Space Time PGA in Geometric Algebra G(1,3,1)

Robert Benjamin Easter
Bangkok, Thailand
Email: reaster2015@gmail.com
Scopus SC: 57190277660
ORCID iD: 0000-0002-8725-1835

Daranee Pimchangthong
Assoc. Prof. Dr. at UTK ISIC in Bangkok, Thailand
Email: daranee.p@mail.rmutk.ac.th
Web: isic.rmutk.ac.th
Scopus SC: 35273219500

Abstract

In Geometric Algebra, G(1,3,1) is a degenerate-metric geometric algebra being introduced in this paper as Space Time PGA [STPGA], based on 3D Homogeneous PGA G(3,0,1) [3DPGA] and 4D Conformal Spacetime CGA G(2,4,0) [CSTA]. In CSTA, there are flat (linear) geometric entities for hyperplane, plane, line, and point as inner product null space (IPNS) geometric entities and dual outer product null space (OPNS) geometric entities. The IPNS CSTA geometric entities are closely related, in form, to the STPGA plane-based geometric entities. Many other aspects of STPGA are borrowed and adapted from 3DPGA, including a new geometric entity dualization operation J_e that is an involution in STPGA. STPGA includes operations for spatial rotation, spacetime hyperbolic rotation (boost), and spacetime translation as versor operators. This short paper only introduces the basics of the STPGA algebra. Further details and applications may appear in a later extended paper or in other papers. This paper is intended as a quick and practical introduction to get started, including explicit forms for all entities and operations. Longer papers are cited for further details.

Keywords: PGA, spacetime, geometric algebra, projective, plane-based, point-based, homogeneous coordinates, homogeneous model, geometric entities, rotation, translation, hyperbolic rotation, boost, versor, Hodge star dual

1 Introduction

This short paper[1] introduces the Spacetime Plane-based/Point-based/Projective Geometric Algebra $\mathcal{G}_{1,3,1}$, which we will just call Space Time PGA (STPGA). This paper assumes familiarity with PGA [5] and CGA [8][2] in Geometric Algebra [7].

1. Version v2, 6 Jan 2024. Original version was v1, 25 Dec 2023, uploaded to viXra.org preprint repository. Version v2 corrects Eq.(21) and adds other clarifications.

STPGA has similarities to PGA $\mathcal{G}_{3,0,1}$ [5] and to CSTA $\mathcal{G}_{2,4}$ [4][3]. Only the basics of the algebra are introduced, including the geometric entities for points, lines, planes, and hyperbola's $J_e(\{\mathbf{P},\mathbf{L},\mathbf{\Pi},\mathbf{H}\})=\{\boldsymbol{p},\boldsymbol{l},\boldsymbol{\pi},\boldsymbol{h}\}$ (point-based and plane based), and operations for rotation R, translation T, and hyperbolic rotation (boost) B, and the geometric entity dualization operation J_e. Applications and further details are not discussed, but may be elaborated on in a later extended paper or in other papers.

For STPGA, we will use $\mathcal{G}_{1,3,1}$ with basis 1-blades $\{\mathbf{e}_0,\mathbf{e}_1,\mathbf{e}_2,\mathbf{e}_3,\mathbf{e}_4\}$ and metric $g_{ij}=[\mathbf{e}_i\cdot\mathbf{e}_j]=\mathrm{diag}(0,-1,-1,-1,1)$. The unit vector $\mathbf{e}_0$ is defined as a degenerate null vector, $\mathbf{e}_0^2=0$. The unit vector $\mathbf{e}_4$ is the time-like direction, $\mathbf{e}_4^2=1$. The unit vectors $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ are a basis for 3D space. The unit pseudoscalars are $\mathbf{I}_3=\mathbf{e}_1\mathbf{e}_2\mathbf{e}_3$, $\mathbf{I}_4=\mathbf{I}_3\mathbf{e}_4$, and $\mathbf{I}_5=\mathbf{e}_0\mathbf{I}_4$. In other literature on Space-Time Algebra $\mathcal{G}_{1,3}$ (STA) [6], a different notation is used, where $\{\mathbf{e}_1,\,\mathbf{e}_2,\,\mathbf{e}_3,\,\mathbf{e}_4\}\mathrel{\hat{=}}\{\boldsymbol{\gamma}_1,\,\boldsymbol{\gamma}_2,\,\boldsymbol{\gamma}_3,\,\boldsymbol{\gamma}_0\}$. A vector in spacetime is written $\boldsymbol{t}=w\mathbf{e}_4+x\mathbf{e}_1+y\mathbf{e}_2+z\mathbf{e}_3=w\mathbf{e}_4+\mathbf{t}$, where often we use $w=ct$, with c the speed of light (usually 299792458m/s) and t time (usually seconds).

The paper is organized as follows. In Section 2, we discuss the STPGA operations for taking duals, rotation, hyperbolic rotation (boost), and translation. In Section 3, we discuss the STPGA point-based entities. In Section 4, we discuss the STPGA plane-based entities. In Section 5, we conclude the paper with a summary and final remarks.

2 STPGA Operations

STPGA versor operations include a spatial rotation operator R (rotor), plane-based spacetime translation operator T (translator) valid on plane-based entities, and a spacetime hyperbolic rotation (boost) operator B. The versor operations are found in [4][3] and adapted here for STPGA. Other operations include dualizations and normalizations. The new Geometric Entity Dualization Operation J_e introduced in the recent paper [5] by the authors is easily adapted to STPGA, in which we found two implementations $\mathcal{J}_e$ for J_e, one in $\mathcal{G}_{5,0,0}$ and another in $\mathcal{G}_{1,4,0}$. In STPGA, J_e is a simple involution for dualizing geometric entities between the point-based and planed-based entities of STPGA, making it easy to use the complete whole STPGA algebra and also to extract values and project vectors from the point entities.

2.1 Pseudo-euclidean Normalization

The normalization of a non-null spacetime vector $\boldsymbol{t}=w\mathbf{e}_4+x\mathbf{e}_1+y\mathbf{e}_2+z\mathbf{e}_3=w\mathbf{e}_4+\mathbf{t}$ is the unit

$$\hat{\boldsymbol{t}}=\boldsymbol{t}/\sqrt{|\boldsymbol{t}^2|}=\boldsymbol{t}/\sqrt{|w^2-x^2-y^2-z^2|}=\boldsymbol{t}/\|\boldsymbol{t}\|. \tag{1}$$

For a null spacetime vector $\boldsymbol{t}^2=0$, the normalization is the unit

$$\hat{\boldsymbol{t}}=\boldsymbol{t}/\sqrt{\boldsymbol{t}\cdot\boldsymbol{t}^\dagger}=\boldsymbol{t}/\sqrt{w^2+x^2+y^2+z^2}=\boldsymbol{t}/\|\boldsymbol{t}\|, \tag{2}$$

where

$$\boldsymbol{t}^\dagger=\mathbf{e}_4\boldsymbol{t}\mathbf{e}_4=w\mathbf{e}_4-\mathbf{t} \tag{3}$$

is the conjugate of $\boldsymbol{t}$ [8]. More generally, for any $A \in \mathcal{G}_{1,3}$ in STA, the conjugate of A is $A^\dagger = \mathbf{e}_4 A^\sim \mathbf{e}_4$, which is a sandwiching with $\mathbf{e}_4$ and the reverse $A^\sim$.

The squared unit is $\hat{\boldsymbol{t}}^2 \in \{-1, 0, 1\}$, for $\boldsymbol{t}$ space-like, light-like null, or time-like, respectively. For $\hat{\boldsymbol{t}}^2 = 0$, we have $\hat{\boldsymbol{t}} \cdot \hat{\boldsymbol{t}}^\dagger = 1$, so that $\hat{\boldsymbol{t}}^\dagger$ is the pseudoinverse. The normalization is euclidean for null vectors and pseudo-euclidean for non-null vectors.

2.2 Space Dualization (Involution)

The dual of $A \in \mathcal{G}_{0,3,0} = \mathcal{S}$ in the spatial subalgebra of STA is

$$A^{*_\mathcal{S}} = -A\mathbf{I}_3. \tag{4}$$

Since $\mathbf{I}_3^2 = 1$, the undual is $A = -A^{*_\mathcal{S}}\mathbf{I}_3$, so that the space dualization is an involution.

2.3 Spacetime Dualization (Anti-Involution)

The dual of $A \in \mathcal{G}_{1,3,0} = \mathcal{M}$ in STA is

$$A^{*_\mathcal{M}} = -A\mathbf{I}_4. \tag{5}$$

Since $\mathbf{I}_4^2 = -1$, the undual is $A = A^{*_\mathcal{M}}\mathbf{I}_4$, so that the spacetime dualization is an anti-involution.

2.4 STPGA Geometric Entity Dualization (Involution)

For the geometric entity dualization operation J_e, for dualizing geometric entities between point-based and plane-based forms, we borrow concepts and notation from the recent paper [5].

The dual of the geometric entity $\mathbf{A} \in \mathcal{G}_{1,3,1}$ in STPGA is

$$\mathbf{A}^\star = J_e(\mathbf{A}), \tag{6}$$

which is an involution dualizing between plane-based and point-based STPGA entities. In [5], it is explained how $J_e(\mathbf{A}) = \mathbf{A}^\star$ is a form of Hodge star $\star$ dual, where we use the notation $\mathbf{A}^\star$ instead of $\star\mathbf{A}$. Since $J_e(\mathbf{A})$ it is an involution, there is no distinct undual or inverse and we have

$$J_e(J_e(\mathbf{A})) = \mathbf{A}^{\star\star} = \mathbf{A}. \tag{7}$$

The dualization of an entity gives the dual entity representing the same geometry and with the same geometric orientation. The dualization operation J_e preserves geometric entity orientation. The point-based entities in Section 3 and the plane-based entities in Section 4 have been formulated to give entities with the same orientation, which is maintained through dualization using J_e.

Following concepts and notation introduced in [5], we can implement $J_e(\mathbf{A})$ for STPGA in $\mathcal{G}_{p,q,0} \in \{\mathcal{G}_{5,0,0}, \mathcal{G}_{1,4,0}\}$ as the dualization

$$J_e(\mathbf{A}) = \mathbf{A}^\star = \mathcal{G}_{1,3,1}(\mathcal{J}_e(\mathcal{G}_{p,q,0}(\mathbf{A}))) = \mathcal{G}_{1,3,1}(\mathcal{J}_e(\boldsymbol{A})) = \mathcal{G}_{1,3,1}(\boldsymbol{A}^\star). \tag{8}$$

In $\mathcal{G}_{5,0,0}$ with metric $[\mathbf{e}_i \cdot \mathbf{e}_j] = \mathrm{diag}(1, 1, 1, 1, 1)$,

$$\mathcal{J}_e(\boldsymbol{A}) = \boldsymbol{I}_3\boldsymbol{I}_4\boldsymbol{A}\boldsymbol{I}_5\boldsymbol{I}_4\boldsymbol{I}_3 = \boldsymbol{e}_4\boldsymbol{A}\boldsymbol{I}_3\boldsymbol{e}_0 = \boldsymbol{I}_3\boldsymbol{I}_4\boldsymbol{I}_5\boldsymbol{A}\boldsymbol{I}_4\boldsymbol{I}_3 = \boldsymbol{I}_3\boldsymbol{e}_0\boldsymbol{A}\boldsymbol{e}_4. \tag{9}$$

In $\mathcal{G}_{1,4,0}$ with metric $[\mathbf{e}_i \cdot \mathbf{e}_j] = \mathrm{diag}(-1,-1,-1,-1,1)$,

$$\mathcal{J}_e(\boldsymbol{A}) = -\boldsymbol{I}_5\boldsymbol{A} = -\boldsymbol{A}\boldsymbol{I}_5. \quad (10)$$

In these formulas, $\boldsymbol{A} = \mathcal{G}_{p,q,0}(\mathbf{A})$, $\boldsymbol{I}_n = \mathcal{G}_{p,q,0}(\mathbf{I}_n)$, $\boldsymbol{e}_i = \mathcal{G}_{p,q,0}(\mathbf{e}_i)$, and $\mathbf{A} = \mathcal{G}_{1,3,1}(\boldsymbol{A})$ transfer coordinates onto corresponding basis blades in the indicated algebra. The element $\boldsymbol{A} \in \mathcal{G}_{p,q,0}$ (italic bold) denotes $\mathbf{A} \in \mathcal{G}_{1,3,1}$ transferred into the non-degenerate algebra $\mathcal{G}_{p,q,0}$ on corresponding basis blades, and $\mathbf{A} = \mathcal{G}_{1,3,1}(\boldsymbol{A})$ is the transfer back into $\mathcal{G}_{1,3,1}$.

We also tried to find an implementation $\mathcal{J}_e$ in $\mathcal{G}_{3,2,0}$, $\mathcal{G}_{2,3,0}$, and $\mathcal{G}_{4,1,0}$, but no working involution for the dual was found in those three algebras.

$J_e(\mathbf{A})$	$\mathbf{A}\downarrow$	1	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$
$\mathbf{A}\uparrow$	$J_e(\mathbf{A})$	$-\mathbf{I}_5$	$\mathbf{I}_4$	$-\mathbf{e}_0\mathbf{e}_2\mathbf{e}_3\mathbf{e}_4$	$\mathbf{e}_0\mathbf{e}_1\mathbf{e}_3\mathbf{e}_4$	$-\mathbf{e}_0\mathbf{e}_1\mathbf{e}_2\mathbf{e}_4$	$-\mathbf{e}_0\mathbf{e}_1\mathbf{e}_2\mathbf{e}_3$

$J_e(\mathbf{A})$	$\mathbf{A}\downarrow$	$\mathbf{e}_0\mathbf{e}_1$	$\mathbf{e}_0\mathbf{e}_2$	$\mathbf{e}_0\mathbf{e}_3$	$\mathbf{e}_0\mathbf{e}_4$	$\mathbf{e}_1\mathbf{e}_2$	$\mathbf{e}_1\mathbf{e}_3$	$\mathbf{e}_1\mathbf{e}_4$	$\mathbf{e}_2\mathbf{e}_3$	$\mathbf{e}_2\mathbf{e}_4$	$\mathbf{e}_3\mathbf{e}_4$
$\mathbf{A}\uparrow$	$J_e(\mathbf{A})$	$\mathbf{e}_2\mathbf{e}_3\mathbf{e}_4$	$-\mathbf{e}_1\mathbf{e}_3\mathbf{e}_4$	$\mathbf{e}_1\mathbf{e}_2\mathbf{e}_4$	$\mathbf{e}_1\mathbf{e}_2\mathbf{e}_3$	$\mathbf{e}_0\mathbf{e}_3\mathbf{e}_4$	$-\mathbf{e}_0\mathbf{e}_2\mathbf{e}_4$	$-\mathbf{e}_0\mathbf{e}_2\mathbf{e}_3$	$\mathbf{e}_0\mathbf{e}_1\mathbf{e}_4$	$\mathbf{e}_0\mathbf{e}_1\mathbf{e}_3$	$-\mathbf{e}_0\mathbf{e}_1\mathbf{e}_2$

Table 1. Geometric entity dualization $J_e(\mathbf{A}) = \mathbf{A}^\star$ of basis blades in STPGA $\mathcal{G}_{1,3,1}$.

Table 1 shows the geometric entity dualization operation $J_e(\mathbf{A})$ on all $2^5 = 32$ basis blades in STPGA $\mathcal{G}_{1,3,1}$, showing that J_e is an involution. Basis blades below and above each other in the tables dualize to each other back and forth using J_e repeatedly. Using either implementation of J_e as discussed above, it can be verified that the duals are as shown in Table 1, and dual entities can be compared and shown to represent the same geometry and orientation by empirical methods demonstrated in the recent paper [5].

Using $\mathcal{GA}$lgebra [1] for SymPy, we offer the following example Python code for implementing $J_e(\mathbf{A})$ in $\mathcal{G}_{5,0,0}$ as `Je_g500`, and in $\mathcal{G}_{1,4,0}$ as `Je_g140`.

```
# Create the algebras.
g131 = Ga('e*0|1|2|3|4',g=[ 0,-1,-1,-1, 1])
g500 = Ga('e*0|1|2|3|4',g=[ 1, 1, 1, 1, 1])
g140 = Ga('e*0|1|2|3|4',g=[-1,-1,-1,-1, 1])

# Get the basis for STPGA G(1,3,1).
(e0,e1,e2,e3,e4) = g131.mv()
# Create the unit pseudoscalars.
I3 = e1^e2^e3; I4 = I3^e4; I5 = e0^I4

# Entity Dualization Operation Je in G(5,0,0)
def Je_g500(A):
    EA = g500.mv(A)
    EI3 = g500.mv(I3)
    EI4 = g500.mv(I4)
    EI5 = g500.mv(I5)
    return g131.mv(EI3*EI4*EA*EI5*EI4*EI3)

# Entity Dualization Operation Je in G(1,4,0)
```

```
def Je_g140(A):
    EA = g140.mv(A)
    EI5 = g140.mv(I5)
    return g131.mv(-EA*EI5)
```

Using the operation J_e, plane-based points can be dualized to point-based points and then spanned by wedge product (join operation), and point-based planes can be dualized to plane-based planes and intersected by wedge product (meet operation). We can use both algebras as we like.

2.5 Spacetime Translation Operator (Plane-based)

The STPGA spacetime translation operator (translator) is

$$T_{\boldsymbol{d}} = \exp(\mathbf{e}_0 \boldsymbol{d}/2) = 1 + \mathbf{e}_0 \boldsymbol{d}/2 \tag{11}$$

for translation by displacement $\boldsymbol{d}$ in spacetime. The translation operator can be used only on the plane-based geometric entities $\boldsymbol{a} \in \{\boldsymbol{h}, \boldsymbol{\pi}, \boldsymbol{l}, \boldsymbol{p}\}$. Using the geometric entity dualization operation $J_e(\mathbf{A}) = \mathbf{A}^\star = \boldsymbol{a}$, any point-based entity $\mathbf{A} \in \{\mathbf{H}, \mathbf{\Pi}, \mathbf{L}, \mathbf{P}\}$ can be dualized to its plane-based entity of same orientation and then translated as $\mathbf{A}' = (T\mathbf{A}^\star T^{-1})^\star$.

As discussed in some detail in Section 6.6.4 in [3] on the similar CSTA $\mathcal{G}_{2,4}$, it is possible to form the translator T as successive reflections in two parallel hyperplanes $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$ or in two parallel planes $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$.

As in 3DPGA, the translator T is again like a dual quaternion point embedding. It is easily seen that T acts as a versor on the point entity $\boldsymbol{p_t}$ for translation as $\boldsymbol{p_{t+d}} = T\boldsymbol{p_t}T^{-1}$, called a versor sandwich product. Then, since the point $\boldsymbol{p_t}$ is the intersection wedge product (meet) of four hyperplanes, we must accept by versor outermorphism that T also correctly translates the hyperplane entity $\boldsymbol{h}$. The plane and line are also intersection wedge products of two or three hyperplanes, so it must be accepted that they also translate correctly using T by versor sandwich products.

2.6 Spatial Rotation Operator

The STPGA spatial rotation 2-versor operator (rotor) is

$$R = \exp(\theta \mathbf{n}^{*\mathcal{S}}/2) = \cos(\theta/2) + \sin(\theta/2)\hat{\mathbf{n}}^{*\mathcal{S}} \tag{12}$$

for rotation centered on the origin around axis $\hat{\mathbf{n}} \in \mathcal{G}^1_{0,3}$ by angle θ. The rotor R can operate on any geometric entity $\mathbf{A}$ as $\mathbf{A}' = R\mathbf{A}R^{-1}$.

Using the plane-based unit 2-blade spatial line

$$\hat{\boldsymbol{l}}_{\mathbf{p},\hat{\mathbf{d}}} = \mathbf{e}_4 \cdot (\hat{\mathbf{d}}^{*\mathcal{M}} + (\mathbf{p}\cdot\hat{\mathbf{d}}^{*\mathcal{M}})\mathbf{e}_0) = \hat{\mathbf{d}}^{*\mathcal{S}} - (\mathbf{p}\cdot\hat{\mathbf{d}}^{*\mathcal{S}})\mathbf{e}_0 \tag{13}$$

through spatial point $\mathbf{p}$ with spatial axis direction $\hat{\mathbf{d}}$, the rotor for rotation around $\hat{\boldsymbol{l}} = \hat{\boldsymbol{l}}_{\mathbf{p},\hat{\mathbf{d}}}$ by angle θ is

$$R_{\boldsymbol{l}} = \exp(\theta\hat{\boldsymbol{l}}/2) = \cos(\theta/2) + \sin(\theta/2)\hat{\boldsymbol{l}}. \tag{14}$$

The translated rotor $R_{\boldsymbol{l}}$ can be used only on the STPGA plane-based entities, which support translation. The rotor $R_{\boldsymbol{l}}$ can also be formed as successive reflections in spatial planes $\hat{\boldsymbol{h}}_1$ and $\hat{\boldsymbol{h}}_2$ as $R_{\boldsymbol{l}} = \hat{\boldsymbol{h}}_2\hat{\boldsymbol{h}}_1$.

2.7 Spacetime Hyperbolic Rotation (Boost) Operator

The STPGA spacetime hyperbolic rotation (boost) 2-versor operator is

$$B_{\mathbf{v}} = \exp(\varphi_{\mathbf{v}}\hat{\mathbf{v}}\mathbf{e}_4/2) = \cosh(\varphi_{\mathbf{v}}/2) + \sinh(\varphi_{\mathbf{v}}/2)\hat{\mathbf{v}}\mathbf{e}_4 \tag{15}$$

for boost by velocity vector $\boldsymbol{v} = c\mathbf{e}_4 + \beta_{\mathbf{v}}c\hat{\mathbf{v}}$, where $\varphi_{\mathbf{v}} = \operatorname{atanh}(\beta_{\mathbf{v}})$. The rotor $B_{\mathbf{v}}$ can operate on any geometric entity $\mathbf{A}$ as $\mathbf{A}' = B_{\mathbf{v}}\mathbf{A}B_{\mathbf{v}}^{-1}$.

Using the plane-based translator T, a translated boost operator centered on spacetime point $\boldsymbol{p}$ can be derived as

$$B_{\mathbf{v}}^{\boldsymbol{p}} = \exp(\varphi_{\mathbf{v}}\hat{\boldsymbol{\pi}}/2) = \cosh(\varphi_{\mathbf{v}}/2) + \sinh(\varphi_{\mathbf{v}}/2)\hat{\boldsymbol{\pi}}, \tag{16}$$

where $\hat{\boldsymbol{\pi}} = \boldsymbol{D}^{*\mathcal{M}} - (\boldsymbol{p}\cdot\boldsymbol{D}^{*\mathcal{M}})\mathbf{e}_0$ with $\boldsymbol{D} = \hat{\mathbf{v}}\mathbf{e}_4\mathbf{I}_4$, or $\hat{\boldsymbol{\pi}} = \hat{\mathbf{v}}\mathbf{e}_4 - (\boldsymbol{p}\cdot(\hat{\mathbf{v}}\mathbf{e}_4))\mathbf{e}_0$. The translated boost $B_{\mathbf{v}}^{\boldsymbol{p}}$ can only be used on the STPGA plane-based entities, which support translation. The translated boost $B_{\mathbf{v}}^{\boldsymbol{p}}$ is derived and discussed in more detail in Section 6.6.9 in [3]. The boost $B_{\mathbf{v}}^{\boldsymbol{p}}$ can also be formed as successive reflections in two spacetime hyperplanes $\hat{\boldsymbol{h}}_1$ and $\hat{\boldsymbol{h}}_2$ as $B_{\mathbf{v}}^{\boldsymbol{p}} = \hat{\boldsymbol{h}}_2\hat{\boldsymbol{h}}_1$.

3 STPGA Point-based Entities

The STPGA point-based entities are homogeneous geometric entities for point $\mathbf{P}_{\boldsymbol{t}}$, line $\mathbf{L}_{\boldsymbol{p},\boldsymbol{d}}$, plane $\boldsymbol{\Pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2}$, and hyperplane $\mathbf{H}_{\boldsymbol{p},\boldsymbol{n}}$ in spacetime, each representing a geometric outer product null space (OPNS) [8].

3.1 STPGA point-based 1-blade point

The STPGA point-based 1-blade point is defined as

$$\mathbf{P}_{\boldsymbol{t}} = \mathbf{e}_0 + \boldsymbol{t} = J_e(\boldsymbol{p}_{\boldsymbol{t}}) = \boldsymbol{p}_{\boldsymbol{t}}^{\star} \tag{17}$$

embedding spacetime vector point $\boldsymbol{t}$ as a homogeneous geometric point entity. Two points $\mathbf{P}_{\boldsymbol{t}}$ and $\mathbf{P}_{\boldsymbol{p}}$ represent the same point if and only if $\mathbf{P}_{\boldsymbol{t}}\wedge\mathbf{P}_{\boldsymbol{p}} = 0$. The point-based 1-blade point is the dual of the plane-based 4-blade point through the dualization operation $J_e(\mathbf{A})$, which is an involution for STPGA.

In the limit, $\lim_{\|\boldsymbol{t}\|\to\infty}\mathbf{P}_{\boldsymbol{t}}/\|\boldsymbol{t}\| = \hat{\boldsymbol{t}}$, and we take $\hat{\boldsymbol{t}}$ as the point at infinity $\mathbf{P}_{\infty\hat{\boldsymbol{t}}} = \hat{\boldsymbol{t}}$ in the direction $\hat{\boldsymbol{t}}$. The points are homogeneous and can be scaled by any non-zero scalar $a \neq 0$ without affecting the point represented. Therefore, we generally take the infinite point $\mathbf{P}_{\infty\boldsymbol{t}} = \boldsymbol{t}$ as the directed point at infinity in the direction $\hat{\boldsymbol{t}}$ with magnitude $\|\boldsymbol{t}\|$. Points in the form $\mathbf{P}_{\infty\boldsymbol{t}}$ can be used like any other point, or to span directionally.

For any finite point $\mathbf{P}_{\boldsymbol{t}}$, the vector $\boldsymbol{t}$ is projected as

$$\boldsymbol{t} = \frac{J_e(\mathbf{P}_{\boldsymbol{t}}\mathbf{I}_5)}{J_e(\mathbf{P}_{\boldsymbol{t}}\wedge\mathbf{I}_4)}. \tag{18}$$

3.2 STPGA point-based 2-blade line

The STPGA point-based 2-blade line is

$$\mathbf{L}_{\boldsymbol{p},\boldsymbol{d}} = \mathbf{P}_{\boldsymbol{p}} \wedge \boldsymbol{d} = J_e(\boldsymbol{l}_{\boldsymbol{p},\boldsymbol{d}}) = \boldsymbol{l}^{\star}_{\boldsymbol{p},\boldsymbol{d}} \tag{19}$$

for the line through point $\mathbf{P}_{\boldsymbol{p}}$ in the direction $\boldsymbol{d}$ in spacetime. The point $\mathbf{P}_{\boldsymbol{t}}$ is on the line $\mathbf{L}_{\boldsymbol{p},\boldsymbol{d}}$ if and only if $\mathbf{P}_{\boldsymbol{t}} \wedge \mathbf{L}_{\boldsymbol{p},\boldsymbol{d}} = 0$. The line can be formed as the wedge product of two points, representing their span or join.

3.3 STPGA point-based 3-blade plane

The STPGA point-based 3-blade plane is

$$\mathbf{\Pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2} = \mathbf{P}_{\boldsymbol{p}} \wedge \boldsymbol{d}_1 \wedge \boldsymbol{d}_2 = J_e(\boldsymbol{\pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2}) = \boldsymbol{\pi}^{\star}_{\boldsymbol{p},\boldsymbol{d}_1\boldsymbol{d}_2} \tag{20}$$

for the plane through point $\mathbf{P}_{\boldsymbol{p}}$ in the plane with direction $\boldsymbol{d}_2 \wedge \boldsymbol{d}_1$. The point $\mathbf{P}_{\boldsymbol{t}}$ is on the plane $\mathbf{\Pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2}$ if and only if $\mathbf{P}_{\boldsymbol{t}} \wedge \mathbf{\Pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2} = 0$. The plane can also be formed as the wedge product of three points, representing their span or join.

3.4 STPGA point-based 4-blade hyperplane

The STPGA point-based 4-blade hyperplane is

$$\mathbf{H}_{\boldsymbol{p},\boldsymbol{n}} = \boldsymbol{n}^{*\mathcal{M}} \wedge \mathbf{P}_{\boldsymbol{p}} = J_e(\boldsymbol{h}_{\boldsymbol{p},\boldsymbol{n}}) = \boldsymbol{h}^{\star}_{\boldsymbol{p},\boldsymbol{n}} \tag{21}$$

for the hyperplane through point $\mathbf{P}_{\boldsymbol{p}}$ normal to $\boldsymbol{n}$. The point $\mathbf{P}_{\boldsymbol{t}}$ is on the hyperplane $\mathbf{H}_{\boldsymbol{p},\boldsymbol{n}}$ if and only if $\mathbf{P}_{\boldsymbol{t}} \wedge \mathbf{H}_{\boldsymbol{p},\boldsymbol{n}} = 0$. The hyperplane can also be formed as the wedge product of four points, representing their span or join.

4 STPGA Plane-based Entities

The STPGA plane-based entities are homogeneous geometric entities for 4-blade point $\boldsymbol{p}_{\boldsymbol{t}}$, 3-blade line $\boldsymbol{l}_{\boldsymbol{p},\boldsymbol{d}}$, 2-blade plane $\boldsymbol{\pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2}$, and 1-blade hyperplane $\boldsymbol{h}_{\boldsymbol{p},\boldsymbol{n}}$ in spacetime, each representing a geometric commutator product null space (OPNS) [8], except for the line entity $\boldsymbol{l}_{\boldsymbol{p},\boldsymbol{d}}$ that represents a geometric anti-commutator product null space. The entities are (hyper)plane-based since each can be formed as the intersection wedge product (meet) of 1-blade hyperplanes.

The STPGA plane-based entities are similar to the inner product null space (IPNS) Conformal Spacetime Algebra (CSTA) entities [4], which may be less familiar than the more commonly known 3D CGA $\mathcal{G}_{4,1}$ entities. We first give the CSTA entity from [4][3], then give the similar new STPGA entity.

4.1 STPGA plane-based 1-blade hyperplane

The CSTA 1-blade hyperplane is $\boldsymbol{E} = \boldsymbol{n} + (\boldsymbol{p} \cdot \boldsymbol{n})\mathrm{e}_{\infty}$, where $\boldsymbol{n}$ is the normal vector in spacetime and $\boldsymbol{p}$ is any spacetime point on the hyperplane.

Similarly, the STPGA 1-blade hyperplane is found to be

$$\boldsymbol{h}_{\boldsymbol{p},\boldsymbol{n}} = \boldsymbol{n} + (\boldsymbol{p}\cdot\boldsymbol{n})\mathrm{e}_0 = J_e(\mathbf{H}_{\boldsymbol{p},\boldsymbol{n}}) = \mathbf{H}^{\star}_{\boldsymbol{p},\boldsymbol{n}}. \tag{22}$$

The 1-blade plane-based hyperplane is the dual of the 4-blade point-based hyperplane. The point $\boldsymbol{p_t}$ is on the hyperplane if and only if $\boldsymbol{p_t}\times\boldsymbol{h}_{\boldsymbol{p},\boldsymbol{n}} = 0$.

A hyperplane can be thought of as fixing 1 of the 4 spacetime coordinates along the given normal direction $\boldsymbol{n}$ and then 3 free coordinates remain to span the orthogonal 3D hyperplane space.

When we limit $\boldsymbol{n}$ and $\boldsymbol{p}$ to 3D space (rejecting the $\mathbf{e}_4$ time component), the hyperplane acts as just a spatial plane in the spatial subalgebra $\mathcal{G}_{0,3,1}$ of STPGA, and the wedge of two spatial planes forms a spatial line $\boldsymbol{l}_{\mathbf{p},\mathbf{d}}$ as discussed further in Section 4.3. The details of the spatial subalgebra are very similar to Section 3.2 in [3], where it is called Conformal Space Algebra (CSA). The CSA plane of Section 3.2.2 in [3] becomes in STPGA the unit spatial plane

$$\hat{\boldsymbol{h}}_{\mathbf{p},\mathbf{n}} = \hat{\mathbf{n}} + (\mathbf{p}\cdot\hat{\mathbf{n}})\mathrm{e}_0. \tag{23}$$

4.2 STPGA plane-based 2-blade plane

The CSTA 2-blade plane is $\mathbf{\Pi} = \boldsymbol{D}^{*\mathcal{M}} - (\boldsymbol{p}\cdot\boldsymbol{D}^{*\mathcal{M}})\mathrm{e}_\infty$, where 2-blade $\boldsymbol{D}$ is the spacetime plane direction and $\boldsymbol{p}$ is any spacetime point on the plane. The dual is $\boldsymbol{D}^{*\mathcal{M}} = \boldsymbol{D}\mathbf{I}_4^{-1} = -\boldsymbol{D}\mathbf{I}_4$, the spacetime (Minkowski) dualization.

Similarly, the STPGA 2-blade plane is

$$\boldsymbol{\pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2} = \boldsymbol{D}^{*\mathcal{M}} - (\boldsymbol{p}\cdot\boldsymbol{D}^{*\mathcal{M}})\mathrm{e}_0 = J_e(\mathbf{\Pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2}) = \mathbf{\Pi}^{\star}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2}, \tag{24}$$

where $\boldsymbol{D}^{*\mathcal{M}} = -(\boldsymbol{d}_2\wedge\boldsymbol{d}_1)\mathbf{I}_4$. The two given direction vectors $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ are assumed to span the plane with right-handed orientation from $\boldsymbol{d}_1$ to $\boldsymbol{d}_2$ (like x and y axes in a right-handed plane). Then, this orientation matches the orientation of STPGA point-based 3-blade plane $\mathbf{\Pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2}$ with same parameters. The 2-blade plane-based plane is the dual of the 3-blade point-based plane. The point $\boldsymbol{p_t}$ is on the plane if and only if $\boldsymbol{p_t}\times\boldsymbol{\pi}_{\boldsymbol{p},\boldsymbol{d}_1,\boldsymbol{d}_2} = 0$.

The plane can also be formed as the intersection wedge product (meet) of two hyperplanes as $\boldsymbol{\pi} = \boldsymbol{h}_1\wedge\boldsymbol{h}_2$, where each hyperplane fixes a coordinate along an orthogonal direction to the plane, in effect fixing 2 coordinates of any point on the plane with the remaining 2 free coordinates spanning the plane in the 2 orthogonal plane directions. The orientation may be $\pm$ depending on the order of the hyperplanes in the wedge product.

4.3 STPGA plane-based 3-blade line

The CSTA 3-blade line is $\boldsymbol{L} = \boldsymbol{d}^{*\mathcal{M}} + (\boldsymbol{p}\cdot\boldsymbol{d}^{*\mathcal{M}})\mathrm{e}_\infty$, where $\boldsymbol{d}$ is the line direction in spacetime, $\boldsymbol{p}$ is any spacetime point on the line, and $\boldsymbol{d}^{*\mathcal{M}} = \boldsymbol{d}\mathbf{I}_4^{-1} = -\boldsymbol{d}\mathbf{I}_4$ (spacetime dualization).

Similarly, the STPGA 3-blade line is

$$\boldsymbol{l}_{\boldsymbol{p},\boldsymbol{d}} = \boldsymbol{d}^{*\mathcal{M}} + (\boldsymbol{p}\cdot\boldsymbol{d}^{*\mathcal{M}})\mathrm{e}_0 = J_e(\mathbf{L}_{\boldsymbol{p},\boldsymbol{d}}) = \mathbf{L}^{\star}_{\boldsymbol{p},\boldsymbol{d}}. \tag{25}$$

The 3-blade plane-based line is the dual of the 2-blade point-based line. The point $\boldsymbol{p_t}$ is on the line if and only if $\boldsymbol{p_t} \overline{\times} \boldsymbol{h_{p,n}} = 0$. This entity uses the symmetric anti-commutator product $A \overline{\times} B = \frac{1}{2}(AB + BA)$, making it different than the other entities that use the anti-symmetric commutator product $A \times B = \frac{1}{2}(AB - BA)$.

The line $\boldsymbol{l}$ can be formed as the intersection wedge product (meet) of three hyperplanes as $\boldsymbol{l} = \boldsymbol{h}_1 \wedge \boldsymbol{h}_2 \wedge \boldsymbol{h}_3$, where each hyperplane fixes a coordinate along an orthogonal direction to the line, in effect fixing a 3 coordinates of any point on the line with the remaining 1 free coordinate spanning the line in the orthogonal line direction. The orientation may be $\pm$ depending on the order of the hyperplanes in the wedge product.

When $\hat{\boldsymbol{d}}$ and $\boldsymbol{p}$ are spatial (rejecting the $\mathbf{e}_4$ time component), forming a unit spatial line, we can contract $\boldsymbol{l_{p,d}}$ to a 2-blade spatial line as $\mathbf{e}_4 \cdot \boldsymbol{l_{p,d}}$ for use as an axis of spatial rotation as shown in Section 2.6. In Section 3.2.3 in [3], the CSA 2-blade line can be adapted more directly into STPGA as the unit 2-blade spatial line

$$\mathbf{e}_4 \cdot \hat{\boldsymbol{l}}_{\boldsymbol{p},\hat{\boldsymbol{d}}} = \hat{\boldsymbol{l}}_{\mathbf{p},\hat{\mathbf{d}}} = \hat{\mathbf{d}}^{*\mathcal{S}} - (\mathbf{p} \cdot \hat{\mathbf{d}}^{*\mathcal{S}})\mathbf{e}_0. \tag{26}$$

The point $\boldsymbol{p_t}$ is on the spatial 2-blade line $\hat{\boldsymbol{l}}_{\mathbf{p},\hat{\mathbf{d}}}$ if and only if $\boldsymbol{p_t} \times \hat{\boldsymbol{l}}_{\mathbf{p},\hat{\mathbf{d}}} = 0$. The 3-blade spacetime line $\hat{\boldsymbol{l}}_{\boldsymbol{p},\hat{\boldsymbol{d}}}$ is anti-commutator product null space, but the spatial line $\hat{\boldsymbol{l}}_{\mathbf{p},\hat{\mathbf{d}}}$ is commutator product null space.

There is a complete spatial PGA subalgebra $\mathcal{G}_{0,3,1}$ that we have not given in detail in this paper, but it should be straightforward to adapt it into STPGA from the similar Conformal Space Algebra $\mathcal{G}_{1,4}$ (CSA) of Section 3.2 in [3]. We take and adapt the similar entities from CSA into the spatial subalgebra of STPGA as we need them, replacing $\mathbf{e}_\infty$ with $\mathbf{e}_0$.

4.4 STPGA plane-based 4-blade point

The STPGA plane-based 4-blade point, embedding the spacetime vector point $\boldsymbol{t}$, is

$$\boldsymbol{p_t} = (1 + \mathbf{e}_0 \boldsymbol{t})\mathbf{I}_4 = J_e(\mathbf{P}_{\boldsymbol{t}}) = \mathbf{P}_{\boldsymbol{t}}^{\star}. \tag{27}$$

The 4-blade plane-based point is the dual of the 1-blade point-based point. Two points $\boldsymbol{p_t}$ and $\boldsymbol{p_p}$ represent the same point if and only if $\boldsymbol{p_t} \times \boldsymbol{p_p} = 0$.

The point can be formed as the intersection wedge product (meet) of four hyperplanes as $\boldsymbol{p_t} = \boldsymbol{h}_1 \wedge \boldsymbol{h}_2 \wedge \boldsymbol{h}_3 \wedge \boldsymbol{h}_4$, where each hyperplane fixes a coordinate along a direction in spacetime. For example, let $\boldsymbol{h}_1 = \boldsymbol{h}_{x\mathbf{e}_1,\mathbf{e}_1}$, $\boldsymbol{h}_2 = \boldsymbol{h}_{y\mathbf{e}_2,\mathbf{e}_2}$, $\boldsymbol{h}_3 = \boldsymbol{h}_{z\mathbf{e}_3,\mathbf{e}_3}$, and $\boldsymbol{h}_4 = \boldsymbol{h}_{w\mathbf{e}_4,\mathbf{e}_4}$. Then, $\boldsymbol{p_t} = \boldsymbol{h}_1 \wedge \boldsymbol{h}_2 \wedge \boldsymbol{h}_3 \wedge \boldsymbol{h}_4$ exactly.

The form of the STPGA point $\boldsymbol{p_t}$ is very similar to the 3DPGA point, including $1 + \mathbf{e}_0 \boldsymbol{t}$, except that now $\boldsymbol{t}$ is a spacetime vector. We do not have dual quaternions in the even subalgebra as in 3DPGA. It is more complicated, but there is an algebra using $1 + \mathbf{e}_0 \boldsymbol{t}$ as the point entity. This other algebra has yet to be fully studied. There are many identities for writing $\boldsymbol{p}_t$ and one is $\boldsymbol{p}_t = (1 + \mathbf{I}_5(\boldsymbol{t}\mathbf{I}_4))\mathbf{I}_4$, where $\mathbf{I}_5 \mathrel{\hat{=}} \varepsilon$, and $-\boldsymbol{t}\mathbf{I}_4 = \boldsymbol{t}^{*\mathcal{M}}$, which is a 3-blade. These elements can be compared to the dual quaternion elements, but it gets very different.

The vector $\boldsymbol{t}$ can be projected as

$$\boldsymbol{t} = \frac{J_e(J_e(\boldsymbol{p_t})\mathbf{I}_5)}{J_e(J_e(\boldsymbol{p_t}) \wedge \mathbf{I}_4)}. \tag{28}$$

5 Conclusion

This paper introduced Space Time PGA $\mathcal{G}_{1,3,1}$, which extends 3D PGA to spacetime.

In Section 1, we introduced the subject of this paper, which is Space Time PGA, an extension of the 3D PGA to a 4D pseudo-euclidean spacetime. We borrowed and adapted many results from papers [4][3] written by the first author some years ago, and from the very recent paper [5] by the authors. STPGA includes a hyperbolic rotation (boost) versor, and also the rotor and translator that are in 3D PGA. In STPGA, we have geometric entities for hyperplane (3D subspace in 4D spacetime), plane (2D subspace in 4D spacetime), line (1D subspace in 4D spacetime), and point (0D subspace in 4D spacetime). Each entity type has a point-based and a plane-based form, related to each other through the geometric entity dualization operation J_e, which is a simple involution.

In Section 2, we discuss the STPGA operations for taking duals, rotation, hyperbolic rotation (boost), and translations. Table 1 shows the dual for each basis blade in STPGA $\mathcal{G}_{1,3,1}$. The geometric entity dualization operation J_e is implemented two ways, either in $\mathcal{G}_{1,4,0}$ or in $\mathcal{G}_{5,0,0}$, so that we do not have to use table lookup for J_e.

In Section 3, we discuss the STPGA point-based entities, which are formed to be dual to the STPGA plane-based entities in same orientation through the dualization operation J_e.

In Section 4, we discuss the STPGA plane-based entities, which are formed to be dual to the STPGA point-based entities in same orientation through the dualization operation J_e. We also briefly discuss the spatial subalgebra of STPGA to include a spatial 1-blade plane and spatial 2-blade line entity, useful as an axis of spatial rotation.

Many more details and applications of STPGA could be given in longer papers, but this short paper introduces the important basics to get started. We have given the geometric entity dualization operation J_e for STPGA as an involution that can be implemented in either of two non-degenerate geometric algebras and that maintains geometric entity orientation through the dualization. The dualization J_e is important to be able to use the algebra fully and seems to be missing in most publications on PGA. We have given explicit forms for all point-based and plane-based algebra entities as a practical guide to using the entities formed with same orientations in the two algebras. We have given explicit formulas for rotation, translation, and boost operators that should be practical to get started with further calculations in spacetime with the entities. We have cited some detailed long papers by the first author that can further elaborate on what has been offered.

We hope STPGA $\mathcal{G}_{1,3,1}$ is found to be interesting and useful for applications and further research.

References

[1] Alan Bromborsky, Utensil Song, Eric Wieser, Hugo Hadfield, and The Pygae Team. Pygae/galgebra: v0.5.0. June 2020.

[2] L. Dorst, D. Fontijne, and S. Mann. *Geometric Algebra for Computer Science (Revised Edition): An Object-Oriented Approach to Geometry*. The Morgan Kaufmann Series in Computer Graphics. Elsevier Science, 2009.

[3] Robert Benjamin Easter. Double Conformal Space-Time Algebra. https://vixra.org/abs/1602.0114, 2016. Accessed: 2023-12-24.

[4] Robert Benjamin Easter and Eckhard Hitzer. Double Conformal Space-Time Algebra. *AIP Conference Proceedings*, 1798(1):20066, 2017. DOI: 10.1063/1.4972658.

[5] Robert Benjamin Easter and Daranee Pimchangthong. Geometric Entity Dualization and Dual Quaternion Geometric Algebra in PGA G(3,0,1) with Double PGA G(6,0,2) for General Quadrics. https://vixra.org/abs/2312.0085, 2023. Accessed: 2023-12-24.

[6] David Hestenes. *Space-Time Algebra*. Springer, Second edition, 2015.

[7] David Hestenes and Garret Sobczyk. *Clifford Algebra to Geometric Calculus, A Unified Language for Mathematics and Physics*, volume 5 of *Fundamental Theories of Physics*. Dordrecht-Boston-Lancaster: D. Reidel Publishing Company, a Member of the Kluwer Academic Publishers Group, 1984.

[8] Christian Perwass. *Geometric Algebra with Applications in Engineering*, volume 4 of *Geometry and Computing*. Springer, 2009. Habilitation thesis, Christian-Albrechts-Universität zu Kiel.