably  $E_8$ , and point out which advantages a Clifford algebraic view has to offer over the naive standard view which involves complexifying the real geometry: in Clifford algebra the complex eigenvalues in fact arise naturally as rotations in mutually orthogonal eigenplanes of the Coxeter element (Section 6) with the bivectors of the planes providing the relevant complex structures; we conclude in Section 7.

## 2. BACKGROUND

In this section, we introduce Coxeter (reflection) groups and their generating root systems:

**Definition 2.1** (Root system). *A root system is a collection  $\Phi$  of non-zero (root) vectors  $\alpha$  spanning an  $n$ -dimensional Euclidean vector space  $V$  endowed with a positive definite bilinear form, which satisfies the two axioms:*

- (1)  $\Phi$  only contains a root  $\alpha$  and its negative, but no other scalar multiples:  $\Phi \cap \mathbb{R}\alpha = \{-\alpha, \alpha\} \forall \alpha \in \Phi$ .
- (2)  $\Phi$  is invariant under all reflections corresponding to root vectors in  $\Phi$ :  $s_\alpha\Phi = \Phi \forall \alpha \in \Phi$ . The reflection  $s_\alpha$  in the hyperplane with normal  $\alpha$  is given by

$$s_\alpha : \lambda \rightarrow s_\alpha(\lambda) = \lambda - 2 \frac{(\lambda|\alpha)}{(\alpha|\alpha)} \alpha,$$

where  $(\cdot|\cdot)$  denotes the inner product on  $V$ .

For a crystallographic root system, a subset  $\Delta$  of  $\Phi$ , called *simple roots*  $\alpha_1, \dots, \alpha_n$ , is sufficient to express every element of  $\Phi$  via  $\mathbb{Z}$ -linear combinations with coefficients of the same sign.  $\Phi$  is therefore completely characterised by this basis of simple roots. In the case of the non-crystallographic root systems  $H_2$ ,  $H_3$  and  $H_4$ , the same holds for the extended integer ring  $\mathbb{Z}[\tau] = \{a + \tau b | a, b \in \mathbb{Z}\}$ , where  $\tau$  is the golden ratio  $\tau = \frac{1}{2}(1 + \sqrt{5}) = 2 \cos \frac{\pi}{5}$ , and  $\sigma$  is its Galois conjugate  $\sigma = \frac{1}{2}(1 - \sqrt{5})$  (the two solutions to the quadratic equation  $x^2 = x + 1$ ). For the crystallographic root systems, the classification in terms of Dynkin diagrams essentially follows the one familiar from Lie groups and Lie algebras, as their Weyl groups are the crystallographic Coxeter groups. A mild generalisation to so-called Coxeter-Dynkin diagrams is necessary for the non-crystallographic root systems:

**Definition 2.2** (Coxeter-Dynkin diagram and Cartan matrix). *A graphical representation of the geometric content of a root system is given by Coxeter-Dynkin diagrams, where nodes correspond to simple roots, orthogonal roots are not connected, roots at  $\frac{\pi}{3}$  have a simple link, and other angles  $\frac{\pi}{m}$  have a link with a label  $m$ . The Cartan matrix of a set of simple roots  $\alpha_i \in \Delta$  is defined as the matrix  $A_{ij} = 2(\alpha_i|\alpha_j)/(\alpha_j|\alpha_j)$ .*

For instance, the root system of the icosahedral group  $H_3$  has one link labelled by 5 (via the above relation  $\tau = 2 \cos \frac{\pi}{5}$ ), as does its four-dimensional analogue  $H_4$ .

The reflections in the second axiom of the root system generate a reflection group. A Coxeter group is a mathematical abstraction of the concept of a reflection group via involutive generators (i.e. they square to the identity, which captures the idea of a reflection), subject to mixed relations that represent  $m$ -fold rotations (since two successive reflections generate a rotation in the plane defined by the two roots).

**Definition 2.3** (Coxeter group). *A Coxeter group is a group generated by a set of involutive generators  $s_i, s_j \in S$  subject to relations of the form  $(s_i s_j)^{m_{ij}} = 1$  with  $m_{ij} = m_{ji} \geq 2$  for  $i \neq j$ .*

The finite Coxeter groups have a geometric representation where the involutions are realised as reflections at hyperplanes through the origin in a Euclidean vector space  $V$ , i.e. they are essentially just the classical reflection groups. In particular, then the abstract generator  $s_i$  corresponds to the simple reflection  $s_i : \lambda \rightarrow s_i(\lambda) = \lambda - 2 \frac{(\lambda|\alpha_i)}{(\alpha_i|\alpha_i)} \alpha_i$  in the hyperplane perpendicular to the simple root  $\alpha_i$ . The action of the Coxeter group is to permute these root vectors, and its structure is thus encoded in the collection  $\Phi \in V$  of all such roots, which in turn form a root system.

We employ a Clifford algebra framework, which via the geometric product affords a uniquely simple prescription for performing reflections  $-\alpha\lambda\alpha$  (assuming unit normalisation) and thus any orthogonal transformation as products of reflections via the Cartan-Diedonné theorem, in spaces of any dimension and signature. For any root system, the quadratic form mentioned in the definition can always be used to enlarge the  $n$ -dimensional vector space  $V$  to the corresponding  $2^n$ -dimensional Clifford algebra. The Clifford algebra is therefore a very natural object to consider in this context, as its unified structure simplifies many problems both conceptually and computationally, rather than applying the linear structure of the space and the inner product separately. In particular, it provides (s)pinor double covers of the (special) orthogonal transformations, as well as geometric quantities that serve as unit imaginaries.

### 3. THE GENERAL SPINOR INDUCTION CONSTRUCTION: $H_4$ AS A GROUP OF ROTATIONS RATHER THAN REFLECTIONS (INDUCED FROM $H_3$ ), TRINITIES AND MCKAY CORRESPONDENCE

In this section we prove that any 3D root system yields a 4D root system via the spinor group obtained by multiplying root vectors in the Clifford algebra [7].

**Proposition 3.1** ( *$O(4)$ -structure of spinors*). *The space of  $Cl(3)$ -spinors  $R = a_0 + a_1e_2e_3 + a_2e_3e_1 + a_3e_1e_2$  can be endowed with a 4D Euclidean norm  $|R|^2 = R\tilde{R} = a_0^2 + a_1^2 + a_2^2 + a_3^2$  induced by the inner product  $(R_1, R_2) = \frac{1}{2}(R_1\tilde{R}_2 + R_2\tilde{R}_1)$  between two spinors  $R_1$  and  $R_2$ .*

This allows one to reinterpret the group of 3D spinors generated from a 3D root system as a set of 4D vectors, which in fact can be shown to satisfy the axioms of a root system as given in Definition 2.1.

**Theorem 3.2** (Induction Theorem). *Any 3D root system gives rise to a spinor group  $G$  which induces a root system in 4D.*

*Proof.* Check the two axioms for the root system  $\Phi$  consisting of the set of 4D vectors given by the 3D spinor group:

- (1) By construction,  $\Phi$  contains the negative of a root  $R$  since spinors provide a double cover of rotations, i.e. if  $R$  is in a spinor group  $G$ , then so is  $-R$ , but no other scalar multiples (normalisation to unity).
- (2)  $\Phi$  is invariant under all reflections with respect to the inner product  $(R_1, R_2)$  in Proposition 3.1 since  $R'_2 = R_2 - 2(R_1, R_2)/(R_1, R_1)R_1 = -R_1\tilde{R}_2R_1 \in G$  for  $R_1, R_2 \in G$  by the closure property of the group  $G$  (in particular  $-R$  and  $\tilde{R}$  are in  $G$  if  $R$  is).

□

Since the number of irreducible 3D root systems is limited to  $(A_3, B_3, H_3)$ , this yields a definite list of induced root systems in 4D – this turns out to be  $(D_4, F_4, H_4)$ , which are exactly the exceptional root systems in 4D. In fact both sets of three are trinities: named after Arnold's observation that many related objects in mathematics form sets of three, beginning with the trinity  $(\mathbb{R}, \mathbb{C}, \mathbb{H})$  [1, 2], and extending to projective spaces, Lie algebras, spheres, Hopf fibrations etc. Arnold's original link between these two trinities  $(A_3, B_3, H_3)$  and  $(D_4, F_4, H_4)$  that arise here is extremely convoluted, and our construction presents a novel direct link between the two.

These root systems are intimately linked to the Platonic solids [5] – there are 5 in three dimensions and 6 in four dimensions:  $A_3$  describes the reflection symmetries of the tetrahedron,  $B_3$  those of the cube and octahedron (which are dual under the exchange of faces and vertices), and  $H_3$  describes the symmetries of the dual pair icosahedron and dodecahedron (the rotational subgroup is denoted by  $I = A_5$ ).

Likewise, the 4D Coxeter groups describe the symmetries of the 4D Platonic solids, but this time the connection is more immediate – the root systems are actually Platonic solids themselves:  $D_4$  is the 24-cell (self-dual), an analogue of the tetrahedron, which is also related to the  $F_4$  root system, and the  $H_4$  root system is the Platonic solid the 600-cell. Its dual, the 120-cell of course has the same symmetry. The root system  $A_1^3$  generates the root system  $A_1^4$ , which constitutes the vertices of the Platonic solid 16-cell, and its dual is the 8-cell. There is thus an abundance of root systems in 4D that are related to the Platonic solids, and in fact the only one not equal or dual to a root system is the 5-cell with symmetry group  $A_4$  – which of course could not be a root system, as it has an odd number (5) of vertices. This abundance of root systems in 4D can in some sense be thought of as due to the accidentalness of our spinor construction. In particular, the induced root systems are precisely the exceptional (i.e. they

do not have counterparts in other dimensions) root systems in 4D:  $D_4$  has the triality symmetry (permutation symmetry of the three legs in the diagram) that is exceptional in 4D and is of great importance in string theory, showing the equivalence of the Ramond-Neveu-Schwarz and the Green-Schwarz strings.  $F_4$  is the only  $F$ -type root system, and  $H_4$  is the largest non-crystallographic root system. In contrast, in arbitrary dimensions there are only  $A_n$  ( $n$ -simplex), and  $B_n$  ( $n$ -hypercube and  $n$ -hyperoctahedron).

Not only is there an abundance of root systems related to the Platonic solids as well as their exceptional nature, but they also have unusual automorphism groups, in that the order of the groups goes as the square of the number of roots. This is also explained via the above spinor construction via the following result:

**Theorem 3.3** (Spinorial symmetries). *A root system induced through the Clifford spinor construction via a binary polyhedral spinor group  $G$  has an automorphism group that trivially contains two factors of the respective spinor group  $G$  acting from the left and from the right.*

This systematises many case-by-case observations on the structure of the automorphism groups [12, 13], and shows that all of the 4D geometry is already contained in 3D [4]. For instance, the automorphism group of the  $H_4$  root system is  $2I \times 2I$  – in the spinor picture, it is not surprising that  $2I$  yields both the root system and the two factors in the automorphism group. We therefore consider the example of the induction  $H_3 \rightarrow H_4$  in more detail.

We construct the spinor group generated by the simple reflections of  $H_3$ . The simple roots are taken as  $\alpha_1 = e_2$ ,  $\alpha_2 = -\frac{1}{2}((\tau - 1)e_1 + e_2 + \tau e_3)$ , and  $\alpha_3 = e_3$ . Under free multiplication, these generate a group with 240 spinors, and the even subgroup consists of 120 spinors, for instance of the form  $\alpha_1\alpha_2 = -\frac{1}{2}(1 - (\tau - 1)e_1e_2 + \tau e_2e_3)$  and  $\alpha_2\alpha_3 = -\frac{1}{2}(\tau - (\tau - 1)e_3e_1 + e_2e_3)$ . These are the double covers of  $I = A_5$  and  $H_3 = A_5 \times \mathbb{Z}_2$ , respectively. The spinors have four components  $(1, e_1e_2, e_2e_3, e_3e_1)$ ; by taking the components of these 120 spinors as a set of vectors in 4D, one obtains the 120 roots in the  $H_4$  root system. This is very surprising from a Coxeter perspective, as one usually thinks of  $H_3$  as a subgroup of  $H_4$ , and therefore of  $H_4$  as more ‘fundamental’; however, one now sees that  $H_4$  does not in fact contain any structure that is not already contained in  $H_3$ , and can therefore think of  $H_3$  as more fundamental [4].

From a Clifford perspective it is not surprising to find this group of 120 spinors, which is the binary icosahedral group  $2I$ , since Clifford algebra provides a simple construction of the Spin groups; however, this is groundbreaking from the conventional Coxeter and Lie group perspective. This spinor group  $2I$  has 120 elements and 9 conjugacy classes.  $I$  has five conjugacy classes and it being of order 60 implies that it has five irreducible representations of dimensions 1, 3,  $\bar{3}$ , 4 and 5 (since the sum of the dimensions squared gives the order of the group  $\sum d_i^2 = |G|$ ). The nine conjugacy classes of the binary icosahedral group  $2I$  of order 120 mean that this acquires a further four irreducible spinorial representations  $2_s, 2'_s, 4_s$  and  $6_s$ . It is worth pointing out that it is convenient to have all these 4 different types of polyhedral groups in a unified framework within the Clifford algebra, rather than using  $SO(3)$  matrices for the rotations and then having to somehow move to  $SU(2)$  matrices for the binary groups.

This binary icosahedral group has a curious connection with the affine Lie algebra  $E_8^+$  (and likewise for the other binary polyhedral groups and the affine Lie algebras of  $ADE$ -type) via the so-called McKay correspondence [14], which is twofold: We can define a graph by assigning a node to each of the nine irreducible representation of the binary icosahedral group with the following rule for connecting nodes: each node corresponding to a certain irreducible representation is connected to the nodes corresponding to those irreducible representations that are contained in its tensor product with the irrep  $2_s$ . For instance, tensoring the trivial representation 1 with  $2_s$  trivially gives  $2_s$  and thus the only link 1 has is with  $2_s$ ;  $2_s \otimes 2_s = 1 + 3$ , such that  $2_s$  is connected to 1 and 3, etc. The graph that is built up in this way is precisely the Dynkin diagram of affine  $E_8$ , as shown in Figure 1. The second connection is the following observation: the Coxeter element is the product of all the simple reflections  $\alpha_1 \dots \alpha_8$  and its



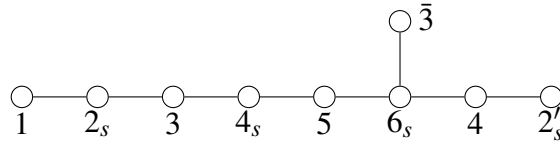


FIGURE 1. The graph depicting the tensor product structure of the binary icosahedral group  $2I$  is the same as the Dynkin diagram of affine  $E_8$ .

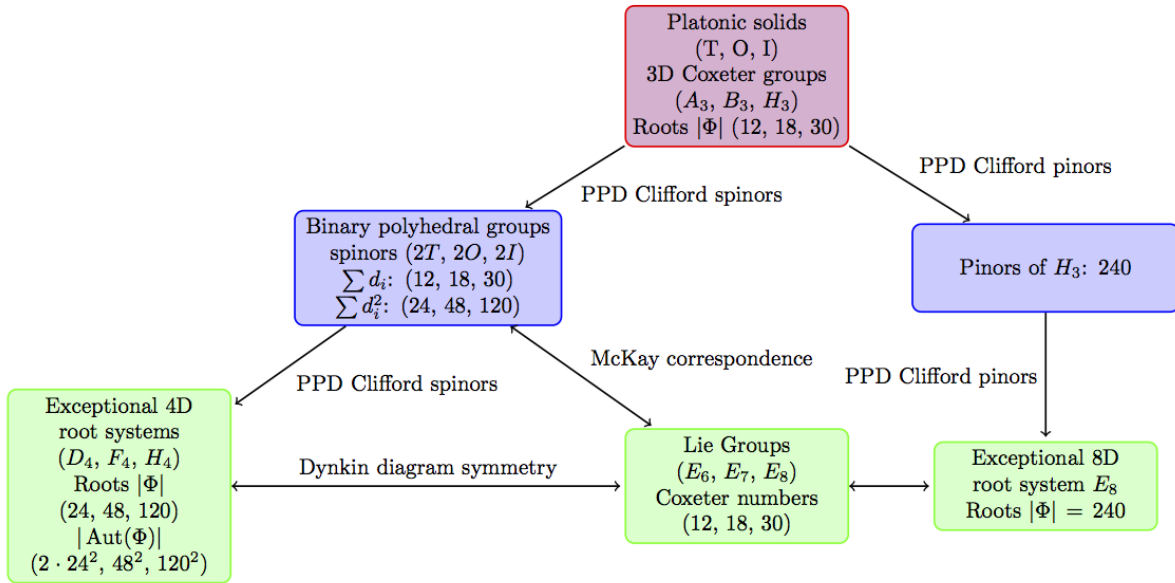


FIGURE 2. Web of connections putting the original McKay correspondence and trinitities into a much wider context. The connection between the sum of the dimensions of the irreducible representations  $d_i$  of the binary polyhedral groups and the Coxeter number of the Lie algebras actually goes all the way back to the number of roots in the 3D root systems  $(12, 18, 30)$  – these then induce the binary polyhedral groups (linked to McKay) and the 4D root systems via the Clifford spinor construction. The new pinor construction links  $H_3$  directly with  $E_8$  explaining the latter entirely within 3D geometry.

order, the Coxeter number  $h$ , is 30 for  $E_8$ . This also happens to be the sum of the dimensions of the irreducible representations of  $2I$ ,  $\sum d_i$ . This extends to all other cases of polyhedral groups and  $ADE$ -type affine Lie algebras (c.f. Figure 2).

The connection between  $(A_3, B_3, H_3)$  and  $(E_6, E_7, E_8)$  via Clifford spinors does not seem to be known. In particular, we note that  $(12, 18, 30)$  is exactly the number of roots  $\Phi$  in the 3D root systems  $(A_3, B_3, H_3)$ , which feeds through to the binary polyhedral groups and via the McKay correspondence all the way to the affine Lie algebras. Our construction therefore makes deep connections between trinitities, and puts the McKay correspondence into a wider framework, as shown in Figure 2. It is also striking that the affine Lie algebra and the 4D root system trinitities have identical Dynkin diagram symmetries:  $D_4$  and  $E_6^+$  have triality  $S_3$ ,  $F_4$  and  $E_7^+$  have an  $S_2$  symmetry and  $H_4$  and  $E_8^+$  only have  $S_1$ , but are intimately related as explained in Section 5. There is thus an indirect connection between the icosahedron and the exceptional  $E_8$ . In the next section we will show a new, explicit direct connection within the Clifford algebra of 3D by identifying the 240 roots of  $E_8$  with the 240 pinors of icosahedral symmetry (right of Figure 2).

4. THE BIRTH OF  $E_8$  OUT OF THE SPINORS OF THE ICOSAHEDRON

Previously, we discussed the construction of the 120 elements of the binary icosahedral group, which can be reinterpreted as the 120 roots of  $H_4$ . We list these here in terms of the 4D basis

$$(\pm 1, 0, 0, 0) \text{ (8 permutations)}$$

$$\frac{1}{2}(\pm 1, \pm 1, \pm 1, \pm 1) \text{ (16 permutations)}$$

$$\frac{1}{2}(0, \pm 1, \pm \sigma, \pm \tau) \text{ (96 even permutations) .}$$

A convenient set of simple roots for  $H_4$  is given by  $a_1 = \frac{1}{2}(-\sigma, -\tau, 0, -1)$ ,  $a_2 = \frac{1}{2}(0, -\sigma, -\tau, 1)$ ,  $a_3 = \frac{1}{2}(0, 1, -\sigma, -\tau)$  and  $a_4 = \frac{1}{2}(0, -1, -\sigma, \tau)$ .

Since the  $H_3$  root system contains three orthogonal roots, e.g.  $(1, 0, 0)$ ,  $(0, 1, 0)$  and  $(0, 0, 1)$ , the pinor group generated under free Clifford multiplication contains the inversion, given by  $\pm e_1 e_2 e_3 = \pm I$ . So when the 120 even products of root vectors stay in the spinor even 4D subalgebra consisting of scalar and bivector parts, then the inversion creates a second copy of this of 120 odd products in the vector and pseudoscalar 4D subalgebra. So what one gets for the 240 pinors double covering the group  $H_3$  of order 120, is a copy of  $H_4$  with 120 roots and another copy multiplied by  $I$ , i.e. in the 8D Clifford algebra of 3D space – which is an 8D vector space – one gets 240 objects, as one would expect for a construction of  $E_8$ .

As an aside, in terms of a quaternionic description of the even subalgebra, the  $H_3$  root system consists precisely of the pure quaternions, i.e. those without a real part, and the full group can be generated from those under quaternion multiplication. This is very poorly understood in the literature, and just hinges on the above description in terms of spinors together with the fact that the inversion  $\pm I$  is contained in the full  $H_3$  group, as one can then trivially Hodge dualise root vectors to bivectors, which are pure quaternions. We have explained this in previous work [4]. For instance, the statement is not true when the inversion is not contained in the group, as is the case for  $A_3$ . However, the spinorial induction construction still works for this, yielding  $D_4$ . Moreover, the ‘quaternionic generators’ generating the 4D groups via quaternion multiplication are just seen to be the even products of 3D simple roots  $\alpha_1 \alpha_2$  and  $\alpha_2 \alpha_3$  so that the 4D group manifestly does not contain anything that was not already contained in the 3D group. We therefore believe the spinor induction point of view of going from 3D to 4D is more fundamental than the pure quaternion approach identifying the 3D group as a subgroup of the 4D group.

For the set of 240 pinors we now define a ‘reduced inner product’: we keep the spinor copy of  $H_4$  and multiply the copy  $IH_4$  by  $\tau I$ , then take inner products taking into account the recursion relation  $\tau^2 = \tau + 1$  but finally in this inner product counterintuitively setting  $\tau$  equal to zero. This set of 240 includes the simple roots of  $H_4$

$$a_1 = \frac{1}{2}(-\sigma, -\tau, 0, -1),$$

$$a_2 = \frac{1}{2}(0, -\sigma, -\tau, 1),$$

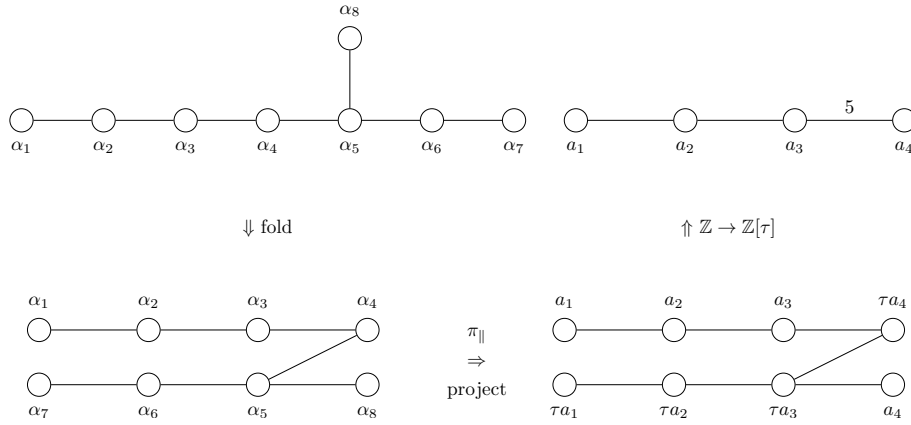
$$a_3 = \frac{1}{2}(0, 1, -\sigma, -\tau) \text{ and}$$

$$a_4 = \frac{1}{2}(0, -1, -\sigma, \tau)$$

along with their  $\tau$ -multiples

$$a_5 = \tau a_1 = \frac{1}{2}(1, -\tau - 1, 0, -\tau),$$

$$a_6 = \tau a_2 = \frac{1}{2}(0, 1, -\tau - 1, \tau),$$

FIGURE 3. Coxeter-Dynkin diagram folding and projection from  $E_8$  to  $H_4$ .

$$a_7 = \tau a_3 = \frac{1}{2}(0, \tau, 1, -\tau - 1) \text{ and}$$

$$a_8 = \tau a_4 = \frac{1}{2}(0, -\tau, 1, \tau + 1).$$

So with this inner product, most of the inner products among the two sets of  $H_4$  are not affected, but crucially, there is one product that makes a link between the two  $H_4$  diagrams, turning it precisely into the  $E_8$  diagram. The 240 icosahedral pinors therefore precisely give the 240 roots of  $E_8$ . The Cartan matrix for this set of simple roots is

$$\begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 2 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 0 & 0 & 0 & -1 & 2 \end{pmatrix},$$

which is the  $E_8$  Cartan matrix in slightly unusual form.

Surprisingly, the  $E_8$  root system has therefore been lurking in plain sight in the geometry of the Platonic solid the icosahedron for three millennia, without anyone noticing. As with the 4D induction construction, this discovery was only possible in Clifford algebra – there is much prejudice against the usefulness of Clifford algebras (since they have matrix representations) and usually matrix methods are equivalent if less insightful – but the 4D and 8D induction constructions are to my knowledge the only results that *require* Clifford algebra and were completely invisible to standard matrix methods.

## 5. $H_4$ AS A GROUP OF ROTATIONS RATHER THAN REFLECTIONS II: FROM $E_8$

The usual view is the reverse of the process shown in the previous section, inducing  $E_8$  from  $H_3$ , via two intermediate copies of  $H_4$ :  $E_8$  has an  $H_4$  subgroup as can be shown via Coxeter-Dynkin diagram foldings [16]. We take the simple roots  $\alpha_1$  to  $\alpha_8$  of  $E_8$  as shown in Fig. 3, and consider now the Clifford algebra in 8D with the usual Euclidean metric. The simple reflections corresponding to the simple roots are thus just given via  $s_{\alpha}v = -\alpha v \alpha$ . The Coxeter element  $w$  is defined as the product of all these eight simple reflections, and in Clifford algebra it is therefore simply given by the corresponding pinor  $W = \alpha_1 \dots \alpha_8$  acting via sandwiching. Its order, the Coxeter number  $h$  (i.e.  $W^h = \pm 1$ ), is 30 for  $E_8$ .

As illustrated in Figure 3, one can define certain combinations of pairs of reflections (corresponding to roots on top of each other in the Dynkin diagram folding), e.g.  $s_{a_1} = s_{\alpha_1}s_{\alpha_7}$  etc, and in a Clifford algebra sandwiching way these are given by the products of root vectors  $a_1 = \alpha_1\alpha_7$ ,  $a_2 = \alpha_2\alpha_6$ ,  $a_3 = \alpha_3\alpha_5$  and  $a_4 = \alpha_4\alpha_8$  (this is essentially a partial folding of the usual alternating folding used in the construction of the Coxeter plane with symmetry group  $I_2(h)$ , see the next section). It is easy to show that the subgroup with the generators  $s_{a_i}$  in fact satisfies the relations of an  $H_4$  Coxeter group [3, 16]: because of the Coxeter relations for  $E_8$  and the orthogonality of the combined pair the combinations  $s_a$  are easily seen to be involutions, and the 3-fold relations are similarly obvious from the Coxeter relations; only for the 5-fold relation does one have to perform an explicit calculation in terms of the reflections with respect to the root vectors. This is thus particularly easy by multiplying together vectors in the Clifford algebra, rather than by concatenating two reflection formulas of the type shown in Definition 2.1 – despite it only consisting of two terms, concatenation gets convoluted quickly, unlike multiplying together multivectors.

Since the combinations  $s_a$  are pairs of reflections, they are obviously rotations in the eight-dimensional space, so this  $H_4$  group acts as rotations in the full space, but as a reflection group in a 4D subspace. The  $H_4$  Coxeter element is given by multiplying together the four combinations  $a_i$  – its Coxeter versor is therefore trivially seen to be the same as that of  $E_8$  (up to sign, since orthogonal vectors anticommute) and the Coxeter number of  $H_4$  is thus the same as that of  $E_8$ , 30. The projection of the  $E_8$  root system onto the Coxeter plane consists of two copies of the projection of  $H_4$  into the Coxeter plane, with a relative factor of  $\tau$  (see the next section and in particular Figure 8). This is also related to the fact that on the level of the root system there is a projection which maps the 240 roots of  $E_8$  onto the 120 roots of  $H_4$  and their  $\tau$ -multiples [15, 9] (right of Fig. 3 and previous section). We therefore now consider the Coxeter plane itself.

## 6. THE COXETER PLANE

In this section, we consider the geometry of the Coxeter plane e.g. achieved by a complete folding of the  $E_8$  Dynkin diagram (Figure 4). For a given Coxeter element  $w$  of any root system, there is a unique plane called the Coxeter plane on which  $w$  acts as a rotation by  $2\pi/h$ . Projection of a root system onto the Coxeter plane is thus a way of visualising any finite Coxeter group, for instance the well-known representation of  $E_8$  is such a projection of the 240 vertices of the root system onto the Coxeter plane. In the standard theory there is an unnecessary complexification of the real geometry followed by taking real sections again just so that complex eigenvalues  $\exp(2\pi im/h)$  of  $w$ , for some integers  $m$  called exponents, can be found [11]. Unsurprisingly, in Clifford algebra these complex structures arise naturally, and the complex ‘eigenvectors’ are in fact eigenplanes where the Coxeter element acts as a rotation. We briefly discuss the 2D case of the two-dimensional family of non-crystallographic Coxeter groups  $I_2(n)$ , followed by the three-dimensional groups  $A_3$ ,  $B_3$  and  $H_3$  [6], before discussing the higher-dimensional examples.

The simple roots for  $I_2(n)$  can be taken as  $\alpha_1 = e_1$ ,  $\alpha_2 = -\cos\frac{\pi}{n}e_1 + \sin\frac{\pi}{n}e_2$ , which yields the Coxeter versor  $W$  describing the  $n$ -fold rotation encoded by the  $I_2(n)$  Coxeter element via  $v \rightarrow wv = \tilde{W}vW$  as

$$(1) \quad W = \alpha_1\alpha_2 = -\exp(-\pi e_1 e_2/n).$$

In Clifford algebra it is therefore immediately obvious that the action of the  $I_2(n)$  Coxeter element is described by a versor that encodes rotations in the  $e_1e_2$ -Coxeter-plane and yields  $h = n$  since trivially  $W^n = (-1)^{n+1}$ . Since  $I = e_1e_2$  is the bivector defining the plane of  $e_1$  and  $e_2$ , it anticommutes with both  $e_1$  and  $e_2$  such that one can take  $W$  through to the left to arrive at the complex eigenvector equation

$$(2) \quad v \rightarrow wv = \tilde{W}vW = \tilde{W}^2v = \exp(\pm 2\pi I/n)v.$$

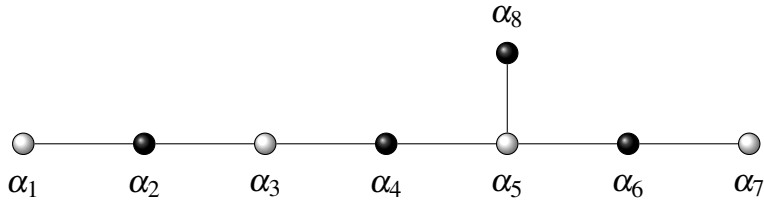


FIGURE 4. Illustration of the geometry of the Coxeter plane via diagram foldings. Since any finite Coxeter group has a tree-like diagram, one can partition the simple roots into two sets (black and white) of roots that are orthogonal within each set. Since the Cartan matrix is positive definite, it has a Perron-Frobenius eigenvector with all positive entries. This allows one to show the existence of the invariant Coxeter plane (by construction) as the bivector defined by the outer product of two vectors that are linear combinations of all the reciprocals of the white (respectively black) simple roots with the corresponding coefficients given by the entries in the Perron-Frobenius eigenvector.

This yields the standard result for the complex eigenvalues, however, in Clifford algebra it is now obvious that the complex structure is in fact given by the bivector describing the Coxeter plane (which is trivial for  $I_2(n)$ ), and that the standard complexification is both unmotivated and unnecessary.

More generally, if  $v$  lies in a plane in which  $W$  acts as a rotation, then

$$(3) \quad v \rightarrow wv = \tilde{W}vW = \tilde{W}^2v$$

still holds, whereas if  $v$  is orthogonal, one just has

$$(4) \quad v \rightarrow wv = \tilde{W}vW = \tilde{W}Wv = v.$$

Thus, if  $W$  factorises into orthogonal eigenspaces  $W = W_1 \dots W_k$  with  $v$  lying in the plane defined by  $W_1$ , then all the orthogonal  $W_i$ s commute through and cancel out, whilst the one that defines the eigenplane  $v$  lies in gives the standard complex eigenvalue equation

$$(5) \quad v \rightarrow wv = \tilde{W}vW = \tilde{W}_1 \dots \tilde{W}_k v W_1 \dots W_k = \tilde{W}_1^2 \dots \tilde{W}_k W_k v = \tilde{W}_1^2 v.$$

If  $m$  is an exponent then so is  $h - m$ ; in particular 1 and  $h - 1$  are always exponents (from the Coxeter plane) – in Clifford algebra it is easy to see that these are just righthanded and lefthanded rotations in the respective eigenplanes, with bivectors giving the complex structures. If  $W$  has pure vector factors then these act as reflections and trivially yield the exponents  $h/2$ .

The Pin group/eigenblade description in GA therefore yields a wealth of novel geometric insight, and we now consider higher-dimensional examples. For 3D and 4D groups, the geometry is completely governed by the above 2D geometry in the Coxeter plane, since the remaining normal vector (3D) or bivector (4D) are trivially fixed. For  $H_3$  one has  $h = 10$  and complex eigenvalues  $\exp(2\pi mi/h)$  with the exponents  $m = \{1, 5, 9\}$ . For simple roots  $\alpha_1 = e_2$ ,  $-\alpha_2 = (\tau - 1)e_1 + e_2 + \tau e_3$  and  $\alpha_3 = e_3$ , the Coxeter plane bivector is  $B_C = e_1 e_2 + \tau e_3 e_1$  and the Coxeter element versor  $2W = -\tau e_2 - e_3 + (\tau - 1)I$  (here  $I = e_1 e_2 e_3$ ) with eigenvalues  $\exp(\pm 2\pi B_C/h)$ , which corresponds to  $m = 1$  and  $m = 9$ . The vector  $b_C = B_C I = -\tau e_2 - e_3$  orthogonal to the Coxeter plane can only get reversed (since the Coxeter element in 3D is an odd operation), so one has  $-\tilde{W}b_C W = -b_C = \exp(\pm 5 \cdot 2\pi B_C/h)b_C$  which gives  $m = 5$ .  $A_3$  and  $B_3$  is very similar, they have Coxeter numbers  $h = 4$  and  $h = 6$ , respectively, and thus exponents  $m = \{1, 2, 3\}$  and  $m = \{1, 3, 5\}$ . The exponents 1 and  $h - 1$  correspond to a rotation in the Coxeter plane in which the Coxeter element acts by  $h$ -fold rotation, whilst the normal to the Coxeter plane gets simply inverted, corresponding to the cases  $h/2$  ( $m = 2$  and  $m = 3$  for  $A_3$  and  $B_3$ , respectively).

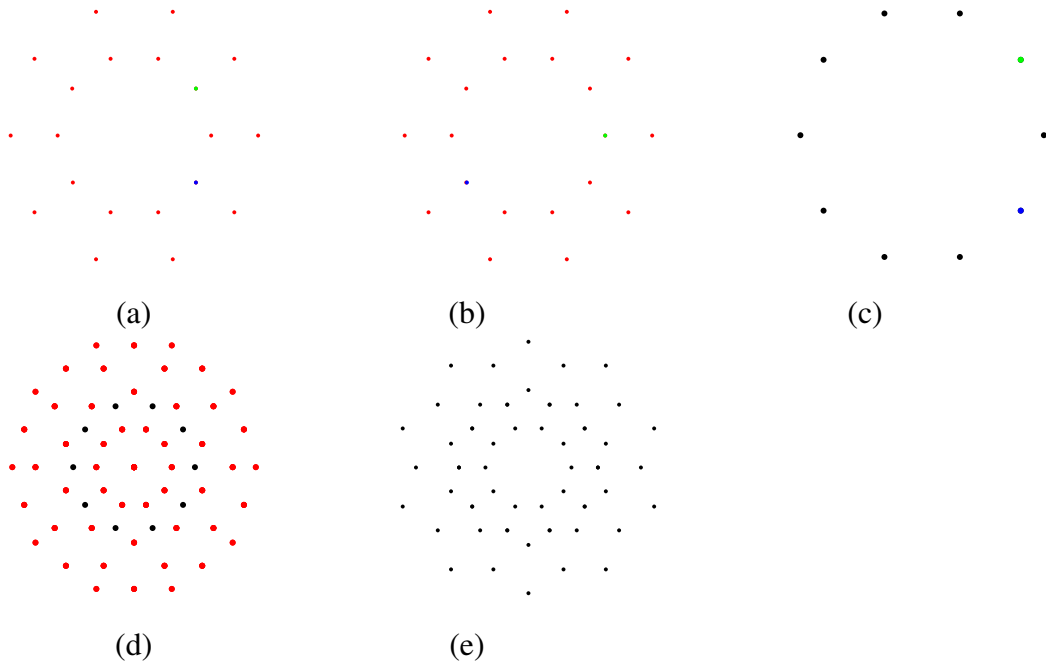


FIGURE 5. Coxeter projections of  $A_4$  (a) and b)) and  $H_2$  (panel c)),  $H_2^{aff}$  (d)) and  $D_6$  Coxeter projection (e)). The action of the Coxeter element as an  $h$ -fold rotation is visualised by the two coloured dots (green and blue) that are rotated into another by this rotation.

We now consider the four-dimensional cases  $A_4$ ,  $B_4$ ,  $D_4$ ,  $F_4$  and  $H_4$ . We explain the case of  $A_4$  in detail, which is known to have exponents  $\{1, 2, 3, 4\}$ . We take as the simple roots  $\alpha_1 = 1/\sqrt{2}(e_2 - e_1)$ ,  $\alpha_2 = 1/\sqrt{2}(e_3 - e_2)$ ,  $\alpha_3 = 1/\sqrt{2}(e_4 - e_3)$  and  $\alpha_4 = 1/2(\tau e_1 + \tau e_2 + \tau e_3 + (\tau - 2)e_4)$ . It is easy to calculate that reflections in these yield a root polytope of 20 vertices as well as the correct  $A_4$  Cartan matrix. This matrix has Perron-Frobenius eigenvector  $(1, \tau, \tau, 1)^T$  and one can construct the Coxeter plane bivector as  $B_C \propto -e_1e_3 - e_1e_4 + e_2e_3 + e_2e_4 - 1/2(\tau - 1)e_3e_4$  via the two vectors  $e_3 + e_4$  and  $-e_1 + e_2 + e_3 + (2\tau + 1)e_4$  arrived at from the Perron-Frobenius eigenvector and the reciprocal frame of the simple roots as illustrated in Figure 4. The Coxeter element  $W = \alpha_3\alpha_1\alpha_2\alpha_4$  is calculated to be  $4W = 1 - e_2e_3 + e_1e_4 + (\tau - 1)(e_3e_4 + e_2e_4 - e_1e_3) - (\tau + 1)e_1e_2 - (2\tau - 1)e_1e_2e_3e_4$ . It is easy to show that  $\tilde{W}B_CW = B_C$  and the Coxeter element therefore indeed stabilises the Coxeter plane. However, we are claiming that the Coxeter element can actually be written as bivector exponentials in the planes defined by  $B_C$  and  $IB_C$ , with angles given by the exponents  $\{1, 2, 3, 4\}$ . These are given as left- and righthanded rotations in the two planes as shown in Figure 5 a) and b). The Coxeter projection of the 20 vertices forms two concentric decagonal circles – in the Coxeter plane  $w$  acts as a rotation by  $2\pi/5$  (as denoted by the two coloured vertices in a) with the Coxeter element taking one into the other), whilst in the plane  $IB_C$  it acts as a rotation by  $4\pi/5$ , as shown in b). It is easy to check that  $W$  can indeed be written as  $W = \exp(\frac{\pi}{5}B_C)\exp(-\frac{2\pi}{5}IB_C)$ . It is clear that taking the product of simple roots in the Coxeter element in a different order introduces overall minus signs as well as minus signs in the exponentials, so we will not worry about these from now on.

$A_4$  is unusual in that the projection from 4D only yields two concentric circles in the Coxeter plane. In fact, it consists of two copies of  $H_2$  (panel c)) with a relative factor of  $\tau$ . This is similar to the situation for  $E_8$  and  $H_4$ , as explained in Fig. 3, since by removing four of the eight nodes one gets a diagram folding from  $A_4$  to  $H_2$ . In [8], we were considering affine extensions of  $H_2$  by adding a translation operator and taking the orbit under the compact group (panel d)). What is striking is that this  $H_2^{aff}$  point set, i.e. an affine extension of the decagon after one

unit translation, is very nearly the Coxeter projection of  $D_6$ , which is shown in panel e). The situation for the other 4D groups is similar, as shown in Figure 6, which shows for the groups  $B_4$ ,  $D_4$ ,  $F_4$  and  $H_4$  that the Coxeter element acts in the Coxeter plane  $B_C$  as rotations by  $\pm 2\pi/h$ , and in the plane defined by  $IB_C$  as  $h$ -fold rotations giving the other exponents. For  $B_4$  one has exponents  $\{1, 3, 5, 7\}$  as shown in panels a) and b) and the decomposition into eigenblades of the Coxeter element  $W = \exp(\frac{\pi}{8}B_C)\exp(\frac{3\pi}{8}IB_C)$  reflects this.

$D_4$  has exponents  $\{1, 3, 3, 5\}$  which is reflected in the fact that the Coxeter versor can be written as  $W = \exp(\frac{-\pi}{6}B_C)\exp(\frac{\pi}{2}IB_C) = -\exp(\frac{-\pi}{6}B_C)IB_C$ . The Coxeter projections of  $D_4$  into the  $IB_C$  plane demonstrate that the other factor in  $W$  that does not come from  $B_C$  is actually the product of two vectors rather than a bivector exponential (since the angle is  $\pi/2$ ): on some vectors it acts as a rotation by  $3 = h/2$  in the plane, others it projects onto the origin (Fig. 6 panels c) and d)).  $F_4$  has exponents  $\{1, 5, 7, 11\}$  which again is evident from the Clifford factorisation  $W = \exp(\frac{\pi}{12}B_C)\exp(\frac{5\pi}{12}IB_C)$  and the way it acts on the two planes (panels e) and f)).  $H_4$  has exponents  $\{1, 11, 19, 29\}$  and factorisation  $W = \exp(\frac{\pi}{30}B_C)\exp(\frac{11\pi}{30}IB_C)$  (panels g) and h)).

The Coxeter versor of  $D_6$  can be written as  $W = \exp(\frac{\pi}{10}B_C)\exp(\frac{3\pi}{10}B_2)B_3$  for certain bivectors  $B_2$  and  $B_3$  from which it is evident that the exponents are indeed  $\{1, 3, 5, 5, 7, 9\}$ , with two reflections and 10-fold rotations in the Coxeter plane and another orthogonal plane (Figure Fig. 7 a) and c)). Since as in Fig. 3 there is also a diagram folding from  $D_6$  to  $H_3$  ( $H_3$  as we saw above has exponents  $\{1, 5, 9\}$ ), the  $D_6$  projection again actually consists of two copies of that of  $H_3$  with a relative factor of  $\tau$  (panel b)), but the  $H_3$  projection already has radii with a relative factor of  $\tau$  such that two orbits of the  $D_6$  projection fall on top of each other.

Not surprisingly, the Coxeter versor for  $E_8$  can be written as

$$W = \exp(\frac{\pi}{30}B_C)\exp(\frac{7\pi}{30}B_2)\exp(\frac{11\pi}{30}B_3)\exp(\frac{13\pi}{30}B_4).$$

This gives the well-known exponents  $\{1, 7, 11, 13, 17, 19, 23, 29\}$  a more geometric meaning as 30-fold rotations in four orthogonal eigenplanes (Fig. 8). As we have alluded to in Section 5, the Coxeter projection of  $E_8$  actually consists of two copies of that of  $H_4$  in the bottom row of Figure 6 with a relative radius of  $\tau$ . We recall that  $H_4$  has exponents  $\{1, 11, 19, 29\}$  and since it is a subgroup of  $E_8$  they have of course the same Coxeter element and number, and share two eigenplanes with exponents 1 and 29 as well as 11 and 19.

## 7. CONCLUSIONS

We have shown that with the help of Clifford algebra **all** exceptional root systems can in fact be constructed from the 3D root systems. This offers a revolutionarily new way of viewing these phenomena in terms of spinorial geometry of 3D, with huge potential implications for the many areas in which these symmetries appear. Likewise, the geometry of the Coxeter plane is best viewed in a Clifford algebra framework, which provides geometric meaning and insight, for instance the complex eigenvalues in standard theory are just seen to be rotations in eigenplanes of the Coxeter element. Since root systems exist on a vector space with an inner product, the associated Clifford algebra of this space is actually the most natural framework to use for such root systems and Coxeter groups, in particular as Clifford algebra affords a uniquely simple reflection formula.

## REFERENCES

- [1] Vladimir Igorevich Arnold. Symplectization, complexification and mathematical trinitities. *The Arnoldfest*, pages 23–37, 1999.
- [2] Vladimir Igorevich Arnold. *Mathematics: Frontiers and perspectives*. Amer Mathematical Society, 2000.
- [3] Nicolas Bourbaki. *Groupes et algèbres de Lie, chapitres 4, 5 et 6*. Masson, Paris, 1981.

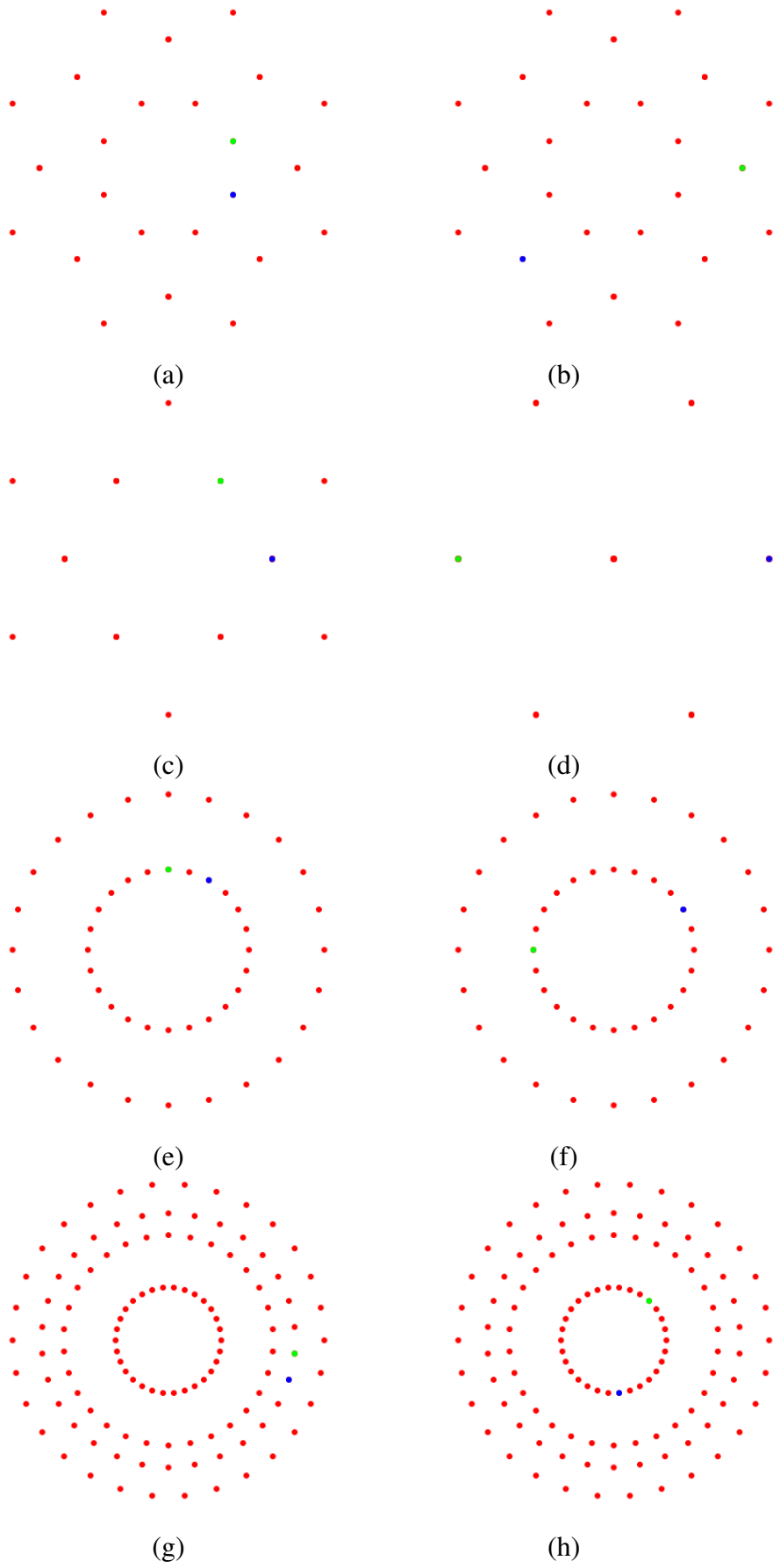


FIGURE 6. Coxeter projections of  $B_4$ ,  $D_4$ ,  $F_4$  and  $H_4$ . In each row the plots are the action of the Coxeter element in the Coxeter plane given by  $B_C$  and in the plane given by  $IB_C$ .



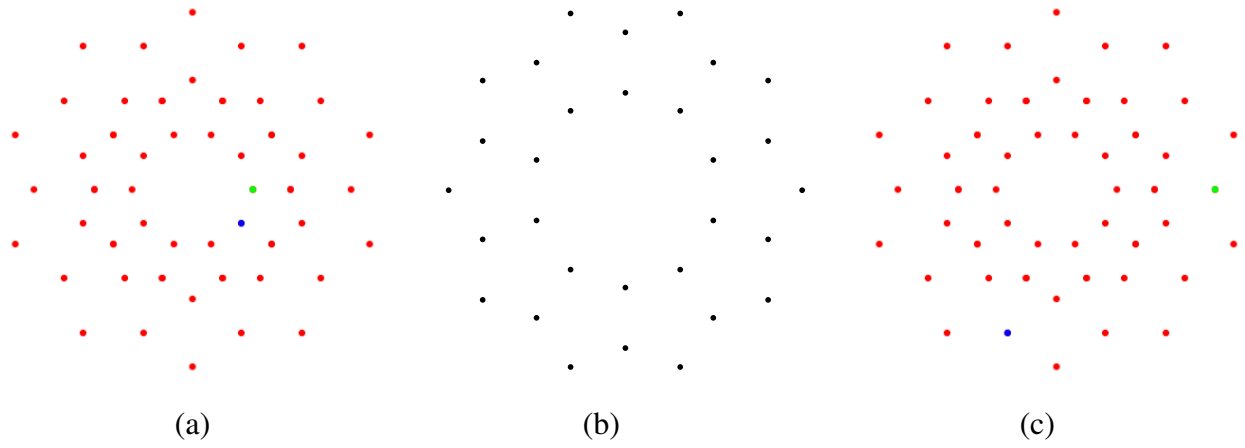


FIGURE 7. Coxeter projections of  $D_6$  (panels a) and c)) and  $H_3$  (panel b)). The two innermost radii of the  $H_3$  projection are precisely in the ratio of  $\tau$ , such that since the  $D_6$  projection consists of two copies of the  $H_3$  with a relative radius of  $\tau$ , two orbits actually coincide. In the Coxeter plane and the other eigenplane the Coxeter element acts by 10-fold rotation, as expected.

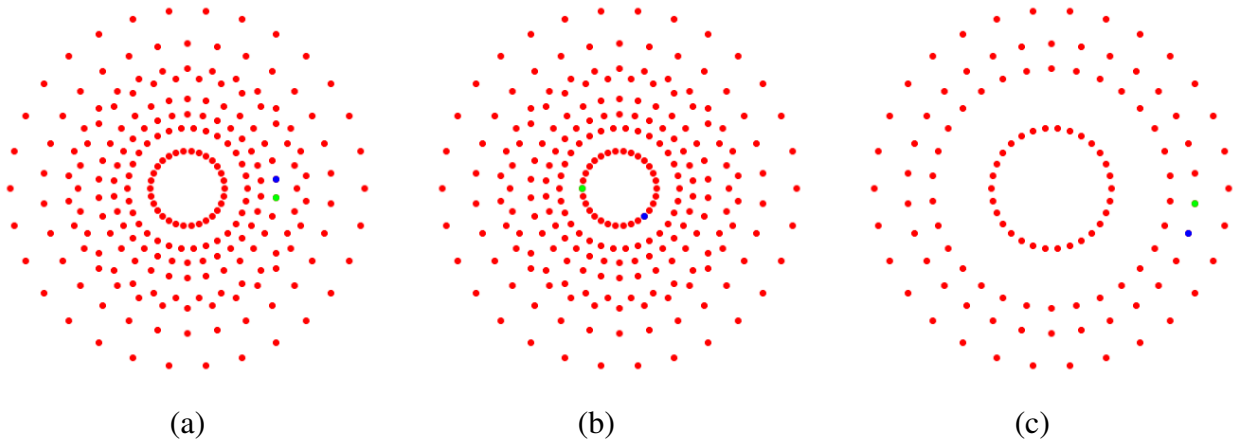


FIGURE 8. Coxeter projections of  $E_8$  and  $H_4$ . Again,  $E_8$  consists of  $H_4$  and  $\tau H_4$  with the Coxeter element acting as a 30-fold rotation in both.

- [4] Pierre-Philippe Dechant. Clifford algebra unveils a surprising geometric significance of quaternionic root systems of Coxeter groups. *Advances in Applied Clifford Algebras*, 23(2):301–321, 2013. 10.1007/s00006-012-0371-3.
- [5] Pierre-Philippe Dechant. Platonic solids generate their four-dimensional analogues. *Acta Crystallographica Section A: Foundations of Crystallography*, 69(6):592–602, 2013.
- [6] Pierre-Philippe Dechant. A Clifford algebraic framework for Coxeter group theoretic computations. *Advances in Applied Clifford Algebras*, 24(1):89–108, 2014.
- [7] Pierre-Philippe Dechant. Rank-3 root systems induce root systems of rank 4 via a new clifford spinor construction. *Journal of Physics: Conference Series*, 597(1):012027, 2015.
- [8] Pierre-Philippe Dechant, Céline Boehm, and Reidun Twarock. Novel Kac-Moody-type affine extensions of non-crystallographic Coxeter groups. *Journal of Physics A: Mathematical and Theoretical*, 45(28):285202, 2012.
- [9] Pierre-Philippe Dechant, Céline Boehm, and Reidun Twarock. Affine extensions of non-crystallographic Coxeter groups induced by projection. *Journal of Mathematical Physics*, 54(9):–, 2013.
- [10] C. Doran, D. Hestenes, F. Sommen, and N. Van Acker. Lie groups as spin groups. *Journal of Mathematical Physics*, 34(8):3642–3669, 1993.
- [11] J. E. Humphreys. *Reflection groups and Coxeter groups*. Cambridge University Press, Cambridge, 1990.
- [12] M. Koca, M. Al-Barwani, and R. Koç. Quaternionic root systems and subgroups of the  $\text{Aut}(\mathbb{F}_4)$ . *Journal of Mathematical Physics*, 47(4):043507–+, April 2006.

- [13] M. Koca, R. Koç, and M. Al-Barwani. Quaternionic roots of  $SO(8)$ ,  $SO(9)$ ,  $F_4$  and the related Weyl groups. *Journal of Mathematical Physics*, 44:3123–3140, July 2003.
- [14] John McKay. Graphs, singularities, and finite groups. In *Proc. Symp. Pure Math*, volume 37, pages 183–186, 1980.
- [15] R. V. Moody and J. Patera. Quasicrystals and icosians. *Journal of Physics A: Mathematical and General*, 26(12):2829, 1993.
- [16] O. P. Shcherbak. Wavefronts and reflection groups. *Russian Mathematical Surveys*, 43(3):149, 1988.



# THE CONSTRUCTION OF 3D CONFORMAL MOTIONS

Leo Dorst

Intelligent Systems Laboratory Amsterdam  
University of Amsterdam, Amsterdam, The Netherlands  
l.dorst@uva.nl [presenter, corresponding]

ABSTRACT. This paper exposes a very geometrical yet directly computational way of working with conformal motions in 3D. With the increased relevance of conformal structures in architectural geometry, and their traditional use in CAD, the paper should be useful to designers and programmers. In brief, we exploit the fact that any 3D conformal motion is governed by two point pairs. The conformal motion of a point is composed of two orthogonal circular motions in the planes determined by those point pairs. The resulting orbit of a point is a conformal spiral on a Dupin cyclide.

These results are compactly expressed and programmed using conformal geometric algebra (CGA), and this paper can serve as an introduction to its usefulness. In CGA language, the conformal motion is a *rotor* (or spinor), which can be written as the exponential of a bivector. That bivector can be decomposed (in 3D almost always uniquely) as a sum of commuting 2-blades  $B_+$  and  $B_-$ . These 2-blades are the point pairs, and they are direct elements of computation in CGA. For a point  $x$ , the 3-blades  $x \wedge B_+$  and  $x \wedge B_-$  are orthogonal circles, and the rotor moves a bit along each, in a ratio determined by the rotor decomposition. (The theoretical background to these statements is the rather technical 2011 paper *Square Root and Logarithm of of Rotors in 3D Conformal Geometric Algebra Using Polar Decomposition* by Dorst and Valkenburg.) The richness of conformal motions (and their Dupin cyclides) derives from the various types of point pairs and their interaction. In CGA, these are all simply 2-blades, and treated in a completely unified manner. However, their geometrical interpretation differs: the 2-blades of 3D CGA may represent a real point pair, an imaginary point pair (but computed with reals since only its negative square enters computations), a tangent vector, a direction vector, a dual line (rotation axis) or a ‘flat point’ (scaling center). With these unifying insights in their construction the specification, computation and interpolation of 3D conformal motions becomes much more straightforward and interactive than in the usual coordinate-based approach, or through the Atiyah factorization in a fixed sequence of standard conformal motions.

The paper will expose this geometrical ‘language’ of 3D conformal motions, and the accompanying presentation will demonstrate some of the interactive playing this enables, by direct manipulation of the defining point pairs. We will for instance show how the usual Chasles decomposition of Euclidean motions is included (and extended to similarities), and how to describe all torus knots by the ratio of weights of two point pairs.

The advantage of using CGA is its covariance: conformal motions for other primitives such as circles are computed using exactly the same formulas, and hence the software operations, as motions of points. This generates an interesting class of easily generated shapes of spatial circles moving conformally on Dupin cyclide spirals.

The figure contains some screen shots of the demos we plan to give during the presentation.

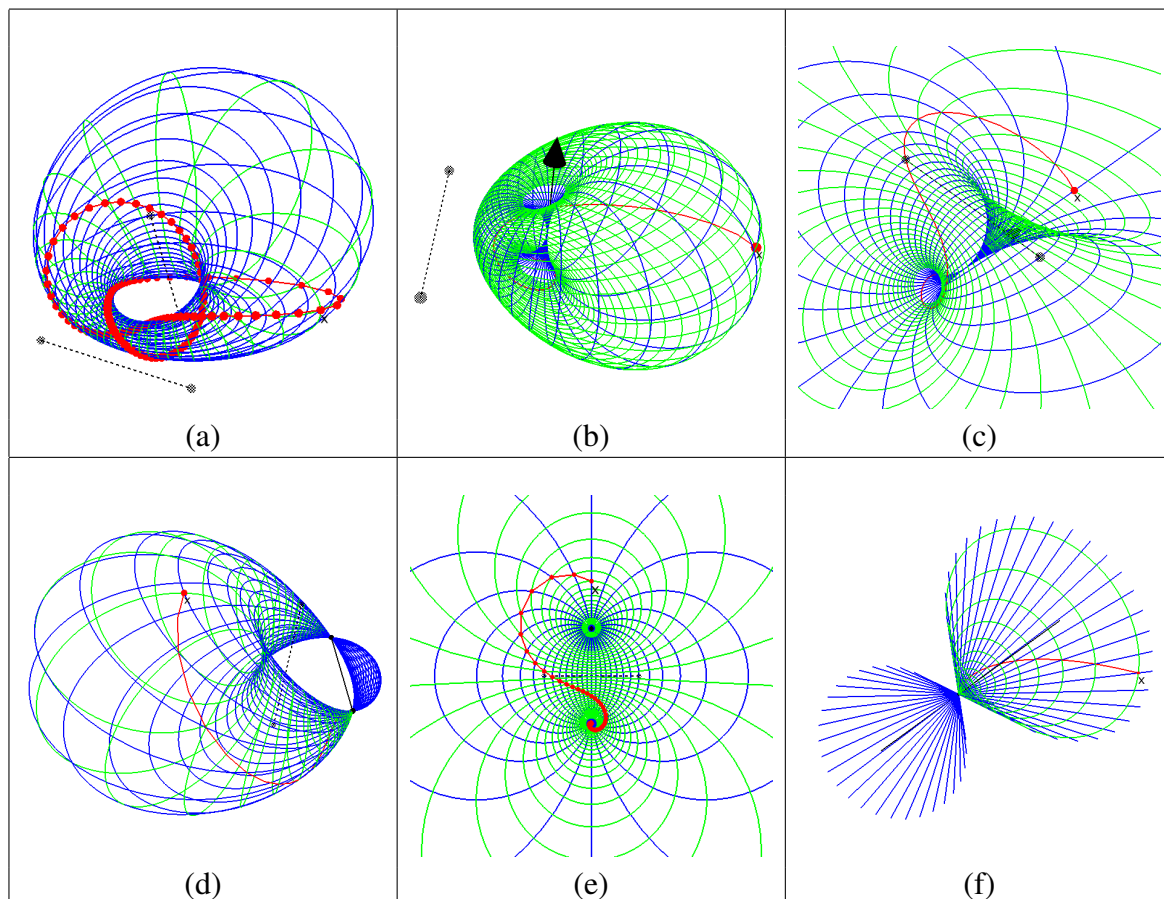


FIGURE 1. Some shots of the interactive demos. The point pairs determining the motion are in black, the orbits of the simple conformal motion of each point pair in green and blue. The orbit of a particular point under the composite motion is indicated in red. (a) a (knotted) ring cyclide spiral motion, determined by two commuting imaginary point pairs; (b) a cuspidal cyclide spiral motion, determined by an imaginary point pair and a tangent vector; (c) a parabolic cuspidal cyclide spiral motion, determined by an imaginary point pair and a tangent vector, for a point on a carrier of a point pair; (d) a spindle cyclide spiral motion, determined by an imaginary point pair and a real point pair; (e) a planar horn cyclide spiral motion, for a point in the common plane of the defining point pairs; (f) a spiral along a cone, the motion determined by a flat point and a dual line.

# GA AND CICT FOR STRONGER ARBITRARY MULTI-SCALE BIOMEDICAL AND BIOENGINEERING SOLUTIONS

R. A. Fiorini<sup>a</sup>

<sup>a</sup>Department of Electronics, Information and Bioengineering (DEIB)  
Politecnico di Milano University, Milano, Italy  
rodolfo.fiorini@polimi.it [presenter, corresponding]

**ABSTRACT.** The aim of the present paper is to provide the first concise overview of a natural framework for arbitrary multi-scale computer science and systems biology computational modeling based on a synergic coupling between GA and CICT to get stronger arbitrary-scale biomedical and bioengineering computational solutions. GA and its extension to geometric calculus (GC) unify, simplify, and generalize many areas of mathematics that involve geometric ideas. For detecting and modeling a minute change in resistance or capacitance at biostructure nanoscale, we need stronger research and computational tools able to overcome classic modeling limitation. We present key points solution to arbitrary multi-scale modeling problems. The fundamental principles on computational information conservation theory (CICT), for arbitrary multi-scale system modeling from basic generator and relation through discrete paths denser and denser to one another, towards a never ending "blending quantum continuum," are recalled. This paper is a relevant contribute towards arbitrary multi-scale computer science and systems biology modeling, to show how GA and GC unified mathematical language combined to CICT approach can offer an effective and convenient "Science 2.0" universal framework to develop innovative application and beyond, towards a more sustainable economy and wellbeing, in a global competition scenario.

## INTRODUCTION

The aim of the present paper is to provide the first concise overview of a natural framework for arbitrary multi-scale computer science and systems biology computational modeling based on a synergic coupling between GA (Geometric Algebra) and *CICT* (Computational Information Conservation Theory) to get stronger arbitrary-scale biomedical and bioengineering computational solutions. GA and its extension to geometric calculus (GC) [14] unify, simplify, and generalize many areas of mathematics that involve geometric ideas. They also provide a unified mathematical language for physics, engineering, and the geometrical aspects of information science and computer science, quite well-known and appreciated by reliable scientific modeling community. There is no doubt that the IC (Infinitesimal Calculus) has played in the past and will play in future a major role in the mathematical treatment of engineering, bioengineering and biomedical modeling problems. Nevertheless, we must be aware that its addiction specifically on arbitrary multi-scale system modeling hides some fundamental features of the natural phenomenon being represented, such as both its intrinsic invariant features (ontological, topological, geometrical, etc.) and its extrinsic interaction with observer, to get overall phenomenon formal observation, description and representation by humans. The addiction is such that, since the digital computer requires a finite formulation of physical laws, it is preferred to discretize the differential equations, rather than considering other more convenient tools like FDC (Finite Differences Calculus), Finite Difference Time Domain (FDTD) or more sophisticated algebraic methods. FDC and FDTD deal especially with discrete functions, but they may be applied to continuous function too and to continuum problems, with no loss of generality. They can deal with both discrete and continuum problem categories conveniently. Unfortunately, even traditional FDC, FDTD and more sophisticated and advanced algebraic approaches are unable to conserve overall system formal information description. As a matter of fact, current Number Theory and modern Numeric Analysis still use mono-directional

interpretation for numeric group generator and relations, so information entropy generation cannot be avoided in current computational algorithm and application [8]. Furthermore, traditional digital computational resources are unable to capture and to manage not only the full information content of a single Real Number  $\mathbb{R}$ , but even Rational Number  $\mathbb{Q}$  is managed by information dissipation (e.g. finite precision machine, truncating, rounding, etc.) For observing, modeling and determining single molecule transport characteristics or for detecting a minute change in resistance or capacitance at biostructure nanoscale, we need stronger research and computational modeling tools able to overcome the above operative limitation. According to fresh *CICT* result, scientific community has acquired new awareness about traditional rational number system  $\mathbb{Q}$  numeric properties, quite recently [7]. Thanks to this line of generative thinking, it is possible to realize that traditional rational number system can be even regarded as a highly sophisticated open logic, powerful and flexible LTR (Left-To-Right) and RTL (Right-To-Left) formal language of languages, with self-defining consistent words and rules, starting from elementary generators and relations [7]. Further, *CICT* ODR (Observation, Description, Representation) approach can take advantage immediately from those properties to develop system computational functional closures to achieve information conservation countermeasure at each operative step automatically [10]. Then, all computational information usually lost by classic information approach, based on the traditional noise-affected data stochastic model only, can be captured and fully recovered to arbitrary precision by a corresponding complementary codomain, step-by-step. Theoretically, codomain information can be used to correct any computed result, achieving computational information conservation (virtually noise-free data), according to *CICT* Infocentric World-view [11]. In this way, overall system resilience and antifragility can be developed quite easily [23]. *CICT* has shown that, by Shannon entropy approach only, even the current, most sophisticated instrumentation system is completely unable to reliably discriminate so called "random noise" (RN) from any combinatorially optimized encoded message, which *CICT* called "deterministic noise" (DN) [10]. Stochastic vs. Combinatorially Optimized Noise generation ambiguity emphasises the major double-bind problem in current most advanced research laboratory and instrumentation system, just at the inner core of human knowledge extraction by experimentation in current science [10]. QFT (Quantum Field Theory) has emerged from a major ontological paradigm shift with respect to Classical Physics which still provides the traditional framework of the vision of nature of most "Science 1.0" current scientists. This change of paradigm has not yet been completely grasped by contemporary science so that not all the implications of this change have been realized hitherto, even less their related applications. So, the discreteness approach, developed under the "discreteness hypothesis" assumption, in specific scientific disciplines, has been considered in peculiar application areas only. It has been further slowly developed by a few specialists and less understood by a wider audience. It is the fresh QFT approach. Unfortunately, over the centuries, the above two large scientific research areas (continuum based and discreteness based) have followed separate mathematical development paths with no or quite little, inconsistent synergic coupling. That is the main reason why QFT is still mostly overlooked by traditional scientific and engineering researchers for arbitrary multi-scale system modeling, from system nano-microscale to macroscale. Unfortunately, the "probabilistic veil" can be very opaque computationally, in a continuum-discrete arbitrary multi-scale environment, and misplaced precision leads to information dissipation and confusion [22]. To grasp stronger physical and biological system correlates, scientists need two intelligently articulated hands: both stochastic and combinatorial approaches synergically articulated by natural coupling. The *CICT* approach brings classic and quantum information theory together in a single framework, by considering information not only on the statistical manifold of model states but also on the combinatorial manifold of low-level discrete, phased generators and empirical measures of noise sources, related to experimental high-level overall perturbation [11]. The advantages of *CICT* approach combined to GA and GC unified mathematical language can offer an effective and convenient "Science 2.0"

universal framework to develop innovative application and beyond, towards a more sustainable economy and wellbeing, in a global competition scenario.

## 1. INERTIAL OBSERVER POV

The development of GA into a full-blown GC capable of handling all aspects of differential geometry required a fusion of differential and integral calculus with the concepts of GA in the past years. This task involved a reformulation of manifold theory at the most fundamental level and was carried out in [14] with considerable detail. As an example with important implications, let us outline its application to classical relativistic electrodynamics, applied to biological system modeling (e.g. fullwave electromagnetic modeling of brain waves). It is well known that both the time domain and frequency domain based numerical computational electromagnetic methods (i.e. Method of Moments (MoM), the Finite Element Method (FEM), etc.) for solving the Maxwell's equations suffer from the so-called "low-frequency-breakdown" problem [15]. They can only go down to a few hundred MHz in frequency, below which the result they yield becomes very inaccurate relatively quickly. It is not uncommon, therefore, to resort to quasi-static solvers once the frequency of interest falls below a certain frequency (say a few MHz), and to ignore the contribution of the displacement currents, and, hence, the coupling between the electric and magnetic fields. Unfortunately, however, this approximation is not valid for most of the materials inside the head, since the  $\sigma/(\omega\epsilon)$  ratio ( $\sigma \equiv$  medium conductivity,  $\omega \equiv$  (angular) frequency),  $\epsilon \equiv$  medium permittivity ratio) of these materials is typically close to 1 [15, 19]. In fact, the quasi-static potential differs from the full-wave potential by nearly 30 % to 50 % [5], supporting the argument that a full-wave solution should be derived even at low frequencies for the head-modeling problem, since the quasi-static approach is not sufficiently accurate for the problem at hand. All the above, taking into account that neural activity inside the brain results in low frequency waves known as brain waves. These brain waves can be further classified into delta (0.1 to 3 Hz), theta (4 to 7 Hz), alpha (8 to 12 Hz), beta (12 to 30 Hz) and gamma (30 to 100 Hz) waves based on the rate of neural activity inside the brain. The successes of neuroscience in the study of the structural and biochemical properties of neurons, glia cells, and all the biological units and cellular structures in the brain have not yet filled the gap between the behavior understood at cellular level (microscale) and the macroscopic dynamics involved in the traffic between the brain and the world around it. There is an essential problem in the study of brain function (mesoscale dynamics) that even today, after so many years since Karl Lashley posed his dilemma, still waits for a solution [17]. In quantum physics, the space-time distribution of matter and energy has a coarse-grained structure which allows its representation as an ensemble of quanta (particle representation). The local phase invariance is shown to hold if a field exists which is connected to the space-time derivatives of the phase. GA can show how spacetime invariant physical quantities can be related to the variables employed by an inertial observer quite easily. We take into consideration a generic electromagnetic field  $F$ , described by Riemann-Silberstein vector [1], and we follow the line of thought reported in spacetime algebra (STA) [14, 12]. STA is built up from combinations of one time-like basis vector  $\gamma_0$  and three orthogonal space-like vectors  $\{\gamma_1, \gamma_2, \gamma_3\}$ , under the multiplication rule:

$$\gamma_\mu \gamma_\nu + \gamma_\nu \gamma_\mu = 2\eta_{\mu\nu},$$

where  $\eta_{\mu\nu}$  is the Minkowski metric with signature (+ - - -).  $F$  is defined as a complexified 3-dimensional vector field. The value of  $F$  at an event is a bivector according to GA [14]. The field bivector  $F$  is the same for all observers; there is no question about how it transforms under a change of reference system. However, it is easily related to a description of electric and magnetic fields in a given inertial system. An inertial system is determined by a single unit timelike vector  $\gamma_0$ , which can be regarded as tangent to the worldline of an observer at rest in the system. According to STA this vector determines a split of spacetime into space and time, in other words a projection from 4-dimensional space into (3+1)-dimensional space, with a chosen reference frame by means of the following two operations:



- a collapse of the chosen time axis, yielding a 3D space spanned by bivectors;
- a projection of the 4D space onto the chosen time axis, yielding a 1D space of scalars;

which are most simply expressed by the equation:

$$(1) \quad x\gamma_0 = t + \mathbf{x},$$

where  $t = x \cdot \gamma_0$  and  $\mathbf{x} = x \wedge \gamma_0$ . This is just an example of a projective transformation. It is a linear mapping of each spacetime point  $x(\gamma_1, \gamma_2, \gamma_3)$  into a scalar  $t$  designating a time and a vector  $\mathbf{x}$  designating a position. The position vectors for all spacetime points compose a 3-dimensional Euclidean vector space  $\mathbb{R}^3$ . We denote the vectors in  $\mathbb{R}^3$  with boldface type to distinguish them from vectors in  $\mathbf{R}^3$ . The equation  $\mathbf{x} = x \wedge \gamma_0$  tells us that a vector in  $\mathbb{R}^3$  is actually a bivector in  $\mathbf{R}_{1,3}^2$ . In fact,  $\mathbb{R}^3$  consists of the set of all bivectors in  $\mathbf{R}_{1,3}^2$  which have the vector  $\gamma_0$  as a factor. Algebraically, this can be characterized as the set of all bivectors in  $\mathbf{R}_{1,3}^2$  which anticommute with  $\gamma_0$ . This determines a unique mapping of the electromagnetic bivector  $F$  into the geometric algebra  $\mathbf{R}_3$  of the given inertial system. The space-time split of  $F$  by  $\gamma_0$  is obtained by decomposing  $F$  into a part:

$$(2) \quad \mathbf{E} = 1/2(F - \gamma_0 F \gamma_0),$$

which anticommutes with  $\gamma_0$  and a part:

$$(3) \quad i\mathbf{B} = 1/2(F + \gamma_0 F \gamma_0),$$

which commutes with  $\gamma_0$ , so we have:

$$(4) \quad F = \mathbf{E} + i\mathbf{B},$$

where  $i = \gamma_0 \gamma_1 \gamma_2 \gamma_3$  is the unit pseudoscalar. Although  $i\mathbf{B}$  commutes with  $\gamma_0$ ,  $\mathbf{B}$  must anticommute since  $i$  does. Therefore, we are right to denote  $\mathbf{B}$  as a vector in  $\mathbb{R}^3$ . Of course,  $\mathbf{E}$  and  $\mathbf{B}$  in (4) are just the electric and magnetic fields in the  $\gamma_0$ -system, and the split of  $F$  into electric and magnetic fields will be different for different inertial systems. The geometric algebra generated by  $\mathbb{R}^3$  is identical with the even subalgebra of  $\mathbf{R}_{1,3}$ , so we write:

$$(5) \quad \mathbf{R}_3 = \mathbf{R}_{1,3}^+.$$

Moreover, (4) determines a split of the bivectors in  $\mathbf{R}_{1,3}^2$  into vectors and bivectors of  $\mathbf{R}_3$ , as expressed by writing

$$(6) \quad \mathbf{R}_{1,3}^2 = \mathbf{R}_3^1 + \mathbf{R}_3^2.$$

This split is not unique, however, as it depends on the choice of the vector  $\gamma_0$ . A complete discussion of space-time splits is given in [12, 13]. Finally, we should point out that the purpose of a spacetime split is merely to relate invariant physical quantities to the variables employed by a specific inertial observer.

## 2. QFT INTERACTION DYNAMICS

For the purpose of mapping the brain, we are interested in estimating the fields at different points inside the head in the frequency range of 0.1–100 Hz when either one or many sources are located inside the head. In the case of a system made up of electrically charged components (nuclei and electrons of atoms), as, for instance, a biological system, this is just the electromagnetic (e.m.) potential  $\mathbf{A}_\mu$ , where  $\mu$  is the index denoting the usual four space-time coordinates  $\gamma_0 = ct, \gamma_1, \gamma_2, \gamma_3$ . The electric and magnetic fields are suitable combinations of the space-time derivatives of  $\mathbf{A}_\mu$ . In order to get the local phase invariance, we should assume that the system Lagrangian is invariant with respect to specific changes of the field  $\mathbf{A}_\mu$ . Thus a specific principle of invariance, named "gauge invariance," emerges; hence the name "gauge field" denotes  $\mathbf{A}_\mu$ . Actually it is well known that the Maxwell equations just obey the gauge invariance, which in quantum physics becomes the natural partner of the phase invariance to produce our world. Quantum fluctuations give rise to e.m. potentials which spread the phase fluctuations beyond

the system at the phase velocity. This gives an intrinsic nonlocalizability to the system and prevents a direct observation of quantum fluctuations. Through the e.m. potential, the system gets a chance to communicate with other systems. Notice that all e.m. interactions occur in a two-level way; the potential keeps the interacting particles phase-correlated whereas the combination of its space-time derivatives, named e.m. field, accounts for the forces involved. The lower level, the potential, becomes physically observable only when the phase of the system assumes a precise value. The structure of electrodynamics makes possible the presence of a potential also when both electric and magnetic fields are absent, whereas on the contrary fields are always accompanied by potentials. The above solution, which stems from the mathematical formalism of QFT [2], opens the possibility of tuning the fluctuations of a plurality of systems, producing therefore their cooperative behavior. However, some conditions must be met in order to implement such a possibility. Let us, first of all, realize that in quantum physics the existence of gauge fields, such as the e.m. potential, dictated by the physical requirement that the quantum fluctuations of atoms should not be observable directly, prevents the possibility of having isolated bodies. For this reason, the description of a physical system is given in terms of a matter field, which is the space-time distribution of atoms/molecules, coupled to the gauge field with the possible supplement of other fields describing the nonelectromagnetic interactions, such as the chemical forces. According to the principle of complementarity, there is also another representation where the phase assumes a precise value; this representation which focuses on the wave-like features of the system cannot be assumed simultaneously with the particle representation. The relation between these two representations is expressed by the uncertainty relation, similar to the Heisenberg relation between position and momentum:

$$(7) \quad \Delta N \Delta \Phi \geq 1/2$$

connecting the uncertainty of the number of quanta (particle structure of the system)  $\Delta N$  and the uncertainty of the phase (which describes the rhythm of fluctuation of the system)  $\Delta \Phi$ . Consequently, the two representations we have introduced above correspond to the two extreme cases.

- (A) If  $\Delta N = 0$ , the number of quanta is well defined, so that we obtain an atomistic description of the system, but lose the information on its capability to fluctuate, since  $\Delta \Phi$  becomes infinite. This choice corresponds to the usual, classic description of objects in terms of the component atoms/molecules.
- (B) If  $\Delta \Phi = 0$ , the phase is well defined, so that we obtain a description of the movement of the system, but lose the information on its particle-like features which become undefined since  $\Delta N$  becomes infinite. Such a system having a well-defined phase is termed coherent in the physical jargon.

In the phase representation, the deepest quantum features appear since the system becomes able to oscillate with a well-defined phase only when the number of its components becomes undefined, so that it is an open system and able to couple its own fluctuations to the fluctuations of the surroundings. In other words, such a coherent system, like a biological one, is able to "feel" the environment which is immersed within, through the e.m. potential created by its phase dynamics. In conclusion, a coherent system involves two kinds of interaction:

- (1) an interaction similar to that considered by Classical Physics, where objects interact by exchanging energy. These exchanges are connected with the appearance of forces. Since energy cannot travel faster than light, this interaction obeys the principle of causality;
- (2) an interaction where a common phase arises among different objects because of their coupling to the quantum fluctuations and hence to an e.m. potential. In this case there is no propagation of matter and/or energy taking place, and the components of the system "talk" to each other through the modulations of the phase field travelling at the phase velocity, which has no upper limit and can be larger than  $c$ , the speed of light.

The process of the emergence of coherent structures out of a crowd of independent component particles has been investigated in the last decades and is presently quite well understood [3, 24]. The presence of this field has received experimental corroboration by the discovery of the so-called "Lamb shift," named after the Nobel prize winner Lamb [16]. He discovered as far back as in 1947 that the energy level of the electron orbiting around the proton in the hydrogen atom is slightly shifted (about one part per million) with respect to the value estimated when assuming that no e.m. field is present. Further corroboration for the existence of vacuum fluctuations is provided by the Casimir effect [18]. Therefore a weak e.m. field is always present, just the one arising from the vacuum quantum fluctuations.

### 3. MESOSCALE STRUCTURE EMERGENT DYNAMICS

We should now pay attention to the fundamental relative dependency of the characteristic scales present in the problem we are dealing with. An atom has a size of about 1 Angstrom ( $\text{\AA}$ ) which amounts to  $10^{-8}$  cm, whereas a typical excitation energy is in the order of some electron volts (eVs), corresponding to a wavelength of the associated e.m. fluctuation in the order of some thousand  $\text{\AA}$ . This means that the tool (the e.m. fluctuation) able to induce a change of configuration in the atom is some thousands of times wider than the atom itself. Hence a single quantum fluctuation can simultaneously involve many atoms. In the case, for instance, of the water vapor at boiling temperature and normal pressure, the exciting e.m. mode (in this case 12 eV) would include in its volume about 20,000 molecules. Let us assume now that in the volume  $V = \lambda^3$  of the fluctuation there are  $N$  atoms. Let  $P$  be the probability (calculated by using 'Lamb shift'-like phenomena [16]) that an isolated atom is excited by an e.m. quantum fluctuation. Therefore the probability  $P_N$  that one out of the  $N$  atoms gets excited by the fluctuation is given by:

$$(8) \quad P_N = PN = P\lambda^3(N/V) = P\lambda^3 d,$$

where  $d$  is the density of atoms. We can see that there is a critical density  $d_{crit}$  such that  $P_N = 1$ , which means that the fluctuation excites with certainty one atom. In such conditions, the virtual photon coming out from the vacuum is "handed over" from one atom to another and gets permanently entrapped within the ensemble of atoms, being busy in keeping always at least one atom excited. According to this dynamics atoms acquire an oscillatory movement between their two configurations. In a short time, many quantum fluctuations pile up in the ensemble, producing eventually a large field which keeps all atoms oscillating between their two configurations. Moreover, the field gets self-trapped in the ensemble of atoms since its frequency becomes smaller; actually the period of oscillation  $T$  of the free field should be extended by adding the time spent within the excited atoms. Like in the cavity of a laser, the field becomes coherent, that is, acquires a well-defined phase, in tune with the oscillations of the atoms, which therefore become coherent, too. The more realistic case of atoms having a plurality of excited states has been also successfully addressed and needs a more sophisticated mathematics [20]. Among all the excited levels, the one selected for giving rise to the coherent oscillation is the level requiring the smallest time to self-produce a cavity. The region becomes a coherence domain (CD) whose size is the wavelength of the e.m. mode, where all atoms have tuned their individual fluctuations to each other and to the oscillation of the trapped field [4]. The size of the coherence domain cannot be arbitrary but is determined in a selfconsistent way by the dynamics underlying the emergence of coherence via the wavelength of the involved e.m. mode. A coherent system is therefore an ensemble of self-determined e.m. cavities. The fact that a biological system appears to be a nested ensemble of cavities within cavities of different sizes (organs, tissues, cells, organelles, etc.) having well-defined sizes is a strong indication for its coherence. In a CD there is a common phase, specific of the CD, which is therefore an object governed by a dynamics which eliminates the independence of the individual components and creates a unitarily correlated behavior of all of them, governed by the e.m. field. A peculiar

feature appears in the case of water. The coherent oscillation of the water molecules, which induces the formation of the CDs, occurs between the molecule's ground state and an excited state at 12.06 eV, which is slightly below the ionization threshold at 12.60 eV. The electron cloud of the water molecule oscillates between a configuration where all electrons are tightly bound (in this configuration water is an insulator and a mild chemical oxidant, since it is able to bind an extra electron) and a configuration where one electron is almost free (in this configuration water becomes a semiconductor and a chemical reducer, since it is able to release electrons). In conclusion, liquid water (which contributes about 70 % of the total mass and 99 % of the total number of component molecules of a living organism) exhibits a twofold inner dynamics [4]. This feature confirms the proposal of Schrodinger [21] about the need of negative entropy (negentropy) for the appearance of order in living systems. The theoretical framework outlined above has increasingly received support by a growing body of evidence. First of all, one should realize that the QFT picture satisfies the two main requirements demanded by biological evidence: the existence of selective recognition and attraction among biomolecules (organic codes) and long-range connections among biocomponents which cannot be accounted for by the very short-range interactions implied by a purely chemical dynamics. Secondly, "Science 1.0" researchers and scientists are completely unaware that QFT picture is already well present and hardwired in our current computational tools. This new awareness leads to our exploitation of more effective and competitive computational modeling tools by CICT.

#### 4. CICT QFT RESULTS

*CICT* is a natural framework for arbitrary multi-scale computer science and systems biology computational modeling in the current landscape of modern QFT [10, 9]. *CICT* new awareness of a discrete HG (hyperbolic geometry) subspace (reciprocal space) of coded heterogeneous hyperbolic structures [10], underlying the familiar  $\mathbb{Q}$  Euclidean (direct space) surface representation can open the way to holographic information geometry (HIG) [7, 9]. *CICT* founding principles are the same on which Riemannian manifold theories are founded, principles of relativity and covariance, of optimization (least action and geodesic principles), applied to scale and accuracy relativity transformations of the reference system in HG. *CICT* interprets natural rational "OpeRational" (OR, [7] for definition) representation as a language of languages of phased directed number systems quite easily. In fact, we can take the concepts of modular magnitude and direction as basic, and introduce the concept of vector as the basic kind of directed number, with an associated phasing relation. Directed numbers are defined implicitly by specifying rules for adding and multiplying vectors. Furthermore, they can be related uniquely to their remainder sequences to identify "quantum support field" sequences, which subspace inner phased generators can be computed from. *CICT* framework is quite flexible and can be used under a few major operational representations. For instance, *CICT* can see rational geometric series as simple recursion sequences in a wider recursive operative framework where all algebraic recursion sequences of any countable higher order include all the lower order ones and they can be optimally mapped to rational number system  $\mathbb{Q}$  operational representations and generating functions (OECS, Optimized Exponential Cyclic Sequences, [11]). For instance, arithmetic progression and Lucas sequences are recursion sequences of the second order. Lucas sequences are certain integer sequences that satisfy Lucas recurrence relation defined by polynomials  $U_n(P, Q)$  and  $V_n(P, Q)$ , where  $U_n, V_n$  are specific polynomials and  $P, Q$  are fixed integer coefficients. Any other sequence satisfying this recurrence relation can be represented as a linear combination of the Lucas sequences  $U_n(P, Q)$  and  $V_n(P, Q)$ . Famous examples of Lucas sequences include the Fibonacci numbers, Mersenne numbers, Pell numbers, Lucas numbers, Jacobsthal numbers, and a superset of Fermat numbers. *CICT* is able to fold any recursion sequence of the first order into one digit number  $D_1$ , any recursion sequence of second order into a two digit number  $D_2$ , any recursion sequence of the third order into a three digit number  $D_3$  and so on to higher orders. Then, you can interpret their asymptotic convergence ratios as

increasing accuracy approximations to related asymptotic roots from corresponding first, second, third, ...,  $n$ -th order equations respectively. *CICT* result can be presented even in term of classic or formal power series to show the close relationships to classic and modern control theory approaches for causal continuous-time and discrete-time linear systems. Increasing the subspace representation accuracy, the total number of allowed convergent paths, as monotonic power series, for instance (as allowed subspace paths), increases accordingly till maximum machine word length and beyond; like discrete quantum paths denser and denser to one another, towards a never ending "blending quantum continuum," called "quantum mixture" by a Top-Down (TD) perspective for composite multi-scale system. The finer geometry of subspace

$$\mathbf{1/(N-1)} \quad \longleftarrow \quad \mathbf{1/N} \quad \longrightarrow \quad \mathbf{1/(N+1)}$$

**Upscale Contiguity Operator**

$$\sum_{k=1}^{\infty} \left(\frac{1}{N}\right)^k = \frac{1}{N-1}, \quad k=1,2,3,\dots \in N$$

**Downscale Contiguity Operator**

$$\sum_{k=0}^{\infty} (-1)^k \left(\frac{1}{N}\right)^{(k+1)} = \frac{1}{N+1}, \quad k=0,1,2,3,\dots \in N$$

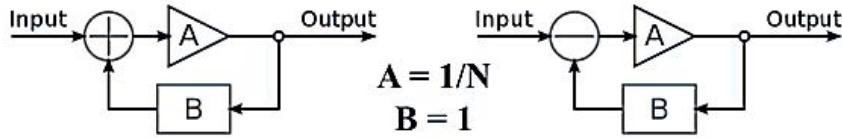


FIGURE 1. Egyptian fractions  $\mathbb{Q}$  subset as the discrete continuum of connected rationals (with no loss of generality) with two basic contiguity operators for LTR (downscale) and RTL (upscale) sequencing.

itself becomes scale dependent. While differentiable trajectories found in standard mathematical physics are automatically scale invariant, it is the main insight of the *CICT* theory that also certain non-differentiable paths (resultant paths, emerging from lower scales combined quantum trajectory interactions, which explicitly depend on the scale and accuracy of the observer) can be scale invariant.  $\mathbb{Q}$  can be thought as a discrete continuum of connected rationals (Fig.1) with two basic contiguity operators, LTR (downscale) and RTL (upscale). The first LTR *CICT* fundamental relationship that ties together numeric body information of LTR convergent monotonic power series in any base (in this case decimal, with no loss of generality) with  $D$  right ending by digit 9 is given by the following equation:

$$(9) \quad \frac{1}{D} = \sum_{k=0}^{\infty} \frac{1}{10^W} \left(\frac{\bar{D}}{10^W}\right)^k,$$

where  $\bar{D}$  is the additive  $10^W$  complement of  $D$ , i.e.  $\bar{D} = (10^W - D)$ ,  $W$  is the word representation precision length of the denominator  $D$  and "Div" means "Divergence of". When  $\bar{D} > D$  the formal power series of (9) can be re-scaled mod $D$ , to give multiple convergence paths to  $1/D$ , but with different "convergence speeds." The second LTR *CICT* fundamental relationship for LTR sequences relates power information to evolutive polynomially ordered representation structure counterpart exactly. For any base  $r$  and for any Natural number  $D$  it is given by the following equation [11]:

$$(10) \quad D_r^k + \bar{D}_r \cdot \left( \sum_{m=0}^{k-1} D_r^m \cdot r^{W \cdot (k-m-1)} \right)_k - r^{W \cdot k} = 0,$$

where  $\bar{D}_r$  is the additive  $r^W$  complement of  $D_r$ , i.e.  $\bar{D}_r = (r^W - D_r)$ . In previous paper [10], we already saw that *CICT* can supply us with co-domain OECSs perfectly tuned to their low-level

multiplicative noise source generators, related to experimental high-level overall perturbation [10]. Now, by (10), polynomial co-domain information functional closure can be used to evaluate any computed result at arbitrary scale, and to compensate conveniently, to achieve multi-scale computational information conservation by LTR sequences. Analogous relationships can be written for RTL power sequences [11]. Eventually, by comparing LTR and RTL sequences, we arrive to the general relationship that ties together numeric body information of divergent (RTL) and convergent (LTR) monotonic power series in any base (in this case decimal, with no loss of generality), with  $D$  ending by digit 9. It is given by the following *CICT* fundamental LTR-RTL correspondence equation:

$$(11) \quad \frac{1}{D} = \sum_{k=0}^{\infty} \frac{1}{10^W} \left( \frac{\bar{D}}{10^W} \right)^k = \frac{1}{\sum_{k=0}^{\infty} (D+1)^k},$$

with the usual meaning of symbols given for (9). Further generalizations of (11) related to  $D$  ending by digit 1 or 3 or 7 are straightforward [11]. Now, it is convenient to use a compact notation for LTR geometric series as follows:

$$(12) \quad \frac{1}{D} = \sum_{k=0}^{\infty} \frac{N}{10^W} \left( \frac{CR}{10^W} \right)^k \equiv N_W(CR_W),$$

where  $D$  is a denominator,  $N$  a numerator,  $CR$  the power series constant ratio, and  $W$  the length of their digit strings. Then, it is immediate to verify that at  $W = 1$  (precision =  $10^{-1}$ ), unity can emerge out of a fundamental symmetrical multiplicity of different countable paths:

$1(9) \equiv 1/1$ ,  $2(8) \equiv 1/1$ ,  $3(7) \equiv 1/1$ ,  $4(6) \equiv 1/1$ ,  $5(5) \equiv 1/1$ ,  $6(4) \equiv 1/1$ ,  $7(3) \equiv 1/1$ ,  $8(2) \equiv 1/1$ ,  $9(1) \equiv 1/1$ .

At  $W = 2$  (precision =  $10^{-2}$ ), we have:  $01(99) \equiv 1/1$ ,  $02(98) \equiv 1/1$ ,  $03(97) \equiv 1/1$ ,  $04(96) \equiv 1/1$ ,  $05(95) \equiv 1/1$ , ...,  $96(04) \equiv 1/1$ ,  $97(03) \equiv 1/1$ ,  $98(02) \equiv 1/1$ ,  $99(01) \equiv 1/1$ .

At  $W = 3$  (precision =  $10^{-3}$ ), we have:  $001(999) \equiv 1/1$ ,  $002(998) \equiv 1/1$ ,  $003(997) \equiv 1/1$ ,  $004(996) \equiv 1/1$ ,  $005(995) \equiv 1/1$ , ...,  $996(004) \equiv 1/1$ ,  $997(003) \equiv 1/1$ ,  $998(002) \equiv 1/1$ ,  $999(001) \equiv 1/1$ . Ad so on for  $W = 4, 5, \dots, W \in \mathbb{N}$ .

## 5. CONCLUSION

*CICT* approach combined to *GA* and *GC* unified mathematical language can offer an effective and convenient "Science 2.0" universal framework, by considering information not only on the statistical manifold of model states but also on the combinatorial manifold of low-level discrete, phased generators and empirical measures of noise sources, related to experimental high-level overall perturbation. Traditional elementary arithmetic long division remainder sequences can be interpreted as *OECS* for hyperbolic structures, as points on a discrete Riemannian manifold, under *HG* metric, indistinguishable from traditional random noise sources by classical Shannon entropy, and current most advanced instrumentation approach. *CICT* defines an arbitrary-scaling discrete Riemannian manifold uniquely, under *HG* metric, that, for arbitrary finite point accuracy  $W$  going to infinity (exact solution theoretically), is isomorphic (even better homeomorphic) to traditional information geometry Riemannian manifold. In other words, *HG* can describe a projective relativistic geometry directly hardwired into elementary arithmetic long division remainder sequences, offering many competitive computational advantages over traditional Euclidean approach. Thanks to its intrinsic self-scaling properties, this system approach can be applied at any system scale: from single quantum system application development to full system governance strategic assessment policies and beyond. This approach allows you even to develop more antifragile anticipatory learning system (*ALS*), for more reliable, safe and secure medical application and system (cybersafety). Specifically, high reliability organization (*HRO*), mission critical project (*MCP*) system, very low technological risk (*VLTR*) and crisis management (*CM*) system will be highly benefited mostly by these new techniques.

## REFERENCES

- [1] A. Aste, *Complex representation theory of the electromagnetic field*, Journal of Geometry and Symmetry in Physics, **28** (2012), 47–58.
- [2] M. Bischof, *Skalarwellen und Quantenfelder als mögliche Grundlage biologischer Information*, Erfahrungsheilkunde, **(47)5** (1998), 295–300.
- [3] M. Bischof, *Holism and field theories in biology—nonmolecular approaches and their relevance to biophysics*, in Biophotons, J. J. Chang, J. Fisch, and F. A. Popp, Eds., Kluwer Academic, Dordrecht, The Netherlands, 1998, pp. 375–394.
- [4] M. Bischof, E. Del Giudice, *Communication and the Emergence of Collective Behavior in Living Organisms: A Quantum Approach*, Molecular Biology International, **2013**, Article ID 987549, 19 pages, DOI: <http://dx.doi.org/10.1155/2013/987549>.
- [5] C. A. Bossetti, M. J. Birdno, W. M. Grill, *Analysis of the quasi-static approximation for calculating potentials generated by neural stimulation*, Journal of Neural Engineering, **(5)1** (2008), 44–53, doi: 10.1088/1741-2560/5/1/005.
- [6] E. Del Giudice, G. Vitiello, *Role of the electromagnetic field in the formation of domains in the process of symmetry breaking phase transitions*, Physical Review A, **(74)2** (2006), DOI: <http://dx.doi.org/10.1103/PhysRevA.74.022105>.
- [7] R. A. Fiorini, G. Laguteta, *Discrete Tomography Data Footprint Reduction by Information Conservation*, Fundamenta Informaticae, **(125)3-4** (2013), 261–272.
- [8] R. A. Fiorini, *The Entropy Conundrum: A Solution Proposal*, First International Electronic Conference on Entropy and Its Applications, Sciforum Electronic Conference Series, a011; doi:10.3390/ecea-1-a011, (2014), available at <http://sciforum.net/conference/ecea-1/paper/2649>.
- [9] R. A. Fiorini, *Computational Information Conservation Theory: An Introduction*, in 2014 Proceedings of the 8th International Conference on Applied Mathematics, Simulation, Modelling (ASM '14), N.E. Mastorakis, M. Demiralp, N. Mukhopadhyay, F. Mainardi, eds., Mathematics and Computers in Science and Engineering Series, No.34, NAUN Conferences, WSEAS Press, 2014, pp.385-394.
- [10] R. A. Fiorini, *How Random is Your Tomographic Noise? A Number Theoretic Transform (NTT) Approach*, Fundamenta Informaticae, **(135)3-4** (2014), 1–36.
- [11] R. A. Fiorini, *Computerized Tomography Noise Reduction by CICT Optimized Exponential Cyclic Sequences (OECS) Co-domain*, Fundamenta Informaticae, (2015), in press.
- [12] D. Hestenes, *Spacetime Algebra*, Gordon and Breach, New York, 1966.
- [13] D. Hestenes, *Proper Dynamics of a Rigid Point Particle*, J. of Math Physics, **15** (1974), 1778–1786.
- [14] D. Hestenes and G. Sobczyk, *Clifford Algebra to Geometric Calculus, a Unified Language for Mathematics and Physics*, G. Reidel Publ. Co., Dordrecht/Boston, 1984, paperback, 1985. Third printing, 1992.
- [15] S. Jain, R. Mitra, J. Wiart, *Full Wave Modeling of Brain Waves as Electromagnetic Waves*, Progress In Electromagnetics Research, **151** (2015), 95–107.
- [16] W. E. Lamb, R. C. Retherford, *Fine structure of the hydrogen atom by a microwave method*, Physical Review, **(72)3** (1947), 241–243.
- [17] K. Lashley, *The Mechanism of Vision*, Journal Press, Provincetown MA, 1948, pp. 302–306.
- [18] U. Mohideen and A. Roy, *Precision measurement of the Casimir force from 0.1 to 0.9  $\mu\text{m}$* , Physical Review Letters, **(81)21** (1998), 4549–4552.
- [19] C. Peratta, A. Peratta, *Dielectric properties of biological tissues*, Topics in Engineering, Modelling the Human Body Exposure to ELF Electric Fields, **47** (2010), 21–40.
- [20] G. Preparata, *QED Coherence in Matter*, World Scientific, Singapore, 1995.
- [21] E. Schroedinger, *What Is Life? The Physical Aspect of the Living Cell*, Cambridge University Press, Cambridge, UK, 1944.
- [22] N. N. Taleb, *Foiled by Randomness*, Random House, New York, 2005, 2nd Ed.
- [23] N. N. Taleb, *Silent Risk, Lectures on Probability*, Vol. 1, January 2015, available at <https://drive.google.com/file/d/0B8nhAlflk3QIR1o1dnk5ZmRaaGs/view?pli=1>.
- [24] E. M. Trukhan, V. N. Anosov, *Vector potential as a channel of informational effect on living objects*, Biofizika, **(52)2** (2007), 376–381.

# A FAMILY OF EMBEDDED COPROCESSORS WITH NATIVE GEOMETRIC ALGEBRA SUPPORT

**S. Franchini<sup>a</sup>, A. Gentile<sup>a</sup>, F. Sorbello<sup>a</sup>, G. Vassallo<sup>a</sup>, and S. Vitabile<sup>b</sup>**

<sup>a</sup> Computer Science and Artificial Intelligence Lab, DICGIM Department,  
University of Palermo, Palermo, Italy  
silvia.franchini@unipa.it [presenter, corresponding], antonio.gentile@unipa.it,  
filippo.sorbello@unipa.it, giorgio.vassallo@unipa.it

<sup>b</sup> Department of Biopathology and Medical Biotechnologies  
University of Palermo, Palermo, Italy  
salvatore.vitabile@unipa.it

ABSTRACT. Clifford Algebra or Geometric Algebra (GA) is a simple and intuitive way to model geometric objects and their transformations. Operating in high-dimensional vector spaces with significant computational costs, the practical use of GA requires, however, dedicated software and/or hardware architectures to directly support Clifford data types and operators. In this paper, a family of embedded coprocessors for the native execution of GA operations is presented. The paper shows the evolution of the coprocessor family focusing on the latest two architectures that offer direct hardware support to up to five-dimensional Clifford operations. The proposed coprocessors exploit hardware-oriented representations of GA elements and operators properly conceived to obtain fast performing implementations. The coprocessor prototypes, implemented on Field Programmable Gate Arrays (FPGA) development boards, show significant speedups of about one order of magnitude with respect to the baseline software library Gaigen running on a general-purpose processor. The paper also presents an execution analysis of different GA-based applications, namely inverse kinematics of a robot, optical motion capture, raytracing, and medical image processing, showing good speedups with respect to the baseline general-purpose implementation.

## 1. INTRODUCTION

Geometric Algebra (GA), also known as Clifford Algebra, is a powerful mathematical tool that allows for a simple and intuitive representation of geometric objects and their transformations [1]-[6]. The coordinate-free representation of GA is made possible by using extended objects in high-dimensional vector spaces to represent three-dimensional (3D) geometry. Four-dimensional (4D) and five-dimensional (5D) Clifford Algebras are used to implement the most powerful models of 3D geometry, namely the homogeneous model and the conformal model, respectively. Since the generic element of an  $n$ -dimensional GA has  $2^n$  coordinates, the computational complexity increases quickly when the algebra dimension increases, growing rapidly from 16 to 32 coefficients when moving from 4D to 5D, and requiring more than a thousand multiply-adds between coefficient combinations. These significant computational costs demand in turn dedicated software and/or hardware architectures to directly support Clifford data types and operators [7]-[19]. Software implementations of GA are conceived to execute Clifford operations on general-purpose processors using dedicated software libraries, such as Gaigen [7] and GluCat [8], packages for symbolic and numerical environments, such as the Clifford [9] and GA [10] packages for Maple, the Gable [11] and Clifford Multivector Toolbox [12] packages for Matlab, and the Grassmann algebra package [13] for Mathematica, or stand-alone programs, such as CluCalc [14]. Since faster performing solutions are needed, recently the attention has turned toward fully hardware



implementations or hardware-software codesigns. A combined software-hardware approach is proposed in [15]-[17] to accelerate GA-based algorithms. A pre-compiler, named Gaalop (Geometric Algebra Algorithm Optimizer), compiles GA algorithms to an intermediate representation converting GA operations in basic arithmetic operations [15]. This intermediate representation can be further compiled to run on different hardware platforms, such as Field Programmable Gate Arrays (FPGAs) or Graphics Processing Units (GPUs) [16]-[17]. A specialized coprocessor implemented on an Application Specific Integrated Circuit (ASIC) is presented in [18]. The coprocessor is specifically designed for color edge detection applications and supports only the 3D GA operations required in the target applications. The FPGA-based coprocessor prototype presented in [19] executes product operations for algebras of dimension up to 8 and uses 24-bit integer numbers to represent scalar coefficients of Clifford operands.

In this paper, a family of embedded coprocessors for the native execution of GA operations is presented, which we have been designing and implementing in the last few years [22]-[27]. The paper shows the evolution of the coprocessor family based on several factors, namely algebra dimension, supported operations, Clifford number representation, execution flow (sequential or parallel, scalar or pipeline), datapath width, data precision. First, the design space of computing architectures to natively execute GA operations has been explored taking into account different architectural parameters, such as number of multiply-add units and coefficient precision. Several alternative architectures, as resulted from the design space exploration, have been implemented and compared in terms of area cost, relative error, latencies, and speedup [25]. The resulting family of coprocessors offers direct hardware support to up to 5D GA operations. Different hardware-oriented representations of GA elements and operations have been properly conceived to obtain fast performing implementations. The representation based on the variable-length homogeneous elements, used for the first two architectures of the family [22]-[23], has been then replaced with a fixed-size representation based on quadruples in the latest coprocessors [24],[26],[27]. This novel representation of algebra elements allows for important simplifications of algebraic operations that in turn lead to a faster and more compact hardware architecture. The latest presented coprocessing architecture exploits a simplified formulation of the Conformal Geometric Algebra (CGA) operations, namely reflections, rotations, translations, and dilations, which results in faster execution of such operations [27]. The coprocessors have all been prototyped using development boards based on FPGA devices. The latest two coprocessors have been implemented as complete embedded Systems-on-Chip (SoCs) [26]-[27]. Experimental tests performed on the prototypes have shown significant speedups with respect to the baseline software library Gaigen running on a general-purpose processor [7]. The latest two coprocessors, named CliffordALU5 and ConformalALU, respectively, show native support of all GA operations in both 4D and 5D spaces with average speedups of about one order of magnitude over the baseline software implementation. To evaluate the coprocessor effectiveness in specific application domains, an application suite composed of different GA-based algorithms, namely inverse kinematics of a robot, optical motion capture, raytracing, and medical image processing, has been used as testbench, showing good speedups with respect to the baseline general-purpose implementation.

The rest of the paper is organized as follows: Section 2 outlines the design space exploration of GA-based computing architectures, while the resulting coprocessor family is presented in Section 3. Section 4 presents experimental results, while conclusions are contained in Section 5. A comprehensive introduction to the fundamental concepts of GA can be found in [1]-[6].

## 2. DESIGN SPACE EXPLORATION

The design space of hardware implementations that natively support GA operations has been analyzed along several axes, namely, Clifford number representation (homogeneous elements versus quadruples), execution flow (sequential versus parallel, scalar versus pipeline), number of multiply-add units, coefficient precision, datapath width, instruction word length. As described in Table 1, several alternative architectures based on different sets of architectural parameters have been explored and different design points have been implemented and prototyped. The resulting family of coprocessing architectures is detailed in the following section. One of the design parameters is related to the representation to be used for algebra elements. The standard variable-length representation, based on homogeneous elements, used for the first two architectures, has been replaced, in the latest coprocessors, with the fixed-size representation based on quadruples. A description of this representation is provided in Sections 3.3 and 3.4.

**Table 1. Design space exploration**

Coprocessor	GA operations	Clifford number representation	Execution flow	Parallelism techniques	N. of multipliers	Precision	Instruction word length
S-CliffoSor [22]	4D products, sums, differences	Homogeneous elements	Sequential	-	-	32-bit integers	15x32-bit words = 480 bits
CliffoSor [23]	4D products, sums, differences 3D rotations (Operations on bivectors not implemented)	Homogeneous elements	Parallel	-	24	32-bit integers	15x32-bit words = 480 bits
Quad-CliffoSor [24]	4D products, sums, differences, unary operations	Quadruples	Parallel	Pipeline	16	16-bit floating point	9x16-bit words = 144 bits
CliffordALU5 [26]	4D/5D products, sums, differences, unary operations	Quadruples	Parallel	Pipeline	16	32-bit floating point	9x32-bit words = 288 bits
ConformalALU [27]	5D conformal geometric operations (reflections, rotations, translations, dilations)	5D vectors	Parallel	Pipeline	5	32-bit floating point	16x32-bit words = 512 bits

## 3. COPROCESSOR FAMILY

In this section, the family of embedded coprocessors for the native support of GA operations is presented. A brief description of the implemented architectures, whose main features are listed in Table 1, is provided in the following subsections.

### 3.1 S-CliffoSor

The first coprocessor to be implemented was S-CliffoSor (Sliced Clifford coprocesSor), which offers direct hardware support to 4D Clifford operations between homogeneous elements using 32-bit integers numbers to represent scalar coefficients of basis blades. A homogeneous element or homogeneous multivector contains only blades of the same grade. Each 4D Clifford multivector is an ordered tuple of 5 homogeneous elements  $m = (s, v, b, t, p)$ , as listed in Table 2. Each homogeneous element is composed of a different number of blades, where each blade is a coefficient-basis blade pair. A 4-bit mask is associated with each basis blade, where each bit is associated with a basis vector  $e_i$ ,  $i=1,2,3,4$ , with  $e_1$  the least significant bit. S-CliffoSor is based on a 32-bit sequential ALU that executes addition, subtraction, multiplication and logical operations. Each 4D Clifford operation is decomposed into the proper sequence of these basic operations whose execution is supervised step-by-step by a microprogrammed control unit. The sequential architecture of the coprocessor leads to long per-operation latencies, which can be however hid by replicating the single S-CliffoSor slice to execute in parallel multiple independent Clifford operations.

**Table 2. 4D GA homogeneous elements**

Homogeneous element	Blades					
Scalar (s)	$a_0$					
Vector (v)	$a_1e_1$	$a_2e_2$	$a_3e_3$	$a_4e_4$		
Bivector (b)	$a_{12}e_1e_2$	$a_{13}e_1e_3$	$a_{14}e_1e_4$	$a_{23}e_2e_3$	$a_{24}e_2e_4$	$a_{34}e_3e_4$
Trivector (t)	$a_{123}e_1e_2e_3$	$a_{124}e_1e_2e_4$	$a_{134}e_1e_3e_4$	$a_{234}e_2e_3e_4$		
Pseudoscalar (p)	$a_{1234}e_1e_2e_3e_4$					

### 3.2 CliffoSor

As S-CliffoSor, the second developed architecture, namely CliffoSor (Clifford coprocesSor), executes 4D Clifford operations between homogeneous elements using 32-bit integer numbers to represent scalar coefficients. While S-CliffoSor is based on a sequential execution flow, CliffoSor uses parallel structures for the fastest execution of Clifford operations. Three different functional units directly support 4D Clifford products, 4D Clifford sums/differences and 3D Clifford rotations, respectively. To save resources, the most complex operations on bivectors are not directly supported in hardware, but they are handled in a higher-level Application Programming Interface (API) with multiple calls to the coprocessor. This design choice allowed us to use 24 parallel multipliers for Clifford products execution, rather than 36 parallel multipliers, as needed to directly support in hardware bivector-bivector products.

### 3.3 Quad-CliffoSor

As CliffoSor, Quad-CliffoSor (Quadruple-based Clifford coprocesSor) is a parallel architecture composed of three dedicated units for the execution of 4D Clifford products, 4D Clifford sums/differences, and 4D Clifford unary operations, respectively. While the previous architectures use the natural representation of GA elements based on homogeneous elements, Quad-CliffoSor first introduces a novel representation based on fixed-size elements, called quadruples. The variable size of homogeneous elements, as described in Table 2, leads to some storage inefficiencies since a six-element vector is used to implement each homogeneous multivector, but only the bivector uses all six elements. Defining the four-size elements, or quadruples, listed in Table 3, the generic 4D multivector can be written as a tuple of 4 quadruples,  $m = (V, T, S, P)$ . Each quadruple is composed of four blades. Using quadruples, rather than homogeneous elements, has two important advantages: first, fixed-size operands are better suited to a

hardware implementation; second, using the 4 quadruples listed in Table 3 allows for significant simplifications of product operations leading in turn to a faster and more compact hardware architecture. As demonstrated in [24], the product of two quadruples always gives as result the sum of two quadruples so that a single fixed format can be used for both input data and output result. Furthermore, a simplified algorithm can be used to execute product operations on quadruples. Any quadruple can be reduced to a quadruple of type V by simple sign changing operations on its coefficients. The product between any couple of quadruples can be therefore reduced to the product between two quadruples of type V by simple sign changing and/or swapping operations. The latter operation is the only operation to be implemented. The Quad-CliffoSor multiplier unit was properly designed to permit the fastest execution of this operation. 16 parallel multipliers are used to calculate the 16 products between the input quadruple coefficients, while a three-stage pipeline allows a product operation result to be provided on every clock cycle. Quad-CliffoSor is the first coprocessor of the family that uses 16-bit floating point numbers to represent scalar coefficients of the basis blades.

**Table 3. 4D GA quadruples**

Quadruple	Blades			
V	$a_1e_1$	$a_2e_2$	$a_3e_3$	$a_4e_4$
T	$a_{234}e_2e_3e_4$	$a_{134}e_1e_3e_4$	$a_{124}e_1e_2e_4$	$a_{123}e_1e_2e_3$
S	$a_{14}e_1e_4$	$a_{24}e_2e_4$	$a_{34}e_3e_4$	$a_0$
P	$a_{23}e_2e_3$	$a_{13}e_1e_3$	$a_{12}e_1e_2$	$a_{1234}e_1e_2e_3e_4$

### 3.4 CliffordALU5

CliffordALU5 is the first coprocessor that natively supports GA operations in the 5D space. The fixed-size representation based on quadruples, introduced in Quad-CliffoSor for 4D Clifford elements, is extended in CliffordALU5 to 5D Clifford elements. The eight quadruples of 5D GA are listed in Table 4.

**Table 4. 5D GA quadruples**

Quadruple	Blades			
V	$a_1e_1$	$a_2e_2$	$a_3e_3$	$a_4e_4$
T	$a_{234}e_2e_3e_4$	$a_{134}e_1e_3e_4$	$a_{124}e_1e_2e_4$	$a_{123}e_1e_2e_3$
S	$a_{14}e_1e_4$	$a_{24}e_2e_4$	$a_{34}e_3e_4$	$a_0$
P	$a_{23}e_2e_3$	$a_{13}e_1e_3$	$a_{12}e_1e_2$	$a_{1234}e_1e_2e_3e_4$
V'	$a_{15}e_1e_5$	$a_{25}e_2e_5$	$a_{35}e_3e_5$	$a_{45}e_4e_5$
T'	$a_{2345}e_2e_3e_4e_5$	$a_{1345}e_1e_3e_4e_5$	$a_{1245}e_1e_2e_4e_5$	$a_{1235}e_1e_2e_3e_5$
S'	$a_{145}e_1e_4e_5$	$a_{245}e_2e_4e_5$	$a_{345}e_3e_4e_5$	$a_5e_5$
P'	$a_{235}e_2e_3e_5$	$a_{135}e_1e_3e_5$	$a_{125}e_1e_2e_5$	$a_{12345}e_1e_2e_3e_4e_5$

As demonstrated in [26], the simplified algorithm used in Quad-CliffoSor for product operations execution can be extended to 5D operations. Since 4D quadruples are a subset of 5D quadruples, CliffordALU5 is a universal coprocessor that can directly execute all 4D and 5D GA operations (geometric products, outer products, left and right contractions, sums, differences, and unary operations) using quadruples as basic elements of computation. The block diagram of the CliffordALU5 coprocessor (Figure 1(a)) shows three dedicated functional units for the execution of 4D/5D Clifford products, 4D/5D Clifford sums/differences, and 4D/5D Clifford unary operations, respectively. The simplified algorithm used for product operations, described in Section 3.3 for the 4D case, allows for a compact architecture of the Clifford multiplier unit, as shown in Figure 1(b). This pipelined unit contains a 16x multiplier bank for the parallel execution of the 16 multiplications required by a product operation between quadruples.

CliffordALU5 is the first coprocessor of the family that uses 32-bit floating point numbers to represent scalar coefficients of Clifford operands.

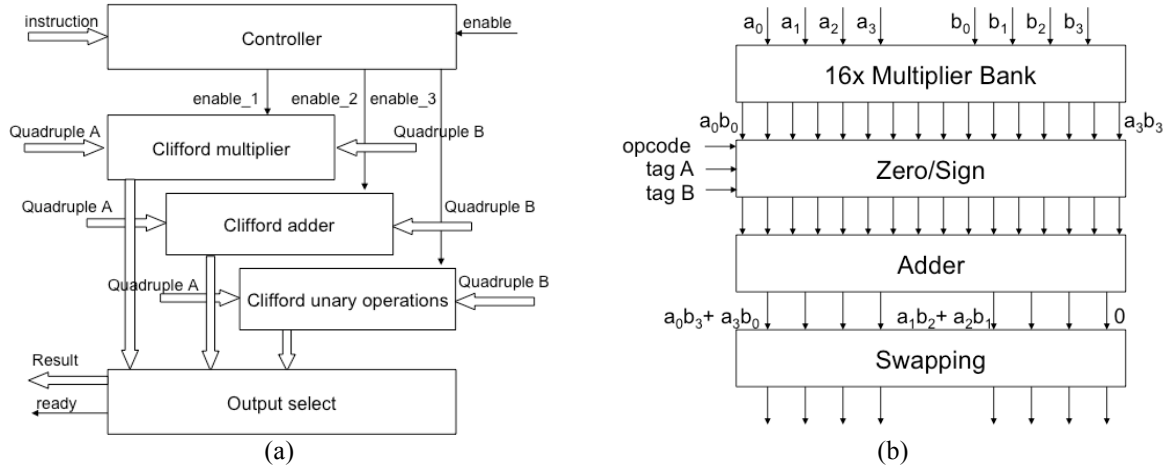


Figure 1. (a) CliffordALU5 block diagram; (b) Clifford multiplier unit block diagram

### 3.5 ConformalALU

The latest architecture, named ConformalALU, has been designed for the direct hardware support of Conformal Geometric Algebra (CGA) or 5D Geometric Algebra geometric operations, namely reflections, rotations, translations, and dilations. The coprocessor exploits a simplified formulation of these operations aimed at a parallel hardware implementation, which derives from two considerations. First, a conformal geometric operation on a generic  $k$ -blade  $A_k$ , represented in CGA by the “sandwich” geometric product, can be decomposed in operations on vectors according to the following formula:

$$XA_k \tilde{X} = X(a_1 \wedge a_2 \wedge \dots \wedge a_k) \tilde{X} = Xa_1 \tilde{X} \wedge Xa_2 \tilde{X} \wedge \dots \wedge Xa_k \tilde{X} \quad (1)$$

where  $X$  is the versor (rotor, translator, or dilator) that represents the conformal transformation. Second, rather than using the standard “sandwich” geometric product of CGA, each conformal geometric operation can be obtained by two non-commuting successive reflections. The basic operation becomes therefore the reflection of a 5D vector. In our implementation, each vector reflection is executed in turn using the following simplified formula based on the classical dot product, rather than the standard “sandwich” geometric product of CGA:

$$a' = a_{\perp} - a_{\parallel} = a - 2a_{\parallel} = a - 2|a_{\parallel}|m = a - 2(a \cdot m)m \quad (2)$$

where  $a$  is the vector to be reflected in a plane with unit-normal  $m$ , while  $a'$  is the reflected vector, as depicted in Figure 2.

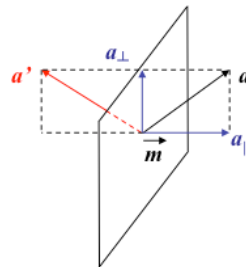


Figure 2. Reflection of a vector  $a$

Requiring only one dot product and two subtractions between vectors, this new formulation leads to a computational advantage and therefore a more compact and faster hardware architecture [27]. The block diagram of the ConformalALU coprocessor is depicted in Figure 3(a), while Figure 3(b) shows the block diagram of the Reflector unit. Two cascade Reflector units are used to execute a whole instruction stream in a pipeline fashion. As CliffordALU5, ConformalALU uses 32-bit floating point numbers to represent scalar coefficients.

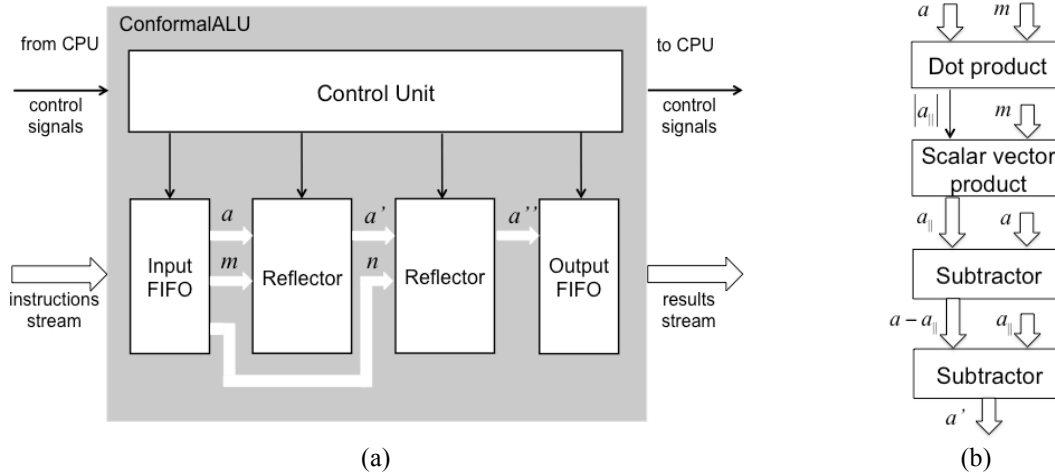


Figure 3. (a) ConformalALU block diagram; (b) Reflector unit block diagram

## 4. EXPERIMENTAL RESULTS

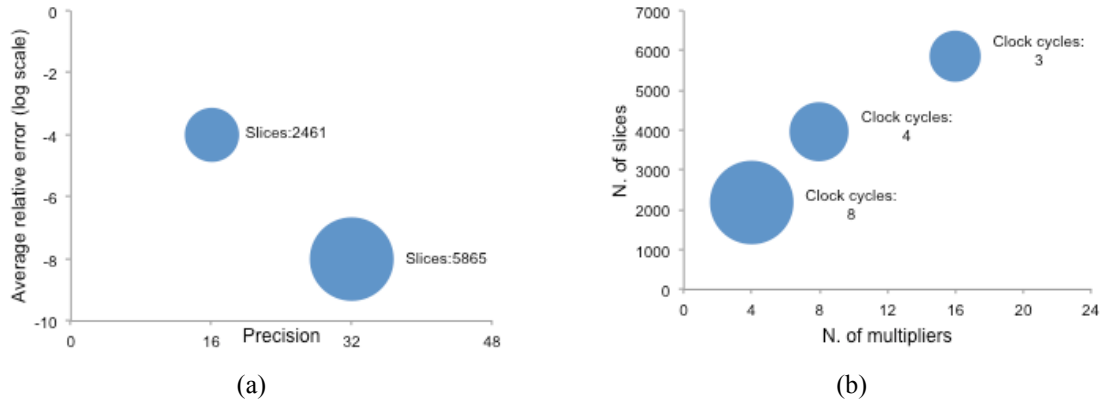
### 4.1 Coprocessor performance analysis

The designed coprocessors have all been prototyped using development boards based on FPGA devices. Several experimental tests have been performed to evaluate the coprocessor performance and compare it with the fast Gaigen software chosen as baseline general-purpose implementation. Table 5 shows the performance analysis of the coprocessor family in terms of clock frequency, area cost, latency per operation, and speedup. CliffoSor shows a higher area cost as well as reduced latencies per operation with respect to S-CliffoSor. These results depend on the different execution flow of the two architectures, namely, sequential for S-CliffoSor and parallel for CliffoSor. The reduced area cost, as well as the increased speedup, of Quad-CliffoSor in comparison with CliffoSor, is an effect of the computational and architectural simplifications of the quadruple-based representation. The higher area cost of CliffordALU5 when compared with Quad-CliffoSor, is due to the higher precision (32-bit rather than 16-bit) of the scalar coefficients. A scalar version and a pipelined version of the ConformalALU coprocessor have been designed. As reported in Table 5, the pipelined ConformalALU consumes more resources, but allows for reduced latency and, consequently, increased throughput. As a result of the design space exploration, Figure 4 presents a performance analysis, in terms of area cost, relative error, and latency, of different alternative architectures based on different sets of design parameters, such as the number of multiply-add units and the coefficient precision. Figure 4(a) shows average relative errors (with respect to the full-precision Gaigen implementation) and area costs of the multiplier units of Quad-CliffoSor and CliffordALU5, which use 16-bit and 32-bit precision, respectively. The higher-precision architecture consumes over two times more resources than the lower one, but a significant reduction of relative errors is observed. Three different versions of the CliffordALU5 coprocessor, which use 4, 8, and 16 parallel multipliers, respectively, for product operations execution are compared

in Figure 4(b) in terms of area costs and latencies per operation. Increasing the number of multipliers, the area cost increases, as well; however, a reduced latency in the product operation execution between quadruples can be observed.

**Table 5. Performance analysis of the coprocessor family**

Coprocessor	Clock frequency	Area (n. of FPGA slices)	Latency (clock cycles)	Speedup over Gaigen software
S-CliffoSor	45 MHz	2,295 (single slice)	(average) Products: 91 Sums/Diff.: 78	(potential) Products: 4x Sums/Diff.: 3x
CliffoSor	50 MHz	8,444	Products: 7 Sums/Diff.: 5	(potential) Products: 4x Sums/Diff.: 12x
Quad-CliffoSor	50 MHz	3,201	Products: 3 Sums/Diff.: 1	(potential) Products: 23x Sums/Diff.: 33x
CliffordALU5	100 MHz	6,011	Products: 3 Sums/Diff.: 1	(real) 4D Products: 5x 5D Products: 4x Sums/Diff.: 2x
ConformalALU	125 MHz	5,876 (scalar) 9,640 (pipelined)	(average) 315 (scalar) 88 (pipelined)	(real) Reflections: 56x Rotations: 15x Translations: 46x Dilations: 41x



**Figure 4. (a) Average relative error and area cost (number of FPGA slices) versus precision for Quad-CliffoSor and CliffordALU5; (b) Area cost and latency per operation (clock cycles) versus number of multipliers for CliffordALU5.**

To evaluate the speedups over the reference Gaigen software, the same test operations were executed using both the Gaigen library and the coprocessor. The first three coprocessors were prototyped on FPGA boards connected via the PCI bus or the Ethernet to the host computer. Only potential speedups in terms of clock cycles were estimated since the coprocessor ran on the FPGA using a clock frequency slower than the software running on the conventional host PC. Conversely, the latest two coprocessors were implemented as complete Systems on Chip (SoCs) using FPGA boards that integrate both a PowerPC general-purpose processor and the specialized coprocessor on the same chip. A real speedup, in terms of wall-clock times, has been therefore measured over the software library running on the PowerPC processor at the same operating frequency as the coprocessor. Gaigen/CliffordALU5 and Gaigen/ConformalALU comparisons are summarized in Figures 5(a) and 5(b), respectively. As reported in Table 5, CliffordALU5 achieves effective average speedups

of 5x for 4D Clifford products, 4x for 5D Clifford products, and 2x for 4D/5D Clifford sums, while effective speedups achieved by ConformalALU are 56x for reflection operations, 15x for rotations, 46x for translations, and 41x for dilations, respectively.

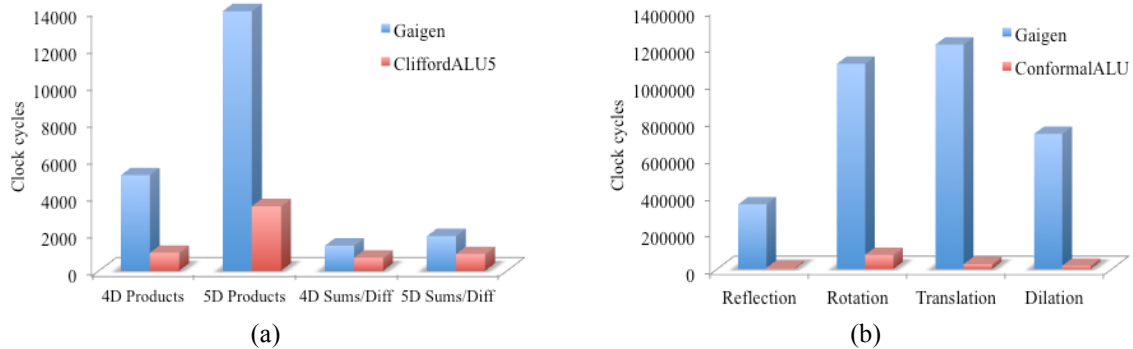


Figure 5. (a) Gaigen/CliffordALU5 comparison; (b) Gaigen/ConformalALU comparison

#### 4.2 Application suite

A suite of GA-based applications, including inverse kinematics of a robot, optical motion capture, raytracing, and medical image processing, has been used as testbench to evaluate the effectiveness of the coprocessor family in specific application domains. A description of these applications can be found in [26], [27]. The testbench algorithms have been executed using the latest two coprocessors, namely CliffordALU5 and ConformalALU, and their performance has been compared with the baseline general-purpose implementation based on the Gaigen software. Table 6 lists the observed speedups for each application. Taking into account the mix of Clifford operations required by each algorithm, the first three applications have been executed on the CliffordALU5 coprocessor, while the ConformalALU coprocessor has been used to accelerate medical image processing algorithms [20],[21]. The medical imaging applications, accelerated by the ConformalALU, massively use CGA operations (translations and rotations). The higher speedups of these applications are an effect of the simplified formulation of CGA operations that allows for faster execution of these operations.

Table 6. Observed speedups for the test applications

Application	Inverse kinematics	Motion capture	Raytracing	Medical image segmentation	Medical image registration
Observed speedup	3.4x	3.8x	4.8x	46x	43x

## 5. CONCLUSIONS

A family of embedded coprocessors that offer direct hardware support to GA operations has been presented in this paper. As overall result, the latest two coprocessors, namely CliffordALU5 and ConformalALU, natively execute all 4D and 5D GA operations showing speedups of about one order of magnitude relative to the baseline software implementation Gaigen. It has been observed that the novel simplified formulation of 5D CGA operations, used in ConformalALU, allows for a further speedup of about 10x with respect to the execution on the CliffordALU5 coprocessor. Future work will be aimed therefore to integrate the two coprocessors CliffordALU5 and ConformalALU in a single architecture to obtain a complete System-on-Chip that supports all basic operations of up to 5D GA (products, sums, unary operations) and accelerates geometric operations (reflections, rotations, translations, uniform scaling) of the 5D conformal model using the fast dedicated unit ConformalALU.



## REFERENCES

- [1] D. Hestenes, *New Foundations for Classical Mechanics*, Kluwer Academic, 1986.
- [2] D. Hestenes and G. Sobczyk, *Clifford Algebra to Geometric Calculus: A Unified Language for Mathematics and Physics*, Kluwer Academic, 1987.
- [3] L. Dorst and S. Mann, *Geometric algebra: A computational framework for geometrical applications (part 1: Algebra)*, in IEEE Comput. Graph. Appl., vol. 22, no. 3, pp. 24–31, May/June 2002.
- [4] L. Dorst and S. Mann, *Geometric algebra: A computational framework for geometrical applications (part 2: Applications)*, in IEEE Comput. Graph. Appl., vol. 22, no. 4, pp. 58–67, Jul./Aug. 2002.
- [5] J. Lasenby, W. J. Fitzgerald, C. J. L. Doran and A. N. Lasenby, *New Geometric Methods for Computer Vision*, in Int. J. Comp. Vision, vol. 36, no. 3, pp. 191–213, 1998.
- [6] D. Hestenes, Hongbo Li, A. Rockwood, *New Algebraic Tools for Classical Geometry*, in Geometric Computing with Clifford Algebras, G. Sommer (ed.), Springer Heidelberg, pp. 3–26, 2000.
- [7] D. Fontijn, *Gaigen 2: A geometric algebra implementation generator*, in Proc. 5th Int. Conf. Generative Programming and Component Eng., 2006, pp. 141–150.
- [8] P. Leopardi, The GluCat Home Page. [Online]. Available: <http://glucat.sourceforge.net/>.
- [9] R. Ablamowicz and B. Fauser, CLIFFORD - A Maple package for Clifford algebra computations. [Online]. Available: <http://math.tntech.edu/rafal/>.
- [10] M. Ashdown, GA package for Maple. Available: <http://www.mrao.cam.ac.uk/~maja1/software/GA/>.
- [11] S. Mann, L. Dorst, and T. Bouma, *The making of GABLE: A geometric algebra learning environment in Matlab*, in Geometric Algebra with Applications in Science and Engineering, E. Bayro-Corrochano and G. Sobczyk, Eds., New York, NY, USA, Springer, 2001, pp. 491–511.
- [12] E. Hitzer, S. Sangwine, Clifford Multivector Toolbox, A toolbox for computing with Clifford algebras in Matlab. [Online]. Available: <http://sourceforge.net/projects/clifford-multivector-toolbox/>.
- [13] J. Browne, The Grassmann Algebra Book Home Page. [Online]. Available: <http://www.grassmannalgebra.info/grassmannalgebra/>.
- [14] C. Perwass, The CLUCalc Home Page. [Online]. Available: <http://www.Clucalc.info>.
- [15] D. Hildenbrand, J. Pitt, and A. Koch, *Gaalop-high performance parallel computing based on conformal geometric algebra*, in Geometric Algebra Computing, Springer, 2010, pp. 477–494.
- [16] P. Charrier and D. Hildenbrand, *Geometric algebra computing technology for accelerated processing units*, presented at the Embedded World Conf., Nürnberg, Germany, 2013.
- [17] D. Hildenbrand, *Geometric algebra computers*, in Foundations of Geometric Algebra Computing, New York, NY, USA, Springer, 2013, pp. 179–188.
- [18] B. Mishra, P. Wilson, and R. Wilcock, *A Geometric Algebra Co-Processor for Color Edge Detection*, in Electronics 2015, 4(1), pp. 94–117.
- [19] C. Perwass, C. Gebken, G. Sommer, *Implementation of a Clifford algebra co-processor design on a field-programmable gate array*, in Clifford Algebras: Applications to Mathematics, Physics, and Engineering, Series: Progress in Mathematical Physics, vol. 34, R. Ablamowicz, Ed., Springer, 2004.
- [20] J. Rivera-Rovelo and E. Bayro-Corrochano, *Surface approximation using growing self-organizing nets and gradient information*, Appl. Bionics Biomech., vol. 4, no. 3, pp. 125–136, 2007.
- [21] E. Bayro-Corrochano and J. Rivera-Rovelo, *The use of geometric algebra for 3d modeling and registration of medical data*, J. Math. Imaging Vis., vol. 34, no. 1, pp. 48–60, May 2009.
- [22] S. Franchini, A. Gentile, M. Grimaudo, C.A. Hung, S. Impastato, F. Sorbello, G. Vassallo, and S. Vitabile, *A Sliced Coprocessor for Native Clifford Algebra Operations*, in Proceedings of the 10th IEEE Euromicro Conference on Digital System Design - Architectures, Methods and Tools (DSD 2007), Lübeck, Germany, August 29–31, 2007, pp. 436–439, IEEE Computer Society Press.
- [23] Silvia Franchini, Antonio Gentile, Filippo Sorbello, Giorgio Vassallo, and Salvatore Vitabile, *An Embedded, FPGA-based Computer Graphics Coprocessor with Native Geometric Algebra Support*, in Integration, The VLSI Journal, Volume 42, Issue 3, pp. 346–355, June 2009.
- [24] Silvia Franchini, Antonio Gentile, Filippo Sorbello, Giorgio Vassallo, and Salvatore Vitabile, *Fixed-size Quadruples for a New, Hardware-Oriented Representation of the 4D Clifford Algebra*, in Advances in Applied Clifford Algebras, Volume 21, Issue 2, pp. 315–340, June 2011.
- [25] Silvia Franchini, Antonio Gentile, Filippo Sorbello, Giorgio Vassallo, and Salvatore Vitabile, *Design Space Exploration of Parallel Embedded Architectures for Native Clifford Algebra Operations*, in IEEE Design and Test of Computers, Volume 29, Issue 3, pp. 60–69, May–June 2012.
- [26] Silvia Franchini, Antonio Gentile, Filippo Sorbello, Giorgio Vassallo, and Salvatore Vitabile, *Design and Implementation of an Embedded Coprocessor with Native Support for 5D, Quadruple-based Clifford Algebra*, in IEEE Transactions on Computers, Vol. 62, No. 12, pp. 2366–2381, Dec. 2013.
- [27] Silvia Franchini, Antonio Gentile, Filippo Sorbello, Giorgio Vassallo, and Salvatore Vitabile, *ConformalALU: a Conformal Geometric Algebra Coprocessor for Medical Image Processing*, in IEEE Transactions on Computers, Vol. 64, No. 4, pp. 955–970, April 2015.

# THE GEOMETRIC SOLUTION TO THE THREE-BODY PROBLEM AND ITS APPLICATION TO COLOUR IMAGE PROCESSING AND QUANTUM MECHANICS

Ramon González Calvet <sup>a</sup>

<sup>a</sup> Institut Pere Calders  
Campus Universitat Autònoma de Barcelona s/n  
08193 Cerdanyola del Vallès, Spain  
rgonzalezcalvet@gmail.com

ABSTRACT. Leibniz's theorem is the starting point to find a geometric solution to the so called *three-body problem*. According to this solution, the weighted quadratic form of absolute Cartesian coordinates of three particles can be written as a separated quadratic form of the centre-of-mass coordinate and relative coordinates. This geometric solution can be applied to colour spaces (in particular to *RGB* space) in order to define a weighted lightness, not equal for all colours, and two additional chroma coordinates, which adjusts the human eye cones sensitivity to different colours better than the 1/3 weights of the LHS model. These new colour coordinates can yield a faster algorithm for colour treatment and image data compression while preserving colour Grassmann's laws. The same change from absolute to relative coordinates allows us to write the weighted Laplacian operator with relative coordinates. Since the quantum non-relativistic Hamiltonian of a system of  $N$  free particles is proportional to a weighted Laplacian operator of their Cartesian coordinates, an exact expression for the kinetic Hamiltonian of three particles using relative coordinates is deduced. An application to some elemental systems (such as the electronic energy levels of helium atom and vibrational levels of carbon dioxide) will be shown.

## 1. INTRODUCTION

The systems of many particles are of high interest in statistical mechanics, quantum physics and celestial mechanics. Since Lagrange submitted his prize memoir *Essai sur le Problème des Trois Corps* in 1772, a lot of research on the three-body and the  $N$ -body problems has been carried out, especially related to celestial mechanics [1]. However, the  $N$ -body problem is not integrable in general as shown by Painlevé, and some techniques as Laplace's perturbation theory and Poincaré's topologic methodes have been developed to approach solutions ([2], p. 400). On the other hand, the dynamics of  $N$  interacting bodies is not always stable and can become chaotic even for the three-body system [3]. Some known solutions to the three-body problem have been masterly explained by Hestenes [2] and let us recall his treatment here. The dynamics of three bodies located at the points  $X_1$ ,  $X_2$  and  $X_3$  under their gravitational attraction can be written through the vector equations:

$$(1) \quad \begin{aligned} \ddot{X}_1 &= -m_2 \frac{X_1 - X_2}{|X_1 - X_2|^3} - m_3 \frac{X_1 - X_3}{|X_1 - X_3|^3} & \ddot{X}_2 &= -m_3 \frac{X_2 - X_3}{|X_2 - X_3|^3} - m_1 \frac{X_2 - X_1}{|X_2 - X_1|^3} \\ \ddot{X}_3 &= -m_1 \frac{X_3 - X_1}{|X_3 - X_1|^3} - m_2 \frac{X_3 - X_2}{|X_3 - X_2|^3} \end{aligned}$$

where the gravitational constant has been absorbed into the definition of mass. Then, after the introduction of the relative coordinates:

$$(2) \quad S_1 = X_3 - X_2 \quad S_2 = X_1 - X_3 \quad S_3 = X_2 - X_1$$

he gets the symmetrical form of the dynamical equations<sup>1</sup>:

<sup>1</sup>According to Hestenes, the equations (3) were found by Broucke and Lass in 1973.

$$(3) \quad \ddot{S}_1 = -m \frac{S_1}{|S_1|^3} + m_1 G \quad \ddot{S}_2 = -m \frac{S_2}{|S_2|^3} + m_2 G \quad \ddot{S}_3 = -m \frac{S_3}{|S_3|^3} + m_3 G$$

where  $m = m_1 + m_2 + m_3$  and  $G$  is:

$$(4) \quad G = \frac{S_1}{|S_1|^3} + \frac{S_2}{|S_2|^3} + \frac{S_3}{|S_3|^3}$$

From this point he deduces the known solutions to the three-body problem.

My interest for the  $N$ -body problem came from Hylleraas' treatment of the helium atom [5, 6], which seems unsatisfactory and suggests that the three-body problem should be reviewed, at least from a geometrical point of view. The question is which is the kinetic energy of a system of  $N$  particles expressed with relative coordinates, and which is the corresponding Hamiltonian operator of quantum mechanics also expressed with relative coordinates. It will be shown that the symmetry of the relative coordinates (2) outlined by Hestenes is the key to find a geometric solution to the three-body problem.

## 2. ANTECEDENTS

Leibniz's theorem [7] states that if  $P$  is any point in the plane and  $ABC$  a triangle whose centroid is  $G$  then the following identity is fulfilled:

$$(5) \quad PA^2 + PB^2 + PC^2 = \frac{1}{3} (AB^2 + BC^2 + CA^2) + 3PG^2$$

An alternative and equivalent version of this theorem states that the geometric locus of the points  $P$  such that  $PA^2 + PB^2 + PC^2 = k^2$  is a circle with centre  $G$  and radius:

$$(6) \quad r = |GP| = \sqrt{\frac{k^2}{3} - \frac{AB^2 + BC^2 + CA^2}{9}}$$

Apollonius' circle theorem states that the locus of the points  $P$  whose distances to two fixed points  $A$  and  $B$  are in a ratio  $k$  is a generalized circle with centre  $O$  and radius  $r$ :

$$(7) \quad |PA| = k |PB| \quad \Rightarrow \quad O = \frac{A - k^2 B}{1 - k^2} \quad r = |OP| = \frac{k |AB|}{1 - k^2}$$

For  $k = 1$  the radius is infinite and the locus is the bisector of  $AB$ . Note that:

$$(8) \quad |PA| = k |PB| \quad \Leftrightarrow \quad PA^2 - k^2 PB^2 = 0$$

which is a special case of the geometric outlining of the two-body problem: *Given two points  $A$  and  $B$ , which is the geometric locus of the points  $P$  such that  $a PA^2 + b PB^2 = k^2$ ?* The solution is known long time ago: The points  $P$  lie on a circle centred at the centre of mass  $G$ :

$$(9) \quad G = \frac{aA + bB}{a + b} \quad |GP| = \sqrt{\frac{k^2}{a + b} - \frac{ab}{(a + b)^2} AB^2}$$

This geometric solution is equivalent to writing:

$$(10) \quad a PA^2 + b PB^2 = (a + b) PG^2 + \frac{ab}{a + b} AB^2$$

Defining the reduced mass of the binary system as  $\mu = ab/(a + b)$  and denoting for  $m = a + b$  the total mass of the system, we have the usual notation in physics:

$$(11) \quad a PA^2 + b PB^2 = m PG^2 + \mu AB^2$$

In ref. [11] we find the following *Satz von Leibniz*: If  $a + b + c = 1$ ,  $G = aA + bB + cC$  and  $X$  is any point then:

$$(12) \quad aXA^2 + bXB^2 + cXC^2 - XG^2 = aGA^2 + bGB^2 + cGC^2$$

whose *Korollar C* is:

$$(13) \quad abAB^2 + bcBC^2 + caCA^2 = \rho^2 - OG^2$$

where  $O$  is the circumcentre and  $\rho$  is the radius of the circumscribed circle. Finally, in [12] the *scalar function of Leibniz*  $f(P)$  is defined as:

$$(14) \quad f(P) = \sum_{i=1}^k a_i PA_i^2$$

and assuming that  $\sum a_i \neq 0$  then there exists a fixed vector  $v$  such that, for any point  $P'$  the following equality is fulfilled:

$$(15) \quad f(P') = f(P) + 2PP' \cdot v$$

### 3. GEOMETRIC SOLUTION TO THE THREE-BODY PROBLEM

The classical three-body problem is formulated as follows: *How can the kinetic energy of a system of three particles be written with relative velocities?* The kinetic energy is a separable quadratic form of the velocities of each particle. Since the derivative with respect to the time is a linear operator, this problem is equivalent to writing  $aA^2 + bB^2 + cC^2$  with relative coordinates, and concretely, to asking whether an expression like (10) can exist for three points  $A$ ,  $B$  and  $C$  of a triangle.

**Theorem 3.1** (Apollonius' lost theorem [8, 9]). *Given a triangle  $ABC$ , the geometric locus of the points  $P$  such that  $aPA^2 + bPB^2 + cPC^2 = k^2$  is a circle centred at the centre of mass  $G = aA + bB + cC$ .*

*Proof.* Without loss of generality, it can be taken  $a + b + c = 1$  in order to remove denominators. After developing  $PG^2 = [P - (aA + bB + cC)]^2$  by means of the scalar product and gathering terms one arrives at [9]:

$$(16) \quad PG^2 = aPA^2 + bPB^2 + cPC^2 - abAB^2 + bcBC^2 + caCA^2$$

whence  $|PG|$  is constant:

$$(17) \quad |PG| = \sqrt{k^2 - (abAB^2 - bcBC^2 - caCA^2)}$$

and therefore  $P$  lies on a circle centred at  $G$ . □

*Remark 3.2.* If  $a + b + c = m$  instead of 1 then (16) becomes:

$$(18) \quad aPA^2 + bPB^2 + cPC^2 = mPG^2 + \frac{ab}{m}AB^2 + \frac{bc}{m}BC^2 + \frac{ca}{m}CA^2$$

*Remark 3.3.* If  $a$ ,  $b$  and  $c$  are the masses of the particles centred at the points  $A$ ,  $B$  and  $C$ , then  $m$  is the mass of the system of three particles,  $G = (aA + bB + cC)/m$  is its centre of mass and  $P$  can be taken as the origin of coordinates. This is just the geometric solution to the three-body problem, a generalization of the solution to two-body problem (10).

*Remark 3.4.* The decomposition of the kinetic energy of a system of three particles is obtained by derivation of (18) with respect to the time, and has an analogous expression.

*Remark 3.5.* The equation (18) although deduced for the plane is also valide for every  $N$ -dimensional Euclidean space because of the addition of squares.

#### 4. APPLICATION TO COLOUR IMAGE PROCESSING

The sensitivity of the human eye to the light power follows a non-linear power law (*gamma law*). When an electric voltage is applied to a CRT (cathodic ray tube) monitor, the light power emitted by the electrons colliding with the screen phosphor also fits a non-linear law. Fortunately, both fuctions are almost coincident, so that the applied voltage can be considered proportional to the light sensitivity of human eye [13]. From now on I will refer to lightness perceived by the human eye, whcih is quite proportional to applied voltages.

*RGB* (red, green, blue) space is the most frequently used to describe the colour space, although there is a small part of colours perceived by the human eye falling out the *RGB* space. The main question to deal with is the fact that the human eye has different sensitivity to the three colours, from green the highest to blue the lowest. This was taken into account when defining the *YUV* coordinates of the colour space used in the PAL TV format. The *Y* (*luma*) colour coordinate was defined as:

$$(19) \quad Y = w_R R + w_G G + w_B B \quad RGB \in [0, 1]^3$$

$R = G = B = 0$  is black,  $R = G = B = 1$  is white, and grey colour is the addition of equal amounts of the three fundamental colours *RGB*. *Luma* was defined with the weights  $w_R = 0.299$ ,  $w_G = 0.587$  and  $w_B = 0.114$  [14] fulfilling  $w_R + w_G + w_B = 1$ , and it gives an approach to the lightness of colours. On the other hand, the chrominance components *U* and *V* were defined as:

$$(20) \quad U = U_{max} \frac{B - Y}{1 - w_B} \approx 0.492(B - Y) \quad U_{max} = 0.436$$

$$(21) \quad V = V_{max} \frac{R - Y}{1 - w_R} \approx 0.877(B - Y) \quad V_{max} = 0.615$$

In the space *YPbPr* the chromatic components are:

$$(22) \quad P_B = 0.5 \frac{B - Y}{1 - w_B} \quad P_R = 0.5 \frac{R - Y}{1 - w_R}$$

Its digital version *YCbCr* uses  $RGB \in [0, 255]^3$  and then  $Y \in [16, 235]$  and  $C_B, C_R \in [16, 240]$ . The JPEG image format enlarges these ranges to  $Y \in [0, 255]$  and  $C_B, C_R \in [0, 255]$ .

From the solution to the three-body problem, I suggest using a new chrominance components corresponding to relative coordinates defined in the following way:

$$(23) \quad J = R - G \in [-1, 1] \quad K = G - B \in [-1, 1] \quad I = B - R \in [-1, 1]$$

which are linear dependent since:

$$(24) \quad I + J + K = 0$$

Of course two chrominance components are enough together with luma *Y* (19) for defining a colour, and since green is the lightest colour of the three fundamental colours it seems most suitable to take only *J* and *K*. Then I will call this colour system as *YJK* whose conversion matrix from *RGB* is:

$$(25) \quad \begin{pmatrix} Y \\ J \\ K \end{pmatrix} = \begin{pmatrix} w_R & w_G & w_B \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

If we consider the three fundamental colours as the coordinates of three masses lying on the segment  $[0, 1]$ , then luma  $Y$  is the centre of mass and  $I, J$  and  $K$  are the relative coordinates, the distances between colours. From the three-body solution (18), it seems natural to define a colour metric (colour norm)  $\|c\|$  given by the scalar function of Leibniz [12]:

$$(26) \quad \|c\|^2 = \|RGB\|^2 = w_R R^2 + w_G G^2 + w_B B^2$$

Note that if  $R = G = B$  then the colour is grey and the colour norm is the grey scale. Therefore (26) is the definition of grey level for every colour. This grey scale definition takes into account the different lightness of the fundamental colours and is much better than the rough definition of lightness as  $(R + G + B)/3$  in the LHS model [15]. By means of (18) the norm  $\|c\|$  (the grey level) can be obtained from the  $YJK$  coordinates as:

$$(27) \quad \|c\|^2 = \|YJK\|^2 = Y^2 + w_R w_B I^2 + w_R w_G J^2 + w_G w_B K^2$$

Moreover, this norm defines clearly a distance between two colours  $c_1$  and  $c_2$  in the colour space:

$$(28) \quad \begin{aligned} d(c_1, c_2) &= \|c_2 - c_1\| = \sqrt{w_R (R_2 - R_1)^2 + w_G (G_2 - G_1)^2 + w_B (B_2 - B_1)^2} \\ &= \sqrt{(Y_2 - Y_1)^2 + w_R w_B (I_2 - I_1)^2 + w_R w_G (J_2 - J_1)^2 + w_G w_B (K_2 - K_1)^2} \end{aligned}$$

Some properties of this distance are the following:

- a) The distance from black to white is 1.
- b) The distance from opposite fundamental colours are also 1. For instance yellow= $(1, 1, 0)_{RGB}$  and blue= $(0, 0, 1)_{RGB}$  then  $d(\text{yellow}, \text{blue})=1$ . In the same way the distance from green to magenta and from cyan to red is 1.
- c) It fulfils the triangular inequality  $d(c_1, c_3) \leq d(c_1, c_2) + d(c_2, c_3)$ .

The chrominance components of the colour coordinates  $YJK$  could be plotted in a plane with Cartesian coordinates  $J$  and  $K$  as done usually for the  $U$  and  $V$  coordinates. However, if we wish to preserve the symmetry of the  $I, J$  and  $K$  chrominance components another graph plot must be used. Let us consider three unitary vectors  $e_I, e_J$  and  $e_K$  forming  $120^\circ$  between them and defining the corresponding coordinate axes:

$$(29) \quad e_I = -\frac{\sqrt{3}}{2}e_1 - \frac{1}{2}e_2 \quad e_J = \frac{\sqrt{3}}{2}e_1 - \frac{1}{2}e_2 \quad e_K = e_2$$

Then the coordinates  $IJK$  of any point  $P(x e_1 + y e_2)$  in this plane are its orthogonal projections over the axes, which are obtained through the scalar product:

$$(30) \quad \begin{aligned} I &= P \cdot e_I = (x e_1 + y e_2) \cdot \left(-\frac{\sqrt{3}}{2}e_1 - \frac{1}{2}e_2\right) = -\frac{\sqrt{3}}{2}x - \frac{1}{2}y \\ J &= P \cdot e_J = (x e_1 + y e_2) \cdot \left(\frac{\sqrt{3}}{2}e_1 - \frac{1}{2}e_2\right) = \frac{\sqrt{3}}{2}x - \frac{1}{2}y \\ K &= P \cdot e_K = (x e_1 + y e_2) \cdot e_2 = y \end{aligned}$$

The components  $IJK$  so obtained fulfil (24). Figure 1 shows the coordinate plot. Since  $-1 \leq I, J, K \leq 1$ , all the possible values of  $I, J$  and  $K$  lie on a hexagon. A point in the plane is obtained from  $IJK$  as:

$$(31) \quad x e_1 + y e_2 = \frac{2}{3}(I e_I + J e_J + K e_K)$$

The main advantages of the  $YJK$  codification is the fact that the chrominance components  $J$  and  $K$  are only obtained from a subtraction, which is a fast operation at the CPU level. It is also easy to implement an electric circuit to encode and to decode the  $YJK$  signal. On the other hand,

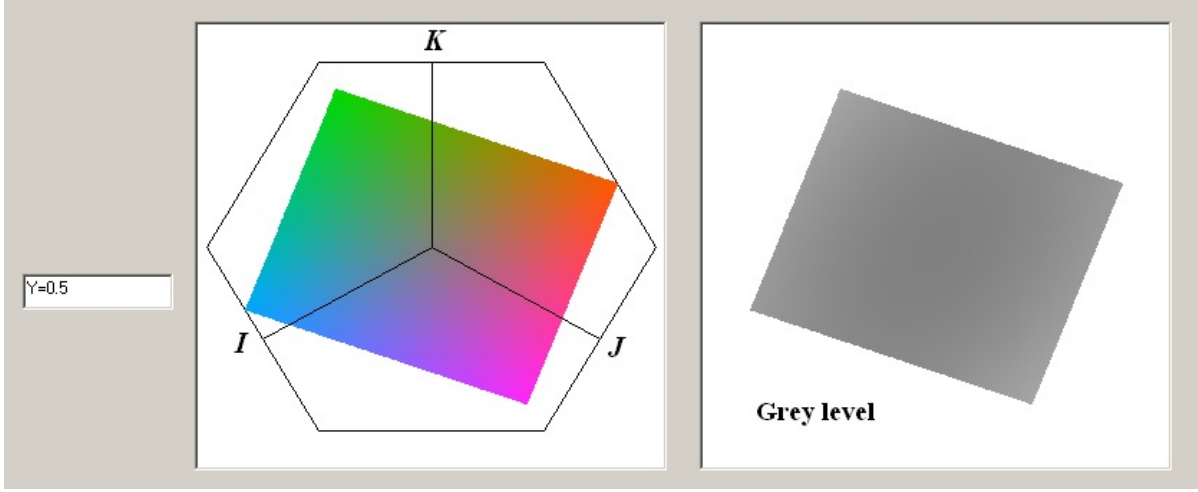


FIGURE 1. Left box: plot of  $IJK$  coordinates for all the possible values of  $RGB$  calculated for luma  $Y = 0.5$ . Right box: grey level of the same colours calculated according to (26).

the  $J$  and  $K$  signals can be applied directly as a voltage difference between adjacent pixels with different colours in a screen, which reduces to  $1/3$  the arithmetic operations to decode video signal. Since  $YJK$  have been defined as the linear combinations (25) of  $RGB$ ,  $YJK$  also fulfil Grassmann's laws of additive colour mixtures [16].

## 5. APPLICATION TO CLASSICAL MECHANICS

The geometric solution to the three-body problem (18) has immediate application to the study of the dynamics of a system of three particles because it implies the decomposition of the kinetic energy. If we call the relative coordinates as:

$$(32) \quad D = B - A \quad E = C - B \quad F = A - C$$

then (18) can be written as:

$$(33) \quad a A^2 + b B^2 + c C^2 = m G^2 + \frac{a b}{m} D^2 + \frac{b c}{m} E^2 + \frac{c a}{m} F^2$$

whence the Lagrange function  $\mathcal{L}$  of a system of three bodies interacting gravitationally and having masses  $a$ ,  $b$  and  $c$  is obtained:

$$(34) \quad \mathcal{L} = \frac{m}{2} \dot{G}^2 + \frac{a b}{2m} \dot{D}^2 + \frac{b c}{2m} \dot{E}^2 + \frac{c a}{2m} \dot{F}^2 + k \left( \frac{a b}{|D|} + \frac{b c}{|E|} + \frac{c a}{|F|} \right) + \vec{\lambda} \cdot (D + E + F)$$

$k$  is the gravitational constant, and the holonomic constraint [17]  $D + E + F = 0$  has been added by multiplication for a Lagrange multiplier vector  $\vec{\lambda}$ . The application of Lagrange equations  $d\mathcal{L}/dq_i = d(d\mathcal{L}/d\dot{q}_i)/dt$  leads to the same equation as (3):

$$(35) \quad \ddot{D} = -\frac{k m}{|D|^3} D + \frac{m}{a b} \vec{\lambda}$$

where now we see that Hestenes'  $G$  in (3) is in fact a Lagrange multiplier proportional to  $\vec{\lambda}$ . The linear dependence of relative coordinates can be always introduced in Lagrange equations of motion via Lagrange multipliers. However, this does not mean that a system of  $N$  interacting particles be integrable, which restricts the usefulness of the geometric solution to the  $N$  body problem and its applicability to studying classical dynamics. Nevertheless, the panorama is different in quantum mechanics.

## 6. APPLICATION TO QUANTUM MECHANICS

The molecular spectra (IR <sup>2</sup>, C<sup>13</sup>-NMR and H-NMR<sup>3</sup>) seem to indicate that some kind of solution for quantum systems of many particles should exist. For instance, each functional group (such as carboxylic acid, ketone, amide, alcohol, anhydride, nitril, etc.) has a characteristic absorption frequency in the IR spectra that is shifted in some degree by the other neighbouring atoms and bonds of the molecule. Also the main NMR frequency of H or C<sup>13</sup> is slightly shifted by the neighbouring atoms of the molecule. When outlining Schrödinger's equation for a system of many particles, the potential energy is usually known as a function of relative coordinates  $V(x_{ij})$ . However, the kinetic energy operator is expressed with absolute coordinates  $x_i$ , and the exchange to relative coordinates  $x_{ij} = x_j - x_i$  were not done before except for the two-particle system or some special cases of more particles. Up to now, Jacobi coordinates  $\xi_i$ :

$$(36) \quad \xi_i = \frac{\sum_{k=1}^j m_k x_k}{\sum_{k=1}^j m_k} - x_{j+1}$$

are likely the more consistent approach to write the Hamiltonian operator with relative coordinates, because they fulfil [18]

$$(37) \quad \sum_{k=1}^N \frac{1}{m_k} \frac{\partial^2 \psi}{\partial x_k^2} = \frac{1}{M} \frac{\partial^2 \psi}{\partial \xi_N^2} + \sum_{j=1}^{N-1} \frac{1}{\mu_j} \frac{\partial^2 \psi}{\partial \xi_j^2}$$

where:

$$(38) \quad \frac{1}{\mu_j} = \frac{1}{M_j} + \frac{1}{m_{j+1}} \quad M_j = \sum_{k=1}^j m_k$$

However, they do not follow the principle of symmetry between relative coordinates  $x_{ij}$ . Therefore, a general expression of the Hamiltonian expressed with relative coordinates was still needed. In this way, let us see the following theorem:

**Theorem 6.1** (Internal Laplacian theorem [10]). *The Laplacian of three particles located at the points A, B and C on a line with weights a, b and c fulfils the following identity for the change from absolute to relative coordinates:*

$$(39) \quad \Delta = \frac{1}{a} \frac{\partial^2}{\partial A^2} + \frac{1}{b} \frac{\partial^2}{\partial B^2} + \frac{1}{c} \frac{\partial^2}{\partial C^2} = \frac{1}{m} \frac{\partial^2}{\partial G^2} + \frac{a+b}{a b} \frac{\partial^2}{\partial D^2} + \frac{b+c}{b c} \frac{\partial^2}{\partial E^2} + \frac{a+c}{a c} \frac{\partial^2}{\partial F^2} \\ - \frac{2}{a} \frac{\partial^2}{\partial D \partial F} - \frac{2}{b} \frac{\partial^2}{\partial D \partial E} - \frac{2}{c} \frac{\partial^2}{\partial E \partial F}$$

*Proof.* By taking into account (32) and  $G = (a A + b B + c C)/m$ , and applying the properties of partial derivatives. For more detailed steps see [10].  $\square$

**Corollary 6.2.** *The internal three-dimensional Laplacian (without the centre-of-mass coordinate) of three particles is:*

$$(40) \quad \Delta_{int} = \frac{a+b}{a b} \nabla_D^2 + \frac{b+c}{b c} \nabla_E^2 + \frac{a+c}{a c} \nabla_F^2 - \frac{2}{a} \nabla_D \cdot \nabla_F - \frac{2}{b} \nabla_D \cdot \nabla_E - \frac{2}{c} \nabla_E \cdot \nabla_F$$

where D, E and F are their relative coordinates, and  $\nabla_D$  has its usual meaning:

$$(41) \quad \nabla_D = e_1 \frac{\partial}{\partial x_D} + e_2 \frac{\partial}{\partial y_D} + e_3 \frac{\partial}{\partial z_D}$$

<sup>2</sup>Infrared spectroscopy.

<sup>3</sup>Nuclear Magnetic Resonance spectroscopy of nuclei with non null spin.



*Proof.* By addition of the three equations (39) obtained for each Cartesian coordinate.  $\square$

The kinetic energy operator  $\hat{T}$  of a system of three particles is proportional to the weighted Laplacian (39) [19]:

$$(42) \quad \hat{T} = -\frac{\hbar^2}{2} \left( \frac{1}{a} \frac{\partial^2}{\partial A^2} + \frac{1}{b} \frac{\partial^2}{\partial B^2} + \frac{1}{c} \frac{\partial^2}{\partial C^2} \right)$$

On the other hand, the first term on the *rhs* of (39) accounts for the kinetic energy of the centre of mass because:

$$(43) \quad \hat{T}_{MC} = -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial G^2}$$

and is excluded when calculating the internal energy levels of the molecules<sup>4</sup> because the kinetic energy of molecule translations is already taken into account in statistical mechanics via the translational partition function [20]. Therefore, the internal kinetic energy operator is proportional to the internal Laplacian (40):

$$(44) \quad \hat{T}_{int} = -\frac{\hbar^2}{2} \Delta_{int}$$

**6.1. Computation of the vibrational levels of CO<sub>2</sub>.** Since  $D$ ,  $E$  and  $F$  are linearly dependent, we can choose without loss of generality a wave function  $\psi(D, E)$  only dependent on two relative coordinates. In this case the internal Laplacian is:

$$(45) \quad \Delta_{int} \psi(D, E) = \frac{1}{\mu_{ab}} \frac{\partial^2 \psi}{\partial D^2} + \frac{1}{\mu_{bc}} \frac{\partial^2 \psi}{\partial E^2} - \frac{2}{b} \frac{\partial^2 \psi}{\partial D \cdot \partial E}$$

The first time when I applied this Laplacian was in order to calculate the vibrational levels of carbon dioxide. The CO<sub>2</sub> molecule is linear and has a central carbon atom bonded to two adjacent oxygen atoms. There are different stable isotopes of carbon (<sup>12</sup>C, <sup>13</sup>C) and oxygen (<sup>16</sup>O, <sup>17</sup>O, <sup>18</sup>O), but the very most frequent molecule (98,4 %) is <sup>12</sup>C<sup>16</sup>O<sub>2</sub>. If the mass of <sup>12</sup>C is taken as 12, then the mass of <sup>16</sup>O is 15.9949. I approached the C=O bond vibration as a harmonic oscillator, which is a well-known and solved quantum system. In this way, Schrödinger's equation of the CO<sub>2</sub> linear vibration is:

$$(46) \quad -\frac{\hbar^2}{2} \left( \frac{1}{\mu_{CO}} \frac{\partial^2 \psi}{\partial x_1^2} + \frac{1}{\mu_{CO}} \frac{\partial^2 \psi}{\partial x_2^2} - \frac{2}{m_C} \frac{\partial^2 \psi}{\partial x_1 \partial x_2} \right) + \frac{k_{CO} x_1^2}{2} \psi + \frac{k_{CO} x_2^2}{2} \psi = E \psi$$

where  $x_1$  and  $x_2$  indicate the increases of the lengths of both CO bonds with respect to the equilibrium length,  $k$  indicates the force constant (in the harmonic oscillator approach) of the corresponding bond, and  $\mu_{CO} = m_C m_O / (m_C + m_O)$  indicates the reduced mass of carbon and oxygen atoms as used in physics. The former equation can be written as:

$$(47) \quad \frac{\partial^2 \psi}{\partial x_1^2} + \frac{\partial^2 \psi}{\partial x_2^2} - \frac{2\mu_{CO}}{m_C} \frac{\partial^2 \psi}{\partial x_1 \partial x_2} - \frac{\mu_{CO} k_{CO} x_1^2}{\hbar^2} \psi - \frac{\mu_{CO} k_{CO} x_2^2}{\hbar^2} \psi = -\frac{2E \mu_{CO}}{\hbar^2} \psi$$

Schrödinger's equation of the harmonic oscillator approach for a single bond is:

$$(48) \quad \frac{\partial^2 \psi}{\partial x^2} - \frac{\mu_{CO} k_{CO} x^2}{\hbar^2} \psi = -\frac{2E \mu_{CO}}{\hbar^2} \psi$$

<sup>4</sup>The centre-of-mass and internal Hamiltonians are separable because they depend on different coordinates. In this case, the wave function is a product of two wavefunctions, one translational times another internal, and the total energy is the translational plus the internal energies. The wave function of a free particle travelling in a box is a well known and solved system, and the real problem is reduced to the computation of the internal energy.

Introducing the vibrational frequency of the bond  $\nu_{\text{CO}} = \frac{1}{2\pi} \sqrt{\frac{k_{\text{CO}}}{\mu_{\text{CO}}}}$  we have:

$$(49) \quad \frac{\partial^2 \psi}{\partial x^2} - \frac{\pi^2 \mu_{\text{CO}}^2 \nu_{\text{CO}}^2 x^2}{\hbar^2} \psi = -\frac{2E \mu_{\text{CO}}}{\hbar^2} \psi$$

In order to compute the energy levels of the linear vibrations, a basis of 64 functions obtained from the multiplication of pairs of 8 eigenstates of each CO bond fulfilling (49) was used. The computation is automatically split into two separate computations, one for even states and the other for odd states. The wave functions of even states are linear combinations of products of pairs of harmonic oscillator eigenstates whose quantum number addition is even (for instance 2+0, 1+1, 0+2, 3+1, ...), and for odd states the addition of quantum numbers is odd (0+1, 1+0, 3+0, 2+1, ...). Both sets of eigenfunctions do not interact between them and the computation is carried out separately for even and odd states. The results of these computations were included in table 1.

Energy levels of $^{12}\text{C}^{16}\text{O}_2$		
Level	Energy / $h\nu_{\text{CO}}$	
	Even states	Odd states
0	0.954110 <sup>g</sup>	1.608763 <sup>g</sup>
1	2.263418 <sup>g</sup>	2.207676 <sup>u</sup>
2	2.862330 <sup>g,u</sup>	2.918076 <sup>g</sup>
3	3.461243 <sup>g</sup>	3.516985 <sup>u</sup>
4	3.572743 <sup>g</sup>	4.115897 <sup>g</sup>
5	4.171644 <sup>g,u</sup>	4.227920 <sup>g</sup>
6	4.770552 <sup>g</sup>	4.714809 <sup>u</sup>
7	4.883310 <sup>g</sup>	4.826533 <sup>u</sup>

TABLE 1. Energy levels of  $^{12}\text{C}^{16}\text{O}_2$  for the linear vibrational motion computed by means of the approach of harmonic oscillator potential energy for each CO bond. Energy given as a multiple of the vibrational frequency of the CO bond. *g* and *u* indicate symmetric and antisymmetric states under interchange of both oxygens. Since both  $^{16}\text{O}$  are bosons, the vibrational wave function must be symmetric and *u* states cannot exist for  $^{12}\text{C}^{16}\text{O}_2$ .

As indicated above, the 98,4 % of carbon dioxide is  $^{12}\text{C}^{16}\text{O}_2$ . Since both oxygens have spin 0 and are bosons, the wave function remains invariant under their exchange. This means that it only has symmetric vibrational states (indicated as *g* from the German word *gerade*). The odd states (indicated as *u*, *ungerade*) will only exist for minor amounts of other isotopic combinations. On the other hand, according to Boltzman's distribution law, the most populated energy level at room temperature is the ground state with a factor  $\exp(\Delta E_{1 \leftarrow 0}/k_B T)$  with respect to the first excited state. Therefore, the visible bands in the IR (infrared) spectra are transitions from the ground state. The most intense IR transitions take place when the electromagnetic radiation generates a change in the electric dipole moment of the molecule [21, 29], although other transitions changing the magnetic dipole moment or the electric quadropole moment can also be IR active but with much lower intensity. On the other hand, electric dipole transitions forbidden by the selection rule of the harmonic oscillator ( $\Delta v = \pm 1$ ) can appear due to anharmonicity<sup>5</sup> [23]. Table 2 has been built from table 1 and indicates IR transitions, which includes those from

<sup>5</sup>The potential curve does not fit exactly to a quadratic function and Morse potential, which is not symmetric with respect to the minimum, is usually applied to calculate the vibrational energy levels of diatomic molecules [25]

the ground state (absorption bands) and also IR emission bands of CO<sub>2</sub> laser. IR absorption and emission bands are composed by a lot of single roto-vibrational transitions (caused by the thermal population of the different rotational levels). The centres of these IR bands are the pure vibrational frequencies table 1 and 2 refer to.

### Infrared transitions of <sup>12</sup>C<sup>16</sup>O<sub>2</sub>

Transition	Frequencies / $\nu_{\text{CO}}$	Band wavenumber / $\text{cm}^{-1}$		Band strength [28]
		Calculated for $\bar{\nu}_{\text{CO}} = 1794 \text{ cm}^{-1}$	Experimental	
$1^{e:g} \leftarrow 0^{e:g}$	1.309308	2348,9	2349.1 [22, 26]	955900
$2^{e:g} \leftarrow 0^{e:g}$	1.908220	3423.3	3612.8 [22, 26]	10400
$2^{o:g} \leftarrow 0^{e:g}$	1.963966	3523.4	3714.8 [22, 26]	15800
$3^{e:g} \leftarrow 0^{e:g}$	2.507133	4497.8	4853.6 [26]	77.8
$4^{e:g} \leftarrow 0^{e:g}$	2.618633	4697.8	5099.7 [26]	109
$4^{o:g} \leftarrow 0^{e:g}$	3.161787	5672.2	6076.0 [26]	0.523
$5^{e:g} \leftarrow 0^{e:g}$	3.217534	5772.3	6227.9 [26]	4.61
$5^{o:g} \leftarrow 0^{e:g}$	3.273810	5873.2	6347.8 [26]	4.58
$1^{e:g} \leftarrow 2^{e:g}$	0.598912	1074.5	961.34* [27]	
$1^{e:g} \leftarrow 2^{o:g}$	0.654658	1174.5	1064.12* [27]	

\*CO<sub>2</sub> laser emission bands.

TABLE 2. IR transitions for the linear vibrational motion of <sup>12</sup>C<sup>16</sup>O<sub>2</sub> by means of the approach of harmonic oscillator potential energy for each bond. Calculated frequencies obtained from table 1.

The fundamental IR absorption band <sup>6</sup> of CO<sub>2</sub> is observed at 2349  $\text{cm}^{-1}$  [21] and a pair of two minor bands are observed at 3613 and 3715  $\text{cm}^{-1}$  [22]. If we assign 2349  $\text{cm}^{-1}$  to the  $1^{e:g} \leftarrow 0^{e:g}$  transition, we obtain as a parameter  $\nu_{\text{CO}} = 1794 \text{ cm}^{-1}$ . This value is comparable to the observed vibration frequencies of the C=O bond in organic molecules <sup>7</sup> going from 1700 to 1850  $\text{cm}^{-1}$  [24]. Then, according to table 2, the transition  $2^{e:g} \leftarrow 0^{e:g}$  should have a frequency of 3423  $\text{cm}^{-1}$ , close to the experimental band at 3612.8  $\text{cm}^{-1}$  and the transition  $2^{o:g} \leftarrow 0^{e:g}$  should have 3523,4  $\text{cm}^{-1}$  while the experimental value is 3714.8  $\text{cm}^{-1}$ . The discordance is easily explained by anharmonicity, that is, the real potential curve is not exactly a quadratic function. In spite of this, the wavenumber differences between both bands of 100.1  $\text{cm}^{-1}$  (calculated) and 102  $\text{cm}^{-1}$  are very similar, which is a very good prediction. On the other hand, the parameter  $\nu_{\text{CO}} = 1794 \text{ cm}^{-1}$  allows us to calculate the relative population of the vibrational levels at room temperature (298.15K):  $N(1^{o:g})/N(0^{e:g}) = 3.44 \cdot 10^{-3}$ . Therefore, the main observed bands are transitions from the ground state. On the other hand, the Raman transition at 1333  $\text{cm}^{-1}$ , usually attributed to the symmetric stretching of CO<sub>2</sub> has been misunderstood, since this transition does not appear from the energy levels of linear vibrations. In fact, it is easy to see that it corresponds to the transition  $2 \leftarrow 0$  of bending, because  $667 \times 2 = 1334$ . The method of analysis using the normal modes of vibration leads to uncertainties in the assignment of the IR and Raman bands for molecules with as few atoms as CO<sub>2</sub> [30], and it should be reviewed

<sup>6</sup>Another strong band is observed at 667  $\text{cm}^{-1}$  generated by bending the CO<sub>2</sub> molecule and is not considered in these computations.

<sup>7</sup>The range of variation of the vibrational frequency (wavenumber) of the C=O bond in organic molecules goes from 1700 to 1850  $\text{cm}^{-1}$  depending on the concrete functional group: ketones 1705-1725  $\text{cm}^{-1}$ , aldehydes 1720-1740  $\text{cm}^{-1}$ , carboxylic acids 1700-1725  $\text{cm}^{-1}$ , esters 1730-1750  $\text{cm}^{-1}$ , anhydrides 1800-1850  $\text{cm}^{-1}$ , amides 1630-1700  $\text{cm}^{-1}$ .

under the scope of the *ab initio* computation of the vibrational energy levels and wave functions by means of the internal Hamiltonian here outlined.

**6.2. Electronic energy levels of the helium atom.** Since the helium atom is a system formed by one nucleus and two electrons, the internal Laplacian (40) was also applied to check Hylleraas' calculation of the electronic energy levels of the helium atom [5]. However, there is not enough space here to explain it, and therefore it will be developed during the AGACSE 2015 conference.

## 7. CONCLUSIONS

The geometric solution to the three-body problem consists of a change from absolute coordinates of weighted points to centre-of-mass and relative coordinates of the system. This solution has applications to the colour space and to quantum systems, although applications to Newtonian dynamics of classical systems are limited by the linear dependence of relative coordinates. According to this geometric solution, in the colour space a new set of chromatic coordinates  $I$ ,  $J$ ,  $K$  are defined as subtraction of pairs of the  $RGB$  coordinates, which is a very simple arithmetic operation with a low charge for a computer CPU. On the other hand, a new metric is defined in the colour space giving different weights (according to the different sensitivity of human eye) to the three fundamental colours. This metric allows us to define the norm of a colour, which is identified with its grey level, and the distance between two colours. In order to apply the geometric solution to the three-body problem to quantum systems, the change from absolute to centre-of-mass and relative coordinates is introduced into the weighted Laplacian. In this way, the internal Hamiltonian is obtained and applied to the computation of the vibrational energy levels of  $\text{CO}_2$ . The results are consistent with experimental data, but they differ from the analysis based on the method of normal modes of vibration, which seems to have a lot of contradictions. The study of the helium atom, another quantum system of three-particles, will be shown during the AGACSE 2015 conference.

## REFERENCES

- [1] F. R. Moulton, *An Introduction to Celestial Mechanics*, 2nd rev. ed., p. 319, Dover, 1970.
- [2] D. Hestenes, *New Foundations for Classical Mechanics*, 2nd ed., pp. 398-418, Kluwer, 2002.
- [3] A. F. Rañada, "Movimiento caótico", *Investigación y Ciencia*, march 1996, p.12.
- [4] N. H. March, W. H. Young, S. Sampanthar, *The Many-Body Problem in Quantum Mechanics*, p. 1, Dover, 1995.
- [5] I. N. Levine, *Química Cuántica*, p. 221, ed. AC, Madrid, 1977.
- [6] E. A. Hylleraas, *Z. Phys.* **48** (1929) 469.
- [7] I. Shariguin, *Problemas de geometría. Planimetría*, p. 95, ed. Mir, 1986.
- [8] R. González Calvet, *Tractat de geometria plana mitjançant l'àlgebra geomètrica*, p. 57, Cerdanyola del Vallès, 1996.
- [9] R. González Calvet, *Treatise of Plane Geometry through Geometric Algebra*, pp. 78 and 248, Cerdanyola del Vallès, 2007.
- [10] *Ibidem*, pp. 79 and 255.
- [11] M. Koecher, A. Krieg *Ebene Geometry*, 2nd ed., p. 163, Springer, 2000.
- [12] M. Audin, *Geometry*, Universitext, p. 59, Springer, 2003.
- [13] Ch. Poynton, *Frequently Asked Questions about Color*, <http://www.poynton.com/GammaFAQ.html>.
- [14] Ch. Poynton, *Frequently Asked Questions about Color*, <http://www.poynton.com/PDFs/ColorFAQ.pdf>, p.7.
- [15] H. Levkowitz, *Color Theory and Modeling for Computer Graphics, Visualization and Multimedia Applications*, p. 53, Kluwer, 1997.
- [16] A. Ford, A. Roberts, *Colour Space Conversions*, p. 8, <http://www.poynton.com/PDFs/coloureq.pdf>, 1998.
- [17] K. R. Symon, *Mecánica*, p. 377, Aguilar, 1968.
- [18] W. Grainer, *Quantum Mechanics. An Introduction*, 3rd ed., p. 221-224, Springer, 1994.
- [19] L. Pauling, E. Jr. Bright Wilson, *Introduction to Quantum Mechanics with Applications to Chemistry*, p. 86, Dover, 1985.
- [20] T. Hill, *An Introduction to Statistical Thermodynamics*, p. 74, Dover, 1986.
- [21] A. Requena, J. Zúñiga, *Espectroscopía*, p. 235, Pearson/Prentice Hall, 2004.

- [22] N. L. Alpert, W. E. Keiser, H. A. Szymanski, *Theory and Practice of Infrared Spectroscopy*, 2nd ed., Fig. 4-33C, Plenum Press, 1970.
- [23] I. N. Levine, *Espectroscopía Molecular*, ed. AC, Madrid, p. 239, 1980.
- [24] N. L. Allinger, M. P. Cava, D. C. de Jongh, C. R. Johnson, N. A. Lebel, C. L. Stevens, *Química Orgánica* vol. I, p. 299, Reverté, 1983.
- [25] D. C. Harris, M. D. Bertolucci, *Symmetry and Spectroscopy. An introduction to Vibrational and Electronic Spectroscopy*, p. 106, Dover, 1978.
- [26] R. B. Wattson, L. S. Rothman, "Determination of Vibrational Energy Levels and Parallel Band Intensities by Direct Numerical Diagonalization", *J. Mol. Spectrosc.* **119** (1986) 23-100.
- [27] W. J. Witteman, *The CO<sub>2</sub> Laser*, p. 24, Springer-Verlag, 1987.
- [28] L. S. Rothman, "Infrared energy levels and intensities of carbon dioxide", *Applied Optics* **25** (1986) 1795-1816.
- [29] Ll. H. Jones, *Inorganic Vibrational Spectroscopy*, vol. 1, p. 9., Marcel Dekker Inc., 1971.
- [30] Ll. H. Jones, *ibidem* p. 53.

# GEOMETRIC ALGEBRA COMPUTING FOR HETEROGENEOUS SYSTEMS

Dietmar Hildenbrand<sup>a</sup>, Justin Albert and Patrick Charrier<sup>b</sup>

<sup>a</sup> dietmar.hildenbrand@gmail.com

<sup>b</sup> University of Technology, Darmstadt, Germany  
justin.albert@gmx.de, charrier@gsc.tu-darmstadt.de

ABSTRACT. The use of Geometric Algebra in engineering applications relies heavily on the availability of software solutions for the new heterogeneous computing architectures. While most of the Geometric Algebra tools are restricted to CPU focused programming languages, the Geometric Algebra **algorithms optimizer** Gaalop supports also FPGA and GPU programming languages. In this paper, we introduce the Gaalop precompiler for heterogeneous systems (CPUs, GPUs, FPGAs, DSPs ...) based on the programming language C++ AMP (Accelerated Massive Parallelism) of the HSA (Heterogeneous System Architecture) Foundation. As a proof-of-concept we present a raytracing application. Starting from the situation 15 years ago with a great gap between the low symbolic complexity on the one hand and the high numeric complexity of coding in GA on the other hand, this paper shows, that, in the meantime, this gap could be closed.

## 1. INTRODUCTION

In 2000, Gerald Sommer stated in the preface of his book [17]:

*Today we have to accept a great gap between the low symbolic complexity on the one hand and the high numeric complexity of coding in GA on the other hand. Because available computers cannot even process complex numbers directly, we have to pay a high computational cost at times, when using GA libraries, ... , full profit in real-time applications is only possible with adequate processors.*

What kind of processors do we need for Geometric Algebra 15 years later? Now we have a world of parallel heterogenous systems and the question is how suitable they are for Geometric Algebra.

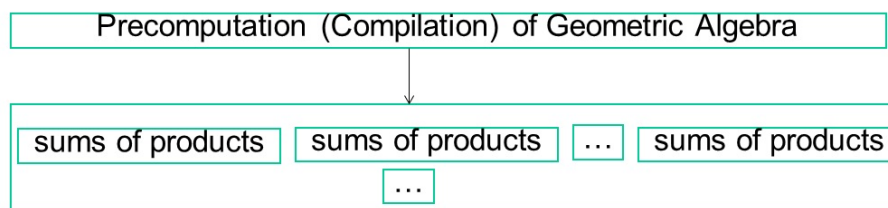


FIGURE 1. Precomputation of Geometric Algebra leads to parallel computations of sums of products

A good way of losing the high complexity of Geometric Algebra before going to the real computing device is to precompute / precompile Geometric Algebra algorithms. What remains after this process are only parallel computations of multivector coefficients each consisting of long sums of products, which are again efficiently to be parallelized [10]. Although not designed for computing with Geometric Algebra, fortunately, most of the current computing devices support these kind of operations.

While our Gaalop precompiler [11] already supports CPUs, GPUs and FPGAs, in this paper, we introduce a solution for an even broader range of heterogeneous computing architectures

defined by the HSA Foundation (see Sect. 3). Since they are focusing on heterogeneous computing with the programming language C++ AMP (see Sect. 4), we extended our Gaalop pre-compiler accordingly (see Sect. 5), in order to support all the solutions of the 40+ companies of this foundation. This Geometric Algebra solution is now part of the ecosystem of the HSA Foundation. As a proof-of-concept we describe a raytracing application implemented by the Gaalop precompiler for C++ AMP in Sect. 7.

## 2. COMPUTING WITH GEOMETRIC ALGEBRA

For many engineering applications runtime performance is a big issue. One method to attempt to overcome the limitations of Geometric Algebra has been to look for dedicated hardware architectures for the acceleration of Geometric Algebra algorithms. Integrated circuit technology offers a means to achieve high performance with field-programmable gate arrays (FPGAs). See, for instance, the solutions by Perwass et al. [15], Gentile et al.[8], Franchini et al. [7] and Mishra and Wilson [13].

Another approach to overcoming the runtime limitations of Geometric Algebra has been through optimized software solutions. Tools have been developed for high-performance implementations of Geometric Algebra algorithms such as the C++ software library generator Gaigen 2 from Daniel Fontijne and Leo Dorst of the University of Amsterdam [5], GMac from Ahmad Hosney Awad Eid of Suez Canal University [4], the Versor library [2] from Pablo Colapinto, the C++ expression template library Gaalet [16] from Florian Seybold of the University of Stuttgart, and our Gaalop compiler [11]. In 2006 we, together with the Amsterdam group, presented the first Geometric Algebra application that was faster than the standard implementation [12]. In the meantime, we have improved our Geometric Algebra Computing technology further in order to support the newest programming languages and parallel computing architectures, as described in the book [10]. We are already supporting GPU programming languages such as CUDA and OpenCL or the FPGA programming language Verilog while most of the other software solutions are restricted to CPU focused programming languages such as C++ or C#. As a solution for the computing architecture of the HSA Foundation we present in Sect. 5 our Gaalop precompiler for C++ AMP.

## 3. HSA FOUNDATION

The HSA Foundation [6] is a not-for-profit industry standards body of 40+ companies, founded by AMD, ARM, Imagination, Mediatek, Qualcomm, Samsung and Texas Instruments. It is focused on making it dramatically easier to program heterogeneous computing devices for parallel computation utilizing CPUs, GPUs, DSPs, etc.

Heterogeneous computing is emerging as a requirement for power-efficient system design: modern platforms no longer rely on a single general-purpose processor, but instead benefit from dedicated processors/accelerators tailored for each task. Traditionally these specialized processors have been difficult to program due to separate memory spaces, kernel-driver-level interfaces, and specialized programming models. The Heterogeneous System Architecture (HSA) aims to bridge this gap by providing a common system architecture and a basis for designing higher-level programming models for all devices (including widely used system-on-chip devices, such as tablets, smartphones, and other mobile devices).

## 4. C++ AMP

C++ AMP is an extension to C++ that enables the acceleration of C++ code on data-parallel hardware (GPUs etc.). The first specification was published by Microsoft in August 2012 as an

open specification. The first implementations were available in Visual Studio 2012 and Visual Studio 2013.

For the goal of making it dramatically easier to program heterogeneous computing devices, the HSA foundation released their C++ AMP (Accelerated Massive Parallelism) compiler for Linux in Aug. 2014. C++ AMP version 1.2 enables C++ developers to accelerate applications across a broad set of hardware and software configurations by supporting three outputs:

- Khronos Group **OpenCL**, supporting AMD CPU/APU/GPU, Intel CPU/APU, NVIDIA GPU, Apple Mac OS X and other OpenCL compliant platforms;
- Khronos Group **SPIR**, supporting AMD CPU/APU/GPU, Intel CPU/APU and future SPIR compliant platforms; and
- HSA Foundation **HSAIL**, supporting AMD APU and future HSA compliant platforms.

As follows, we describe the main extensions of C++ AMP for accelerators:

**parallel\_for\_each** describes a computation to be performed by an accelerator across some  $N$ -dimensional execution domain. It expects the number of threads and a lambda function describing the functionality to be executed for each thread.

The ADT (abstract data type) **array\_view**  $\langle T, N \rangle$  logically represents an  $N$ -dimensional space of type  $T$  which resides either on the memory space of the host or of the accelerator, for instance a 2-dimensional pixel array of colours as described in listing 1.

The ADT **index**  $\langle N \rangle$  represents an  $N$ -dimensional point, for instance one point of a 2-dimensional pixel array.

For details see [9].

## 5. THE GAALOP PRECOMPILER FOR C++ AMP

In order to simplify the use of Geometric Algebra in engineering applications, we have developed Gaalop GPC, a precompiler, which integrates Geometric Algebra into standard programming languages [10]. Figure 2 outlines the concept for the C++ AMP programming language.

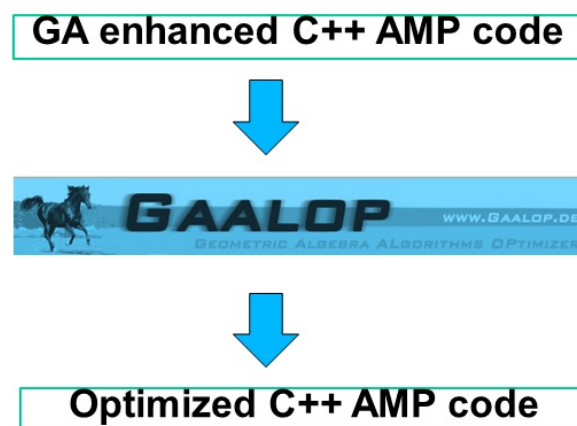


FIGURE 2. Gaalop Precompiler for C++ AMP.

With the Gaalop precompiler, we are able to enhance ordinary C++ AMP code with Geometric Algebra code and automatically generate optimized C++ AMP code.



TABLE 1. Gaalop precompiler functions for constructing and accessing multivectors

<b>coeff = mv_getbladecoeff(mv,blade);</b>	Get the coefficient of blade <b>blade</b> of multivector <b>mv</b> .
<b>array = mv_to_array(mv, blades ,...);</b>	Write the blades <b>blades ,...</b> of multivector <b>mv</b> to array <b>array</b> . Example <b>array = mv_to_array (mv,e1,e2,e3,e0, einf );</b> .
<b>mv = mv_from_array(array,blades,..);</b>	Construct multivector <b>mv</b> from array <b>array</b>

A precompiler is an elegant way of extending the features of a programming language. For Geometric Algebra Computing, it is of high interest to use both the power of high-performance languages and the elegance of expression of a domain-specific language such as CLUCalc [14]. We have therefore embedded Geometric Algebra Computing code into C++ AMP code, and compile it by utilizing the precompiler concept and the fast optimizations and code generation features of Gaalop.

The Gaalop precompiler enhances C++ AMP programs by embedding

- Geometric Algebra code using multivectors;
- functionality to interact with multivectors.

It transforms these enhanced C++ AMP programs to optimized C++ AMP programs without any explicit Geometric Algebra functionality. The embedding of Geometric Algebra code is done based on pragmas with the following structure:

```
#pragma gpc begin
...
Import of multivectors
...
#pragma clucalc begin
...
Geometric Algebra code based on CLUCalc
...
#pragma clucalc end
...
Export of multivectors
...
#pragma gpc end
```

Each gpc (Gaalop precompiler) block includes a clucalc block with the Geometric Algebra functionality. The functions to import/export multivectors are defined in Table 1. The purpose of these functions is the transformation between multivectors and the C++ AMP language concepts of **float** variables and arrays. **mv\_get\_bladecoeff()** is responsible for extracting a blade coefficient from a multivector, whereas **mv\_to\_array()** constructs an array from a multivector and **mv\_from\_array()** constructs a multivector from a C-like array.

## 6. CONFORMAL GEOMETRIC ALGEBRA

As shown in [3] for Gaigen and in [4] for GMAC, Conformal Geometric Algebra is very well suitable to realize raytracing applications. This is because of its easy handling of objects such as spheres, planes and lines. In this section we give a brief overview of Conformal Geometric Algebra.

TABLE 2. The two representations (IPNS and OPNS) of conformal geometric entities. The IPNS and OPNS representations are dual to each other, which is indicated by the asterisk symbol.

Entity	IPNS representation	OPNS representation
Point	$P = \mathbf{x} + \frac{1}{2}\mathbf{x}^2 e_\infty + e_0$	
Sphere	$S = P - \frac{1}{2}r^2 e_\infty$	$S^* = P_1 \wedge P_2 \wedge P_3 \wedge P_4$
Plane	$\pi = \mathbf{n} + d e_\infty$	$\pi^* = P_1 \wedge P_2 \wedge P_3 \wedge e_\infty$
Circle	$Z = S_1 \wedge S_2$	$Z^* = P_1 \wedge P_2 \wedge P_3$
Line	$L = \pi_1 \wedge \pi_2$	$L^* = P_1 \wedge P_2 \wedge e_\infty$
Point pair	$Pp = S_1 \wedge S_2 \wedge S_3$	$Pp^* = P_1 \wedge P_2$

Conformal Geometric Algebra uses the three Euclidean basis vectors  $e_1, e_2, e_3$  and two additional basis vectors  $e_+, e_-$  with positive and negative signatures, respectively, which means that they square to +1 as usual ( $e_+$ ) and to -1 ( $e_-$ ).

$$(1) \quad e_+^2 = 1, \quad e_-^2 = -1, \quad e_+ \cdot e_- = 0.$$

Another basis  $e_0, e_\infty$ , with the geometric meaning

- $e_0$  represents the 3D origin,
- $e_\infty$  represents infinity,

can be defined with the relations

$$(2) \quad e_0 = \frac{1}{2}(e_- - e_+), \quad e_\infty = e_- + e_+.$$

Conformal Geometric Algebra provides a great variety of basic geometric entities to compute with, namely points, spheres, planes, circles, lines, and point pairs, as listed in Table 2. These entities have two algebraic representations: the IPNS (inner product null space) and the OPNS (outer product null space). These representations are duals of each other (a superscript asterisk denotes the dualization operator). In Table 2,  $\mathbf{x}$  and  $\mathbf{n}$  are in bold type to indicate that they represent 3D entities obtained by linear combinations of the 3D basis vectors  $e_1, e_2$ , and  $e_3$ :

$$(3) \quad \mathbf{x} = x_1 e_1 + x_2 e_2 + x_3 e_3.$$

The  $\{S_i\}$  represent different spheres, and the  $\{\pi_i\}$  represent different planes. In the OPNS representation, the outer product " $\wedge$ " indicates the construction of a geometric object with the help of points  $\{P_i\}$  that lie on it. A sphere, for instance, is defined by four points ( $P_1 \wedge P_2 \wedge P_3 \wedge P_4$ ) on this sphere. In the IPNS representation, the meaning of the outer product is an intersection of geometric entities. A circle, for instance, is defined by the intersection of two spheres  $S_1 \wedge S_2$ . Accordingly, the intersection of a line and a sphere can easily be expressed with the help of the outer product of these two geometric entities (Fig. 3).

## 7. THE RAYTRACER PROOF-OF-CONCEPT

Here, we present our raytracer application as a proof-of-concept for our Gaalop precompiler for C++ AMP. Raytracing is a technique for generating a 2D image by tracing the path of rays from a camera through the pixels in an image plane and simulating the light effects at the intersection with objects of a 3D scene (see Fig. 4).

Listing 1 describes the main routine of the raytracer. ImageView is defined based on the C++ AMP ADT array\_view as a 2-dimensional pixel array (with extension HEIGHT and WIDTH) of colours. With parallel\_for\_each we describe the raytracing functionality for each pixel. In the current version of the proof-of-concept project [1], we use simplified functionality without the handling of shadow rays (raycasting).

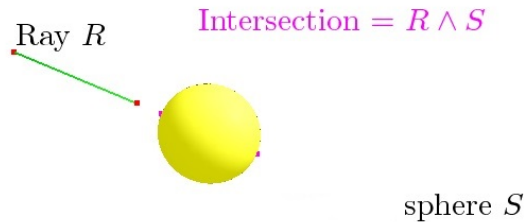
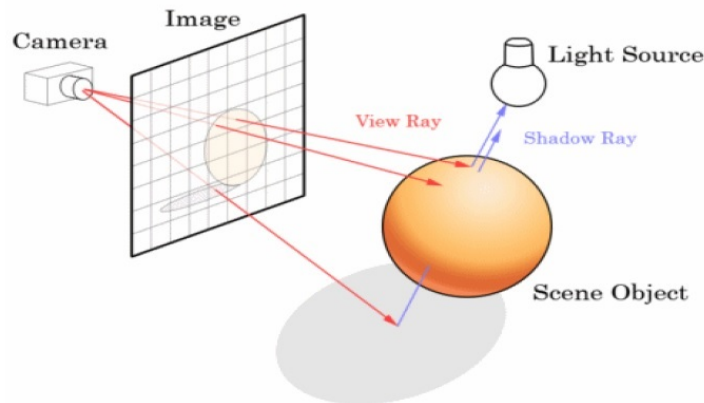


FIGURE 3. Spheres and lines are basic entities of Geometric Algebra that one can compute with. Operations such as the intersection of these objects are easily expressed with the help of their outer product. In our raytracing application, for instance, the result of the intersection of a ray and a sphere is another geometric entity: the point pair consisting of the two points where the line intersects the sphere. The sign of the square of the point pair indicates easily whether there is a real intersection or not.



<http://blog.codinghorror.com/real-time-raytracing/>

FIGURE 4. Raytracing principle.

LISTING 1. The Scene C++ AMP main routine

```
void Scene::renderOnGPU(std::vector<Colour>& imageData,
                       Camera camera, Light light) {
    array_view<Colour, 2> imageView(HEIGHT, WIDTH, &imageData[0]);
    array_view<Object, 1> allObjects(objectSize, objects);
    int length = objectSize;
    Color backgroundColor = background;
    parallel_for_each(imageView.get_extent(), [=](index<2> idx)
                    restrict(amp) {
        const auto y = idx[0]; // inverse order...
        const auto x = idx[1];
        // create a new ray...
        Ray ray = createNewRay(camera, x, y);
        imageView[idx] = rayCastAlgorithm(x, y, ray, allObjects, light,
                                         length, backgroundColor);
    });
}
```

Listing 2 describes the integration of some Geometric Algebra functionality into C++ AMP. It is written in the Geometric Algebra language CLUCalc [14] and part of the above mentioned raycasting functionality.

In the first gpc block, a sphere  $S$  is computed and assigned to the array *sphere* (see the second row of Table 1). The predefined function VecN3() computes the conformal point of the Euclidean center point with the coordinates  $C_x, C_y, C_z$  (see the first row of Table 2). The sphere  $S$  is computed with corresponding radius *radius* according to the second row of Table 2.

LISTING 2. Computations with sphere and ray

```
#pragma gpc begin
#pragma clucalc begin
    ?S = VecN3(Cx, Cy, Cz) - 0.5*radius*radius*einf;
#pragma clucalc end
sphere = mv_to_array(S, e1, e2, e3, einf, e0);
#pragma gpc end
// Ray defined by origin (Ox, Oy, Oz) and direction (Lx, Ly, Lz)
#pragma gpc begin
#pragma clucalc begin
    O = VecN3(Ox, Oy, Oz);
    L = VecN3(Lx, Ly, Lz);
    ?Ray = *(O ^ L ^ einf);
#pragma clucalc end
    newRay.ray = mv_to_array(Ray, e1^e2, e1^e3,
                            e1^einf, e2^e3, e2^einf, e3^einf);
#pragma gpc end
```

The second gpc block computes the ray *Ray* and assigns its relevant coefficients to an array. The ray is computed based on the outer product of two of its points  $O$  and  $L$  and infinity (see the fifth row of Table 2). Note that the ray has to be dualized (with a leading asterisk) since the standard representation in Gaalop is the IPNS representation.

LISTING 3. Intersection of line and sphere with the Gaalop precompiler for C++ AMP

```
#pragma clucalc begin
?PP = Ray ^ Sphere;
?hasIntersection = PP.PP;
#pragma clucalc end
```

The listing 3 describes the integration of the Geometric Algebra functionality of the intersection of a sphere and a line into C++ AMP as well as the computation of an intersection indicator (see Figure 3).

## 8. CONCLUSION

Starting from the situation 15 years ago with a great gap between the low symbolic complexity on the one hand and the high numeric complexity of coding in GA on the other hand, this paper shows, that, in the meantime, this gap could be closed. While the computing architectures evolved from sequential to more and more parallel heterogeneous systems, the GA computing technology evolved from GA libraries to a precomputing / precompiling technology cutting the high complexity before going to the real computing device. This Geometric Algebra Computing technology based on Gaalop is now supporting a broad range of heterogenous systems.

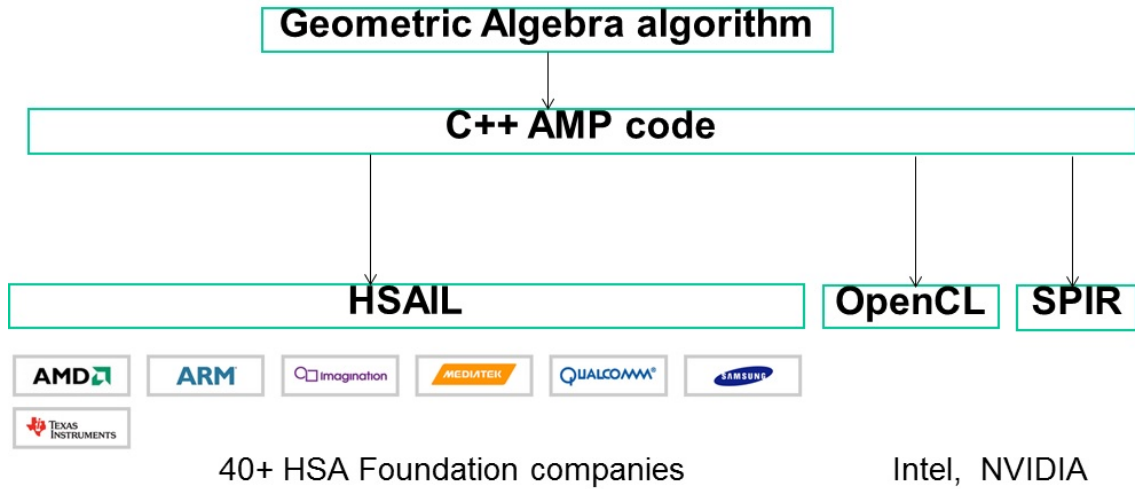


FIGURE 5. Solutions of 40+ companies of the HSA foundation as well as of Intel and NVIDIA can be supported by the Gaalop precompiler for C++ AMP.

The Gaalop precompiler for C++ AMP is able to support the solutions of 40+ companies of the HSA foundation via the HSAIL output format of their C++ AMP compiler. Since this compiler also supports OpenCL and SPIR, also Intel and NVIDIA solutions are supported.

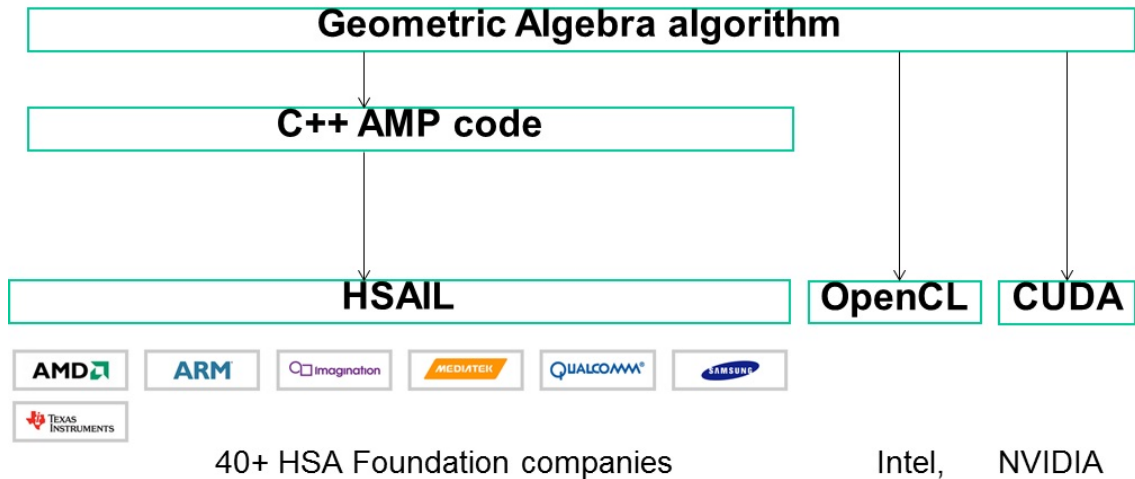


FIGURE 6. With the Gaalop precompilers for OpenCL and CUDA, Intel and NVIDIA solutions can be supported directly.

Since Gaalop precompilers are also available for OpenCL and CUDA, there is also a direct way to support Intel and NVIDIA according to Figure 6.

## REFERENCES

- [1] Justin Albert. The download page of the proof-of-concept raytracer based on the gaalop precompiler for C++ AMP. Available at [https://bitbucket.org/justinator\\_92/geometric-algebra-raytracer/wiki/Home](https://bitbucket.org/justinator_92/geometric-algebra-raytracer/wiki/Home), 2015.
- [2] Pablo Colapinto. The versor home page. Available at <http://versor.mat.ucsb.edu/>, 2015.
- [3] Leo Dorst, Daniel Fontijne, and Stephen Mann. *Geometric Algebra for Computer Science, An Object-Oriented Approach to Geometry*. Morgan Kaufmann, 2007.
- [4] Ahmad Hosney Awad Eid. *Optimized Automatic Code Generation for Geometric Algebra Based Algorithms with Ray Tracing Application*. PhD thesis, Suez Canal University, Port Said, 2010.
- [5] Daniel Fontijne, Tim Bouma, and Leo Dorst. Gaigen 2: A geometric algebra implementation generator. Available at <http://staff.science.uva.nl/~fontijne/gaigen2.html>, 2007.
- [6] HSA Foundation. The hsa foundation home page. Available at <http://www.hsafoundation.com/>, 2015.
- [7] S. Franchini, A. Gentile, M. Grimaudo, C.A. Hung, S. Impastato, F. Sorbello, G. Vassallo, and S. Vitabile. A sliced coprocessor for native Clifford algebra operations. In *Euromico Conference on Digital System Design, Architectures, Methods and Tools (DSD)*, 2007.
- [8] Antonio Gentile, Salvatore Segreto, Filippo Sorbello, Giorgio Vassallo, Salvatore Vitabile, and Vincenzo Vullo. Cliffosor, an innovative FPGA-based architecture for geometric algebra. In *ERSA 2005*, pages 211–217, 2005.
- [9] Kate Gregory and Ade Miller. *C++AMP*. Microsoft Press, 2012.
- [10] Dietmar Hildenbrand. *Foundations of Geometric Algebra Computing*. Springer, 2013.
- [11] Dietmar Hildenbrand, Patrick Charrier, Christian Steinmetz, and Joachim Pitt. Gaalop home page. Available at <http://www.gaalop.de>, 2015.
- [12] Dietmar Hildenbrand, Daniel Fontijne, Yusheng Wang, Marc Alexa, and Leo Dorst. Competitive runtime performance for inverse kinematics algorithms using conformal geometric algebra. In *Eurographics Conference Vienna*, 2006.
- [13] Biswajit Mishra and Peter R. Wilson. Color edge detection hardware based on geometric algebra. In *European Conference on Visual Media Production (CVMP)*, 2006.
- [14] Christian Perwass. The CLU home page. Available at <http://www.clucalc.info>, 2010.
- [15] Christian Perwass, Christian Gebken, and Gerald Sommer. Implementation of a Clifford algebra co-processor design on a field programmable gate array. In Rafal Ablamowicz, editor, *Clifford Algebras: Application to Mathematics, Physics, and Engineering*, Progress in Mathematical Physics, pages 561–575. 6th International Conference on Clifford Algebras and Applications, Cookeville, TN., Birkhäuser, 2003.
- [16] Florian Seybold. Gaalet – a C++ expression template library for implementing geometric algebra, 2010.
- [17] Gerald Sommer, editor. *Geometric Computing with Clifford Algebra*. Springer, 2001.



# ELEMENTS OF LINE GEOMETRY WITH GEOMETRIC ALGEBRA

Hongbo Li,<sup>a</sup> Lei Huang,<sup>a,b</sup> Lei Dong,<sup>a</sup> Changpeng Shao<sup>a</sup>

<sup>a</sup> Key Laboratory of Mathematics Mechanization  
Academy of Mathematics and Systems Science  
Chinese Academy of Sciences, Beijing 100190, China  
hli@mmrc.iss.ac.cn [corresponding]

<sup>b</sup> lhuang@mmrc.iss.ac.cn [presenter]

**ABSTRACT.** The line geometric model of 3-D projective geometry has the nice property that the Lie algebra  $sl(4)$  of 3-D projective transformations is isomorphic to the bivector algebra of  $CL(3,3)$ , and line geometry is closely related to the classical screw theory for 3-D rigid-body motions. The canonical homomorphism from  $SL(4)$  to  $Spin(3,3)$  is not satisfying because it is not surjective, and the projective transformations of negative determinant do not induce orthogonal transformations in the Plücker coordinate space of lines.

After introducing the line geometric model to the Geometric Algebra community in AGACSE 2009, we picked up the topic again in 2014 to try to establish the mathematical completeness by overcoming the unsatisfying facts of the Plücker correspondence, meanwhile to loosen the connection of line geometry with the Lie algebra of rigid-body motions, so that projective transformations other than Euclidean ones can meet with screw theory and other mechanical theories such as the virtual work principle, and new screw theories and virtual works may be established by extending the Euclidean motions to projective motions.

In this paper, we make a brief introduction to our contributions in finishing the above goal of research. A full version of the work is presented in [24].

**Key words:** Line Geometry; Screw Theory; Projective Geometry; Geometric Algebra; Lie Algebra.

## 1. INTRODUCTION

The study of the geometry of lines in space was invented by Plücker with his introduction of the now so-called Plücker coordinates of lines. It became an active research topic with the establishment of screw theory by Balls [1], where the 6-D Plücker coordinates of a line are decomposed into a pair of 3-D vectors, called the *screw form* of the line, and the inner product and cross product of vector algebra are extended to screw forms.

A pair of force and torque, called a *wrench*, are naturally represented by a pair of 3-D vectors, and are geometrically interpreted as a line in space along which the force acts, together with a line at infinity about which the torque acts. On the other hand, a pair of infinitesimal rotation and translation, called an infinitesimal *screw motion* or *rigid-body motion* or *twist*, are represented by the rotation axis and the translation vector, and are again naturally represented by a pair of 3-D vectors. Geometrically the translation is a special “rotation” about an axis that is at infinity, so the translation vector represents a line at infinity. For a wrench  $(\mathbf{f}, \mathbf{q})$  and an infinitesimal screw motion  $(\mathbf{v}, \mathbf{t})$ , where  $\mathbf{f}$  is the force direction multiplied with the magnitude of force,  $\mathbf{q}$  is the composed torque,  $\mathbf{v}$  is the rotation axis direction multiplied with the angle of rotation, and  $\mathbf{t}$  is the moment of the screw motion, the *virtual work* of the wrench along the infinitesimal screw motion is the “crossed” inner product

$$(1.1) \quad \begin{pmatrix} \mathbf{f} \\ \mathbf{q} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{v} \\ \mathbf{t} \end{pmatrix} := \mathbf{f} \cdot \mathbf{t} + \mathbf{q} \cdot \mathbf{v}.$$

---

Partially supported by NSFC project 2011CB302404.



The inner product (1.1) gives the 6-D space of wrenches a signature  $\mathbb{R}^{3,3}$ , where a pure force has zero inner product with itself, called a *null vector*. A *positive* (or *negative*) vector of  $\mathbb{R}^{3,3}$  is one having positive (or negative) inner product with itself. A positive vector is interpreted as a pure force together with an extraneous torque so that the pair follow the right-hand rule, while for the negative vector, the force and torque follow the left hand rule. The group  $SL(4)$  which acts in the 4-space of homogeneous coordinates of points, can be lifted to a group action in the 6-D space of wrenches by acting upon the Plücker coordinates of the lines representing the wrenches. The image of the lift is  $SO_0(3,3)$ , the connected component of  $SO(3,3)$  containing the identity [3]. As  $PSL(4) = SL(4)/\mathbb{Z}_2$  is the group of orientation-preserving projective transformations, the crossed inner product provides a 6-D orthogonal geometric model of wrenches to study 3-D projective geometry of points.

The same inner product (1.1) also gives the 6-D space of twists the same signature  $\mathbb{R}^{3,3}$ . The interpretation of a null vector of  $\mathbb{R}^{3,3}$  in the setting of twists, is that it represents an infinitesimal pure rotation or pure translation. A positive vector represents an infinitesimal screw motion where the translation along the screw axis follow the right-hand rule with the orientation of the rotation, while a negative vector represents a left-handed infinitesimal screw motion. The lift of group  $SL(4)$  to  $SO_0(3,3)$  then makes the space of infinitesimal **rigid-body motions** a 6-D orthogonal geometric model to study the orientation-preserving **projective geometry** of points.

The bold-faced words clearly reveal a conflict. The group of rigid-body motions is 6-D, while the group  $SL(4)$  is 15-D; the former is much smaller. What sense does it make to investigate projective transformations via rigid-body motions? Furthermore, the inner product (1.1) is between the space of wrenches and the space of twists, indicating that the two spaces need to be identified, yet they have to be different spaces by nature. Understanding (1.1) as a pairing between a linear space and its dual space does not make much difference, as the same inner product exists in either space.

Line geometry and screw theory are closely related to each other. In history, the screw forms were first used by Clifford in the name of biquaternions, also known as dual quaternions, in describing 3-D Euclidean transformations. Later on, Balls [1], Study [29], Blaschke[4] established screw theory and developed *dual vector algebra* out of Clifford's dual quaternions, also known as *screw algebra*. Nowadays line geometry together with screw theory have important applications in mechanism analysis, robotics, computer vision and computational geometry [5], [2], [27], [8], [25], [26], [7].

For two vectors  $(\mathbf{x}_1, \mathbf{y}_1)^T \in \mathbb{R}^{3,3}$  and  $(\mathbf{x}_2, \mathbf{y}_2)^T \in \mathbb{R}^{3,3}$ , where  $\mathbf{x}_i, \mathbf{y}_j \in \mathbb{R}^3$ , their *cross product*, also called *dual vector product*, is defined as follows:

$$(1.2) \quad \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{y}_1 \end{pmatrix} \times \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{y}_2 \end{pmatrix} := \begin{pmatrix} \mathbf{x}_1 \times \mathbf{x}_2 \\ \mathbf{x}_1 \times \mathbf{y}_2 + \mathbf{y}_1 \times \mathbf{x}_2 \end{pmatrix}.$$

This product is covariant under the subgroup of  $SO_0(3,3)$  that is the lift of the group of rigid-body motions  $SE(3)$ , but not so under the whole group  $SO_0(3,3)$ . In other words, it is not a valid operator in 3-D projective geometry; it is valid only for Euclidean geometry.

In dual vector algebra, the *dual inner product* of two vectors of  $\mathbb{R}^3$  is defined to be a *dual number*. A dual number is of the form  $\lambda + \varepsilon\mu$  where  $\lambda, \mu \in \mathbb{R}$ ,  $\varepsilon^2 = 0$  and  $\varepsilon$  commutes with everything. This numbers system is a ring instead of a field, and the corresponding polynomials and modules are drastically different from the usual ones. The dual inner product is invariant under the lift of  $SE(3)$  to  $SO_0(3,3)$ , so it is suitable for Euclidean geometry only.

In [21], it was pointed out that dual vector algebra and dual quaternions can be realized in the conformal geometric algebra  $CL(4,1)$ , and can be extended to arbitrary dimensions. The Euclidean geometric parts of the wrench model and the twist model have no conflict, and their identification is natural. The twist model should not have anything beyond Euclidean geometry,

otherwise it would be absurd. The wrench model, or more generally the model of lines in space, deserves further attention.

For invariant computing in projective geometry, the traditional algebraic tool is Grassmann-Cayley algebra and bracket algebra [22]. The study of projective geometry by Clifford algebra was initiated by Hestenes and Ziegler [15], and Stolfi [28]. The representation of projective transformations by spinors was initiated by Doran *et al.* [9], where a homomorphism of the Lie algebra  $gl(n)$  into  $so(n, n)$  was proposed, making it possible to construct projective transformations by elements of  $Pin(n, n)$ . Following this line, Goldman and Mann [13] discovered for many 3-D projective transformations their bivector generators in  $CL(4, 4)$ . Considering that the dimension of  $so(4, 4)$  is  $C_8^2 = 28$ , while the dimension of  $sl(4)$  is 15, the embedding space of  $sl(4)$  seems too high [11].

A classical result [3] states that the group  $SL(4)$ , which acts upon the 4-space of homogeneous coordinates of points, is in fact isomorphic to the group  $Spin_0(3, 3)$ , the connect component of  $Spin(3, 3)$  containing the identity, and the isomorphism is realized via the Plücker coordinates of lines and the adjoint action of  $Spin(3, 3)$  upon  $\mathbb{R}^{3,3}$ . This canonical isomorphism indicates the possibility of using the wrench model, the model of spatial lines, to study projective geometry with  $CL(3, 3)$ .

In AGACSE 2009, Li and Zhang [23] proposed a new model of 3-D projective geometry by taking the null vectors of  $\mathbb{R}^{3,3}$  as algebraic generators, and defining points and planes as the two connected components of the set of null 3-spaces of  $\mathbb{R}^{3,3}$  respectively. Whenever an element of  $Spin_0(3, 3)$  acts upon  $\mathbb{R}^{3,3}$ , it induces a projective transformation via the outermorphism of the action upon the null 3-vectors representing 3-D points and planes. This approach was later followed by Klawitter [19], who proposed an explicit expression of the spinor inducing a projective transformation in  $4 \times 4$  matrix form, and recently by Dorst [11], who constructed bivector generators for many 3-D projective transformations.

When viewed from the homogeneous coordinates model  $\mathbb{R}^4$  of 3-D projective geometry, the  $\mathbb{R}^{3,3}$  model seems to have too many defects. The map from  $SL(4)$  to  $SO(3, 3)$  is not surjective, nor injective. The projective transformations of negative determinant cannot be lifted to  $O(3, 3)$ , and conversely, the elements of  $O(3, 3)$  with negative determinant do not correspond to any projective transformation, but represent projective polarities where points are all mapped to planes. In the  $\mathbb{R}^{3,3}$  model, while lines are represented by vectors, the 3-D points and planes are represented by null 3-vectors, whose embedding vector space has dimension  $C_6^3 = 20$ . To make things worse, the mapping from  $\mathbb{R}^4$  to the null 3-vectors is quadratic, and defining the subset of null 3-vectors in the 20-D vector space they span is difficult.

From the mathematical viewpoint, establishing the space  $\mathbb{R}^4$  of homogeneous coordinates from the 6-space  $\mathbb{R}^{3,3}$  spanned by lines requires rigorous mathematical argument. It is the converse procedure of Plücker's construction of line coordinates from point coordinates. The well-definedness of the points and planes, and the covariance of the construction under suitable transformations of  $\mathbb{R}^{3,3}$  need to be established. The benefits of using the null 3-vectors instead of the linear space  $\mathbb{R}^4$  to represent points need to be discovered. The groups  $SO_0(3, 3)$  and  $Spin_0(3, 3)$  are too small to cover the whole group of all 3-D projective transformations and polarities, and finding suitable covering groups to provide spin representations for all 3-D projective transformations and polarities is indispensable.

So compared with other models of Geometric Algebra for classical geometries [14], [16], [20], [10], [22], [17], [12], [18], the line geometric model of 3-D projective geometry is much less developed. When every problem raised above is solved, then for the group of 3-D Euclidean transformations, a highly mature subject of study in Geometric Algebra, a one-to-one correspondence among the representations in the line geometric model and in other models need to be set up.

As mentioned before, the screw algebra is valid only for Euclidean motions, and the corresponding group  $SE(3)$  is only a subgroup of  $SL(4)$ . When  $SE(3)$  is replaced by another 6-D Lie subgroup of  $SL(4)$ , then the Lie algebra  $se(3)$  of Euclidean motions is replaced by another 6-D Lie subalgebra of  $sl(4)$ . Correspondingly, we can introduce new screw forms for the 6-D Lie subalgebra, together with the new “virtual work” of a wrench, which is still a vector of  $\mathbb{R}^{3,3}$ , along an infinitesimal “projective motion” represented by a screw form of the Lie subalgebra. The 6-D Lie subalgebra has its own Lie bracket, so the corresponding screw forms should have a different cross product. The new “virtual work” should be related to the new cross product, or even be completely determined by it.

We can go one step further by decomposing the 15-D algebra  $sl(4)$  into the direct sum of five 3-D vector spaces, so that instead of using only pairs of 3-D vectors as in classical screw theory for the screw forms of  $se(3)$ , we can use 5-tuples of 3-D vectors to represent screw forms of  $sl(4)$ , and develop a “super-screw theory”, equipped with “super-cross product” and “super-virtual work”.

For the purpose of developing a mathematically rigorous model out of the peculiar, unfamiliar and seemingly ineffective line geometric model, for more effectively describing and manipulating 3-D projective transformations with Geometric Algebra, and with the ambition to further extend screw theory to projective geometry, we picked up the research subject again in 2014, and after one-year’s hard work, we are confident to announce that from the algebraic viewpoint, this model is sufficiently mature now.

A lot of exciting results have been obtained, and some highlights will be introduced in this paper. The full version [24] of 62 pages can be downloaded from arXiv. The main contributions are the following:

1. Rigorous establishment of the  $\mathbb{R}^{3,3}$  model for 3-D projective geometry.

While in the classical model of projective line geometry only  $SL(4)$  has spin representation, and all the spinors are in  $Spin_0(3,3)$ , a rather unsatisfying limitation, the new model completely overcomes the limitation by providing pin group representations for all 3-D projective transformations and polarities, thus enlarging the transformation group four times.

The group of *linear regularities* of  $\mathbb{R}^{3,3}$  is defined by

$$(1.3) \quad RL(3,3) := \{\mathbf{B} \in GL(3,3) \mid \mathbf{B}^T \mathbf{J} \mathbf{B} = \pm \mathbf{J}\},$$

where  $\mathbf{J}$  is the matrix form of the metric of  $\mathbb{R}^{3,3}$ . Only when we computed the group acting upon the null 3-vectors induced by  $RL(3,3)$  did we find the complete version of the line geometric model. The group  $RL(3,3)$  double covers the whole group of 3-D projective transformations and polarities in this manner, and the group  $Pin^{sp}(3,3)$  quadruple-covers the latter, hence it can be used to construct versors for all kinds of 3-D projective transformations and polarities.

The well-definedness of points and planes in the  $\mathbb{R}^{3,3}$  model, and the covariance of the representations are established. Some nice properties of reflections in  $\mathbb{R}^{3,3}$  are found, together with the classification of 3-D projective transformations induced by two reflections in  $\mathbb{R}^{3,3}$ .

2. Construction of spinors in factored form inducing 3-D reflections and rigid-body motions, and discovery of the relation between the cross product of the screw forms of  $se(3)$  and the virtual work.

For 3-D reflections and rigid-body motions, the spinors inducing them in  $Pin^{sp}(3,3)$  in factored form are discovered. Since the bivector Lie algebra of  $CL(3,3)$  is isomorphic to  $sl(4)$ , any element of  $se(3)$  has a bivector form, and the cross product of the bivectors equals the cross product of their screw forms as vectors of  $\mathbb{R}^{3,3}$ . On the other hand, a wrench is only a vector of  $\mathbb{R}^{3,3}$ . To make pairing with a bivector, a vector needs to be first upgraded to a bivector of  $\Lambda^2(\mathbb{R}^{3,3})$  by making inner product (tensor contraction) with a trivector. We show that this trivector is exactly the one complementary to the trivector defining the cross product of the

screw forms of  $se(3)$ , and the latter trivector is exactly the lift of the quadratic form of  $\mathbb{R}^{3,0,1}$ . This correspondence shows the intrinsic connection between the virtual work and the cross product of  $se(3)$ . The connection between the wrench interpretation and the twist interpretation of the line geometric model is now clarified.

3. Extension of the cross product and virtual work of the screw forms of  $se(3)$  to other 6-D Lie subalgebras of  $sl(4)$ .

For many 6-D Lie subalgebras of  $sl(4)$ , we have developed the corresponding screw forms together with the cross product and virtual work that are completely determined by the Lie bracket of the subalgebra. In particular, for  $so(\mathbf{K})$  where  $\mathbf{K}$  is a quadratic form of  $\mathbb{R}^4$  with rank  $\geq 3$ , we have established the corresponding screw forms, cross products and virtual works, and discovered a striking fact: the trivectors for constructing new cross products and new virtual works are exactly the lifts of the quadratic form  $\mathbf{K}$  by the Plücker correspondence and the dual Plücker correspondence to the trivector space. This result demonstrates that there is no intrinsic connection between the  $se(3)$ -interpretation and the wrench interpretation of line geometry.

## 2. $\mathbb{R}^{3,3}$ MODEL OF 3-D PROJECTIVE GEOMETRY

The null 3-spaces of  $\mathbb{R}^{3,3}$  is composed of two connected components, the intersection dimension of every couple of elements in the same connected component is odd, while the intersection dimension of two elements, one from each component, is even. This property makes it possible to separate 3-D points and planes to be defined via  $\mathbb{R}^{3,3}$  without resorting to the now missing base space  $\mathbb{R}^4$ .

The *unitary regularity group*  $UR(4)$  is the union of  $SL(4), SL^-(4), SD(4), SD^-(4)$  acting on  $\mathbb{R}^4$ , where elements of  $SL^-(4), SD^-(4)$  have determinant  $-1$ , and elements of the other two have determinant 1. The set  $SD(4), SD^-(4)$  is composed of linear maps from the space of projective points  $\mathbb{R}^4$  to the space of projective planes  $(\mathbb{R}^4)^*$ .

The *Plücker transform* is defined by  $\mathbf{A} \in GL(4) \mapsto \wedge^2 \mathbf{A} \in GL(3, 3)$ ; the *dual Plücker transform* is defined by  $\mathbf{A} \in GD(4) \mapsto \vee^2 \mathbf{A} \in GL(3, 3)$ . For example,  $\wedge^2 \mathbf{A}(\mathbf{X} \wedge \mathbf{Y}) = (\mathbf{A}\mathbf{X}) \wedge (\mathbf{A}\mathbf{Y})$  for  $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^4$  and  $\mathbf{X} \wedge \mathbf{Y} \in \mathbb{R}^{3,3}$ . The Plücker transform and dual Plücker transform are a coupled group homomorphism double-covering  $RL_0(3, 3)$ , a subgroup of  $RL(3, 3)$  composed of 4 connected components. The group  $RL(3, 3)$  itself is composed of 8 connected components.

$Pin(3, 3)$  has 4 connected components:

- (1)  $Spin_0(3, 3)$ : Clifford product of even number of positive vectors, and even number of negative vectors; *e.g.*,  $\pm 1$  are included.
- (2)  $Spin_1(3, 3)$ : Clifford product of odd number of positive vectors, and odd number of negative vectors; *e.g.*,  $\pm \mathbf{I}_{3,3}$  are included, where  $\mathbf{I}_{3,3}$  is a unit pseudoscalar.
- (3)  $Spin_0^-(3, 3)$ : Clifford product of even number of positive vectors, and odd number of negative vectors; *e.g.*, negative vectors are included.
- (4)  $Spin_1^-(3, 3)$ : Clifford product of odd number of positive vectors, and even number of negative vectors; *e.g.*, positive vectors are included.

Fix an orthonormal basis  $\mathbf{e}_0, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$  of  $\mathbb{R}^4$ . The induced basis of  $\mathbb{R}^{3,3} = \Lambda^2(\mathbb{R}^4)$  is

$$(2.4) \quad \mathbf{E}_1 = \mathbf{e}_{01}, \quad \mathbf{E}_2 = \mathbf{e}_{02}, \quad \mathbf{E}_3 = \mathbf{e}_{03}, \quad \mathbf{E}'_1 = \mathbf{e}_{23}, \quad \mathbf{E}'_2 = \mathbf{e}_{31}, \quad \mathbf{E}'_3 = \mathbf{e}_{12}.$$

Denote  $\mathbf{E}_{ij} = \mathbf{E}_i \wedge \mathbf{E}_j$ . The following is the *symplectic form* [6] of the Witt decomposition  $\mathbb{R}^{3,3} = \mathbf{E}_{123} \oplus \mathbf{E}_{1'2'3'}$ :

$$(2.5) \quad \mathbf{K}_2 = \mathbf{E}_{11'} + \mathbf{E}_{22'} + \mathbf{E}_{33'}.$$

It depends only on the Witt decomposition, although the definition requires a fixed basis.

The linear transformation defined by  $\mathbf{K}_2$  is denoted by  $\mathcal{T}$ : for any  $\mathbf{X} \in \mathbb{R}^{3,3}$ ,  $\mathcal{T}\mathbf{X} = \mathbf{X} \cdot \mathbf{K}_2 \in \mathbb{R}^{3,3}$ .

$\mathcal{T}$  also has an action upon  $Pin(3,3)$ : for  $\mathbf{U} = \mathbf{X}_1\mathbf{X}_2 \cdots \mathbf{X}_r$ ,  $\mathbf{U}^{\mathcal{T}} = (\mathcal{T}\mathbf{X}_1)(\mathcal{T}\mathbf{X}_2) \cdots (\mathcal{T}\mathbf{X}_r)$ .

Define an associative product with the commutativity  $\mathcal{T} \circ \mathbf{U} = \mathbf{U}^{\mathcal{T}} \circ \mathcal{T}$ . Then  $Pin^{sp}(3,3) = Pin(3,3) \cup \mathcal{T}Pin(3,3)$  is a group. Define a new adjoint action of the group upon  $\mathbb{R}^{3,3}$ . Then  $Pin^{sp}(3,3)$  double covers  $RL(3,3)$ , and quadruple-covers  $PR(3) = UR(4)/\mathbb{Z}_2$ . The kernel of the latter covering homomorphism is  $\{\pm 1, \pm \mathbf{I}_{3,3}\}$ .

Let  $\mathcal{F}$  be the map from  $\mathbb{R}^4$  to the set of null 3-vectors, and let  $\mathcal{F}''$  be the map from  $(\mathbb{R}^4)^*$  to the set of null 3-vectors. They provide null 3-vector representations of 3-D points and planes. Are they covariant in that the following diagrams commute?

$$\begin{array}{ccc} \mathbf{X} \in \mathbb{R}^4 & \xrightarrow{\text{Plücker}} & \mathbf{Y}_3 \in \Lambda^3(\mathbb{R}^{3,3}) \\ \mathbf{A} \in GL(4) \downarrow & & \downarrow \mathbf{B} = \wedge^2 \mathbf{A} \\ \mathbf{A}\mathbf{X} \in \mathbb{R}^4 & \xrightarrow{\text{Plücker?}} & (\wedge^3 \mathbf{B})\mathbf{Y}_3 \in \Lambda^3(\mathbb{R}^{3,3}) \end{array}$$

The answer is: no, but after slight modification by multiplying the bottom Plücker transform with  $\det(\mathbf{A})$ , the diagram commutes. Similarly, the following diagram commutes, where  $GD(4)$  is the group of general linear polarities of  $\mathbb{R}^4$ :

$$\begin{array}{ccc} \mathbf{X} \in \mathbb{R}^4 & \xrightarrow{\text{Plücker}} & \mathbf{Y}_3 \in \Lambda^3(\mathbb{R}^{3,3}) \\ \mathbf{A} \in GD(4) \downarrow & & \downarrow \mathbf{B} = \sqrt{2} \mathbf{A} \\ \mathbf{A}\mathbf{X} \in (\mathbb{R}^4)^* & \xrightarrow{\det(\mathbf{A}) \text{ dual Plücker}} & (\wedge^3 \mathbf{B})\mathbf{Y}_3 \in \Lambda^3(\mathbb{R}^{3,3}) \end{array}$$

The maps  $\mathcal{F}$  and  $\mathcal{F}''$  are thus covariant, and points and planes are well-defined. By this covariance, we get explicit formulas for the matrix form of the induced group element of  $PR(3)$  from a given element of  $RL(3,3)$  being also in matrix form.

### 3. SPINORS GENERATING $SE(3)$ AND THE CORRESPONDING SCREW FORMS

A *screw form* of a vector of  $\mathbb{R}^{3,3}$  refers to  $\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}$ . This notion makes possible the use of vector algebra in  $CL(3,3)$ , and the representation of two different inner products by the same symbol within the same expression.

For example, the reflection with respect to an invertible vector  $\mathbf{U} = \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \in \mathbb{R}^{3,3}$  induces the following projective transformation in  $\mathbb{R}^4$ :

$$(3.6) \quad \begin{pmatrix} 0 & -\mathbf{y}^T \\ \mathbf{y} & \mathbf{x} \times \mathbf{I}_{3 \times 3} \end{pmatrix}.$$

This simple formula has the following application.

For any vector  $\mathbf{U} = \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \in \mathbb{R}^{3,3}$ , not necessarily invertible, as a bivector of  $\Lambda^2(\mathbb{R}^4)$ , it defines a linear transformation in  $\mathbb{R}^4$ :  $\mathbf{X} \in \mathbb{R}^4 \mapsto \mathbf{U} \cdot \mathbf{X} \in \mathbb{R}^4$ , where the inner product is in  $CL(\mathbb{R}^4)$ . The matrix form of this transformation is exactly the inverse-transpose of (3.6). In other words,

when the vector is invertible, then the matrix form of the transformation is the  $4 \times 4$  matrix of the reflection with respect to  $\mathbf{U}^{\mathcal{J}} = \begin{pmatrix} \mathbf{y} \\ \mathbf{x} \end{pmatrix} \in \mathbb{R}^{3,3}$ .

Hence, given a versor  $\mathbf{U} = \mathbf{V}_r \mathbf{V}_{r-1} \cdots \mathbf{V}_1$  in factored form, its adjoint action upon  $\mathbb{R}^{3,3}$  induces a projective transformation in  $\mathbb{R}^4$ , and the transformation is:

$$(3.7) \quad \begin{aligned} \mathbf{X} \in \mathbb{R}^4 &\mapsto \mathbf{V}_{2k} \cdot (\mathbf{V}_{2k-1}^{\mathcal{J}} \cdot (\mathbf{V}_{2k-2} \cdot (\cdots (\mathbf{V}_1^{\mathcal{J}} \cdot \mathbf{X}) \cdots))), & \text{for } r = 2k, \\ \mathbf{X} \in \mathbb{R}^4 &\mapsto \mathbf{V}_{2k-1}^{\mathcal{J}} \cdot (\mathbf{V}_{2k-2} \cdot (\cdots (\mathbf{V}_1^{\mathcal{J}} \cdot \mathbf{X}) \cdots)), & \text{for } r = 2k - 1. \end{aligned}$$

Consider the composition of two reflections in  $\mathbb{R}^{3,3}$ . Except for 3-D rotations of angle  $\pi$ , no other rigid-body motion is generated by two reflections in  $\mathbb{R}^{3,3}$ . The rotation of angle  $\pi$  about the axis at point  $\mathbf{y} \in \mathbb{R}^3$  in direction  $\mathbf{x} \in \mathbb{R}^3$ , where  $\mathbf{y}$  is chosen so that  $\mathbf{x} \cdot \mathbf{y} = 0$ , is generated by two reflections induced by the following vectors sequentially (from right to left):

$$(3.8) \quad \begin{pmatrix} \mathbf{x} \\ \mathbf{x} + \mathbf{y} \times \mathbf{x} \end{pmatrix}, \quad \begin{pmatrix} \mathbf{x} \\ -\mathbf{x} + \mathbf{y} \times \mathbf{x} \end{pmatrix}.$$

For a general rigid-body motion, we construct its explicit spinor generator in both factored form and in exponential form. For example, let  $\mathbf{v}_1, \mathbf{v}_2$  be unit vectors of  $\mathbb{R}^3$ . The rotation of  $\mathbb{R}^3$  in the  $\mathbf{v}_1 \mathbf{v}_2$  plane with angle  $2\angle(\mathbf{v}_1, \mathbf{v}_2)$  is generated by four reflections in  $\mathbb{R}^{3,3}$  with respect to the following vectors sequentially (from right to left):

$$(3.9) \quad \begin{pmatrix} \mathbf{v}_2 \\ -\mathbf{v}_2 \end{pmatrix}, \quad \begin{pmatrix} \mathbf{v}_2 \\ \mathbf{v}_2 \end{pmatrix}, \quad \begin{pmatrix} \mathbf{v}_1 \\ -\mathbf{v}_1 \end{pmatrix}, \quad \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_1 \end{pmatrix}.$$

The first two vectors induce the rotation of angle  $\pi$  in the plane normal to  $\mathbf{v}_2$  at the origin, and the last two vectors induce the rotation of angle  $\pi$  in the plane normal to  $\mathbf{v}_1$  at the origin. It is a geometric fact that the two rotations leads to the rotation of angle  $2\angle(\mathbf{v}_1, \mathbf{v}_2)$  in the  $\mathbf{v}_1 \mathbf{v}_2$  plane.

As another example, let  $\mathbf{x}, \mathbf{y}$  be nonzero vectors of  $\mathbb{R}^3$  satisfying  $\mathbf{x} \cdot \mathbf{y} = 0$ . The translation of  $\mathbb{R}^3$  by vector  $\mathbf{x}$  is generated by four reflections induced by the following vectors in  $\mathbb{R}^{3,3}$ :

$$(3.10) \quad \begin{pmatrix} \mathbf{x} \\ \mathbf{x} - \mathbf{y} + \frac{\mathbf{x} \times \mathbf{y}}{\mathbf{y}^2} \end{pmatrix}, \quad \begin{pmatrix} \mathbf{x} \\ \mathbf{x} + \frac{\mathbf{x} \times \mathbf{y}}{\mathbf{y}^2} \end{pmatrix}, \quad \begin{pmatrix} \mathbf{x} \\ \mathbf{x} + \mathbf{y} \end{pmatrix}, \quad \begin{pmatrix} \mathbf{x} \\ \mathbf{x} \end{pmatrix}.$$

The spinor has another factored form:

$$(3.11) \quad \begin{pmatrix} \mathbf{y} \\ -\mathbf{y} + \frac{\mathbf{x} \times \mathbf{y}}{2} \end{pmatrix}, \quad \begin{pmatrix} \mathbf{y} \\ \mathbf{y} + \frac{\mathbf{x} \times \mathbf{y}}{2} \end{pmatrix}, \quad \begin{pmatrix} \mathbf{y} \\ -\mathbf{y} \end{pmatrix}, \quad \begin{pmatrix} \mathbf{y} \\ \mathbf{y} \end{pmatrix}.$$

We see that the factored forms of a spinor are not unique. In contrast, the bivector generator is unique. The spinor generating the screw motion of angle of rotation  $\theta$  and screw driving distance  $d$  about the axis in unit direction  $\mathbf{v} \in \mathbb{R}^3$  at point  $\mathbf{c} \in \mathbb{R}^3$ , is the exponential of bivector

$$(3.12) \quad \frac{1}{2} \{ (\mathbf{E}_{>} \mathbf{E}' - \mathbf{E}_{<} \mathbf{E}') (\theta \mathbf{v}) - \mathbf{E}' \mathbf{E}' (\theta \mathbf{c} \times \mathbf{v} + d \mathbf{v}) \}.$$

In summary, any Euclidean transformation is found a generating spinor in factored form and in exponential form. In particular, a 3-D reflection has both a screw form of  $se(3)$  and a spinor in factored form in coset  $\mathcal{S}Spin(3,3)$ . For the reflection with respect to plane  $(\mathbf{n}, d)$ , where  $d$  is the signed distance from the plane to the origin and  $\mathbf{n}$  is the unit normal direction from the origin to the plane, the generator in  $\mathcal{S}Spin(3,3)$  is

$$(3.13) \quad \mathcal{S} \begin{pmatrix} \mathbf{v}_2 \\ -\mathbf{v}_2 - d \mathbf{v}_1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_2 \\ \mathbf{v}_2 - d \mathbf{v}_1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ -\mathbf{v}_1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_1 \end{pmatrix},$$

where  $\mathbf{v}_1, \mathbf{v}_2, \mathbf{n}$  form an orthonormal basis of  $\mathbb{R}^3$ . When  $d = 0$ , the above spinor can be simplified to

$$(3.14) \quad \mathcal{F} \left( \begin{array}{c} \mathbf{n} \\ -\mathbf{n} \end{array} \right) \left( \begin{array}{c} \mathbf{n} \\ \mathbf{n} \end{array} \right).$$

The screw form of the bivector generator of the above spinor is  $(\pi\mathbf{n}, 2d\mathbf{n})^T \in \mathbb{R}^{3,3}$ .

The cross product of  $\mathbb{R}^{3,3}$  is induced by the following trivector of  $CL(3,3)$ :

$$(3.15) \quad \mathbf{C}_3 = \mathcal{F}(\mathbf{e}_1) + \mathcal{F}(\mathbf{e}_2) + \mathcal{F}(\mathbf{e}_3),$$

such that for any  $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{3,3}$ ,

$$(3.16) \quad \mathbf{X} \times \mathbf{Y} = -(\mathbf{X} \wedge \mathbf{Y}) \cdot \mathbf{C}_3,$$

and  $\mathbf{X} \cdot \mathbf{C}_3$  is the element of  $se(3)$  whose screw form is  $\mathbf{X}$ .

By the following image of  $\mathbf{C}_3$  under  $\wedge^3 \mathcal{J}$ :

$$(3.17) \quad \mathbf{D}_3 = \mathcal{F}''(\mathbf{e}_1^*) + \mathcal{F}''(\mathbf{e}_2^*) + \mathcal{F}''(\mathbf{e}_3^*),$$

a vector  $\mathbf{X} \in \mathbb{R}^{3,3}$  is mapped to  $\mathbf{X} \cdot \mathbf{D}_3 \in so(3,0,1)$ , the Lie algebra preserving the metric  $\mathbb{R}^{3,0,1}$ . By

$$(3.18) \quad \left( \left( \begin{array}{c} \mathbf{v} \\ \mathbf{u} \end{array} \right) \cdot \mathbf{C}_3 \right) \cdot \left( \left( \begin{array}{c} \mathbf{f} \\ \mathbf{q} \end{array} \right) \cdot \mathbf{D}_3 \right) = -\mathbf{f} \cdot \mathbf{u} - 2(\mathbf{q} \cdot \mathbf{v}),$$

the virtual work of a wrench  $(\mathbf{f}, \mathbf{q})^T$  along a twist  $(\mathbf{v}, \mathbf{u})^T$  is realized by trivector  $\mathbf{D}_3$  up to the relative weight of the work done by the torque part  $\mathbf{q}$  over the work done by the force part  $\mathbf{f}$ .

#### 4. PROJECTIVE SCREW FORMS AND CORRESPONDING CROSS PRODUCTS

As the cross product in classical screw theory only reflects the Lie bracket of  $se(3)$ , and the virtual work of a wrench along an element of  $se(3)$  is in fact determined by the Lie bracket of  $se(3)$ , we can further extend screw theory to other 6-D Lie subalgebras of  $sl(4)$ , define the cross product of the corresponding screw forms, and the associative virtual work done by a wrench along an infinitesimal projective motion, with the hope that for non-Newtonian mechanics this may make sense.

The following is a typical class of 6-D Lie subalgebras of  $sl(4)$ . Let  $\mathbf{K}$  be a quadratic form of rank  $\geq 3$  defined on  $\mathbb{R}^4$ . Denote  $SO(\mathbf{K}) = \{\mathbf{A} \in SL(4) \mid \mathbf{A}^T \mathbf{K} \mathbf{A} = \mathbf{K}\}$ . Its Lie algebra  $so(\mathbf{K})$  is a 6-D subalgebra of  $sl(4)$ . Up to isomorphism there are only the following 5 kinds:

$$so(4), so(3,1), so(3,0,1), so(2,2), so(2,1,1).$$

1.  $so(4)$ :

The cross product of two screw forms of this algebra is

$$(4.19) \quad \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{y}_1 \end{pmatrix} \times_{so(4)} \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{y}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1 \times \mathbf{x}_2 + \mathbf{y}_1 \times \mathbf{y}_2 \\ \mathbf{x}_1 \times \mathbf{y}_2 + \mathbf{y}_1 \times \mathbf{x}_2 \end{pmatrix},$$

where  $\mathbf{x}_i, \mathbf{y}_j \in \mathbb{R}^3$ . The trivector of  $CL(3,3)$  realizing this cross product is

$$(4.20) \quad \mathcal{F}(\mathbf{e}_0) + \mathcal{F}(\mathbf{e}_1) + \mathcal{F}(\mathbf{e}_2) + \mathcal{F}(\mathbf{e}_3).$$

It corresponds to the matrix  $\text{diag}(1, 1, 1, 1)$  of the quadratic form preserved by  $SO(4)$ .

2.  $so(3,1)$ :

The cross product of two screw forms of this algebra is

$$(4.21) \quad \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{y}_1 \end{pmatrix} \times_{so(3,1)} \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{y}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1 \times \mathbf{x}_2 - \mathbf{y}_1 \times \mathbf{y}_2 \\ \mathbf{x}_1 \times \mathbf{y}_2 + \mathbf{y}_1 \times \mathbf{x}_2 \end{pmatrix}.$$

The trivector realizing the cross product is

$$(4.22) \quad -\mathcal{F}(\mathbf{e}_0) + \mathcal{F}(\mathbf{e}_1) + \mathcal{F}(\mathbf{e}_2) + \mathcal{F}(\mathbf{e}_3).$$

It corresponds to the matrix  $\text{diag}(-1, 1, 1, 1)$  of the quadratic form preserved by  $SO(3, 1)$ .

3.  $so(3, 0, 1)$ :

The cross product of two screw forms of this algebra is

$$(4.23) \quad \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{y}_1 \end{pmatrix} \times_{so(3,0,1)} \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{y}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1 \times \mathbf{x}_2 \\ \mathbf{x}_1 \times \mathbf{y}_2 + \mathbf{y}_1 \times \mathbf{x}_2 \end{pmatrix}.$$

The trivector realizing the cross product is

$$(4.24) \quad \mathcal{F}(\mathbf{e}_1) + \mathcal{F}(\mathbf{e}_2) + \mathcal{F}(\mathbf{e}_3).$$

It corresponds to the matrix  $\text{diag}(0, 1, 1, 1)$  of the quadratic form preserved by  $SO(3, 0, 1)$ .

4.  $so(2, 2)$ :

The cross product of two screw forms of this algebra is

$$(4.25) \quad \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{y}_1 \end{pmatrix} \times_{so(2,2)} \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{y}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1 \times_1 \mathbf{x}_2 + \mathbf{y}_1 \times \mathbf{y}_2 \\ \mathbf{y}_1 \times_2 \mathbf{x}_2 - \mathbf{y}_2 \times_2 \mathbf{x}_1 \end{pmatrix},$$

where for  $\mathbf{x} = (x_1, x_2, x_3)^T$  and  $\mathbf{y} = (y_1, y_2, y_3)^T$  of  $\mathbb{R}^3$ ,

$$(4.26) \quad \mathbf{x} \times_1 \mathbf{y} := \begin{pmatrix} x_2 y_3 - x_3 y_2 \\ -(x_3 y_1 - x_1 y_3) \\ -(x_1 y_2 - x_2 y_1) \end{pmatrix}, \quad \mathbf{x} \times_2 \mathbf{y} := \begin{pmatrix} -(x_2 y_3 - x_3 y_2) \\ -(x_3 y_1 + x_1 y_3) \\ x_1 y_2 + x_2 y_1 \end{pmatrix}.$$

The trivector realizing the cross product is

$$(4.27) \quad -\mathcal{F}(\mathbf{e}_0) - \mathcal{F}(\mathbf{e}_1) + \mathcal{F}(\mathbf{e}_2) + \mathcal{F}(\mathbf{e}_3).$$

It corresponds to the matrix  $\text{diag}(-1, -1, 1, 1)$  of the quadratic form preserved by  $SO(2, 2)$ .

5.  $so(2, 1, 1)$ :

The cross product of two screw forms of this algebra is

$$(4.28) \quad \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{y}_1 \end{pmatrix} \times_{so(2,1,1)} \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{y}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1 \times_1 \mathbf{x}_2 \\ \mathbf{y}_1 \times_2 \mathbf{x}_2 - \mathbf{y}_2 \times_2 \mathbf{x}_1 \end{pmatrix}.$$

The trivector realizing the cross product is

$$(4.29) \quad -\mathcal{F}(\mathbf{e}_1) + \mathcal{F}(\mathbf{e}_2) + \mathcal{F}(\mathbf{e}_3).$$

It corresponds to the matrix  $\text{diag}(0, -1, 1, 1)$  of the quadratic form preserved by  $SO(2, 1, 1)$ .

We have also studied other 6-D Lie subalgebras, for example the Lie algebra of the 6-D general non-uniform scaling group, and the 6-D subgroups of the 7-D invariant group of  $\mathbb{R}^{2,2}$ , and established the corresponding cross products.

Any element of  $sl(4)$  is of the form

$$(4.30) \quad \begin{pmatrix} -\text{tr}(\mathbf{N}_{3 \times 3}) & \mathbf{p}^T \\ \mathbf{t} & \mathbf{N}_{3 \times 3} \end{pmatrix},$$

where  $\mathbf{p}, \mathbf{t} \in \mathbb{R}^3$ . The matrix  $\mathbf{N}_{3 \times 3}$  can be decomposed into the diagonal part (a 3-vector denoted by  $\mathbf{d}$ ), the upper diagonal part (another 3-vector denoted by  $\mathbf{u}$ ), and the lower diagonal part (yet another 3-vector denoted by  $\mathbf{l}$ ). The whole Lie algebra  $sl(4)$  is then naturally decomposed into the direct sum of five 3-D Lie subalgebras, which are the five 3-spaces represented by the five vectors  $\mathbf{p}, \mathbf{t}, \mathbf{d}, \mathbf{u}, \mathbf{l}$  respectively. A bivector of  $CL(3, 3)$  has similar decomposition, leading to the *super-screw form* of an infinitesimal 3-D projective transformation: a 5-tuple of 3-D vectors. The corresponding ‘‘super-cross product’’ can be set up.



## 5. CONCLUSION

This introductory paper summarizes our recent work [24] on developing a solid mathematical foundation for the line geometric model of 3-D projective geometry, and extending the screw theory from Euclidean motions to projective motions.

The connection between the  $CL(3, 3)$  model of 3-D projective geometry and the  $CL(4, 1)$  model of 3-D conformal geometry is investigated in another paper. In that paper, the roles of the conformal point at infinity and the projective points at infinity relative to the Euclidean affine 3-space are clarified when both occur in the same model; the projective geometry of 3-D non-Euclidean geometry together with its various realizations in Euclidean space are investigated in the conformal model via different pin-hole cameras.

## REFERENCES

- [1] Balls, R.S. *A Treatise on the Theory of Screws*. Cambridge University Press, Cambridge, 1900.
- [2] Bayro-Corrochano, E. and Lasenby, J. Object Modeling and Motion Analysis Using Clifford Algebra. In: Mohr, R. and Wu, C. (eds.), *Proc. Europe-China Workshop on Geometric Modeling and Invariants for Computer Visions*, Xi'an, China, pp. 143-149, 1995.
- [3] Beffa, G.M., and Eastwood, M. Geometric Poisson Brackets on Grassmannians and conformal spheres. arXiv:1006.5753v1 [math.DG] 30 Jun 2010.
- [4] Blaschke, W. *Kinematik und Quaternionen*. VEB Deutscher Verlag der Wissenschaften, Berlin, 1960.
- [5] Chevallier, D.P. Lie Algebras, Modules, Dual Quaternions, and Algebraic Methods in Kinematics. *Machine and Mechanical Theory* **26**: 613-627, 1991.
- [6] Crumeyrolle, A. *Orthogonal and Symplectic Clifford Algebras*. D. Reidel, Dordrecht, Boston, 1990.
- [7] Dai, J. *Screw Algebra and Kinematic Approaches for mechanisms and Robotics*. Springer, London, 2014.
- [8] Daniilidis, K. Hand-Eye calibration Using Dual Quaternions. *International J. Robotics Research* **18**: 286-298, 1999.
- [9] Doran, C., D. Hestenes, D., Sommen, F., and Van Acker, N. Lie groups as spin groups. *Journal of Mathematical Physics* **34**(8) (1993), 3642-3669.
- [10] Dorst, L., Fontijne, D., and Mann, S. *Geometric Algebra for Computer Science*. Morgan-Kaufmann, 2007.
- [11] Dorst, L. 3D Oriented Projective Geometry through Versors of  $\mathbb{R}^{3,3}$ . Manuscript submitted to AACA in June, 2015.
- [12] Fleuystad, G. The Exterior Algebra and Central Notions in Mathematics. *Notices of the AMS*, Volume 62, Number 4, April 2015, 364-371.
- [13] Goldmann, R. and Mann, S.  $R(4,4)$  As a Computational Framework for 3-Dimensional Computer Graphics. *Adv. Appl. Clifford Algebras* **25** (2015), 113-149.
- [14] Hestenes, D. and Sobczyk, G. *Clifford Algebra to Geometric Calculus*. D. Reidel, Dordrecht, Boston, 1984.
- [15] Hestenes, D. and Ziegler, R. Projective Geometry with Clifford Algebra, *Acta Appl. Math.* **23**: 25-63, 1991.
- [16] Hestenes, D. *New Foundations for Classical Mechanics*. D. Reidel, Dordrecht, Boston, 2nd edition, 1998.
- [17] Hildenbrand, D. *Foundations of Geometric Algebra Computing*. Springer, 2012.
- [18] Kanatani, K. *Understanding Geometric Algebra*. CRC Press, Taylor & Francis Group, 2015.
- [19] Klawitter, D. A Clifford algebraic Approach to Line Geometry. *Advances in Applied Clifford Algebra* **24**: 713-736, 2014.
- [20] Lasenby, A. and Doran, C. *Geometric Algebra for Physicists*. Cambridge University Press, Cambridge, 2003.
- [21] Li, H., Hestenes, D. and Rockwood, A. Generalized Homogeneous Coordinates for Computational Geometry. In: Sommer, G. (ed.), *Geometric Computing with Clifford Algebras*, Springer, Heidelberg, pp. 27-60, 2001.
- [22] Li, H. *Invariant Algebras and Geometric Reasoning*. World Scientific, Singapore, 2008.
- [23] Li, H. and Zhang, L.: Line Geometry in Terms of the Null Geometric Algebra over  $\mathbb{R}^{3,3}$ , and Application to the Inverse Singularity Analysis of Generalized Stewart Platforms. In: *Guide to Geometric in Practice*, Dorst, L. and lasenby, J. (eds.) pp. 253-272, Springer, London, 2011.
- [24] Li, H., Huang, L., Dong, L., and Shao, C. Three-Dimensional projective geometry with Geometric Algebra. 62 pages, Manuscript uploaded to arXiv:submit/1292201 [cs.CG] 30 Jun 2015.
- [25] Merlet, J.-P. *Parallel Robot*. Kluwer Academic Publishers, Dordrecht, 2000.
- [26] Pottmann, H. and Wallner, J. *Computational Line Geometry*. Springer, Berlin, New York, 2001.
- [27] Selig, J.M. *Geometrical Methods in Robotics*. Springer, New York, 1996.
- [28] Stolfi, J. *Oriented Projective Geometry*. Academic Press, 1991.
- [29] Study, E. *Geometrie der Dynamen*. Leipzig, 1903.

# UNIFYING THEORY OF PYTHAGOREAN-NORMAL SURFACES

R. Krasauskas

Faculty of Mathematics and Informatics  
Vilnius University, Vilnius, Lithuania  
rimvydas.krasauskas@mif.vu.lt

Rational surfaces have become standard for industrial Computer-aided Design (e.g. Bézier or NURBS surfaces). Unfortunately, this class is not closed under the operation of offsetting. This problem motivated the concept of a Pythagorean-normal (PN) surface, defined as a rational surface admitting rational offsets. Peternell and Pottmann [1] started to study PN surfaces using Laguerre geometry. For example, they proved that duality defines a 1-1 correspondence between non-developable PN-surfaces in the Euclidean space  $\mathbb{R}^3$  and rational surfaces in the Blaschke cylinder, which is a particular model for Laguerre geometry. Other models, like the cyclographic model (CM) or the isotropic model (IM), also appeared useful for manipulation with PN-surfaces (see a survey [2] and a recent paper [3]).

Our idea is to place all these models into one space, which is the classical Lie sphere geometry space  $P^5$  defined by pseudo-euclidean vector space  $\mathbb{R}^{4,2}$  with signature  $(++++--)$ . Then we consider the geometric algebra  $Cl(4,2)$  generated by  $\mathbb{R}^{4,2}$ . This allows us to define all the different models of Laguerre geometry and the maps between them in terms of closed formulas in the algebra  $Cl(4,2)$ . Therefore, our approach provides a unifying formalism to deal with PN-surfaces. Here are some details of the construction. Consider the orthonormal basis  $\{e_1, \dots, e_6\}$  of  $\mathbb{R}^{4,2}$ , where  $e_i \cdot e_i = 1$ ,  $i = 1, \dots, 4$ ,  $e_i \cdot e_i = -1$ ,  $i = 5, 6$ , and two auxiliary vectors  $e_\infty = e_4 + e_5$ ,  $e_0 = (-e_4 + e_5)/2$ . Then define the Lie quadric  $\Lambda = \{X \in \mathbb{R}^{4,2} \mid X \cdot X = 0\}$ , and several subsets in  $\mathbb{R}^{4,2}$  corresponding to Laguerre geometry models (cyclographic model, Blaschke cylinder, and isotropic model):  $CM = \langle e_1, e_2, e_3, e_6 \rangle = \mathbb{R}^{3,1}$ ,  $BC = \Lambda \cap \{x_0 = 0\}$ ,  $IM = \langle e_1, e_2, e_\infty \rangle$ . Oriented spheres in  $\mathbb{R}^3$  are represented by vectors in  $CM$ : the first three coordinates are for the center and the last one is for the radius. Oriented planes  $n_1x_1 + n_2x_2 + n_3x_3 + d = 0$  in  $\mathbb{R}^3$  with normals  $n = n_1e_1 + n_2e_2 + n_3e_3$ ,  $|n| = 1$ , are represented by vectors  $n + de_\infty + e_6 \in BC$ . The key point of the proposed construction is the location of  $IM$ , which has the expected signature  $(++0)$ .  $BC$  and  $IM$  are related by the stereographic projection  $BC \rightarrow IM$  that simply forgets coordinates  $x_3$  and  $x_6$ , and its inverse, defined by the quadratic map

$$IM \rightarrow BC, \quad z \mapsto z + \frac{1}{2}z^2e_B + e_I, \quad e_B = -e_3 + e_6, \quad e_I = (e_3 + e_6)/2.$$

## REFERENCES

- [1] M. Peternell, and H. Pottmann, A Laguerre geometric approach to rational offsets, *Computer Aided Geometric Design* 15, (1998) 223–249.
- [2] R. Krasauskas and M. Peternell, Rational offset surfaces and their modeling applications, in: *IMA Volume 151: Nonlinear Computational Geometry*, (eds.) I.Z. Emiris, F. Sottile, and Th. Theobald, p. 109–135, 2010.
- [3] R. Krasauskas, S. Zube, S. Cacciola, Bilinear Clifford-Bezier Patches on Isotropic Cyclides, in: *Mathematical Methods for Curves and Surfaces*, Lect. Notes Comput. Sc. 8177 (2014), 283–303.



# SOAP FILMS AND THE GAUSS MAP

P. Lewintan <sup>a</sup>

<sup>a</sup> University Duisburg-Essen, Germany  
 peter.lewintan@uni-due.de  
 [presenter, corresponding]

2010 *Mathematics Subject Classification.* 58A05, 53A10, 30G35, 53Cxx

*Keywords.* Differential geometry, Clifford algebra, minimal surfaces, harmonic functions

The usage of geometric product not only simplifies many proofs, but also provides a deeper insight into differential geometry. We are going to present new connections between Geometric Analysis and Geometric Calculus:

Let  $\mathcal{M}$  be an  $m$ -dimensional smooth submanifold of  $\mathbb{R}^{m+k}$  and let  $a$  and  $b$  be vector fields on  $\mathcal{M}$ . The normal part of the directional derivative of  $b$  in the direction  $a$  can be expressed by

$$((a \cdot \partial)b)^\perp = a \lrcorner \mathcal{S}_b$$

cf. [HS, 4–3.16] where  $\mathcal{S}$  stands for the *shape operator* on  $\mathcal{M}$ . Furthermore we have  $a \lrcorner \mathcal{S}_b = b \lrcorner \mathcal{S}_a$ , so that  $\cdot \lrcorner \mathcal{S}(\cdot)$  is a symmetric bilinear form on  $T_x \mathcal{M}$  with values in the normal space  $N_x \mathcal{M}$ , namely the *second fundamental form* on  $\mathcal{M}$ , cf. [Sms prop 2.2.2].

In [HS] “the spur” of a manifold is defined to be the vector  $\partial_a \lrcorner \mathcal{S}_a$ . For an orthonormal basis  $\{\tau_1, \dots, \tau_m\}$  of  $T_x \mathcal{M}$  we arrive at

$$\mathcal{H} = \partial_a \lrcorner \mathcal{S}_a = \sum_{j=1}^m \tau_j \lrcorner \mathcal{S}_{\tau_j},$$

hence  $\mathcal{H}$  is the trace of the second fundamental form and, by definition, the *mean curvature vector* on  $\mathcal{M}$ , cf. [Sms p. 68], [DHT p. 301]. This insight allows us to approach the theory of Minimal Surfaces from the Geometric Calculus point of view:

Recall the definition [DHT sec 4.3 def 9]:  $\mathcal{M}$  is called *minimal*, iff  $\mathcal{H} \equiv 0$  on  $\mathcal{M}$ .

Moreover, we obtain [HS, 4–4.6]:  $\mathcal{H} = -(\partial \mathfrak{T}) \mathfrak{T}^{-1}$  (#)

where  $\partial = \partial_x$  denotes the vector derivative on  $\mathcal{M}$  and  $\mathfrak{T} = \mathfrak{T}(x)$  is the unit pseudoscalar in the Geometric Algebra generated from the tangent space  $T_x \mathcal{M}$ . For a good reason we call the map  $x \mapsto \mathfrak{T}(x)$  the *Gauss map* of  $\mathcal{M}$ . As a function  $F$  on  $\mathcal{M}$  is called *left monogenic*, iff  $\partial F \equiv 0$ , an interpretation of (#) yields the

**Theorem.**  $\mathcal{M}$  is minimal, iff the Gauss map is monogenic.

In our talk we will discuss the meaning of this theorem, consider the second derivatives of  $\mathfrak{T}$  and other operators on  $\mathcal{M}$ , i.e. further examples will be given.

Many of these relations are missing in the classical literature on Differential Geometry since an intrinsic view is usually preferred.

## REFERENCES

- [DHT] U. DIERKES, S. HILDEBRANDT and A. J. TROMBA: **Regularity of Minimal Surfaces.** Rev. and enlarged 2nd ed. Grundlehren der mathematischen Wissenschaften **340**, Springer 2010.
- [HS] D. HESTENES and G. SOBczyk: **Clifford Algebra to Geometric Calculus.** A unified language for mathematics and physics, D. Reidel Publishing Co., 1984.
- [Sms] J. SIMONS: *Minimal Varieties in Riemannian Manifolds*, Annals of Mathematic (2) **88** (1968), 62 – 105.



# FUNDAMENTALS OF 3D CLIFFORD BRACKET ALGEBRA

Hongbo Li<sup>a</sup>

<sup>a</sup> Key Laboratory of Mathematics Mechanization  
Academy of Mathematics and Systems Science  
Chinese Academy of Sciences, Beijing 100190, China  
hli@mmrc.iss.ac.cn [presenter, corresponding]

In classical invariant theory, the transformation group is the general linear group, and the invariants are polynomials of the *brackets* that are the determinants of the homogeneous coordinates of points. The brackets are algebraically dependent, and their algebraic relations are called *syzygies*. All the *syzygies* form an ideal, and modulo this ideal any invariant has a unique form called *straight form*. The procedure leading to the straight form is called *straightening*, and is made by a classical algorithm of Young early in the 20th century. In a straight form, any monomial is up to coefficient the product of several brackets, say  $k$  brackets each of length  $n$ , such that when the bracket factors are piled up in a column and the whole monic monomial forms a tableau, then the entries of each row is increasing, and the entries of each column is non-decreasing.

When the transformation group is restricted to the orthogonal group, then the invariants are polynomials of the brackets and the inner products of decomposable extensors, called *blades* by Hestenes. A straightening algorithm exists and is a special case of the normalization in Hodge algebra. In geometric application, however, the generators of the ring of invariants each have limited length, as when the surrounding vector space has dimension  $n$ , then each bracket has length at most  $n$ , and each inner product has length at most  $2n$  (it is also called a bideterminant when the length reaches  $2n$ ). The limitation of the lengths of the basic invariants often makes the symbolic manipulation of orthogonal invariants complicated.

By the “universal ungrading” technique in Clifford algebra, when the orthogonal group acts in 3D space, any orthogonal invariants can be changed into a polynomial where each term is the scalar part of the Clifford product of vector variables, called the *Clifford bracket* of the vectors. In other words, in a Clifford bracket polynomial, the length of a bracket is not limited, and can grow infinitely with the increase of the degree of the polynomial. The benefit is that within each Clifford bracket, the associativity of the Clifford product among the vectors brings nice symmetries, making the manipulation of such “long” brackets much easier than that of the short classical ones.

For 3D Clifford bracket algebra, one needs to compute a Gröbner basis of the syzygies among the Clifford brackets for arbitrary number of vector variables, and use it to normalize (also called straighten) Clifford bracket polynomials. Once every Clifford bracket polynomial is in normal form, then the leading term of the product of two Clifford bracket polynomials in normal form needs to be found, so that the division of one Clifford bracket polynomial by another can be carried out in the classical manner of top reduction of the dividend by only one divisor, instead of computing again a Gröbner basis and then making top reduction to the dividend by the whole set of computed Gröbner basis.

In the last two years, we have successfully solved the above two fundamental problems on 3D Clifford bracket algebra. In the procedure of solving the second problem, we have also solved

the same problem in classical bracket algebra, *i.e.*, making invariant division of one bracket polynomial by another by top reduction of the dividend with respect to only one divisor. This talk addresses our progress in solving the two problems.

# Application of Geometric Algebra to the Electroweak Sector of the Standard Model of Particle Physics

Gene E. McClellan

Applied Research Associates, Inc.  
Arlington, Virginia, U.S.A.  
[gmclellan@ara.com](mailto:gmclellan@ara.com)

Geometric algebra (GA) offers an alternative approach to understanding the fields of the Standard Model (SM) of high energy particle physics. This presentation examines a geometric view of electron and neutrino fields in the electroweak sector of the SM. These fields are related by the rotations of the  $SU(2)$  Lie group. Because our perceived space is three-dimensional and because the Lie algebra of  $SU(2)$  is isomorphic to that of the  $SO(3)$  Lie group, it is common within geometric algebra to address electroweak symmetry in terms of the generators of the Lie algebra of  $SO(3)$ . However, Hestenes and Sobczyk [1] point out that the natural representation of the special unitary group  $SU(n)$  in GA is in terms of generators that are compound (non-blade) bivectors in  $\mathcal{G}_{2n}$ , the GA of  $2n$ -dimensional Euclidean space. Therefore, a natural approach to electroweak theory mathematically is to work with  $SU(2)$  generators as compound bivectors in  $\mathcal{G}_4$ . This approach leads one to consider electroweak fields as multivector fields in  $\mathcal{G}_4$  that are solutions of the Dirac equation in four spatial dimensions and one time dimension. This presentation will examine such multivector fields, offer a new point of view on chiral projection of  $\mathcal{G}_3$  fields, and consider what spatial boundary conditions might lead to our perceived 3-D physical space.

## REFERENCE

- [1] D. Hestenes and G. Sobczyk, *Clifford Algebra to Geometric Calculus*, Dordrecht: D. Reidel Publishing Co., 1984.





# QUATERNION ATOMIC PHASE MAGNIFICATION FOR 3D MOTION

E. Ulises Moya-Sanchez and Marcela Bonell Manjarrez

Posgrado en Ciencias Computacionales UAG  
Universidad Autónoma de Guadalajara, Guadalajara, México  
eduardo.moya@colaborador.uag.mx [presenter, corresponding], marcelabonell@gmail.com

**ABSTRACT.** The phase concept is used in many applications of image processing such as edge, line and symmetry detection, image analysis and recently to show small motion or color changes. Eulerian motion magnification is a linear magnification based on the Laplace and Euler model of fluids and allows to show small color changes and motions invisible to the naked eye. However, noise power is amplified linearly with the amplification factor using in the Eulerian magnification. Manipulating the local phase variations (related to the Fourier shift theorem) with a Riesz-pyramid decomposition is possible to amplify the small motions, with two important advantages: achieves larger magnifications, and has better noise performance.

Exists three different types of phases, the global or Fourier phase, instantaneous phase and local phase. In this work, we use local phase concept based on the quaternions or bivector elements of  $\mathcal{G}_{3,0,0}$  in order to exhibit small and imperceptible motions in a 3D sequence acquisition manipulating the local phase coefficients of a hypercomplex multiscale pyramid. The main contributions of this work is, i) a new motion magnification method based on a unique compact supported window infinitely differentiable, the atomic function  $up(x)$ . ii) We apply our the phase and Eulerian magnification to 3D motion sequence magnification using a RGBD sensor (Kinect) that has not been reported so far. iii) We compare, the Riesz Pyramids method and the linear magnification in terms of the Signal to noise ratio (SNR), we found that the phase magnifications has better response than the linear method.

## 1. INTRODUCTION

The phase concept is commonly used in many applications areas of signal processing such as telecommunications, image processing, geophysics, etc. [11, 9, 13, 14]. Exists three different types of phases, the global or Fourier phase, the instantaneous phase and the local phase. In this work, we use local phase concept based on the quaternions or bivector elements of  $\mathcal{G}_{3,0,0}$  in order to exhibit small motions in a image sequence. Manipulating the local phase coefficients of a hypercomplex pyramid of an image sequence is an effective way to amplify small motions [1, 4, 5].

In this work, we propose a new technique in order to amplify 3D motion by using a an atomic function and the atomic function  $up(x)$  for filter space and short-term and long-term temporal variation (1D signals) on image sequences. There are three main reasons to use the atomic functions. The atomic function have a compact support in space domain, the  $n$ -order derivative is easy to compute and we can compute the Hilbert and the Riesz transform by using the first derivative [2]. Additionally we use a 3D sensor in order to do a 3D magnification. According to our results the phase magnification has better response than the linear method. The magnification technique has a lot of applications and actually can be use as a visual microphone using speed cameras.

## 2. QUATERNION ALGEBRA $\mathcal{H}$

T. Bülow and M. Felsberg [6, 7], claims that, the complex algebra is not enough to formulate the local phase  $n$ -dimensional generalization of the analytic signal. Therefore, the quaternion

---

This work has been supported by SNI-CONACYT..

algebra  $\mathcal{H}$  or  $\mathcal{G}_{3,0,0}$  in the Geometric algebra  $GA$  is used to compute the local phase. An element of  $\mathcal{H}$  consists of one real element adding three imaginary elements  $(i, j, k)$  *i.e.*

$$(2.1) \quad q = a + bi + cj + dk,$$

where  $a, b, c, d \in \mathcal{R}$  and  $i, j$  obey the relations  $i^2 = j^2 = -1, ij = k$ . The real part of  $q$  is noted by  $Re(q) = a$  and the pure part is  $Pu(q) = bi + cj + dk$  [8].  $\mathcal{H}$  is geometrically inspired, due to the quaternions may be used to represent rotations (as bivector) in  $\mathcal{R}^3$  and  $\mathcal{R}^4$  and translations (as vector) in  $\mathcal{R}^3$  [8].  $GA$  allows to distinguish naturally objects and operations [7]. The imaginary components can be described in terms of the basis of  $\mathcal{R}^3$  space,  $i \rightarrow e_{32}, j \rightarrow e_{13}, k \rightarrow e_{21}$  [7].

### 3. PHASE INFORMATION

The Fourier phase is the most well known phase and denotes the angular phase of a signal in frequency domain [6]. When another kind of information is needed such as the structural information of the signal the instantaneous or the local phase is used [6]. The instantaneous phases is used to know what is the phase at a certain position of the real signal [?, 6, 7]. In order to compute the instantaneous phase we first construct the analytic signal. For 1D signals, the instantaneous phase computation is based on the analytic signal and the Hilbert transform ( $f_{\mathbf{H}}[f(x)] = f(x) * \frac{1}{\pi x}$ ). The main idea of the Hilbert transform is to create a signal related with the original with a phase shift of  $\pi/2$  [6, 7] and is given by

$$(3.1) \quad f_A(f(x)) = f(x) + if_{\mathbf{H}}[f(x)],$$

$$(3.2) \quad f_A(f(x)) = |A|e^{i\theta},$$

where  $|A| = \sqrt{f(x)^2 + f_{\mathbf{H}}(x)^2}$  and  $\theta = \arctan(\frac{f(x)}{f_{\mathbf{H}}(x)})$ . The analytic signal permit us to extract the magnitude and instantaneous phase independently in the space domain. However, the Hilbert transform is a global transform and the instantaneous phase can be changed for others parts of the signal affecting the phase values in the region of interest [6, 7]. To solve this problem, it is common to use a window (or a bandpass filter) in convolution of the Hilbert transform [6, 7] in order to compute a local phase approximation.

**3.1. Local Phase .** The local phase means the computation of the phase restricted to a certain part or bandwidth of the real signal. The local phase is useful to separate the signal structure into impulses (even) and jumps (odd) [11, 9]. Additionally, the phase information allows us to use the invariant or equivariant properties of the signal [7]. For instance, it has been shown that the phase has an invariant response to image brightness and it can also be invariant to the rotations [7, 9].

In  $2D$  signals, the Hilbert transform is not enough to compute the magnitude and phase independently in any direction [7, 6]. Then, the quaternionic analytic signal and the monogenic signal have been proposed [6, 7, 11].

**3.2. Monogenic Signal.** The monogenic signal, was proposed by M. Felsberg and G. Sommer, and generalize the analytic signal to  $n-D$ . The monogenic signal for  $2D$  signals is represented by [7]

$$(3.3) \quad f_{\mathbf{M}}(\mathbf{x}) = f(\mathbf{x}) + (i, j)f_{\mathbf{R}}(\mathbf{x}) = f(\mathbf{x}) + (i, j)f(\mathbf{x}) * \frac{\mathbf{x}}{2\pi|\mathbf{x}|^3}.$$

The magnitude of the signal is computed by  $|f_{\mathbf{M}}(\mathbf{x})| = \sqrt{(i, j)f_{\mathbf{R}}^2 + f(\mathbf{x})^2}$ . Since the monogenic signal is constructed from the original signal and its Riesz transform, we can express the local phase  $\phi$  and the local orientation  $\theta$  as [7]

$$(3.4) \quad \phi = \arctan \left( \frac{|(i, j)f_{\mathbf{R}} * f(\mathbf{x})|}{f(\mathbf{x})} \right)$$

$$(3.5) \quad \theta = \arctan \left( \frac{jf_{\mathbf{R}} * f(\mathbf{x})}{if_{\mathbf{R}} * f(\mathbf{x})} \right).$$

#### 4. ATOMIC FUNCTIONS

**Definition 4.1.** The atomic functions (*AF*) are compactly supported, infinitely differentiable solutions of differential functional equations with a shifted argument [15], i.e.,

$$(4.1) \quad Lf(x) = \lambda \sum_{k=1}^M c(k)f(ax - b(k)), |a| > 1, b, c, \lambda \in \mathbb{N},$$

where  $L = \frac{d^n}{dx^n} + a_1 \frac{d^{n-1}}{dx^{n-1}} + \dots + a_n$  is a linear differential operator with constant coefficients. Among other *AFs*, the atomic function  $up(x)$  is the simplest and, at the same time, the most useful primitive function to generate other *AFs* [15].

**Definition 4.2.** The atomic function  $up(x)$  is generated by infinite convolutions of rectangular impulses. The function  $up(x)$  has the following representation in terms of the Fourier transform [15, 10]:

$$(4.2) \quad up(x) = \frac{1}{2\pi} \int_{\mathcal{R}} \prod_{k=1}^{\infty} \frac{\sin(v2^{-k})}{v2^{-k}} e^{ivx} dv,$$

$$(4.3) \quad = \frac{1}{2\pi} \int_{\mathcal{R}} \hat{u}p(v) e^{ivx} dv.$$

Some properties of the *AF* that we take advantage of have been reported in [10, 15]; these properties include the following attributes:

- The  $up(x)$  function is a compactly supported function in the space domain. Therefore, it can obtain good local characteristics.
- Since derivatives of any order can be represented in terms of shifts, any derivative can be represented as an operator, and the  $n$ -order derivatives are defined by

$$(4.4) \quad d^{(n)}up(x) = 2^{n(n+1)/2} \sum_{k=1}^{2^n} \delta_k up(2^n x + 2^n + 1 - 2k),$$

where  $\delta_{2k} = -\delta_k$ ,  $\delta_{2k-1} = \delta_k$ ,  $\delta_{2k} = 1$ . For example the first order derivative  $dup(x)$

$$(4.5) \quad dup(x) = 2up(2x+1) - 2up(2x-1).$$

- The *AFs* are infinitely differentiable ( $C^\infty$ ). As a result, the *AFs* and their Fourier transforms are rapidly decreasing functions. Therefore, their Fourier transforms decrease on the real axis faster than any power function.
- The *AF* windows were compared with classic ones by means of parameters such as the equivalent noise bandwidth, the 50% overlapping region correlation, the parasitic modulation amplitude, the maximum conversion losses (in decibels), the maximum side-lobe level (in decibels), the asymptotic decay rate of the side lobes (in decibels per

octave), the window width at the six-decibel level, the coherent gain, etc. All atomic windows exceed classic ones in terms of the asymptotic decay rate.

A radial atomic function was mentioned in [2] as  $up(\sqrt{x^2 + y^2})$  (see Fig 1). However in [2], the function  $Plop(x, y)$  was defined as a radial infinite differentiable function with compact support (see Fig 1), i.e.

$$(4.6) \quad P\hat{lop}(v, v) = \prod_{h=0}^{\infty} \sum_{k=0}^{\infty} \frac{[-(u^2 + v^2)]^k}{3^{2k(h+1)} [(k+1)!]^2}$$

and is solution of the following functional-differential equation [?]

$$(4.7) \quad \nabla^2 Plop(x, y) = \lambda \int_{\partial S} Plop[3(x - \xi_1), 3(y - \xi_2)] ds + \mu Plop(3x, 3y),$$

where  $\nabla^2 = \Delta = e_1 \frac{\partial^2}{\partial x^2} + e_2 \frac{\partial^2}{\partial y^2}$ ,  $\xi_1^2 + \xi_2^2 = 4/9$ ,  $\mu = -4\pi\lambda/3$  and  $\lambda = 3^5/4\pi$ .

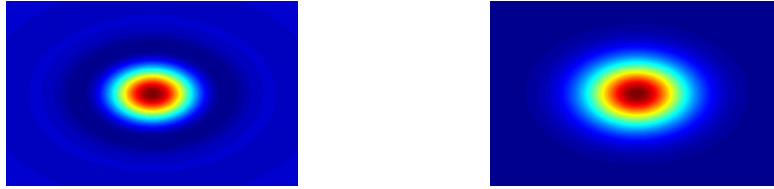


FIGURE 1. Left:  $\hat{u}p(\sqrt{v^2 + v^2})$ ; right:  $P\hat{lop}(v, v)$ .

**4.1. Riesz Transform Using  $Plop(x, y)$ .** According to [16] the Hilbert transform of wavelets are wavelets. Actually the Hilbert transform acts as an involution on the space of solutions of linear differential-functional equations solutions such as atomic functions [16]. The Riesz transform can be seen as a generalized Hilbert transform to  $n$ -dimensions [7]. A Riesz transform based on an atomic function was presented at [2] using this equation.

$$(4.8) \quad f_{\mathbf{R}}(\mathbf{x}) = f(\mathbf{x}) * \left( \nabla^2 Plop(\mathbf{x}) * -\frac{1}{2\pi} \text{sign}(|\mathbf{x}|) \log(|\mathbf{x}|) \right).$$

## 5. QUATERNION ATOMIC-RIESZ MULTIREOLUTION

Atomic-Riesz pyramid coefficients consists of a real part and two imaginary parts with two Riesz transforms based on equation 4.8.

$$(5.1) \quad f_{\mathbf{M}}(\mathbf{x}) = f(\mathbf{x}) + if_{\mathbf{R}_1}(\mathbf{x}) + jf_{\mathbf{R}_2}(\mathbf{x})$$

and this information is used to determine the local amplitude  $A = ||f_{\mathbf{M}}||$ , local phase  $\phi$  and local orientation  $\theta$ .

$$(5.2) \quad f_{\mathbf{M}} = A \cos(\phi) + iA \sin(\phi) \cos(\theta) + jA \sin(\phi) \sin(\theta)$$

$$(5.3) \quad \log \frac{f_{\mathbf{M}}}{||f_{\mathbf{M}}||} = iA\phi \cos(\theta) + jA\phi \sin(\theta)$$

Equation 5.3 uses a normalized quaternion, and is invariant to whether the local phase and orientation are  $\phi$  and  $\theta$  or the antipode  $-\phi$  and  $\theta + \pi$ .

## 6. LINEAR MAGNIFICATION

Eulerian video magnification, introduced by Wu et al [1, 4, 5]. is able to amplify small motions in videos without explicitly computing optical flow. In their work, the temporal brightness changes in frame sub-bands are amplified to amplify motions. Because this method amplifies brightness changes, the total amplification is limited and the noise power is amplified linearly with the amplification factor. Some characteristics of this approach are: each pixel are processed independently, each pixel has as a time series (spatio-temporal window) and amplify particular temporal frequencies [4]. Figure 2 shows the Euler magnification approach on one image by increasing temporal variation it is possible to increase spatial motion or color.

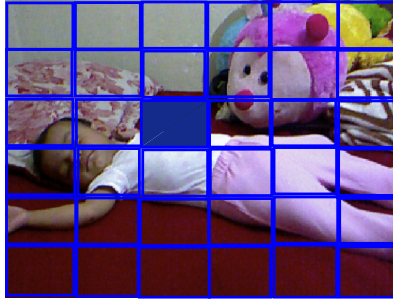


FIGURE 2. Magnification using a spatio-temporal window.

A simple case of a 1D signal let  $I(x, t)$  denote the image intensity at position  $x$  and time  $t$ . We can express the observed intensities with respect to a displacement as a function  $\delta(t)$ ,

$$(6.1) \quad I(x, t) = f(x + \delta(t))$$

$I(x, 0) = f(x)$ . The goal of motion magnification is to synthesize the signal  $I(x, t)$  to  $I(x, t)_M$  (motion amplification)

$$(6.2) \quad I(x, t)_M = f(x + (1 + \alpha)\delta(t))$$

where  $\alpha$  is the amplification factor and using a Taylor series expansion about  $x$

$$(6.3) \quad I(x, t)_M \approx f(x) + (1 + \alpha)\delta(t) \frac{\partial f(x)}{\partial x}$$

In order to do the implementation 6.3 Wadhwa et al [4] use a Laplace pyramid in addition to temporal filter. The main problem of linear magnification is that noise power is amplified.

## 7. PHASE MAGNIFICATION

Several papers has demonstrated that the local phase using bandpass filtered can be used for local structure description and motion estimation, shown the link between phase and motion and could be exploited in an Eulerian manner for motion magnification [9, 12]. Additionally, the main problem of linear magnification were solved by amplifying temporal phase variations using a complex steerable pyramid [1]. In this work we propose use our Riesz transform definition based on atomic functions.

The local phase computation has an ambiguous sign, therefore a filter a sequence of unit quaternions can be used. According to [4, 5] the local orientation  $\theta$  is supposed as constant over time at every pixel [5]. The equation 5.3 can be expressed in terms of  $n - frames$

$$(7.1) \quad iA\phi_n \cos(\theta) + jA\phi_n \sin(\theta)$$

At every pixel, a temporal filtering 1D on this quantity to isolate motions or changes of interest. In this case we propose use an 1D atomic function  $up(x)$ . Spatial filtering can be achieved by weighted blur with Plop function on the  $i$  and  $j$  components.

$$(7.2) \quad iPlop(x,y)A\phi_n \cos(\theta) + jPlop(x,y)A\phi_n \sin(\theta)$$

Motion amplify coefficients in the phase approach are in the same way as phase-shift a complex number. A quaternion expression is amplified by  $\alpha$  factor such as

$$(7.3) \quad A \cos(\alpha\phi) + iA \sin(\alpha\phi) \cos(\theta) + jA \sin(\alpha\phi) \sin(\theta)$$

When multiply  $I_M$  this unit quaternion by the original coefficient  $(\mathbf{x}) + if_{\mathbf{R}_1}(\mathbf{x}) + jf_{\mathbf{R}_2}(\mathbf{x})$  the real part can be expressed by [5]

$$(7.4) \quad I_M = A \cos(\alpha\phi) - Af_{\mathbf{R}_1} \sin(\alpha\phi) \cos(\theta) - Af_{\mathbf{R}_2} \sin(\alpha\phi) \sin(\theta)$$

As a summary, the first step to phase magnification is to express the image sequence in terms of the quaternion Riesz pyramid. The next step is to filter the local phase in a temporal way (as a 1D signal). Then an amplify factor ( $\alpha$ ) is apply to the local phase(magnitude and local orientation are not affected) , finally the Quaternion Riesz pyramid of the  $n$  frames is reconstructed. This method is very time consuming due to we need 3 image sequence pyramids instead of one Laplacian pyramid.

## 8. RESULTS AND ANALYSIS

**8.1. 2D Magnification.** Linear magnification of a RGB image sequence is shown in the Fig 3. In order to show the difference between the original image sequence and the phase magnification we select a Regio Of Interest (ROI) in the chest of a baby sleeping.



FIGURE 3. Left: ROI of the image sequence for 2D magnification.

Phase magnifications using the Riesz transform are presented in the Fig 4. From left to right, original ROI, phase amplification of ROI and difference of each frame. From top to down we show 5 frames at  $t = 0 : 0.5 : 2sec$ .

**8.2. 3D magnification.** One contribution of this work is the use a InfraRed (IR) 5 depth sensors *Kinect<sup>TM</sup>* (is a trademark of Microsoft Inc.) sensor. Kinect sensor provides a RGB-Depth images with  $640 \times 480$  images at  $30 fps$ . Due to IR images of both sensor has more noise than RGB images we use a phase technique in order to improve a linear Eulerian magnification method in two aspects : it supports larger magnification, and it has better noise performance. We select the same chest baby in order to show the breath frequency.

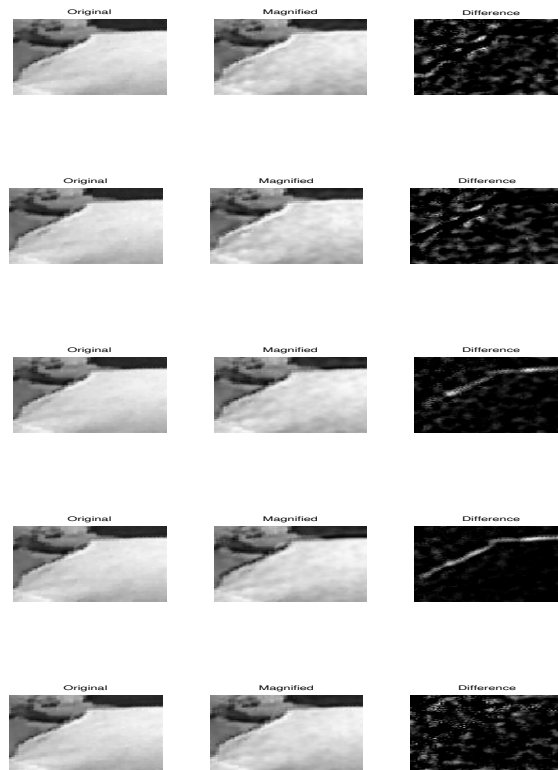


FIGURE 4. Left: Linear magnification of a ROI of the image sequence.

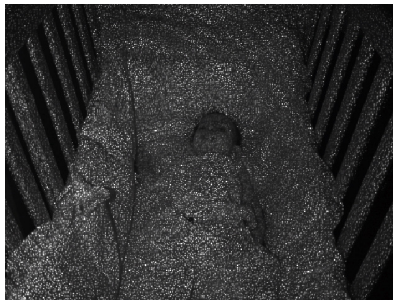


FIGURE 5. IR image. It is possible to see the IR pattern has a lot of noise.

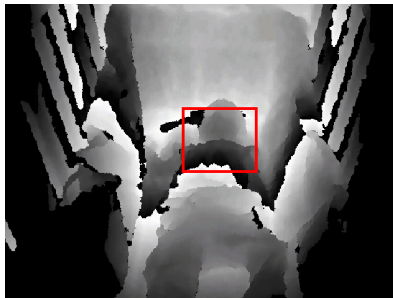


FIGURE 6. Depth image. We select a ROI of 3D image using the Kinect.

We use the depth map from the Kinect we present the 3D motion magnification Fig 7. From left to right, original ROI, phase amplification of ROI and difference of each frame. From top to down we show 5 frames at  $t = 0 : 0.5 : 2sec$ .



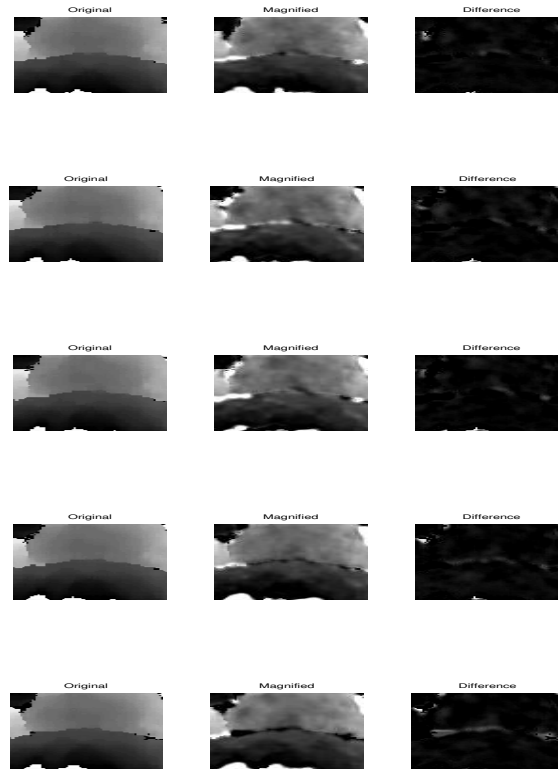


FIGURE 7. Depth image phase magnification of the selected ROI .

In the selected ROI we compute the Signal to Noise ratio in terms of the mean of the ROI (signal) and the standard deviation of the ROI (Noise) of the magnification areas. We can see that the linear magnification Fig 8. has more variations, in contrast phase magnification 9 has a shift in the breath peaks. Fig 10 shows the standard deviation with the linear magnification of the ROI and the phase magnification of the same ROI we can see how the phase computation has better noise performance.

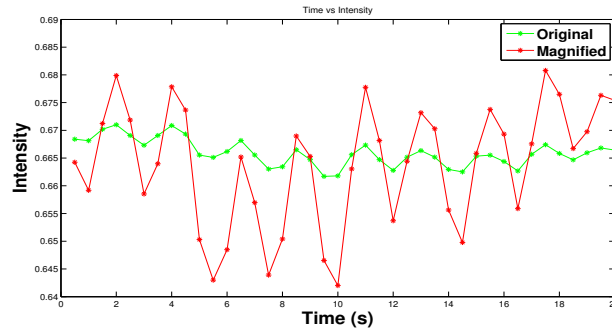


FIGURE 8. Linear magnification of the selected ROI.

## 9. CONCLUSIONS

As a conclusion, we present a new quaternionic method based on a unique compact supported window infinitely differentiable, the atomic function  $up(x)$  . Additionally we present a 3D magnification based on IR depth sensor *Kinect<sup>TM</sup>* . Due to IR images of both sensor has more noise than RGB images we use a phase technique to improve the linear Eulerian magnification method in two aspects it supports larger magnification, and it has better noise performance.

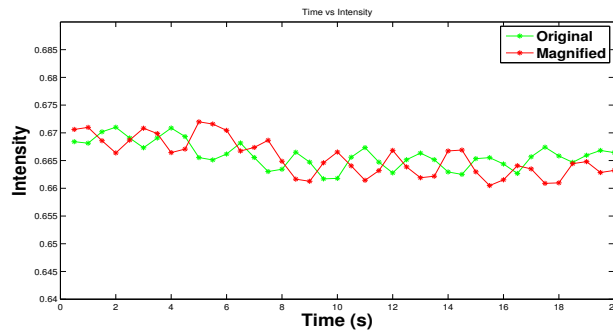


FIGURE 9. Phase magnification of the selected ROI.

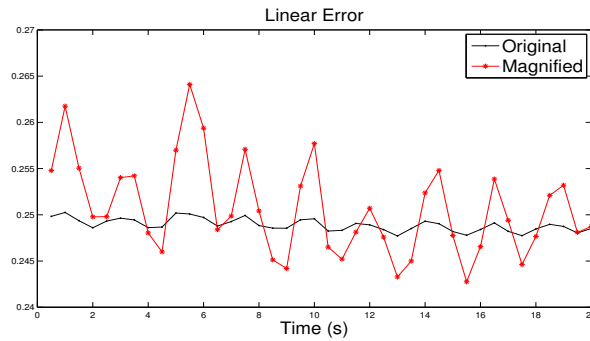


FIGURE 10. Standard deviation of the selected ROI using a linear magnification.

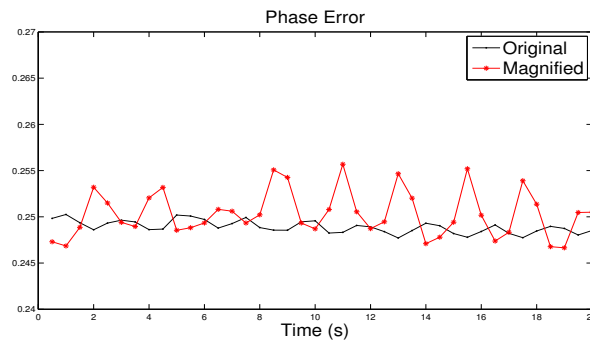


FIGURE 11. Standard deviation of the selected ROI using a Phase magnification.

Finally we compare, the Riesz Pyramids method and the linear magnification in terms of the Signal to noise ratio (SNR), we found that the phase magnifications has better response than the linear method. The phase motion magnification method has higher quality, however it is also more expensive to compute . As a future work we want to use this approach in LEAP Motion sensor and working in a faster method based on a Riesz approximation and quadrature filters using atomic functions.

## REFERENCES

- [1] M. Rubinstein. Analysis and Visualization of Temporal Variations in Video. PhD thesis, Massachusetts Institute of Technology, Feb 2014.
- [2] Moya-Sánchez, E.U.; Bayro-Corrochano, E. *Hilbert and Riesz transforms using atomic function for quaternionic phase computation* Adv. Appl. Clifford Algebr. 23, No. 4, 929-949 (2013)
- [3] M. Felsberg and G. Sommer. The monogenic signal. IEEE Trans. Signal Process., 49(12):3136–3144, 2001.
- [4] Neal Wadhwa and Michael Rubinstein and Frédo Durand and William T. Freeman. Riesz Pyramids for Fast Phase-Based Video Magnification, 2014 IEEE International Conference on Computational Photography (ICCP).

- [5] Neal Wadhwa and Michael Rubinstein and Frédo Durand and William T. Freeman. Quaternionic Representation of the Riesz Pyramid for Video Magnification, <http://people.csail.mit.edu/nwadhwa/riesz-pyramid>
- [6] T. Bülow . *Hypercomplex spectral signal representations for the processing and analysis of images*. PhD thesis, Christian- Albert, Kiel University, Kiel, Germany, 1999.
- [7] M. Felsberg. *Low-level image processing with the structure multivector*. PhD thesis, Christian- Albert, Kiel University, Kiel, Germany, 2002.
- [8] P. Lunesto. *Clifford Algebras and Spinors*. Cambridge University Press, 2001.
- [9] P. Kovesi. *Invariant measures of image features from phase information*. PhD thesis, University of Western Australia, 1996.
- [10] Leo Dorts Joan Lasenby. *Guide to Clifford Algebras in Practice*. Springer, 2010.
- [11] W. Lennart. Local feature detection by higher order Riesz transforms on images. Master's thesis, Christian-Albert, Kiel University, Kiel Germany, 2008.
- [12] E. Bayro Corrochano. The theory and use of the quaternion wavelet transform. *Journal of Mathematical Imaging and Vision*, 24:19–35, 2006.
- [13] T.A Ell and S.J. Sangwine Hypercomplex Fourier Transforms of Color Images *IEEE Trans Image Proc*, 16(1):22–35, 2007.
- [14] E.M.S.Hitzer Quaternion Fourier Transform on Quaternion fields and Generalizations *J Adv Appl Clifford Alg*, 17(3):497–517, 2007.
- [15] V. M. Kolodyazhnyya and V. A. Rvachev. *Cybernetics and Systems Analysis*, 43, 2007.
- [16] J Weiss The Hilbert transform of wavelets are wavelets, Technical Report Applied Mathematics Group, 1995.

# GEOMETRIC CONTROL OF THE TRIDENT SNAKE ROBOT BASED ON CGA

**J. Hrdina, A. Návrat, P. Vašík<sup>a</sup> and R. Matoušek<sup>b</sup>**

<sup>a</sup> Institute of Mathematics, Faculty of Mechanical Engineering,  
Brno University of Technology, Brno, Czech Republic,  
hrdina@fme.vutbr.cz [corresponding], navrat.a@fme.vutbr.cz [presenter], vasik@fme.vutbr.cz

<sup>b</sup> Institute of Automation and Computer Science, Faculty of Mechanical Engineering,  
Brno University of Technology, Brno, Czech Republic,  
matousek@fme.vutbr.cz

**ABSTRACT.** Local control of a (general) trident snake robot is solved by means of conformal geometric algebra. The equations of the direct and differential kinematics are assembled. The Pfaff constraints are written in a geometric form which allows a universal solution for various modifications of the mechanism. Also the inverse kinematics and the singular postures are discussed and a solution is found. The functionality is demonstrated on a virtual model in CLUCalc programme.

## 1. INTRODUCTION

Originally the general trident snake robot has been introduced in [9]. It is a planar robot with a body in the shape of a triangle and with three legs consisting of  $\ell$  links. Its precise description is given below. Then, its simplest nontrivial version, corresponding to  $\ell = 1$ , has been mainly discussed, see e.g. [10], [11]. Within this paper, we focus on the general case of  $\ell$ -links. The aim of this article is to solve the complete local control in a new geometric form.

In terms of generalized coordinates, the non-holonomic forward kinematics equations can be understood as a Pfaff system and its solution as a distribution in the configuration space. Rachevsky-Chow Theorem implies that the appropriate non-holonomic system is locally controllable if the corresponding distribution is not integrable and the span of the Lie algebra generated by the controlling distribution has to be of the same dimension as the configuration space. The spanned Lie algebra is then naturally endowed by a filtration which shows the way to realize the movements by means of the vector field brackets [6, 4]. In the case  $\ell = 1$ , the system is locally controllable and the filtration is (3, 6).

The classical approach composes the kinematic chain of homogeneous matrices using the moving frame methods and Euler angles, [3]. Instead of this, our aim is to use the notions of conformal geometric algebra (CGA), where the Euclidean space  $\mathbb{E}_3$  is included. In this geometric setting, we can easily handle both linear objects and spheres of dimensions 2, 1 and 0, see [2, 5, 7].

In particular, the 0-dimensional sphere, referred to as a point pair, is used to derive the kinematic equations and for the control of the non-holonomic snake like robotic mechanisms, consequently. More precisely, to any link of a single point pair is assigned and the mechanism is transformed by rotations and translations. We introduce the forward kinematic equations (6), the differential kinematic equations (7), as well as the non-holonomic conditions (8). We also derive an equation for singular postures of the robot, (9). We demonstrate the theory on the 1-link trident snake and the functionality in the CLUCalc software designed for the computations in Clifford algebra, particularly in conformal geometric algebra.

---

*Date:* June 30, 2015.

The authors were supported by a grant no. FSI-S-14-2290.

## 2. CONFORMAL GEOMETRIC ALGEBRA – CGA

The classical approach composes the kinematic chain of homogeneous matrices using the moving frame methods and Euler angles, or in advance using the quaternion algebra  $\mathbb{H}$  by conjugation  $x \mapsto q^{-1}xq$ , where we view an Euclidean point  $x$  as a quaternion

$$x = (x_1, x_2, x_3) \rightleftharpoons x_1i + x_2j + x_3k.$$

and  $q$  is a quaternion given by

$$q = \cos \frac{\theta}{2} + u \sin \frac{\theta}{2},$$

where  $u$  is an axis of rotation  $u_1i + u_2j + u_3k$ . Instead of this, we use the notions of conformal geometric algebra, i.e. the Clifford algebra  $\mathcal{C}l(4, 1)$  where the Euclidean space  $\mathbb{E}_3$  is included by a mapping  $x \mapsto x + x^2e_\infty + e_0$ . In this geometric setting, we can easily handle both linear objects and spheres of dimensions 2, 1 and 0. Namely, these objects are simply elements of the algebra and can be transformed and intersected with ease. In addition, rotations, translation, dilations and inversions all become rotations in our 5-dimensional space, see [2, 5, 7].

More precisely, let  $\mathbb{R}^{4,1}$  denote a vector space  $\mathbb{R}^5$  equipped with the scalar product of signature  $(4, 1)$  and let  $\{e_1, e_2, e_3, e_+, e_-\}$  be an orthonormal basis. The Clifford algebra  $\mathcal{C}l(4, 1)$  can be described as a free, associative and distributive algebra such that the *geometric product*  $e_i e_j$  (i) coincides with the scalar product in the case  $i = j$  (ii) is equal to  $-e_j e_i$  for  $i \neq j$ . Hence the dimension of the algebra is  $2^5 = 32$ .

Next to the geometric product, we define two additional products on  $\mathbb{R}^{4,1}$  based on the geometric one for any  $u, v, \in \mathbb{R}^{4,1}$ , *inner product* and *wedge product*, respectively:

$$u \cdot v = \frac{1}{2}(uv + vu), \quad u \wedge v = \frac{1}{2}(uv - vu)$$

and thus the basis elements are derived as  $uv = u \cdot v + u \wedge v$ . The definition of these product extends to the whole algebra. Namely, given two basis blades  $E_i = e_{a_1} \wedge \dots \wedge e_{a_k}$  and  $E_j = e_{a_1} \wedge \dots \wedge e_{a_l}$  of grades  $k$  and  $l$  respectively the wedge (outer) product is defined as

$$E_i \wedge E_j := \langle E_i E_j \rangle_{k+l}$$

while the inner product is defined as

$$\begin{aligned} E_i \cdot E_j &:= \langle E_i E_j \rangle_{|k-l|}, \quad i, j, > 0 \\ &:= 0, \quad i = 0 \text{ or } j = 0, \end{aligned}$$

where  $\langle \rangle_k$  is the grade projection into grade  $k$ . These products can be used effectively to compute an intersection of geometric objects and distances respectively.

The basis conformal geometric elements can be represented by the multi-vectors from  $\mathcal{C}l(4, 1)$  either in the inner product null space (IPNS) representation or in the outer product null space (OPNS) representation. To work with CGA effectively, one defines  $e_0 = \frac{1}{2}(e_- + e_+)$  and  $e_\infty = (e_- - e_+)$ . Consequently, the following properties hold.

$$e_0^2 = 0, \quad e_\infty^2 = 0, \quad e_\infty e_0 + e_0 e_\infty = -2$$

The geometric objects which we use in this paper are then given as follows.

object	CGA element
Point	$Q = x + \frac{1}{2}x^2e_\infty + e_0$ (IPNS)
Point pair $Q_1, Q_2$	$P^* = Q_1 \wedge Q_2$ (OPNS)
PLine $L$	$L^* = Q_1 \wedge Q_2 \wedge e_\infty$ (OPNS)

Each geometric transformation (rotation, translation, dilation, inversion) of a geometric object represented by an algebra element  $\mathcal{O}$  is realized by conjugation  $\mathcal{O} \mapsto M\mathcal{O}\tilde{M}$ , where  $M$  is an appropriate multi-vector. For instance, the translation in the direction  $t = t_1e_1 + t_2e_2 + t_3e_3$  is realized by conjugation by the multi-vector

$$T = 1 - \frac{1}{2}te_\infty,$$

which can be written as  $e^{-\frac{1}{2}te_\infty}$ , and the rotation around the axis  $L$  by angle  $\phi$  is realized by conjugation by the multi-vector

$$R = \cos \frac{\phi}{2} - L \sin \frac{\phi}{2}$$

where  $L = a_1e_2e_3 + a_2e_1e_3 + a_3e_1e_2$ . Similarly to the case of a translation, the rotation can be also written as  $e^{-\frac{1}{2}\phi L}$ .

### 3. CONTROL THEORY OF THE TRIDENT SNAKE ROBOT

Model of the general trident snake robot is illustrated in Figure 3. It is a planar robot which consists of a body in the shape of an equilateral triangle with circumscribed circle of the unit radius and three branche legs. Each of the leg consists further of  $\ell$  rigid links of constant unit length interconnected by motorised joints and linked with the vertices of the triangular body by motorised joints. Each link has a passive wheel at its center which provide an important snake-like property that the ground friction in the direction perpendicular to the link is considerably higher than the friction of a simple forward move. In particular, this prevents the slipping and sliding sideways. We assume the wheels are placed in the link centers, but the case of a general position is also discussed.

To describe the actual position of a trident snake robot we need the set of  $3 + 3\ell$  generalized coordinates as shown in Figure 3. According to [9], we call  $g := (x, y, \theta) \in \mathbb{SE}(2)$  the configuration vector and

$$\phi := (\phi_{11}, \dots, \phi_{1\ell}, \phi_{21}, \dots, \phi_{2\ell}, \phi_{31}, \dots, \phi_{3\ell}) \in (S^1)^{3\ell}$$

the shape vector. Then the generalized coordinates associated to our system are

$$(1) \quad q = (g, \phi) \in \mathbb{SE}(2) \times (S^1)^{3\ell}$$

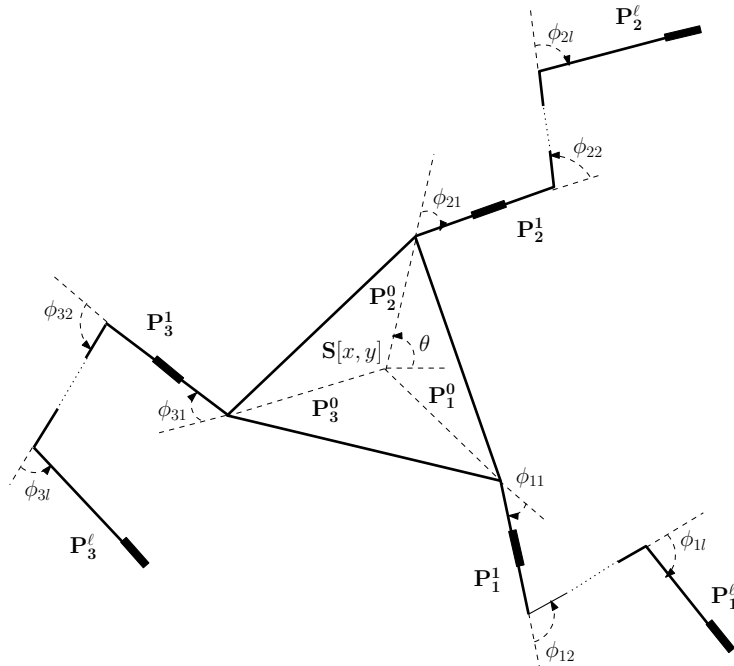


FIGURE 1. Trident snake robot model

**3.1. Euclidean description.** In this section, we rephrase the results of [9], where the kinematics of the trident snake robot is derived classically using euclidean geometry. First the euclidean position vector of link centers is expressed in terms of generalized coordinates (joint angles) via so called moving frame algorithm and then  $3\ell$  nonholonomic constraints are assembled. This gives the following kinematic equation.

$$A(\phi)R_\phi^T \dot{g} = B(\phi)\dot{\phi},$$

where the matrices occurring in this equation are defined by

$$A(\phi) = \begin{pmatrix} A_1(\phi) \\ A_2(\phi) \\ A_3(\phi) \end{pmatrix} \in \mathbb{R}^{3\ell \times 3},$$

$$A_i = \begin{pmatrix} \sin(\phi_{i1} + \alpha_i) & -\cos(\phi_{i1} + \alpha_i) & -1 - \cos \phi_{i1} \\ \sin(\phi_{i1} + \phi_{i2} + \alpha_i) & -\cos(\phi_{i1} + \phi_{i2} + \alpha_i) & -1 - \cos(\phi_{i1} + \phi_{i2}) - \cos(\phi_{i2}) \\ \vdots & \vdots & \vdots \\ \sin(\beta_{j_0}^i + \alpha_i) & -\cos(\beta_{j_0}^i + \alpha_i) & -1 - \sum_{k=0}^{l-1} \cos \beta_{jk}^l \end{pmatrix}$$

$$B = \begin{pmatrix} B_1 & 0 & 0 \\ 0 & B_2 & 0 \\ 0 & 0 & B_3 \end{pmatrix} \in \mathbb{R}^{3\ell \times 3\ell},$$

$$B_i = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ \cos \beta_{21}^i & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \cos \beta_{l1}^i & \cos \beta_{l2}^i & \cdots & 1 \end{pmatrix},$$

and where  $\alpha_1 = -\frac{2}{3}\pi$ ,  $\alpha_2 = 0$ ,  $\alpha_3 = \frac{2}{3}\pi$  and  $\beta_{jk}^i := \sum_{h=k+1}^j \phi_{ih}$  and  $R_\theta$  is the rotation matrix by the angle  $\theta$  in the  $xy$ -plane. The control inputs are the angular velocities of the joints, i.e.  $u := \dot{\phi} \in \mathbb{R}^{3\ell}$ . In order to derive the state equation, one defines an input transformation  $u = B(\phi)^{-1}A(\phi)v$ , where  $v$  is a (three dimensional) virtual input. Note that it is always possible since  $B(\phi)$  is regular for all  $\phi$ . Then one obtains a control (state) equation  $\dot{q} = Gv$ , where the control matrix is equal to

$$(2) \quad G = \begin{pmatrix} B(\phi)^{-1}A(\phi) \\ R_\theta \end{pmatrix}.$$

**3.2. Direct and inverse kinematics via CGA.** Now we show how to describe the system exclusively in terms of the conformal geometric algebra. We can view the robot as three ordinary  $(\ell + 1)$ -link snakes joint by their tails such that the tail links has a specific configuration. Following this idea we denote by  $Q_k^0$  the center  $S$  of the body, and by  $Q_k^1, \dots, Q_k^\ell$  the successive joints of the  $k$ -th branch leg,  $k = 1, 2, 3$ . As a central object that describe the state of the system we choose the set of point pairs which represent individual leg links

$$(3) \quad P = (P_1^0, \dots, P_1^\ell, P_2^0, \dots, P_2^\ell, P_3^0, \dots, P_3^\ell).$$

These point pairs are computed in terms of the wedge product in CGA as  $P_k^i := Q_k^i \wedge Q_k^{i+1}$ . On the other hand,  $Q_k^i$  is easily extracted from the pair point by a projection

$$\frac{-\sqrt{P_k^i \cdot P_k^i} + P_k^i}{-e_\infty \cdot P_k^i}.$$

Consequently, we may freely switch between point pairs and points defining their ends. Of course, not all triples of pair points define a state of the robot. In terms of the CGA inner product, the consistency relations read  $Q_j^1 \cdot Q_k^1 = -\frac{3}{2}$  for each  $j \neq k$ , and  $Q_k^i \cdot Q_k^{i+1} = -\frac{1}{2}$  for each  $i = 0, \dots, \ell$ . These equations tell that the joints have constant distance  $\sqrt{3}$  and the length of links is 1 respectively. Implicitly, it also says that  $Q_k^0$  are equal.

Having such an admissible state (3) we can assess the kinematic equations. At first, let us look at the zero position  $q = 0$ . Since  $Q_2^i(0) = [i, 0]$ , the elements in CGA corresponding to  $P_2^i$  are established as

$$(4) \quad \begin{aligned} P_2^i(0) &= (ie_1 + \frac{1}{2}i^2e_\infty + e_0) \wedge ((i+1)e_1 + 2(i+1)^2e_\infty + e_0) \\ &= \frac{1}{2}i(i+1)e_{1\infty} - e_{10} - \frac{1}{2}(2i+1)e_{\infty 0}, \end{aligned}$$

where we have used a shortened notation  $e_{1\infty} = e_1 \wedge e_\infty$  etc. The algebra elements  $P_{1,3}^i(0)$  which correspond to the zero position of links of the first and the third branch leg are obtained from the corresponding links of the first branch by rotation by angle  $\frac{2}{3}\pi$  and  $-\frac{2}{3}\pi$  respectively, i.e.

$$(5) \quad P_{1,3}^i(0) = (\frac{1}{2} \pm \frac{\sqrt{3}}{2}e_{12})P_1^i(0)(\frac{1}{2} \mp \frac{\sqrt{3}}{2}e_{12}).$$

The particular pair points in a general position  $q$  as in (1) are obtained by a translation to  $[x, y]$  composed by a trident body rotation  $\theta$  and a series of rotations of the corresponding leg links by angles  $\phi_{ki}$ . In CGA, it is expressed for each  $k = 1, 2, 3$  and  $i = 0, \dots, \ell$  as a conjugation

$$(6) \quad \begin{aligned} P_k^i(q) &= M_k^i(q)P_k^i(0)\tilde{M}_k^i(q), \\ M_k^i(q) &= R_{\phi_{ki}} \cdots R_{\phi_{k1}}R_\theta T_{x,y}, \end{aligned}$$

where the translation  $T_{x,y}$  and the rotations  $R_{\phi_{ki}}$  are given by

$$\begin{aligned} T_{x,y} &= 1 - \frac{1}{2}(xe_1 + ye_2)e_\infty, \\ R_{\phi_{ki}} &= \cos \frac{\phi_{ki}}{2} - L_{ki} \sin \frac{\phi_{ki}}{2}, \end{aligned}$$

and where the axes of rotations are given by

$$L_{ki} = (1 - \frac{1}{2}Q_k^i e_\infty)e_{12}(1 + \frac{1}{2}Q_k^i e_\infty).$$

Note that we have used the notation  $\phi_{k0} = \theta$  for each  $k = 1, 2, 3$  and note that the procedure is recursive. Namely, given a point pair  $P_k^i$ , we compute the projection  $Q_k^{i+1}$  first, then we compute  $L_{(i+1)k}$ . From this axis we compute  $R_{\phi_{k(i+1)}}$  and then by (6) we get  $P_k^{i+1}$ .

The CGA approach is convenient also for solving problems of the inverse kinematics. In CGA, it can be done in a geometrically very intuitive way due to its easy handling of intersections of geometric objects like spheres, circles, planes. A basic problem is finding the generalized coordinates in terms of a robot position. In our case, having an admissible state (3), we first compute the center  $S = Q_k^0$  by a projection of a point pair  $P_k^0$ , and for each  $i = 1, 2, 3$  we form lines through two consecutive links  $P_k^i$  and  $P_k^{i+1}$ . Then we compute the coordinates via the inner product as

$$\begin{aligned} x &= S \cdot e_1 \\ y &= S \cdot e_2 \\ \cos \theta &= (P_2^0 \wedge e_\infty) \cdot e_{1\infty 0} \\ \cos \phi_{ki} &= (P_k^{i-1} \wedge e_\infty) \cdot (P_k^i \wedge e_\infty) \end{aligned}$$

**3.3. Differential kinematics and singular points via CGA.** Let us now compute the velocity of the direct kinematics, which is obtained by differentiating (6). It is proved in [7] that the total differential of a general kinematic chain

$$\mathcal{O} = R_1 \dots R_n \mathcal{O}(0) \tilde{R}_n \dots R_1$$

containing any geometric object  $\mathcal{O}$  and rotations  $R_1, \dots, R_n$  is equal to

$$d\mathcal{O} = \sum_{j=1}^n [\mathcal{O} \cdot L_j] dq_j,$$



where  $[P \cdot L_j]$  is the inner product of the geometric object (in the actual position) and the axis of the rotation  $R_j$ . This formula follows basically from the fact that each rotation can be expressed as an exponential. But the same is true for translations since we may view each translation as a degenerate rotation, with an ‘axis‘ containing  $e_\infty$ . Hence the formula above holds true also if we allow  $R_i$  to be a generalized rotation, i.e. a rotation or a translation.

In our case, the equation of the direct kinematics is given by (6). A nice consequence of the geometric formulation is that the same equation holds for an arbitrary chosen point  $Q$  attached to  $P_k^i$ , i.e.  $Q = M_k^i(q)Q(0)\tilde{M}_k^i(q)$ . The differentiation of this kinematic chain then yields the following differential formula for any  $Q$  on  $P_k^i$ .

$$(7) \quad \dot{Q} = [Q \cdot e_{1\infty}]\dot{x} + [Q \cdot e_{2\infty}]\dot{y} + \sum_{j=0}^i [Q \cdot L_{k,j}]\dot{\phi}_{k,j}$$

This equation can be seen as an analogue of the classical equation of differential kinematics. Namely, we have a system of a usual form  $\dot{Q} = J\dot{q}$  but the entries of the ‘Jacobi‘ matrix  $J$  are the inner products of a point and an axis and thus belong to the algebra (and not to a field of functions). If  $Q$  is the position of a wheel, then, as the wheels do not slip to the side direction, its velocity is parallel to  $P_k^i$ , which in CGA reads

$$(8) \quad \dot{Q} \wedge P_k^i \wedge e_\infty = 0.$$

Now, if we substitute (7) in (8), we obtain a system of linear ODEs, which can be written in a classical form  $A\dot{q} = 0$ , where the Pfaff matrix  $A$  is given by

$$A_{ij} = J_{ij} \wedge P_i \wedge e_\infty.$$

In our particular case, the wheels located at link centers  $Q = P_k^i e_\infty \tilde{P}_k^i$ , for each  $i = 1, \dots, \ell - 1$ , and at the end of each of the leg branches  $Q = Q_k^\ell$ . Hence we get a system of  $3\ell$  first order differential equations with  $3\ell + 3$  variables. It is easy to see that each  $A_{ij}$  is a multiple of  $(e_3)^*$ . Thus the Pfaff equation  $A\dot{q} = 0$  can be solved for  $A$  considered as a matrix over the field of functions. Let us also remark that the particular position of wheels does not play any role formally. The equations (7) and (8) are valid for any position. The positions of wheels influence the inner products with joint axis and consequently the matrices  $J$  and  $A$ .

At the end of this section, let us discuss the postures of the robot which are critical for the control. Such critical (singular) postures occur when the velocity constraints degenerate. It is in the cases when two or more wheel axles are either parallel or intersect in one point. Obviously, the former singular postures coincide with the latter if the center of rotation is in infinity. Thus, in CGA, we have one equation describing such singular postures. Denoting by  $o_1, o_2, o_3$  the wheel axles it reads

$$(9) \quad o_1 \vee o_2 \vee o_3 = 0,$$

where  $\vee$  is the meet operation in CGA. We can equivalently express this equation in terms of the geometric product as  $\langle o_1 o_2 o_3 \rangle_1 = 0$ . If we denote by  $Q$  the position of a wheel attached to the link  $P_k^i$ , then its axle  $o$  is computed by  $o = R_{\pi/2} P_k^i \tilde{R}_{\pi/2} \wedge e_\infty$ , where the rotation is given by the formula

$$R_{\pi/2} = (1 - \frac{1}{2}Qe_\infty)(\frac{\sqrt{2}}{2} - \frac{\sqrt{2}}{2}e_{12})(1 + \frac{1}{2}Qe_\infty).$$

#### 4. 1-LINK TRIDENT SNAKE ROBOT

Here we demonstrate the theory on the simplest nontrivial case, i.e. the case  $\ell = 1$ . For more details see [14].

**4.1. Kinematics.** For simplicity we omit the upper index denoting the first link from now on. By (4) and (5), the zero position of point pairs associated to the leg links is given by

$$P_2(0) = e_{1\infty} - e_{10} - \frac{3}{2}e_{\infty 0},$$

$$P_{1,3}(0) = -\frac{1}{2}e_{1\infty} + \frac{1}{2}e_{10} \pm \frac{\sqrt{3}}{2}e_{2\infty} \mp \frac{\sqrt{3}}{2}e_{20} - \frac{3}{2}e_{\infty 0}.$$

By (6), the general position of a point pair  $P_k$  is given by

$$P_k = R_{\phi_{k1}} R_{\theta} T_{x,y} P_k(0) \tilde{T}_{x,y} \tilde{R}_{\theta} \tilde{R}_{\phi_{k1}}.$$

The same kinematic chain holds for any point  $Q$  on  $P_k$ . Thus, for the leg ends  $Q = (Q_1, Q_2, Q_3)^T$ , we get a differential equation  $\dot{Q} = J\dot{q}$ , where

$$(10) \quad J = \begin{pmatrix} Q_1 \cdot e_{1\infty} & Q_1 \cdot e_{2\infty} & Q_1 \cdot L_0 & Q_1 \cdot L_1 & 0 & 0 \\ Q_2 \cdot e_{1\infty} & Q_2 \cdot e_{2\infty} & Q_2 \cdot L_0 & 0 & Q_2 \cdot L_2 & 0 \\ Q_3 \cdot e_{1\infty} & Q_3 \cdot e_{2\infty} & Q_3 \cdot L_0 & 0 & 0 & Q_3 \cdot L_3 \end{pmatrix}.$$

Obviously, the nonholonomic constraints read  $Q_i \wedge P_i \wedge e_{\infty} = 0$ . It leads to a Pfaff constraint  $A\dot{q} = 0$  with  $A_{ij} = J_{ij} \wedge P_i \wedge e_{\infty}$  and its solution in a point gives a control system  $\dot{q} = G\mu$ , where the control matrix  $G$  is a  $6 \times 3$  matrix spanned by vector fields  $g_1, g_2, g_3$ , where

$$g_1 = \cos \theta \partial_x + \sin \theta \partial_y + \sin \phi_1 \partial_{\phi_1} + \sin(\phi_2 + \frac{2\pi}{3}) \partial_{\phi_2} + \sin(\phi_3 + \frac{4\pi}{3}) \partial_{\phi_3},$$

$$g_2 = \sin \theta \partial_x + \cos \theta \partial_y - \cos \phi_1 \partial_{\phi_1} - \cos(\phi_2 + \frac{2\pi}{3}) \partial_{\phi_2} - \cos(\phi_3 + \frac{4\pi}{3}) \partial_{\phi_3},$$

$$g_3 = \partial_{\theta} - (1 + \cos \phi_1) \partial_{\phi_1} - (1 + \cos \phi_2) \partial_{\phi_2} - (1 + \cos \phi_3) \partial_{\phi_3}.$$

It is easy to check that it agrees with the general result (2) obtained by the euclidean geometry. By a direct computation, one can also check that equation (9) describing singular points gives exactly the equation in [9, Remark 1].

By [9], these vector fields define a bracket generating distribution with growth vector (3,6) in regular points. It means that in each regular point the vectors  $g_1, g_2, g_3$  together with their Lie brackets span the whole tangent space. Consequently, the system is controllable by Chow–Rashevsky theorem.

**4.2. CLUCalc implementation.** The proposed trident snake control was tested in CLUCalc software, [2, 5], which is designed exactly for calculations in arbitrary predefined geometric algebra. The following code piece contains the definition of the initial position:

```
// INITIAL POSITION
S0=VecN3(0,0,0);
LB0=VecN3(0,0,1);
R=RotorN3(0,0,1,2*Pi/3);
// Joints
pb10=VecN3(1,0,0);
pb20=R*pb10*~R;
pb30=R*pb20*~R;
// Axes
L10=TranslatorN3(pb10)*LB0*TranslatorN3(-pb10);
L20=R*L10*~R;
L30=R*L20*~R;
// Ends of legs
p10=VecN3(2,0,0);
p20=R*p10*~R;
p30=R*p20*~R;
```

The initial position is thus recalculated with respect to the controlling parameters change to

get a current position. The code we demonstrate corresponds to the body and the first leg of the trident snake robot. The other legs are computed in the same way.

```

T=TranslatorN3(x,y,0);
// BODY
// Center
S=T*S0*~T;
// Axis
LB=T*LB0*~T;
// Motor
MB=TranslatorN3(LB)*RotorN3(0,0,1,d)*~TranslatorN3(LB);

// FIRST LEG
// Joint
:Blue;
:pb1=MB*T*pb10*~T*~MB;
// Axis
L1=MB*T*L10*~T*~MB;
// Motor
M1=TranslatorN3(L1)*RotorN3(0,0,1,a)*TranslatorN3(-L1);
// End
:Black;
:p1=M1*MB*T*pb10*~T*~MB*~M1;

```

The following three sets of pictures demonstrate the evolution from 0 in the direction of  $g_1, g_2$  and  $g_3$  vector fields.

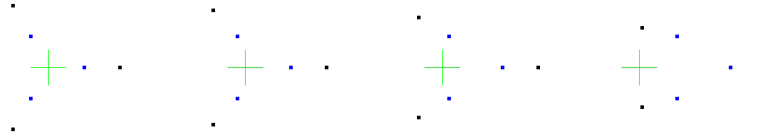


FIGURE 2.  $g_1$  direction (pictured by CLUCalc)

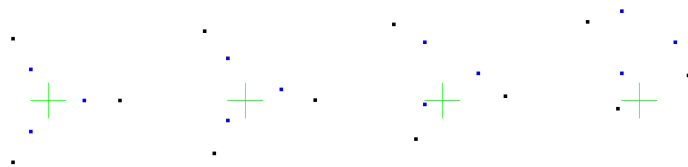


FIGURE 3.  $g_2$  direction (pictured by CLUCalc)



FIGURE 4.  $g_3$  direction (pictured by CLUCalc)

## REFERENCES

- [1] L. Gonzalez–Jimenez, O. Carbajal–Espinosa, A. Loukianov, and E. Bayro–Corrochano, *Robust Pose Control of Robot Manipulators Using Conformal Geometric Algebra*, *Advances in Applied Clifford Algebras* **24,2** (2014), 533–552.

- [2] D. Hildenbrand, *Foundations of Geometric Algebra Computing*, Springer, Geometry and Computing, Vol. 8, 2013.
- [3] P. Liljebäck, K.Y. Pettersen, Ø. Stavdahl and J.T. Gravdahl, *Snake Robots, Modelling, Mechatronics and Control*, Springer, Advances in Industrial Control, 2013.
- [4] R. M. Murray, L. Zexiang and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 1994.
- [5] Ch. Perwass, *Geometric Algebra with Applications in Engineering*, Springer, Geometry and Computing, Vol. 4, 2009.
- [6] J.M. Selig, *Geometric Fundamentals of Robotics*, Springer, Monographs in Computer Science, 2004.
- [7] J. Zamora–Esquivel and E. Bayro–Corrochano, *Kinematics and diferential kinematics of binocular robot heads*, Robotics and Automation, ICRA 2006.
- [8] J. Zamora–Esquivel and E. Bayro–Corrochano, *Parallel Forward Dynamics: a geometric approach*, In *The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.
- [9] M. Ishikawa, *Trident snake robot: locomotion analysis and control*, In: Proc. IFAC NOLCOS, pp. 1169–1174. Stuttgart, Germany 2004
- [10] M. Ishikawa, Y. Minami, T. Sugie, *Development and control experiment of the trident snake robot.*, In: Proc. 45th IEEE CDC, pp. 6450–6455. San Diego, CA, 2006
- [11] M. Ishikawa, Y. Minami, T. Sugie, *Development and control experiment of the trident snake robot.* IEEE/ASME Trans. Mechatronics 15, 9–15, 2010
- [12] J. Hrdina, A. Návrát, P. Vašík, *3-link robotic snake control based on CGA*, Advances in Applied Clifford Algebras (accepted, 2015)
- [13] J. Hrdina and P. Vašík, *Notes on differential kinematics in conformal geometric algebra approach*, Recent Advances in Soft Computing, Advances in Intelligent Systems and Computing, Springer, volume 378., pages 363–374, 2015.
- [14] R. Matoušek and A. Návrát, *Trident snake control based on CGA*, Recent Advances in Soft Computing, Advances in Intelligent Systems and Computing vol 378, Springer, 2015, pp 375–385.



## A Conformal Geometric Algebra framework for Mixed Reality and mobile display

Margarita Papaefthymiou<sup>a</sup>, George Papagiannakis<sup>a</sup>, Andreas Aristidou<sup>b</sup> and Marinos Ioannides<sup>b</sup>

<sup>a</sup>Foundation for Research and Technology – Hellas and  
University of Crete, Heraklion, Greece  
mpapae02@ics.forth.gr [presenter], papagian@ics.forth.gr [corresponding]

<sup>b</sup>Department of Electrical Engineering, Computer Engineering and Informatics  
Cyprus University of Technology, Limassol, Cyprus  
a.aristidou@ieee.org, marinos.ioannides@cut.ac.cy

Over the last few years, recent advances in user interface and mobile computing, introduce the ability to create new experiences that enhance the way we acquire, interact and display information within the world that surrounds us with virtual characters [1], [2]. Virtual Reality (VR) is a 3D computer simulated environment that gives the user the experience of being physically present in real or computer-generated worlds; on the other hand, Augmented Reality (AR) is a live direct or indirect view of a physical environment whose elements are augmented (or supplemented) by computer-generated sensory inputs. Both technologies use interactive devices to achieve the optimum adaptation of the user in the immersive world achieving enhanced presence [1], harnessing latest advances in computer vision, glasses or head-mounted-displays featuring embedded mobile devices. A common issue in all of them is interpolation errors while using different linear and quaternion algebraic methods when a) tracking the user's position and orientation (translation and rotation) using computer vision b) tracking using mobile sensors c) using gesture input methods to allow the user to interactively edit the augmented scene (translation, rotation and scale) d) animation blending of the virtual characters that augmented the mixed reality scenes (translation and rotation).

In this proposed talk, we aim to enhance the conformal model of Geometric Algebra (CGA) [3], [4] as the mathematical background for camera, display and character animation control [2] in immersive and virtual technology, such as head-mounted displays (e.g. Google Cardboard™) or modern smartphones; a framework that offers a smooth and stable calibration/control can be used in real-time mobile mixed reality systems that featured realistic, animated virtual human actors who augmented real environments. The conformal model of Geometric Algebra is a mathematical framework that provides a convenient mathematical notation for representing orientations and rotations of objects in three dimensions, a compact and geometrically intuitive formulation of algorithms, and an easy and immediate computation of rotors; CGA extends the usefulness of the 3D GA by expanding the class of rotors to include translations, dilations and inversions. Rotors are simpler to manipulate than Euler angles; they are more numerically stable and more efficient than rotation matrices, avoiding the problem of gimbal lock. The results of this work allow us to a) unify and improve the performance of previously separated linear and quaternion algebra camera transformations b) fully replace quaternions for rotation interpolation with faster CGA rotors, c) blend rotations and translations between character animations using CGA, under a single geometric algebraic framework using CGA for Mixed Reality applications.

### REFERENCES

- [1] Egges, A., Papagiannakis, G., and Magnenat-Thalmann, N. (2007). Presence and interaction in mixed reality environments. *Visual Computer* 23, 5, 317–333.
- [2] Papagiannakis, G., Elissavet, G., Trahanias, P., and Tsioumas, M. (2014). A Geometric Algebra Animation Method for Mobile Augmented Reality Simulations in Digital Heritage Sites. In: *Digital Heritage. Progress in Cultural Heritage: Documentation, Preservation, and Protection*. Springer International Publishing, 258–267.
- [3] Hestens, D., Sobczyk, G. (1984). *Clifford Algebra to Geometric Calculus: A Unified Language for Mathematics and Physics*. Reidel, Dordrecht
- [4] Aristidou, A., Lasenby, J. (2011). Inverse Kinematics solutions using Conformal Geometric Algebra, In L. Dorst and J. Lasenby (Eds), *Guide to Geometric Algebra in Practice*, Springer Verlag.



# CLIFFORD ALGEBRA REPRESENTATION OF GRASPING AND MANIPULATIVE HAND ACTIONS FOR KINEMATIC SYNTHESIS

Alba Perez-Gracia<sup>a</sup> and Federico Thomas<sup>b</sup>

<sup>a</sup> Department of Mechanical Engineering  
Idaho State University, Pocatello, ID, USA  
perealba@isu.edu

<sup>b</sup> Institut de Robotica i Informatica Industrial (CSIC-UPC)  
Barcelona, Spain  
thomas@iri.upc.edu

**ABSTRACT.** The dimensional synthesis of wristed, multi-fingered hands can be used for simultaneous tasks of all fingertips. When defining unconstrained positions for the fingers, the synthesis is solved by ensuring the desired displacement for each branch of the hand. The displacements and related geometric objects are expressed in the even Clifford subalgebra  $\mathcal{C}_{0,3,1}^+$ . If velocities are also defined, the corresponding Lie algebra elements, defined on the same algebra, are also equated to the linear combination of joint twist; accelerations can be defined in a similar way.

This approach successfully captures the independent motion of each fingertip. However, when the design task includes holding and manipulating an object, the constraints between the fingertips need to be considered too.

In this work the conditions to create a hand motion compatible with grasping and moving a given object, expressed in the Clifford algebra, are shown. The design methodology is outlined and some of the simple grasping and manipulation cases are presented.

## 1. INTRODUCTION

The design of end-effector robotic tools has focused on three different strategies [10], which yield very different designs: anthropomorphism, designing for grasping tasks, and designing for dexterous manipulation. Hands for in-hand manipulation tend to be more complex, especially if a wide range of manipulation actions are targeted, while underactuation is targeted for grasping and limited manipulation [11], [15], [5]. In this research, we focus on creating multi-fingered hand designs specifically tailored to desired groups of manipulation tasks.

We define a multi-fingered robotic hand as a series of common joints branching at least once in several other serial chains (the fingers or branches), ending in a finite set of end-effector links (the fingertips). Recently a methodology has been developed for the design of new multi-fingered hands for kinematic tasks [20], both for finite and infinitesimal motion [21]. This methodology offers a systematic process to design innovative end-effectors for a simultaneous task of all the fingertips [8]. However the design for manipulation of a grasped objects requires a more careful strategy.

When the hand grasps and object, the constraints on the relative motion among fingers need to be taken into account; the kinematic topology is switched from a tree topology to a hybrid topology. In this application, we focus on a simple type of in-hand manipulation, for a given fingertip contact point [6] and some specified relative motion, compatible with the contact. The mobility of the grasped object can be calculated for the general case using mobility formulas, or using grasp or Jacobian matrix techniques [3] if the hand kinematics and object geometry are at least partially known.



Given a hand topology and the mobility for a generally-grasped object, a series of positions and subspaces of potential velocities are defined so that some properties of the grasp are being checked while allowing the manipulation of the object. These positions and velocities are expressed using the even Clifford subalgebra  $\mathcal{C}_{0,3,1}^+$ . Several authors have used a Clifford or geometric algebra to define displacements and velocities of rigid bodies in robotic systems, among others [7], [19], [1] or [2]. In particular, the analysis of contacts using Grassman-Cayley algebra was developed in [22], the definition of contacts for synthesis problem of planar linkages was developed in [12], and the analysis and planning of grasping using geometric algebra has been studied in [23].

It is expected that the use of the Clifford algebra will allow a more compact and more homogeneous expression of the grasping and manipulation actions including finite positions and its derivatives, for their use in the design of innovative robotic hands.

## 2. DISPLACEMENTS, VELOCITIES AND FORCES

Let  $\mathcal{C}_{0,3,1}^+$  be the even Clifford subalgebra of the projective space  $\mathbb{P}^4$  with the degenerated scalar product. Starting with the basis vectors  $\{e_1, e_2, e_3, e_4\} \in \mathbb{P}^4$ , the well-known notation for the even blades,

$$(1) \quad \begin{aligned} e_{23} &= i, e_{31} = j, e_{12} = k, \\ e_{41} &= i\varepsilon, e_{42} = j\varepsilon, e_{43} = k\varepsilon, \\ e_{1234} &= \varepsilon \end{aligned}$$

is used along this work.

Consider a general element of this algebra as  $A = a_0 + a_1i + a_2j + a_3k + \varepsilon(a_4i + a_5j + a_6k + a_7)$ , and the geometric product of two 1-vectors as the sum of the inner product and the exterior product,  $ab = a \cdot b + a \wedge b$ .

The *conjugation* is defined for blades as  $(e_1e_2 \dots e_k)^* = (-1)^k e_k \dots e_2e_1$ ; for scalars,  $1^* = 1$  and for basis vectors,  $e_i^* = -e_i$ .

The norm of an element is  $\|A\|^2 = a + \varepsilon a^0 = AA^*$  and it has nonzero scalar and dual part. For unit elements,  $\|A\|^2 = 1$ . The inverse of an element is defined as  $A^{-1} = A^* / \|A\|^2$ , so that for unit elements, such as displacements, the inverse is  $A^{-1} = A^*$ .

A point is defined as  $p = 1 + \varepsilon(p_xi + p_yj + p_zk)$ , and a line is defined as  $L = l_xi + l_yj + l_zk + \varepsilon(l_x^0i + l_y^0j + l_z^0k)$ , being such that  $LL^* = 1$ .

A finite displacement is a unit element of the subalgebra, and can be expressed as a function of the invariants of the displacement, the screw axis  $S$  and the rotation  $\phi$  and slide  $t$  about and along the axis. In particular,

- A translation of magnitude  $d$  along a direction  $\mathbf{s}$  is  $D = 1 + \frac{d}{2}\varepsilon(s_xi + s_yj + s_zk)$
- A rotation of magnitude  $\phi$  and rotation axis  $\mathbf{s}$  is  $R = \cos \frac{\phi}{2} + \sin \frac{\phi}{2}(s_xi + s_yj + s_zk)$
- A general displacement of screw axis  $S = \mathbf{s} + \varepsilon s^0$ , rotation  $\phi$  and slide  $t$  is  $Q = DR = \cos \frac{\phi}{2} + \sin \frac{\phi}{2}(s_xi + s_yj + s_zk) + \varepsilon((\sin \frac{\phi}{2}s_x^0 + \frac{t}{2}\cos \frac{\phi}{2}s_x)i + (\sin \frac{\phi}{2}s_y^0 + \frac{t}{2}\cos \frac{\phi}{2}s_y)j + (\sin \frac{\phi}{2}s_z^0 + \frac{t}{2}\cos \frac{\phi}{2}s_z)k) - \frac{t}{2}\sin \frac{\phi}{2}$

**2.1. Twists as 2-vectors.** In order to define the velocities, we consider the differentiation of the action of a finite displacement on a geometric element  $x$  expressed in the moving frame,

$$(2) \quad \dot{X} = \dot{Q}xQ^* + Qx\dot{Q}^*,$$

which leads to

$$(3) \quad \dot{X} = (\dot{Q}Q^*)X + X(\dot{Q}Q^*)^*.$$

Define

$$(4) \quad V = 2\dot{Q}Q^*.$$

Recalling that  $Q$  needs to be a unit element in order to be a displacement, and taking derivatives in the unit condition we obtain that  $\dot{Q}Q^* = -(\dot{Q}Q^*)^*$ , which makes  $V$  a pure element, that is, an element with zero scalar and pseudoscalar components. This element is also denoted a *2-vector*, as it is a linear combination of 2-blades only.

Using the definition in Eq.(4), the derivative becomes

$$(5) \quad \dot{X} = \frac{1}{2}(VX + XV^*).$$

The definition in (4) can be arranged as

$$(6) \quad \dot{Q} = \frac{1}{2}VQ,$$

which coincides with that in [7]. The element  $V = w + \varepsilon v$  is the *twist*, which describes the velocity of the rigid body, as the angular velocity of the body and the linear velocity of a point of the body. For the calculations above, the point is the origin of the fixed frame as considered part of the moving body.

If  $s = s_x i + s_y j + s_z k$  and  $s^0 = s_x^0 i + s_y^0 j + s_z^0 k$ , the computation of  $V$  using the expression of displacement  $Q$  in (4) yields

$$(7) \quad V = \dot{\phi}J = \dot{\phi}(s + \varepsilon(s^0 + hs)).$$

Here  $S = s + \varepsilon s^0$  is the line defining the screw axis of the displacement, corresponding to the minimal motion for the finite displacement. The magnitude  $h = \frac{t}{\phi}$  is called the *pitch*. This 2-vector that contains both the angular velocity of the rigid body and the linear velocity of the origin point, can be immediately identified with the six-dimensional twist  $V = (\omega, \mathbf{v})$  defined in screw theory, while  $J$  would correspond to the unit twist, or screw. [19].

Due to the different action used for lines and points (and planes), the conjugate 2-vector in Eq.(5) becomes, for directions and lines,

$$(8) \quad V^* = \dot{\phi}J^* = \dot{\phi}(-s - \varepsilon(s^0 + hs)),$$

and for points and planes,

$$(9) \quad V^* = \dot{\phi}(-s + \varepsilon(s^0 + hs)),$$

The Lie algebra  $se(3)$  can be built on this geometric algebra if we consider the 2-vectors and define the commutator product of two elements  $V_1, V_2$  of the algebra as

$$(10) \quad [V_1, V_2] = \frac{1}{2}(V_1V_2 - V_2V_1),$$

which is closed for the pure elements. Using the commutator, we can define

$$(11) \quad V_1V_2 = [V_1, V_2] + V_1 \wedge V_2 + V_1 \cdot V_2.$$

Notice that  $[V_1, V_2]$  yields the dual cross product used in the dual vector calculus [18], while the exterior and inner product yield the minus dual dot product.

Integrating Eq.(6) and considering  $V$  as constant, we obtain the finite displacement as the exponential of the twist, defined as a power series, starting at the identity.

$$(12) \quad Q = e^{\frac{1}{2}Vt} = e^{J\frac{\phi}{2}}$$

if we consider  $\dot{\phi} = \phi/t$ .

This derivation can be found in detail for instance in [13] and it is used to create the forward kinematics of serial robots as the Clifford product of exponentials.

**2.2. Linear property of 2-vectors.** The Clifford algebra is not a graded multi vector algebra, however it is a graded vector space [4]: it can be decomposed as sum of linear subspaces of homogeneous grade. The twists, or 2-vectors, form a vector subspace within the Clifford algebra over the scalars (0-vectors) and also over the *dual scalars*, which are the multi vectors constructed with elements of degree zero and degree four,  $K = k_0 + \varepsilon k_7$  [14]. The addition and product by scalar and pseudoscalar elements yields a 2-vector in all cases,

$$(13) \quad K_1 V_1 + K_2 V_2 = k_{10} w_1 + k_{20} w_2 + \varepsilon(k_{10} v_1 + k_{20} v_2 + k_{17} w_1 + k_{27} w_2).$$

In summary, considering either scalars or dual scalars, the 2-elements, which we identify with screws for both twists and wrenches, can form vector subspaces in the Clifford algebra. We can see the finite screw systems as vector subspaces formed by elements of degree 2 of the Clifford algebra.

**2.3. Wrenches as reciprocal screws.** The 2-vectors are used to express the twist defined as before,  $W = w + \varepsilon v$  and also the wrench  $F = m + \varepsilon f$ , where  $m = m_x i + m_y j + m_z k$  is the resultant moment and  $f = f_x i + f_y j + f_z k$  is the resultant force at a given point of the rigid body. The reciprocal product is defined in screw theory as  $W * F = w \cdot m + v \cdot f$ ; when this scalar quantity is zero, it is said that the twist and the wrench are reciprocal.

Given a wrench or wrench subspace, representing the contact forces on a body, there exists a reciprocal subspace of twists for the potential velocities allowed for the body. This reciprocity, which exists at the level of first derivatives and for convex and polygonal objects, is used in this work to define the grasping and manipulation actions as a function of the twists.

The inner and outer products in Eq.(11) yield the scalar and pseudoscalar,

$$(14) \quad \begin{aligned} W \cdot F &= -\mathbf{w} \cdot \mathbf{m}, \\ W \wedge F &= -(\mathbf{w} \cdot \mathbf{f} + \mathbf{v} \cdot \mathbf{m})\varepsilon \end{aligned}$$

### 3. KINEMATICS OF GRASPING FOR TREE TOPOLOGIES

**3.1. End-effector twist and Jacobian.** Let  $S_1$  to  $S_n$  be the ordered  $n$  axes of a serial chain, and  $J_1, \dots, J_n$  the corresponding unit screws, in which the pitch  $h = t/\theta$  is used to identify either a prismatic or a revolute joint. The product of exponentials

$$(15) \quad Q = e^{\frac{\theta_1}{2} J_1} \dots e^{\frac{\theta_n}{2} J_n}$$

yields the relative motion of the rigid body attached to the last joint (the end effector) with respect to a reference configuration. These are the relative forward kinematics equations. The Jacobian matrix of the serial chain can be derived by finding the twist of the end effector as in (4), noticing that in the product of exponentials, only the joint variables  $\theta_i$  are a function of time,

$$(16) \quad V = 2\dot{Q}\tilde{Q} = 2\left(\sum_{i=1}^n \frac{\partial Q}{\partial \theta_i} \dot{\theta}_i\right)\tilde{Q} = 2\sum_{i=1}^n \left(\frac{\partial Q}{\partial \theta_i} \tilde{Q}\right)\dot{\theta}_i.$$

If we expand the above calculation, we obtain that for each axis,

$$(17) \quad \left(\frac{\partial Q}{\partial \theta_i} \tilde{Q}\right) \dot{\theta}_i = \frac{1}{2} e^{J_1 \frac{\theta_1}{2}} \dots e^{J_{i-1} \frac{\theta_{i-1}}{2}} J_i e^{-J_{i-1} \frac{\theta_{i-1}}{2}} \dots e^{-J_1 \frac{\theta_1}{2}} \dot{\theta}_i$$

Notice that this is the action of the displacement of the preceding joints, up to the  $i-1$ <sup>th</sup> joint, on the  $i$ <sup>th</sup> joint, for the current configuration of the serial chain. Denote as  $J'_i = Q_{i-1} J_i \tilde{Q}_{i-1}$  the current position of the screw, with  $Q_{i-1}$  being the displacement due to the previous joints to joint  $i$ ; the end effector twist becomes

$$(18) \quad V = \sum_{i=1}^n J'_i \dot{\theta}_i.$$

Notice that the expression of the screw  $J'_i$  is parameterized by the previous  $i-1$  joint variables.

The twist vector  $V$  is a linear combination of unit twists times scalar joint rates, and contains the feasible velocities and angular velocities of the end-effector with respect to the fixed frame. When written in matrix form with the screws as columns, they form the *fixed-frame Jacobian* or *spatial Jacobian* matrix of the serial robot.

When the Jacobian matrix is to be created for a particular geometric entity  $X$ , consider  $\dot{X} = \frac{1}{2}(VX + XV^*) = V \times X$ , which yields

$$(19) \quad \dot{X} = \left(\sum_{i=1}^n J'_i \dot{\theta}_i\right) \times X = \sum_{i=1}^n (J'_i \times X) \dot{\theta}_i$$

with the columns of the Jacobian matrix modified for the particular geometric element to consider. The derivation of the Jacobian can be found for instance in [1]. For wristed robotic hands with a tree topology, the same derivation can be made, in which some of the joints are common to some of the branches.

**3.2. Grasp analysis.** Grasping consists on locating several end effectors in contact with the surface of an object, so that the forces applied at the contact points ensure some desired resultant force property on the object, force closure being one of them. A grasp is *force closed* if it can balance any external wrench applied at the object. In order to analyze the grasping, it is then important to look at the static forces exerted or their reciprocal potential velocities.

Let  $V_i$  be the 2-vector for the twist of the end effector  $i$ , and  $F_i$  the reciprocal 2-vector corresponding to the wrench of the end effector  $i$ , and let us consider  $b$  end effectors able to exert contact forces on the object. Then we need to impose

$$(20) \quad \sum_{i=1}^b c_i F_i + F = 0,$$

where  $F$  is the external wrench on the object, and  $c_i \geq 0$  are the scalars defining a positive grasp, all transformed to either the fixed frame or to an object frame. Figure 1 shows the typical local finger frame and the object frame in which the force balance is stated.

The force closure can be expressed using reciprocal twists. A *form-closed grasp* is that in which the space of feasible velocities is zero. Form and force-closed grasps coincide for polyhedral, convex objects; for other cases, the curvature of the object needs to be taken into account for the form-closed grasp.

For a pointy finger with no friction, we define a local frame such that  $F_i = c_i k \epsilon$  with  $c_i \geq 0$ . In this frame, the  $z$ -axis is pointing towards the object, as shown in Figure 1. It has been proved that for a general object, seven fingers are needed with positive forces

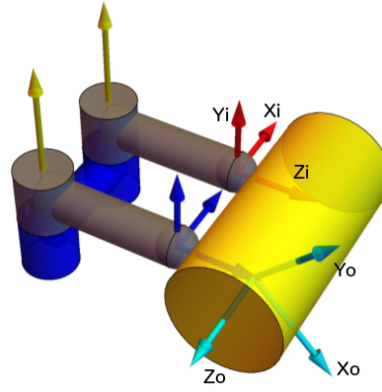


FIGURE 1. Two fingers in contact with a convex object and their respective finger frames.

in order to ensure force-closed grasp. Obviously, if the scalar  $c_i$  is allowed to take any value, then six fingers are needed to balance any external wrench. When friction is considered, the number of fingers can be further reduced.

The wrench of each finger can be transformed to an object frame (located for instance at the center of mass) as  $F_{oi} = Q_i F_i Q_i^*$ , with  $Q_i$  the displacement from the finger frame to the object frame.

**3.3. Kinematic model of a grasping hand.** The definition of the number and position of the contact points and the contact forces created on the object allows us to analyze several aspects of the grasp. Assuming that some grasp synthesis method (see for instance [16]) is used to compute those points, then the next step is to analyze whether the end-effectors of the robotic hand (usually the fingertips) can reach the desired positions and whether the needed forces can be applied.

If a robotic hand is represented as a tree graph as shown in Figure 2, then the robotic hand grasping an object becomes a hybrid graph, in which the fingertip contacts can be modeled as different types of joints. In Figure 2, the robotic hand in the left has five fingertips, two palms and a wrist. The numbers on the edges denote the number of joints in the serial chain. The graph in the right shows the same hand grasping an object (square box) with the five fingertips and with contact also in one of the palms. The contact joint is denoted as F.

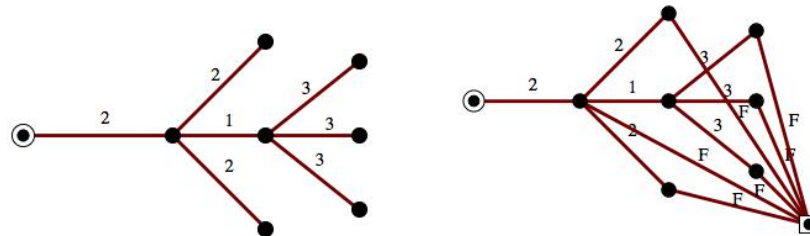


FIGURE 2. A five-fingered robotic hand, left; grasping an object, right.

Several fingertip contacts are considered, depending on the modeling of the friction. The simplest fingertip is what is called the *pointy finger*, in which a point contact with no friction is considered. The description of the different fingertip contacts can be found for instance in [17].

Mason [9] defines, following Salisbury, contact types and the mobility and connectivity (relative mobility of two links) of a robotic hand with several fingers using the well-known Kutzbach-Gruebler formula,  $M$  for the mobility when the finger joints are allowed to move, and  $M'$  for the mobility when the finger joints are locked, which will give the subspace of twists of the object when the hand is trying to immobilize the object in a grasp. In addition we define  $M_w$  as the mobility of the hand minus the degrees of freedom of the common joints up to the first split. These values are used a priori to select appropriate hand topologies for a given task.

#### 4. GRASPING AND MANIPULATION ACTIONS FOR SYNTHESIS

For this work we consider convex objects and spherical fingertips, and use twists in order to define the grasp as potential velocities allowed on the contact.

Kinematic tasks can be defined for the following situations:

- Tasks in which the fingertips are compatible with the object geometry and the object motion, without changing the position of the fingertips. The contact point does not change.
- Tasks of rolling on the object surface, for a moving object.
- Tasks of sliding on the object surface, for a moving object.

In all these situations, force equilibrium can be enforced. This work focuses on the first two points for particular geometries.

Assume that the geometry of the convex object is known and has principal curvatures  $\rho_{ai}$  and  $\rho_{bi}$  at the contact point with finger  $i$ ,  $P_i$ . Each point has a corresponding *surface frame*, with the local  $z$  axis is directed towards the object and the local  $x$  and  $y$  axes at the plane tangent to the surface and correspond to the directions of maximum and minimum curvature, and so that they form a direct trihedron.

Consider the fingertip as the center of a sphere of radius  $r$ , for a hand with  $b$  fingers. At the reference configuration, the local frames of the fingers are translated in the negative  $z$  direction from their corresponding surface frames, with the origin of these local frames located at the center of the sphere corresponding to each fingertip, see Figure 3.

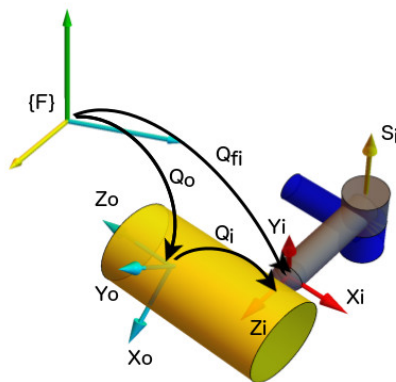


FIGURE 3. Transformation to finger frame.

An arbitrarily-located moving frame is attached to the object to be manipulated. The contact points  $P_1, \dots, P_b$  on the object can be calculated using a grasp synthesis method and are given as local displacements from the object frame to the surface frame,  $Q_i$ , with  $i = 1, \dots, b$ . The end-effector for each finger  $i$  must be then located as

$$(21) \quad Q_{fi} = Q_o Q_i \left(1 - \frac{r_i}{2} k \varepsilon\right), \quad i = 1, \dots, b,$$

where  $Q_o$  is the known position of the object.

**4.1. Motion of the object with fixed contact points.** In this simplest case of motion, the transformations  $Q_1, \dots, Q_b$  are fixed; no rolling, sliding or losing of contact is allowed for the fingers. If nonfiction is assumed, then the rotation about the  $z$  axis,  $Q_{Rz} = \cos \frac{\phi}{2} + \sin \frac{\phi}{2} z$ , is allowed.

For an  $m$ -position synthesis task, with  $m = m_p + m_v$  for finite displacement and velocities, define  $m_p$  positions of the object,  $Q_o^1, \dots, Q_o^m$ , the simultaneous task for each finger is

$$(22) \quad \begin{aligned} Q_{fi}^j &= Q_o^j Q_i (1 - \frac{r_i}{2} k \epsilon), & j &= 1, \dots, m_p, \\ & & i &= 1, \dots, b. \end{aligned}$$

Usually the synthesis uses relative positions with respect to a reference configuration, taken as the first position. In this case, the relative position for each finger is identical,

$$(23) \quad \begin{aligned} Q_{fi}^{1j} &= Q_{fi}^j (Q_{fi}^1)^* = Q_o^j (Q_o^1)^* = Q_o^{1j}, & j &= 2, \dots, m_p, \\ & & i &= 1, \dots, b, \end{aligned}$$

that is, considering the fingers as rigidly attached to the object. This is similar to the synthesis problem for parallel robots, in which each of the legs of the robot has to reach the relative motion of the platform. The finite-position synthesis is possible in this case if the mobility of the hand-object system is positive.

A task with object velocities can be transformed to a task with positions or positions and velocities at the fingertips. Consider the object twist  $V_o^j$  associated to the object displacement  $Q_o^j$ . This fully defines the velocities of the fingertips, as

$$(24) \quad \begin{aligned} V_{fi}^j &= V_o^j, & j &= 1, \dots, m_v, \\ & & i &= 1, \dots, b. \end{aligned}$$

The velocity of the origin of the finger frame can be easily calculated from the object twist and the displacement as

$$(25) \quad v_i = v_0 + w \times p_i.$$

**4.2. Motion of the object with sliding fingers.** In this case, the transformations  $Q_1^j, \dots, Q_b^j$  are to be specified for each position  $j$  in such a way that they are compatible with the geometry of the object. The sliding allows to keep the same contact point in the spherical finger, so that the transformation from the surface to the finger frame is constant,  $(1 - \frac{r_i}{2} k \epsilon)$ .

For an  $m$ -position synthesis task, define again  $m_p$  positions of the object,  $Q_o^1, \dots, Q_o^m$ , the simultaneous task for each finger is

$$(26) \quad \begin{aligned} Q_{fi}^j &= Q_o^j Q_i^j (1 - \frac{r_i}{2} k \epsilon), & j &= 1, \dots, m_p, \\ & & i &= 1, \dots, b. \end{aligned}$$

The relative position with respect to the first position for each finger is

$$(27) \quad \begin{aligned} Q_{fi}^{1j} &= Q_o^j Q_i^j (Q_i^1)^* (Q_o^1)^*, & j &= 2, \dots, m_p, \\ & & i &= 1, \dots, b. \end{aligned}$$

The selection of the displacements  $Q_i$  can follow a surface trajectory. Assume that the initial points obtained in the grasp planning stage are  $Q_i^1$ . Global trajectories can be

calculated if the geometry is well know. A local approximation from the first position can be calculated using Taylor's series, with

$$(28) \quad Q_i(t) = Q_i^1 + \frac{dQ_i}{dt} \Big|_1 t + \dots,$$

so that the trajectory of a point on the surface is given by

$$(29) \quad \begin{aligned} X(t) &= (Q_i^1 + \frac{dQ_i}{dt} \Big|_1 t + \dots)(Q_i^1)^* X Q_i^1 ((Q_i^1)^* + (\frac{dQ_i}{dt} \Big|_1 t)^* + \dots) \\ &= (1 + \frac{1}{2} V_1 t + \dots) X (1 + \frac{1}{2} V_1^* t + \dots), \end{aligned}$$

where  $V_1$  must be such that the point velocity is tangent to the surface, that is, in the local  $x - y$  plane. For curved surfaces, it is necessary to incorporate the second derivative in order to move along the surface. For this work we focus on convex polygonal objects.

In the surface frame, the relative twist of the finger is

$$(30) \quad V_{ri} = (v_x i + v_y j) \mathcal{E}$$

so that the overall finger twist can be linearly calculated,

$$(31) \quad \begin{aligned} V_{fi}^j = V_i^j = V_o^j + Q_{fi}^j V_{ri}^j Q_{fi}^{j*}, \quad j = 1, \dots, m_v, \\ i = 1, \dots, b. \end{aligned}$$

For a synthesis task, more than one twist can be defined at a given position, effectively defining the subspace of potential velocities of the fingertip. Given a hand with mobility  $M > 0$  and  $M' = 0$ , defining  $M$  velocities for the fingertips fully specifies the allowable twists and ensures that the fingertip's motion will be in the desired subspace for each position.

For the case of polygonal objects,

$$(32) \quad \begin{aligned} V_{ri_1} &= v_x i \mathcal{E} \\ V_{ri_2} &= v_y i \mathcal{E} \end{aligned}$$

defines the sliding on the surface of the object for a particular position. Applying these conditions, the fingertips of the synthesized hand will move on the surface of the object while performing the specified motion of the object.

## 5. CONCLUSIONS

The design of hands for specific grasping and manipulation tasks using kinematic synthesis requires a definition of fingertip displacements and velocities compatible with the object geometry, selected contact points, desired object motion and type of contact fingers. In this work a first analysis is developed in order to create kinematic tasks of fingers to create hands able to manipulate objects while keeping some grasping constraints. The initial cases presented here need to be developed to include other type of finger actions and link contacts. The use of these tasks may lead to the design of hands better tailored to specific applications.

## REFERENCES

- [1] E. Bayro-Corrochano and J. Zamora-Esquivel. Differential and inverse kinematics of robot devices using conformal geometric algebra. *Robotica*, 25:46–31, 2007.
- [2] S. Berman, D.G. Liebermann, and T. Flash. Application of motor algebra to the analysis of human arm movements. *Robotica*, 2007.
- [3] J. Borras and A.M. Dollar. A parallel robots framework to study precision grasping and dexterous manipulation. In *IEEE International Conference of Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 2013.



- [4] B. Fauser and R. Ablamowicz. On the decomposition of clifford algebras of arbitrary bilinear form. In *Proceedings of the 5th International Conference on Clifford Algebras and their Applications in Mathematical Physics*, June 27- July4, 1999, Ixtapa, Mexico, 1999.
- [5] Markus Grebenstein. Analysis of the current state of robot hands. In *Approaching Human Performance*, volume 98 of *Springer Tracts in Advanced Robotics*, pages 11–37. Springer International Publishing, 2014.
- [6] L. Han and J.C. Trinkle. Dextrous manipulation by rolling and finger gaiting. In *IEEE International Conference of Robotics and Automation (ICRA)*, 1998.
- [7] D. Hestenes. New tools for computational geometry and rejuvenation of screw theory. In E. Bayro Corrochano and G. Scheuermann, editors, *Geometric Algebra Computing in Engineering and Computer Science*. Springer, 2010.
- [8] A. Makhal and A. Perez-Gracia. Solvable multi-fingered hands for exact kinematic synthesis . In *Advances in Robot Kinematics*, Ljubljana, Slovenia, June 2014.
- [9] M.T. Mason. *Mechanics of Robotic Manipulation*. The MIT Press, 2001.
- [10] M.T. Mason, S.S. Srinivasa, A.S. Vazquez, and A. Rodriguez. Generality and simple hands. *International Journal of Robotics Research*, 2010.
- [11] L.U. Odhner and et al. A compliant, underactuated hand for robust manipulation. *The International Journal of Robotics Research*, 33(5):736–752, 2014.
- [12] N. Patarinsky Robson and J. M. McCarthy. Kinematic Synthesis With Contact Direction and Curvature Constraints on the Workpiece. volume 8, pages 581–588, Las Vegas, NV, 2007. ASME.
- [13] A. Perez Gracia and J. M. McCarthy. The kinematic synthesis of spatial serial chains using clifford algebra exponentials. *Proceedings of the Institution of Mechanical Engineers, Part C, Journal of Mechanical Engineering Science*, 220(7):953–968, 2006.
- [14] I.R. Porteous. *Clifford Algebras and the Classical Groups*. Cambridge Studies on Advanced Mathematics, 50, Cambridge University Press, 1995.
- [15] M. Quigley, C. Salisbury, A.Y. Ng, and J.K. Salisbury. Mechatronic design of an integrated robotic hand. *The International Journal of Robotics Research*, 33(5):706–720, 2014.
- [16] A. Sahbani, S. El-Khoury, and P. Bidaud. An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336, 2012.
- [17] J.K. Salisbury and J.J. Craig. Articulated hands: Force control and kinematic issues. *The International Journal of Robotic Research*, 1(1):4–17, 1982.
- [18] J. M. Selig. *Geometric Fundamentals of Robotics (Monographs in Computer Science)*. SpringerVerlag, 2005.
- [19] J.M. Selig and E. Bayro-Corrochano. Rigid body dynamics using clifford algebra. *Advances in Applied Clifford Algebras*, 20:141–154, 2010.
- [20] E. Simo-Serra and A. Perez-Gracia. Kinematic synthesis using tree topologies. *Mechanism and Machine Theory*, 72 C:94–113, 2014.
- [21] E. Simo-Serra, A. Perez-Gracia, H. Moon, and N. Robson. Design of multi fingered robotic hands for finite and infinitesimal tasks using kinematic synthesis. In *Advances in Robot Kinematics*, Innsbruck, Austria, June 2012.
- [22] E. Staffetti and F. Thomas. Analysis of rigid body interactions for compliant motion tasks using grassmann-cayley algebra. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)*, Takamatsu, Japan, November 5-9, 2000.
- [23] J. Zamora-Esquivel and E. Bayro-Corrochano. Robot perception and handling actions using the conformal geometric algebra framework. *Advances in Applied Clifford Algebras*, 20:959–990, 2010.

# SLICE-REGULAR FUNCTIONS OVER CLIFFORD ALGEBRAS AND HARMONIC FUNCTIONS

A. Perotti <sup>a</sup>

<sup>a</sup> Department of Mathematics  
University of Trento, Trento, Italy  
perotti@science.unitn.it

ABSTRACT. The Cauchy-Riemann operator on the real Clifford algebra of signature  $(0, n)$  factorizes the Laplacian operator of  $\mathbb{R}^{n+1}$ . Here we present some new relations between the Cauchy-Riemann operator and the class of *slice-regular* functions on the Clifford algebra. Slice-regular functions, which comprise all polynomials, constitute a recent function theory in several hypercomplex settings, including quaternions and Clifford algebras (cf. [6, 8, 5]). A formula, relating the differential operator characterizing slice regularity [2, 7], the *spherical derivative* of a slice regular function [8], and the Cauchy-Riemann operator, is given. The computation of the Laplacian of the spherical derivative of a slice regular function gives a result which implies, in particular, the Fueter-Sce Theorem for monogenic functions. In the two four-dimensional cases represented by the Clifford algebra  $\mathbb{R}_{0,3}$  and by the space of quaternions, these results are related with the zonal harmonics on the three-dimensional sphere and the Poisson kernel of the unit ball.

## INTRODUCTION

Let  $\mathbb{R}_{0,n}$  denote the real Clifford algebra of signature  $(0, n)$ , with basis vectors  $e_1, \dots, e_n$ . The Cauchy-Riemann operator

$$\mathcal{D} = \frac{\partial}{\partial x_0} + e_1 \frac{\partial}{\partial x_1} + \dots + e_n \frac{\partial}{\partial x_n}$$

on  $\mathbb{R}_{0,n}$  factorizes the Laplacian operator of the paravector space

$$\mathbb{R}^{n+1} = \{x_0 + x_1 e_1 + \dots + x_n e_n \mid x_0, \dots, x_n \in \mathbb{R}\}.$$

We show some new relations between the Cauchy-Riemann operator, the Laplacian operator and the class of *slice-regular* functions on a Clifford algebra. Slice-regular functions constitute a recent but rapidly expanding function theory in several hypercomplex settings, including quaternions and real Clifford algebras (cf. e.g. [6, 8, 3]). This class of functions was introduced by Gentili and Struppa in 2006-2007 [6] for functions of a quaternionic variable. Let  $\mathbb{H}$  denote the skew field of quaternions. For each imaginary unit  $J$  in the sphere

$$\mathbb{S}_{\mathbb{H}} = \{J \in \mathbb{H} \mid J^2 = -1\} = \{x_1 i + x_2 j + x_3 k \mid x_1^2 + x_2^2 + x_3^2 = 1\},$$

let  $\mathbb{C}_J = \langle 1, J \rangle \simeq \mathbb{C}$  be the subalgebra generated by  $J$ . Then we have the “slice” decomposition

$$\mathbb{H} = \bigcup_{J \in \mathbb{S}_{\mathbb{H}}} \mathbb{C}_J, \quad \text{with } \mathbb{C}_J \cap \mathbb{C}_K = \mathbb{R} \quad \text{for every } J, K \in \mathbb{S}_{\mathbb{H}}, J \neq \pm K.$$

A differentiable function  $f : \Omega \subseteq \mathbb{H} \rightarrow \mathbb{H}$  is called (*left*) *slice-regular* on  $\Omega$  if, for each  $J \in \mathbb{S}_{\mathbb{H}}$ , the restriction

$$f|_{\Omega \cap \mathbb{C}_J} : \Omega \cap \mathbb{C}_J \rightarrow \mathbb{H}$$

is holomorphic w.r.t. the complex structure defined by  $L_J(v) = Jv$ . For example, polynomials  $f(x) = \sum_m x^m a_m$  with quaternionic coefficients on the right, or more generally convergent power

---

Partially supported by grants FIRB 2012 “Differential Geometry and Geometric Function Theory”, MIUR Project “Proprietà geometriche delle varietà reali e complesse” and GNSAGA of INdAM.

series, are slice-regular on  $\mathbb{H}$ . Observe that (non-constant) polynomials are not in the kernel of the Cauchy-Fueter operator

$$\mathcal{D}_{CF} = \frac{\partial}{\partial x_0} + i \frac{\partial}{\partial x_1} + j \frac{\partial}{\partial x_2} + k \frac{\partial}{\partial x_3}.$$

Every quaternionic polynomial  $f(x) = \sum_m x^m a_m$  lifts to a unique polynomial function  $F : \mathbb{C} \rightarrow \mathbb{H} \otimes \mathbb{C}$  which makes the following diagram commutative for all  $J \in \mathbb{S}_{\mathbb{H}}$ :

$$\begin{array}{ccc} \mathbb{C} \simeq \mathbb{R} \otimes_{\mathbb{R}} \mathbb{C} & \xrightarrow{F} & \mathbb{H} \otimes_{\mathbb{R}} \mathbb{C} \\ \Phi_J \downarrow & & \downarrow \Phi_J \\ \mathbb{H} & \xrightarrow{f} & \mathbb{H} \end{array}$$

where  $\Phi_J : \mathbb{H} \otimes \mathbb{C} \rightarrow \mathbb{H}$  is defined by  $\Phi_J(a + ib) := a + Jb$ . (namely  $F(z) = \sum_m z^m a_m$ , with  $z = a + ib \in \mathbb{C}$ .) This property is equivalent to the following: for each  $z = \alpha + i\beta \in \mathbb{C}$ , the restriction of  $f$  to the 2-sphere  $\alpha + \mathbb{S}_{\mathbb{H}}\beta = \cup_{J \in \mathbb{S}_{\mathbb{H}}} \Phi_J(z)$ , is an affine function w.r.t.  $J \in \mathbb{S}_{\mathbb{H}}$ . In this lifting, the usual product of polynomials with coefficients in  $\mathbb{H}$  corresponds to the pointwise product in the algebra  $\mathbb{H} \otimes \mathbb{C}$ .

This approach to slice regularity can be pursued on a ample class of real algebras. Given a real alternative  $*$ -algebra  $A$ , with linear antiinvolution  $x \mapsto x^c$ , such that  $(xy)^c = y^c x^c$  for all  $x, y \in A$  and  $x^c = x$  for  $x$  real, let  $t(x) := x + x^c \in A$  be the *trace* of  $x$  and  $n(x) := xx^c \in A$  the *norm* of  $x$ . The *quadratic cone* of  $A$

$$\mathcal{Q}_A = \mathbb{R} \cup \{x \in A \mid t(x) \in \mathbb{R}, n(x) \in \mathbb{R}, 4n(x) > t(x)^2\}$$

admits the slice decomposition in complex lines:

$$\mathcal{Q}_A = \bigcup_{J \in \mathbb{S}_A} \mathbb{C}_J, \quad \text{with } \mathbb{C}_J \cap \mathbb{C}_K = \mathbb{R} \quad \text{for each } J, K \in \mathbb{S}_A, J \neq \pm K, \text{ where}$$

$\mathbb{S}_A = \{J \in \mathcal{Q}_A \mid J^2 = -1\} = \{J \in A \mid t(x) = 0, n(x) = 1\}$  is the “sphere” of imaginary units of  $A$ . Observe that  $\mathcal{Q}_A = A$  if and only if  $A \simeq \mathbb{R}, \mathbb{C}, \mathbb{H}, \mathbb{O}$  (the algebra of octonions).

By imposing commutativity of diagrams

$$\begin{array}{ccc} \mathbb{C} \simeq \mathbb{R} \otimes \mathbb{C} & \xrightarrow{F} & A \otimes \mathbb{C} \\ \Phi_J \downarrow & & \downarrow \Phi_J \\ \mathcal{Q}_A & \xrightarrow{f} & A \end{array}$$

for all  $J \in \mathbb{S}_A$  (with  $\Phi_J : A \otimes \mathbb{C} \rightarrow A$  defined by  $\Phi_J(a + ib) := a + Jb$ ), we get the class of *slice functions* on  $A$ . More precisely, let  $D \subseteq \mathbb{C}$ , invariant w.r.t. complex conjugation. If  $F : D \rightarrow A \otimes \mathbb{C}$  satisfies  $F(\bar{z}) = \overline{F(z)}$  for every  $z \in D$ ,  $F$  is called a *stem function* on  $D$ . Let  $\Omega_D = \cup_{J \in \mathbb{S}_A} \Phi_J(D) \subseteq \mathcal{Q}_A$ . Then the stem function  $F = F_1 + iF_2 : D \rightarrow A \otimes \mathbb{C}$  induces the (*left*) *slice function*  $f = \mathcal{I}(F) : \Omega_D \rightarrow A$  in the following way: if  $x = \alpha + J\beta = \Phi_J(z) \in \Omega_D \cap \mathbb{C}_J$ , then

$$f(x) = F_1(z) + JF_2(z).$$

The slice function  $f$  is called (*left*) *slice-regular* if  $F$  is holomorphic. The function  $f$  is called *intrinsic* if  $F_1$  and  $F_2$  are real-valued (a case already considered by Fueter [4] for quaternionic functions). In this case, the condition  $f(x^c) = f(x)^c$  holds for each  $x \in \Omega_D$ . When  $A$  is the algebra of real quaternions and the domain  $D$  intersects the real axis, this definition is equivalent to the one proposed by Gentili and Struppa.

Let  $A$  be the Clifford algebra  $\mathbb{R}_{0,n}$ , with basis elements  $e_K = e_{i_1} \cdots e_{i_k}$ , with  $e_\emptyset = 1$ ,  $K = (i_1, \dots, i_k)$  multi-index,  $0 \leq k \leq n$ . Every  $x \in \mathbb{R}_{0,n}$  can be written as  $x = \sum_K x_K e_K$ , with  $x_K \in \mathbb{R}$ . The Clifford conjugation  $x \mapsto x^c$  is the unique antiinvolution of  $\mathbb{R}_{0,n}$  such that  $e_i^c = -e_i$  for

$i = 1, \dots, n$ . If  $x = x_0 + x_1 e_1 + \dots + x_n e_n \in \mathbb{R}^{n+1}$ , then  $x^c = x_0 - x_1 e_1 - \dots - x_n e_n$ . Therefore  $t(x) = x + x^c = 2x_0$  and  $n(x) = xx^c = |x|^2$ . The same relations hold on the entire quadratic cone

$$\mathcal{Q}_{\mathbb{R}_{0,n}} = \{x \in \mathbb{R}_{0,n} \mid t(x) \in \mathbb{R}, n(x) \in \mathbb{R}\} \supseteq \mathbb{R}^{n+1}.$$

For example,  $\mathcal{Q}_{\mathbb{R}_{0,1}} = \mathbb{R}_{0,1} \simeq \mathbb{C}$ ,  $\mathcal{Q}_{\mathbb{R}_{0,2}} = \mathbb{R}_{0,2} \simeq \mathbb{H}$ , while

$$\mathcal{Q}_{\mathbb{R}_{0,3}} = \{x \in \mathbb{R}_{0,3} \mid x_{123} = x_1 x_{23} - x_2 x_{13} + x_3 x_{12} = 0\} \supset \mathbb{R}^4$$

is a real algebraic set of dimension 6.

Each  $x \in \mathcal{Q}_{\mathbb{R}_{0,n}}$  can be written as  $x = \operatorname{Re}(x) + \operatorname{Im}(x)$ , with  $\operatorname{Re}(x) = \frac{x+x^c}{2}$ ,  $\operatorname{Im}(x) = \frac{x-x^c}{2} = \beta J$ , with  $\beta = |\operatorname{Im}(x)|$  and  $J \in \mathbb{S}_{0,n} := \mathbb{S}_{\mathbb{R}_{0,n}}$  (the ‘‘sphere’’ of imaginary units in  $\mathbb{R}_{0,n}$ ).

In the next sections we will introduce a differential operator  $\bar{\vartheta}$  characterizing slice regularity [2, 7] and the notion of *spherical derivative* of a slice function [8]. Then we will prove a formula, relating the operator  $\bar{\vartheta}$ , the spherical derivative and the Cauchy-Riemann operator on  $\mathbb{R}_{0,n}$ . The computation of the Laplacian of the spherical derivative of a slice regular function gives a result which implies, in particular, the Fueter-Sce Theorem for monogenic functions (i.e. belonging to the kernel of the Cauchy-Riemann operator  $\mathcal{D}$ ). We recall that the Fueter’s Theorem [4], generalized by Sce [10], Qian [9] and Sommen [11] on Clifford algebras and octonions, in our language states that applying to an intrinsic slice regular function the Laplacian operator of  $\mathbb{R}^4$  (in the quaternionic case) or the iterated Laplacian operator  $\Delta^{(n-1)/2}$  of  $\mathbb{R}^{n+1}$  (in the Clifford algebra case with  $n$  odd), one obtains a function in the kernel, respectively, of the Cauchy-Fueter operator  $\mathcal{D}_{CF}$  or of the Cauchy-Riemann operator  $\mathcal{D}$ .

These results take a particularly neat form in the two four-dimensional cases represented by the Clifford algebra  $\mathbb{R}_{0,3}$  and by the space of quaternions. As we will show in Sections 2 and 3, in these cases there appear unexpected relations between slice regular functions, the zonal harmonics on the three-dimensional sphere and the Poisson kernel of the unit ball.

## 1. THE OPERATOR $\bar{\vartheta}$

For each alternative  $*$ -algebra  $A$  there exists [7] a differential operator

$$\bar{\vartheta} : \mathcal{C}^1(\Omega \setminus \mathbb{R}, A) \longrightarrow \mathcal{C}^0(\Omega \setminus \mathbb{R}, A)$$

which characterizes slice-regular functions on  $\Omega \subseteq \mathcal{Q}_A$  among the class of slice functions. In particular, when  $A$  is the Clifford algebra  $\mathbb{R}_{0,n}$ , the operator  $\bar{\vartheta}$  has the following expression

$$\bar{\vartheta} = \frac{\partial}{\partial x_0} + \frac{\operatorname{Im}(x)}{n(\operatorname{Im}(x))} \sum_{|K|=1,2 \pmod{4}} x_K \frac{\partial}{\partial x_K}$$

(cfr. [2] for an equivalent operator defined on the paravector space  $\mathbb{R}^{n+1} \subseteq \mathcal{Q}_{\mathbb{R}_{0,n}}$ ).

**Theorem 1** ([7]). *If  $f \in \mathcal{C}^1(\Omega)$  is a slice function, then  $f$  is slice-regular if and only if  $\bar{\vartheta} f = 0$  on  $\Omega \setminus \mathbb{R}$ . If  $\Omega \cap \mathbb{R} \neq \emptyset$  and  $f \in \mathcal{C}^1(\Omega)$ , then  $f$  is slice-regular if and only if  $\bar{\vartheta} f = 0$ .*

## 2. THE LAPLACIAN OF THE SPHERICAL DERIVATIVE

Let  $f = \mathcal{I}(F)$  be a slice function, with  $F = F_1 + iF_2$  and  $F_1, F_2 : D \subseteq \mathbb{C} \longrightarrow \mathbb{R}_{0,n}$ .

**Definition 1.** *The function  $f_s^\circ : \Omega \longrightarrow \mathbb{R}_{0,n}$ , called spherical value of  $f$ , and the function  $f_s' : \Omega \setminus \mathbb{R} \longrightarrow \mathbb{R}_{0,n}$ , called spherical derivative of  $f$ , are defined as*

$$f_s^\circ(x) := \frac{1}{2}(f(x) + f(x^c)) \quad \text{and} \quad f_s'(x) := \frac{1}{2}\operatorname{Im}(x)^{-1}(f(x) - f(x^c)).$$

If  $x = \alpha + \beta J$ ,  $z = \alpha + \beta i$ , then  $f_s^\circ(x) = F_1(z)$  and  $f_s'(x) = \beta^{-1}F_2(z)$ . Therefore  $f_s^\circ$  and  $f_s'$  are slice functions, constant on every sphere  $\mathbb{S}_x = \alpha + \mathbb{S}_{0,n}\beta$ . Moreover,

$$f(x) = f_s^\circ(x) + \text{Im}(x)f_s'(x)$$

for each  $x \in \Omega \setminus \mathbb{R}$  (this holds also on  $\Omega \cap \mathbb{R}$  if  $F \in \mathcal{C}^1$ ).

Since the paravector space  $\mathbb{R}^{n+1}$  is contained in the quadratic cone  $\mathcal{Q}_{\mathbb{R}_{0,n}}$ , we can consider the restriction of a slice or a slice-regular function on an open domain in  $\mathbb{R}^{n+1}$ . These restrictions uniquely determine the slice function.

**Theorem 2.** *For each slice function  $f : \Omega \subseteq \mathbb{R}^{n+1} \longrightarrow \mathbb{R}_{0,n}$ , of class  $\mathcal{C}^1(\Omega)$ , the following formula holds:*

$$\mathcal{D}f - \bar{\mathcal{D}}f = (1 - n)f_s'.$$

**Corollary 3.** *Let  $f : \Omega \subseteq \mathbb{R}^{n+1} \longrightarrow \mathbb{R}_{0,n}$  be a slice function, of class  $\mathcal{C}^1(\Omega)$ . Then*

- a)  $f$  is slice-regular if and only if  $\mathcal{D}f = (1 - n)f_s'$ .
- b) Let  $n > 1$ . Then  $f$  is slice-regular and belongs to the kernel of  $\mathcal{D}$  if and only if  $f$  is (locally) constant.

Let  $f = \mathcal{I}(F)$ , with  $F = F_1 + iF_2$  a stem function with real analytic components  $F_1, F_2$ . Since  $F(\bar{z}) = \overline{F(z)}$  for every  $z$ , the functions  $F_1, F_2 : D \subseteq \mathbb{C} \longrightarrow \mathbb{R}_{0,n}$  are, respectively, even and odd functions w.r.t. the variable  $\beta$ . Therefore there exist  $G_1$  and  $G_2$  (again real analytic) such that

$$F_1(\alpha, \beta) = G_1(\alpha, \beta^2), \quad F_2(\alpha, \beta) = \beta G_2(\alpha, \beta^2).$$

If  $x = \alpha + \beta J$ ,  $z = \alpha + \beta i$ , then

$$f_s^\circ(x) = G_1(\alpha, \beta^2) = G_1(\text{Re}(x), n(\text{Im}(x))),$$

$$f_s'(x) = G_2(\alpha, \beta^2) = G_2(\text{Re}(x), n(\text{Im}(x))).$$

The functions  $G_1$  and  $G_2$  are useful in the computation the Laplacian of the spherical derivative and of the spherical value of a slice regular function.

**Theorem 4.** *Let  $f = \mathcal{I}(F) : \Omega \subseteq \mathbb{R}^{n+1} \longrightarrow \mathbb{R}_{0,n}$  be (the restriction of) a slice-regular function. Let  $\partial_2 G_2(u, v)$  stand for the partial derivative  $\frac{\partial G_2}{\partial v}(u, v)$  and let  $\Delta_{n+1}$  be the Laplacian operator on  $\mathbb{R}^{n+1}$ . Then it holds:*

a)

$$\Delta_{n+1}f_s'(x) = 2(n-3)\partial_2 G_2(\text{Re}(x), n(\text{Im}(x))).$$

b) For each  $k = 0, 1, \dots, \lfloor \frac{n-1}{2} \rfloor$ ,

$$\Delta_{n+1}^k f_s'(x) = 2^k(n-3)(n-5)\cdots(n-2k-1)\partial_2^k G_2(\text{Re}(x), n(\text{Im}(x))).$$

c)

$$\Delta_{n+1}f_s'(x) = \frac{n-3}{n(\text{Im}(x))} \left( \frac{\partial f_s^\circ}{\partial x_0}(x) - f_s'(x) \right).$$

To obtain a) and b) is sufficient that  $F_2$  has harmonic components on  $D$ .

**Four dimensional case ( $n = 3$ ).**

**Corollary 5.** *Let  $f : \Omega \subseteq \mathbb{R}^4 \longrightarrow \mathbb{R}_{0,3}$  be slice-regular. Then*

- a)  $f'_s$  is harmonic on  $\Omega$ , i.e. its eight components are harmonic real functions.
- b) The following generalization of Fueter-Sce Theorem for  $\mathbb{R}_{0,3}$  holds:

$$\mathcal{D}\Delta_4 f = \Delta_4 \mathcal{D}f = -2\Delta_4 f'_s = 0.$$

- c)  $\Delta_4^2 f = 0$ , i.e. every slice regular function on  $\mathbb{R}_{0,3}$  is biharmonic.

To have point a) it is sufficient that  $F_2$  has harmonic real components on  $D$ , for example that  $f$  be slice-harmonic, i.e. induced by an harmonic stem function on  $D \subseteq \mathbb{C}$ .

**Higher dimensional case ( $n > 3$  odd).**

**Corollary 6.** *Let  $n > 3$  odd. If  $f : \Omega \subseteq \mathbb{R}^{n+1} \longrightarrow \mathbb{R}_n$  is slice-regular, then*

- a)  $(\Delta_{n+1})^{\frac{n-3}{2}} f'_s$  is harmonic on  $\Omega$ .
- b) The following generalization of Fueter-Sce Theorem for  $\mathbb{R}_{0,n}$  holds:

$$\mathcal{D}(\Delta_{n+1})^{\frac{n-1}{2}} f = (\Delta_{n+1})^{\frac{n-1}{2}} \mathcal{D}f = (1-n)(\Delta_{n+1})^{\frac{n-1}{2}} f'_s = 0.$$

- c)  $(\Delta_{n+1})^{\frac{n+1}{2}} f = 0$ , i.e. every slice regular function on  $\mathbb{R}_{0,n}$  is polyharmonic.

As regards the spherical value of a slice-regular function, we can still compute its Laplacian. In general, even in the four dimensional case, it is not a harmonic function, but it is biharmonic.

**Theorem 7.** *Let  $f = \mathcal{I}(F) : \Omega \subseteq \mathbb{R}^{n+1} \longrightarrow \mathbb{R}_{0,n}$  be a slice-regular function. It holds:*

- a)

$$\Delta_{n+1} f_s^\circ(x) = (\Delta_{n+1} f)_s^\circ(x) = 2(n-1) \partial_2 G_1(\operatorname{Re}(x), n(\operatorname{Im}(x))).$$

- b) For each  $k = 0, 1, \dots, \lfloor \frac{n-1}{2} \rfloor$

$$\Delta_{n+1}^k f_s^\circ(x) = 2^k (n-1)(n-3) \cdots (n-2k+1) \partial_2^k G_1(\operatorname{Re}(x), n(\operatorname{Im}(x))).$$

- c)

$$\Delta_{n+1} f'_s(x) = (1-n) \frac{\partial f'_s}{\partial x_0}(x).$$

- d) When  $n = 3$ ,  $\Delta_4^2 f_s^\circ = 0$ . In general,  $(\Delta_{n+1})^{\frac{n+1}{2}} f_s^\circ = 0$ .

Again, to obtain a) and b) it is sufficient that  $F_1$  has harmonic real components on  $D$ .

**Four dimensional case: zonal harmonics and the Poisson kernel.** Thanks to Corollary 5, for any polynomial  $f = \sum_{m=0}^d x^m a_m$  with coefficients in  $\mathbb{R}_{0,3}$ , the spherical derivative of  $f$  is a harmonic polynomial on  $\mathbb{R}^4$ :  $f'_s = (\sum_{m=0}^d x^m a_m)'_s = \sum_{m=0}^d (x^m)'_s a_m$ . In particular, the spherical derivative of a Clifford power  $x^m$  is homogeneous harmonic polynomial of degree  $m-1$ , with real coefficients, in the variables  $x_0, x_1, x_2, x_3$ . Moreover, for each slice-regular  $f$ , the spherical derivative  $f'_s$  is constant on the 2-spheres  $\mathbb{S}_x = \alpha + \mathbb{S}_{0,3}\beta$ .

Let  $\mathbb{B}$  be the unit ball in  $\mathbb{R}^4$ . Let  $\mathcal{Z}_m(x, a)$  denote the four-dimensional (solid) zonal harmonic of degree  $m$  with pole  $a \in \partial\mathbb{B}$  (see e.g. [1]). From the uniqueness properties of zonal harmonics, and its invariance w.r.t. four-dimensional rotations, we then get the following result.

**Proposition 8.** *Let  $\mathbb{B}$  be the unit ball in  $\mathbb{R}^4$ . For every  $m \geq 1$  and every  $a \in \partial\mathbb{B}$ , it holds:*

- a)  $\mathcal{Z}_{m-1}(x, 1) = m(x^m)'_s$ .
- b)  $\mathcal{Z}_{m-1}(x, a) = \mathcal{Z}_{m-1}(xa^{-1}, 1) = \mathcal{Z}_{m-1}(xa^c, 1) = m(x^m)'_s(xa^c)$ .
- c)  $(x^{-m})'_s = -\mathcal{K}[(x^m)'_s]$ , where  $\mathcal{K}$  is the Kelvin transform.

Let  $\mathcal{P}(x, a) = \frac{1-|x|^2}{|x-a|^4}$  be the Poisson kernel for the unit ball  $\mathbb{B}$  in  $\mathbb{R}^4$  ( $x \in \mathbb{B}$ ,  $a \in \partial\mathbb{B}$ ). From the series expansion (see [1])  $\mathcal{P}(x, a) = \sum_{m=0}^{\infty} \mathcal{L}_m(x, a)$ , we get:

**Proposition 9.** *The Clifford Koebe function  $g(x) = (1-x)^{-2}x$  is slice-regular on  $\mathbb{R}^4 \setminus \{1\}$  and has the following properties: for every  $x \in \mathbb{B}$ ,*

$$g'_s(x) = \mathcal{P}(x, 1) = \frac{1-|x|^2}{|x-1|^4}.$$

For every  $a \in \partial\mathbb{B}$  and  $x \in \mathbb{B}$ ,

$$g'_s(xa^c) = \mathcal{P}(x, a) = \frac{1-|x|^2}{|x-a|^4}.$$

### 3. THE QUATERNIONIC CASE

When  $n = 2$ , the Clifford algebra  $\mathbb{R}_{0,2}$  is isomorphic to the field  $\mathbb{H}$  of quaternions. In this case the paravector space has dimension three and then the corollaries of Section 2 are not applicable. However, similar results still hold since the computations made in Theorem 2 on the paravector space can be repeated anytime there is a real subspace of the quadratic cone containing the real axis. The simplest example of this setting is given by the quaternions, where the quadratic cone coincides with the whole algebra:  $\mathcal{Q}_{\mathbb{H}} = \mathbb{H}$ .

By means of the identifications  $e_1 = i$ ,  $e_2 = j$ ,  $e_{12} = ij = k$ , in coordinates  $(x_0, x_1, x_2, x_3)$  of  $x = x_0 + x_1i + x_2j + x_3k \in \mathbb{H}$ , the differential operator  $\bar{\vartheta}$  takes the form [7]

$$\bar{\vartheta} = \frac{\partial}{\partial x_0} + \frac{\text{Im}(x)}{n(\text{Im}(x))} \sum_{i=1}^3 x_i \frac{\partial}{\partial x_i}.$$

For every slice function  $f : \Omega \subseteq \mathbb{R}^3 \rightarrow \mathbb{R}_{0,2}$ , of class  $\mathcal{C}^1$  on a domain  $\Omega$  in the three-dimensional space of (quaternionic) paravectors, Theorem 2 gives

$$\mathcal{D}f - \bar{\vartheta}f = -f'_s.$$

If we consider the whole quaternion algebra, we must instead use the Cauchy-Fueter operator

$$\mathcal{D}_{CF} = \frac{\partial}{\partial x_0} + i \frac{\partial}{\partial x_1} + j \frac{\partial}{\partial x_2} + k \frac{\partial}{\partial x_3} = \mathcal{D} + k \frac{\partial}{\partial x_3}.$$

Let  $\vartheta$  and  $\bar{\mathcal{D}}_{CF}$  be the conjugated differential operators:

$$\vartheta = \frac{\partial}{\partial x_0} - \frac{\text{Im}(x)}{n(\text{Im}(x))} \sum_{i=1}^3 x_i \frac{\partial}{\partial x_i}$$

and

$$\bar{\mathcal{D}}_{CF} = \frac{\partial}{\partial x_0} - i \frac{\partial}{\partial x_1} - j \frac{\partial}{\partial x_2} - k \frac{\partial}{\partial x_3}.$$

For the Cauchy-Fueter operator the analogous of Theorem 2 is the following:

**Theorem 10.** *For every slice function  $f : \Omega \subseteq \mathbb{H} \rightarrow \mathbb{H}$ , of class  $\mathcal{C}^1(\Omega)$ , the following formulas hold:*

$$\mathcal{D}_{CF}f - \bar{\vartheta}f = -2f'_s \quad \text{and} \quad \bar{\mathcal{D}}_{CF}f - \vartheta f = 2f'_s.$$

**Corollary 11.** *Let  $f : \Omega \subseteq \mathbb{H} \rightarrow \mathbb{H}$  be a slice function of class  $\mathcal{C}^1(\Omega)$ . Then*

- $f$  is slice-regular if and only if  $\mathcal{D}_{CF}f = -2f'_s$ .
- $f$  is slice-regular and belongs to the kernel of  $\mathcal{D}_{CF}$  if and only if  $f$  is (locally) constant.
- $\vartheta f'_s = \bar{\mathcal{D}}_{CF}f'_s$ .

The last statement follows from the fact that  $(f'_s)'_s = 0$  for every slice function  $f$ .

**Theorem 12.** *If  $f : \Omega \subseteq \mathbb{H} \rightarrow \mathbb{H}$  is slice-regular, then*

- a) *The spherical derivative  $f'_s$  is harmonic on  $\Omega$  (i.e. its four components are harmonic real functions).*
- b) *The following generalization of Fueter's Theorem holds:*

$$\mathcal{D}_{CF}\Delta_4 f = \Delta_4 \mathcal{D}_{CF} f = -2\Delta_4 f'_s = 0$$

*As a consequence,  $\Delta_4^2 f = 0$ : every quaternionic slice-regular function is biharmonic.*

For each slice function  $f = \mathcal{S}(F)$ , its slice derivative  $\frac{\partial f}{\partial x}$  can be defined as the slice function induced by the stem function  $\frac{\partial F}{\partial z}$ . It turns out that  $\frac{\partial f}{\partial x} = \vartheta f$ . From Theorem 12 it follows that if  $f$  is slice-regular, then  $\mathcal{D}_{CF}\left(\frac{\partial f'_s}{\partial x}\right) = \mathcal{D}_{CF}(\vartheta f'_s) = 0$  and therefore also  $\frac{\partial f'_s}{\partial x}$  is harmonic.

**Corollary 13.** *Let  $\mathbb{B}$  be the unit ball in  $\mathbb{R}^4$ . For every  $m \geq 1$ , it holds:*

- a)  *$m(x^m)'_s = \mathcal{L}_{m-1}(x, 1)$ . Therefore  $\mathcal{D}_{CF}(x^m) = -\frac{2}{m}\mathcal{L}_{m-1}(x, 1)$ .*
- b) *The restriction of  $(x^m)'_s$  to the unit sphere  $\partial\mathbb{B}$  coincides with the Gegenbauer polynomial  $C_{m-1}^{(1)}(x_0)$ .*
- c) *The quaternionic Koebe function  $g(x) = (1-x)^{-2}x$  is slice-regular on  $\mathbb{H} \setminus \{1\}$  and it holds*

$$g'_s(x) = \mathcal{P}(x, 1) = \frac{1 - |x|^2}{|x - 1|^4}$$

*for every  $x \in \mathbb{B}$ .*

#### REFERENCES

- [1] S. Axler, P. Bourdon, and W. Ramey, *Harmonic function theory*, Graduate Texts in Mathematics, Vol. 137, Springer-Verlag, New York, 1992.
- [2] F. Colombo, J. O. González-Cervantes, and I. Sabadini, *A nonconstant coefficients differential operator associated to slice monogenic functions*, Trans. Amer. Math. Soc., **365**(1) (2013), 303–318.
- [3] F. Colombo, I. Sabadini, and D.C. Struppa, *Slice monogenic functions*, Israel J. Math. **171** (2009), 385–403.
- [4] R. Fueter, *Die Funktionentheorie der Differentialgleichungen  $\Delta u = 0$  und  $\Delta\Delta u = 0$  mit vier reellen Variablen*, Comment. Math. Helv. **7**, No. 1 (1934), 307–330.
- [5] G. Gentili, C. Stoppato, D. C. Struppa, *Regular Functions of a Quaternionic Variable*, Springer Monographs in Mathematics, Springer, 2013.
- [6] G. Gentili, D. Struppa, *A new theory of regular functions of a quaternionic variable*, Adv. Math. **216** (2007), 279–301.
- [7] R. Ghiloni, A. Perotti, *Global differential equations for slice regular functions*, Math. Nach., **287**, No. 5-6 (2014), 561–573.
- [8] R. Ghiloni, A. Perotti, *Slice regular functions on real alternative algebras*, Adv. Math. **226** (2011), 1662–1691.
- [9] T. Qian, *Generalization of Fueter's result to  $\mathbf{R}^{n+1}$* , Atti Accad. Naz. Lincei Cl. Sci. Fis. Mat. Natur. Rend. Lincei (9) Mat. Appl. **8**, No. 2 (1997), 111–117.
- [10] M. Sce, *Osservazioni sulle serie di potenze nei moduli quadratici*, Atti Accad. Naz. Lincei. Rend. Cl. Sci. Fis. Mat. Nat. (8) **23** (1957), 220–225.
- [11] F. Sommen, *On a generalization of Fueter's theorem*, Z. Anal. Anwendungen **19**, No. 4 (2000), 899–902.





# CLIFFORD ALGEBRA SUPPORT IN MAXIMA

Dimiter Prodanov<sup>a</sup>

<sup>a</sup> Department of Environment, Health and Safety  
Neuroscience Research Flanders  
IMEC vzw, Leuven, Belgium

dimiterpp@gmail.com; dimiter.prodanov@imec.be [presenter, corresponding]

**ABSTRACT.** *Maxima* is the open source descendant of the first ever computer algebra system. It is written entirely in Lisp and is distributed under GNU General Public License. At present there are two packages which support Clifford algebras. *Maxima* has its own programming language, which is particularly well suited for handling formal mathematical expressions. The scripts can be compiled also to Lisp within the program itself. The package *atensor* partially implements generalized (tensor) algebras, including Clifford, Grassmann, and Lie-algebras. It is distributed as a shared package with the main program distribution. The package *clifford*, authored by the presenter, implements specifically Clifford algebras. *Clifford* relies heavily on the rule-based simplification system of Maxima for simplification of Clifford products, outer products, scalar products and inverses.

## 1. BRIEF INTRODUCTION TO MAXIMA

*Maxima* is the descendant of the first ever computer algebra system, MACSYMA. Maxima is derived from the *Macysma* system, developed at MIT in the years 1968 – 1982 as part of Project MAC. *Maxima* is written entirely in Lisp and is distributed under GNU General Public License. At present the system is supported by a team of volunteer developers. *Maxima* has its own programming language, which is particularly well suited for handling formal mathematical expressions. The scripts can be compiled also to Lisp within the program itself. Lisp programs can be also loaded and addressed within the system. The system also offers the possibility of running batch tests and demonstrations.

**1.1. Data types.** Maxima supports the following primitive data types[2]: *numbers* (rational, float and arbitrary precision); *strings* and *symbols*. In addition there are compound data types, such as *lists*; *arrays*; *matrices* and *structs*. Mathematical and physical constants are represented as arbitrary precision numbers. There are also undefined special symbolic constants, such as the boolean symbols *true* and *false* and the complex imaginary unit *i* and undefined special symbolic constants, such as different infinities and infinitesimals.

**1.2. Operators.** There several types of operators supported by Maxima. An operator is a symbol that may be either of *unary prefix* or *unary postfix*, *binary infix*, *n-ary infix*, *matchfix*, or *nofix*. "Matchfix" means a pair of symbols which enclose their argument or arguments, and "nofix" means an operator which takes no arguments. It is possible to define new operators with specified precedence or to redefine the precedence of existing operators (see example in Fig. 1). Maxima distinguishes between operators which are *nouns* and operators which are *verbs*. A verb form of an operator evaluates its arguments and produces an output result. A noun form of an operator appears as a symbol in an expression, without being executed. By default, function names are verbs. A verb can be changed into a noun and vice-versa.

---

*Date:* June 12, 2015.

The author is partially supported by a grant from Research Fund – Flanders (FWO), contract number 0880.212.840.

FIGURE 1. Infix operator definitions in Maxima

```

1  (
    "innter product",
    infix ("~", 130, 129),
    "~"(a, b) := if scalarp(a) and scalarp(b) then a*b else expand((a.b + b.
        a)/2),
    texput ("~", "\\circ ", infix),
6
    "outer product",
    infix("&", 130, 129),
    "&(a, b) := if scalarp(a) and scalarp(b) then 0 else expand((a.b - b.a)
        /2),
    texput("&", "\\wedge ", infix)
11 );

```

**1.3. Expression processing in Maxima.** An *expression* contains a sequence of operators, numbers and symbols. The value of an expression is the value of the last assigned member. In such way every expression in Maxima is a lambda construct.

Different transformation rules can be associated with any given operators in Maxima. In addition it is possible to assign also transformation rules to general symbols. Maxima has an advanced pattern matching mechanism, which supports such operations. Rules can be add to the built-in simplifier in two ways: by using the commands *tellsimp* or *tellsimpafter*. *tellsimp* rules are applied when appropriate, based on the main operator, before the built-in simplification while *tellsimpafter*, are applied after the built in simplification (Fig. 2). The augmented simplification is then treated as built in, so subsequent *tellsimp* rules are applied before the previous ones.

The simplifier subroutine operates by descending the tree of an expression until it gets to atoms, and then simplifies the smallest pieces and backs out.. e.g.

$$\text{simp} : f(a,b) \rightarrow f(\text{simp}(a), \text{simp}(b)) \rightarrow \text{simp}(f).$$

## 2. BASIC PROPERTIES OF CLIFFORD ALGEBRAS IN VIEW OF IMPLEMENTATION STRATEGIES

In this section are given only the essential properties of Clifford algebras over the reals. These properties are directly implemented in available Maxima packages. The exposition is given after [1].

A Clifford algebra is an associative algebra which is generated by a vector space  $V$  over a field  $K$  of characteristic different from 2. The algebra contains a copy of the vector space  $V$  and is equipped with a quadratic form  $Q(v)$ . The algebra is equipped with a *scalar unit* denoted conventionally by  $1$  and a (non-commutative) product referred to as Clifford product. The square in the Clifford algebra  $Cl(V, Q)$  is defined by

$$vv := Q(v)1, \quad \forall v \in V$$

The unit is usually skipped from notation and the square of the vector is denoted conveniently by  $v^2$ . The notation  $Cl_{p,q,r}(K)$  is interpreted as the convention that in the so-specified algebra  $p$  imaginary unit elements square to 1,  $q$  to -1 and  $r$  to 0. For a given Clifford algebra  $Cl_{p,q,r}(R)$  over the real numbers the form is given by

$$Q(\mathbf{v}) := v_1^2 + \dots + v_p^2 - v_{p+1}^2 - \dots - v_{p+q}^2$$

For any two basis vectors of the algebra  $Span\{e_1 \dots e_n\}$  the following anticommuting relation holds

$$(1) \quad e_i e_j = -e_j e_i, \quad i \neq j$$

Every vector for which  $Q(\mathbf{v}) \neq 0$  has an inverse vector

$$\mathbf{v}^{-1} = \frac{\mathbf{v}}{Q(\mathbf{v})}$$

The inner (dot) and outer (wedge) products of two vectors  $a$  and  $b$  are defined by

$$(2) \quad a \cdot b := \frac{ab + ba}{2}$$

$$(3) \quad a \wedge b := \frac{ab - ba}{2}$$

The general multivector (blade)  $A$  is decomposed by the grade projection operator  $\langle \cdot \rangle$  :

$$(4) \quad A = \sum_{g=0}^{2^n} \langle A \rangle_g$$

where by convention  $\langle A \cdot \rangle_0$  is the scalar part,  $\langle A \cdot \rangle_1$  is the vector part. The inner and outer products are extended to blades in the following way:

$$(5) \quad A_r \cdot B_s := \langle A_r B_s \rangle_{|r-s|}$$

$$(6) \quad A_r \wedge B_s := \langle A_r B_s \rangle_{r+s}$$

FIGURE 2. Expression simplification in Maxima

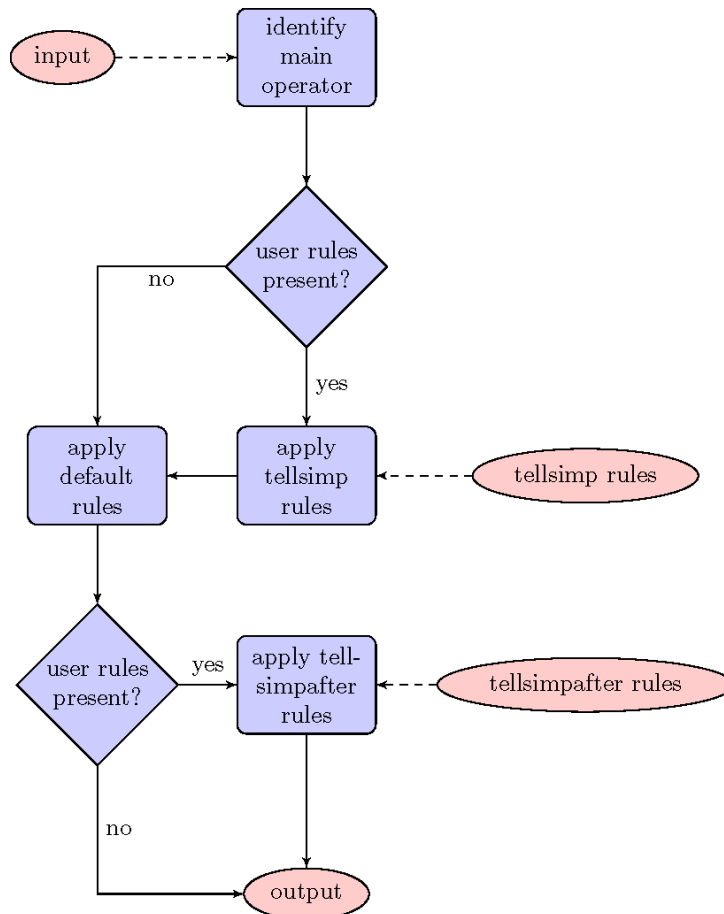


FIGURE 3. Simplification rules in *clifford*

```

/*
simplification rules
*/
4  matchdeclare(dd, lambda([u], freeof(asymbol,u)));
   defrule (clifsimp1, dd*aa, dd*dotsimp(aa));
   defrule (clifsimp10, aa, dotsimp(aa));
   defrule (clifsimp2, dd/aa, dd*dotinvsimp(1/aa));

9  /*
   full simplification of expressions
   */
   declare (cliffsimpall, evfun);
   cliffsimpall(expr) := block([res],
14  res:expand(expr),
   res:apply1(res, clifsimp2, clifsimp1, clifsimp10),
   ratsimp(res)
   );

```

### 3. IMPLEMENTATIONS OF CLIFFORD ALGEBRAS

At present there are two packages which support Clifford algebras. Both packages set simplification rules for the non-commutative dot-product operator (“.”).

**3.1. The package *atensor*.** The package *atensor* authored by Viktor Toth partially implements generalized (tensor) algebras, including Clifford, Grassmann, and Lie-algebras [3]. At present it is distributed as a shared package with the main program distribution [2].

Supported algebras can be one of the following:

**Universal:** The universal algebra has no commutation rules.

**Grassmann:** The product is defined by the commutation relation

$$u \cdot v + v \cdot u = 0$$

**Clifford:** The product is defined by the commutation relation

$$u \cdot v + v \cdot u = -2 sf(u, v)$$

where  $sf(u, v)$  is a symmetric scalar-valued function.

**Symmetric:** The product is defined by the commutation relation

$$u \cdot v - v \cdot u = 0$$

Therefore, it is commutative.

**Symplectic:** The product is defined by the commutation relation

$$u \cdot v + v \cdot u = 2 af(u, v)$$

where  $af(u, v)$  is an antisymmetric scalar-valued function.

**Lie envelop:** : The product in the algebra of the Lie envelope is defined by the commutation relation

$$u \cdot v - v \cdot u = 2 av(u, v)$$

where  $av(u, v)$  is an antisymmetric function.

The *atensor* package uses the matrix *aform* where the function values for base vector arguments are stored. When the algebra type is selected by the user, this matrix is preinitialized. For this purpose, when a Clifford, Symplectic, or Lie enveloping algebra is selected, the user can

FIGURE 4. Demonstration session of *atensor*

Definition of Lie-bracket

$$lbr(u, v) := u.v - v.u$$

$$lbr(u, v) := u.v - v.u$$

$$A : atensimp(lbr(u, lbr(v, w)))$$

$$2(u.av(v, w)) - 2(u.av(v, w) - 2av(u, av(v, w)))$$

$$B : atensimp(lbr(v, lbr(w, u)))$$

$$2(av(u, w).v) - 2(av(u, w).v - 2av(av(u, w), v))$$

$$C : atensimp(lbr(w, lbr(u, v)))$$

$$2(av(u, w) \cdot v - 2av(av(u, w), v)) - 2(av(u, w) \cdot v) + 2(u \cdot av(v, w) - 2av(u, av(v, w))) - 2(u \cdot av(v, w))$$

$$A + B + C$$

$$0$$

optionally enter values that determine the algebras dimensionality. For a Clifford algebra, up to three values can be used, specifying the positive, degenerate, and negative dimensions of the algebra, respectively. For a symplectic algebra, the numbers of regular and degenerate dimensions are specified. For Lie enveloping algebras the user is asked for a single number specifying the algebra's dimensionality. The *atensor* package comes also with built-in support for complex, quaternion, Pauli and Dirac algebras.

**3.2. The package Clifford.** The package *clifford*, designed by the present author, implements only on Clifford algebras. In contrast to *atensor*, *clifford* relies extensively on the Maxima simplification routines and is fully integrated into the command prompt evaluation. The package defines several geometric product simplification rules which are integrated in the built-in Maxima simplifier (see Fig. 3). The code is distributed under GNU Lesser General Public License from GitHub <http://dprodanov.github.io/clifford/>. *Clifford* defines multiple rules for pre- and post-simplification of Clifford products, outer products, scalar products, inverses and powers of Clifford vectors. A demonstration is presented in Fig. 5.

FIGURE 5. Simplification in *clifford*

Simplification of inverses:

$$1/(1 + e[1]), cliffsimpall;$$

$$\frac{-1 + e_1}{2}$$

Outer product calculation:

$$e[1] \& e[2];$$

$$e_1 \cdot e_2$$

$$(1 + e[1]) \& (1 + e[1]), expand;$$

$$0$$

Inner product calculation:

$$(1 + e[1]) \sim (1 + e[1]), expand;$$

$$2e_1$$

FIGURE 6. Quaternion algebra with *clifford*

Computation of Quaternion multiplication table:

*mtable2()*;

$$\begin{pmatrix} 1 & e_1 & e_2 & e_1.e_2 \\ e_1 & -1 & e_1.e_2 & -e_2 \\ e_2 & -e_1.e_2 & -1 & e_1 \\ e_1.e_2 & e_2 & -e_1 & -1 \end{pmatrix}$$

Computation of Quaternion inverses :

```
block(declare([a,b,c,d], scalar),
cc:a+b*e[1]+c*e[2]+d*e[1] . e[2],
dd:cinv(cc)
);
```

$$\frac{a - e_1 b - e_2 c - (e_1.e_2) d}{a^2 + b^2 + c^2 + d^2}$$

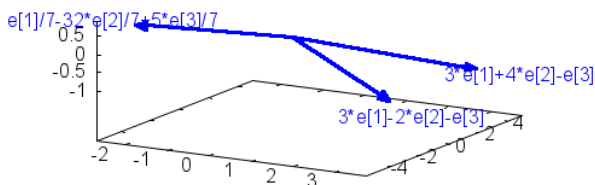
Checking the obtained result:

```
ev(dd.cc, expand, ratsimp)
1
```

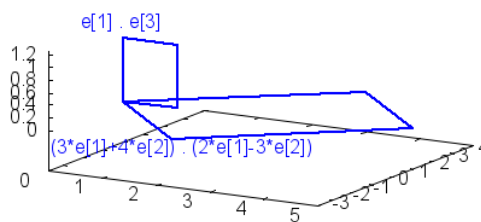
Reflection of the vector  $a = 3e_1 + 4e_2 - e_3$  by  $b = 3e_1 - 2e_2 - e_3$  is demonstrated in Fig. 7A. Bi-vectors spanned by the vectors  $Span\{e_1, e_2\}$  and  $Span\{3e_1 + 4e_2, 2e_1 - 3e_2\}$  are demonstrated in Fig. 7B.

FIGURE 7. Vector and bi-vector display in *clifford*

A



B



A – plotting of vectors; B – plotting of bi-vectors.

## REFERENCES

- [1] L. Dorst. *Geometric Computing with Clifford Algebras*, chapter Honing Geometric Algebra for Its Use in the Computer Sciences. Springer, 2001.
- [2] Maxima Project, <http://maxima.sourceforge.net/docs/manual/maxima.html>. *Maxima 5.35.1 Manual*, Dec 2014.
- [3] Viktor Toth. Tensor manipulation in GPL Maxima. 2007.

# QUATERNIONIC POLYNOMIAL PROBLEMS FOR THE CONSTRUCTION OF PYTHAGOREAN-HODOGRAPH CURVES

R. T. Farouki <sup>a</sup>, G. Gentili <sup>b</sup>, C. Giannelli <sup>c</sup>, A. Sestini <sup>b</sup>, and C. Stoppato <sup>c</sup>

<sup>a</sup> Department of Mechanical and Aerospace Engineering  
University of California, Davis, CA 95616-5294, USA  
farouki@ucdavis.edu

<sup>b</sup> Dipartimento di Matematica e Informatica “U. Dini”  
Università di Firenze, Viale Morgagni 67/A, I-50134 Firenze, Italia  
gentili@math.unifi.it, alessandra.sestini@unifi.it

<sup>c</sup> Istituto Nazionale di Alta Matematica  
Unità di Ricerca di Firenze c/o DiMaI “U. Dini” Università di Firenze  
Viale Morgagni 67/A, I-50134 Firenze, Italia  
carlotta.giannelli@unifi.it, stoppato@math.unifi.it

A curve  $\mathbf{r} : \mathbb{R} \rightarrow \mathbb{R}^3$  having polynomial components  $\mathbf{r}_1(\xi), \mathbf{r}_2(\xi), \mathbf{r}_3(\xi) \in \mathbb{R}[\xi]$  is called a *Pythagorean-hodograph curve* (or PH curve) if the squared Euclidean norm of its tangent vector  $\mathbf{r}'(\xi)$  equals the square of a real polynomial  $\sigma(\xi) \in \mathbb{R}[\xi]$ :

$$\mathbf{r}'_1(\xi)^2 + \mathbf{r}'_2(\xi)^2 + \mathbf{r}'_3(\xi)^2 = \sigma(\xi)^2.$$

This requirement makes the arc length function of  $\mathbf{r}$  a polynomial function and it makes the unit tangent vector a rational curve. Details on these and other useful properties, which make PH curves particularly attractive for computer-aided design, can be found in [1].

Pythagorean-hodograph curves can be effectively studied using the quaternionic notation. Within the algebra of quaternions

$$\mathbb{H} = \mathbb{R} + \mathbf{i}\mathbb{R} + \mathbf{j}\mathbb{R} + \mathbf{k}\mathbb{R},$$

it turns out that a polynomial curve  $\mathbf{r} : \mathbb{R} \rightarrow \mathbf{i}\mathbb{R} + \mathbf{j}\mathbb{R} + \mathbf{k}\mathbb{R}$  is a PH curve if, and only if,

$$\mathbf{r}'(\xi) = \mathcal{A}(\xi)\mathbf{i}\mathcal{A}(\xi)^*$$

for some quaternionic polynomial  $\mathcal{A}(\xi) \in \mathbb{H}[\xi]$ . See again [1], which also points out the original references.

Because of this feature, many properties of the PH curve  $\mathbf{r}$  can be studied by means of the corresponding quaternionic polynomial  $\mathcal{A}(\xi)$ . The talk will describe the work under construction [2], which uses this technique to study the subclass of PH curves that admit *rational rotation–minimizing frames*. It will also mention the previous work [3] on a quadratic equation that arises in the construction of a surface patch with prescribed boundary curves and with PH isoparametric curves.

## REFERENCES

- [1] R. T. Farouki. *Pythagorean–Hodograph Curves: Algebra and Geometry Inseparable*. Springer, Berlin, 2008.
- [2] R. T. Farouki, G. Gentili, C. Giannelli, A. Sestini, and C. Stoppato. *Analysis of the set of polynomial curves that admit rational rotation–minimizing frames*. Under construction.
- [3] R. T. Farouki, G. Gentili, C. Giannelli, A. Sestini, and C. Stoppato. *Solution of a quadratic quaternion equation with mixed coefficients*. Preprint, 2015.





## Electromagnetism-like Equations for Fluids in Higher Dimensions

Murat Tanisli <sup>a</sup> and Neslihan Sahi <sup>b</sup>

<sup>a</sup> Anadolu University, Turkey, [mtanisli@anadolu.edu.tr](mailto:mtanisli@anadolu.edu.tr)  
[presenter, corresponding]

<sup>b</sup> Anadolu University, Turkey, [neslihansahin@anadolu.edu.tr](mailto:neslihansahin@anadolu.edu.tr)

Octons form a new algebraic structure, which is non-comutative but associative. For every different physical system; the scalar, pseudoscalar, vector and pseudovector parts of octon can be easily used for different purposes. It is clearly thought that octons will also be useful for representations of physical quantities in plasma and find an extensive application area in Physics. An alternative formulation is written for Fluid Maxwell Equations in terms of octon. It is demonstrated that the associative octons present a convenient and capable tool to describe the Fluid Maxwell Equations in compact and simple way.

Fluid is governed by Euler's equation of motion and equations of continuity, entropy and vorticity. Enthalpy and velocity vector are analogous to the vector potential and scalar potential in the classical electromagnetism. These can also be presented in terms of octons. The analogy between Fluid Maxwell Equations and Electromagnetic Maxwell equations in terms of octons is obtained and this is very interesting. When this analogy is used, a Faraday's-like law and Ampere's -like law in fluids are found. Equations of continuity for fluids can be written in terms of octons.

There are analogy between octonic fluid field and octonic electromagnetic field. Four Fluid Maxwell equations can be given by a single equation with octons. This equation is compact form so it is very useful and this situation is similar to compact Electromagnetic Maxwell equations.



# ON CAYLEY'S FACTORIZATION OF 4D ROTATIONS AND APPLICATIONS

Federico Thomas<sup>a</sup> and Alba Pérez-Gracia<sup>b</sup>

<sup>a</sup> Institut de Robòtica i Informàtica Industrial (CSIC-UPC)  
Llorens Artigas 4-6, 08028 Barcelona, Spain  
fthomas@iri.upc.edu [presenter, corresponding]

<sup>b</sup> Department of Mechanical Engineering  
Idaho State University, Pocatello, Idaho, 83209  
perealba@isu.edu

**ABSTRACT.** A 4D rotation can be decomposed into a left-isoclinic and a right-isoclinic rotation. This decomposition, known as Cayley's factorization of 4R rotations, can be performed using Elfrinkhof-Rosen method. In this paper, we present a more straightforward alternative approach based on the fact that there is an orthogonal basis, in the sense of Hilbert-Schmidt, for the space of  $4 \times 4$  real orthonormal matrices representing isoclinic rotations.

Cayley's factorization has many important applications. It can actually be seen as a unifying procedure to obtain the double quaternion representation of 4D rotations, the quaternion representation of 3D rotations, and the dual quaternion representation of 3D rigid-body transformations. Hence its interest in different Geometric Algebras.

As a practical application of the proposed method, it is shown how Cayley's factorization can be used to efficiently compute the screw parameters of 3D rigid-body transformations.

## 1. INTRODUCTION

Any rotation in  $\mathbb{R}^4$  can be seen as the composition of two rotations in a pair of orthogonal two-dimensional subspaces [1]. When the module of the rotated angles in these two subspaces are equal, the rotation is said to be isoclinic. It can be proved that any rotation in  $\mathbb{R}^4$  can be factored into the commutative composition of two isoclinic rotations. Cayley realized this fact when studying the double quaternion representation of rotations in  $\mathbb{R}^4$  [2]. This is why this factorization is herein named after him. It is actually Cayley whom we must thank for the correct development of quaternions as a representation of rotations, and for establishing the connection with the results published by Rodrigues three years before the discovery of quaternions [3]. Although Cayley's papers contain enough information to derive a practical method to perform this factorization, he wrote them before the full development of matrix algebra thus remaining somewhat cryptic to most modern readers.

The development of the first effective procedure for computing Cayley's factorization is attributed in [4] to Van Elfrinkhof [5]. Since this work, written in Dutch, remained unnoticed, other sources (see, for example, [6]) attribute to Rosen, a close collaborator of Einstein, the first procedure to obtain it [7]. The methods of Elfrinkhof and Rosen are equivalent. They are based on a clever manipulation of the 16 algebraic scalar equations resulting from imposing the factorization to an arbitrary 4D rotation matrix (see [4, 8] for a detailed explanation of this method).

In this paper, it is shown how Cayley's factorization admits a closed-form matrix formula whose derivation requires no other tools than elementary linear algebra. It is thus shown that, contrarily to what seemed unavoidable in previous formulations, there is no need to manipulate any set of algebraic equations.

---

This work was partially supported by the Spanish Ministry of Economy and Competitiveness through project DPI2014-57220-C2-2-P.

Cayley's factorization has important applications. Recently, it has been shown how it allows converting a rigid-body transformation in homogeneous coordinates to its corresponding dual quaternion representation in a very straightforward way [8]. This leads to a two-fold matrix and dual quaternion formalism for the representation of rigid-body transformations that permits a better understanding of dual quaternions and how they can be advantageously used in Kinematics. In this paper, it is shown how the application of the derived closed-form formula leads to a neat way of obtaining the dual quaternion representation of rigid-body transformation, thus providing a simple alternative to the standard approach based on the computation of screw parameters [9, p. 100]. Screw parameters can actually be seen as a by-product of Cayley's factorization.

This paper is organized as follows. Section 2 summarizes some basic facts about 4D rotations that are used in Section 3 to derive a spectral decomposition of isoclinic rotations. Then, a closed-form formula for the computation of Cayley's factorization is presented in Section 4. Section 5 gives details on a mapping between general displacements in 3D and some 4D rotations which is used in Section 6, together with the the derived closed-form formula, to obtain the dual quaternion representation of rigid-body transformation and, as a by-product, their screw parameters. Some conclusions are finally drawn in Section 7.

## 2. ISOCLINIC ROTATIONS

After a proper change in the orientation of the reference frame, an arbitrary 4D rotation matrix (*i.e.*, an orthogonal matrix with determinant +1) can be expressed as [10, Theorem 4]:

$$(1) \quad \begin{pmatrix} \cos \alpha_1 & -\sin \alpha_1 & 0 & 0 \\ \sin \alpha_1 & \cos \alpha_1 & 0 & 0 \\ 0 & 0 & \cos \alpha_2 & -\sin \alpha_2 \\ 0 & 0 & \sin \alpha_2 & \cos \alpha_2 \end{pmatrix}.$$

Thus, a 4D rotation is defined by two mutually orthogonal planes of rotation, each of which is fixed in the sense that points in each plane stay within the planes. Then, a 4D rotation has two angles of rotation,  $\alpha_1$  and  $\alpha_2$ , one for each plane of rotation, through which points in the planes rotate. All points not in the planes rotate through an angle between  $\alpha_1$  and  $\alpha_2$ . See [11] for details on the geometric interpretation of rotations in four dimensions.

If  $\alpha_1 = \pm\alpha_2$ , the rotation is called an *isoclinic rotation*. An isoclinic rotation can be *left-* or *right-isoclinic* (depending on whether  $\alpha_1 = \alpha_2$  or  $\alpha_1 = -\alpha_2$ , respectively) which can be represented by a rotation matrix of the form

$$(2) \quad \mathbf{R}^L = \begin{pmatrix} l_0 & -l_3 & l_2 & -l_1 \\ l_3 & l_0 & -l_1 & -l_2 \\ -l_2 & l_1 & l_0 & -l_3 \\ l_1 & l_2 & l_3 & l_0 \end{pmatrix},$$

and

$$(3) \quad \mathbf{R}^R = \begin{pmatrix} r_0 & -r_3 & r_2 & r_1 \\ r_3 & r_0 & -r_1 & r_2 \\ -r_2 & r_1 & r_0 & r_3 \\ -r_1 & -r_2 & -r_3 & r_0 \end{pmatrix},$$

respectively. Since (2) and (3) are rotation matrices, their rows and columns are unit vectors. As a consequence,

$$(4) \quad l_0^2 + l_1^2 + l_2^2 + l_3^2 = 1$$

and

$$(5) \quad r_0^2 + r_1^2 + r_2^2 + r_3^2 = 1.$$

Without loss of generality, we have introduced some changes in the signs and indices of (2) and (3) with respect to the notation used by Cayley [2, 6] to ease the treatment given below and to provide a neat connection with the standard use of quaternions for representing rotations in three dimensions.

Isoclinic rotation matrices have three important properties:

- (1) The product of two right- (left-) isoclinic matrices is a right- (left-) isoclinic matrix.
- (2) The product of a right- and a left-isoclinic matrix is commutative.
- (3) Any 4D rotation matrix, according to Cayley's factorization, can be decomposed into the product of a right- and a left-isoclinic matrix.

Then, a 4D rotation matrix, say  $\mathbf{R}$ , can be expressed as:

$$(6) \quad \mathbf{R} = \mathbf{R}^L \mathbf{R}^R = \mathbf{R}^R \mathbf{R}^L$$

where

$$(7) \quad \mathbf{R}^L = l_0 \mathbf{I} + l_1 \mathbf{A}_1 + l_2 \mathbf{A}_2 + l_3 \mathbf{A}_3$$

and

$$(8) \quad \mathbf{R}^R = r_0 \mathbf{I} + r_1 \mathbf{B}_1 + r_2 \mathbf{B}_2 + r_3 \mathbf{B}_3,$$

where  $\mathbf{I}$  stands for the  $4 \times 4$  identity matrix and

$$\mathbf{A}_1 = \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{A}_2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \quad \mathbf{A}_3 = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

$$\mathbf{B}_1 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{B}_2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}, \quad \mathbf{B}_3 = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}.$$

Therefore,  $\{\mathbf{I}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3\}$  and  $\{\mathbf{I}, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3\}$  can be seen, respectively, as bases for left- and right-isoclinic rotations.

Now, it can be verified that

$$(9) \quad \mathbf{A}_1^2 = \mathbf{A}_2^2 = \mathbf{A}_3^2 = \mathbf{A}_1 \mathbf{A}_2 \mathbf{A}_3 = -\mathbf{I},$$

and

$$(10) \quad \mathbf{B}_1^2 = \mathbf{B}_2^2 = \mathbf{B}_3^2 = \mathbf{B}_1 \mathbf{B}_2 \mathbf{B}_3 = -\mathbf{I}.$$

We can recognize in these two expressions the quaternion definition. Actually, (9) and (10) reproduce the celebrated formula that Hamilton carved into the stone of Brougham Bridge.

Expression (9) determines all the possible products of  $\mathbf{A}_1$ ,  $\mathbf{A}_2$ , and  $\mathbf{A}_3$  resulting in

$$(11) \quad \begin{aligned} \mathbf{A}_1 \mathbf{A}_2 &= \mathbf{A}_3, & \mathbf{A}_2 \mathbf{A}_3 &= \mathbf{A}_1, & \mathbf{A}_3 \mathbf{A}_1 &= \mathbf{A}_2, \\ \mathbf{A}_2 \mathbf{A}_1 &= -\mathbf{A}_3, & \mathbf{A}_3 \mathbf{A}_2 &= -\mathbf{A}_1, & \mathbf{A}_1 \mathbf{A}_3 &= -\mathbf{A}_2. \end{aligned}$$

Likewise, all the possible products of  $\mathbf{B}_1$ ,  $\mathbf{B}_2$ , and  $\mathbf{B}_3$  can be derived from expression (10). All these products can be summarized in the following product tables:

$$(12) \quad \begin{array}{c|cccc} & \mathbf{I} & \mathbf{A}_1 & \mathbf{A}_2 & \mathbf{A}_3 \\ \hline \mathbf{I} & \mathbf{I} & \mathbf{A}_1 & \mathbf{A}_2 & \mathbf{A}_3 \\ \mathbf{A}_1 & \mathbf{A}_1 & -\mathbf{I} & \mathbf{A}_3 & -\mathbf{A}_2 \\ \mathbf{A}_2 & \mathbf{A}_2 & -\mathbf{A}_3 & -\mathbf{I} & \mathbf{A}_1 \\ \mathbf{A}_3 & \mathbf{A}_3 & \mathbf{A}_2 & -\mathbf{A}_1 & -\mathbf{I} \end{array}$$

$$(13) \quad \begin{array}{c|cccc} & \mathbf{I} & \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 \\ \hline \mathbf{I} & \mathbf{I} & \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 \\ \mathbf{B}_1 & \mathbf{B}_1 & -\mathbf{I} & \mathbf{B}_3 & -\mathbf{B}_2 \\ \mathbf{B}_2 & \mathbf{B}_2 & -\mathbf{B}_3 & -\mathbf{I} & \mathbf{B}_1 \\ \mathbf{B}_3 & \mathbf{B}_3 & \mathbf{B}_2 & -\mathbf{B}_1 & -\mathbf{I} \end{array}$$

Moreover, it can be verified that

$$(14) \quad \mathbf{A}_i \mathbf{B}_j = \mathbf{B}_j \mathbf{A}_i.$$

which is actually a consequence of the commutativity of left- and right-isoclinic rotations. Then, in the composition of two 4D rotations, we have:

$$(15) \quad \mathbf{R}_1 \mathbf{R}_2 = (\mathbf{R}_1^L \mathbf{R}_1^R)(\mathbf{R}_2^L \mathbf{R}_2^R) = (\mathbf{R}_1^L \mathbf{R}_2^L)(\mathbf{R}_1^R \mathbf{R}_2^R).$$

It can be concluded that  $\mathbf{R}_i^L$  and  $\mathbf{R}_i^R$  can be seen either as  $4 \times 4$  rotation matrices or, when expressed as in (7) and (8) respectively, as unit quaternions.

### 3. A SPECTRAL DECOMPOSITION

The set of matrices  $\{\mathbf{I}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3\}$  form an orthogonal basis in the sense of Hilbert-Schmidt for the real Hilbert space of  $4 \times 4$  real orthonormal matrices representing left-isoclinic rotations. Then, (7) can be seen as a spectral decomposition. If we left-multiply it by each of the elements of the set  $\{\mathbf{I}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3\}$ , to obtain the different projection coefficients, we have that

$$(16) \quad l_0 \mathbf{I} = -\mathbf{R}^L + l_1 \mathbf{A}_1 + l_2 \mathbf{A}_2 + l_3 \mathbf{A}_3,$$

$$(17) \quad l_1 \mathbf{I} = -\mathbf{A}_1 \mathbf{R}^L - l_0 \mathbf{A}_1 + l_2 \mathbf{A}_3 - l_3 \mathbf{A}_2,$$

$$(18) \quad l_2 \mathbf{I} = -\mathbf{A}_2 \mathbf{R}^L - l_0 \mathbf{A}_2 - l_1 \mathbf{A}_3 + l_3 \mathbf{A}_1,$$

$$(19) \quad l_3 \mathbf{I} = -\mathbf{A}_3 \mathbf{R}^L - l_0 \mathbf{A}_3 + l_1 \mathbf{A}_2 - l_2 \mathbf{A}_1.$$

Then, by iterative substituting and rearranging terms in (16)-(19), we conclude that the coefficients of the spectral decomposition (7) can be expressed as:

$$(20) \quad l_0 \mathbf{I} = -\frac{1}{4} (-\mathbf{R}^L + \mathbf{A}_1 \mathbf{R}^L \mathbf{A}_1 + \mathbf{A}_2 \mathbf{R}^L \mathbf{A}_2 + \mathbf{A}_3 \mathbf{R}^L \mathbf{A}_3),$$

$$(21) \quad l_1 \mathbf{I} = -\frac{1}{4} (\mathbf{R}^L \mathbf{A}_1 + \mathbf{A}_1 \mathbf{R}^L + \mathbf{A}_3 \mathbf{R}^L \mathbf{A}_2 - \mathbf{A}_2 \mathbf{R}^L \mathbf{A}_3),$$

$$(22) \quad l_2 \mathbf{I} = -\frac{1}{4} (\mathbf{R}^L \mathbf{A}_2 + \mathbf{A}_2 \mathbf{R}^L + \mathbf{A}_1 \mathbf{R}^L \mathbf{A}_3 - \mathbf{A}_3 \mathbf{R}^L \mathbf{A}_1),$$

$$(23) \quad l_3 \mathbf{I} = -\frac{1}{4} (\mathbf{R}^L \mathbf{A}_3 + \mathbf{A}_3 \mathbf{R}^L + \mathbf{A}_2 \mathbf{R}^L \mathbf{A}_1 - \mathbf{A}_1 \mathbf{R}^L \mathbf{A}_2).$$

Likewise, we can consider the set of matrices  $\{\mathbf{I}, \mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3\}$  as an orthogonal basis in the sense of Hilbert-Schmidt for right-isoclinic rotations. Then, the coefficients in (8) could also be obtained as above.

## 4. MATRIX FORMULATION OF CAYLEY'S FACTORIZATION

Let us define the following matrix linear operators for arbitrary 4D rotation matrices:

$$\begin{aligned}
 \mathcal{L}_0(\mathbf{R}) &= -\frac{1}{4}(-\mathbf{R} + \mathbf{A}_1\mathbf{R}\mathbf{A}_1 + \mathbf{A}_2\mathbf{R}\mathbf{A}_2 + \mathbf{A}_3\mathbf{R}\mathbf{A}_3), \\
 \mathcal{L}_1(\mathbf{R}) &= -\frac{1}{4}(\mathbf{R}\mathbf{A}_1 + \mathbf{A}_1\mathbf{R} + \mathbf{A}_3\mathbf{R}\mathbf{A}_2 - \mathbf{A}_2\mathbf{R}\mathbf{A}_3), \\
 \mathcal{L}_2(\mathbf{R}) &= -\frac{1}{4}(\mathbf{R}\mathbf{A}_2 + \mathbf{A}_2\mathbf{R} + \mathbf{A}_1\mathbf{R}\mathbf{A}_3 - \mathbf{A}_3\mathbf{R}\mathbf{A}_1), \\
 \mathcal{L}_3(\mathbf{R}) &= -\frac{1}{4}(\mathbf{R}\mathbf{A}_3 + \mathbf{A}_3\mathbf{R} + \mathbf{A}_2\mathbf{R}\mathbf{A}_1 - \mathbf{A}_1\mathbf{R}\mathbf{A}_2).
 \end{aligned}
 \tag{24}$$

According to (20)-(23),  $\mathcal{L}_i(\mathbf{R}^L) = l_i\mathbf{I}$ ,  $i = 0, \dots, 3$ . Then, using the commutativity property of left- and right-isoclinic rotations, it is straightforward to prove that

$$\mathcal{L}_i(\mathbf{R}) = \mathcal{L}_i(\mathbf{R}^L\mathbf{R}^R) = \mathcal{L}_i(\mathbf{R}^L)\mathcal{L}_i(\mathbf{R}^R) = l_i\mathbf{R}^R.
 \tag{25}$$

We arrive at an important conclusion:  $\mathcal{L}_i(\mathbf{R})$  and  $\mathbf{R}^R$  are equal up to a constant factor. Moreover, since  $\mathbf{R}^R$  is a rotation matrix, the norm of any of the rows and columns of  $\mathcal{L}_i(\mathbf{R})$  is  $l_i^2$ . This provides a straightforward way to compute Cayley's factorization. Indeed,

$$\mathbf{R}^R = \frac{\mathcal{L}_i(\mathbf{R})}{[\det(\mathcal{L}_i(\mathbf{R}))]^{1/4}}
 \tag{26}$$

and

$$\mathbf{R}^L = \mathbf{R}(\mathbf{R}^R)^T = \frac{\mathbf{R}\mathcal{L}_i(\mathbf{R})^T}{[\det(\mathcal{L}_i(\mathbf{R}))]^{1/4}}.
 \tag{27}$$

Observe that we have two possible solutions for the factorization depending on the sign chosen for the quartic roots in (26)-(27). Actually, it is well-known that quaternions provide a double covering of the space of rotations.

## 5. A USEFUL MAPPING

Chasles' theorem states that the general spatial motion of a rigid body can be produced a rotation about an axis and a translation along the direction given by the same axis. Such a combination of translation and rotation is called a general screw motion [12]. In the definition of screw motion, a positive rotation corresponds to a positive translation along the screw axis by the right-hand rule.

In Fig. 1, a screw axis is defined by  $\mathbf{n} = (n_x, n_y, n_z)^T$ , a unit vector defining its direction, and  $q\mathbf{p}$ , the position vector of a point lying on it, where  $\mathbf{p} = (p_x, p_y, p_z)^T$  is also a unit vector. The angle of rotation  $\theta$  and the translational distance  $d$  are called the screw parameters. These screw parameters together with the screw axis completely define the general displacement of a rigid body.

In [8], the following mapping between 3D transformations in homogeneous coordinates and a subset of 4D rotation matrices was proposed:

$$\mathbf{T} = \begin{pmatrix} \mathbf{R}_{3 \times 3} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \iff \tilde{\mathbf{T}} = \begin{pmatrix} \mathbf{R}_{3 \times 3} & \boldsymbol{\varepsilon}\mathbf{t} \\ -\boldsymbol{\varepsilon}\mathbf{t}^T\mathbf{R}_{3 \times 3} & 1 \end{pmatrix},
 \tag{28}$$

where  $\boldsymbol{\varepsilon}$  is the standard dual unit ( $\boldsymbol{\varepsilon}^2 = 0$ ). The interesting thing about this mapping is that the Cayley's factorization of  $\tilde{\mathbf{T}}$  can be expressed as  $\tilde{\mathbf{T}}^L\tilde{\mathbf{T}}^R$  where

$$\tilde{\mathbf{T}}^R = \cos\left(\frac{\hat{\theta}}{2}\right)\mathbf{I} + \sin\left(\frac{\hat{\theta}}{2}\right)(\hat{n}_x\mathbf{B}_1 + \hat{n}_y\mathbf{B}_2 + \hat{n}_z\mathbf{B}_3)
 \tag{29}$$



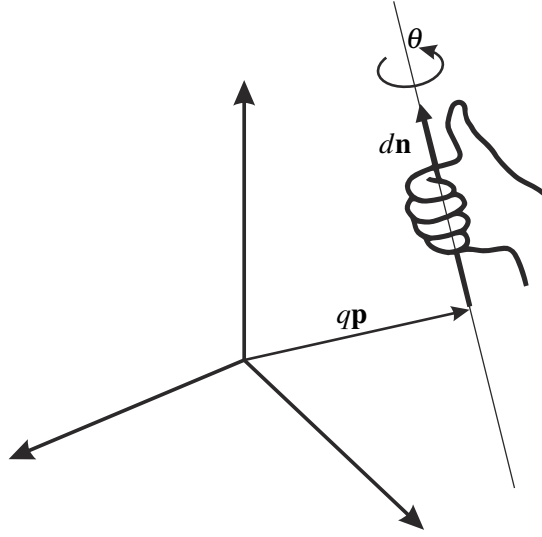


FIGURE 1. Geometric parameters used to describe a general screw motion.

where  $\hat{\mathbf{n}} = (\hat{n}_x, \hat{n}_y, \hat{n}_z)^T = \mathbf{n} + \varepsilon q (\mathbf{p} \times \mathbf{n})$  and  $\hat{\theta} = \theta + \varepsilon d$  (see [8] for details). Thus, the coefficients of the Cayley's factorization of  $\tilde{\mathbf{T}}$  give us the screw parameters of  $\mathbf{T}$ . This is exemplified in the next section.

## 6. EXAMPLE

Let us consider, as an example, the transformation in homogeneous coordinates

$$(30) \quad \mathbf{T} = \begin{pmatrix} 0 & 0 & 1 & 4 \\ 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & 7 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Then, according to (28),

$$(31) \quad \tilde{\mathbf{T}} = \begin{pmatrix} 0 & 0 & 1 & 4\varepsilon \\ 1 & 0 & 0 & -3\varepsilon \\ 0 & 1 & 0 & 7\varepsilon \\ 3\varepsilon & -7\varepsilon & -4\varepsilon & 1 \end{pmatrix},$$

and, according to (24),

$$(32) \quad \begin{aligned} \mathcal{L}_0(\tilde{\mathbf{T}}) &= -\frac{1}{4} \left( -\tilde{\mathbf{T}} + \mathbf{A}_1 \tilde{\mathbf{T}} \mathbf{A}_1 + \mathbf{A}_2 \tilde{\mathbf{T}} \mathbf{A}_2 + \mathbf{A}_3 \tilde{\mathbf{T}} \mathbf{A}_3 \right) \\ &= -\frac{1}{4} \begin{pmatrix} 1 & -1-11\varepsilon & 1+4\varepsilon & 1+\varepsilon \\ 1+11\varepsilon & 1 & -1-\varepsilon & 1+4\varepsilon \\ -1-4\varepsilon & 1-\varepsilon & 1 & 1+11\varepsilon \\ -1-\varepsilon & -1-4\varepsilon & -1-11\varepsilon & 1 \end{pmatrix} \\ &= -\frac{1}{4} [\mathbf{I} + (1+\varepsilon)\mathbf{B}_1 + (1+4\varepsilon)\mathbf{B}_2 + (1+11\varepsilon)\mathbf{B}_3] \end{aligned}$$

Therefore,

$$(33) \quad \tilde{\mathbf{T}}^R = -\frac{1}{4l_0} [\mathbf{I} + (1+\varepsilon)\mathbf{B}_1 + (1+4\varepsilon)\mathbf{B}_2 + (1+11\varepsilon)\mathbf{B}_3].$$

Since, according to (5),  $r_0^2 + r_1^2 + r_2^2 + r_3^2 = 1$ , we have that

$$(34) \quad \frac{1}{16l_0^2} [1 + (1+\varepsilon)^2 + (1+4\varepsilon)^2 + (1+11\varepsilon)^2] = \frac{4+32\varepsilon}{16l_0^2} = 1.$$

Thus,

$$(35) \quad l_0 = \pm \left( \frac{1}{2} + 2\varepsilon \right).$$

If we take the negative sign (remember that the solution is unique up to a sign change), we conclude that

$$r_0 = \frac{1}{2+8\varepsilon} = \frac{1}{2} - 2\varepsilon, \quad r_1 = \frac{1+\varepsilon}{2+8\varepsilon} = \frac{1}{2} - \frac{3}{2}\varepsilon,$$

$$r_2 = \frac{1+4\varepsilon}{2+8\varepsilon} = \frac{1}{2}, \quad r_3 = \frac{1+11\varepsilon}{2+8\varepsilon} = \frac{1}{2} + \frac{7}{2}\varepsilon.$$

That is, the unit dual quaternion representing the transformation in homogenous coordinates given by  $\mathbf{T}$  can be expressed as:

$$(36) \quad \tilde{\mathbf{T}}^R = \left( \frac{1}{2} - 2\varepsilon \right) \mathbf{I} + \left( \frac{1}{2} - \frac{3}{2}\varepsilon \right) \mathbf{B}_1 + \left( \frac{1}{2} \right) \mathbf{B}_2 + \left( \frac{1}{2} + \frac{7}{2}\varepsilon \right) \mathbf{B}_3.$$

To obtain the corresponding screw parameters for this rigid-body transformation, we can simply identify (36) with (29). This identification yields:

$$(37) \quad \cos\left(\frac{\hat{\theta}}{2}\right) = 0.5 - 2\varepsilon,$$

$$(38) \quad \hat{n}_x \sin\left(\frac{\hat{\theta}}{2}\right) = 0.5 - 1.5\varepsilon,$$

$$(39) \quad \hat{n}_y \sin\left(\frac{\hat{\theta}}{2}\right) = 0.5,$$

$$(40) \quad \hat{n}_z \sin\left(\frac{\hat{\theta}}{2}\right) = 0.5 + 3.5\varepsilon.$$

Solving (37) for  $\hat{\theta} = \theta + \varepsilon d$  we get

$$(41) \quad \theta = \frac{2}{3}\pi \text{ and } d = \frac{8}{\sqrt{3}}.$$

Then, substituting  $\hat{\theta} = \frac{2}{3}\pi + \varepsilon \frac{8}{\sqrt{3}}$  in (38)-(40), we conclude that

$$(42) \quad \mathbf{n} = \left( \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}} \right)^T,$$

and

$$(43) \quad q(\mathbf{p} \times \mathbf{n}) = \left( -\frac{6\sqrt{3}-1}{6}, -\frac{1}{6}, \frac{14\sqrt{3}-1}{6} \right)^T.$$

If  $\mathbf{p}$  and  $\mathbf{n}$  are assumed to be orthogonal, it is concluded from (43) that  $q = \sqrt{\frac{699-40\sqrt{3}}{36}}$ . As a consequence,

$$(44) \quad \mathbf{p} \times \mathbf{n} = (-0.3742, -0.03984, 0.9264)^T.$$

Finally, using (42) and (44), we have that

$$(45) \quad \mathbf{p} = \mathbf{n} \times (\mathbf{p} \times \mathbf{n}) = (-0.5579, 0.7509, -0.1930)^T.$$

## 7. CONCLUSIONS

Cayley's factorization can be used to obtain the double quaternion representation of 4D rotations and, as a particular case, the quaternion representation of 3D rotations. Nevertheless, a much more interesting application arises when observing that this factorization can also straightforwardly be used to derive the dual quaternion representation of 3D rigid-body transformations. This requires the application of a simple mapping between 3D rigid-body transformations in homogeneous coordinates and 4D rotation matrices. To exemplify this, a detailed worked example has been presented.

The conversion of a rigid-body transformation in homogeneous coordinates to its corresponding dual quaternion counterpart has traditionally been performed by computing the screw parameters of the rigid-body transformation. We have shown how Cayley's factorization provides an alternative and straightforward way to perform this conversion.

#### REFERENCES

- [1] J.L. Weiner and G.R. Wilkens, "Quaternions and rotations in  $E^4$ ," *The American Mathematical Monthly*, Vol. 112, No. 1, pp. 69-76, 2005.
- [2] A. Cayley, "Recherches ultérieures sur les déterminants gauches," *The Collected Mathematical Papers Of Arthur Cayley*, article 137, p. 202-215, Cambridge University Press, 1891.
- [3] A. Cayley, "On certain results relating to quaternions," *Philosophical Magazine*, Vol. 26, pp. 141-145, 1845.
- [4] J.E. Mebius, *Applications of Quaternions to Dynamical Simulation, Computer Graphics and Biomechanics*, Ph.D. Thesis, Delft University of Technology, 1994.
- [5] L. van Elfrinkhof, "Eene eigenschap van de orthogonale substitutie van de vierde orde," *Handelingen van het zesde Nederlandsch Natuur- en Geneeskundig Congres* (Acts of the sixth Dutch nature and medical congress), pp. 237-240, Delft, 1897.
- [6] F.L. Hitchcock, "Analysis of rotations in euclidean four-space by sedenions," *Journal of Mathematics and Physics*, University of Massachusetts, Vol. 9, No. 3, pp. 188-193, 1930.
- [7] N. Rosen, "Note on the general Lorentz transformation," *Journal of Mathematics and Physics*, Vol. 9, pp. 181-187, 1930.
- [8] F. Thomas, "Approaching dual quaternions from matrix algebra," *IEEE Transactions on Robotics*, Vol. 30, No. 5, pp. 1037-1048, 2014.
- [9] J. Angeles, *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*, Springer, 2006.
- [10] C.Y. Hsiung and G.Y. Mao, *Linear Algebra*, Allied Publishers, 1998.
- [11] L. Pertti, *Clifford algebras and spinors*, Cambridge University Press, 2001.
- [12] J.K. Davidson and K.H. Hunt, *Robots and Screw Theory: Applications of Kinematics and Statics to Robotics*, Oxford Univ Press, Oxford, 2004.

## Quaternionic $F(p, q, s)$ Function spaces

Augusto G. Miss <sup>a</sup>, Lino F. Reséndis O. <sup>b</sup>  
Luis M Tovar S. <sup>c</sup>

<sup>a</sup>UPIITA IPN México City, MEXICO  
[amiss@ipn.mx](mailto:amiss@ipn.mx)

<sup>b</sup>UAM ATZCAPOTZALCO, México City, MEXICO  
[lfro@correo.azc.uam.mx](mailto:lfro@correo.azc.uam.mx)

<sup>c</sup>ESFM, IPN México City, MEXICO  
[tovar@esfm.ipn.mx](mailto:tovar@esfm.ipn.mx) [presenter, corresponding]

**Abstract:** In this talk we present the necessary tools to generalize the function spaces  $F(p, q, s)$  and  $F_{-o}(p, q, s)$  introduced by Zhao in [1] to the case of the monogenic functions defined in the three dimensional unit ball. Obtaining at the same time the generalization of  $Qp$  monogenic spaces presented by Guerlebeck *et al* in [2] and El-Sayed *et al* in [3].

## REFERENCES

- [1] Zhao, R.: On a general family of function spaces. Ann. Acad. Sci. Fenn. Math. Diss. No. 105 (1996)
- [2] Gürlebeck, K., Kähler, U., Shapiro, M.V., Tovar, L.M.: On  $Qp$ -Spaces of Quaternion-Valued Functions. As part of Complex Variables, vol. 39. OPA (Overseas Publishers Associations), N.Y. (1999).
- [3] El-Sayed Ahmed A, Gürlebeck, K., Reséndis, L.F., Tovar S., Luis M.: Characterizations for  $B_{p,q}$  spaces in Clifford analysis. Complex Var. Elliptic Equ. 51(2), 119–136 (2006)



# GEOMETRIC CONTROL OF ROBOTIC SNAKES BASED ON CGA: (4,5,7,8-10) FILTRATION

J. Hrdina, A. Návrat, P. Vašík<sup>a</sup> and R. Matoušek<sup>b</sup>

<sup>a</sup> Institute of Mathematics, Faculty of Mechanical Engineering,  
Brno University of Technology, Brno, Czech Republic,  
hrdina@fme.vutbr.cz [corresponding], navrat.a@fme.vutbr.cz, vasik@fme.vutbr.cz [presenter]

<sup>b</sup> Institute of Automation and Computer Science, Faculty of Mechanical Engineering,  
Brno University of Technology, Brno, Czech Republic,  
matousek@fme.vutbr.cz

ABSTRACT. We describe the local control of a (6–8)–link snake like robot endowed with omnidirectional wheels on two links (head and tail). All calculations including the position, direct kinematics, differential kinematics and inverse kinematics are described in the terms of CGA only.

## 1. INTRODUCTION

Let a non–holonomic system be described by the appropriate dynamic (Pfaff) system of ODEs, which defines the vectors of admissible motion directions (w.r.t. the controlling parameters). These vectors together with the Lie bracket operation generate a Lie algebra corresponding to the original system. Consequently, a non–holonomic system is said to be locally controllable if the linear span of its appropriate Lie algebra generators is of the same dimension as the configuration space (Rachevsky–Chow Theorem). Furthermore, while generating the Lie algebra, one obtains a natural filtration (w.r.t. the number of Lie bracket applications). The elements of the higher filtration parts correspond to the motions, which can not be realized locally but are necessary for local controllability. A general interpretation of such motions is crucial for local controllability description and realization. The main goal of the talk is to employ an advanced mathematical tool of the Conformal Geometric Algebra (CGA) in the snake robot modeling in order to achieve the optimal control of particular solutions.

## 2. CONFORMAL GEOMETRIC ALGEBRA – CGA

Let  $\mathbb{R}^{4,1}$  denote a vector space  $\mathbb{R}^5$  equipped with the scalar product of signature (4, 1), let  $\mathcal{C}l(4, 1)$  denote the corresponding Clifford algebra, i.e. a free, associative and distributive algebra as a span of the set  $\{e_1, e_2, e_3, e_+, e_-\}$  such that the following identities are satisfied:

$$e_1^2 = e_2^2 = e_3^2 = e_+^2 = 1, \quad e_-^2 = -1,$$
$$e_i e_j = -e_j e_i, \quad i \neq j, \quad i, j \in \{1, 2, 3, +, -\}.$$

In this case, we get  $2^5 = 32$ –dimensional vector space. Let us note that the norm in  $\mathbb{R}^{4,1}$  can be understood as a vector square  $x^2 = \|x\|^2$ . Now, we define two additional products on  $\mathbb{R}^{4,1}$  based on the geometric one for any  $u, v \in \mathbb{R}^{4,1}$ , *dot product* and *wedge product*, respectively:

$$u \cdot v = \frac{1}{2}(uv + vu), \quad u \wedge v = \frac{1}{2}(uv - vu)$$

and thus the formula for the geometric product can be derived as

$$uv = u \cdot v + u \wedge v.$$

---

*Date:* June 29, 2015.

The authors were supported by a grant no. FSI-S-14-2290.

Generally, the wedge (outer) product of two basis blades  $E_i$  and  $E_j$ , with  $k = \text{gr}(E_i)$ ,  $l = \text{gr}(E_j)$  is defined as

$$E_i \wedge E_j := \langle E_i E_j \rangle_{k+l}$$

and the dot (inner) product is defined as

$$E_i \cdot E_j = \begin{cases} \langle E_i E_j \rangle_{|k-l|} & i, j, > 0 \\ 0 & i = 0 \text{ or } j = 0, \end{cases}$$

where  $\text{gr}(E)$  is a grade of the basis blade  $E$  and  $\langle \rangle_k$  is the grade projection into the blade of the grade  $k$ . To work with CGA effectively, we have to define a new basis of  $\mathbb{R}^{4,1}$  as a set  $\{e_1, e_2, e_3, e_0, e_\infty\}$  such that  $e_0 = \frac{1}{2}(e_- + e_+)$  and  $e_\infty = (e_- - e_+)$ . Consequently, the following properties hold:

$$\begin{aligned} e_0^2 &= 0, \quad e_\infty^2 = 0, \quad e_\infty e_0 = -1 - e_- \wedge e_+, \quad e_0 e_\infty = -1 + e_- \wedge e_+, \\ e_\infty e_0 &= -e_0 e_\infty - 2. \end{aligned}$$

In CGA, we can represent the basis geometric elements by the following multivectors from  $\mathcal{C}l(4,1)$ :

$$\begin{aligned} \text{point } x &\rightsquigarrow Q = x + \frac{1}{2}x^2 e_\infty + e_0 \\ \text{sphere of radius } & \\ r \text{ and center } C &\rightsquigarrow S = C - \frac{1}{2}r^2 e_\infty \\ \text{point pair } Q_1, Q_2 &\rightsquigarrow P = Q_1 \wedge Q_2. \end{aligned}$$

In CGA (in fact in GA generally), any transformation of the element  $O$  is realized by conjugation

$$O \mapsto M O \tilde{M}$$

where  $M$  is the appropriate multivector from  $\mathcal{C}l(4,1)$ . For instance, the translation in the direction  $t = t_1 e_1 + t_2 e_2 + t_3 e_3$  is realized by the multivector

$$M := T = 1 - \frac{1}{2}t e_\infty$$

and the rotation around the axis  $L$  by angle  $\phi$  is realized by the multivector

$$M := R = \cos \frac{\phi}{2} - L \sin \frac{\phi}{2}$$

where  $L = a_1 e_2 e_3 + a_2 e_1 e_3 + a_3 e_1 e_2$ . For more information refer to books [2, 8] or papers [3, 4].

### 3. THE FLAG STRUCTURE

The snake robot described in this paper consists of 6–8 rigid links of constant length 2 interconnected by motorized joints. Except for the first and the last, to each link, in the center of mass, a pair of wheels is attached to provide an important snake-like property that the ground friction in the direction perpendicular to the link is considerably higher than the friction of a simple forward move. In particular, this prevents the slipping sideways. To determine the actual position of a snake robot we need the set of 8–10 generalized coordinates

$$(1) \quad q = (x, y, \theta, \Phi_i, i \in \{1, \dots, N\}) \in M,$$

where  $N \in \{5, 6, 7\}$  and  $M = \mathbb{R}^2 \times (\mathbb{S}^1)^{N+1}$  which describes the configuration of the snake robot.

The control theory generally leads to a nonholonomic system

$$\dot{q} = \sum_{i=1}^m u_i X_i(q), \quad q \in M,$$

where  $X_i$  are vector fields on  $M$ , where  $m \in \mathbb{N}$ ,  $m < \dim M$ . In particular, the controllability of the nonholonomic system is fully characterized by the properties of the Lie algebra generated by  $X_1, \dots, X_m$ . We define

$$\Delta^1 = \text{span}\{X_1, \dots, X_m\}$$

and  $\Delta^{s+1} = \Delta^s + [\Delta^1, \Delta^s]$ , where  $[\Delta^1, \Delta^s] = \text{span}\{[X, Y] : X \in \Delta^1, Y \in \Delta^s\}$ . The Lie algebra  $\bar{\Delta}$  generated by  $X_1, \dots, X_m$  is defined as

$$\bar{\Delta} = \bigcup_{s \geq 1} \Delta^s.$$

Let us note that our system satisfies Chow's condition, i.e.  $\bar{\Delta}(q) = T_q M, \forall q \in M$  and the vectors at  $q \in M$  form a flag of subspaces of  $T_q M$ , that is

$$\Delta^1(q) \subset \Delta^2(q) \subset \dots \subset \Delta^{r-1}(q) \subset \Delta^r(q) = T_q M,$$

where  $r = r(p)$  is so-called degree of nonholonomy at  $p$ . Set  $n_i(q) = \dim \Delta^i(q)$ . The  $r$ -tuple of integers  $(n_1(q), \dots, n_r(q))$  is called the growth vector at  $q$ . In our cases the growth vectors are

$$(4, 5, 7, 8), (4, 5, 7, 9), (4, 5, 7, 10)$$

and the degree of nonholonomy is 4. The structure of the flag may also be described by another sequence of integers. We define the weights at  $q$ ,  $w_i = w_i(q)$ ,  $i = 1, \dots, m$ , by setting  $w_j = s$  if  $n_{s-1}(q) < j \leq n_s(q)$ , where  $n_0 = 0$ . In our cases the weights are

$$(1, 1, 1, 1, 2, 3, 3, 4), (1, 1, 1, 1, 2, 3, 3, 4, 4), (1, 1, 1, 1, 2, 3, 3, 4, 4, 4).$$

As the first and the last link are endowed with the omnidirectional wheels and do not affect the control, the controlling Lie algebra is of a special form, particularly it contains a two-dimensional center  $\mathcal{Z}$ . The flag structures on factor space  $\bar{\Delta}/\mathcal{Z}$  are characterized by the growth vectors

$$(2, 3, 5, 6), (2, 3, 5, 7), (2, 3, 5, 8),$$

where the last example is of full dimension as the corresponding P. Hall basis contains exactly 8 elements. As a result, we discuss all possibilities with the degree of nonholonomy equal to 4 after the center factorisation.

#### 4. KINEMATICS

Note that a fixed coordinate system  $(x, y)$  is attached. The points  $p_i := (x_i, y_i)$ , denote the centers of mass of each link. To describe the robotic snake we use as a central object the set of point pairs

$$(P_0, P_2, P_4, \dots, P_N)$$

where  $P_0 = Q_0 \wedge Q_1$ ,  $P_1 = Q_1 \wedge Q_2$  and  $P_N = Q_N \wedge Q_{N+1}$ , where  $Q_i$  are the joint points and head and tail points. Consequently, the kinematic equations can be assessed and if we consider the projections

$$Q_i = -\frac{\sqrt{P_i \cdot P_i} + P_i}{e_\infty \cdot P_i}, \quad Q_{i+1} = \frac{\sqrt{P_i \cdot P_i} + P_i}{e_\infty \cdot P_i},$$

we are able to express the coordinates of every point from any point pair. The coordinates of particular position of link centers are the following

$$p_i = P_i e_\infty \tilde{P}_i, \quad \text{s.t.} \quad P_i = R_{\Phi_i} \cdots R_{\Phi_1} R_\theta T_{x,y} P_{i,0} \tilde{T}_{x,y} \tilde{R}_\theta \tilde{R}_{\Phi_1} \cdots \tilde{R}_{\Phi_i}$$



and, for the robotic snake initial position  $x = y = \theta = \Phi_i = 0$ , the appropriate point pairs are denoted by  $P_{i,0}$  and calculated directly, e.g.

$$\begin{aligned} P_{1,0} &= (e_0) \wedge (2e_1 + 2e_\infty + e_0) = 2e_0e_1 - 2e_+e_-, \\ P_{2,0} &= (2e_1 + 2e_\infty + e_0) \wedge (4e_1 + 8e_\infty + e_0) = 2e_0e_1 + 8e_1e_\infty - 6e_+e_-, \\ P_{3,0} &= (4e_1 + 8e_\infty + e_0) \wedge (6e_1 + 18e_\infty + e_0) = 2e_0e_1 + 24e_1e_\infty - 10e_+e_-, \\ &\vdots \end{aligned}$$

Now, the transformations corresponding to the generalized coordinates can be written as

$$\begin{aligned} T_{x,y} &= 1 - \frac{1}{2}(xe_1 + ye_2)e_\infty, \\ T_{Q_i} &= 1 - \frac{1}{2}Q_i e_\infty, \\ R_0 &= \cos \frac{\theta}{2} - L_0 \sin \frac{\theta}{2}, \text{ where } L_0 = T_{x,y}e_1e_2\tilde{T}_{x,y}, \\ R_i &= \cos \frac{\Phi_1}{2} - L_i \sin \frac{\Phi_1}{2}, \text{ where } L_i = T_{Q_i}e_1e_2\tilde{T}_{Q_i}. \end{aligned}$$

The direct kinematics for the snake robot is obtained similarly as the kinematics for serial robot arms [10]. In general, it is given by a succession of generalised rotations  $R_i$  and it is valid for all geometric objects, including point pairs. A point pair  $P$  in a general position is computed from its initial position  $P_0$  as follows

$$(2) \quad P = \prod_{i=1}^n R_i P_0 \prod_{i=1}^n \tilde{R}_{n-i+1}.$$

## 5. DIFFERENTIAL KINEMATICS

Unlike the fixed serial robot arms, we allow  $R_i$  to be also a translation. We view translations as degenerate rotations. Then the differential kinematics is expressed by means of the total differential as follows

$$dP = \sum_{j=1}^n \partial_{q_j} \left( \prod_{i=1}^n R_i P_0 \prod_{i=1}^n \tilde{R}_{n-i+1} \right) dq_j.$$

**Theorem 5.1.** *Let  $c$  be a centre of a sphere  $S$  (including a point pair as a  $0D$ -sphere) whose final position is given by the kinematic chain (2). Then the differential kinematics of  $c$  is given by*

$$\dot{c} = \sum_{j=1}^n [c \cdot L'_j] dq_j.$$

*Proof.* See [4]. □

Concretely, we obtain the system

$$\begin{aligned} \dot{p}_1 &= [p_1 \cdot e_1 e_\infty] \dot{x} + [p_1 \cdot e_2 e_\infty] \dot{y} + [p_1 \cdot L_0] \dot{\theta}, \\ \dot{p}_i &= [p_2 \cdot e_1 e_\infty] \dot{x} + [p_2 \cdot e_2 e_\infty] \dot{y} + [p_i \cdot L_0] \dot{\theta} + [p_2 \cdot L_1] \dot{\Phi}_1 + \cdots + [p_2 \cdot L_1] \dot{\Phi}_i, \end{aligned}$$

which in the matrix notation is of the form

$$(3) \quad \dot{p} = J\dot{q},$$

where  $q$  are our coordinates (1) and  $J = (j_{kl})$  is a matrix with the elements defined by scalar products of points and axes

$$\begin{aligned} j_{i1} &= [p_i \cdot e_1 e_\infty], \quad j_{i2} = [p_i \cdot e_2 e_\infty], \\ j_{ik} &= [p_i \cdot L_{k-3}] \text{ for } 3 \leq k < 3+i, \\ j_{ik} &= 0 \text{ for } 3+i \leq k. \end{aligned}$$

As the wheels do not slip to the side direction, the velocity constraint condition is satisfied for each link  $i$  and in terms of CGA can be written as

$$(4) \quad \dot{p}_i \wedge P_i \wedge e_\infty = 0.$$

Thus if we substitute (3) in (4), we obtain a system of linear ODEs, which has a simple Pfaff matrix form

$$(5) \quad A\dot{q} = 0,$$

where  $A = (a_{ij})$  is a matrix with the elements defined by

$$(6) \quad a_{ik} = j_{ik} \wedge P_i \wedge e_\infty.$$

Note that the entries of  $A$  are multiples of  $e_1 e_2 e_+ e_-$ . Taking the conjugate and multiplying with  $e_3$   $A$  can be considered simply as a matrix over the field of functions. For example, the solution of this system with respect to  $\hat{\theta}$  parameterized by  $\dot{x}, \dot{y}$ , (i.e.  $\dot{x} = t_1$  and  $\dot{y} = t_2$ ) is of the form

$$\hat{\theta} = -\frac{[p_1 \cdot e_1 e_\infty] \wedge P_1 \wedge e_\infty}{[p_1 \cdot L_0] \wedge P_1 \wedge e_\infty} t_1 - \frac{[p_1 \cdot e_2 e_\infty] \wedge P_1 \wedge e_\infty}{[p_1 \cdot L_0] \wedge P_1 \wedge e_\infty} t_2.$$

If we denote by  $e^*$  the dual to  $e$  in CGA which is realized by the multiplication of the inverse unit pseudoscalar, the straightforward computation leads to  $[p_1 \cdot L_0] \wedge P_1 \wedge e_\infty = 2e_3^*$ , i.e. the solution always exists, because  $([p_1 \cdot L_0] \wedge P_1 \wedge e_\infty)^{-1} = -\frac{1}{2}e_3^*$ .

The singular posture of the system is in the case that the wheel axes, i.e. lines perpendicular to each link containing the link center point, intersect in precisely one point or are parallel, see Figure 2. In our setting this is one condition only because in CGA the parallel lines intersect in exactly one point which is  $e_\infty$ . It is easy to see that this happens in such case that all joints lie on a single circle, i.e. in CGA they satisfy a simple condition

$$(7) \quad Q_0 \wedge Q_1 \wedge Q_2 \wedge Q_i = 0, \quad \forall i \in \{3, \dots, N+1\}.$$

Finally, note that the non-singular solution forms a 2-dimensional distribution which can be parametrized e.g. as follows:

$$(8) \quad \dot{q} = G \begin{pmatrix} t_1 \\ t_2 \end{pmatrix},$$

where  $G = (g_{ij})$  is a control matrix. Thus if we consider the snake robot configuration space with coordinates (1) as manifold  $M$ , the solution above forms a set of vector fields  $\{g_1, g_2, g_3, g_4\}$ , such that  $g_3, g_4 \in \mathcal{L}$ .

It is clear, that the space  $\text{span}\{g_1, g_2\}$  determines the set of accessible spatial velocity vectors and thus, taking into account the vector field flows  $\exp(tg_1)$ ,  $\exp(tg_2)$ , the possible trajectories of the snake robot. On the other hand, due to non-commutativity of  $\exp(tg_1)$ ,  $\exp(tg_2)$ , the robot can move even along the flow of the Lie bracket by means of the composition

$$\exp(-tg_2) \circ \exp(-tg_1) \circ \exp(tg_2) \circ \exp(tg_1).$$

Extending this idea, the space  $D_q$  of all movement directions at the point  $q$  is given by all possible Lie brackets of  $g_1(q)$  and  $g_2(q)$  and the resulting vector fields. From the geometric control theory point of view, it is quite necessary that the dimension of  $D_q$  is equal to the dimension of

the tangent space  $T_qM, q \in M$ , which in our case is 6,7 and 8, respectively. Note that this is the condition on the model local controllability given by the Rashevsky–Chow Theorem.

## 6. NOTES ON THE INVERSE KINEMATICS

Note that the geometric meaning of the inner product of two conformal vectors  $U$  and  $V$  is the following.

TABLE 1. Geometric meaning of the inner product

$U \cdot V$	Plane	Sphere	Point
Plane	Angle between planes	Euclidean distance from center	Euclidean distance
Sphere	Euclidean distance from center	Distance measure	Distance measure
point	Euclidean distance	Distance measure	Euclidean distance

In our case, we have the set of point pairs  $\{P_0, \dots, P_N\}$  which determines the mechanism configuration uniquely. Using the notation of Section 4, the set of the admissible points  $\{Q_0, \dots, Q_{N+1}\}$  with respect the set of point pairs  $\{P_0, \dots, P_N\}$  has to satisfy the equations

$$(9) \quad Q_i \cdot Q_{i+1} = 2,$$

where  $i \in \{0, \dots, N\}$ . A point in the configuration space  $M$  is determined by the following direct computation:

$$\begin{aligned} x &= Q_0 \cdot e_1, \\ y &= Q_0 \cdot e_2, \\ \cos \theta &= (P_1 \wedge e_\infty) \cdot (e_0 \wedge e_1 \wedge e_\infty), \\ \cos \phi_i &= (P_i \wedge e_\infty) \cdot (P_{i+1} \wedge e_\infty). \end{aligned}$$

Various inverse problems can be solved by these equations. For instance assume the position of the first and the last link is fixed (as it does not affect the control process) and all possible resulting configurations are computed. We shall use the property of the scalar product of a point  $Q$  and a sphere  $S$  that the number  $\sqrt{2|Q \cdot S|}$  determines the Euclidean distance of  $Q$  and the point of tangency on  $S$  appropriate to the tangent containing  $Q$ . To avoid the singular initial positions we suggest to employ the assumption for small  $\varepsilon \in \mathbb{R}$

$$2 - \varepsilon \leq Q_i \cdot (Q_{i+2} - 2e_\infty) \leq 2 + \varepsilon,$$

which reads that the angle  $\phi_i$  is in the vicinity of  $\frac{\pi}{2}$  or less restrictive assumption

$$\text{sgn}((P_i \wedge e_\infty) \cdot (P_{i+1} \wedge e_\infty)) = -\text{sgn}((P_i \wedge e_\infty) \cdot (P_{i+1} \wedge e_\infty))$$

which reads that the angle signum is changing in each joint.

## REFERENCES

- [1] L. Gonzalez–Jimenez, O. Carbajal–Espinosa, A. Loukianov, and E. Bayro–Corrochano, *Robust Pose Control of Robot Manipulators Using Conformal Geometric Algebra*, Advances in Applied Clifford Algebras **24,2** (2014), 533–552.
- [2] D. Hildenbrand, *Foundations of Geometric Algebra Computing*, Springer, Geometry and Computing, Vol. 8, 2013.
- [3] J. Hrdina, A. Návrat, P. Vašík, *3-link robotic snake control based on CGA*, Advances in Applied Clifford Algebras (accepted, 2015)
- [4] J. Hrdina and P. Vašík, *Notes on differential kinematics in conformal geometric algebra approach*, Recent Advances in Soft Computing, Advances in Intelligent Systems and Computing, Springer, volume 378., pages 363–374, 2015.
- [5] P. Liljebäck, K.Y. Pettersen, Ø. Stavdahl and J.T. Gravdahl, *Snake Robots, Modelling, Mechatronics and Control*, Springer, Advances in Industrial Control, 2013.

- [6] R. Matoušek and A. Návrat, *Trident snake control based on CGA*, Recent Advances in Soft Computing, Advances in Intelligent Systems and Computing vol 378, Springer, 2015, pp 375–385.
- [7] R. M. Murray, L. Zexiang and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 1994.
- [8] Ch. Perwass, *Geometric Algebra with Applications in Engineering*, Springer, Geometry and Computing, Vol. 4, 2009.
- [9] J.M. Selig, *Geometric Fundamentals of Robotics*, Springer, Monographs in Computer Science, 2004.
- [10] J. Zamora–Esquivel and E. Bayro–Corrochano, *Kinematics and diferential kinematics of binocular robot heads*, Robotics and Automation, ICRA 2006.
- [11] J. Zamora–Esquivel and E. Bayro–Corrochano, *Parallel Forward Dynamics: a geometric approach*, In *The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010.



# COMPUTATIONAL ELECTROMAGNETISM BY THE METHOD OF LEAST ACTION

**Terje G. Vold**

Continuum Technology, Inc.  
Anacortes, WA 98221 USA  
terjevold@gmail.com [presenter, corresponding]

**ABSTRACT.** A new general method of computational electromagnetism based on extremizing the electromagnetic action using the geometric algebra of space-time is described. Special cases include a boundary element method and a finite element method. These methods are derived and discussed, and compared with the well known Method of Moments and Finite Difference Time Domain method.

## INTRODUCTION

Computational electromagnetism is the application of computational methods to simulate electromagnetic fields. Although this is a well developed field [4], this development has been constrained by the use of 3D vector algebra and calculus. Although Hestenes described the use in electrodynamics of the geometric algebra of space-time [1] in the same decade that computational electromagnetism began to receive significant attention by applied physicists and electronic engineers [3], there has been little awareness of geometric algebra by researchers in this field.

This paper describes an application of the geometric algebra of space-time to computational electromagnetism first described in [6], starting with the Lagrangian and Hamilton's principle [2]. The result is a short development of simple but heretofore unused fundamental equations useful in computational electromagnetism, that result in methods that are similar to but significantly different from methods commonly used today.

The new methods described here are similar to, but simpler and with advantages over, two popular and well studied methods: the Method of Moments and various Finite Element methods. These new methods are described below and compared with the popular methods.

## 1. LEAST ACTION WITH BOUNDARIES

This section reviews the application of Lagrangian techniques to electromagnetism, with the addition of integrals over boundaries on which normal derivatives of the potential may be discontinuous. These boundary integrals are crucial in the next section where these results are modified to apply to computational electromagnetism.

We begin by considering variation  $\delta$  of the electromagnetic action,

$$(1) \quad \delta \int L[A, \nabla A] d^4x = 0$$

under all possible variations  $\delta A$  of a continuous space-time vector potential  $A$ .

We use the electromagnetic Lagrangian

$$(2) \quad L[A, \nabla A] = (1/2)F * F + U[F] - A \cdot J$$

where  $F = \nabla \wedge A$ , the interaction energy  $U[F]$  of the polarizable medium in the field also depends on the 4-velocity and polarizability of the medium, and the free charged current density  $J$  is specified, not varied, in this example.

After an integration by parts, we obtain

$$(3) \quad \int \delta A \cdot (\nabla \cdot G - J) d^4x = 0$$

where

$$(4) \quad G = F + \partial_F U[F]$$

Since this is true for any variation  $\delta A$  of  $A$ , the term in parenthesis must equal zero at all points of space-time which gives us the Euler-Lagrange equation (i.e., Maxwell's inhomogeneous equation):

$$(5) \quad \nabla \cdot G - J = 0$$

If derivatives of  $A$ , and therefore  $F$  and  $G$ , are discontinuous across boundary points — possibly because the medium polarizability is discontinuous across a boundary or because we have simply chosen basis potentials that have discontinuous derivatives across boundary points — instead of (3) we obtain

$$(6) \quad \int_V \delta A \cdot (\nabla \cdot G - J) d^4x + \int_S \delta A \cdot (n \cdot \Delta G - K) d^3x = 0$$

where the first integral is over space-time volume  $V$  excluding boundary points, and the second integral is over space-time boundaries or surfaces  $S$  across which derivatives of  $A$  may be discontinuous, with change in  $G$  across the boundary denoted by  $\Delta G$ , a unit vector normal to the boundary by  $n$ , and the boundary charged current density by  $K$ . Each term in parentheses must be zero, which gives us Maxwell's inhomogeneous equation

$$(7) \quad \nabla \cdot G - J = 0$$

and the usual associated boundary condition

$$(8) \quad n \cdot \Delta G - K = 0$$

## 2. COMPUTATIONAL LEAST ACTION

For the numerical work we're interested in, we set the space-time vector potential  $A$  equal to a sum of basis functions  $a_i[x]$  of space-time position  $x$  times coefficients  $c_i$ ,

$$(9) \quad A = a_i c_i$$

with a corresponding expression for  $F = f_i c_i$  where  $f_i = \nabla \wedge a_i$ . Note that although  $G = G[f_i c_i]$ , in the general case of nonlinear  $G[F]$ ,  $G \neq G[f_i] c_i$ . We consider only variations of  $A$  that we can represent by variations  $\delta c_i$  of the coefficients  $c_i$ : that is,  $\delta A = a_i \delta c_i$ . Since any set of values of variations  $\delta c_i$  is allowed, equation (6) then becomes equivalent to the set of equations

$$(10) \quad \int_V a_i \cdot (\nabla \cdot G - J) d^4x + \int_S a_i \cdot (n \cdot \Delta G - K) d^3x = 0$$

indexed by  $i$ . This set of equation determines the coefficients  $c_i$  that extremize the action via their expression in  $G[f_i c_i]$ . If  $U$  is bilinear in  $\nabla \wedge A$ , then  $G = g_j c_j$  with  $g_j = G[f_j]$ , and therefore these equations are linear in the  $c_j$  and so can be easily solved for the  $c_j$ .

These equations are especially useful in computational applications if the integrals over volumes  $V$  are identically zero because we then have to compute integrals over only boundaries or surfaces  $S$ . We therefore consider choosing a set of basis functions  $a_i$  such that for any set of coefficient values  $c_i$ , the first term in parenthesis, and therefore the first integral, is identically zero. With such a choice of basis functions, we extremize the action by choosing coefficients  $c_i$  such that

$$(11) \quad \int_S a_i \cdot (n \cdot \Delta G - K) d^3x = 0$$

This is the main equation used by the methods described in this paper.

Signs in (11) can be confusing with space-time's mixed signature. This may be clarified by using a local orthonormal frame  $\{e_0, e_1, e_2, e_3\}$  with three frame vectors tangential to the boundary, and replacing the symbol  $n$  for the normal vector with that of the reciprocal frame vector  $e^n$  normal to the boundary for some  $n \in \{0, 1, 2, 3\}$ . Then

$$(12) \quad \nabla \cdot G d^4x = (e^\alpha e_\alpha \cdot \nabla) \cdot G dx^k dx^l dx^m dx^n$$

where  $\alpha$  is summed over  $\{0, 1, 2, 3\}$  and  $\{k, l, m, n\}$  is some permutation of  $\{0, 1, 2, 3\}$ . For a boundary of thickness  $dx^n$  with  $G$  changing linearly across the boundary and negligible rates of change tangential to the boundary, we can replace the sum over  $\alpha$  with the single index  $n$ :

$$(13) \quad \nabla \cdot G d^4x = e^n \cdot (dx^n \partial_{x^n} G) dx^k dx^l dx^m = e^n \cdot (G[x + \frac{dx^n}{2} e_n] - G[x - \frac{dx^n}{2} e_n]) dx^k dx^l dx^m$$

$$(14) \quad \equiv e^n \cdot \Delta G d^3x$$

This shows that  $\nabla \cdot G d^4x$  is independent of the choice of sign of  $e_n$  since  $e_n \cdot e^n = 1$  implies that if the sign of  $e_n$  is changed, the sign of  $e^n$  is, too; but it also shows that the overall sign is different for parts of a boundary with time-like versus space-like normal vectors  $e_n$  and  $e^n$ .

Two especially useful methods of choosing a set of basis functions such that  $\nabla \cdot G - J = 0$  for any choice of coefficient values  $c_i$  are described briefly below, and in greater detail in following sections.

**2.1. Regions of uniform linear polarizability.** If the medium is linearly polarizable with discontinuous polarizability across boundaries and uniform polarizability within each region enclosed by a boundary, then  $G[F]$  is a linear function of  $F = \nabla \wedge A$  so that  $G = g_i c_i$  where  $g_i = G[f_i]$  and  $f_i = \nabla \wedge a_i$ . In this case we choose basis functions  $a_i = a_i[x]$  that are defined by their values at boundary points plus the requirement that they satisfy an appropriate Euler-Lagrange equation at all non-boundary points: We may choose one basis function, say  $a_0$ , to satisfy  $\nabla \cdot g_0 - J = 0$  and choose all other basis functions,  $a_i$  for  $i \neq 0$ , to satisfy  $\nabla \cdot g_i = 0$ , so that  $\nabla \cdot G = J$  identically at all non-boundary points for any choice of basis coefficients  $c_i$ . The value of any basis function may be found at any non-boundary point from these boundary values by using Green's 3rd Identity as detailed in a later section. Application of (11) to solve for the solution coefficients  $c_i$  then results in a boundary element method that is similar to, but quite different from, the well established Method of Moments used in computational electrodynamics.

**2.2. Regions of nonlinear polarizability.** If  $G$  is not a linear function of  $F = \nabla \wedge A$  in one or more regions of space-time, we may divide each region of interest into contiguous 4-simplices (tetrahedrons are 3-simplices, triangles 2-simplices) that share adjacent faces - that is, into a simplicial complex - and choose  $A = A[x]$  to be a linear function of position  $x$  at all points within each simplex, so that  $F$  is constant within each simplex and  $\nabla \cdot G$  is therefore zero. For example, we may choose each  $a_i = a_i[x]$  to be linear in position  $x$  within each simplex and set it equal to a nonzero parameter at one vertex of the simplices and zero at every other vertex. We may also approximate any charged current volume density inside a simplex with a corresponding charged current surface density  $K$  on faces of the simplices and set  $J = 0$  within 4-simplices. The only contribution to the integral for extremizing the action is then from the boundaries between the simplices, that is, the faces of the simplices. This results in a set of difference equations that are analogous to, but different from, equations used by conventional Finite Difference Time Domain methods. Because of the simple structure of these difference equations, they are easy to use and analyze and may be more accurate for comparable computation time. For example, the points defining the simplices need not be in any regular array but can be quite irregularly placed in space-time, as long as a simplicial complex can be drawn between them that covers the region of interest.



With an interaction term in the Lagrangian representing magnetic hysteresis, possibly using a statistical mechanical model, method 2.2 may also be useful for nonlinear magnetic media with hysteresis.

These two methods 2.1 and 2.2 may be used in the same problem: for example, a region containing nonlinear magnetic media may be treated with method 2.2, while linearly polarizable or empty space outside of this region may be treated with method 2.1.

Variations include simplified equations for situations with symmetries, such as translation or rotational symmetry of the polarizability, and harmonic time variation in a chosen frame.

### 3. UNIFORM LINEAR POLARIZABILITY

This section develops in greater detail the special case of section 2.1 above: space-time filled with bounded regions, each filled with media having uniform polarizability that is a linear function of the field  $F$ . In practice we first compute "medium quantities" such as  $A_m$ ,  $J_m$ , and  $G_m$  instead of corresponding "physical quantities"  $A$ ,  $J$ , and  $G$ , as outlined in a later section, but for notational simplicity we omit the subscript  $m$  in this and other sections.

We restrict analysis to the simple case of harmonically varying electromagnetic quantities in a rest frame identified by a 4-velocity  $\gamma_0$  in which all media and boundaries are stationary. For convenience, we represent harmonic variation in this rest frame with the real or imaginary parts of complex quantities including both scalars and space-time vectors. Space-time vectors are represented by non-bold type such as  $x$  and  $\nabla$ , and relative 3D space vectors by bold type such as  $\mathbf{x} = x \wedge \gamma_0$  and  $\nabla = \gamma_0 \wedge \nabla$ . Then the space-time vector potential  $A$ , for example, may be given as the space-time vector valued function  $A = Re[A[\mathbf{x}]Exp[-i\omega t]]$  of  $\mathbf{x} = x \wedge \gamma_0$  and  $t = x \cdot \gamma_0$ , with the  $Re[...]$  function and  $Exp[...]$  factor implied in most algebra.

Our goal is to be able to quickly calculate the electromagnetic field  $F$  at any point of space-time that contains i) any number of stationary physical objects of any shape, each made of a material with uniform electric and magnetic polarizability and electrical conductivity of any values as long as they are linear functions of the field, and ii) any specified charged current density  $J$  oscillating at any specified frequency  $\omega$ . This can be done using Greens 3rd Identity

$$(15) \quad A[\mathbf{x}'] = \int_S (\nabla_n \phi[\mathbf{x} - \mathbf{x}'] A[\mathbf{x}] - \phi[\mathbf{x} - \mathbf{x}'] \nabla_n A[\mathbf{x}]) d^3x + \int_V \phi[\mathbf{x} - \mathbf{x}'] J[\mathbf{x}] d^4x$$

where an acceptable Green's function is

$$(16) \quad \phi[\mathbf{X}] = \frac{1}{4\pi|\mathbf{X}|} Exp[ik|\mathbf{X}|]$$

with medium wavenumber  $k$ , given the potential  $A$  and its directional derivative  $\nabla_n A$  normal to the boundary at all boundary points. The goal of the method in this case is therefore to calculate an acceptable approximation to these quantities  $A$  and  $\nabla_n A$  on the boundary.

The method calculates these boundary quantities for the final solution in two steps:

i) Basis potential functions  $a_i$  are first chosen so that  $A = a_i c_i$  extremizes the integral of the Lagrangian over non-boundary points for any choice of coefficients  $c_i$ , but does not, generally, extremize the integral of the Lagrangian over boundary points. This results in basis functions, each one a function defined at every point of every region in space (although characterized and uniquely identified by values at only boundary points), such that  $A$  satisfies the Euler-Lagrange equation within each region but not across the boundaries between regions.

ii) The particular set of coefficient values  $c_i$  that extremizes the integral of the Lagrangian over all boundary point is then chosen; that is, that satisfies equation (11). The linear combinations  $a_i c_i$  and  $\nabla_n a_i c_i$  then give the desired boundary quantities  $A$  and  $\nabla_n A$ .

Details of these steps follow.

**3.1. Choice of Basis Functions.** The first step, choosing basis potential functions  $a_i$  that extremize the integral of the Lagrangian over all non-boundary points, is performed as follows.

Extremizing the integral of the Lagrangian over region points is equivalent to choosing functions that satisfy the Euler-Lagrange equation (Maxwell's inhomogeneous equations for the electromagnetic Lagrangian) at all region points. This Euler-Lagrange equation has a "gauge freedom" - it is independent of  $\nabla \cdot A$  - that we conveniently eliminate by also requiring that the Lorenz condition is satisfied. With the Lorenz condition, the Euler-Lagrange equation reduces to the simpler wave equation, so we choose basis functions  $a_i$  that satisfy the wave equation and the Lorenz condition.

A unique solution to any 2nd-order differential equation, including the wave equation, in any region enclosed by a boundary, is identified by specifying the value of the solution at every point on the boundary (Dirichlet boundary conditions), the normal derivative of the solution at every point on the boundary (Neuman boundary conditions), or some combination of value and normal derivative (mixed boundary conditions). In this case of a space-time vector valued function such as an electromagnetic potential  $a_i$ , we have 4 degrees of freedom for each the value and normal derivative, for a total of 8 degrees of freedom at each point of the boundary; we must specify the vector value of the function (Dirichlet), its vector-valued normal derivative (Neuman), or some combination totaling 4 degree of freedom (mixed) at each point to uniquely specify a solution. We choose to specify mixed boundary conditions: the three space-time tangential components  $nn \wedge a_i$  of the potential  $a_i$  (equal to the two components of the 3D vector potential that are tangential to the 2D boundary, and the scalar potential) and the normal component  $nn \cdot (\nabla_n a_i)$  of the normal derivative  $\nabla_n a_i \equiv n \cdot \nabla a_i$  at every point on the boundary. These boundary conditions identify a unique solution to the wave equation.

But further, we choose  $nn \cdot (\nabla_n a_i)$  to satisfy the Lorenz condition,  $0 = \nabla \cdot a_i$ , on the boundary: The divergence may be expanded to rewrite the Lorenz condition as  $0 = (nn \cdot \nabla)(nn \cdot a_i) + (nn \wedge \nabla)(nn \wedge a_i)$ , then multiplying this by  $n$  on the left and using the notation  $\nabla_n = n \cdot \nabla$ , we have  $nn \cdot (\nabla_n a_i) = -(n \wedge \nabla)(nn \wedge a_i)$ . This choice of  $nn \cdot (\nabla_n a_i)$  ensures that the Lorenz condition is satisfied at all boundary points. This, along with  $a_i$  satisfying the wave equation  $\nabla^2 a_i = 0$  at all points enclosed by the boundary, guarantees that  $a_i$  satisfies the Lorenz condition at all points enclosed by the boundary: Taking the divergence of  $\nabla^2 a_i = 0$  gives  $\nabla^2(\nabla \cdot a_i) = 0$ , so  $\nabla \cdot a_i$  satisfies the wave equation with the boundary condition that it equals 0, which has the unique solution  $\nabla \cdot a_i = 0$  everywhere.

We now have shown that a solution  $a_i$  that satisfies the wave equation and the Lorenz condition is uniquely and exactly specified by our choice of just the three tangential components  $nn \wedge a_i$  of  $a_i$  at all boundary points. We also can immediately calculate the one normal component  $nn \cdot (\nabla_n a_i)$  of  $\nabla_n a_i$  at all boundary points by applying the Lorenz condition. But we need all 4 components of  $a_i$  and all 4 component of  $\nabla_n a_i$  on the boundary in order to apply Green's 3rd Identity to calculate the potential  $a_i$  at any point in the region enclosed by the boundary. We find the 4 still-unknown components as follows.

We define good approximations to the desired boundary functions  $nn \cdot a_i$  and  $nn \wedge (\nabla_n a_i)$  by choosing functional forms that depend on a finite number of parameters, and then use Green's 3rd Identity to write constraints on the parameters that determine their values.

For example, we might cover the boundary with triangular patches having a total of  $V$  vertices, and for each component, choose the functional form to be a simple linear function of position on each triangle, with parameters given by the values of  $nn \cdot a_i$  and  $nn \wedge (\nabla_n a_i)$  at the vertices of the triangles. Then we apply Green's 3rd Identity to each vertex, resulting in  $V$  space-time vector equations that can be solved for the  $4V$  degrees of freedom of  $nn \cdot a_i$  and  $nn \wedge (\nabla_n a_i)$  at the  $V$  vertices.

This simple example for functional forms results in electromagnetic basis fields  $f_i = \nabla \wedge a_i$  that may be discontinuous at the boundaries between patches. These discontinuities can be removed by using smooth, though more complex, functional forms for the boundary and for the potential, such as by using Bezier techniques.

In summary, we do the following for each basis function  $a_i$ :

a) We specify the 3 space-time vector boundary components  $nn \wedge a_i$  (given in the rest frame of the boundary by the two 3D vector potential components that are tangential to the 2D boundary, and the scalar potential) exactly at every boundary point (typically by a parameterized boundary potential function with any desired number of parameters), then calculate the value of  $nn \cdot (\nabla_n a_i)$  exactly at every boundary point on both sides of the boundary from the specified boundary components using the Lorenz condition. These values exactly determine the corresponding solution that satisfies both the wave equation and the Lorenz condition throughout the region enclosed by the boundary.

b) Although this *determines* the exact potential  $a_i$  at all enclosed points, we cannot easily *calculate* values at arbitrary points enclosed by the boundary until we also have values of the remaining 4 boundary components  $nn \cdot a_i$  and  $nn \wedge (\nabla_n a_i)$ . We find approximate values of these components by representing them with 4V parameterized functions and then determining optimal values of the parameters by choosing them so that each of the 4 components of Green's 3rd Identity is exactly satisfied at V boundary points. This is most easily done using the media quantities  $A_m$  and  $n \cdot \nabla_m$  defined in a later section, instead of  $A$  and  $n \cdot \nabla \equiv \nabla_n$ .

**3.2. Continuity of the Normal Component of the Potential.** For potentials defined in this way, the normal component of each basis potential,  $nn \cdot a_i$ , will generally be discontinuous across a boundary, while a fundamental physical assumption is that the potential is continuous everywhere in space-time. This section shows that this discontinuity can always be removed by choosing a different gauge; but knowing this, we conclude that we should continue using the original gauge because of its computational simplicity.

To remove the discontinuity in  $nn \cdot a_i$ , we can perform a gauge transformation to  $a'_i = a_i + \nabla \chi_i$  for the potential on one side of the boundary, with  $\chi_i[x]$  at each boundary point  $x$  chosen so that  $n \cdot \nabla \chi_i = -n \cdot a_i$ , resulting in  $n \cdot a'_i = 0$  at all boundary points  $x$ . Then, in order for the tangential components of  $a'_i$  to remain continuous, we ensure that  $nn \wedge \nabla \chi_i = 0$  at the boundary by setting  $\chi_i = 0$  on the boundary. We choose  $\chi_i$  to be *any* function  $\chi_i[x]$  that satisfies these two boundary requirements, and then choose the gauge condition  $\nabla \cdot a_i = \nabla^2 \chi_i$  for our new potential  $a'_i$ .

All components of this potential  $a'_i$  are continuous at all points, and  $a'_i$  results in the same field  $f = f'_i$  since  $\nabla \wedge \nabla \chi_i = 0$ . However,  $a'_i$  will generally be more difficult to compute because of its gauge condition, so we continue use the potential  $a_i$  with its discontinuous normal component but simple Lorenz gauge condition, knowing that it is equivalent - at least for non-quantum applications - to a continuous physical potential.

**3.3. Choice of Linear Combination of Basis Functions.** At this point we have a simple computable expression for each potential basis function  $a_i[x]$  and its normal derivative  $\nabla_n a_i[x]$  at every boundary point  $x$ . From this we can immediately calculate the corresponding basis field  $g_i[x]$  as a linear function of the corresponding basis field  $f_i = \nabla \wedge a_i$  at any boundary point, on either side. Note that even though the normal component of  $a_i$  can be made continuous by a different choice of gauge, all components of the normal *derivative* of  $a_i$  are generally different on the two sides of a boundary separating regions with different polarizabilities.

We now extremize the action boundary integral under variation of the coefficients in the linear combination of basis functions by solving the set of linear equations

$$(17) \quad \int_S a_i \cdot (n \cdot \Delta g_j) d^3 x c_j = \int_S a_i \cdot K d^3 x$$

indexed by  $i$ , each with a sum over  $j$ . This is a matter of simply evaluating the boundary integrals and solving the resulting set of linear equations for the  $c_j$ , allowing us to write the solution  $A[\mathbf{x}] = a_i[\mathbf{x}]c_i$  and  $F[\mathbf{x}] = f_i[\mathbf{x}]c_i$ .

Although derivation and algebraic manipulation of the basic equations is generally easier using space-time algebra, seeing these equations in terms of electric and magnetic fields in the rest frame of the medium is helpful for practical considerations:

$$(18) \quad \int_S (\phi_i(\mathbf{n} \cdot \Delta \mathbf{d}_j) - \mathbf{a}_i \cdot (\mathbf{n} \times \Delta \mathbf{h}_j)) c_j d^2x = \int_S (\phi_i \sigma - \mathbf{a}_i \cdot \mathbf{K}) d^2x$$

where the integral is over all points of the 3D boundary, which in this case of stationary boundaries and harmonic oscillation in the rest frame of the boundary, is a 2D boundary in 3D space that is swept uniformly through time.

**3.4. Implementation Details.** This section describes some algebraic and computational details that are useful for implementation.

**3.4.1. Functions specifying boundaries and potentials on boundaries.** To avoid singularities in computational electromagnetism, we generally need smooth boundaries; that is, normal vectors to the boundary must be continuous. To avoid discontinuities in the fields  $F$  and  $G$ , the potential  $A$  must also be have continuous first derivatives. Bezier techniques [5], especially representations of triangular patches, turn out to be very useful and convenient for both representation of boundaries and of potentials and fields.

Perhaps the simplest approximate representation of a boundary is a covering of flat triangles, although this has normals that are discontinuous across triangle edges. Similarly simple parameterized functions to represent values of the potential and of the normal derivative of the potential on such a boundary are also linear functions of position on each triangle. We know that the physical field  $F$  near such edges diverges, corresponding a diverging value of the normal derivative of the potential near edges, so approximation of these normal derivatives  $\nabla_n a_i$  on the boundary with linear functions is expected to result in some numerical error, and up to a few percent discrepancy is observed between the parameterized potential value on the boundary and the potential value at a boundary point calculated from all boundary values using Green's Third Identity.

Quantitative results for a simple application of this is given in Figures 1 through 4 below, showing the total electric and magnetic fields near an icosahedron filled with a medium having permittivity  $\epsilon_m = 1 + 3i$  and permeability  $\mu_m = 3$ , while a magnetic dipole approximately 10 radii to the right is oscillating with a frequency  $\omega = 0.1$  resulting in radiation from the right with a wavelength of about 63 radii. The plane of the page is tangential to the dipole axis in Figures 1 and 2 depicting the magnetic field  $B$ , and perpendicular to the dipole axis in Figures 3 and 4 depicting the electric field  $E$ . The scale of vector lengths differs between figures.

These linear functions are in fact the simplest Bezier representations. Higher order Bezier representations of a continuous boundary with specified vertices and boundary normals of triangular patches can be created fairly easily; with slightly more work, continuous normals all along each edge can be specified and described. Potentials can be similarly specified, resulting in smooth boundaries and smooth first derivatives of the potential.

**3.4.2. Media Quantities.** In polarizable media, all quantities and equations become more complex because the charged current  $J$  includes induced  $J$  that depends on the field. But in the case of media with polarization that is linear in the field, we can write linear transformations of the relative vector and scalar of the potential  $A$  and of the vector derivative  $\nabla$  in the rest frame of the medium, such that all equations have the same form as those in vacuum but with different constants.

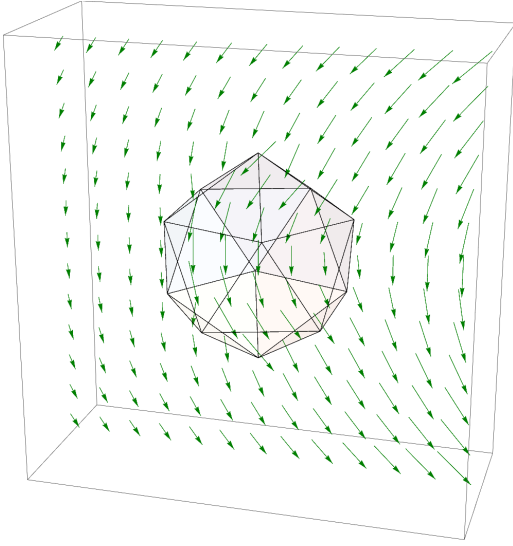


FIGURE 1. Real part of B field

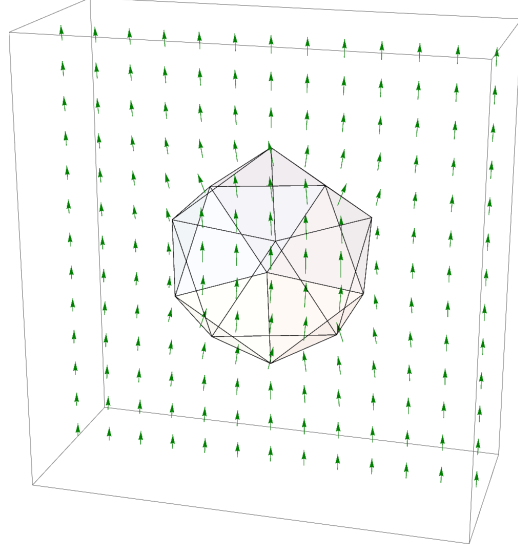


FIGURE 2. Imaginary part of B field

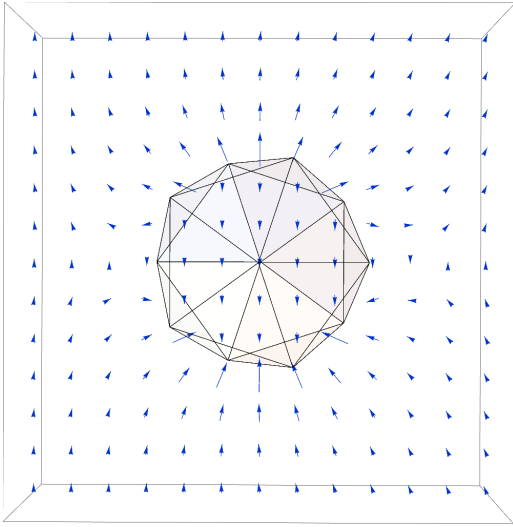


FIGURE 3. Real part of E field

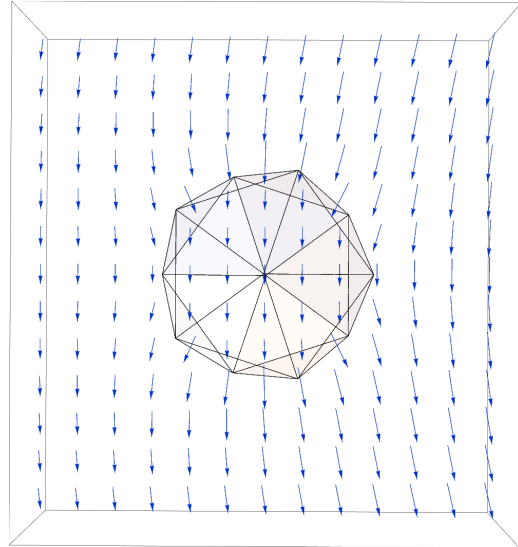


FIGURE 4. Imaginary part of E field

Space-time vector quantities such as  $A$  are related to scalar and 3D space relative scalars and vectors in a frame with 4-velocity  $\gamma_0$ . We write these to give our notation for free space or nonpolarizable media for which the total charged current density  $J$  equals the free charged current density  $J_f$ :

$$(19) \quad J_f = (\rho_f + \mathbf{J}_f)\gamma_0$$

$$(20) \quad A = (\phi + \mathbf{A})\gamma_0$$

$$(21) \quad \nabla = (\partial_0 - \nabla)\gamma_0$$

$$(22) \quad \nabla \wedge A = F = \mathbf{E} + \mathbf{I}\mathbf{B}$$

In polarizable media, the total charged current is the sum of free and bound parts,  $J = J_f + J_b$ , expressed as the divergence of a space-time polarization bivector  $P$ , where  $P$  itself is expressible in terms of an electric space vector polarization  $\mathbf{P}_E$  and magnetic space vector polarization  $\mathbf{P}_M$  in the medium rest frame identified by  $\gamma_0$ ,

$$(23) \quad J_b = \nabla \cdot P = (\rho_b + \mathbf{J}_b)\gamma_0 = (-\nabla \cdot \mathbf{P}_E + \partial_t \mathbf{P}_E - \mathbf{I} \nabla \wedge \mathbf{P}_M)\gamma_0$$

where

$$(24) \quad P = P \cdot \gamma_0 \gamma_0 + P \wedge \gamma_0 \gamma_0 = \mathbf{P}_E - \mathbf{I}\mathbf{P}_M$$

and we also define the field

$$(25) \quad G = (\mathbf{E} + \mathbf{P}_E) + \mathbf{I}(\mathbf{B} - \mathbf{P}_M) = \mathbf{D} + \mathbf{I}\mathbf{H}$$

With linearly polarizable media  $m$  having permittivity  $\epsilon_m$ , permeability  $\mu_m$ , and phase velocity  $v_m = 1/\sqrt{\epsilon_m\mu_m}$ , the total charged current  $J = J_f + J_b$  is equal to the sum of free charged current  $J_f$  and bound charged current  $J_b$  induced in the polarizable media. We refer to the quantities above as "physical" quantities  $Q$ , and define "media" quantities  $Q_m$  with relative 3D vectors and scalars related to those of the physical quantities  $Q$  by

$$(26) \quad J_m = (\rho_m + \mathbf{J}_m)\gamma_0 \equiv (\rho_f + \frac{1}{v_m}\mathbf{J}_f)\gamma_0$$

$$(27) \quad A_m = (\phi_m + \mathbf{A}_m)\gamma_0 \equiv (\epsilon_m\phi + \frac{1}{v_m}\frac{1}{\mu_m}\mathbf{A})\gamma_0$$

$$(28) \quad \nabla_m = (\partial_m - \nabla_m)\gamma_0 \equiv (\frac{1}{v_m}\partial_0 - \nabla)\gamma_0$$

$$(29) \quad G_m \equiv \nabla_m \wedge A_m = \epsilon_m\mathbf{E} + \mathbf{I}\frac{1}{v_m}\frac{1}{\mu_m}\mathbf{B}$$

With these definitions, we can rewrite the Euler-Lagrange and Lorenz condition equations as

$$(30) \quad \nabla_m \cdot G_m - J_m = 0$$

$$(31) \quad n \cdot \Delta G_m - K_m = 0$$

$$(32) \quad \nabla_m \cdot A_m = 0$$

Equations that apply to electromagnetism in vacuum then also apply to electromagnetism in linearly polarizable media, with quantities  $Q$  replaced by  $Q_m$ . For example, Green's 3rd Identity can be applied in polarizable media with this replacement to find the media potential at any point in a linearly polarizable region enclosed by a boundary, and the resulting media potential transformed back to the physical potential.

The general method of using media quantities is that we refer to the physical potential  $A$  on boundaries, but for regions enclosed by boundaries we calculate the corresponding value of the media potential  $A_m$  on the boundary (which is generally different for the two sides of the boundary at each boundary point) and then use  $A_m$  on the boundary in Green's 3rd Identity to find the  $A_m$  at any point in the enclosed region. We may then use  $A_m$  at neighboring points to numerically calculate the derivative  $G_m$ , or alternatively, we may calculate  $G_m$  at every point on the boundary and used Green's 3rd Identity for the field to calculate  $G_m$  at any enclosed point.

For notational simplicity, we generally omit the subscript  $m$ , with its presence implied as necessary by context.

Numerical and graphical results will be presented for some examples of solutions using this method for bounded regions of uniform linear polarizability.

#### 4. NONLINEAR POLARIZABILITY

This section sketches more details of the method of section 2.2: some or all of space-time divided into simplices with the potential being a linear function of position within each simplex and the media possibly having nonlinear polarizability.

In this case, unlike the situation of section 2.1 for which we assume that all quantities vary harmonically with time and so choose the boundary to be tangential to the world line of the rest frame, we cover part of space-time with simplices that necessarily have faces that are not perpendicular to the world line of the rest frame of the medium. Nevertheless, for computation we still will choose the rest frame to define and evaluate scalar quantities to store in computer memory, so we would like to express equation (11) with the basis potentials  $a_i$ , the boundary

normals  $n$ , and the change in field  $\Delta G$ , in terms of quantities naturally defined and specified in the rest frame of the medium characterized by  $\gamma_0$ .

To do this we express these quantities in terms of relative scalars and vectors:

$$(33) \quad a_i = (\phi_i + \mathbf{a}_i) \gamma_0$$

$$(34) \quad n = (n_0 + \mathbf{n}) \gamma_0$$

$$(35) \quad \Delta G = \Delta \mathbf{D} + \mathbf{I} \Delta \mathbf{H}$$

$$(36) \quad K = (\boldsymbol{\sigma} + \mathbf{K}) \gamma_0$$

Then equation (11) becomes

$$(37) \quad \int_S ((\mathbf{a}_i n_0 - \phi_i \mathbf{n}) \cdot \Delta \mathbf{D} - \mathbf{a}_i \cdot (\mathbf{n} \times \Delta \mathbf{H})) d^3x = \int_S (\phi_i \boldsymbol{\sigma} - \mathbf{a}_i \cdot \mathbf{K}) d^3x$$

which is a set of equations indexed by  $i$ , to be solved for the unknown basis function coefficients  $c_i$  that define  $F = f_i c_i$  and therefore the function  $G = \mathbf{D} + \mathbf{I} \mathbf{H}$  of  $F$ . We choose each potential basis function  $a_i$  to be very localized, equal to some non-zero parameter at one simplex vertex, zero at every other simplex vertex, and linear within each simplex. This equation is then very local, and the system of equations may be solved by, for example, specifying boundary conditions over all of space at two adjacent times, and then choosing coefficients  $c_i$  to satisfy these equations, one set of coupled equations on a space-like set of points at a time.

## 5. COMPARISON WITH OTHER FINITE ELEMENT METHODS

Comparison of these methods with conventional methods used in computation electromagnetics is difficult because although the goals are similar, the theoretical starting points are very different. A few remarks are made here.

In particular, although discussion of variational methods and derivation of practical methods from them are common [4], the starting point of such treatments is generally Maxwell's 4 equations written using relative 3D vectors and scalars and 3D vector algebra. Maxwell's two equations containing time derivatives - one being part of the Euler-Lagrange equations, the other being the result of the mathematical identity  $\nabla \wedge F = 0$  - are then combined to give a 2nd order equation for either the electric field  $\mathbf{E}$  or magnetic field  $\mathbf{B}$ . This is then used to *construct* an integral whose variation - typically with respect to  $\mathbf{E}$  or  $\mathbf{B}$ , not  $\mathbf{A}$  - gives this desired second order differential equation. This method is oriented toward using general mathematical methods for solving differential equations rather than exploiting the fact that a Lagrangian and action already exist. It becomes quite complicated, and does not have advantages of an action formulation such as the existence of Noether's theorems and associated symmetries.

The method of least action described in this paper is both much simpler and more general, has the benefits of expression not tied to a Cartesian or any other coordinate system, better represents the actual structure of space and time which is so fundamental to electromagnetism, and exploits the exceptional fit between geometric algebra and electrodynamics.

## REFERENCES

- [1] David Hestenes, *Space-Time Algebra*, Gordon and Breach, New York, 1966.
- [2] C. Doran and A. Lasenby, *Geometric Algebra for Physicists*, Cambridge U Press, New York, 2003.
- [3] R. Harrington, *Field Computation by Moment Methods*, Macmillan, New York, 1968.
- [4] J. Jin, *The Finite Element Method in Electromagnetics*, Wiley, New York, 2002.
- [5] H. Prautzsch, W. Boehm, M. Paluszny, *Bezier and B-Spline Techniques*, Springer, New York, 2002.
- [6] T. Vold, *Computing Electromagnetic Fields by the Method of Least Action*, Advances in Applied Clifford Algebras, 2014, 10.1007/s00006-013-0417-1.

# A Clifford Bundle Approach to the Wave Equation of a Spin 1/2 Fermion in the de Sitter Manifold.

W. A. Rodrigues Jr.<sup>(1)</sup>, S. A. Wainer<sup>(1)</sup>, M. Rivera-Tapia<sup>(2)</sup>, E. A. Notte-Cuello<sup>(3)</sup> and I. Kondrashuk<sup>(4)</sup>

<sup>(1)</sup>Institute of Mathematics, Statistics and Scientific Computation

IMECC-UNICAMP

walrod@ime.unicamp.br samuelwainer@ime.unicamp.br

<sup>(2)</sup>Departamento de Física, Universidad de La Serena, La Serena-Chile.

marivera@userena.cl

<sup>(3)</sup>Departamento de Matemáticas, Universidad de La Serena, La Serena-Chile.

enotte@userena.cl

<sup>(4)</sup>Grupo de Matemática Aplicada, Departamento de Ciencias Básicas,  
Universidad del Bío-Bío, Campus Fernando May, Casilla 447, Chillán, Chile

igor.kondrashuk@ubiobio.cl

May 22 2015

## Abstract

In this paper we give a Clifford bundle motivated approach to the wave equation of a free spin 1/2 fermion in the de Sitter manifold, a brane with topology  $M = S0(4, 1)/S0(3, 1)$  living in the bulk spacetime  $\mathbb{R}^{4,1} = (\dot{M} = \mathbb{R}^5, \dot{\mathbf{g}})$  and equipped with a metric field  $\mathbf{g} := -\dot{\mathbf{i}}^* \dot{\mathbf{g}}$  with  $\dot{\mathbf{i}} : M \rightarrow \dot{M}$  being the inclusion map. To obtain the analog of Dirac equation in Minkowski spacetime in the structure  $\dot{M}$  we appropriately factorize the two Casimir invariants  $C_1$  and  $C_2$  of the Lie algebra of the de Sitter group using the constraint given in the linearization of  $C_2$  as input to linearize  $C_1$ . In this way we obtain an equation that we called **DHESS1**. Next we derive a wave equation (called **DHESS2**) for a free spin 1/2 fermion in the de Sitter manifold using a heuristic argument permitting a derivation of the Dirac equation in Minkowski spacetime and which shows that such famous equation express nothing more than the fact that the momentum of a free particle is a constant vector field over timelike integral curves of a given velocity field. It is a remarkable fact that **DHESS1** and **DHESS2** coincide. One of the main ingredients in our paper is the use of the concept of Dirac-Hestenes spinor fields.

**Keywords:** de Sitter Manifold, Clifford Bundle, Dirac Equation.

## 1 Introduction

The Dirac equation (**DE**) in a Minkowski spacetime can be obtained using Dirac's original procedure through a linearization of  $C_1^2 - m^2 = 0$  (where  $C_1^2$  is the first Casimir invariant of the enveloping algebra of the Poincaré group) and its application to covariant spinor fields (sections of  $P_{\text{Spin}_{1,3}^e} \times_{\mu} \mathbb{C}^4$ ). Using the Clifford and spin-Clifford bundles formalism<sup>1</sup> and an almost trivial heuristic argument we present a derivation of an equivalent equation to **DE** which is called the Dirac-Hestenes equation (**DHE**). Our derivation makes clear the fact that the **DE** (or the equivalent **DHE**) express nothing more than the fact that a free spin 1/2 particle moves with a constant velocity in Minkowski spacetime following an integral line of a well defined velocity

---

<sup>1</sup>This means the Clifford and spin-Clifford bundles formalism as developed in [14]. We use the notations of that book and the reader is invited to consult the book if he needs to improve his knowledge in order to be able to follow all calculations of the present article.



field. This observation is a crucial one for the main objective of this paper, the one of writing wave equations for a free spin 1/2 moving in a de Sitter manifold equipped with a metric field inherited from a bulk spacetime  $\mathbb{R}^{4,1}$  (see Section 2). It is intuitive (given the topology of the de Sitter manifold) that such a motion must happen with a constant angular momentum<sup>2</sup> and as we will see a heuristic deduction of a Dirac-Hestenes like equation in this case results identical from the one which we get if we linearize  $C_1 - m^2 = 0$  (where  $C_1$  is the first Casimir invariant of the enveloping algebra of the Lie algebra of the de Sitter group) taking into account a constraint coming from the linearization of  $C_2$ , the second Casimir invariant of the enveloping algebra of the Lie algebra of the de Sitter group

To be more precise, in Sections 3 and 4 we will present two Dirac-Hestenes like equations for a spin 1/2 fermion field<sup>3</sup>  $\phi$  living in de Sitter manifold equipped with a metric field  $\mathbf{g}$  (see Section 2), which will be abbreviate as **DHESS1** and **DHESS2**. The **DHESS1** will be obtained by linearizing the first Casimir operator  $C_1$  using a constraint imposed on the **DHSF** arising from the linearization of  $C_2$ . On the other hand **DHESS2** will be obtained by a physically and heuristically derivation resulting by simply imposing that the motion of a free particle in the de Sitter manifold is described by a constant angular momentum 2-form as seem by an hypothetical observer living in the bulk spacetime  $\mathbb{R}^{4,1}$ . Of course, as we are going to see the heuristic derivation is only possible using the Clifford bundle formalism. It is a remarkable result that **DHESS1** and **DHESS2** coincide and moreover translation of those equations in the covariant spinor field formalism gives a first order partial differential equation (which is equivalent to the one first postulated by Dirac [4]). It will be shown that **DHESS1** (and thus **DHESS2**) reduces to the Dirac-Hestenes equation (**DHE**) in Minkowski spacetime when  $\ell \rightarrow \infty$ , where  $\ell$  is the radius of the de Sitter manifold.

In Section 5 we study the limit of **DHESS1** and **DHESS2** when  $\ell \rightarrow \infty$  ( $\ell$  being the radius of the de Sitter manifold) showing that it gives the Dirac-Hestenes equation in Minkowski spacetime.

In Section 6 we present our conclusions.

## 2 The Lorentzian de Sitter $M^{dSL}$ Structure and its (Projective) Conformal Representation

Let  $SO(4,1)$  and  $SO(3,1)$  be respectively the special pseudo-orthogonal groups in vector manifolds  $\mathbb{R}^{4,1} = \{M \simeq \mathbb{R}^5, \hat{\mathbf{g}}\}$  and in  $\mathbb{R}^{3,1} = \{\mathbb{R}^4, -\boldsymbol{\eta}\}$  where  $\hat{\mathbf{g}}$  is a metric of signature (4,1) and  $\boldsymbol{\eta}$  a metric of signature (1,3). The manifold  $M = SO(4,1)/SO(3,1)$  will be called the *de Sitter manifold*. Since

$$M = SO(4,1)/SO(3,1) \approx SO(1,4)/SO(1,3) \approx \mathbb{R} \times S^3 \quad (1)$$

this manifold can be viewed as a brane [10] (a submanifold) in the structure  $\mathbb{R}^{4,1}$ . In General Relativity studies it is introduced a Lorentzian spacetime, i.e., the structure  $M^{dSL} = (M = \mathbb{R} \times S^3, \mathbf{g}, \mathbf{D}, \tau_{\mathbf{g}}, \uparrow)$  which will be called *Lorentzian de Sitter spacetime structure*<sup>4</sup> where if  $\iota : \mathbb{R} \times S^3 \rightarrow \mathbb{R}^5$  is the inclusion mapping,  $\mathbf{g} := -\iota^* \hat{\mathbf{g}}$  and  $\mathbf{D}$  is the parallel projection on  $M$  of the pseudo Euclidian metric compatible connection in  $\mathbb{R}^{4,1}$  (details in [?]). As well known,  $M^{dSL}$  is a spacetime of constant Riemannian curvature. It has ten Killing vector fields. The Killing vector fields are the generators of infinitesimal actions of the group  $SO(4,1)$  (called the de Sitter group) in  $M = \mathbb{R} \times S^3 \approx SO(4,1)/SO(3,1)$ . The group  $SO(4,1)$  acts transitively<sup>5</sup> in  $SO(4,1)/SO(3,1)$ , which is thus a homogeneous space (for  $SO(4,1)$ ).

<sup>2</sup>On this respect see also section XIV.3 of [1].

<sup>3</sup>Called in what follows a Dirac-Hestnes spinor field and denoted **DHSF**.

<sup>4</sup>It is a vacuum solution of Einstein equation with a cosmological constant term. We are not going to use this structure in this paper.

<sup>5</sup>A group  $G$  of transformations in a manifold  $M$  ( $\sigma : G \times M \rightarrow M$  by  $(g, x) \mapsto \sigma(g, x)$ ) is said to act transitively on  $M$  if for arbitraries  $x, y \in M$  there exists  $g \in G$  such that  $\sigma(g, x) = y$ .

We now give a description of the manifold  $\mathbb{R} \times S^3$  as a pseudo-sphere (a submanifold) of radius  $l$  of the pseudo Euclidean space  $\mathbb{R}^{4,1} = \{\mathbb{R}^5, \mathring{\mathbf{g}}\}$ . If  $(X^1, X^2, X^3, X^4, X^0)$  are the global orthogonal coordinates of  $\mathbb{R}^{4,1}$ , then the equation representing the pseudo sphere is

$$(X^1)^2 + (X^2)^2 + (X^3)^2 + (X^4)^2 - (X^0)^2 = \ell^2. \quad (2)$$

Introducing projective *conformal* coordinates  $\{x^\mu\}$  by projecting the points of  $\mathbb{R} \times S^3$  from the “north-pole” to a plane tangent to the “south pole” we see immediately that  $\{x^\mu\}$  covers all  $\mathbb{R} \times S^3$  except the “north-pole”. We have

$$X^\mu = \Omega x^\mu, \quad X^4 = -\ell \Omega \left(1 + \frac{\sigma^2}{4\ell^2}\right) \quad (3)$$

$$\mathbf{g} := -\iota^* \mathring{\mathbf{g}} = \Omega^2 \eta_{\mu\nu} dx^\mu \otimes dx^\nu, \quad (4)$$

$$\Omega := \left(1 - \frac{\sigma^2}{4\ell^2}\right)^{-1}, \quad \sigma^2 = \eta_{\mu\nu} x^\mu x^\nu. \quad (5)$$

and the matrix with entries  $\eta_{\mu\nu}$  is the diagonal matrix  $\text{diag}(1, -1, -1, -1)$ .

### 3 Linearization of the Casimir Invariants of the $\text{spin}_{4,1}^e$ Lie algebra

The classical angular momentum biform of a free particle following a “timelike” curve  $\sigma$  with momentum 1-form  $\mathbf{p}$  in  $\mathring{M} = \mathbb{R}^{4,1}$  is

$$\mathbf{l} = \mathbf{x} \wedge \mathbf{p}, \quad (6)$$

$$\mathbf{x} := X^A \mathring{\mathbf{E}}_A, \quad \mathbf{p} := P_A \mathring{\mathbf{E}}^A \quad (7)$$

are respectively the position 1-form and the momentum of the free particle. Moreover,  $\{\mathring{\mathbf{E}}^A = dX^A\}$  is an orthonormal cobasis of  $T^*\mathring{M}$  dual to the orthonormal basis  $\{\mathring{\mathbf{e}}_A = \frac{\partial}{\partial X^A}\}$  of  $T\mathring{M}$ . and  $\{\mathring{\mathbf{E}}_A\}$  is an orthonormal cobasis of  $T^*\mathring{M}$ , called the reciprocal basis of  $\{\mathring{\mathbf{E}}^A\}$  and it is  $\mathring{\mathbf{g}}(\mathring{\mathbf{E}}^A, \mathring{\mathbf{E}}_B) = \delta_B^A$  where<sup>6</sup>

$$\mathring{\mathbf{g}} = \eta^{AB} \mathring{\mathbf{e}}_A \otimes \mathring{\mathbf{e}}_B \quad (8)$$

is the metric for  $T^*\mathring{M}$ . If

$$\mathring{\mathbf{g}} = \eta_{AB} \mathring{\mathbf{E}}^A \otimes \mathring{\mathbf{E}}^B \quad (9)$$

is the metric of  $T\mathring{M}$ , it is  $\eta^{AC} \eta_{CB} = \delta_B^A$ . We have

$$\mathbf{l} = \frac{1}{2} L_{AB} \mathring{\mathbf{E}}^A \wedge \mathring{\mathbf{E}}^B = \frac{1}{2} L_{AB} \mathring{\mathbf{E}}^A \mathring{\mathbf{E}}^B \quad (10)$$

$$L_{AB} = \eta_{AC} X^C P_B - \eta_{BC} X^C P_A \quad (11)$$

**Remark 1** *It is quite obvious that for a classical particle living in de Sitter spacetime and following a timelike worldline  $\sigma$  parametrized by proptime  $\tau$  if we write*

$$\mathbf{x} = X^A(\tau) \mathring{\mathbf{E}}_A, \quad \mathbf{p} = m \frac{dX^A(\tau)}{d\tau} \mathring{\mathbf{E}}_A \quad (12)$$

it is (since  $\mathbf{x} \cdot \mathbf{x} = \mathring{\mathbf{g}}(\mathbf{x}, \mathbf{x}) = \ell^2$ )

$$\mathbf{x} \cdot \mathbf{p} = 0. \quad (13)$$

Thus,

$$\mathbf{x}\mathbf{p} = \mathbf{x} \wedge \mathbf{p} \quad (14)$$

<sup>6</sup>The matrix with entries  $\eta_{AB}$  is the diagonal matrix  $\text{diag}(1, 1, 1, 1, -1)$

and as a consequence

$$\mathbf{l}^2 = \mathbf{xpxp} = -\mathbf{pxxp} = -\ell^2 \mathbf{p}^2 \quad (15)$$

which implies that

$$\mathbf{l} \wedge \mathbf{l} = 0 \quad (16)$$

$$\mathbf{l}^2 = \mathbf{L} \lrcorner \mathbf{L}. \quad (17)$$

As we are going to see the classical condition given by Eq.(16) cannot be assumed in quantum theory where the classical angular momentum is substituted by a quantum angular momentum operator.

So, to continue we define  $\mathring{\mathcal{H}}$ , the Hilbert space of a one quantum spin 1/2 particle living  $\mathring{M}$  as the set of all square integrable mappings<sup>7</sup>

$$\phi \in \text{sec } \mathcal{C}\ell^0(\mathring{M}, \mathring{\mathfrak{g}}) \quad (18)$$

called representatives in  $\mathcal{C}\ell^0(\mathring{M}, \mathring{\mathfrak{g}})$  relative to a spin coframe of (DHSF) [7].

The quantum angular momentum operator  $\mathbf{L} \in \mathcal{L}(\mathring{\mathcal{H}})$  is

$$\mathbf{L} := \frac{1}{2} \mathring{\mathbf{E}}^A \mathring{\mathbf{E}}^B \mathbf{L}_{AB} \quad (19)$$

$$\mathbf{L}_{AB} = \eta_{AC} X^C \mathbf{P}_B - \eta_{BC} X^C \mathbf{P}_A \quad (20)$$

with  $\mathbf{P}_A \in \mathcal{L}(\mathring{\mathcal{H}})$  defined by

$$\mathbf{P}_A \phi := \partial_A \phi \mathring{\mathbf{E}}^2 \mathring{\mathbf{E}}^1 \quad (21)$$

Now<sup>8</sup>,

$$\mathbf{L}^2 = \mathbf{L} \lrcorner \mathbf{L} + \mathbf{L} \wedge \mathbf{L}. \quad (22)$$

$\mathbf{L} \lrcorner \mathbf{L}$  is clearly a scalar invariant under the action of  $\text{Spin}_{4,1}^e$  group and it is:

$$\mathbf{L} \lrcorner \mathbf{L} = -\frac{1}{2} L_{AB} L^{AB}. \quad (23)$$

The first Casimir operator of the Lie algebra  $\text{spin}_{4,1}^e$  is defined by

$$C_1 = \frac{1}{\ell^2} \mathbf{L} \lrcorner \mathbf{L} = -\frac{1}{\ell^2} \mathbf{L} \cdot \mathbf{L} = -\frac{1}{2\ell^2} L_{AB} L^{AB} = m^2, \quad (24)$$

with  $m^2 \in \mathbb{R}$ . We have the

**Proposition 2** *Call*

$$\mathbf{W} := \star \frac{1}{8\ell} (\mathbf{L} \wedge \mathbf{L}) = \frac{1}{8\ell} (\mathbf{L} \wedge \mathbf{L}) \lrcorner \tau. \quad (25)$$

Then,

$$\begin{aligned} \mathbf{W} \cdot \mathbf{W} &= \mathbf{W} \mathbf{W} = -\frac{1}{64\ell^2} (\mathbf{L} \wedge \mathbf{L}) \lrcorner (\mathbf{L} \wedge \mathbf{L}) = -\frac{1}{64\ell^2} (\mathbf{L} \wedge \mathbf{L}) \cdot (\mathbf{L} \wedge \mathbf{L}) \\ &= -\frac{1}{64\ell^2} (\mathbf{L} \wedge \mathbf{L})(\mathbf{L} \wedge \mathbf{L}). \end{aligned} \quad (26)$$

**Remark 3** *The second Casimir invariant of  $\text{spin}_{4,1}^e$  is defined by*

$$C_2 = \mathbf{W} \cdot \mathbf{W} = -\frac{1}{64\ell^2} (\mathbf{L} \wedge \mathbf{L})(\mathbf{L} \wedge \mathbf{L}) = -m^2 s(s+1), \quad (27)$$

where  $s = 0, 1/2, 1, 3/2, \dots$ . It is thus quite obvious that contrary to the classical case the operator  $\mathbf{L} \wedge \mathbf{L}$  cannot be null for otherwise from Eq.(27) it would be necessary that  $m = 0$  or  $s = 0$ .

<sup>7</sup>By square integrable we mean that  $\int \psi \cdot \psi \tau_{\mathring{\mathfrak{g}}} = 1$ .

<sup>8</sup>The action of  $\mathbf{L}^2$  on  $\phi$  (in analogy to the square of the Dirac operator) is defined by

$\mathbf{L}^2 \phi := \mathbf{L}(\mathbf{L}\phi) = (\mathbf{L} \lrcorner \mathbf{L})\phi + (\mathbf{L} \wedge \mathbf{L})\phi$ .

Observe that the spin 1/2 wave function needs to satisfy the fourth order equation

$$\left(\frac{1}{64\ell^2}(\mathbf{L} \wedge \mathbf{L})(\mathbf{L} \wedge \mathbf{L}) - m^2 s(s+1)\right) \phi = 0. \quad (28)$$

We can factorize the invariant  $C_2$  as

$$\left(\frac{1}{8\ell}\mathbf{L} \wedge \mathbf{L} + m\sqrt{s(s+1)}\right) \left(\frac{1}{8\ell}\mathbf{L} \wedge \mathbf{L} - m\sqrt{s(s+1)}\right) = 0. \quad (29)$$

Then, a possible second order equation that we will impose to be satisfied by  $\phi$  is<sup>9</sup>

$$(\mathbf{L} \wedge \mathbf{L} - 4\sqrt{3}m\ell) \phi = 0. \quad (30)$$

To continue observe that we cannot factorize  $\mathbf{L} \lrcorner \mathbf{L} - \ell^2 m^2 = 0$  in two first order operators. However taking into account Eq.(22) we can write

$$\frac{1}{\ell^2}\mathbf{L}^2 - m^2 - \frac{1}{\ell^2}\mathbf{L} \wedge \mathbf{L} = 0. \quad (31)$$

Thus, taking into account Eq.(30) we can write

$$\left(\frac{1}{\ell^2}\mathbf{L}^2 - m^2 - \frac{4\sqrt{3}m}{\ell}\right) \phi = 0 \quad (32)$$

$$\lambda^2 := m^2 + \frac{4\sqrt{3}m}{\ell} \quad (33)$$

we can now factor Eq.(32) as

$$\left(\frac{1}{\ell^2}\mathbf{L}^2 - \lambda^2\right) \phi = \left(\frac{1}{\ell}\mathbf{L} + \lambda\right) \left(\frac{1}{\ell}\mathbf{L} - \lambda\right) \phi = 0. \quad (34)$$

### 3.1 The Tangency Condition and the DHESS1

Let  $\{\hat{\theta}^0, \hat{\theta}^1, \hat{\theta}^2, \hat{\theta}^3, \hat{\theta}^4\}$  be an orthonormal basis for  $\wedge^1 T^* \mathring{M}$  such that  $\{\theta^\mu = \hat{\theta}^\mu, \mu = 0, 1, 2, 3\}$  is a tangent cotetrad basis for de Sitter spacetime, i.e.,  $\theta^\mu \in \sec \wedge^1 T^* M \hookrightarrow \sec \mathcal{C}\ell(M, \mathbf{g}) \subset \sec \mathcal{C}\ell(\mathring{M}, \mathring{\mathbf{g}})$  with  $\hat{\theta}^4$  orthogonal to  $M$ , i.e.,  $\hat{\theta}^4 \lrcorner \tau_{\mathbf{g}} = 0$ .

We now propose taking into account Eq.(34) that the electron wave function in de Sitter spacetime must satisfy the linear equation

$$\left(\frac{1}{\ell}\mathbf{L} - \lambda\right) \phi = 0, \quad (35)$$

with the constrain that  $\phi$  is tangent to  $M$ , i.e., it does not contain in its expansion terms containing  $\hat{\theta}^A \hat{\theta}^4$ .

Eq.(35) will be called the Dirac-Hestenes equation in de Sitter spacetime (**DHESS1**).

## 4 An Heuristic Derivation of the DHESS2

We start recalling that a classical free particle in de Sitter manifold structure  $(M, \mathbf{g})$  certainly follows a timelike worldline  $\sigma : \mathbb{R} \supset I \rightarrow M$ . To unveil the nature of that motion we suppose the existence of a 2-form field  $\mathbf{L} \in \sec \wedge^2 T^* \mathring{M} \hookrightarrow \sec \mathcal{C}\ell(\mathring{M}, \mathring{\mathbf{g}})$  such that its restriction over  $\sigma$  is  $\mathbf{l}$  i.e.,  $\mathbf{L}|_\sigma = \mathbf{l}$ . given by Eq.(10).

---

<sup>9</sup>We used that  $s = 1/2$ .

It is a very reasonable hypothesis that the classical motion of a free particle in de Sitter manifold structure  $(M, \mathbf{g})$  will happen only under the condition that  $\mathbf{L} \in \sec \bigwedge^2 T^* \overset{\circ}{M} \hookrightarrow \sec \mathcal{C}\ell(\overset{\circ}{M}, \overset{\circ}{\mathbf{g}})$  is a constant 2-form  $\mathbf{B}$  as registered by an hypothetical ‘‘observer’’ living in the bulk space-time  $\mathbb{R}^{4,1}$ . This is an obvious generalization of the fact that the free motion of a classical particle in the bulk  $\mathbb{R}^{4,1}$  happens with constant momentum. Let  $\{\overset{\circ}{\theta}^A, A = 0, 1, 2, 3, 4\}$  be an orthonormal basis for  $\bigwedge^1 T^* \overset{\circ}{M}$  such that  $\{\theta^\mu = \overset{\circ}{\theta}^\mu, \mu = 0, 1, 2, 3\}$  is a cotetrad basis for  $\bigwedge^1 T^* M \hookrightarrow \mathcal{C}\ell(M, \mathbf{g}) \subset \mathcal{C}\ell(\overset{\circ}{M}, \overset{\circ}{\mathbf{g}})$  with  $\overset{\circ}{\theta}^4$  orthogonal to  $M$ , i.e.,  $\overset{\circ}{\theta}^4 \lrcorner \tau_{\mathbf{g}} = 0$ . Then, the condition that  $\mathbf{L}$  is a constant 2-form may be written

$$\frac{1}{\ell} \mathbf{L} = \sigma \mathbf{B}. \quad (36)$$

Now, let  $\phi'$  be an invertible **DHSF** (i.e.,  $\phi' \tilde{\phi}' \neq 0$ ) such that

$$\mathbf{B} = \phi'^{-1} \overset{\circ}{\theta}^1 \overset{\circ}{\theta}^2 \phi' = \overset{\circ}{\mathbf{E}}^1 \overset{\circ}{\mathbf{E}}^2. \quad (37)$$

Before continuing we also suppose that

$$\phi'^{-1} \overset{\circ}{\theta}^A \overset{\circ}{\theta}^B \phi' = \overset{\circ}{\mathbf{E}}^A \overset{\circ}{\mathbf{E}}^B, \quad A, B = 0, 1, 2, 3, 4. \quad (38)$$

Using Eq.(37) in Eq.(36) we can write a purely classic **DHESS** equation, namely,

$$\frac{1}{\ell} \mathbf{L} \phi'^{-1} \overset{\circ}{\theta}^2 \overset{\circ}{\theta}^1 = \sigma \phi'^{-1}. \quad (39)$$

This is compatible with a quantum **DHESS1**, i.e.,

$$\frac{1}{\ell} \mathbf{L} \phi'^{-1} - \sigma \phi'^{-1} = 0 \quad (40)$$

with the *postulate* that when we restrict our considerations to **DHSFs** living in the de Sitter structure  $(M, \mathbf{g})$  it is:

$$\frac{1}{\ell} \mathbf{L} \phi'^{-1} \overset{\circ}{\theta}^2 \overset{\circ}{\theta}^1 \mapsto \frac{1}{\ell} \mathbf{L} \phi'^{-1} := \overset{\circ}{\theta}^A \overset{\circ}{\theta}^B (X_A \mathbf{P}_B - X_B \mathbf{P}_A) \phi'^{-1} \quad (41)$$

$$\mathbf{P}_A \phi'^{-1} \mapsto \mathbf{P}_A \phi'^{-1} := \frac{\partial}{\partial X^A} \phi'^{-1} \overset{\circ}{\theta}^2 \overset{\circ}{\theta}^1. \quad (42)$$

Note that under the above conditions the **DHESS1** equation (Eq.(35)) will be identical to the **DHESS2** (Eq.(40)) if

$$\lambda \mapsto \sigma, \quad \phi'^{-1} \mapsto \phi. \quad (43)$$

## 5 The limit $\ell \rightarrow \infty$ of Eq.(40)

Expressing  $\mathbf{L}_{AB}$  in terms of the projective coordinates we get

$$\mathbf{L}_{\alpha 4} \phi = l \partial_\alpha \phi \overset{\circ}{\theta}^2 \overset{\circ}{\theta}^1 - \frac{1}{4l} (2\eta_{\alpha\lambda} x^\lambda x^\nu - \sigma^2 \delta_\alpha^\nu) \partial_\nu \phi \overset{\circ}{\theta}^2 \overset{\circ}{\theta}^1. \quad (44)$$

$$\mathbf{L}_{\mu\nu} \phi = -\eta_{\mu\lambda} x^\lambda \partial_\nu \phi \overset{\circ}{\theta}^2 \overset{\circ}{\theta}^1 + \eta_{\nu\lambda} x^\lambda \partial_\mu \phi \overset{\circ}{\theta}^2 \overset{\circ}{\theta}^1 \quad (45)$$

We put

$$\phi := \left( \varphi + \frac{1}{\ell} \Omega^2 \overset{\circ}{\theta}^4 \overset{\circ}{\theta}^{\alpha} x_{\alpha\rho} \right) \in \sec \mathcal{C}\ell^0(\overset{\circ}{M}, \overset{\circ}{\mathbf{g}}). \quad (46)$$

with the condition that  $\varphi, \rho \in \sec \mathcal{C}\ell^0(\mathring{M}, \mathring{\mathfrak{g}})$  ( $\varphi$  living in a particular ideal<sup>10</sup> of  $\mathcal{C}\ell^0(\mathring{M}, \mathring{\mathfrak{g}})$  and tangent to the de Sitter manifold). Then Eq.(41) can be written (in the limit  $\ell \rightarrow \infty$ ) as

$$\begin{aligned} & \mathring{\theta}^\alpha \mathring{\theta}^4 \frac{\partial}{\partial x^\alpha} \phi \mathring{\theta}^2 \mathring{\theta}^1 - \frac{1}{4\ell^2} \mathring{\theta}^\alpha \mathring{\theta}^4 (2\eta_{\alpha\lambda} x^\lambda x^\nu - \sigma^2 \delta_\alpha^\nu) \frac{\partial}{\partial x^\nu} \phi \mathring{\theta}^2 \mathring{\theta}^1 \\ & - \frac{1}{\ell} \mathring{\theta}^\mu \mathring{\theta}^\nu \left( \eta_{\mu\lambda} x^\lambda \frac{\partial}{\partial x^\nu} - \eta_{\nu\lambda} x^\lambda \frac{\partial}{\partial x^\mu} \right) \phi \mathring{\theta}^2 \mathring{\theta}^1 - \lambda \phi = 0. \end{aligned} \quad (47)$$

Now, calling  $\mathring{\theta}^\alpha \mathring{\theta}^4 = \Gamma^\alpha$ ,  $\alpha = 0, 1, 2, 3$  we easily verify that

$$\Gamma^\alpha \Gamma^\beta + \Gamma^\beta \Gamma^\alpha = 2\eta^{\alpha\beta}. \quad (48)$$

Finally, recalling Eq.(46) we can write Eq.(47) in the limit  $\ell \rightarrow \infty$  as

$$\Gamma^\alpha \frac{\partial}{\partial x^\alpha} \varphi \Gamma^2 \Gamma^1 - m\varphi = 0. \quad (49)$$

which is clearly a *representative* of the **DHE** in Minkowski spacetime in the  $\mathcal{C}\ell(M \simeq \mathbb{R}^4, \boldsymbol{\eta})$  bundle which reads<sup>11</sup>

$$\gamma^\alpha \frac{\partial}{\partial x^\alpha} \psi \gamma^2 \gamma^1 - m\psi \gamma^0 = 0. \quad (50)$$

Indeed, multiplying Eq.(50) on the right by the idempotent  $\frac{1}{2}(1 + \gamma^0)$  it reads (calling  $\zeta = \psi \frac{1}{2}(1 + \gamma^0)$ )

$$\gamma^\alpha \frac{\partial}{\partial x^\alpha} \zeta \gamma^2 \gamma^1 - m\zeta = 0. \quad (51)$$

So, Eqs. (49) and (50) can be identified with the identifications

$$\Gamma^\alpha \leftrightarrow \gamma^\alpha, \quad \varphi \leftrightarrow \zeta \quad (52)$$

**Remark 4** *It is well known that when  $\ell \rightarrow \infty$  the **DHESS1** (Eq.(35)) which is a Clifford bundle representation of the Dirac equation (written with the standard matrix formalism) is also equivalent to the **DHE** in Minkowski spacetime [6].*

## 6 Conclusions

We gave a Clifford bundle motivated approach to the wave equation of a free spin 1/2 fermion in the de Sitter manifold, a brane with topology  $M = \text{S0}(4, 1)/\text{S0}(3, 1)$  living in the bulk spacetime  $\mathring{M} = \mathbb{R}^{4,1} = (\mathbb{R}^5, \mathring{\mathfrak{g}})$  and equipped with a metric field  $\mathfrak{g} := -i^* \mathring{\mathfrak{g}}$  with  $i : M \rightarrow \mathring{M}$  being the inclusion map. To obtain the analog of Dirac equation in Minkowski spacetime we appropriately factorize the two Casimir invariants  $C_1$  and  $C_2$  of the Lie algebra of the de Sitter group using the constraint given in the linearization of  $C_2$  as input to linearize  $C_1$ . In this way we obtain an equation that we called **DHESS1** (which is simply postulated in previous studies ([4, 6])). Next we derive a wave equation (called **DHESS2**) for a free spin 1/2 fermion in the de Sitter manifold using an heuristic argument which is an obvious generalization of an heuristic argument permitting a derivation of the Dirac equation in Minkowski spacetime which shows that such famous equation express nothing more that the momentum of a free particle is a constant vector field over timelike integral curves of a given velocity field. It is a nice fact that **DHESS1** and **DHESS2** coincide. We emphasize moreover that our approach leaves clear the *nature and meaning of the Casimir invariants* [2] and thus of the object  $\lambda$  (Eq.(33)), something that is not clear in other papers on the subject such as, e.g., [4, 6, 11, 12] which use the standard covariant Dirac spinor fields.

<sup>10</sup>I.e., we must have  $\varphi = \varphi \frac{1}{2}(1 + \mathring{\theta}^0) = \varphi \mathring{\theta}^0$ .

<sup>11</sup>In Eq.(50)  $\{x^\mu\}$  are coordinates in Einstein-Lorentz-Poincaré gauge, and  $\gamma^\mu := dx^\mu$ .

As a last comment here we recall that if the de Sitter manifold is supposed to be a spacetime, i.e., a structure[14]  $(M, \mathbf{g}, \tau_{\mathbf{g}}, \nabla, \uparrow)$  where  $\nabla$  is an arbitrary connection compatible with  $\mathbf{g}$  then the writing of Dirac equation in such a structure is supposed to be given by very different arguments from the ones used in this paper.

**Acknowledgement 5** *M. Rivera-Tapia and E. A. Notte-Cuello were supported by the Direccion de Investigación de la Universidad de La Serena DIULS. The work of I. Kondrashuk was supported by Fondecyt (Chile) Grants Nos. 1040368, 1050512, and 1121030, and by DIUBB (Chile) Grants Nos. 102609 and 121909 GI/C-UBB.*

## A Description of the Dirac Equation in the Clifford Bundle

To fix the notation let  $(M \simeq \mathbb{R}^4, \eta, D, \tau_{\eta})$  be the Minkowski spacetime structure where  $\eta \in \text{sec } T_0^2 M$  is Minkowski metric and  $D$  is the Levi-Civita connection of  $\eta$ . Also,  $\tau_{\eta} \in \text{sec } \wedge^4 T^* M$  defines an orientation. We denote by  $\eta \in \text{sec } T_2^0 M$  the metric of the cotangent bundle. It is defined as follows. Let  $\{x^\mu\}$  be coordinates for  $M$  in the Einstein-Lorentz-Poincaré gauge [14]. Let  $\{e_\mu = \partial/\partial x^\mu\}$  a basis for  $TM$  and  $\{\gamma^\mu = dx^\mu\}$  the corresponding dual basis for  $T^*M$ , i.e.,  $\gamma^\mu(e_\alpha) = \delta_\alpha^\mu$ . Then, if  $\eta = \eta_{\mu\nu} \gamma^\mu \otimes \gamma^\nu$  then  $\eta = \eta^{\mu\nu} e_\mu \otimes e_\nu$ , where the matrix with entries  $\eta_{\mu\nu}$  and the one with entries  $\eta^{\mu\nu}$  are the equal to the diagonal matrix  $\text{diag}(1, -1, -1, -1)$ . If  $a, b \in \text{sec } \wedge^1 T^* M$  we write  $a \cdot b = \eta(a, b)$ . We also denote by  $\langle \gamma_\mu \rangle$  the reciprocal basis of  $\{\gamma^\mu = dx^\mu\}$ , which satisfies  $\gamma^\mu \cdot \gamma_\nu = \delta_\nu^\mu$ .

We denote the Clifford bundle of differential forms<sup>12</sup> in Minkowski spacetime by  $\mathcal{C}\ell(M, \eta)$  and use notations and conventions in what follows as in [14] and recall the fundamental relation

$$\gamma^\mu \gamma^\nu + \gamma^\nu \gamma^\mu = 2\eta^{\mu\nu}. \quad (53)$$

If  $\{\gamma^\mu, \mu = 0, 1, 2, 3\}$  are the Dirac gamma matrices in the *standard representation* and  $\{\gamma_\mu, \mu = 0, 1, 2, 3\}$  are as introduced above, we define

$$\sigma_k := \gamma_k \gamma_0 \in \text{sec } \wedge^2 T^* M \hookrightarrow \text{sec } \mathcal{C}\ell^0(M, \eta), \quad k = 1, 2, 3, \quad (54)$$

$$\mathbf{i} = \gamma_5 := \gamma_0 \gamma_1 \gamma_2 \gamma_3 \in \text{sec } \wedge^4 T^* M \hookrightarrow \text{sec } \mathcal{C}\ell(M, \eta), \quad (55)$$

$$\gamma_5 := \gamma_0 \gamma_1 \gamma_2 \gamma_3 \in \mathbb{C}(4) \quad (56)$$

Noting that  $M$  is parallelizable, in a given global spin frame a covariant spinor field can be taken as a mapping  $\psi : M \rightarrow \mathbb{C}^4$ . In standard representation of the gamma matrices where  $(i = \sqrt{-1}, \phi, \varsigma : M \rightarrow \mathbb{C}^2)$  to  $\psi$  given by

$$\psi = \begin{pmatrix} \phi \\ \varsigma \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} m^0 + im^3 \\ -m^2 + im^1 \end{pmatrix} \\ \begin{pmatrix} n^0 + in^3 \\ -n^2 + in^1 \end{pmatrix} \end{pmatrix}, \quad (57)$$

there corresponds the **DHSF**  $\psi \in \text{sec } \mathcal{C}\ell^0(M, \eta)$  given by<sup>13</sup>

$$\psi = \phi + \varsigma \sigma_3 = (m^0 + m^k \mathbf{i} \sigma_k) + (n^0 + n^k \mathbf{i} \sigma_k) \sigma_3. \quad (58)$$

<sup>12</sup>We recall that  $\mathcal{C}\ell(T_x^* M, \eta) \simeq \mathbb{R}_{1,3}$  the so-called spacetime algebra. Also the even subalgebra of  $\mathbb{R}_{1,3}$  denoted  $\mathbb{R}_{1,3}^0$  is isomorphic to the Pauli algebra  $\mathbb{R}_{3,0}$ , i.e.,  $\mathbb{R}_{1,3}^0 \simeq \mathbb{R}_{3,0}$ . The even subalgebra of the Pauli algebra  $\mathbb{R}_{3,0}^0 := \mathbb{R}_{3,0}^{00}$  is the quaternion algebra  $\mathbb{R}_{0,2}$ , i.e.,  $\mathbb{R}_{0,2} \simeq \mathbb{R}_{3,0}^0$ . Moreover we have the identifications:  $\text{Spin}_{1,3}^0 \simeq \text{SI}(2, \mathbb{C})$ ,  $\text{Spin}_{3,0} \simeq \text{SU}(2)$ . For the Lie algebras of these groups we have  $\text{spin}_{1,3}^0 \simeq \text{sl}(2, \mathbb{C})$ ,  $\text{su}(2) \simeq \text{spin}_{3,0}$ . The important fact to keep in mind for the understanding of some of the identifications we done below is that  $\text{Spin}_{1,3}^0, \text{spin}_{1,3}^0 \subset \mathbb{R}_{3,0} \subset \mathbb{R}_{1,3}$  and  $\text{Spin}_{3,0}, \text{spin}_{3,0} \subset \mathbb{R}_{0,2} \subset \mathbb{R}_{1,3}^0 \subset \mathbb{R}_{1,3}$ .

<sup>13</sup>Remember the identification:

$$\mathbb{C}(4) \simeq \mathbb{R}_{4,1} \supseteq \mathbb{R}_{4,1}^0 \simeq \mathbb{R}_{1,3}.$$

We then have the useful formulas in Eq.(59) below that one can use to immediately translate results of the standard matrix formalism in the language of the Clifford bundle formalism and vice-versa<sup>14</sup>

$$\begin{aligned}
\gamma_\mu \psi &\leftrightarrow \gamma_\mu \psi \gamma_0, \\
i\psi &\leftrightarrow \psi \gamma_{21} = \psi \mathbf{i} \sigma_3, \\
i\gamma_5 \psi &\leftrightarrow \psi \sigma_3 = \psi \gamma_3 \gamma_0, \\
\bar{\psi} &= \psi^\dagger \gamma^0 \leftrightarrow \tilde{\psi}, \\
\psi^\dagger &\leftrightarrow \gamma_0 \tilde{\psi} \gamma_0, \\
\psi^* &\leftrightarrow -\gamma_2 \psi \gamma_2.
\end{aligned} \tag{59}$$

Using the above dictionary the standard Dirac equation<sup>15</sup> for a Dirac spinor field  $\psi : M \rightarrow \mathbb{C}^4$

$$i\gamma^\mu \partial_\mu \psi - m\psi = 0 \tag{60}$$

translates immediately in the so-called Dirac-Hestenes equation, i.e.,

$$\partial \psi \gamma_{21} - m\psi \gamma_0 = 0. \tag{61}$$

## B Heuristic Derivation of the DHE in Minkowski Spacetime

We start recalling that a classical spin 1/2 free particle is supposed to have its story described by a geodesic timelike worldline  $\sigma : \mathbb{R} \supset I \rightarrow M$  in the Minkowski spacetime structure. Let  $\sigma_*$  be the velocity of the particle and let  $\mathbf{v} = \mathbf{g}(\sigma_*, \cdot)$  be the *physically equivalent* 1-form. We that  $\mathbf{v} \in \sec T_\sigma^* M \hookrightarrow \sec \mathcal{C}\ell(M, \mathbf{g})$ . Its classical momentum 1-form is

$$\mathbf{p} = m\mathbf{v}. \tag{62}$$

To continue, we suppose the existence of a 1-form field  $V \in \sec \wedge^1 T^* M \hookrightarrow \sec \mathcal{C}\ell(M, \mathbf{g})$  such that its restriction over  $\sigma$  is  $\mathbf{v}$ , i.e.,  $V|_\sigma = \mathbf{v}$ . Also we impose that  $V^2 = 1$ . We introduce also the  $\mathbf{P}$  vector field such that  $\mathbf{P}|_\sigma = \mathbf{p}$  and consider the equation

$$\mathbf{P} = m\mathbf{V} \tag{63}$$

Let  $\psi \in \sec \mathcal{C}\ell^0(M, \mathbf{g})$ , be the representative (in the spin coframe  $\Xi$ ) of a *particular* invertible Dirac-Hestenes spinor field such that

$$\psi \tilde{\psi} \neq 0 \tag{64}$$

and since  $\psi = \rho^{\frac{1}{2}} e^{\frac{\beta}{2} \gamma^5} R$  we have

$$\psi \gamma^0 \tilde{\psi} = \rho e^{\beta \gamma^5} R \gamma^0 \tilde{R}. \tag{65}$$

$$\mathbf{P} = m\psi \gamma^0 \tilde{\psi} \tag{66}$$

$$\mathbf{P}\psi = m\psi \gamma^0 \tag{67}$$

Eq.(67) is a purely classical equation which is simply another way of writing Eq.(63). To get a quantum mechanics wave equation we must now change  $\mathbf{P}$  into  $\mathbf{P}$ , the quantum mechanics momentum operator. We know that

$$\mathbf{P}\psi = \partial \psi \gamma^2 \gamma^1 = \gamma^\mu \partial_\mu \psi \gamma^2 \gamma^1. \tag{68}$$

Substituting this result in Eq.(67) we get

$$\partial \psi \gamma^2 \gamma^1 - m\psi \gamma^0 = 0. \tag{69}$$

which is the DHE, which, as well known, is completely equivalent to the standard Dirac equation formulated in terms of covariant Dirac spinor fields.

<sup>14</sup>  $\tilde{\psi}$  is the reverse of  $\psi$ . If  $A_r \in \sec \wedge^r T^* M \hookrightarrow \sec \mathcal{C}\ell(M, \eta)$  then  $\tilde{A}_r = (-1)^{\frac{r}{2}(r-1)} A_r$ .

<sup>15</sup>  $\partial_\mu := \frac{\partial}{\partial x^\mu}$ .



## References

- [1] Arcidiacono, G. , *Relativit  e Cosmologia*, vol. II (iv edizione), Libreria Eredi Virgilio Vechi, Roma, 1987.
- [2] Cot escu, I. I., The Physical Meaning of the de Sitter Invariants, *Gen. Rel. Grav.* **43**, 1639-1656 (2011). [arXiv:1006.1472v6 [gr-qc] ]
- [3] Crummeyle, A., *Orthogonal and Symplectic Clifford Algebras*, Kluwer Acad. Publ., Dordrecht, 1.990.
- [4] Dirac, P. A. M., The Electron Wave Equation in De-Sitter Space, *Ann. Math.* **36**, 657-669 (1935).
- [5] Geroch, R., Spinor Structure of Space-Times in General Relativity I, *J. Math. Phys.* **9**, 1739-1744 (1968).
- [6] G rsey, F., Introduction to Group Theory, in DeWitt, C. and DeWitt, B. (eds.), *Relativity, Groups and Topology*, pp 91-161, Gordon and Breach, New York, 1964.
- [7] Hestenes D., and Sobczyk, G., *Clifford Algebras to Geometrical Calculus*, D. Reidel Publ. Co., Dordrecht, 1984.
- [8] Le o, R. F. , Rodrigue, W. A. Jr. and Wainer, S. A., *Concpet of Lie Derivative of Spinor Fields. A Geometric Motived Approach*. [arXiv:1411.7845 [math-ph] ]
- [9] Lounesto P., *Clifford Algebras and Spinors*, Cambridge Univ. Press, Cambridge, 1997.
- [10] Rodrigues, W A. Jr., and Wainer, S. A., A Clifford Bundle Approach to the Differential Geometry of Branes, *Adv. in Appl.Clifford Algebras* **24**, 617-847 (2014). [arXiv:1309.4007]
- [11] Notte-Cuello, E. A. and Capelas de Oliveira, E., Klein-Gordon and Dirac Equations in de Sitter Space Time, *Int. J. Theor. Phys.* **38**, 585-598(1999).
- [12] Riordan, F., Solutions of the Dirac Equation in Finite de Sitter Space 9Representations of  $SO_{4,1}$ , *N. Cimento B* **20**, 309-325 (1974).
- [13] Rodrigues, W. A. Jr., Algebraic and Dirac-Hestenes Spinors and Spinor Fields, *J. Math. Phys.* **45** , 2908-2994 (2004) [ arXiv:math-ph/0212030].
- [14] Rodrigues, W. A. Jr. and Capelas de Oliveira, E., *The Many Faces of Maxwell, Dirac and Einstein Equation,. A Clifford Bundle Approach*, Lecture Notes in Physics **722**, Springer, Heidelberg, 2007. A preliminary enlarged second edition may be found at <http://www.ime.unicamp.br/~walrod/svmde04092013.pdf>
- [15] Rodrigues, W. A. Jr., Vaz Jr., J. and Pavsic, M., The Clifford Bundle and the Dynamics of the Superparticle, Banach Center Publications. *Polish Acad. Sci.* **37**, 295-314 (1996).
- [16] Vaz Jr., J., *Space-Time Algebra, Dirac-Hestenes Spinors and the Theory of the Electron*, Ph.D. Thesis UNICAMP (1993).

# LINE SEGMENTS EXTRACTION FROM IMAGES USING RBF IN CGA

**Julio Zamora<sup>a</sup> , Allen Galaviz<sup>b</sup> , Alejandro Madrigal<sup>b</sup> , Miguel Padilla<sup>b</sup> and Ana Paulina Cassale<sup>b</sup>**

<sup>a</sup> Intel Labs Guadalajara  
julio.c.zamora.esquivel@intel.com [presenter, corresponding]

<sup>b</sup> Universidad Panamericana, Guadalajara, Mexico  
0163820@up.edu.mx

## 1. ABSTRACT

The geometric primitive extraction from images is essential in image processing to optimize the compression of files, image analysis, pattern recognition and three-dimensional reconstruction of scenes. There is a wide range of algorithms that address the problem in different ways, from Hough's Transform, to some based on genetic and optimization algorithms. However, the proposed method uses Radial Basis Neural Networks and line segments to speed up tracking and detection of the main geometric primitives that compose the given picture. This technique allows three-dimensional reconstruction of scenes avoiding the use of a Point Cloud Data, which is the most common method, but which turns out to be computationally expensive and very demanding of memory, and therefore, of difficult implementation in mobile devices.

## 2. INTRODUCTION

The main purpose of this paper is the extraction of line segments from an image. For this purpose, the image under analysis is pre-processed using a derivative filter in order to detect the borders of the objects on the image. This pre-processed image is used by the Neural Network for its own training, and finally, to imitate the image's topology. While this process is taking place, the lines that represent our geometric primitives align with the object's borders. Additionally, the gradient calculation in the image is accelerated due to the Neural Network interpolating the image. This allows the calculation of the derivative at each point without creating an histogram.

For the sake of clarity, a brief description of what a Radial Basis Function Network is, will be delivered:

**2.1. Radial Basis Function Neural Networks.** The RBF neural network is a special type of artificial neural network composed by two layers (see fig 1) in the first layer all neurons use radial basis functions as activation function. Such as

$$(1) \quad a = e^{-(x-x_0)^2 - (y-y_0)^2},$$

Being Gaussian function (1) the most commonly used, this function is also called kernel. The output ( $f$ ) of this type of neural networks is based on a linear combination ( $S = \sum w_i a_i$ ) of all activation function (outputs) in the first layer  $a_i$ . As visually represented in (Fig 1b) thanks to

the linear combination of all the Gaussian functions it is possible to interpolate functions of any kind, in this case the image should be interpreted as a function, sampled by pixels.

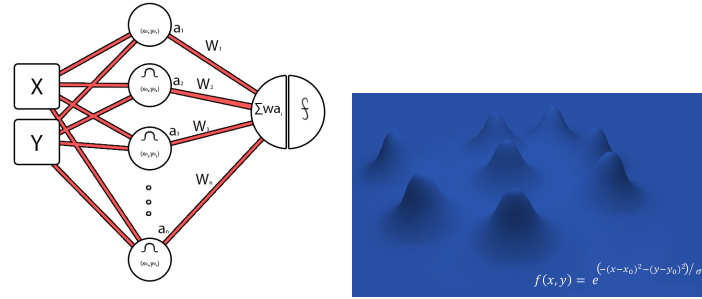


FIGURE 1. a)RBF Neural Network, b)Neural network evaluation in a 2D space.

Unfortunately, due to the image complexity (because most of images contain many high frequency components) too many Gaussian functions are required for the first layer, increasing the computational overhead of the neural network evaluation.

Because both neural network topology and the training are equally important, the later process will be explained below:

**2.2. Training a RBF using Gaussian kernel.** There are many training techniques. Because of its simplicity, the gradient descendant technique is used. Using mean square error:

$$(2) \quad E = \frac{1}{2}(d - f)^2$$

where  $d$  is the desired value and  $f$  the output of the network. In this case being computed by the use of the sigmoid activation function:

$$(3) \quad f = \frac{1}{1 + e^{-S}}$$

computing the partial derivative of the error in relation to each input weight ( $w_i$ ):

$$(4) \quad \frac{\partial E}{\partial w_i} = (d - f) \frac{\partial f}{\partial w_i}$$

$$(5) \quad \frac{\partial f}{\partial w_i} = f(1 - f) \frac{\partial S}{\partial w_i} = f(1 - f)a_i$$

$$(6) \quad \frac{\partial E}{\partial w_i} = (d - f)(1 - f)f a_i$$

Equation (6) represents the training rule for the second layer of the neural network. To generate the training rules for the first layer, the partial derivative of the error  $E$  is computed ( $\frac{\partial E}{\partial x_0}$ ) for each variable  $x_0$  and  $y_0$ :

$$(7) \quad \frac{\partial E}{\partial x_0} = (d - f) \frac{\partial f}{\partial x_0} = (d - f)(1 - f)f \frac{\partial S}{\partial x_0}$$

recalling  $s = \sum x_i w_i$ ,

$$(8) \quad \frac{\partial S}{\partial x_0} = \frac{\partial \sum x_i w_i}{\partial x_0} = w_i \frac{\partial a_i}{\partial x_0}$$

and using  $a_i = e^{-(x-x_0)^2-(y-y_0)^2}$  as kernel, the partial derivative is given by:

$$(9) \quad \frac{\partial a_i}{\partial x_0} = \underbrace{-e^{-(x-x_0)^2-(y-y_0)^2}}_{a_i} \cdot 2(x-x_0)$$

Finally the error gradient can be expressed as:

$$(10) \quad \frac{\partial E}{\partial x_0} = -[(d-f)f(1-f)w_i a_i \cdot 2(x-x_0)]$$

The equation (10) represents the training rule for  $x_0$ . The same methodology can be used to compute a rule for  $y_0$ :

$$(11) \quad \frac{\partial E}{\partial y_0} = -[(d-f)f(1-f)w_i a_i \cdot 2(y-y_0)]$$

After this brief overview of Gaussian RBFs, we will introduce a new kernel that will be used in the first layer of the network which is composed by the Gaussian and the sigmoid kernel

### 3. GAUSS-SIGMOID KERNEL

Based on the geometric entity pair of points  $PP = P_1 \wedge P_0$  defined in CGA ( $G_{4,1}$ ) a definition line segment function will be created. Such function is smooth and its derivative is also defined in the domain, and can be used as kernel for the neural network.

Considering the circle  $Z = PP \wedge p_2$  given by the pair of points  $PP$  and the point  $p_2 = xe_1 + ye_2 + 1/2(x^2 + y^2)e_\infty + e_o$ , the intensity of  $a(x, y)$  is being showed in figure 3a. By using kernel in the equation (12) the "segmentoid" function is being defined as:

$$(12) \quad a = uv = \frac{e^{-r^2}}{1 + e^{-s}}$$

Where  $r = Z \cdot e_{12}e_\infty$  and  $\rho = Z \cdot B$ . The kernel (12) is the result of the multiplication of the Gaussian kernel  $u = e^{-r^2}$  (see figure 3a) and the sigmoid kernel  $v = \frac{1}{1+e^{-s}}$  (figure3b). It is important to highlight that  $e_{12}e_\infty$  and  $B = (e_1 + e_2)e_{45}$  represents constant multivector, the graphical representation of the kernel  $a$  is being presented in the figure (3).

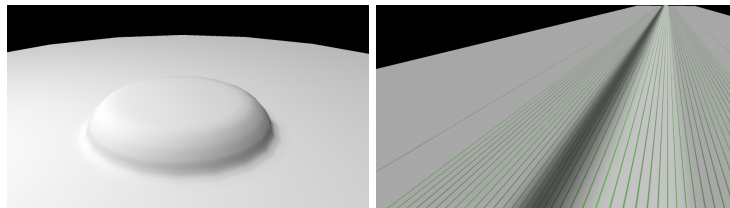


FIGURE 2. a) Sigmoid kernel  $v = \frac{1}{1+e^p}$  based having a circle as treshold function  
b) Gaussian kernel  $u = e^{-r^2}$  based in point line distance.



FIGURE 3. Evaluation of the "segmentoid" function  $a(x,y)$ .

3.1. **Training.** As we mention before, the output layer was created by a single perceptron, so the training is made following the gradient descendant and using as activation function the sigmoid  $f = \frac{1}{1+e^{-s}}$ , where  $S = \sum wx$  is the linear combination of all inputs (first layer's outputs). In order to use the gradient descendant algorithm, the error function needs to be defined:

$$(13) \quad E = \frac{1}{2}(d - f)^2$$

where  $d$  represents the desired output for a given input  $(x, y)$ . The training rule for the  $i$ -th weight  $w_i$  is given by the error derivative:

$$(14) \quad \frac{\partial E}{\partial w_i} = (d - f)(1 - f)fx_i$$

Following the same procedure, it is possible to compute the training rules for the components of  $Pp$ , that allow reducing the error and adjusting each line segment to the image border. In order to find the training rules using the gradient descendant, the partial derivatives of the kernel (15) will be computed  $\frac{\partial a(x,y)}{\partial x_0}$ .

$$(15) \quad a = uv = \frac{e^{-r^2}}{1 + e^{-\rho}}$$

$$(16) \quad \frac{\partial a}{\partial x_0} = \frac{\partial uv}{\partial x_0} = u \frac{\partial v}{\partial x_0} + v \frac{\partial u}{\partial x_0}$$

Since  $u = e^{-r^2}$  and  $v = \frac{1}{1+e^{-\rho}}$  with partial derivatives  $\frac{\partial u}{\partial x_0} = -2ru \frac{\partial r}{\partial x_0}$  and  $\frac{\partial v}{\partial x_0} = v(1 - v) \frac{\partial \rho}{\partial x_0}$  the equation (16) can be written as

$$(17) \quad \frac{\partial a}{\partial x_0} = uv(1 - v) \frac{\partial \rho}{\partial x_0} - 2ruv \frac{\partial r}{\partial x_0} = a((1 - v) \frac{\partial \rho}{\partial x_0} - 2r \frac{\partial r}{\partial x_0})$$

Then the derivative of  $r$  is given by  $\frac{\partial r}{\partial x_0} = \frac{\partial Z}{\partial x_0} \cdot e_{12}e_\infty$  while the  $\rho$  derivative is  $\frac{\partial \rho}{\partial x_0} = \frac{\partial Z}{\partial x_0} \cdot B$ . Replacing such derivatives in 17 to have partial derivative of  $a$  in terms of circle  $Z$ ;

$$(18) \quad \frac{\partial a}{\partial x_0} = a((1 - v)B - 2re_{12}e_\infty) \cdot \frac{\partial Z}{\partial x_0}$$

Following the same procedure the partial derivative of kernel (15) respect to  $x_1, y_0$  and  $y_1$ , provides the rules for training. Simplifying the equation (18) and doing the same procedure for the rest of the variables in the pair of points:

$$(19) \quad x'_0 = x_0 - \delta((1 - v)(x_1 - x) - 2r(y_1 - y))$$

$$(20) \quad x'_1 = x_1 - \delta((1 - v)(x_0 - x) - 2r(y - y_0))$$

$$(21) \quad y'_0 = y_0 - \delta((1 - v)(y_1 - y) - 2r(x - x_1))$$

$$(22) \quad y'_1 = y_1 - \delta((1 - v)(y_0 - y) - 2r(x_0 - x))$$

where  $\delta = \eta(d - f)(1 - f)fw_i a_i$  and  $\eta$  is an scalar value that represents the leaning rate. So far we have the neural network description and the definition of the training rules. Now lets discuss about our application and results obtained.

**3.2. Line extraction.** As we mention before, the main goal is to proximate a line segment for each border in the analyzed image (borders image). The borders image is the result of applying a filter(sobel filter) in the original image. In order to achieve our goal, the borders image will be used to train the neural network (NN). After the training, the NN should be capable to recreate the image (i.e. learn how to redo the same image, see figure 3.2). In this way, it is possible to evaluate the NN in a coordinate (x,y) and it should return the value of the intensity of the border in that particular pixel. After the training of the NN, it is possible to change the resolution of the image, while keeping the same behavior. Another good thing about this technique is regarding the noise removal, meaning small segments will be ignored due to the number of neurons in our NN topology. Additionally, after the training, since each neuron represents a line segment all, we need to do to extract the line segments from the image is to look for the weights in the first layer, because they actually are the extreme point in the segment  $(x_0, y_0)$  and  $(x_1, y_1)$ .

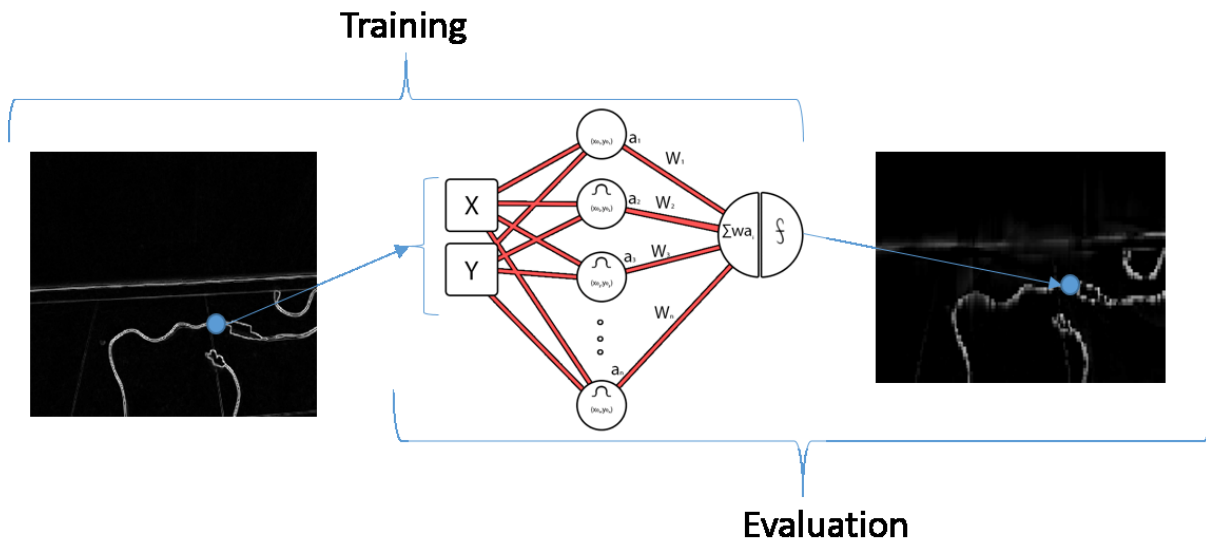


FIGURE 4. Usage Pipeline

The algorithm was implemented using C++ showing good performance (processing 29FPS), the image (3.2a) shows the input image (borders image) and (3.2b) shows the line segments iterating to converge at the book borders.

#### 4. CONCLUSIONS

By using this algorithm we were able to extract and approximate line segments faster than using Hough's transform, with the advantage that we have the segments not the entire line as we do

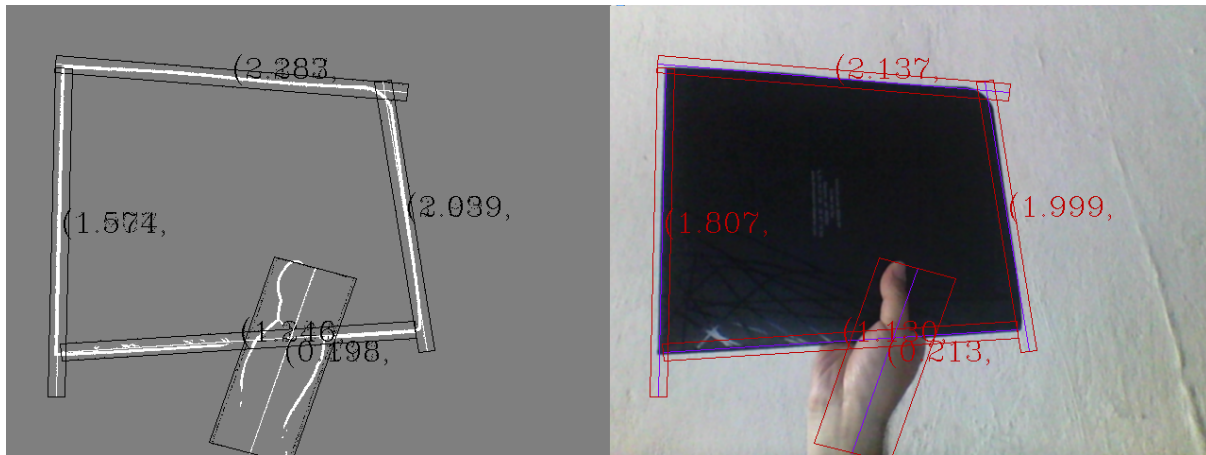


FIGURE 5. a) shows the input image, b) line segments iterating over the original image to converge around the book.

using Hough transform. In some simple scenarios lines segments moves to reduce the error achieving the real position of the edge. It is possible to increase and decrease the number of line segments by increasing the number of neurons having an impact in the performance, the memory used increases linearly with the increment of the line segments and the computational time increases also linearly. Our next goal will be the geometric primitive extraction using an stereo camera to retro-project the pairs of line segments and build a 3D environment around the camera.

#### REFERENCES

- [1] Wang, S.W., Kaufman, A.E., *Volume sampled voxelization of geometric primitives*, pages 78 - 84, San Jose, CA., 1993.
- [2] G. Roth, M.D. Levine, *Extracting Geometric Primitives*, pages 1–22, Ontario, Canada, 1993.
- [3] Marc Levoy, Turner Whitted, *The Use of Points as a Display Primitive*, pages 1 - 19, North Carolina.
- [4] Lutton, E., Martinez, P., *A genetic algorithm for the detection of 2D geometric primitives in images*, pages 526 - 528 vol.1, Jerusalem, 1994.
- [5] El-Maleh, A., al Zahir, S., Khan, E., *A geometric-primitives-based compression scheme for testing systems-on-a-chip*, pages 54 - 59, Marina Del Rey, CA, 2001.
- [6] Cristiano Premebida, *Segmentation and Geometric Primitives Extraction from 2D Laser Range Data for Mobile Robot Applications*, pages 1 - 24, Coimbra, Portugal, 2005.
- [7] Clark F. Olson, *Locating geometric primitives by pruning the parameter space*, pages 1 - 10, Pasadena, CA., 2001.
- [8] Thomas Chaperon, Francois Goulette, *Extracting cylinders in full 3D data using a random sampling method and the Gaussian image*, pages 1 - 42, Paris, France, 2001.
- [9] Qifa Ke, Tianzi Jiang, Song De Ma, *A tabu search method for geometric primitive extraction*, pages 1443–1451, Beijing, 1998.
- [10] Dmitry Lagunovsky, Sergey Ablameyko, *Straight-line-based primitive extraction in grey-scale object recognition*, pages 1005–1014, Minsk, Byelorussia, 1999.
- [11] Serge Belongie, Jan Puzicha, *Shape Matching and Object Recognition Using Shape Contexts*, pages 509 - 522, Pattern Analysis and Machine Intelligence, IEEE Transactions on, Apr 2002.
- [12] Levente Kovács, *Shape retrieval and recognition on mobile devices*, pages 90-101, Springer Berlin Heidelberg, 2012.

# CLASSICAL FIELD THEORIES FROM HAMILTONIAN CONSTRAINT: CANONICAL EQUATIONS OF MOTION AND LOCAL HAMILTON-JACOBI THEORY

Václav Zatloukal

Faculty of Nuclear Sciences and Physical Engineering,  
Czech Technical University in Prague,  
Břehová 7, 115 19 Praha 1, Czech Republic and  
Max Planck Institute for the History of Science,  
Boltzmannstrasse 22, 14195 Berlin, Germany

**ABSTRACT.** Classical field theory is considered as a theory of unparametrized surfaces embedded in a configuration space, which accommodates, in a symmetric way, spacetime positions and field values. Dynamics is defined by a (Hamiltonian) constraint between multivector-valued generalized momenta, and points in the configuration space. Starting from a variational principle, we derive local equations of motion, that is, differential equations that determine classical surfaces and momenta. A local Hamilton-Jacobi equation applicable in the field theory then follows readily. The general method is illustrated with three examples: non-relativistic Hamiltonian mechanics, scalar field theory, and string theory.

Throughout, we use the mathematical formalism of geometric algebra and geometric calculus, which allows to perform completely coordinate-free manipulations.

## 1. INTRODUCTION

In non-relativistic mechanics, the trajectory of a particle is a function  $x(t)$ , which describes how the position of the particle changes with time. In relativistic mechanics, space and time are treated equally, and the particle's trajectory is regarded as a sequence of spacetime points  $(t, x)$ .

In field theory, the field configuration is usually viewed as a function  $\phi(x)$  that describes how the field varies from point to point. However, general relativity suggests [1] that the spacetime is a dynamical entity, which should be put with fields on the same footing. Mathematically, instead of a function  $\phi(x)$  one should therefore consider the respective graph, i.e., the collection of points  $(x, \phi)$ .

In this article, we study the mathematical formalism proposed in [1, Ch. 3] that treats time, space, and fields equally. All these entities are collectively called *partial observables*, and together they form a finite-dimensional *configuration space*. Classical field theory predicts that certain correlations between partial observables can be realized in nature. These are then called *physical motions*, and have the form of surfaces embedded in the configuration space.

Our dynamical description utilizes multivector-valued momentum variable, which can be thought of as conjugated to the motion's tangent planes; thus generalizing the canonical momentum conjugated to the velocity vector in classical mechanics. Individual theory is specified by a choice of the *Hamiltonian*  $H$ , which is a function of a configuration space point  $q$  and momentum  $P$ . This Hamiltonian enters into a variational principle (Section 2) via the *Hamiltonian constraint*  $H(q, P) = 0$ .

The aim of this article is to establish, in the first place, equations of motion that follow from the variational principle. This is done in Section 3, Eqs. (13). These equations generalize the Hamilton's canonical equations of motion of classical mechanics. From Eqs. (13) we derive the local Hamilton-Jacobi equation (20), which generalizes to the field theory the respective



concept from classical mechanics (in this context, see also Refs. [2] and [3]). It should be stressed that both, Eqs. (13) and (20), contain only partial, not variational, derivatives.

Three examples are provided in Section 5 to demonstrate universality of the present formalism. In the first example (5.1), non-relativistic mechanics is deduced when we assume that the motions are one dimensional curves, and choose the Hamiltonian  $H$  appropriately. Eqs. (13) then reduce to the Hamilton's canonical equations, accompanied by the law of energy conservation, and the Hamilton-Jacobi equation of classical mechanics is recovered.

The second example (5.2) discusses the theory of real one-component scalar field defined on a Euclidean spacetime of any dimension. Eqs. (13) produce the correct field equation, and, at the same time, they incorporate the continuity equation for the energy-momentum tensor. Hamilton-Jacobi equation reproduces the one invented by Weyl [4].

In the last example (5.3) we treat relativistic particle, string, or higher-dimensional membrane, depending on the dimensionality of the motions. The configuration space is identified with the target space of the string theory, motions are the worldsheets, and the corresponding Hamiltonian is essentially the simplest and most symmetric function of the momentum variable. The equations of motion have simple geometric meaning, namely, they ensure that the mean curvature of the physical motion vanishes. In fact, this is exactly the condition that defines *minimal surfaces* [5].

One more remark is in order before we start. All manipulations are performed in the mathematical formalism of geometric (or Clifford) algebra and calculus developed by D. Hestenes [6] (see also Ref. [7]). It is a coordinate-free language that is more universal than the calculus of differential forms, nevertheless, it is yet not well-recognized by a broad audience. Reader unfamiliar with geometric algebra or calculus is recommended to first read, e.g., Appendix A of Ref. [8], where the basics are briefly introduced.

## 2. VARIATIONAL PRINCIPLE

We start with a set of partial observables that constitute a  $(D + N)$ -dimensional Euclidean configuration space  $\mathcal{C}$ . A point  $q$  in the configuration space represents simultaneous measurement of all partial observables, e.g.,  $q = (x, \phi)$ . To establish a *physical* theory, one has to specify correspondence between the partial observables, and physical measuring devices, such as clock, rulers, or instruments measuring components of the field. In this article we take such correspondence for granted, as we are only concerned with the *mathematical* aspects of the theory.

Denote by  $D$  the dimensionality of motions, i.e., submanifolds  $\gamma$  of the configurations space  $\mathcal{C}$ . With  $D = 1$  we can study particle mechanics, with  $D = 2$  we can do string theory, or field theory in two dimensions, and so on. We shall not deal with systems with gauge invariance, for which the mathematical motion (the surface in  $\mathcal{C}$ ) has higher dimensionality than the actual physical motion (the trajectory).

Tangent space of  $\gamma$  at point  $q$  is spanned by  $D$  linearly independent vectors  $a_1, \dots, a_D$ , which are conveniently combined into a grade- $D$  multivector  $a_1 \wedge \dots \wedge a_D$ . Normalized version of the latter is called the *unit pseudoscalar of  $\gamma$* , and it is denoted by  $I_\gamma$ . In the terminology used in Ref. [10, Ch. 6], the function  $I_\gamma(q)$  represents a  $D$ -dimensional distribution on  $\mathcal{C}$ , with  $\gamma$  being its integral manifold.

Fundamental for the following formulation of dynamics is the concept of generalized momentum, which is a grade- $D$  multivector, denoted by  $P$ , defined at each point of  $\gamma$  (see Fig. 1). It serves as a quantity conjugated to  $I_\gamma$ , thus generalizing the canonical momentum of particle mechanics.

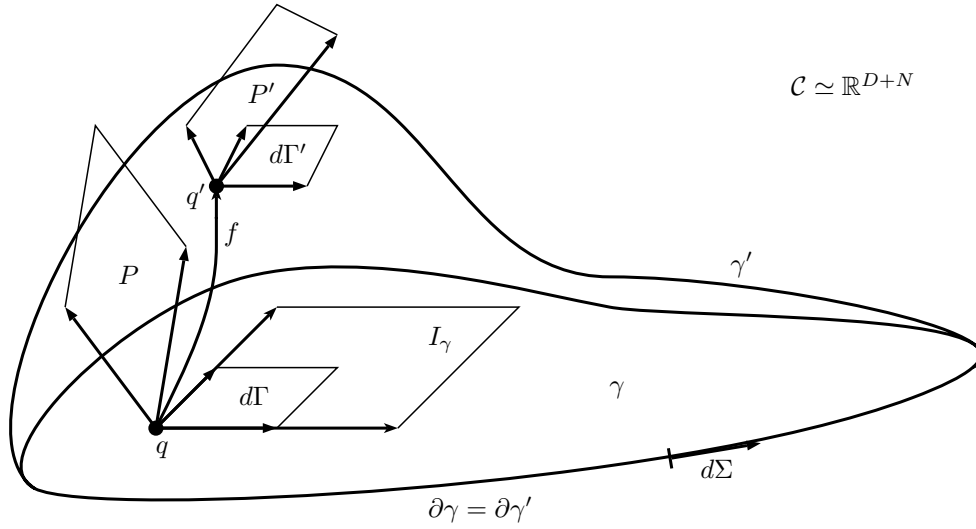


FIGURE 1. Variational principle.

The last ingredient is the Hamiltonian  $H(q, P)$ , which is supposed to be scalar-valued. (Generalization to the case of multicomponent  $H$  is straightforward.)

The variational principle can now be stated as follows (cf. [1, Ch. 3.3.2]):

**Variational principle.** *A surface  $\gamma_{\text{cl}}$  with boundary  $\partial\gamma_{\text{cl}}$  is a physical motion, if the couple  $(\gamma_{\text{cl}}, P_{\text{cl}})$  extremizes the (action) functional*

$$(1) \quad \mathcal{A}[\gamma, P] = \int_{\gamma} P(q) \cdot d\Gamma(q)$$

in the class of pairs  $(\gamma, P)$ , for which  $\partial\gamma = \partial\gamma_{\text{cl}}$ , and  $P$  defined along  $\gamma$  satisfies

$$(2) \quad H(q, P(q)) = 0 \quad \forall q \in \gamma.$$

(The subscript “cl” stands for “classical” motion, or trajectory, which we sometimes use instead of the expression “physical” motion.)

The integral in (1) is defined in [6, Ch. 7-1] (see also Ref. [9]) without having recourse to a parametrization of the surface  $\gamma$ . Of course, if desired, the oriented surface element  $d\Gamma$  can be cast, using arbitrary coordinates on  $\gamma$ , as

$$(3) \quad d\Gamma = \left( \frac{\partial q}{\partial \tau_1} \wedge \dots \wedge \frac{\partial q}{\partial \tau_D} \right) d\tau_1 \dots d\tau_D.$$

The integrand in (1) may be rephrased as a differential form  $\theta = p_{j_1 \dots j_D} dq^{j_1} \wedge \dots \wedge dq^{j_D}$ . However, the merit of geometric calculus consists, in fact, in splitting of the differential form into two parts,  $d\Gamma$  and  $P$ , where  $P$  is able to enter into functions such as  $H(q, P)$ .

Finally, let us note that in [1, Ch. 3.3.2] the action is an integral of  $\theta$  over the submanifolds of the bundle of  $D$ -forms over  $\mathcal{C}$ . Since we hesitate to work in spaces mixing points  $q$  and multivectors  $P$ , we rather operate with surfaces in  $\mathcal{C}$ , on which the momentum field is defined.

### 3. CANONICAL EQUATIONS OF MOTION

We will now derive the equations of motion that follow from the variational principle of Section 2. For this purpose, we incorporate the Hamiltonian constraint (2) into the action (1) by means of a scalar Lagrange multiplier  $\lambda$ . The augmented action is the functional

$$(4) \quad \mathcal{A}[\gamma, P, \lambda] = \int_{\gamma} [P(q) \cdot d\Gamma(q) - \lambda(q)H(q, P(q))],$$

where  $\lambda$  is in fact an infinitesimal quantity with magnitude comparable to  $|d\Gamma|$  – the magnitude of  $d\Gamma$ .

Varied action  $\mathcal{A}[\gamma', P', \lambda']$  is the integral taken over a new surface  $\gamma'$ , and featuring new functions  $P'$  and  $\lambda'$ , which are defined along  $\gamma'$  (see Fig. 1). Let  $f$ , where

$$(5) \quad f(q) = q + \delta q(q),$$

be the infinitesimal diffeomorphism mapping between  $\gamma$  and  $\gamma'$ , i.e.,  $\gamma' = \{q' = f(q) \mid q \in \gamma\}$ , and denote by

$$(6) \quad \delta P(q) \equiv P'(f(q)) - P(q) \quad \text{and} \quad \delta \lambda(q) \equiv \lambda'(f(q)) - \lambda(q)$$

the variations of momentum and Lagrange multiplier, respectively.

Variation of the action (4),  $\delta \mathcal{A} \equiv \mathcal{A}[\gamma', P', \lambda'] - \mathcal{A}[\gamma, P, \lambda]$ , is given by

$$(7) \quad \begin{aligned} \delta \mathcal{A} = & \int_{\gamma'} [P'(f(q)) \cdot \underline{f}(d\Gamma(q); q) - \lambda'(f(q))H(f(q), P'(f(q)))] \\ & - \int_{\gamma} [P(q) \cdot d\Gamma(q) - \lambda(q)H(q, P(q))], \end{aligned}$$

where we have employed the integral substitution theorem [6, Ch. 7-5] to transform the integral over  $\gamma'$  into an integral over  $\gamma$ . For the infinitesimal diffeomorphism  $f$ , the outermorphism mapping  $\underline{f}$  that specifies the transformation rule for multivectors, is given by Formula (A29) in Ref. [8]. Therefore, up to first order in  $\delta q$ ,  $\delta P$  and  $\delta \lambda$ , we find

$$(8) \quad \begin{aligned} \delta \mathcal{A} = & \int_{\gamma} [(P + \delta P) \cdot (d\Gamma + (d\Gamma \cdot \partial_q) \wedge \delta q) - (\lambda + \delta \lambda)H(q + \delta q, P + \delta P) - P \cdot d\Gamma + \lambda H(q, P)] \\ \approx & \int_{\gamma} [-\delta \lambda H(q, P) + \delta P \cdot (d\Gamma - \lambda \partial_P H(q, P)) - \lambda \delta q \cdot \dot{\partial}_q H(\dot{q}, P) + P \cdot ((d\Gamma \cdot \partial_q) \wedge \delta q)], \end{aligned}$$

where the *vector derivative*  $\partial_q$ , and the *multivector derivative*  $\partial_P$  are defined in [6, Ch. 2-1], and [6, Ch. 2-2], respectively. The “overdot” notation is used here to indicate the scope of the differential operator  $\partial_q$ , and has nothing to do with time derivative. Without an overdot, any differential operator is supposed to act on functions that stand to its right.

The last term in (8) can be recast with a help of the *Fundamental theorem of geometric calculus* (see [6, Ch. 7-3]),

$$(9) \quad \int_{\gamma} P \cdot ((d\Gamma \cdot \partial_q) \wedge \delta q) = \int_{\partial\gamma} P \cdot (d\Sigma \wedge \delta q) - \int_{\gamma} \dot{P} \cdot ((d\Gamma \cdot \dot{\partial}_q) \wedge \delta q),$$

where  $d\Sigma$  is the oriented volume element of the boundary  $\partial\gamma$ . Now, the first term on the right-hand side vanishes, since we assume that  $\gamma$  and  $\gamma'$  have common boundary.

For  $D = 1$ ,  $d\Gamma \cdot \partial_q$  is algebraically a scalar, and so the integrand in the second term is readily reshuffled,

$$(10) \quad \dot{P} \cdot ((d\Gamma \cdot \dot{\partial}_q) \wedge \delta q) = \delta q \cdot (d\Gamma \cdot \partial_q P).$$

(Mind the priority of the inner product “ $\cdot$ ”, and the outer product “ $\wedge$ ” before the geometric product, which is denoted by an empty symbol.)

For  $D > 1$ , we find

$$(11) \quad \dot{P} \cdot ((d\Gamma \cdot \dot{\partial}_q) \wedge \delta q) = (\dot{P} \cdot (d\Gamma \cdot \dot{\partial}_q)) \cdot \delta q = (-1)^{D-1} \delta q \cdot ((d\Gamma \cdot \partial_q) \cdot P).$$

The two cases have to be treated separately due to the definition [6, Eq. 1-1.21] of the inner product.

After these rearrangements we arrive at our final expression for the variation of the action,

$$(12) \quad \delta \mathcal{A} \approx \int_{\gamma} \left[ -\delta \lambda H(q, P) + \delta P \cdot (d\Gamma - \lambda \partial_P H(q, P)) \right. \\ \left. + \delta q \cdot \left( (-1)^D (d\Gamma \cdot \partial_q) \cdot P - \lambda \dot{\partial}_q H(\dot{q}, P) \right) \right],$$

which holds for  $D > 1$ , while the case  $D = 1$  is obtained simply by replacing  $(d\Gamma \cdot \partial_q) \cdot P$  with  $d\Gamma \cdot \partial_q P$ . The requirement that the variation of the action be zero for all  $\delta P$ ,  $\delta q$  and  $\delta \lambda$  yields the following

**Canonical equations of motion.** *Physical motions  $\gamma_{\text{cl}}$  are obtained by solving the system of equations*

$$(13a) \quad \lambda \partial_P H(q, P) = d\Gamma,$$

$$(13b) \quad (-1)^D \lambda \dot{\partial}_q H(\dot{q}, P) = \begin{cases} d\Gamma \cdot \partial_q P & \text{for } D = 1 \\ (d\Gamma \cdot \partial_q) \cdot P & \text{for } D > 1, \end{cases}$$

$$(13c) \quad H(q, P) = 0.$$

(We use the adjective ‘‘canonical’’, because these equations generalize, as we shall see in Example 5.1, Hamilton’s canonical equations of motion of classical mechanics.)

The first canonical equation (13a) furnishes a relation between the momentum  $P$ , and the tangent planes of  $\gamma$  represented by the oriented surface element  $d\Gamma$ . It asserts that the multivector derivative  $\partial_P H$ , which is a grade- $D$  multivector, is proportional to  $d\Gamma$ , with the proportionality constant equal to  $\lambda$ . Note that one can always normalize  $d\Gamma$  and  $\lambda$  by the magnitude  $|d\Gamma|$  to free Eqs. (13) from infinitesimal quantities.

The second canonical equation (13b) describes how the momentum multivector  $P$  changes as it slides along the surface  $\gamma$ . It is important to note that  $P$  is differentiated effectively only in the directions parallel to  $\gamma$ , as a consequence of the inner product between the surface element  $d\Gamma$ , and the vector derivative  $\partial_q$ . Moreover, the ‘‘overdot’’ on the left-hand side assures that only explicit dependence of  $H$  on  $q$  is being differentiated, not the dependence through  $P(q)$ .

The last canonical equation (13c) is the Hamiltonian constraint (2). Let us remark that had we started with several constraints  $H_j(q, P) = 0$  in the variational principle, we would have introduced the corresponding number of Lagrange multipliers  $\lambda_j$ , and the canonical equations would contain the terms  $\sum_j \lambda_j H_j$  instead of  $\lambda H$ .

#### 4. LOCAL HAMILTON-JACOBI THEORY

One possible method to approach the canonical equations (13) is the following. Suppose  $P(q)$  is given, that obeys the Hamiltonian constraint

$$(14) \quad H(q, P(q)) = 0$$

on some open subset of the configuration space  $\mathcal{C}$ . By differentiation, we obtain, according to the chain rule,

$$(15) \quad \dot{\partial}_q H(\dot{q}, P(q)) + \dot{\partial}_q \dot{P}(q) \cdot \partial_P H(q, P(q)) = 0,$$

and using the first canonical equation (13a), we find

$$(16) \quad \lambda \dot{\partial}_q H(\dot{q}, P(q)) = -\dot{\partial}_q \dot{P}(q) \cdot d\Gamma.$$

But the right-hand side may be rearranged,

$$(17) \quad \lambda \dot{\partial}_q H(\dot{q}, P(q)) = \begin{cases} d\Gamma \cdot (\partial_q \wedge P(q)) - d\Gamma \cdot \partial_q P(q) & \text{for } D = 1 \\ (-1)^{D-1} d\Gamma \cdot (\partial_q \wedge P(q)) + (-1)^D (d\Gamma \cdot \partial_q) \cdot P(q) & \text{for } D > 1, \end{cases}$$

and therefore we observe that if

$$(18) \quad \partial_q \wedge P(q) = 0,$$

the second canonical equation (13b) is automatically fulfilled. Momentum satisfying this condition can be expressed, at least locally, as  $P(q) = \partial_q \wedge S(q)$ , where  $S$  is a multivector of grade  $D - 1$ . Canonical equations (13) are then reduced to two equations,

$$(19) \quad \lambda \partial_P H(q, \partial_q \wedge S) = d\Gamma,$$

and the *local Hamilton-Jacobi equation*

$$(20) \quad H(q, \partial_q \wedge S) = 0.$$

If we succeed in finding a solution of Eq. (20), we can plug it into Eq. (19), which then defines a distribution of tangent planes of a classical motion surface. This distribution is integrable only if certain conditions are satisfied (see [10, Ch. 6.1]).

In addition, if we find a whole family of solution  $S(q; \alpha)$ , parametrized by a continuous parameter  $\alpha$ , then differentiating Eq. (20) with respect to  $\alpha$ , and substituting Eq. (19), yields

$$(21) \quad 0 = \lambda \partial_\alpha H(q, \partial_q \wedge S) = \lambda \dot{\partial}_\alpha (\partial_q \wedge \dot{S}) \cdot \partial_P H(q, \partial_q \wedge S) = d\Gamma \cdot (\partial_q \wedge (\partial_\alpha S)).$$

Now, for  $D = 1$ , the Hamilton-Jacobi function  $S$  is scalar-valued, and we have

$$(22) \quad d\Gamma \cdot \partial_q (\partial_\alpha S) = 0 \Rightarrow \partial_\alpha S(q; \alpha) = \beta \quad \forall q \in \gamma_{cl},$$

for some constant  $\beta$ , meaning that the quantity  $\partial_\alpha S(q; \alpha)$  is conserved along physical motion. If one finds  $N$  such parameters (recall that the dimension of the configuration space is  $N + 1$ ), the physical motion  $\gamma_{cl}$  can be given implicitly by a set of constraints between partial observables,

$$(23) \quad \begin{aligned} \partial_{\alpha_1} S(q; \alpha_1, \dots, \alpha_N) &= \beta_1 \\ &\vdots \\ \partial_{\alpha_N} S(q; \alpha_1, \dots, \alpha_N) &= \beta_N. \end{aligned}$$

Of course, we assume that the  $N$  constraints are independent, i.e., that the gradients  $\partial_q (\partial_{\alpha_1} S), \dots, \partial_q (\partial_{\alpha_N} S)$  are at each point linearly independent. In Example 5.3 we will illustrate the Hamilton-Jacobi method with the case of a relativistic particle.

When  $D > 1$ , we can cast Eq. (21) as

$$(24) \quad (d\Gamma \cdot \partial_q) \cdot (\partial_\alpha S) = 0 \Rightarrow \int_{\bar{\gamma}_{cl}} (d\Gamma \cdot \partial_q) \cdot (\partial_\alpha S) = \int_{\partial \bar{\gamma}_{cl}} d\Sigma \cdot (\partial_\alpha S) = 0,$$

where  $\bar{\gamma}_{cl}$  is an arbitrary  $D$ -dimensional subset of  $\gamma_{cl}$  (a ‘‘patch’’ on  $\gamma_{cl}$ ). Therefore, in the multi-dimensional case, the conservation law (22) is replaced with a certain continuity equation.

One remark is in order before closing this section. In classical particle mechanics, one of the solutions of the Hamilton-Jacobi equation is the action along classical trajectory, regarded as a function of one of the endpoints. In field theory, the classical action may be viewed as a functional of the boundary  $\partial \gamma_{cl}$ . Some authors (e.g. [1, Ch. 3.3.4]) have therefore considered a variational differential equation that describes how the classical action changes under variations of the boundary, naming it also ‘‘Hamilton-Jacobi equation’’. Note that Eq. (20) is substantially different from this kind of approach, for it contains only partial, not variational, derivatives. That is why we call it ‘‘local Hamilton-Jacobi equation’’. Local Hamilton-Jacobi theory is also treated, e.g., in Refs. [2] and [3].

## 5. EXAMPLES

In the following examples, we illustrate the general theory by specifying a concrete form of the Hamiltonian  $H(q, P)$ .

**5.1. Non-relativistic Hamiltonian mechanics.** Let us consider  $D = 1$ , and choose a constant unit vector  $e_t$  in the configuration space  $\mathcal{C} \simeq \mathbb{R}^{1+N}$ . Arbitrary point  $q$  can be decomposed as  $q = t + x$ , where  $t$  is parallel to  $e_t$ , while  $x$  is perpendicular to  $e_t$ .

Define the Hamiltonian as follows:

$$(25) \quad H(q, P) = P \cdot e_t + H_0(q, P),$$

where  $e_t \cdot \partial_P H_0 = 0$ , and  $H_0$  is identified as the non-relativistic Hamiltonian of a mechanical system.

When we present the physical motion  $\gamma_{\text{cl}}$  as

$$(26) \quad \gamma_{\text{cl}} = \{q = g(t) = t + x(t) \mid t \in \text{span}\{e_t\} \simeq \mathbb{R}\},$$

the line element on the  $t$ -axis is related to the line element on  $\gamma_{\text{cl}}$  via the differential  $\underline{g}$ ,

$$(27) \quad d\Gamma = \underline{g}(dt) = dt + dt \cdot \partial_t x.$$

It is shown in [8, Sec. V-A] that Eqs. (13) reduce to Hamilton's canonical equations of motion for a non-relativistic system with non-relativistic Hamiltonian  $H_0$ , together with the law of conservation of the total energy  $H_0$  (if  $e_t \cdot \partial_q H_0 = 0$ , i.e., if  $H_0$  does not depend explicitly on time). Intuitively, the Hamilton's canonical equations follow from variations of the trajectory  $\gamma$  in the  $x$ -space, while the energy conservation is a result of variations in the  $t$ -space.

Hamilton-Jacobi equation (20) for a scalar function  $S(q)$  reads

$$(28) \quad H(q, \partial_q S) = e_t \cdot \partial_q S + H_0(q, \partial_q S) = 0,$$

and hence reproduces the standard Hamilton-Jacobi equation of classical mechanics.

**5.2. Scalar field theory.** In this example we will show that the formalism based on Hamiltonian constraint can accommodate the theory of real one-component scalar field  $\phi(x)$ , defined on a  $D$ -dimensional Euclidean spacetime by the Lagrangian

$$(29) \quad \mathcal{L}(\phi, \partial_x \phi) = \frac{1}{2}(\partial_x \phi)^2 - V(\phi).$$

For this purpose, let us assume  $D > 1$ , choose a unit  $D$ -blade  $I_x$  in a  $(D + 1)$ -dimensional configuration space  $\mathcal{C}$ , and define the Hamiltonian

$$(30) \quad H(q, P) = P \cdot I_x + H_0(q, P),$$

where  $I_x \cdot \partial_P H_0 = 0$ . The blade  $I_x$  defines a splitting of the configuration space  $\mathcal{C}$  into a  $D$ -dimensional spacetime, spanned by an orthonormal set of vectors  $\{e_1, \dots, e_D\}$ ,  $I_x = e_1 \dots e_D$ , and the field space, which is its one-dimensional orthogonal complement, represented by a unit vector  $e_y$ .

A physical motion  $\gamma_{\text{cl}}$  assumes the form

$$(31) \quad \gamma_{\text{cl}} = \{q = g(x) = x + y(x) \mid x \in \text{span}\{e_1, \dots, e_D\} \simeq \mathbb{R}^D\}.$$

The surface element  $d\Gamma$  is related to the infinitesimal element of the  $x$ -space  $dX = |dX|I_x$  via the outermorphism mapping induced by the function  $g$ ,  $d\Gamma = \underline{g}(dX)$ .

Taking specifically

$$(32) \quad H_0 = \frac{1}{2} \sum_{j=1}^D (P \cdot E_j)^2 + V(\phi),$$

where  $\phi \equiv e_y \cdot y$ , Eqs. (13) yield

$$(33) \quad \partial_x^2 \phi = -\partial_\phi V(\phi),$$

which is the equation of motion of a scalar field described by the Lagrangian (29). In addition, owing to the fact that  $H_0$  does not depend on  $x$  explicitly, Eqs. (13) provide the continuity equation

$$(34) \quad e_k \cdot \partial_x \mathcal{T}_{jk} = 0,$$

where

$$(35) \quad \mathcal{T}_{jk} \equiv -\delta_{jk} \mathcal{L}(\phi, \partial_x \phi) + (e_j \cdot \partial_x \phi)(e_k \cdot \partial_x \phi)$$

is the canonical energy-momentum tensor of the scalar field with Lagrangian (29).

Detailed derivation may be found in [8, Sec. V-B], where the connection with De Donder-Weyl Hamiltonian field theory [4, 11, 12] is also discussed. Intuitively, the equation of motion (33) is a result of variations of the surface  $\gamma$  in the  $y$ -direction, while the continuity equation (34) follows from variations in the  $x$ -plane.

Finally, let us say a few words about the Hamilton-Jacobi theory. Equation (20) with Hamiltonian (30) reads

$$(36) \quad I_x \cdot (\partial_q \wedge S) + H_0(q, \partial_q \wedge S) = 0,$$

where  $S(q)$  is a multivector of grade  $D - 1$ . For concreteness, consider  $H_0$  given by Eq. (32), and assume  $S$  is parallel to  $I_x$ . Defining the vector  $s(q) \equiv S(q) \cdot I_x$ , also parallel to  $I_x$ , and blades  $E_j \equiv I_x e_j e_y$ , we observe that

$$(37) \quad \begin{aligned} I_x \cdot (\partial_q \wedge S) &= \partial_q \cdot s, \\ E_j \cdot (\partial_q \wedge S) &= e_y \cdot \partial_q e_j \cdot s. \end{aligned}$$

Eq. (36) then takes the form (note that  $\partial_\phi s = e_y \cdot \partial_q s$ )

$$(38) \quad \partial_q \cdot s + \frac{1}{2}(\partial_\phi s)^2 + V(\phi) = 0,$$

which coincides with the field-theoretic Hamilton-Jacobi equation derived formerly by Weyl [4].

**5.3. String theory.** Probably the simplest nontrivial Hamiltonian to consider is

$$(39) \quad H = \frac{1}{2}(|P|^2 - \Lambda^2),$$

where  $\Lambda > 0$  is a scalar constant, and  $|P|$  is the magnitude of  $P$ .

According to the basic multivector derivatives listed in [6, Ch. 2-2], the first canonical equation (13a) takes the form

$$(40) \quad d\Gamma = \lambda \tilde{P},$$

where  $\tilde{P}$  denotes the *reversion* of  $P$ , and substituting this relation into the Hamiltonian constraint (13c) fixes the absolute value of the Lagrange multiplier  $\lambda$ ,

$$(41) \quad |d\Gamma| = |\lambda| \Lambda.$$

Furthermore, substituting Eq. (40) into the second canonical equation of motion (13b), dividing by  $\lambda$ , and using Eq. (41) we find

$$(42) \quad \begin{aligned} I_\gamma \cdot \partial_q I_\gamma &= 0 \quad (D = 1), \\ (I_\gamma \cdot \partial_q) \cdot I_\gamma &= 0 \quad (D > 1), \end{aligned}$$

where  $I_\gamma \equiv d\Gamma/|d\Gamma|$  is the unit pseudoscalar of the surface  $\gamma$ . This equation has a simple geometric interpretation. It entails vanishing of the mean curvature of  $\gamma$ , or of its generalization, the *spur* vector (see Ref. [6, Ch. 4-4]).

Eqs. (40) and (41) allow us to rewrite the action (1) in terms of  $d\Gamma$ ,

$$(43) \quad \int_{\gamma} P \cdot d\Gamma = \int_{\gamma} \frac{1}{\lambda} |d\Gamma|^2 = \pm \Lambda \int_{\gamma} |d\Gamma|,$$

where “ $\pm$ ” is the sign of  $\lambda$ . This is the Euclidean Nambu-Goto action of the string theory [13]. It is proportional to the volume of the worldsheet  $\gamma$ , with  $\Lambda$  playing the role of string tension (the speed of light is set to unity). Extremals of this action, i.e., solutions of Eq. (42), minimize the volume, and so are called *minimal surfaces* in mathematical literature [5].

In this example the Hamilton-Jacobi equation (20) takes a particularly compact form (cf. Ref. [2, Ch. 7])

$$(44) \quad |\partial_q \wedge S| = \Lambda.$$

From now on, let us focus on the case  $D = 1$ , which describes the relativistic particle in the Euclidean spacetime. We will present two methods for finding the physical motions.

First, suppose that two points,  $q_0$  and  $q$ , lie on  $\gamma_{cl}$ , multiply Eq. (42) by  $|d\Gamma|$ , and integrate along  $\gamma_{cl}$  from  $q_0$  to  $q$ . The Fundamental theorem of calculus implies that

$$(45) \quad I_{\gamma}(q) - I_{\gamma}(q_0) = 0,$$

i.e.,  $I_{\gamma}$  is constant along a physical motion, and  $\gamma_{cl}$  are therefore a straight lines in  $\mathcal{C}$ ,

$$(46) \quad \gamma_{cl} = \{q = v\tau + q_0 \mid \tau \in \mathbb{R}\}$$

where  $q_0 \in \mathcal{C}$  and  $v$  is arbitrary constant vector.

Second method utilizes a family of solutions of the Hamilton-Jacobi equation (44), for example,

$$(47) \quad S(q; q_0) = \Lambda |q - q_0|.$$

According to Formula (22), derivative of  $S$  with respect to the parameters  $q_0$  yields conserved quantities

$$(48) \quad \partial_{q_0} S = -\Lambda \frac{q - q_0}{|q - q_0|}.$$

Physical motion are then obtained readily,

$$(49) \quad \gamma_{cl} = \left\{ q \mid \frac{q - q_0}{|q - q_0|} = v \right\},$$

where  $v$  is an arbitrary constant unit vector.

## 6. CONCLUSION AND OUTLOOK

In this article we elaborated on the formulation of classical field theory presented in [1, Ch. 3], which is based on the notion of partial observables, and on the Hamiltonian constraint. The latter is a function of configuration-space point and the generalized multivector-valued momentum. Starting from the variational principle of Section 2 we derived canonical equations of motion (13). We also deduced local Hamilton-Jacobi equation (20), which can be a useful tool to find the physical motions.

With three ensuing examples we showed how non-relativistic mechanics, scalar field theory, and string theory can be described in one unifying framework by appropriately selecting the Hamiltonian constraint. In particular, we noticed that equations of motion, and the continuity equation for the energy-momentum tensor (which reduces to the energy conservation equation in the case of non-relativistic mechanics) are in fact of the same origin in the Hamiltonian constraint formalism. Therefore, this formalism may be of interest even for theories that do not assume symmetry between time and space, or spacetime and fields.



Hamiltonian formalism is especially important when it comes to quantization. In particle mechanics, momentum is promoted to a differential operator, and the Schrödinger equation is postulated. What is the quantum operator corresponding to the multivector-valued generalized momentum of the Hamiltonian constraint approach? And what does the Schrödinger equation look like, once we know the classical Hamilton-Jacobi equation (20). Although these questions have not been addressed in general, let us note that there have been studies of quantization in the De Donder-Weyl Hamiltonian theory, where the quantization of momenta is based on generalized Poisson brackets, and a field-theoretic generalization of the Schrödinger equation is proposed that features Clifford-valued wave function [12, 14].

**Acknowledgement.** The author would like to thank Igor Kanatchikov for valuable discussions, and the following institutions for financial support: Grant Agency of the Czech Technical University in Prague, Grant SGS13/217/OHK4/3T/14, Czech Science Foundation (GAČR), Grant GA14-07983S, and Deutsche Forschungsgemeinschaft (DFG), Grant KL 256/54-1.

#### REFERENCES

- [1] C. Rovelli, *Quantum Gravity*, Cambridge Univ. Press (2004).
- [2] H. Kastrup, *Canonical theories of Lagrangian dynamical systems in physics*, Phys. Rep. **101** (1983), 1-167.
- [3] H. Rund, *The Hamilton-Jacobi Theory in the Calculus of Variations*, D. van Nostrand, Toronto (1966).
- [4] H. Weyl, Ann. Math. (2) **36** (1935) 607-629.
- [5] R. Osserman, *A Survey of Minimal Surfaces*, New York: Dover Publications (1986).
- [6] D. Hestenes and G. Sobczyk, *Clifford Algebra to Geometric Calculus*, Springer (1987).
- [7] C. Doran and A. Lasenby, *Geometric Algebra for Physicists*, Cambridge Univ. Press (2007).
- [8] V. Zatloukal, [arXiv:1504.08344] (2015).
- [9] G. E. Sobczyk, in *Clifford Algebras and their Applications in Mathematical Physics: Proceedings of Second Workshop held at Montpellier, France, 1989*, edited by A. Micali, R. Boudet, and J. Helmstetter, pp 279-292 (1992), available at [http://geocalc.clas.asu.edu/pdf-preAdobe8/SIMP\\_CAL.pdf](http://geocalc.clas.asu.edu/pdf-preAdobe8/SIMP_CAL.pdf).
- [10] T. Frankel, *The Geometry of Physics*, Cambridge Univ. Press, (2004).
- [11] T. De Donder, *Théorie invariante du calcul des variations*, Nouv. éd, Gauthiers-Villars, Paris (1935).
- [12] I. V. Kanatchikov, Rep. Math. Phys. **43** (1999) 157-170, [arXiv:hep-th/9810165].
- [13] B. Zwiebach, *A First Course in String Theory, 2nd Ed.*, Cambridge Univ. Press (2009).
- [14] I. V. Kanatchikov, [arXiv:1312.4518] (2013).