Cybernetics with Transdisciplinary Geometric Algebra

by Ahmad Eid - Friday, January 27, 2017

https://gacomputing.info/2017/01/27/ga-cybernetics/

> Hard sciences are successful because they deal with the soft problems; soft sciences are struggling because they deal with the hard problems. -- Heinz von Foerster

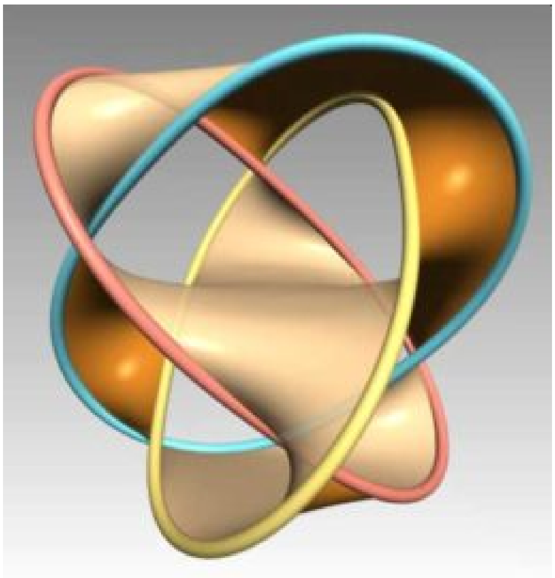
[dropcap]A[/dropcap] Cyberneticist, or cybernetician, is a person practicing Cybernetics: a transdisciplinary approach for exploring regulatory systems; their structures, constraints, and possibilities. Professor Eduardo Bayro-Corrochano is one such leading academic who uses Geometric Algebra to handle the diverse fields of theoretical knowledge and practical application he needs. Such fields include Robotics, Neural Computing, Computer Vision, and Lie Algebras. In this post, I interview Prof. Eduardo Bayro who tells us about how using GA in his work can simplify dealing with such diverse fields, and how can GA relate, generalize, and unify ideas from these fields together in his mind and the minds of his students.

Professor Eduardo Bayro started working on the applications of GA in the late 1990’s. His work is to stress the application of GA in cybernetics which typically involves Robotics, Neural Computing, Computer Vision, and Lie Algebras. Using GA-based geometric computing he tackled problems related to the building of intelligent machines, particularly Humanoids, to formulate algorithms for dealing with computer graphics curves, spheres, hyperplanes, and simple geometric tasks. Another goal is to build intelligent machines with real devices using geometric control, geometric perception and even quaternion wavelet transform, quaternion spike, or quantum quaternion neurons. Currently, he is preparing a new book series of 3 volumes on these topics. He is laying out the theory of future geometric intelligent robots and machines with powerful FPGA and Nvidia geometric accelerators.

Tell us about the benefits of using Geometric Algebra in Geometric Computing applications.

Geometric Algebra (GA) is a powerful geometric language for the modeling of the physics of systems. It is a highly intuitive language and provides many algebraic tools very useful to tackle complex equation manipulations. GA allows economical representations of transformations due to the GA multivector representation as opposite to analytic geometry or vector calculus. GA-based representations avoid redundant parameters and elegantly encapsulate the geometry of the problem. As a result due to its economical representations, GA is a well-suited framework for the development of real-time computations.

How can using GA simplify dealing with the diverse fields of interdisciplinary applications you are working on? What about its range of applicability?



"Borromean Rings," by Jarke J. van Wijk (Technische Universiteit Eindhoven). Image courtesy of Jarke J. van Wijk.

GA achieves the unification of many domains like multilinear algebra, multivariable analysis, conformal geometry, Lie group and Lie algebras, projective geometry, quantum mechanics and general relativity. GA allows:

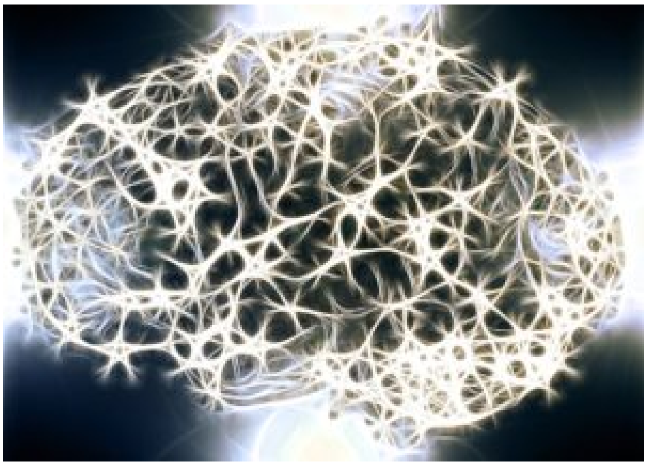
- Extending and completing algebraic operations on vectors and multi-vectors.
- Providing unified concepts for geometry and algebra.
- Providing explicit interpretation of geometric entities and operations on them in arbitrary dimensions with Euclidean and pseudo-Euclidean metrics.
- Acting as an advanced formalism for rotations and in general Lie groups in arbitrary dimensions.
- Acting as a metalanguage for high-level reasoning.

Because of these features, Geometric Algebra has a wide range of applications: graphics engineering, signal and image processing, computer vision, neural networks and quantum computing, machine learning, and robotics.

Tell us about the recent developments of applying GA in your research activities.

Recent achievements in geometric accelerators using packets like Versor (P. Colapinto) or Gaalop (Geometric Algebra Algorithms Optimizer, D. Hildenbrand) allow us with FPGA or NVidia to accelerate computations even more. In my lab, we develop and build intelligent machines like in medical robotics or humanoids. The development of GA algorithms follows: algorithmic modeling, using our own C++ programs or Versor, then simply the algorithms with Maple or Gaalop and accelerate them with either FPFGA or Nvidia. The results are quite impressive for real time applications and the visualization of computing in higher dimensions which in turn helps us to gain insight and achieve correctness in the prove of theorems and improve the performance of our algorithms. We are working now to develop in the next 3 years a single small chip for geometric computing using GA with a power superior as the Jetson TK1 Nvidia card.

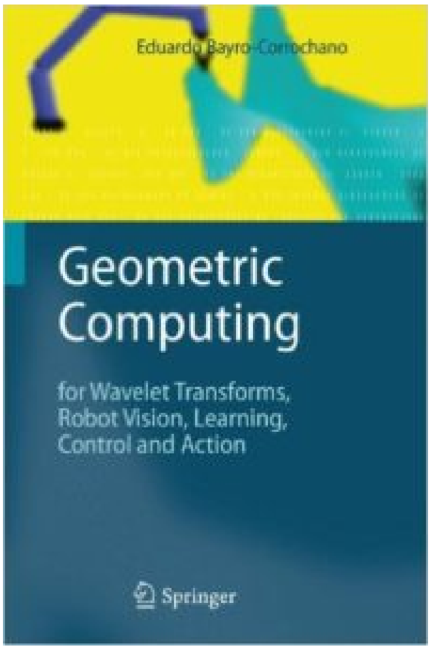
In your opinion, are there any relations between Neural Learning and Geometric Learning? How do you use GA as a tool for exploring such relations?



The tricky question is what should be a true geometric learning? We started by generalizing the McCulloch-Pitts neuron replacing its inner product via the geometric product, so the neural network weights are then multivectors and the learning rules are formulated using the gradient of multivectors as well. Our networks for classification via their kernels map the data using multivectors to a direct product of Hilbert spaces to separate multivector clusters, this helps a lot to find optimal hyperplanes. We have worked on geometric neural computing since 1995, in fact, we have generalized in GA most of the neural networks: i.e. multivector valued [SOM, NeuralGAS, MLP, ART], Clifford SVM and Quaternion Spike neural Network, currently we are developing the quaternion quantum Neural Network. This year we are also focusing on geometric high-order recurrent

Neural Networks.

Tell us about your books describing your valuable work and experience with GA, and the new books you are currently preparing.


Eduardo Bayro-Corrochano
Geometric Computing
for Wavelet Transforms, Robot Vision, Learning, Control and Action
Springer

I have been teaching since 1997 using the material of my work which was then published in two books:

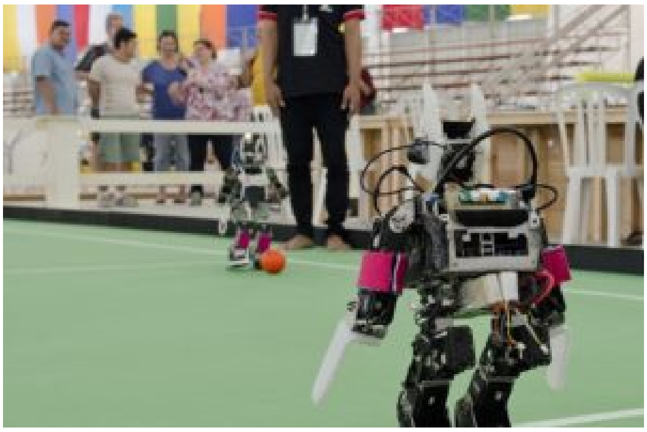
- Geometric Computing for Perception Action Systems. Eduardo Bayro-Corrochano, Springer-Verlag, April 2001.
- Geometric Computing: for Wavelet Transforms, Robot Vision, Learning, Control, and Action. Eduardo Bayro-Corrochano, Springer-Verlag, 2010.

I am preparing a more comprehensive series in three volumes:

- Vol. I: GA for computer vision and machine learning,
- Vol. II: GA for Robotics,
- Vol. III: GA for Signal Processing, Filtering, Transforms and Quantum Computing.

These three volumes are based on my first two books, of course, while including new exciting results. My books generally differ from many the current books on GA applications because my books cover computer vision, graphics engineering, neural networks, machine learning, robotics, integral transforms, and ultimately the endeavor of building intelligent machines with GA; they present real applications. The other books are more related to the foundations of GA and its restricted use to the develop algorithms in computer science and its sub-fields like computer graphics. That is why my continuous work during all these years is helping to be able to prepare these new three volumes, which I believe will certainly establish the true geometric computing for the future development of intelligent machines.

Tell us about your practical experience teaching applied Geometric Algebra. How do you overcome the challenges you face while using GA in teaching?



I call for studying and applying GA for future cybernetics in my lectures at different universities in Europe, North America (MIT) and Latin America. I have students with different backgrounds as in physics, mathematics, or engineering; some well prepared for some topics and less for others, so I start as in a fractal approach for conveying knowledge:

- We study 2D- and 3D-GA to become acquainted with the formulation of the geometric primitives and the use of different Clifford operations.
- Then we model the kinematics of points, lines and planes particularly to understand well screw theory and differential kinematics.
- Then I explain the basic of differential geometry like the Geometric Laplace-Beltrami Operator for applications in Euclidean and pseudo-Euclidean spaces.
- We study in depth Lie Groups and Lie algebras using spinors and versors for Euclidean, Affine, and 2D and 3D projective geometry.
- Then I explain projective geometry using Euclidean and Minkowski metrics for the projective plane and spaces and extensions of the projective space in R(3,3).

In this manner, the students have learned the most important ingredients to move the Conformal Geometric Algebra and high dimensional geometric algebras like G(6,3) and G(8,2). Thanks to our optimized routines for GA in C++ (with NVidia) and Maple (using the Clifford 14.0 package of R. Ablamowicz), we can prove theorems and develop real-time algorithms for intelligent machines.