IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 September 2022, accepted 6 October 2022, date of publication 21 October 2022, date of current version 1 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3216266

RESEARCH ARTICLE

Computing in the Conformal Space Objects, Incidence Relations, and Geometric Constrains for Applications in AI, GIS, Graphics, Robotics, and Human-Machine Interaction

EDUARDO JOSE BAYRO-CORROCHANO, (Senior Member, IEEE), G. ALTAMIRANO-ESCOBEDO, A. ORTIZ-GONZALEZ, V. FARIAS-MORENO, AND N. CHEL-PUC

Department Electrical Engineering and Computer Science, Cinvestav, Jalisco 45119, Mexico

Corresponding author: Eduardo Jose Bayro-Corrochano (eduardo.bayro@cinvestav.mx)

This work was supported by the Conacyt Project under Grant A1-S-10412.

ABSTRACT This article presents a revisited proposal for the formulation of objects and geometric relations and constraints in the conformal space. For modeling, graphics engineering, kinematics, and dynamics, the solution of problems using only points and lines; or the formulation of rigid motion (SE(3) using vectors calculus, matrix algebra, or calculus is indeed very awkward. In contrast, we use incidence algebra and conformal geometric algebra to effectively represent geometric objects and compute relations and constraints between geometric entities. In conformal geometric algebra, one can compute efficiently the linear transformations SO(3) and SE(3) of these geometric entities using rotors, translators, and motors. Since these operators and geometric entities have no redundant coefficients, they can be computed very fast. The authors present a new and complete set of equations using incidence algebra and conformal geometric algebra. The use of the proposed equations depends upon the applications. You can enclose certain objects with geometric shapes in your setting using points, lines, planes, circles, spheres, hyperplanes, and hyperspheres. Then, quadratic programming for optimization can be applied to find the minimal directed distance or a minimal path to be followed among many geometric objects. These methods and equations can be used to tackle a variety of problems in graphics, augmented virtual reality, GIS, Robotics, and Human-Machine Interaction. For real-time applications, the procedures and equations presented in this work can be used to develop efficient algorithms, which can be sped up using FPGA or CUDA (Nvidia).

INDEX TERMS Conformal geometric algebra, Clifford algebra, incidence algebra, graphics engineering, GIS, virtual reality, augmented virtual reality, medical robotics, robotics, human-machine interaction.

I. INTRODUCTION

This article presents a revisited proposal for the formulation of geometric objects and geometric relations in the conformal geometric algebra framework. This work presents a geometric approach and a new and complete set of equations using incidence algebra and conformal geometric algebra. This approach and equations can be used to solve many problems in graphic engineering, GIS, Robotics, and Human-Machine Interaction.

For real-time applications, the procedures and equations presented in this work, can be used to develop efficient algorithms, which can be sped up using FPGA or CUDA (Nvidia), see [1]. Next, we will be described related works.

A. ANALYSIS OF RELATED WORKS

In linear algebra, tensor calculus vector calculus, and quaternion algebra, for the modeling, graphics engineering,

The associate editor coordinating the review of this manuscript and approving it for publication was Santosh Kumar.

kinematics, and dynamics, the use of only points and lines as in [2] and [3] is very complicated mainly due to three reasons:

i) these mathematical systems induce a coordinate dependency;

ii) by a formulation using vectors, matrices, and tensors, one is detached from the geometry of the problem resulting in a loss of intuition leading to complicated algebraic equations;

iii) in these frameworks the computation utilizes redundant coefficients, as a result, the real-time computations are slow-down.

B. INCIDENCE ALGEBRA IN THE CONFORMAL SPACE

When we are dealing with problems in graphic engineering computer vision, robotics, or neural computing, an important question is in which metric space we should work. In this article, we are concerned with three well-understood space models:

i. Models for 2D and 3D spaces with a Euclidean metric: 2D and 3D are well suited to handle the algebra of directions in the plane and 3D physical space. 3D rotations are represented using rotors (isomorph to quaternions). You can model the kinematics of points, lines, and planes using G_3. Rotors can be used for interpolation in graphics and estimation of rotations of rigid bodies.

ii. Models for 4D spaces with non-Euclidean metric: If you are interested in linearizing a rigid motion transformation, you will need a homogeneous representation. For that, we should use geometric algebra for the 4D space. Here, it is more convenient to choose the motor algebra $G^{+}_{3,0,1}$. It is the algebra of Plucker lines, which can be used to model the kinematics of points, lines, and planes better than with G_3. Lines belong to the nonsingular Study 6D quadric and the motors to the 8D Klein quadric. In $G^{+}_{3,0,1}$, you can formulate a motor-based equation of motion for constant velocity wherein the exponent you use a bivector for twists. You can also use motors for interpolation of 3D rigid motion using 8D homogenous vectors in the Study manifold and estimate trajectories using the Motor Extended Filter (MEKF).

When you are dealing with problems of projective geometry like in computer vision or graphics engineering, again you need a homogeneous coordinate representation, so that the image plane becomes P^2 and the visual space P^3. To handle the so-called n-view geometry [4] based on tensor calculus and invariant theory, you require G_3, 1 (Minkowski metric) for the visual space and G_3 for the image plane. Note that the intrinsic camera parameters are modeled with an affine transformation within geometric algebra as part of the projective mapping via a projective split between the projective space and the image plane.

iii. Incidence algebra, an algebra of oriented subspaces, can be used in $G_{3,1}$ and G_3 to treat problems involving geometric constraints as in projective geometry, computer vision, and graphics engineering. In the seminal article of Hestenes, [5], projective geometry is formulated using Clifford algebra. Hestenes introduced the incidence algebra and the duality concept for the treatment of intersections and unions of points, lines, and planes by using the meet operation $J = A \vee B$ and the join operation $C = A \wedge B$ respectively. Hestenes showed that the roots of incidence algebra are the Dual Algebra [5] and proved classic theorems of projective geometry using incidence algebra. Later on, Lasenby and Bayro [6] used Hestenes' approach for applications in computer vision like projective reconstruction and invariant theory. In 1983 Havel [7] introduced the concept of *directed distance* to describe using a vector the distance between points, lines, and planes. in 2003 Bayro and Sobzcyk [8] formulated the directed distance and incidence algebra in the conformal geometric algebra framework. Lasenby et al. [9]published a report describing certain basic equations of incidence algebra in conformal algebra. After these pioneering works, other authors in Vinze [10], Gunn [11], and Hildebrand [12] presented methods for projective geometry and incidence algebra for graphics engineering. However, these last authors used in the description of incidence algebra the equations published by the above-cited pioneer works; they present a limited amount of incidence algebra-based relation equations, moreover they don't use the concept of directed distance algebra though.

iv. Conformal models: If you consider conformal transformations (angle preserving), conformal geometric algebra offers a non-Euclidean geometric algebra $G_{n,1}$ that includes in its multivector basis the null vectors, the origin, and the point at infinity. As a computational framework, it utilizes the powerful horosphere (the meet between a hyperplane and the null cone). Even though the computational framework uses a nonlinear representation for the geometric entities, one can recover the Euclidean metric. The basic geometric entity is the sphere and you can represent points, planes, lines, planes, circles, and spheres as vectors or in dual forms, the latter being useful to reduce the complexity of algebraic expressions, see [13].

C. INCIDENCE ALGEBRA IN THE CONFORMAL SPACE REVISITED

Since we detected that the authors have not completed developing the whole extension of the theory of incidence algebra and directed distance in the conformal geometric algebra framework, we decided the derive all the equations to treat the geometric relations and generation of constraints between points, lines, planes, circles and spheres using incidence algebra, directed distance in conformal geometric algebra $G_{4,1}$. For example, we have five geometric entities (points, lines, planes, circles, and spheres, so you can compute 28 combinations of intersections and direct distances. These equations are illustrated graphically, showing how these geometric relations take place. Since we cover all the possible cases, deriving all the necessary equations, the reader can understand the complexity of these geometric relations without missing any detail. Therefore practitioners can use easily these equations to tackle a variety of problems in projective geometry, computer vision, graphics engineering, interpolation, and tracking using EKF techniques as well.

D. ARTICLE ORGANIZATION

Section II presents the mathematical preliminaries of conformal geometric algebra and motor algebra. Section III the concept of directed distance is explained. Section IV describes the intersection of geometric entities. Section V shows the use of conformal geometric algebra and incidence algebra in graphics, robotics, and medical robotics. Finally, in section VI, the conclusion of this work is given.

II. CONFORMAL GEOMETRIC ALGEBRA

The reader can find a more detailed description of conformal geometric algebra in [14].

Conformal geometric algebra $G_{n+1,1}$ has an orthonormal vector basis $\{e_1, \ldots, e_n, e_{n+1}, e_{n+2}\}$ where $e_i^2 = 1,\ i = 1, \ldots, n,\ e_{n+1}^2 = 1,\ e_{n+2}^2 = -1,\ e_i \cdot e_{n+1} = e_i \cdot e_{n+2} = e_{n+1} \cdot e_{n+2} = 0,\ \ i = 1, \ldots, n$. In addition, the origin and the point at infinity $\{e_0, e_\infty\}$ (origin and point at infinity) are

$$e_0 = \frac{(e_{n+1} - e_{n+2})}{2}, \quad e_\infty = e_{n+1} + e_{n+2}, \tag{1}$$

These vectors are called null vectors, because $e_0^2 = e_\infty^2 = 0$ and $e_\infty \cdot e_0 = -1$. Their wedge product build the Minkowski plane $\boldsymbol{E} = e_\infty \wedge e_0$.

A. POINTS, LINES, PLANES AND SPHERES

A 3D point in conformal geometric algebra $G_{4,1}$ is represented as

$$x_c = x_e + \frac{1}{2}x_e^2 e_\infty + e_0. \tag{2}$$

The line as IPNS is formulated as

$$\boldsymbol{L} = \mathbf{n}I_E - e_\infty \boldsymbol{m} = (\boldsymbol{n} + e_\infty \boldsymbol{m}), \tag{3}$$

where $\boldsymbol{n}$ and $\boldsymbol{m}$ represent the orientation and the moment of the line respectively.

The plane in its IPNS representation is

$$\pi = \boldsymbol{n}I_E - e_\infty d = \mathbf{n} - e_\infty d; \tag{4}$$

the unit vector $\mathbf{n}$ and the scalar d stand for the orientation and the Hesse distance respectively.

In the IPNS representation, the sphere equation is

$$s = p_c - \frac{1}{2}\rho^2 e_\infty = p_e + (\frac{p_e^2 - \rho^2}{2})e_\infty + e_0. \tag{5}$$

The constraint for a point lying on a sphere is given by

$$x_c \wedge s^* = x_c \wedge (s \cdot I_c) = 0. \tag{6}$$

The equations of the circle and the sphere in the Outer Product Null Space (OPNS) representation are computed using three or four points respectively

$$\begin{aligned} z^* &= x_{c_1} \wedge x_{c_2} \wedge x_{c_3} \\ s^* &= x_{c_1} \wedge x_{c_2} \wedge x_{c_3} \wedge x_{c_4}. \end{aligned} \tag{7}$$

Replacing one of these points with the point at infinity in the equations (7), one obtains the line and plane equations in the OPNS representation

$$\begin{aligned} l^* &= x_{c_1} \wedge x_{c_2} e_\infty, \\ \pi^* &= x_{c_3} \wedge x_{c_1} \wedge x_{c_2} \wedge e_\infty + \\ &\quad + x_{e_3} \wedge x_{e_1} \wedge x_{e_2} \wedge e_\infty + ((x_{e_3} - x_{e_1}) \wedge (x_{e_2} - x_{e_1}))\boldsymbol{E}. \end{aligned}$$

B. RIGID TRANSFORMATIONS

Given a geometric identity O, the Translator operator for a translation is computed by two successive reflections with parallel planes π_1 and π_2 as follows,

$$O' = \underbrace{(\pi_2 \pi_1)}_{T_a} O \underbrace{\left(\pi_1^{-1}\pi_2^{-1}\right)}_{\tilde{T}_a}, \quad T_a = 1 + \frac{1}{2}ae_\infty = e^{-\frac{a}{2}e_\infty},$$

with $a = 2dn$, d the distance of translation, and n the direction of translation.

On the other hand, a rotation operator can be computed as the composition of two reflections of non-parallel planes π_1 and π_2, which cross the origin. Then, a rotation is defined by

$$O' = \underbrace{(\pi_2 \pi_1)}_{R_\theta} O \underbrace{\left(\pi_1^{-1}\pi_2^{-1}\right)}_{\tilde{R}_\theta}. \tag{8}$$

The geometric product of the normal planes n_1 and n_2 yields a rotor

$$R_\theta = n_2 n_1 = \cos\left(\theta/2\right) - \sin\left(\theta/2\right)\boldsymbol{n} = e^{-\theta \boldsymbol{n}/2}, \tag{9}$$

where unit bivector $\boldsymbol{n} = n_1 \wedge n_2$, and θ corresponds to twice the angle between the planes π_1 and π_2.

Finally, the operator for screw motion called Motor consist of a composition of a Translator and a Rotor

$$\begin{aligned} \boldsymbol{M} &= \cos\left(\frac{\theta + e_\infty d}{2}\right) + \boldsymbol{L}\sin\left(\frac{\theta + e_\infty d}{2}\right) \\ &= \boldsymbol{T}_s \boldsymbol{R}_s = \boldsymbol{R}_s + e_\infty \boldsymbol{R}'_s = \exp^{-\theta \boldsymbol{L}}, \end{aligned} \tag{10}$$

where $\boldsymbol{L} = \boldsymbol{n} + e_\infty \boldsymbol{m}$ stands for the screw axes line. The motor transformation for any geometric entity $O \in G_{4,1}$ is given by

$$O' = \boldsymbol{T}_s \boldsymbol{R}_s O \tilde{\boldsymbol{R}}_s \tilde{\boldsymbol{T}}_s = \boldsymbol{M} O \tilde{\boldsymbol{M}}, \tag{11}$$

see [14] for an introduction to conformal geometric algebra.

III. DIRECTED DISTANCE

The Euclidean distance is computed as the norm of the difference between two vectors or points. The directed distance is the vector between two vectors, i.e. indicates with its norm the Euclidean distance and as a vector has an orientation. Points are represented as vectors.

The Euclidean distance is the distance between two points, for three points in general position, (lying on a plane), one could compute the Euclidean distance w.r.t. to their centroid. Similarly, in the case of 4 or n points in a general position. As we show in this work, using Incidence Algebra, one utilizes the directed distance to compute the minimal distance vector between objects, i.e. between a plane and a line, a volume and a plane or a plane and another plane, etc.

A. POINT TO POINT

The IPNS representation of two points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is given by

$$x_{c_1} = \boldsymbol{x}_1 + \frac{1}{2}\boldsymbol{x}_1^2 e_\infty + e_0, \tag{12}$$

$$x_{c_2} = \boldsymbol{x}_2 + \frac{1}{2}\boldsymbol{x}_2^2 e_\infty + e_0, \tag{13}$$

$\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are 3D vectors spanned by the vector basis e_1, e_2 and e_3. The formulas for the Euclidean distance and directed distance from x_{c_1} to x_{c_2} are

$$d = \sqrt{-2(x_{c_1} \cdot x_{c_2})} \tag{14}$$

$$\boldsymbol{d} = \boldsymbol{x}_2 - \boldsymbol{x}_1. \tag{15}$$

B. POINT TO LINE: METHOD 1

If the point $\boldsymbol{q}$ lies on the line L. The parameterization of the line L is given as $\boldsymbol{x}(\alpha) = \boldsymbol{q} + \alpha\boldsymbol{n}$; where alpha is a real number and $\boldsymbol{n}$ is a parallel vector to the line L. As the parameter, α sweeps through all real numbers, the point $\boldsymbol{x}(\alpha)$ sweeps out the line L. The IPNS representation of $\boldsymbol{x}(\alpha)$ is given by

$$x_c(\alpha) = \boldsymbol{x}(\alpha) + \frac{1}{2}\boldsymbol{x}(\alpha)^2 e_\infty + e_0. \tag{16}$$

Therefore, the Euclidean distance between the point $\boldsymbol{p}$ and the point $\boldsymbol{x}(\alpha)$ is

$$d(\alpha) = \sqrt{-2p_c \cdot x_c(\alpha)}. \tag{17}$$

Taking the derivative with respect to α and setting the result equal to zero, we find the value of α that minimizes the square of the Euclidean distance:

$$\begin{aligned}\alpha_{\min} &= \frac{n_x(p_x - q_x) + n_y(p_y - q_y) + n_z(p_z - q_z)}{n_x^2 + n_y^2 + n_z^2} \\ &= \frac{\boldsymbol{m} \cdot (\boldsymbol{p} - \boldsymbol{q})}{\boldsymbol{n}^2}.\end{aligned} \tag{18}$$

Consequently the directed distance from the point $\boldsymbol{p}$ to the line L is equal to the directed distance from the point $\boldsymbol{p}$ to the point $\boldsymbol{x}_{\min} = \boldsymbol{x}(\alpha_{\min})$

$$\boldsymbol{d} = \boldsymbol{x}_{\min} - \boldsymbol{p}. \tag{19}$$

The Euclidean distance can be obtained by taking the magnitude of $\boldsymbol{d}$ or using the following expression in terms of the inner product of p_c and $x_{c_{\min}}$

$$d = \sqrt{-2p_c \cdot x_{c_{\min}}}. \tag{20}$$

C. POINT TO LINE: METHOD 2

A line L in the OPNS representation can be expressed as

$$L^* = e_\infty \wedge x_{c_1} \wedge x_{c_2}, \tag{21}$$

the IPNS representation is obtained by taking its dual

$$L = (L^*)^* = (e_\infty \wedge x_{c_1} \wedge x_{c_2})I_c. \tag{22}$$

The orthogonal projection of a point p_c onto the line L is found with the rejection formula

$$P_{A_r}^{\perp} = (v \wedge A_r)A_r^{-1}. \tag{23}$$

In our case $v = p_c$ and $A_r = L$. So the orthogonal projection is given by

$$P_L^{\perp} = (p_c \wedge L)L^{-1}. \tag{24}$$

For simplicity let us denote $P_L^{\perp}$ simply as p_L. To extract a Euclidean point from p_L we use the following formula

$$\boldsymbol{p}_L = (p_L \wedge E)E \tag{25}$$

Now is straightforward to compute the directed distance and the Euclidean distance from the point $\boldsymbol{p}$ to the line L:

$$\boldsymbol{d} = \boldsymbol{p}_L - \boldsymbol{p} \tag{26}$$

$$d = ||\boldsymbol{d}||. \tag{27}$$

D. POINT TO SPHERE

A sphere s with radius ρ, its center $\boldsymbol{c}$ and a point $\boldsymbol{x}$ using the IPNS representation are given as follows

$$s = \boldsymbol{c} + \frac{1}{2}(\boldsymbol{c}^2 - \rho^2)e_\infty + e_0, \tag{28}$$

$$c_c = \boldsymbol{c} + \frac{1}{2}\boldsymbol{c}^2 e_\infty + e_0, \tag{29}$$

$$x_c = \boldsymbol{x} + \frac{1}{2}\boldsymbol{x}^2 e_\infty + e_0, \tag{30}$$

The Euclidean distance and directed distance from the point x_c to the sphere s is given by the following formulas

$$d = \sqrt{-2(x_c \cdot c_c)} - \rho \tag{31}$$

$$\boldsymbol{d} = d\boldsymbol{n} \tag{32}$$

where $\boldsymbol{n} = \frac{\boldsymbol{c}-\boldsymbol{x}}{||\boldsymbol{c}-\boldsymbol{x}||}$, is a unit vector pointing from $\boldsymbol{x}$ to $\boldsymbol{c}$.

E. POINT TO PLANE

A point x_c and a plane π with normal vector $\boldsymbol{n}$ and distance d_π can be expressed as

$$x_c = \boldsymbol{x} + \frac{1}{2}\boldsymbol{x}^2 e_\infty + e_0, \tag{33}$$

$$\pi = \boldsymbol{n} + d_\pi e_\infty. \tag{34}$$

The Euclidean distance and directed distance from the point to the plane are

$$d = x_c \cdot \pi, \tag{35}$$

$$\boldsymbol{d} = -d\boldsymbol{n}, \tag{36}$$

where d_π is the Hesse distance of the plane. The Euclidean distance d has a sign according to

- $d > 0$: $\boldsymbol{x}$ is in the direction of the normal $\boldsymbol{n}$
- $d = 0$: $\boldsymbol{x}$ is on the plane
- $d < 0$: $\boldsymbol{x}$ is not in the direction of the normal $\boldsymbol{n}$.

We have taken into account the sign of the Euclidean distance within the expression of the directed distance by adding the minus sign.

F. LINE TO LINE

Let's take the parametrization of two lines L_1 and L_2

$$\boldsymbol{x}_1(\alpha_1) = q_1 + \alpha_1 \boldsymbol{n_1}, \tag{37}$$
$$\boldsymbol{x}_2(\alpha_2) = q_2 + \alpha_2 \boldsymbol{n_2}. \tag{38}$$

Their IPNS representation is given by

$$x_{c_1}(\alpha_1) = \boldsymbol{x}_1(\alpha_1) + \frac{1}{2}\boldsymbol{x}_1(\alpha_1)^2 e_\infty + e_0, \tag{39}$$
$$x_{c_2}(\alpha_2) = \boldsymbol{x}_2(\alpha_2) + \frac{1}{2}\boldsymbol{x}_2(\alpha_2)^2 e_\infty + e_0. \tag{40}$$

As the parameters α_1 and α_2 sweep through all real numbers,the points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ sweep out the lines L_1 and L_2. The Euclidean distance between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ can be expressed as

$$d(\alpha_1, \alpha_2) = \sqrt{-2x_{c_1}(\alpha_1) \cdot x_{c_2}(\alpha_2)}. \tag{41}$$

To find the critical points we need to take partials and set them equal to zero. Then we need to solve the following equations simultaneously.

$$\frac{\partial d(\alpha_1, \alpha_2)}{\partial \alpha_1} = 0, \tag{42}$$
$$\frac{\partial d(\alpha_1, \alpha_2)}{\partial \alpha_2} = 0. \tag{43}$$

We obtained long expressions for α_1 and α_2 as follows

$$\begin{aligned}\alpha_1 = {} & (n_{1_x} n_{2_x} n_{2_y} q_{1_y} - n_{1_x} n_{2_x} n_{2_y} q_{2_y} + n_{1_x} n_{2_x} n_{2_z} q_{1_z} \\ & - n_{1_x} n_{2_x} n_{2_z} q_{2_z} - n_{1_x} n_{2_y}^2 q_{1_x} + n_{1_x} n_{2_y}^2 q_{2_x} - n_{1_x} n_{2_z}^2 q_{1_x} \\ & + n_{1_x} n_{2_z}^2 q_{2_x} - n_{1_y} n_{2_x}^2 q_{1_y} + n_{1_y} n_{2_x}^2 q_{2_y} \\ & + n_{1_y} n_{2_x} n_{2_y} q_{1_x} - n_{1_y} n_{2_x} n_{2_y} q_{2_x} + n_{1_y} n_{2_y} n_{2_z} q_{1_z} \\ & - n_{1_y} n_{2_y} n_{2_z} q_{2_z} - n_{1_y} n_{2_z}^2 q_{1_y} + n_{1_y} n_{2_z}^2 q_{2_y} - n_{1_z} n_{2_x}^2 q_{1_z} \\ & + n_{1_z} n_{2_x}^2 q_{2_z} + n_{1_z} n_{2_x} n_{2_z} q_{1_x} - n_{1_z} n_{2_x} n_{2_z} q_{2_x} \\ & - n_{1_z} n_{2_y}^2 q_{1_z} \\ & + n_{1_z} n_{2_y}^2 q_{2_z} + n_{1_z} n_{2_y} n_{2_z} q_{1_y} - n_{1_z} n_{2_y} n_{2_z} q_{2_y}) / (n_{1_x}^2 n_{2_y}^2 \\ & + n_{1_x}^2 n_{2_z}^2 - 2 n_{1_x} n_{1_y} n_{2_x} n_{2_y} - 2 n_{1_x} n_{1_z} n_{2_x} n_{2_z} + n_{1_y}^2 n_{2_x}^2 \\ & + n_{1_y}^2 n_{2_z}^2 - 2 n_{1_y} n_{1_z} n_{2_y} n_{2_z} + n_{1_z}^2 n_{2_x}^2 + n_{1_z}^2 n_{2_y}^2), \end{aligned} \tag{44}$$

the expression for α_2 is similar.

G. LINE TO LINE: GEOMETRIC METHOD

For a line L in IPNS representation, its direction can be computed as follows

$$\mathbf{n} = \langle L I_E \rangle_1 \tag{45}$$

For two skew lines L_1 and L_2 with directions $\mathbf{n}_1$ and $\mathbf{n}_2$, the unit vector normal to both lines is given by

$$\hat{\mathbf{n}} = \frac{\mathbf{n}_1 \times \mathbf{n}_2}{\| \mathbf{n}_1 \times \mathbf{n}_2 \|} \tag{46}$$

Let $\mathbf{x}_{1_e}$ and $\mathbf{x}_{2_e}$ be two points lying in L_1 and L_2, respectively. Then the shortest directed distance and the shortest Euclidean distance between the two skew lines are expressed as follows

$$\mathbf{d} = (\mathbf{x}_{2_e} - \mathbf{x}_{1_e}) \cdot \hat{\mathbf{n}}, \tag{47}$$

$$d = \| \mathbf{d} \| . \tag{48}$$

The points $\mathbf{x}_{1_e}$ and $\mathbf{x}_{2_e}$ can be obtained by orthogonally projecting an arbitrary point x onto the lines L_1 and L_2, respectively.

H. LINE TO PLANE

Here we have two cases:

- The line intersects the plane or the line lies on it, then the distance is equal to zero.
- The line lies on a plane that is parallel to the other plane. In this case, we only need to compute the distance and directed distance between a point that lies on the line and the other plane as explained in section III-D.

I. SPHERE TO LINE

Let d_{cL} and $\boldsymbol{d}_{cL}$ denote the Euclidean distance and the directed distance from the center of the sphere to the line, respectively. The Euclidean distance and the directed distance from the surface of the sphere to the line are

$$d_{sL} = d_{cL} - \rho, \tag{49}$$
$$\boldsymbol{d}_{sL} = d_{sL} \boldsymbol{n}_{cL} \tag{50}$$

where $\boldsymbol{n}_{cL}$ is the unit vector pointing from the center of the sphere to the line.

J. SPHERE TO PLANE: METHOD 1

The Euclidean distance from the center of a sphere to a plane is computed via their scalar (inner) product

$$d_{c,\pi} = \pi \cdot s. \tag{51}$$

This distance $d_{c,\pi}$ has a sign according to

- $d_{c,\pi} > 0$: $\boldsymbol{c}$ is in the direction of the normal $\boldsymbol{n}$
- $d_{c,\pi} = 0$: $\boldsymbol{c}$ is on the plane
- $d_{c,\pi} < 0$: $\boldsymbol{x}$ is not in the direction of the normal $\boldsymbol{n}$. Therefore the distance between the plane and the surface of the sphere with radius ρ is given by

$$d = |d_{c,\pi}| - \rho, \tag{52}$$

provided that $|d_{c,\pi}| > \rho$.

Additionally, the directed distance can be expressed as follows

$$\boldsymbol{d} = d\boldsymbol{n}, \tag{53}$$

where the direction of the directed distance is determined by the direction of the normal of the plane.

K. SPHERE TO PLANE: METHOD 2

Another method is computing the distance between the center of the sphere and the plane minus the radius given by

$$d = c_c \cdot \pi - \text{sign}(c_c \cdot \pi)\rho, \tag{54}$$

and the directed distance from the center point of the sphere s and the plane π can be obtained from the following formula

$$\boldsymbol{d} = -d\boldsymbol{n}. \tag{55}$$

The difference between the first and second methods is that the latter conserves the sign convention in d while the former sacrifices it by introducing the absolute value.

L. SPHERE TO SPHERE

The inner product of two spheres s_1 and s_2 is given by

$$s_1 \cdot s_2 = \frac{(\rho_1^2 + \rho_2^2) - (\boldsymbol{c}_2 - \boldsymbol{c}_1)^2}{2}. \tag{56}$$

This, the square of the Euclidean distance between the centers of the spheres is given by:

$$d_{c_1,c_2} = \sqrt{\rho_1^2 + \rho_2^2 - 2(s_1 \cdot s_2)}. \tag{57}$$

Thus the Euclidean distance between the surfaces of the spheres can be expressed as

$$d_{s_1,s_2} = \sqrt{\rho_1^2 + \rho_2^2 - 2(s_1 \cdot s_2)} - (\rho_1 + \rho_2), \tag{58}$$

and the directed distance is finally

$$\boldsymbol{d}_{s_1,s_2} = d_{s_1,s_2}\boldsymbol{n}_{c_1c_2}. \tag{59}$$

where $\boldsymbol{n}_{c_1c_2}$ is the unit vector pointing from c_1 to c_2.

M. PLANE TO PLANE

1) ANGLE BETWEEN PLANES

The angle between two planes π_1 and π_2 with normal vectors $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ can be obtained from their inner product

$$\cos(\theta) = \pi_1 \cdot \pi_2. \tag{60}$$

If the angle θ equals 0 or π, then the planes are parallel. In this case, we can take only one point that lies on a plane. Then the problem reduces to computing the distance and directed distance from a point to a plane as explained in section III-D.

2) THE INTERSECTION OF TWO PLANES

The intersecting line L between two planes π_1 and π_2 is obtained using the outer product.

$$L = \pi_1 \wedge \pi_2. \tag{61}$$

Depending on the result, there are two possible outcomes:

- The planes are not parallel, in which case L results in a bivector that represents the line intersecting both planes.
- The planes are parallel, in this case, L returns 0.

N. CIRCLE TO POINT

We have two cases

1) The point and the circle lie in the same plane.
2) The point does not lie in the circle's plane.

1) CASE 1: THE POINT AND THE CIRCLE LIE IN THE SAME PLANE

A circle z^* can be constructed using three conformal points

$$z^* = x_1 \wedge x_2 \wedge x_3. \tag{62}$$

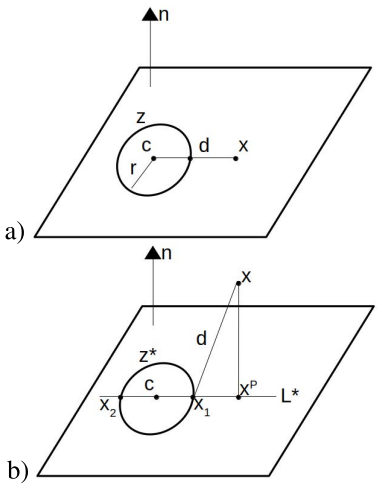


FIGURE 1. a) The point x and the circle z lie in the same plane. b) The point x and the circle z lie in different planes.

To find the radius r of the circle, a sphere s which has the circle as an equator can be constructed

$$s = \frac{z^*}{z^* \wedge e_\infty}, \tag{63}$$

then, the radius can be found using the inner product

$$r^2 = s \cdot s \tag{64}$$

The center c of the circle is computed by applying a sandwich product

$$c = -\frac{1}{2} z^* e_\infty z^*. \tag{65}$$

Therefore, in this case, the problem reduces to finding the distance between the center c and the point x

$$d_{z,x} = \sqrt{-2(c \cdot x)} - r. \tag{66}$$

Additionally the directed distance can be expressed as

$$\boldsymbol{d}_{z,x} = d_{z,x}\boldsymbol{n}_{cx}, \tag{67}$$

where $\boldsymbol{n}_{cx} = \frac{\boldsymbol{x}-\boldsymbol{c}}{||\boldsymbol{x}-\boldsymbol{c}||}$ is the unit vector from $\boldsymbol{c}$ to $\boldsymbol{x}$.

2) CASE 2: THE POINT AND THE CIRCLE LIE IN DIFFERENT PLANES

Using a plane

$$\phi = z^* \wedge e_\infty \tag{68}$$

and the nearest point on the circle, which can be computed reflecting that point in the plane ϕ. The line L^* is computed through the projected point x^p and the circle center,

$$L^* = x^p \wedge c \wedge e_\infty. \tag{69}$$

The nearest point is then computed by comparing the distance

$$(x_1, x_2) = L^* \vee z^* \tag{70}$$

$$x = argmin(\sqrt{-2(x_1 \cdot x)}, \sqrt{-2(x_2 \cdot x)}). \tag{71}$$

O. CIRCLE TO CIRCLE

In this case, the problem reduces to finding the distance and directed distance from the first circle center to the second circle center. This is similar to the distance point to point, see section III-A.

P. CIRCLE TO PLANE

We have some cases:

1) The circle lies in the plane, hence the distance is zero.
2) The plane where the circle lies and the plane is parallel or the circle does not lie in the plane, hence the distance reduces by computing the distance from the circle center to the plane (section III-D).

Q. CIRCLE TO LINE

The problem reduces to finding the distance and directed distance of the center point of a circle and a line as explained in section III-B.

R. SPHERE TO CIRCLE

After we obtain the center of the circle by using the sandwich product $c = -(1/2)ze_\infty z$ this problem reduces to computing the distance and directed distance from a point (the center of the circle) and a sphere (section III-C).

IV. INTERSECTION OF GEOMETRIC ENTITIES

Intersections can be performed using the meet operator. Suppose that we want to intersect the objects O_r (an r-blade) and O_s (an s-blade) these objects belong to the OPNS representation, if X lies on the intersection of O_r and O_s then

$$X \wedge O_r = 0 \quad \text{and} \quad X \wedge O_s = 0 \tag{72}$$

which can be shown to be equivalent to

$$X \wedge \{[\langle O_r O_s \rangle_{2n-r-s}]I_n\} = 0. \tag{73}$$

Therefore, the meet is defined as follows

$$\mathscr{M} = O_r \vee O_s = [\langle O_r O_s \rangle_{2n-r-s}]^* \tag{74}$$

where $[\cdot]^*$ denotes multiplication by the pseudoscalar I_n, n is the dimension of the space, in this case $n = 5$, and $\langle X \rangle_j$ denotes the extraction of the $j-$ grade component from X. The virtue of this formula is that we can intersect objects in a general manner regardless of the nature of our objects (spheres, circles, planes, and lines).
With few exceptions, the calculation of $\mathscr{M}^2$ will serve as a discriminator of all possible intersection cases. These cases are summarized in the following tables:

TABLE 1. Euclidean distance d and directed distance d involving several geometric entities described in conformal geometric algebra.

Condition	$s_1^* \vee s_2^*$ or $s^* \vee \pi^*$	$s^* \vee z^*$ or $s^* \vee L^*$	$\pi_1^* \vee \pi_2^*$
$\mathscr{M}^2 > 0$	Intersect in a circle	2 points of intersection	Intersect in a line
$\mathscr{M}^2 = 0$	1 point of intersection	1 point of intersection	Parallel planes without intersection
$\mathscr{M}^2 < 0$	No intersection	0 points of intersection	-

We can extract A and B from $A \wedge B$ using projectors. Let us consider the 2-blade $T = A \wedge B$. It is useful to define

$$F = \frac{1}{\beta} A \wedge B \tag{75}$$

Condition	$\pi^* \vee z^*$	$\pi^* \vee L^*$	$z_1^* \vee z_2^*$
$\mathscr{M}^2 > 0$	2 points of intersection	1 point of intersection	No intersection
$\mathscr{M}^2 = 0$	1 point of intersection	No intersection	1 point of intersection
$\mathscr{M}^2 < 0$	0 points of intersection	-	No intersection
$\mathscr{M} = 0$	The circle lies in the plane	The line lies in the plane	2 points of intersection

where $\beta > 0$ and $\beta^2 = T^2$, so that $F^2 = 1$ if $\beta^2 \neq 0$. Two projection operators can be defined in terms of F

$$P = \frac{1}{2}(1 + F) \tag{76}$$

$$\tilde{P} = \frac{1}{2}(1 - F) \tag{77}$$

where $\tilde{P}$ denotes the normal reversion operation applied to P. We can extract the two points A and B from $A \wedge B$ via

$$A = -\tilde{P}[(A \wedge B) \cdot e_\infty] \equiv -\tilde{P}[(A \wedge B) \cdot e_\infty]P \tag{78}$$

$$B = \tilde{P}[(A \wedge B) \cdot e_\infty] \equiv P[(A \wedge B) \cdot e_\infty]\tilde{P}. \tag{79}$$

In some cases, the above expressions will be used to obtain the intersection points from $\mathscr{M}$.

There are two cases in which we have to apply the above formulas differently when the intersecting objects lie in the same plane

- z_1^* and z_2^*
- L^* and z^*.

In both cases, we replace one circle with a sphere s which has the circle as an equator. This sphere can be constructed as follows

$$s = \frac{z^*}{z^* \wedge e_\infty}, \tag{80}$$

and the dual sphere can be calculated using $s^* = sI_c$. Therefore these two cases reduce to

- z_1^* and s_2^*
- L^* and z^*.

Now the intersection points can be computed using the corresponding formulas for these new cases.

In the case of a line-line intersection before the calculation of the meet, a test for checking if the lines are parallel is applied. If this condition is satisfied, they intersect at the point at infinity, so no further analysis related to the intersection is needed. For a line L in IPNS representation, its direction can be computed as follows

$$\mathbf{n} = \langle LI_E \rangle_1. \tag{81}$$

Therefore, two lines L_1 and L_2 are parallel if the following condition is satisfied

$$\mathbf{n}_1 \wedge \mathbf{n}_2 = 0. \tag{82}$$

TABLE 2. Euclidean distance d and directed distance d involving several geometric entities described in conformal geometric algebra.

Condition	$z^* \vee L^*$	$L_1^* \vee L_2^*$
$\mathscr{M}^2 > 0$	The line does not pass through the circle	-
$\mathscr{M}^2 = 0$	The circle and the line have one intersection	-
$\mathscr{M}^2 < 0$	The line passes through the circle	-
$\mathscr{M} = 0$	The circle and the line have two intersections	The lines intersect
$\mathscr{M} \propto e_\infty$	-	The lines do not intersect

Intersection objects			
$s_1^* \vee s_2^*$ or $s^* \vee \pi^*$	$s^* \vee z^*$, $s^* \vee L^*$ or $\pi^* \vee z^*$	$\pi_1^* \vee \pi_2^*$	$\pi^* \vee z^*$
For $\mathscr{M}^2 > 0$, $\mathscr{M}^2 < 0$ or $\mathscr{M}^2 = 0$: $z^* = \mathscr{M}$ $z = z^* I_c$ $\rho^2 = z^2$ $c_c = z e_\infty z$	For $\mathscr{M}^2 > 0$: $x_{1_c} = -\bar{P}[\mathscr{M} \cdot e_\infty]P$ $x_{2_c} = P[\mathscr{M} \cdot e_\infty]\bar{P}$ For $\mathscr{M}^2 = 0$: $x_c = \mathscr{M} e_\infty \mathscr{M}$	For $\mathscr{M}^2 > 0$: $L^* = \mathscr{M}$	For $\mathscr{M} = 0$: The intersection is z.

A. CIRCLE-CIRCLE INTERSECTION

For intersections between circles see Figures 2, 3, 4 and 5. We construct circles using the following Euclidean points:

$$
\begin{aligned}
x_{e_1} &= e_1, \quad x_{e_2} = -e_1, \quad x_{e_3} = e_2, \\
x_{e_4} &= e_1, \quad x_{e_5} = 4\,e_1, \quad x_{e_6} = 3\,e_1 + e_3, \\
x_{e_7} &= e_1, \quad x_{e_8} = 3\,e_1, \quad x_{e_9} = 2\,e_1 + e_2, \\
x_{e_{10}} &= e_2, \quad x_{e_{11}} = -e_2, \quad x_{e_{12}} = e_3, \\
x_{e_{13}} &= 0.5\,e_1 \quad x_{e_{14}} = 2.5\,e_1, \quad x_{e_{15}} = 1.5\,e_1 + e_2, \\
x_{e_{16}} &= 1.5\,e_1, \quad x_{e_{17}} = 3.5\,e_1, \quad x_{e_{18}} = 2.5\,e_1 + e_2, \\
x_{e_{19}} &= 1.5\,e_1, \quad x_{e_{20}} = 3.5\,e_1, \quad x_{e_{21}} = 2.5\,e_1 + e_3.
\end{aligned}
$$

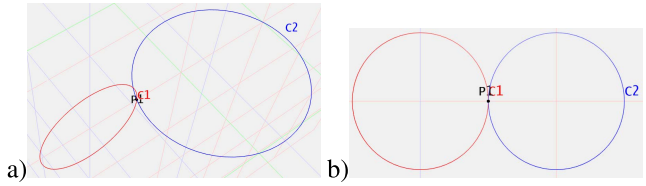


FIGURE 2. a) Intersection between z_1^* and z_2^*. The circles lie in different planes and have one point of intersection. b) Intersection between z_1^* and z_3^*. The circles lie in the same plane and have one point of intersection.

Intersection objects		
$\pi^* \vee L^*$	$A^* \vee z_2^*$ with $A^* = L^*$ or z_1^*	$L_1^* \vee L_2^*$
For $\mathscr{M}^2 > 0$: $\boldsymbol{x} = (\mathscr{M} \wedge e_o) \cdot E$ For $\mathscr{M} = 0$: The intersection is L.	For $\mathscr{M} = 0$: $\mathscr{M}_2 = A^* \vee (z_2^* \wedge e_\infty)$ $x_{1_c} = -\bar{P}[\mathscr{M}_2 \cdot e_\infty]P$ $x_{2_c} = P[\mathscr{M}_2 \cdot e_\infty]\bar{P}$ For $\mathscr{M}^2 = 0$: $x_c = \mathscr{M}$	For $\mathscr{M} = 0$: $L_1^{*\prime} = L_2^* L_1^* L_2^*$, $L_1^{*\prime\prime} = L_1^* - L_1^{*\prime}$ $X' = L_1^{*\prime\prime} X L_1^{*\prime\prime}$, $X'' = \frac{1}{2}(X + X')$ $X''' = L_2^* X'' L_2^*$, $P' = \frac{1}{2}(X'' + X''')$ $P = \frac{-(P' e_\infty P')}{2(P' \cdot e_\infty)^2}$, X is an arbitrary point. Simplest case: $X = e_o$ $\boldsymbol{P} = (P \wedge e_o) \cdot E$

The circles before normalization are given by:

$$
\begin{aligned}
z_1^* &= x_{c_1} \wedge x_{c_2} \wedge x_{c_3}, \quad z_2^* = x_{c_4} \wedge x_{c_5} \wedge x_{c_6}, \\
z_3^* &= x_{c_7} \wedge x_{c_8} \wedge x_{c_9}, \quad z_4^* = x_{c_{10}} \wedge x_{c_{11}} \wedge x_{c_{12}}, \\
z_5^* &= x_{c_{13}} \wedge x_{c_{14}} \wedge x_{c_{15}}, \\
z_6^* &= x_{c_{16}} \wedge x_{c_{17}} \wedge x_{c_{18}}, \quad z_7^* = x_{c_{19}} \wedge x_{c_{20}} \wedge x_{c_{21}}.
\end{aligned}
$$

B. CIRCLE-LINE INTERSECTION

For intersections between circles and lines see Figures 6, 7 and 8.a. We construct a circle and different lines using the following Euclidean points:

$$
x_{e_1} = e_1, \quad x_{e_2} = 3\,e_1, \quad x_{e_3} = 2\,e_1 + e_2,
$$

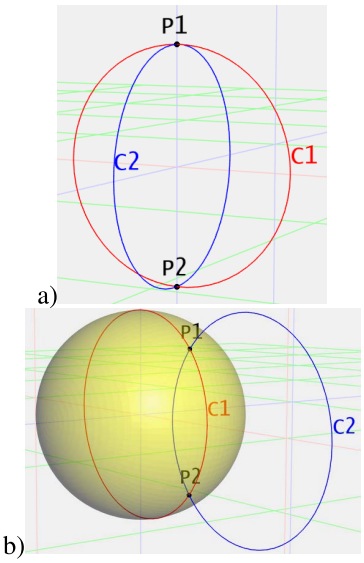


FIGURE 3. a) Intersection between z_1^* and z_4^*. The circles lie in different planes and have two points of intersection. b) Intersection between z_1^* and z_5^*. The circles lie in the same plane and have two points of intersection. In this case, we have replaced the circle z_1^* with a sphere with the circle z_1^* as the equator.

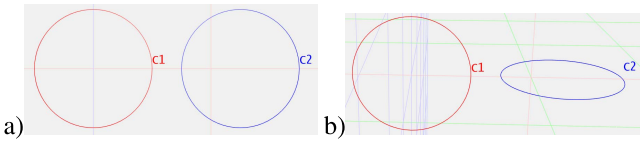


FIGURE 4. a) Intersection between z_1^* and z_6^*. The circles lie in the same plane and do not intersect. b)Intersection between z_1^* and z_7^*. The circles lie in different planes and do not intersect.

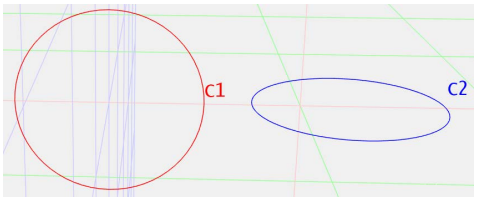


FIGURE 5. Intersection between z_1^* and z_7^*. The circles lie in different planes and do not intersect.

$$
\begin{aligned}
x_{e_4} &= -e_2, \quad x_{e_5} = 3\,e_1 + e_2, \quad x_{e_6} = e_1 + 3\,e_3, \\
x_{e_7} &= e_2, \quad x_{e_8} = e_1 + e_2, \quad x_{e_9} = e_3.
\end{aligned}
$$

The circle and the lines using the OPNS representation are

$$
\begin{aligned}
z^* &= x_{c_1} \wedge x_{c_2} \wedge x_{c_3}, \quad L_1^* = e_\infty \wedge x_{c_4} \wedge x_{c_5}, \\
L_2^* &= e_\infty \wedge x_{c_1} \wedge x_{c_6}, \quad L_3^* = e_\infty \wedge x_{c_7} \wedge x_{c_8}, \\
L_4^* &= e_\infty \wedge e_0 \wedge x_{c_7}, \quad L_5^* = e_\infty \wedge e_0 \wedge x_{c_9}.
\end{aligned}
$$

C. LINE-LINE INTERSECTIONS

For intersections between lines see Figures 8.b and 9.a-b.

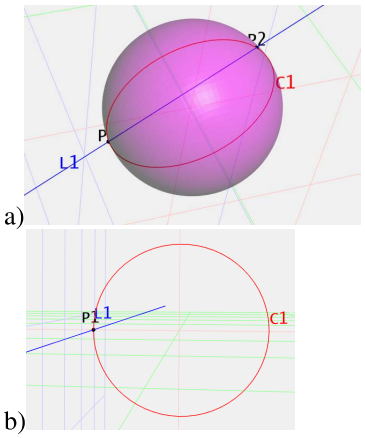


FIGURE 6. a) Intersection between z^* and L_1^*. Both lie in the same plane. Therefore, the circle z^* was replaced by a sphere with the circle as the equator and there are two points of intersection. b) Intersection between z^* and L_2^*. The circle and the line lie in different planes and have one point of intersection.

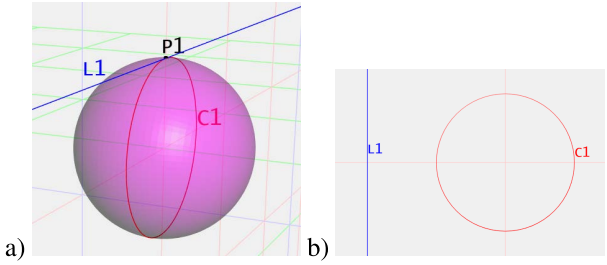


FIGURE 7. a) Intersection between z^* and L_3^*. Both lie in the same plane. Again the circle z^* was replaced by a sphere with the circle as the equator and there is only one point of intersection. b) Intersection between z^* and L_4^*. The circle and the line lie in the same plane but they do not intersect.

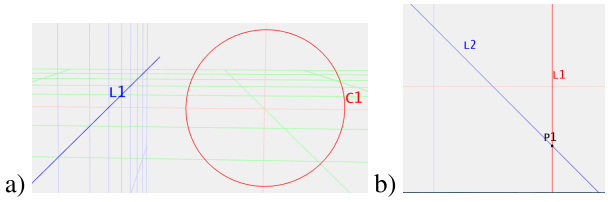


FIGURE 8. a) Intersection between z^* and L_5^*. Both lie in different planes and do not intersect. b)The lines L_1^* and L_2^* have one point of intersection.

The lines are built with the aid of the following Euclidean points:

$$x_{e_1} = 2\,e_1, \quad x_{e_2} = 2\,e_1 + e_2, \quad x_{e_3} = e_1$$
$$x_{e_4} = e_2, \quad x_{e_5} = 3\,e_1, \quad x_{e_6} = 3\,e_1 + e_3$$

In the OPNS representation, the lines are given by

$$L_1^* = e_\infty \wedge x_{c_1} \wedge x_{c_2}, \quad L_2^* = e_\infty \wedge x_{c_3} \wedge x_{c_4}$$
$$L_3^* = e_\infty \wedge x_{c_5} \wedge x_{c_6}, \quad L_4^* = e_\infty \wedge e_0 \wedge x_{c_4}.$$

D. PLANE-CIRCLE INTERSECTIONS

Figure 10.a shows a plane and circle lying in the same plane.

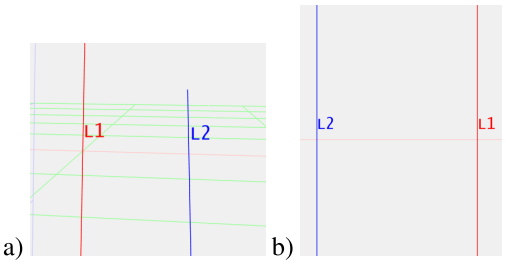


FIGURE 9. a) The lines L_1^* and L_3^* lie in different planes and do not intersect. b) The lines L_1^* and L_4^* lie in the same plane and do not intersect.

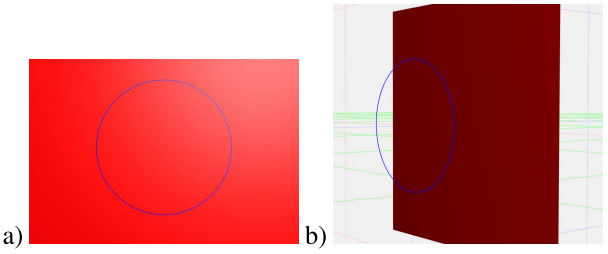


FIGURE 10. a) The plane and circle lie in the same plane. b) The plane and the circle do not intersect.

Figure 10.b shows that a plane and circle do not intersect.

Figure 11.a shows a plane and circle have one point of intersection.

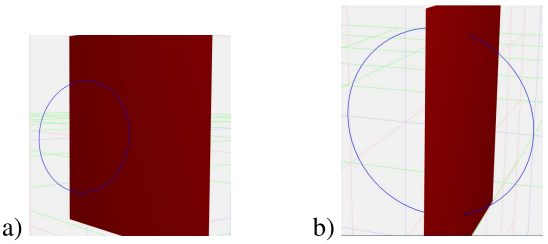


FIGURE 11. a) The plane and the circle have one point of intersection. b) The plane and the circle have two points of intersection.

Figure 11.b shows a plane and circle having two points of intersection.

E. PLANE-LINE INTERSECTIONS

Figure 12.a shows a plane and a line intersecting at one point.

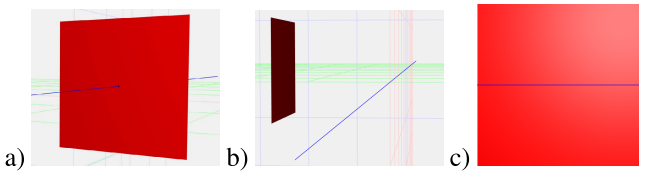


FIGURE 12. a) The plane and the line intersect at one point. b) The plane and the line do not intersect. c) The line lies on the plane.

Figure 12.b shows a plane and a line do not intersect. Figure 12.c shows a line lying in a plane.

F. PLANE INTERSECTION

Figure 13.a shows two planes intersecting in a line.

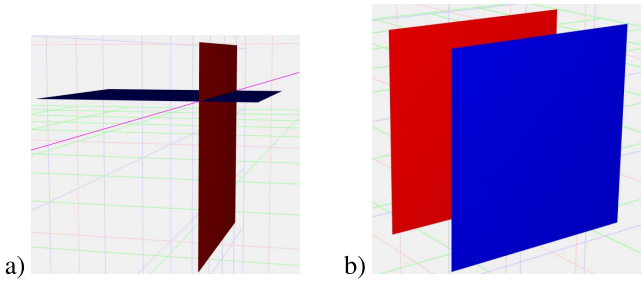


FIGURE 13. a) The two planes intersect in a line. b) The planes do not intersect.

Figure 13.b shows two planes that do not intersect.

G. SPHERE-CIRCLE INTERSECTION

Figure 14.a shows a sphere and a circle which do not intersect.

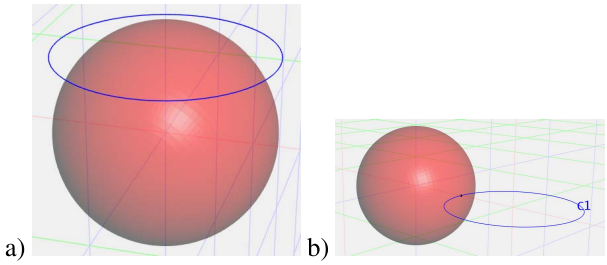


FIGURE 14. a) The sphere and the circle do not intersect. b) The sphere and the circle intersect at one point.

Figure 14.b shows a sphere and a circle intersecting at a point.

The Figure 15.a shows a circle lying on a sphere.

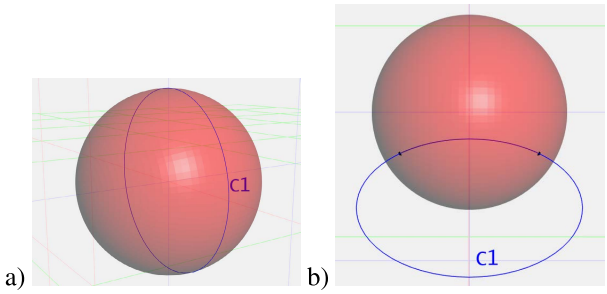


FIGURE 15. a) Special case: the circle lies on the sphere. b) Two circles intersecting at two points.

Figure 15.b shows two circles intersecting at two points.

H. SPHERE-LINE INTERSECTIONS

Figure 16.a shows a sphere and a line that does not intersect.

Figure 16.b shows a sphere and a line intersecting at a point. and Figure 16.c shows a sphere and a line that intersect at two points.

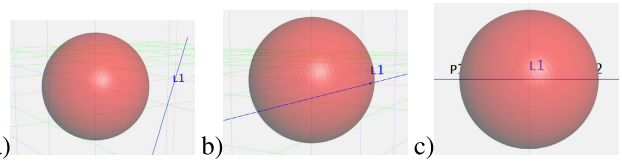


FIGURE 16. a) The sphere and the line do not intersect. b) The sphere and the line intersect at one point. c) The sphere and the line have two points of intersection.

I. SPHERE-PLANE INTERSECTIONS

The Figure 17.a shows a sphere and a plane that intersect in a circle.

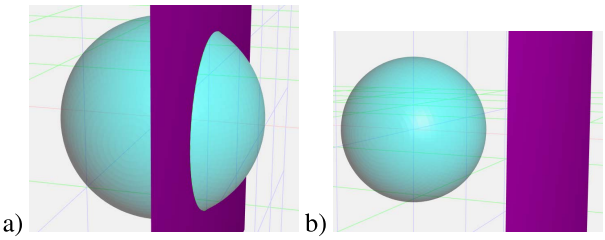


FIGURE 17. a) The sphere and the plane intersect in a circle. b) The sphere and the plane do not intersect.

Figure 17.b shows a sphere and a plane that do not intersect.

Figure 18 shows a plane tangent to a sphere.

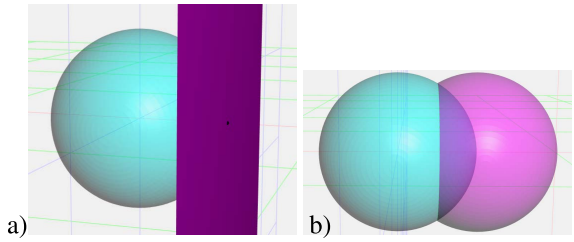


FIGURE 18. a) The plane is tangent to the sphere. b) Two spheres intersecting in a circle.

J. SPHERE-SPHERE INTERSECTION

Figure 18.b shows two spheres that intersect in a circle. The Figure 19.a shows two spheres that do not intersect.

Figure 19.b shows two spheres intersecting at one point.

V. APPLICATIONS

The use of the proposed equations depends upon the applications. You can enclose with geometric shapes certain objects in your setting using points, lines, planes, circles, spheres, hyperplanes, and hyperspheres. Then, the optimization can be done to find the minimal directed distance or a minimal path to be followed among many geometric objects. For example, by applying Quadratic Programming subject to geometric constraints, we can find the optimal solution for tracking.

The measurements obtained using biosensors can be used as parameters needed to define points, lines, planes, or spheres in areas of human tissue. Having these geometric

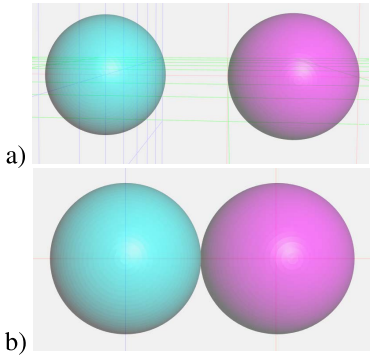


FIGURE 19. a) Two spheres that do not intersect. b)The spheres intersect at one point.

representations, it can be computed directed distances, geometric relations of these entities which lie on the tissue and then compute algorithms to perform surgery maneuvers as shown in 5.C.

Generally speaking, biosensors' applications are for screening infectious to early detection, chronic disease treatment, health management, and well-being surveillance. Improved biosensors technology qualities allow the ability to detect disease and track the body's response to care [15] Our article addresses the use of biosensors' measurement to determine the parameters required for the geometric equations presented in this paper.

The purpose of this subsection is to illustrate the use of the versors for handling rotations and translations and algebra of incidence of conformal geometric primitives like lines, planes, circles, and spheres to model the geometric constraints pertinent for the computation of the inverse kinematics of serial robot manipulator, the interpolation of geometric entities and a couple of tasks for kidney surgery.

A. INVERSE KINEMATICS USING GEOMETRIC PRIMITIVES AND GEOMETRIC CONSTRAINTS

We use points, lines, planes, circles, and spheres in conformal geometric algebra. The procedure consists of five steps, see [16] for more details about the computation of inverse kinematics in the conformal geometric algebra.

Step 1: Compute the position of p_2.

The sphere represented with a center at p_t and radius d_3 reads

$$S_t = p_t - \frac{1}{2}d_3^2 e_\infty. \tag{83}$$

The second constraint describes that the gripper is parallel to the plane π_t. The plane π_t intersects the sphere S_t, see Fig. 20.(a).

$$z_t = S_t \wedge \pi_t \tag{84}$$

The last alignment condition: l_t passes through the point p_t. This lies on the plane π_t and intersects also the circle z_t at the point p_2.

$$P_{p2} = l_t \wedge S_t \tag{85}$$

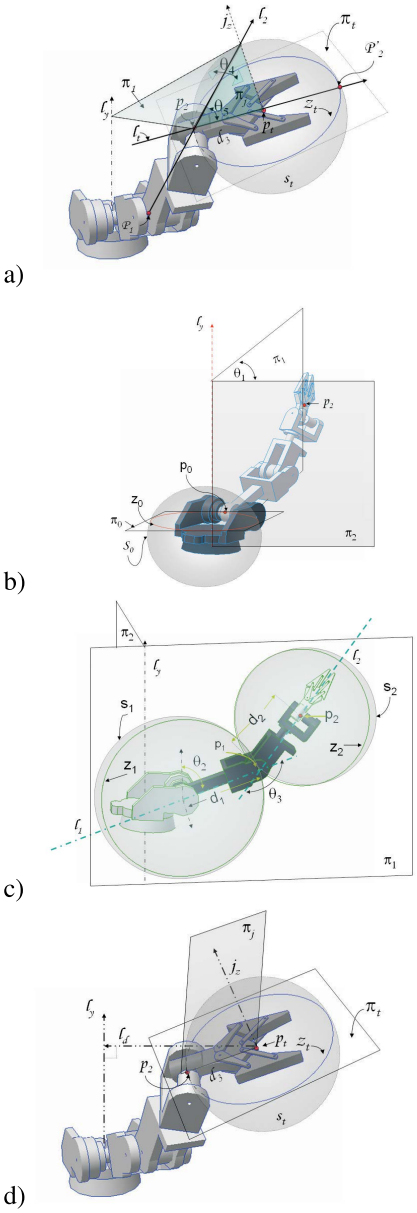


FIGURE 20. (a) Touchpoint p_t, grasp plane π_t, and direction l_t. (b) p_0 point as intersection of the planes π_1, π_0 and the sphere s_0. c) Point p_1 intersection of the spheres s_1, s_2 and the plane π_1. d) Point p_2 computed by intersecting the plane π_t, s_t and the plane. π_j.

Step 2: Compute the position of the point p_0.

The y-axis (l_y) is the line going through the origin with direction e_2;

$$l_y^* = e_2 E. \tag{86}$$

When the base rotates around the y-axis (see Fig.20.(b)), the point p_0 describes the circle z_0. This circle is the intersection of the plane π_0 and sphere, with the center at the origin and radius d_0:

$$S_0 = e_o - \frac{d_o^2}{2}e_\infty, \quad \pi_0 = e_2 + he_\infty, \quad z_0 = S_o \wedge \pi_0. \tag{87}$$

Point p_0 must lie on the plane π_1 generated by the y-axis (l_y in Eq. (86)) and the point p_2 is calculated in step 1 (Eq. (85)), as we can see in Fig. 20.(b), so that p_0 can be determined by intersecting the plane π_1 with the circle z_0.

$$\pi_1^* = l_y^* \wedge p_2, \quad P_{p0} = z_0 \wedge \pi_1. \tag{88}$$

Note that we get as the solution a pair of points, thus one selects a point towards the closest robot joint.

Step 3: Compute the point p_1.

The position p_1 is computed considering the intersection of the spheres S_1 and S_2 with the plane π_1, see Fig. 20.(c):

$$\begin{aligned} S_1 &= p_1 - \frac{d_1^2}{2} e_\infty, \quad S_2 = p_2 - \frac{d_2^2}{2} e_\infty \\ P_{p1} &= S_1 \wedge S_2 \wedge \pi_1. \end{aligned} \tag{89}$$

As previously indicated, we choose the closest point to the robot joint.

Step 4: Compute the lines and planes between the robot joints.

Once p_0, p_1, and p_2 have been computed, the lines l_1, l_2, and l_3 and the planes π_2 and π_3 can be formulated. These lines and planes are needed to compute the angles $\theta_1 \cdots \theta_5$:

$$\begin{aligned} \pi_3^* &= p_1 \wedge p_2 \wedge p_t \wedge e_\infty, \quad \pi_2^* = e_3 I_c, \\ l_1^* &= p_0 \wedge p_1 \wedge e_\infty, \\ l_2^* &= p_1 \wedge p_2 \wedge e_\infty, \quad l_3^* = p_2 \wedge p_3 \wedge e_\infty. \end{aligned} \tag{90}$$

Step 5: Find the angles $\theta_1 \ldots \theta_5$.

after these computations, the angles are calculated as follows:

$$\begin{aligned} \cos(\theta_1) &= \frac{\pi_1^* \cdot \pi_2^*}{|\pi_1^*|\,|\pi_2^*|}, \quad \cos(\theta_2) = \frac{l_1^* \cdot l_y^*}{|l_1^*|\,|l_y^*|}, \\ \cos(\theta_3) &= \frac{l_1^* \cdot l_2^*}{|l_1^*|\,|l_2^*|}, \\ \cos(\theta_4) &= \frac{\pi_1^* \cdot \pi_3^*}{|\pi_1^*|\,|\pi_3^*|}, \quad \cos(\theta_5) = \frac{l_2^* \cdot l_3^*}{|l_2^*|\,|l_3^*|}. \end{aligned} \tag{91}$$

B. INTERPOLATION OF GEOMETRIC ENTITIES

Using the motor $\boldsymbol{M}_B$ for blend interpolation, the IPNS geometric entities can be interpolated. To dilate spheres, we apply a Dilator $\boldsymbol{D} = e^{ln\rho}$

$$x_c^t = \boldsymbol{M}_B^t x_c^{t-1} \tilde{\boldsymbol{M}}_B^t \tag{92}$$

$$L^t = \boldsymbol{M}_B^t L^{t-1} \tilde{\boldsymbol{M}}_B^t \tag{93}$$

$$\pi^t = \boldsymbol{M}_B^t \pi^{t-1} \tilde{\boldsymbol{M}}_B^t \tag{94}$$

$$s^t = \boldsymbol{D}^t \boldsymbol{M}_B^t s^{t-1} \tilde{\boldsymbol{M}}_B^t \boldsymbol{D}^{-1^t}. \tag{95}$$

Figure 21 presents the interpolation of points, lines, planes, and spheres under the action of a motor $\boldsymbol{M}_B^t$ and only for spheres with a dilator $\boldsymbol{D}^t$.

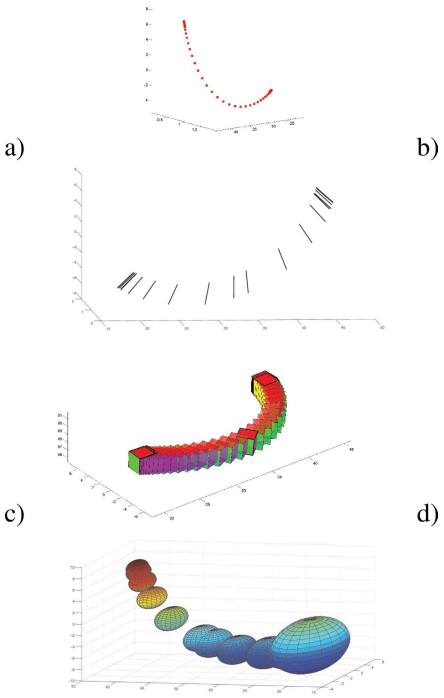


FIGURE 21. Interpolation of geometric entities: a) points; b) lines; c) planes; and d) spheres.

C. PROCEDURES FOR KIDNEY SURGERY

In tasks of human-machine interaction, one requires a good interface to carry out certain tasks. It is of most importance to offer the user an intuitive procedure to carry out the handling of robot arms or controlled devices to improve the efficiency of certain tasks like in surgery. A human-machine interface for the handling of certain tasks has to enhance the experience and intuition of the user and not inhibit his ability to carry out the task [19]. The use of geometric methods to design algorithms for human-machine interaction is the right way to go, to close the gap between the specialized user and the physics of the problem in question. Next, we illustrate the design of geometric procedures using incidence algebra in conformal geometric algebra. We show two algorithms for kidney surgery, see [17] for more details about these surgery procedures using the conformal geometric algebra.

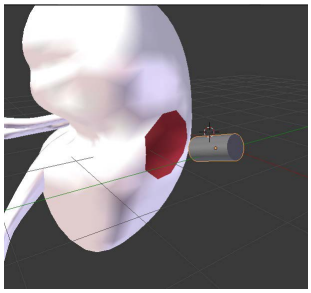


FIGURE 22. Virtual models of a kidney and ultrasound probe.

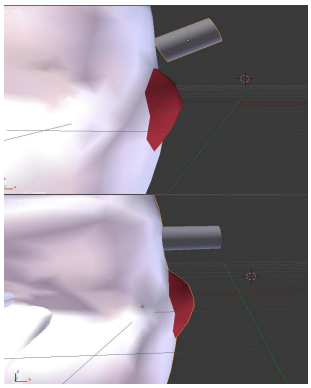


FIGURE 23. Contact and poses of the organ and the USP.

1) INSPECTION USING ULTRA SOUND PROBE

For surgery, one needs to track the Ultra Sound Probe (USP), see Figure 22.

It is required the compute the USP pose and update it in the virtual world. This is important to sustain an adequate haptic interaction, see Figure 23.

To compute the pose, one plane is selected with tree points $Xus_{1,2,3}$ on the face of the US probe

$$\pi_1 = Xus_1 \wedge Xus_2 \wedge Xus_3, \tag{96}$$

and a plane belonging to the organ's surface computed with tree points in general position

$$\pi_2 = Xkn_1 \wedge Xkn_2 \wedge Xkn_3. \tag{97}$$

Figure 24 shows the planes on the US probe and the organ. The contact is computed as follows

$$\pi_1\pi_2 + \pi_2\pi_1, \tag{98}$$

if the planes are in contact the equation (98) becomes zero.

We used the affine plane $x^h = x_e + e_o$, Maple 18 y eClifford [18]. Compute firstly the direct distance d between the line $L_1^h = a_1^h \wedge a_2^h$ and the plane $\pi_2^h = b_1^h \wedge b_2^h \wedge b_3^h$

$$\begin{aligned} d[a_1^h \wedge a_2^h, b_1^h \wedge b_2^h \wedge b_3^h] \\ \equiv \{\overline{e} \cdot (a_1^h \wedge a_2^h)\} \wedge \{\overline{e} \cdot (b_1^h \wedge b_2 \wedge b_3^h)\}]^{-1} \\ [\overline{e} \cdot (a_1^h \wedge a_2^h \wedge b_1^h \wedge b_2^h \wedge b_3^h)] \\ = [(a_2 - a_1) \wedge (b_2 - b_1) \wedge (b_3 - b_2)]^{-1} \\ \times [(a_2 - a_1) \wedge (b_1 - a_2) \wedge (b_2 - b_1) \wedge (b_3 - b_2)]. \end{aligned} \tag{99}$$

The intersection point of the line L_1^h and π_2^h is given by

$$p = L_1^h \cap \pi_2^h = L_1^h \cdot (\pi_2^h)^* = L_1^h \cdot (\pi_2^h \cdot \bar{I}). \tag{100}$$

If the direct distance $d \neq 0$, the plane ϕ_1^h, where the line lies, is parallel to the plane π_2^h, see Figure 25.

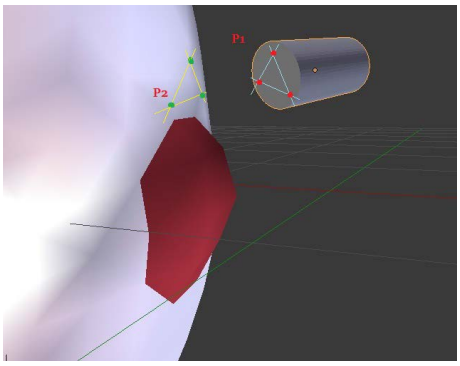


FIGURE 24. Representation of the incidence relations between planes.

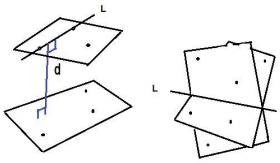


FIGURE 25. (left) The direct distance from the plane π_1^h to the other plane π_1^2. (right) The planes intersect in a line.

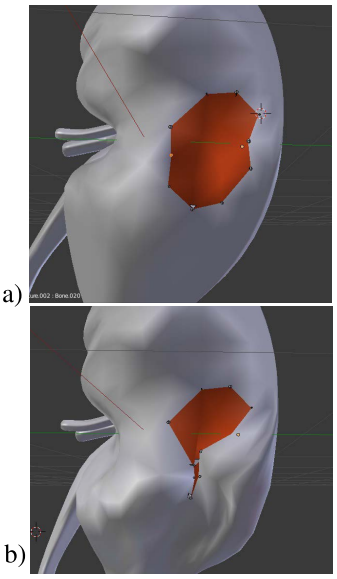


FIGURE 26. a) Selected points are shown in the virtual kidney. b) The suture of the kidney wound.

2) SUTURE

In real surgery, one can point out the suture points with a laser pointer and then transfer them to the virtual organ. Figure 26.a shows the selected points.

Figure 26.b illustrates how the suture follows sequentially opposite points.

For the maneuver, firstly a translator $\boldsymbol{T}$ is computed to translate a point onto another

$$T = 1 + e_\infty \left(\frac{X_i - X_j}{2} \right), \tag{101}$$

where X_i and X_j stand for a couple of points that are being to be closed with a string. Then, the need is carried to the opposite point

$$Z_i = TZ_i\tilde{T}, \tag{102}$$

Figure 27 depicts the closed wound.

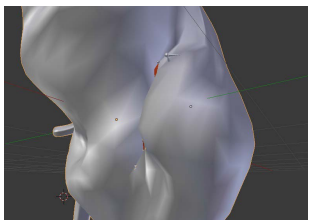


FIGURE 27. Kidney after the suture.

One assumes while suturing the needle has to pull the string, thus a rotor $\boldsymbol{R}$ has to be computed to ensure the motion of the needle along a semicircle.

Figure 28.a shows the geometry of this operation and Figure 28.b depicts the needle trajectory.

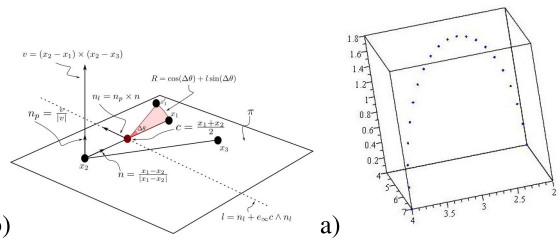


FIGURE 28. a) The rotor R is computed. b) The semicircle trajectory followed by the needle.

D. INTERPOLATION OF SURGERY MOTION

In this subsection, we show the use of the quadratic Study interpolation algorithm to interpolate geometric entities and to guide the suture of a kidney wound, see [20] for more details about this interpolation.

According Gfrerrer [21], an interpolation algorithm can be derived using the Study's manifold. The idea is to map Euclidean transformations ($SE(3)$) into the Study quadric $\mathcal{M}_6$ (projective space $\mathbb{P}^7$) as 8-D homogeneous points.

$$\mathfrak{G} = \begin{Bmatrix} SE(3) \to \mathbb{P}^7 \\ \alpha \to X = (x_0, \ldots, x_7)\,, \end{Bmatrix} \tag{103}$$

$SE(3)$ is the special Euclidean group, α stands for any Euclidean rigid transformation and the vector X has homogeneous coordinates $X \in \mathcal{M}_6$. In this work, we use motor algebra $G^+_{3,0,1}$ to model SE(3). Note that $G^+_{3,0,1}$ is a sub-algebra of the 3D conformal geometric algebra $G_{4,1}$ as well.

Consider a set $\mathbf{X} \in \mathcal{M}_6$ of 3 homogeneous points $X_1, X_2, X_3 \in\in \mathcal{M}_6$. One generates a interpolation curve by interpolation of the homogeneous points $X'_0, \ldots, X'_N \in \mathbb{P}^7$ which in turn satisfy the points computed as follows

$$\mathcal{X} = f_0(t)X_1^T \mathcal{Q}X_2X_0 + f_1(t)X_0^T \mathcal{Q}X_2X_1 + f_2(t)X_0^T \mathcal{Q}X_1X_2. \tag{104}$$

Note, that this equation is a discretization of the curve $X(t)$ into N points, here the intermediate points are computed as well.

We formulate the interpolation polynomials $f_0(t)$, $f_1(t)$, and $f_2(t)$ as follows

$$\begin{aligned} f_0 &= (t_0 - t_1)(t_0 - t_2)(t - t_1)(t - t_2) \\ f_1 &= (t_1 - t_0)(t_1 - t_2)(t - t_0)(t - t_2) \\ f_2 &= (t_2 - t_0)(t_2 - t_1)(t - t_0)(t - t_1), \end{aligned} \tag{105}$$

where t, t_0, t_1, and t_2 stand for the interpolation values between 0 and 1.

This Study's quadric-based motor interpolation was used to interpolate trajectories in medical robotics for kidney surgery, see Figure 29.

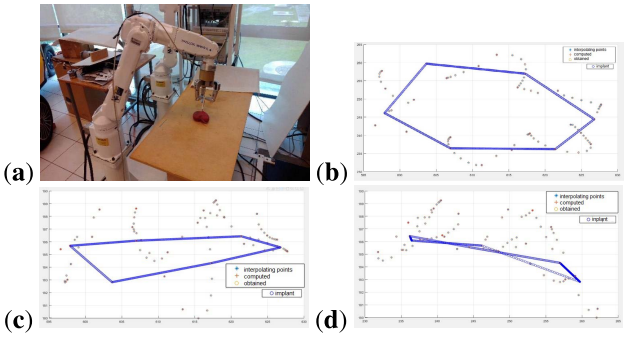


FIGURE 29. (a) Robot; views: (b) X-Y, (c) X-Z, (d) Y-Z.

VI. CONCLUSION

In this article, the authors present geometric methods in terms of a complete set of equations using incidence algebra and conformal geometric algebra. These methods and equations can be used to tackle a variety of problems in graphic engineering, GIS, augmented virtual reality, Robotics, and Human-Machine Interaction. To apply these methods for real-time applications, the algorithms can be sped up using FPGA of CUDA (Nvidia), as shown in [1] and [22]. We illustrate the use of Conformal Geometric algebra together with incidence algebra in cases of geometric interpolation, inverse kinematics of a robot arm, and some surgical operations.

REFERENCES

[1] G. Soria-Garcia, G. Altamirano-Gomez, S. Ortega-Cisneros, and E. Bayro-Corrochano, "Conformal geometric algebra voting scheme implemented in reconfigurable devices for geometric entities extraction," *IEEE Trans. Ind. Electron.*, vol. 64, no. 11, pp. 8747–8755, Nov. 2017.

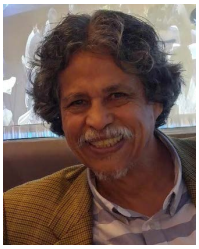
[2] T. Koolen, S. Bertrand, G. Thomas, T. de Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot atlas," *Int. J. Hum. Robot.*, vol. 13, no. 1, Mar. 2016, Art. no. 1650007.

[3] F. Kanehiro, F. Lamiraux, O. Kanoun, E. Yoshida, and J.-P. Laumond, "A local collision avoidance method for non-strictly convex polyhedra," in *Robotics: Science and Systems IV*. MIT Press, Jun. 2008, pp. 151–158.

[4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Press Univ., 2000.

[5] D. Hestenes and R. Ziegler, "Projective geometry with Clifford algebra," *Acta Applicandae Mathematicae*, vol. 23, no. 1, pp. 25–63, Apr. 1991.

[6] J. Lasenby and E. Bayro-Corrochano, "Analysis and computation of projective invariants from multiple views in the geometric algebra framework," in *Invariants for Pattern Recognition and Classification* (Series in Machine Perception and Artificial Intelligence), vol. 42, M. Rodrigues, Ed. Singapore: World Scientific, 2000, p. 233.
[7] A. W. M. Dress and T. F. Havel, "Distance geometry and geometric algebra," *Found. Phys.*, vol. 23, no. 10, pp. 1357–1374, Oct. 1999.
[8] E. E. Bayro–Corrochano and G. G. Sobczyk, "Applications of Lie algebras and the algebra of incidence," in *Advances in Geometric Algebra With Applications in Science and Engineering*, E. Bayro–Corrochano and G. Sobczyk, Eds. New York, NY, USA: Birkhauser, 2001, pp. 252–276.
[9] J. Lasenby, A. Lasenby, and R. Wareham, "A covariant approach to geometry using geometric algebra," Dept. Eng., Cambridge Univ., Tech. Rep. CUED/F-INFENG/TR-483.
[10] J. Vince, *Geometric Algebra for Computer Graphics*. London, U.K.: Springer, 2008.
[11] C. G. Gunn, "Geometric algebras for Euclidean geometry," 2014, *arXiv:1411.6502*.
[12] D. Hildenbrand, *Introduction to Geometric Algebra Computing*. Boca Raton, FL, USA: CRC Press, 2019.
[13] E. Bayro-Corrochano and L.-E. Falcon, "Geometric algebra of points, lines, planes, and spheres for computer vision and robotics," *Robotica*, vol. 60, pp. 1–16, Nov. 2005.
[14] E. Bayro-Corrochano, *Geometric Algebra Applications* (Computer Vision, Graphics and Neurocomputing), vol. 1. Heidelberg, Germany: Springer-Verlag, 2019.
[15] A. Haleem, M. Javaid, R. P. Singh, R. Suman, and S. Rab, "Biosensors applications in medical field: A brief review," *Sensors Int.*, vol. 2, Jan. 2021, Art. no. 100100.
[16] J. Zamora-Esquivel and E. Bayro-Corrochano, "Parallel forward dynamics: A geometric approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taiwan, Oct. 2010, pp. 2377–2382.
[17] E. Bayro-Corrochano, A. M. Garza-Burgos, and J. L. Del-Valle-Padilla, "Geometric intuitive techniques for human machine interaction in medical robotics," *Int. J. Social Robot.*, vol. 12, no. 1, pp. 91–112, Jan. 2020.
[18] R. Ablamowicks. *eCLIFFORD Software Packet Using Maple for Clifford Algebra. Computations*. Accessed: Aug. 1, 2016. [Online]. Available: http://math.tntech.edu/rafal
[19] K. Retch, *Transendoscopy Ultrasound for Neurosurgery*. Heidelberg, Germany: Springer-Verlag, 2006, ch. 5.
[20] E. Bayro-Corrochano, L. Lechuga-Gutiérrez, and M. Garza-Burgos, "Geometric techniques for robotics and HMI: Interpolation and haptics in conformal geometric algebra and control using quaternion spike neural networks," *Robot. Auto. Syst.*, vol. 104, pp. 72–84, Jun. 2018.
[21] A. Gfrerrer, "Study's kinematic mapping—A tool for motion design," in *Advances in Robot Kinematics*. Heidelberg, Germany: Springer, 2000, pp. 7–16.
[22] G. Martinez-Terán, O. Ureña-Ponce, G. Soria-García, S. Ortega-Cisneros, and E. Bayro-Corrochano, "Fast study quadric interpolation in the conformal geometric algebra framework," *J. Electron.*, vol. 11, no. 10, p. 1527, May 2022.



EDUARDO JOSE BAYRO-CORROCHANO (Senior Member, IEEE) is currently a Full Professor of geometric cybernetics with Cinvestav, Guadalajara, Mexico. He is a fellow of the International Association of Pattern Recognition Society and the Asian Artificial Intelligence Association. He was the General Chair of IAPR-ICPR'2016 in December, Cancun, Mexico; and of IEEE/RAS Humanoids 2016, November, Cancun, México.

G. ALTAMIRANO-ESCOBEDO is currently pursuing the Ph.D. degree with the Cinvestav, Guadalajara, Mexico. His research interests include signal and image processing, artificial intelligence, medical image processing, and geometric algebra applications.

A. ORTIZ-GONZALEZ is currently pursuing the Ph.D. degree in computer science and electrical engineering with the Center of Research and Advanced Studies, Cinvestav, Guadalajara, Mexico. His research interests include signal and image processing, artificial intelligence, medical image processing, graphics engineering, and geometric algebra applications.

V. FARIAS-MORENO is currently pursuing the Ph.D. degree in computer science and electrical engineering with the Center of Research and Advanced Studies, Cinvestav, Guadalajara, Mexico. His research interests include mechatronics, control engineering, artificial intelligence, robotics, and geometric algebra applications.

N. CHEL-PUC is currently pursuing the Ph.D. degree in computer science and electrical engineering with the Center of Research and Advanced Studies, Cinvestav, Guadalajara, Mexico. His research interests include control engineering, robotics, humanoids, and geometric algebra applications.