CGAlgebra: a Mathematica package for conformal geometric algebra. v.2.0

E. Alejandra Ortíz-Durán and José L. Aragón

Centro de Física Aplicada y Tecnología Avanzada,
Universidad Nacional Autónoma de México,
Apartado Postal 1-1010, Querétaro 76000, Mexico
jlaragon@unam.mx

Abstract

A tutorial of the Mathematica package CGAlgebra, for conformal geometric algebra calculations is presented. Using rule-based programming, the 5-dimensional conformal geometric algebra is implemented and the defined functions simplify the calculations of geometric, outer and inner products, as well as many other calculations related with geometric transformations. CGAlgebra is available from https://github.com/jlaragonvera/Geometric-Algebra

1 Introduction

In the 5D conformal geometric algebra geometric objects such as lines, planes, circles and spheres in 3D are represented in a simple way by algebraic identities. It is an extension of the 4D projective geometric algebra and was proposed by D. Hestenes [1, 2] as a powerful framework for computational Euclidean geometry. Extensive applications of the conformal geometric algebra to computer graphics, computer vision and robotics have been reported. As general references we recommend the books [3, 4, 5, 6].

Definition 1 (The conformal space) The five-dimensional space spanned by \{e_0, e_1, e_2, e_3, e_\infty\} with the inner product

\[ \langle x, y \rangle = x_1y_1 + x_2y_2 + x_3y_3 - x_0y_\infty - x_\infty y_0, \]  

(1)
for \( \mathbf{x} = x_0 e_0 + x_1 e_1 + x_2 e_2 + x_3 e_3 + x_\infty e_\infty \), and \( \mathbf{y} = y_0 e_0 + y_1 e_1 + y_2 e_2 + y_3 e_3 + y_\infty e_\infty \), vectors in this space, is called the conformal space.

From (1) we see that \( \|e_0\|^2 = \langle e_0, e_0 \rangle = 0 \) and similarly \( \|e_\infty\|^2 = 0 \), so \( e_0 \) and \( e_\infty \) are null vectors.

The inner product (1) has signature (4, 1) and the conformal space is the Minkowsky space \( \mathbb{R}^{4,1} \).

For the purpose of the Mathematica package developed here, a definition of the conformal geometric algebra based on generators and relations is more suitable.

**Definition 2** \( (\mathbb{G}^{4,1}) \) The conformal geometric algebra \( \mathbb{G}^{4,1} \) over \( \mathbb{R}^{4,1} \), equipped with the inner product (1), is an algebra generated by the identity 1 and the symbols \( \{e_0, e_1, e_2, e_3, e_\infty\} \), by an associative multiplication operation called geometric product, and the following relations:

\[
\begin{align*}
\text{(2a)} & \quad e_ie_j + eje_i = 0, \\
\text{(2b)} & \quad e_i e_0 + e_0 e_i = 0, \\
\text{(2c)} & \quad e_i e_\infty + e_\infty e_i = 0, \\
\text{(2d)} & \quad e_i^2 = 1, \\
\text{(2e)} & \quad e_0 e_\infty + e_\infty e_0 = -2, \\
\text{(2f)} & \quad e_0^2 = e_\infty^2 = 0,
\end{align*}
\]

for \( i, j = 1, 2, 3 \).

Another way to build the conformal geometric algebra \( \mathbb{G}^{4,1} \) is to start with the Clifford algebra \( \mathbb{C}^{4,1} \) over \( \mathbb{R}^{4,1} \), which is the five-dimensional space generated by \( \{e_1, e_2, e_3, e_4, e_5\} \) with the inner product

\[
\langle \mathbf{x}, \mathbf{y} \rangle = x_1 y_1 + x_2 y_2 + x_3 y_3 + x_4 y_4 - x_5 y_5,
\]

for \( \mathbf{x} = x_1 e_1 + x_2 e_2 + x_3 e_3 + x_4 e_4 + x_5 e_5 \), and \( \mathbf{y} = y_1 e_1 + y_2 e_2 + y_3 e_3 + y_4 e_4 + y_5 e_5 \), vectors in \( \mathbb{C}^{4,1} \). In this space we now define the null vectors

\[
e_0 = \frac{e_4 + e_5}{2}, \quad \text{and} \quad e_\infty = e_5 - e_4.
\]

With this definition, the conformal geometric algebra \( \mathbb{G}^{4,1} \) is the generated by \( \{e_0, e_1, e_2, e_3, e_\infty\} \).
The inverse of the transformation (4) is
\[ e_4 = e_0 - \frac{e_\infty}{2}, \quad \text{and} \quad e_5 = e_0 + \frac{e_\infty}{2}. \] (5)

Both approaches are used in the *Mathematica* package.

Examples of the representation and transformations of the basic geometric objects of this conformal space, using *CGAlgebra* will be given in what follows.

## 2 Getting started with CGAlgebra

*CGAlgebra* is freely available and can be downloaded from

https://github.com/jlaragonvera/Geometric-Algebra

In a *Mathematica* session the package is loaded with the command

```
<<"DIR/CGAlgebra.m"
```

where DIR is the full path of the directory where the package is located.

In *CGAlgebra*, a basis element of the conformal space is denoted by \( e[i] \).

With this notation, \( \mathbb{G}^{4,1} \) is generated by \{e[0],e[1],e[2],e[3],e[\infty]\}.

The geometric product between basis elements \( e_ie_j\ldots e_k \) will be denoted as \( e[i,j,\ldots,k] \). To warm up we can test some of the relations (2):

\[
\begin{align*}
\text{In}[1] & := e[2,1] \\
\text{Out}[1] & := -e[1,2] \\
\text{In}[2] & := e[\infty,0] \\
\text{Out}[2] & := -2-e[0,\infty] \\
\text{In}[3] & := e[\infty,\infty] \\
\text{Out}[3] & := 0 \\
\text{In}[3] & := e[0,0] \\
\text{Out}[3] & := 0
\end{align*}
\]

The geometric product between an arbitrary number basis elements is, for example:

\[
\begin{align*}
\text{In}[4] & := e[1,\infty,2,0] \\
\text{Out}[4] & := 2e[1,2]+e[0,1,2,\infty]
\end{align*}
\]
Table 1: Basic functions of CGAlgebra

| Expression                        | Output                                                                 |
|-----------------------------------|------------------------------------------------------------------------|
| GeometricProduct[A,B,C,...]       | The geometric product of the multivectors A, B, C, ...                 |
| OuterProduct[A,B,C,...]           | The outer (Grasmann) product of the multivectors A, B, C, ...          |
| InnerProduct[A,B]                 | The inner product (left contraction) of the multivectors A and B.      |
| Grade[A,k]                        | The k-vector part of the multivector A.                                |

To operate with arbitrary multivectors, the geometric product between basis elements is extended by linearity to all \( \mathbb{G}^{4,1} \). The basic functions defined in CGAlgebra are listed in Table 1.

As an example consider the geometric product between the multivectors \( A = e_1e_2e_3 + a e_{\infty}e_3e_2 \), \( B = a e_2 \), \( C = 3 \) and \( D = 4 + e_1e_3 \). It is computed as

\[
\text{In[5]} := \text{GeometricProduct}[e[1,2,3]+a e[\infty,3,2], a e[2],3,4+e[1,3]]
\]
\[
\text{Out[5]} := 3a - 12a e[1,3] + 3a^2 e[1,\infty] - 12a^2 e[3,\infty]
\]

The expressions \( e[] \), \( \text{GeometricProduct}[] \) and \( \text{Grade}[] \) constitute the basis of the CGAlgebra package. Extra defined functions and functionality will be presented through examples concerning the representation and transformation of geometrical objects in the conformal space.

It is important to mention that all the results are given in terms of geometric products of the basis vectors \( \{e[0],e[1],e[2],e[3],e[\infty]\} \). Thus, for instance, the pseudoscalar \( I_5 = e_0 \wedge e_1 \wedge e_2 \wedge e_3 \wedge e_\infty \) is computed as

\[
\text{In[6]} := \text{OuterProduct}[e[0], e[1], e[2], e[3], e[\infty]]
\]
\[
\text{Out[6]} := -e[1,2,3] + e[0,1,2,3,\infty]
\]
That is, \( I_5 = -e_1e_2e_3 + e_0e_1e_2e_3e_\infty \).

The pseudoscalar is predefined in the package as I5:

\[
\text{In[7]} := I5
\]
\[
\text{Out[7]} := -e[1,2,3] + e[0,1,2,3,\infty]
\]
3 Representation of geometric objects

Examples of direct and dual representations in the conformal space of some geometric objects are presented in what follows.

3.1 Points

A point at the location \( x = (x, y, z) \in \mathbb{R}^3 \) is represented by the vector

\[
x = e_0 + x + \frac{1}{2}x^2 e_\infty,
\]

in \( \mathbb{G}^{4,1} \).

We can immediately check that \( x^2 = xx = \|x\|^2 = 0 \):

\[
\begin{align*}
\text{In}[8] & := X = x_1 e[1] + x_2 x[2] + x_3 e[3]; \\
\text{In}[9] & := x = e[0] + X + (\text{GeometricProduct}[X,X]/2) e[\infty]; \\
\text{In}[10] & := \text{GeometricProduct}[x, x] // \text{Simplify} \\
\text{Out}[10] & := 0
\end{align*}
\]

3.2 Direct representation of lines

A line \( L \) passing through two points \( p_1 \) and \( p_2 \) is represented by

\[
L = p_1 \wedge p_2 \wedge e_\infty.
\]

Thus if \( p \) is a point in the line, it satisfies

\[
p \wedge L = 0. \quad (6)
\]

Let \( x = (x, y, z) \), \( x_1 = (x_1, y_1, z_1) \), \( x_2 = (x_2, y_2, z_2) \), \( p = e_0 + x + \frac{x^2}{2} e_\infty \), \( p_1 = e_0 + x_1 + \frac{x_1^2}{2} e_\infty \) and \( p_2 = e_0 + x_2 + \frac{x_2^2}{2} e_\infty \). The equation of the line passing through \( p_1 \) and \( p_2 \) can be obtained as follows:

\[
\begin{align*}
\text{In}[11] & := X = x e[1] + y x[2] + z e[3]; \\
\text{In}[12] & := Y = x_1 e[1] + y_1 x[2] + z_1 e[3]; \\
\text{In}[13] & := Z = x_2 e[1] + y_2 x[2] + z_2 e[3]; \\
\text{In}[14] & := p = e[0] + X + (\text{Magnitude}[X]^2/2) e[\infty]; \\
\text{In}[15] & := p_1 = e[0] + Y + (\text{Magnitude}[Y]^2/2) e[\infty]; \\
\text{In}[16] & := p_2 = e[0] + Z + (\text{Magnitude}[Z]^2/2) e[\infty];
\end{align*}
\]
In[17]:= line = OuterProduct[p, p1, p2, e[∞]] // FullSimplify
Out[17]:= (-x1 y + x2 y + x y1 - x2 y1 - x y2 + x1 y2) e[1,2] +
(-y1 z + x2 z + x z1 - x2 z1 - x z2 + x1 z2) e[1,3] +
(-y1 z + y2 z + y z1 - y2 z1 - y z2 + y1 z2) e[2,3]
(x2 y + x y1 - x2 y1 - x y2 + x1 y2)(-y2) e[0,1,2,∞] +
(-x1 z + x2 z + x z1 - x2 z1 - x z2 + x1 z2) e[0,1,3,∞] +
(-y1 z + y2 z + y z1 - y2 z1 - y z2 + y1 z2) e[0,2,3,∞] +
(-x2 y + x1 y2 z + x2 y z1 - x y2 z1 - x1 y z2 - x y1 z2) e[1,2,3,∞]

Then from (6), all the coefficients of the 2-vectors and the 4-vectors must be
set to zero:

In[18]:= Solve[
Coefficient[line, e[1,2] == 0,
Coefficient[line, e[1,3] == 0,
Coefficient[line, e[2,3] == 0,
Coefficient[line, e[0,1,2,∞] == 0,
Coefficient[line, e[0,1,3,∞] == 0,
Coefficient[line, e[0,2,3,∞] == 0,
Coefficient[line, e[1,2,3,∞]] == 0},
{x, y, z}]
Out[18]:= Solve::svars: Equations may not give solutions for all
"solve" variables.
{{y -> -((x (-y1 + y2))/(x1 - x2)) - (x2 y1 - x1 y2)/(x1 - x2),
z -> -((-x (-z1 + z2))/(x1 - x2)) - (x2 z1 - x1 z2)/(x1 - x2))
That is:
\[
\begin{align*}
y &= \frac{x(y1 - y2)}{x1 - x2} - \frac{x2y1 - x1y2}{x1 - x2}, \\
z &= \frac{y(z1 - z2)}{x1 - x2} - \frac{x2z1 - x1z2}{x1 - x2}.
\end{align*}
\]

If we define \(x = (x1 - x2)t\), the parametric equations of the line are:
\[
\begin{align*}
x &= (x1 - x2)t, \\
y &= (y1 - y2)t - \frac{x2y1 - x1y2}{x1 - x2}, \\
z &= (z1 - z2)t - \frac{x2z1 - x1z2}{x1 - x2}.
\end{align*}
\]
3.3 Direct representation of planes

A plane $P$ passing through three points $p_1$, $p_2$ and $p_3$ is represented by

$$ P = p_1 \wedge p_2 \wedge p_3 \wedge e_\infty. \quad (7) $$

If $p$ is a point in the line, its equation if given as

$$ p \wedge P = 0. \quad (8) $$

As above, let $x = (x, y, z)$, $x_1 = (x_1, y_1, z_1)$, $x_2 = (x_2, y_2, z_2)$, $x_3 = (x_3, y_3, z_3)$, and

$$ p = e_0 + x + \frac{x^2}{2} e_\infty, \quad (9a) $$

$$ p_1 = e_0 + x_1 + \frac{x_1^2}{2} e_\infty, \quad (9b) $$

$$ p_2 = e_0 + x_2 + \frac{x_2^2}{2} e_\infty, \quad (9c) $$

$$ p_3 = e_0 + x_3 + \frac{x_3^2}{2} e_\infty. \quad (9d) $$

In 3D, the equation of the plane passing through $x_1$, $x_2$ and $x_3$ can be obtained as follows:

$$ \text{In}[19]:= X = x \ e[1] + y \ x[2] + z \ e[3]; $$
$$ \text{In}[20]:= Y = x1 \ e[1] + y1 \ x[2] + z1 \ e[3]; $$
$$ \text{In}[21]:= Z = x2 \ e[1] + y2 \ x[2] + z2 \ e[3]; $$
$$ \text{In}[22]:= W = x3 \ e[1] + y3 \ x[2] + z3 \ e[3]; $$
$$ \text{In}[23]:= p = e[0] + X + (\text{Magnitude}[X]^2/2) \ e[\infty]; $$
$$ \text{In}[24]:= p1 = e[0] + Y + (\text{Magnitude}[Y]^2/2) \ e[\infty]; $$
$$ \text{In}[25]:= p2 = e[0] + Z + (\text{Magnitude}[Z]^2/2) \ e[\infty]; $$
$$ \text{In}[26]:= p3 = e[0] + W + (\text{Magnitude}[W]^2/2) \ e[\infty]; $$
$$ \text{In}[27]:= \text{plane} = \text{OuterProduct}[p, p1, p2, p3, e[\infty]] \text{ // FullSimplify} $$

$$ \text{Out}[27]:= -((-x1 \ y2 \ z + x1 \ y3 \ z + x \ y2 \ z1 - x \ y3 \ z1 + x1 \ y \ z2 - x \ y1 \ z2 + x \ y3 \ z2 - x1 \ y3 \ z2 + x3 (-y1 \ z + y2 \ z + y \ z1 - y2 \ z1 - y \ z2 + y1 \ z2) + (-x1 \ y + x \ y1 - x \ y2 + x1 \ y2) \ z3 + x2 (-y3 \ z - y \ z1 + y3 \ z1 + y1 \ (z - z3) + y \ z3)) (e[1, 2, 3] - e[0, 1, 2, 3, \infty])) $$
By equating the coefficient of \(e_0 e_1 e_2 e_3 e_\infty\) to zero, and rearranging terms, the equation of the plane is:

\[
((z_3 - z_2) y_1 + (z_1 - z_3) y_2 + (z_2 - z_1) y_3) x +
((z_2 - z_3) x_1 + (z_3 - z_1) x_2 + (z_1 - z_2) x_3) y +
((y_3 - y_2) x_1 + (y_1 - y_3) x_2 + (y_2 - y_1) x_3) z +
(x_2 y_3 - x_3 y_2) z_1 + (x_3 y_1 - x_1 y_3) z_2 + (x_1 y_2 - x_2 y_1) z_3 = 0.
\] (10)

### 3.4 Direct representation of spheres

A sphere passing through four points \(p_1, p_2, p_3\) and \(p_4\), is represented by

\[
S = p_1 \wedge p_2 \wedge p_3 \wedge p_4,
\]

and its equations is

\[
p \wedge S = 0.
\]

Consider \(x, x_1, x_2, x_3, p, p_1, p_2\) and \(p_3\) from the above examples and \(x_4 = (x_4, y_4, z_4), p_4 = e_0 + x_4 + \frac{x_4^2}{2} e_\infty\).

\[\text{In}[28]:= V = x_4 e[1] + y_4 x[2] + z_4 e[3];\]
\[\text{In}[29]:= p_4 = e[0] + V + (\text{Magnitude}[V]^2/2) e[\infty];\]
\[\text{In}[30]:= \text{sphere} = \text{OuterProduct}[p, p_1, p_2, p_3, p_4]\]

The result produces a large output (omitted), times \(e_1 e_2 e_3 - e_0 e_1 e_2 e_3 e_\infty = I_5\). As an example, consider the sphere passing through the points \(x_1 = (1, -1, 3), x_2 = (4, 1, -2), x_3 = (-1, -1, 1), x_4 = (1, 1, 1)\):

\[\text{In}[31]:= \text{sphere} = \text{sphere} /\{x_1-> 1, y_1-> -1, z_1-> 3, x_2-> 4, y_2-> 1, z_2-> -2, x_3-> -1, y_3-> -1, z_3-> 1, x_4-> 1, y_4-> 1, z_4-> 1\}\\
\text{Out}[31]:= -12 (-4+(-5+x) x+y (5+y)+z^2) (e[1, 2, 3] - e[0, 1, 2, 3, \infty])\]
\[\text{In}[32]:= \text{sphere} = \text{Coefficient}[\text{sphere}, (e[1, 2, 3] - e[0, 1, 2, 3, \infty])]\]
\[\text{Out}[32]:= -12 (-4+(-5+x) x+y (5+y)+z^2)\]

The equation of the sphere is then:

\[
(x - 5)x + (y + 5)y + (z + 1)z - 4 = 0.
\]

And

\[\text{In}[33]:= \text{ContourPlot3D}[\text{sphere} == 0,\{x,-4,7\},\{y,-7,4\},\{z,-5,5\}, \text{Mesh->None},\text{ContourStyle} -> \text{Directive[Opacity[0.8], Specularity[White, 30]]}]\]

produces the sphere displayed in Fig.I.
3.5 Dual representation of planes

Consider a plane passing through the points $p_1$, $p_2$ and $p_3$ given in Equations 9(b)-(d). This plane is represented by the blade (7):

\[
\begin{align*}
\text{In[34]:= } & P = \text{OuterProduct}[p_1, p_2, p_3, e[\infty]] \quad \text{// GFactor} \\
\text{Out[34]:= } & (-x_2 y_1 + x_3 y_1 + x_1 y_2 - x_3 y_2 - x_1 y_3 + x_2 y_3) e[1, 2] \\
& + (-x_2 z_1 + x_3 z_1 + x_1 z_2 - x_3 z_2 - x_1 z_3 + x_2 z_3) e[1, 3] \\
& + (-y_2 z_1 + y_3 z_1 + y_1 z_2 - y_3 z_2 - y_1 z_3 + y_2 z_3) e[2, 3] \\
& + (-x_2 y_1 + x_3 y_1 + x_1 y_2 - x_3 y_2 - x_1 y_3 + x_2 y_3) e[0, 1, 2, \infty] \\
& + (-x_2 z_1 + x_3 z_1 + x_1 z_2 - x_3 z_2 - x_1 z_3 + x_2 z_3) e[0, 1, 3, \infty] \\
& + (-y_2 z_1 + y_3 z_1 + y_1 z_2 - y_3 z_2 - y_1 z_3 + y_2 z_3) e[0, 2, 3, \infty] \\
& + (-x_3 y_2 z_1 + x_2 y_3 z_1 + x_1 y_2 z_3 - x_2 y_1 z_3 + x_1 y_2 z_3) e[1, 2, 3, \infty]
\end{align*}
\]

Where GFactor[] is a function defined in the package that groups terms with common e[..]’s.

The dual $P^*$ can be calculated using the Dual[] function defined in the package, or:

\[
\begin{align*}
\text{In[35]:= } & P_{\text{dual}} = -\text{InnerProduct}[P, I_5] \\
\text{Out[34]:= } & (y_2 z_1 - y_3 z_1 - y_1 z_2 + y_3 z_2 + y_1 z_3 - y_2 z_3) e[1] \\
& + (-x_2 z_1 + x_3 z_1 + x_1 z_2 - x_3 z_2 - x_1 z_3 + x_2 z_3) e[2]
\end{align*}
\]
+ (x2 y1-x3 y1-x1 y2+x3 y2+x1 y3-x2 y3) e[3]
+ (x3 y2 z1-x2 y3 z1-x3 y1 z2+x1 y3 z2+x2 y1 z3-x1 y2 z3) e[∞]

By defining $h$ and $\mathbf{n} = (n_1, n_2, n_3)$, where

\[
\begin{align*}
n_1 &= y_2 z_1 - y_3 z_1 - y_1 z_2 + y_3 z_2 + y_1 z_3 - y_2 z_3, \\
    &= (z_3 - z_2) y_1 + (z_1 - z_3) y_2 + (z_2 - z_1) y_3 \\
n_2 &= x_3 z_1 - x_2 z_1 + x_1 z_2 - x_3 z_2 - x_1 z_3 + x_2 z_3, \\
    &= (z_2 - z_3) x_1 + (z_3 - z_1) x_2 + (z_1 - z_2) x_3 \\
n_3 &= x_3 y_1 - x_2 y_1 - x_1 y_2 + x_3 y_2 + x_1 y_3 - x_2 y_3, \\
    &= (y_3 - y_2) x_1 + (y_1 - y_3) x_2 + (y_2 - y_1) x_3 \\
h &= (z_3 y_2 z_1 - z_2 y_3 z_1 - z_1 y_3 z_2 + x_1 y_3 z_2 + x_2 y_1 z_3 - x_1 y_2 z_3), \\
    &= (x_3 y_2 - x_2 y_3) z_1 + (x_1 y_3 - x_3 y_1) z_2 + (x_2 y_1 - x_1 y_2) z_3,
\end{align*}
\]

then

\[ P^* = n_1 e_1 + n_2 e_2 + n_3 e_3 - h e_∞ = \mathbf{n} - h e_∞. \]

If $x = (x, y, z)$ is a point in the plane, represented by $p$ in (9a), the equation of the plane is

\[ p \cdot P^* = 0, \]

\begin{Verbatim}
In[36]:= pdual = n1 e[1] + n2 e[2] + n3 e[3] - h e[∞];
In[37]:= InnerProduct[p, pdual]
Out[37]:= -h + n1 x + n2 y + n3 z
\end{Verbatim}

that is

\[ n_1 x + n_2 y + n_3 z - h = 0, \]

which, as expected, equals (10).

3.6 Dual representation of spheres

Consider the point $p_1$ given in (9b); a sphere of radius $r$ centered at $p_1$ is represented as

\[ S = p_1 - \frac{r^2}{2} e_∞. \] (11)

If $x = (x, y, z)$ is a point in the sphere, represented by $p$ in (9a), we notice that $p \cdot S$ is
That is \( p \cdot S = \left( r^2 - \|x - x_1\|^2 \right) / 2 \), and \( p \cdot S = 0 \):

\[
\|x - x_1\|^2 = r^2,
\]

is the equation of the sphere with center \( x_1 \) and radius \( r \), and \( S \) in (11) is the dual representation of this sphere.

4 Transformations

One of the advantages of the conformal model is that conformal transformations in \( \mathbb{R}^3 \) can be represented by orthogonal transformations in \( \mathbb{G}^{4,1} \). In what follows CGAlgebra is applied to describe translations, rotations and rigid motions.

4.1 Translations

Translations are obtained as reflections by two parallel planes. The translation of a point \( p \) in \( \mathbb{R}^3 \) by the vector \( t = t_1 e_1 + t_2 e_2 + t_3 e_3 \) is

\[
T_t \, p \, T_t^{-1},
\]

where

\[
T_t = 1 - \frac{1}{2} te_\infty.
\]

We can easily check that \( T_t^{-1} = T_{-t} \):

\[
\text{In}[40] := t = t_1 e[1] + t_2 e[2] + t_3 e[3];
\]
\[
\text{In}[41] := T_t = 1\text{-GeometricProduct}[t,e[\infty]]/2;
\]
\[
\text{Out}[41] := 1 - (t_1 e[1,\infty]- t_2 e[2,\infty]- t_3 e[3,\infty])/2;
\]
\[
\text{In}[42] := T_{ti} = \text{MultivectorInverse}[T_t];
\]
\[
\text{Out}[42] := 1+ (t_1 e[1,\infty]+ t_2 e[2,\infty]+ t_3 e[3,\infty])/2;
\]
\[
\text{In}[43] := \text{T_{ti} == 1-1/2 GeometricProduct[-t,e[\infty]] \quad \text{// Simplify}}
\]
\[
\text{Out}[43] := \text{True}
\]

where the function \text{MultivectorInverse[]} defined in the package returns the Inverse (if it exists) of a multivector. Some simple examples are as follows.
• Translation of the origin \( e_0 \) by the vector \( t \):

\[
\text{In}[44]:= \text{GeometricProduct}[T_t, e[0], T_{ti}];
\text{Out}[44]:= e[0] + t_1 e[1] + t_2 e[2] + t_3 e[3] + \frac{1}{2}(t_1^2 + t_2^2 + t_3^2) e[\infty]
\]

thus \( T_t e_0 T_t^{-1} = e_0 + t + \frac{1}{2}t^2e_\infty = t \), where \( t \) is the representation of \( t \) in \( \mathbb{G}^{4,1} \).

• Translation of the infinity \( e_\infty \):

\[
\text{In}[45]:= \text{GeometricProduct}[T_t, e[\infty], T_{ti}];
\text{Out}[45]:= e[\infty]
\]

• Translation of a vector \( x = (x_1, x_2, x_3) \):

\[
\text{In}[46]:= x = x_1 e[1] + x_2 e[2] + x_3 e[3];
\text{In}[47]:= \text{GeometricProduct}[T_t, x, T_{ti}];
\text{Out}[47]:= x_1 e[1] + x_2 e[2] + x_3 e[3] + (t_1 x_1 + t_2 x_2 + t_3 x_3) e[\infty]
\]

that is, \( T_t x T_t^{-1} = x + (x \cdot t) e_\infty \).

### 4.2 Rotations

The rotation of a vector \( x \in \mathbb{R}^3 \) around an axis orthogonal to the vectors \( a \) and \( b \) is given by

\[
x' = R x R^{-1}, \tag{12}
\]

where \( R = ab \).

The rotation plane is specified by the vectors \( a \) and \( b \) and if a rotation angle \( \phi \) is given, then

\[
x' = \left( \cos \left( \phi/2 \right) - \hat{A} \sin \left( \phi/2 \right) \right) x \left( \cos \left( \phi/2 \right) + \hat{A} \sin \left( \phi/2 \right) \right),
\]

where

\[
\hat{A} = \frac{a \wedge b}{|a \wedge b|}.
\]

In (12) the rotation angle is the angle between \( a \) and \( b \).

We can easily verify that \( e_0 \) and \( e_\infty \) are not affected by a rotation:
In[48]:= a = a1 e[1] + a2 e[2] + a3 e[3];
In[49]:= b = b1 e[1] + b2 e[2] + b3 e[3];
In[50]:= R = GeometricProduct[a, b];
In[51]:= GeometricProduct[R, e[0], MultivectorInverse[R]];
Out[51]:= e[0]
In[52]:= GeometricProduct[R, e[∞], MultivectorInverse[R]];
Out[52]:= e[∞]

5 Additional predefined functions

The basic functions listed in Table 1 are the starting point to define more functions depending on the applications. To make thing easier, some complementary useful functions are defined in CGAlgebra, and they are listed in Table 2.

With the basic and complementary functions described in Tables 1 and 2 any other required function can be built to enlarge the package using the capability of the Wolfram language to define our own functions. Consider for example the sphere inversion.

Let be an sphere with center and radius . From (11) we have:

\[ S = p - \frac{r^2 e_∞}{2}. \]

The inversion of a point \( x \) in \( \mathbb{G}^{4,1} \), with respect to the sphere \( S \) is

\[ -SxS^{-1}. \]

The Mathematica expression that represents the inversor operator is defined as follows:

In[53]:= Clear[x,p,r]
In[54]:= Inversor[x_, p_, r_] :=
-GeometricProduct[p-r^2 e[∞]/2, x, MultivectorInverse[p-r^2 e[∞]/2]]

Thus, the user-defined function \( \text{Inversor}[x, p, r] \) inverts a point \( x \) with respect to a sphere centered at \( p \) and with radius \( r \). Let us try \( \text{Inversor[]} \) by considering inversions with respect to a sphere centered at the origin, that is \( p = e[0] \).

The inversion of the origin \( e_0 \) is \( e_∞ \):
Table 2: Complementary functions of CGAlgebra

| Expression            | Output                                                                 |
|-----------------------|------------------------------------------------------------------------|
| Magnitude[A]          | $A^2$                                                                   |
| Reversion[A]          | The reversion of the multivector $A$                                   |
| Involution[A]         | The grade involution ($\dagger$) of the multivector $A$.               |
| MultivectorInverse[A] | The inverse (if it exists) of the multivector $A$.                     |
| GradeQ[A,k]           | True if $A$ is a $k$-vector.                                           |
| Dual[A]               | The dual of the multivector $A$                                        |
| Rotation[x,a,b,θ]     | Rotation of the vector $x$ by an angle $θ$ along the plane defined by the vectors $a$ and $b$. The sense of the rotation is from $a$ to $b$. The default value of $θ$ is the angle between $a$ and $b$. |
| ToVector[v]           | The $\mathbb{R}^3$ vector $v=x e[1]+y e[2]+z e[3]$ transformed to the Mathematica input form \{x,y,z\}. |
| GFactor[A]            | Factors terms of the expression $A$ with common $e[i,j,k]$’s.          |
| I5                    | The pseudoscalar $I$ of $\mathbb{G}^{4,1}$                            |
| I5i                   | $I^{-1}$                                                               |

\begin{verbatim}
In[55]:= Inversor[e[0],e[0],r]
Out[55]:= r^2 e[\infty]/2

The infinity $e_\infty$ is inverted to the origin:

In[56]:= Inversor[e[\infty],e[0],r]
Out[56]:= 2 e[0]/r^2

Vectors in $\mathbb{R}^3$ are unaffected by the inversion:

In[57]:= v = v1 e[1] + v2 e[2] + v3 e[3];
In[58]:= Inversor[v,e[0],r]
Out[58]:= v1 e[1] + v2 e[2] + v3 e[3]
\end{verbatim}
In this way many other expressions can be defined. Please feel free to contact the authors if further assistance is required to implement CGAlgebra to a particular application.

References

[1] D. Hestenes, Old wine in new bottles: A new algebraic framework for computational geometry. In: E. Bayro-Corrochano and G. Sobczyk (Editors), Geometric Algebra with Applications in Science and Engineering. (Birkäuser, Boston, 2001). Ch. 1.

[2] H. Li, D. Hestenes, A. Rockwood, Generalized homogeneous coordinates for computational geometry. In: G. Sommer (Ed.), Geometric Computing with Clifford Algebras (Springer-Verlag, Heidelberg, 2000). Ch. 2.

[3] L. Dorst, D. Fontijne and S. Mann, Geometric Algebra for Computer Science (Revised Edition), Morgan Kaufmann Publishers (Burlington, MA, 2009)

[4] C. Perwass, Geometric Algebra with Applications in Engineering. Springer-Verlag (Berlin, 2009).

[5] D. Hildenbrand, Foundations of Geometric Algebra Computing. Springer-Verlag (Berlin, 2013).

[6] K. Kanatani, Understanding Geometric Algebra. Hamilton, Grassmann, and Clifford for Computer Vision and Graphics (CRC Press, Florida USA, 2015).