Preprint version v1, January 15, 2024.
Expanded full paper on the Dual Quaternion Geometric Algebra in PGA introduced first in the research monograph *Geometric Entity Dualization and Dual Quaternion Geometric Algebra in PGA G(3,0,1) with Double PGA G(6,0,2) for General Quadrics* dated 2023-12-16.

Dual Quaternion Geometric Algebra in PGA G(3,0,1)

Robert Benjamin Easter and Daranee Pimchangthong

Abstract. In Geometric Algebra, the algebra G(3,0,1) is known as PGA, the plane-based and point-based geometric algebras, or projective geometric algebra, of points, lines, and planes in 3D space. The even-grades subalgebra of PGA, which we call Dual Quaternion Geometric Algebra (DQGA), represents Dual Quaternion Algebra (DQA). In the plane-based algebra of PGA, there are entities for points, lines, and planes and many operations on them, including dualization to the point-based entities, reflections in planes, rotations, translations, projections, rejections, and intersections (meet products). In this paper, we derive a complete set of identities that relate all of the plane-based entities and operations in PGA to their corresponding entities and operations in DQGA. Therefore, this paper contributes into the literature on dual quaternions and PGA the complete details on how to use DQA or DQGA as a geometric algebra of points, lines, and planes with many useful operations. All DQGA entities and operations are defined or derived such that the orientations of the entities are maintained correctly through all of the operations. We also define three new part operators for taking the point, line, or plane part of a dual quaternion, which may improve the computational efficiency of intersection (meet) operations. Dual quaternions already have some applications in computer graphics and kinematics. This paper expands on the understanding of dual quaternions and introduces DQA as a versatile geometric algebra of points, lines, and planes with many new operations that do not appear in prior literature, expanding the possible applications of dual quaternions.

Mathematics Subject Classification (2010). Primary 15A66, 11R52; Secondary 14M17, 51F25.

Keywords. dual quaternions, geometric algebra, homogeneous, point, line, plane.

1. Introduction

This paper[1] is about the Dual Quaternion Geometric Algebra (DQGA) in the even-grades subalgebra $\mathcal{G}^+_{3,0,1}$ of 3D PGA $\mathcal{G}_{3,0,1}$ [14][7][8][9][10][3][5]. We assume the reader is familiar with Geometric Algebra (GA) [12], dual numbers, quaternions, and dual quaternions. We review dual numbers, quaternions, and dual quaternions as we introduce our notations. Familiarity with Conformal Geometric Algebra (CGA) $\mathcal{G}_{4,1,0}$ [4][13] is also recommended.

In this paper, we provide new details on using dual quaternions (§3) for representing points (§3.5.1), lines (§3.5.2), and planes (§3.5.3) as homogeneous geometric entities with many operations on them, including reflection (§3.6), translation (§3.7), rotation (§3.8), intersection (§3.9), projection (§3.10), and rejection (§3.11). This paper contributes many new details and some new operations on dual quaternions that may not be found in prior literature. Another contribution of this paper is that, all entities and operations are defined or derived such that entities maintain the correct orientation ($\pm$ signs) through all operations. However, we disclaim any responsibility for any errors or other problems that may appear in any application of the algebra. The paper is organized as follows.

Section 2 provides a detailed review and introduction to 3D PGA $\mathcal{G}_{3,0,1}$ to provide the foundation for Section 3 on the Dual Quaternion Geometric Algebra. Section 2.1 introduces and gives an overview of 3D PGA $\mathcal{G}_{3,0,1}$ for points, lines, and planes in both the point-based and plane-based algebras of PGA. Section 2.2 is about the outer product null space (OPNS) entities in the point-based algebra of PGA, which we call OPNS PGA. Section 2.3 is about the commutator product null space (CPNS) entities in the plane-based algebra of PGA, which we call CPNS PGA. Section 2.4 is about the PGA operations, including the new PGA geometric entity dualization operation J_e (§2.4.1), translation operator T (§2.4.2), rotation operator R (§2.4.3), join and meet operations (§2.4.4 and §2.4.5), reflection operations (§2.4.6), projection operations (§2.4.7), and rejection operations (§2.4.8).

Section 3, the main contribution of this paper, explores the details of the Dual Quaternion Geometric Algebra (DQGA) in the even-grades subalgebra $\mathcal{G}^+_{3,0,1}$ of PGA. In DQGA, we rediscover many results that may be known in prior published literature on dual quaternions, while we also contribute new details and results on representing lines and planes and various operations on them, derived through identities to the plane-based entities and operations in CPNS PGA. Section 3.1 gives an overview and introduction to DQGA. Section 3.2 reviews Dual Number Algebra (DNA) and gives the details on its representation in $\mathcal{G}^+_{3,0,1}$ as Dual Number Geometric Algebra (DNGA). Section 3.3 reviews Quaternion Algebra (QA) and its representation in $\mathcal{G}^+_{3,0,1}$ as Quaternion Geometric Algebra (QGA). Section 3.4 reviews Dual Quaternion Algebra (DQA) and its representation in $\mathcal{G}^+_{3,0,1}$ as Dual Quaternion Geometric Algebra (DQGA). In Section 3.5, we derive the DQGA geometric entities for

[1]This paper expands on part of [5].

points (§3.5.1), lines (§3.5.2), and planes (§3.5.3). Important new identities are derived that also allow many operations to be correctly converted from the plane-based PGA to DQGA forms. Our line and plane entities are different than in the prior literature by having the correct form and orientation derived through new identities to the plane-based entities of PGA. In Sections 3.6 (reflections), 3.7 (translations), 3.8 (rotations), 3.9 (intersections), 3.10 (projections), and 3.11 (rejections), we derive the DQGA forms of operations by exactly converting them from their plane-based PGA forms by using identities to DQGA. All of these operations maintain correct orientation. In Section 3.12, we conclude the discussion on DQGA.

In Section 4, we conclude the paper with final remarks.

2. The Geometric Algebra PGA G(3,0,1)

In this section, we review and provide an introduction to the geometric algebra known as 3D PGA $\mathcal{G}_{3,0,1}$ for points, lines, and planes in 3D space, which we will simply call PGA. It is named PGA after the Point-based Geometric Algebra and the Plane-based Geometric Algebra within $\mathcal{G}_{3,0,1}$. PGA also stands for Projective Geometric Algebra, which is another name for the algebra $\mathcal{G}_{3,0,1}$.

2.1. Introduction to 3D PGA

In PGA $\mathcal{G}_{3,0,1}$, the unit 1-blade (vector) basis is $\{\mathbf{e}_0, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ with metric $[g_{ij}] = [\mathbf{e}_i \cdot \mathbf{e}_j] = \mathrm{diag}(0, 1, 1, 1)$. Homogenenous coordinates $[w, x, y, z]$ are put onto the vector basis as $w\mathbf{e}_0 + x\mathbf{e}_1 + y\mathbf{e}_2 + z\mathbf{e}_3 = w\mathbf{e}_0 + \mathbf{t}$, embedding 3D vector $\mathbf{t}$ into the hyperplane $w = 1$. The unit pseudoscalar of $\mathcal{G}_{3,0,0}$ is $\mathbf{I}_3 = \mathbf{e}_1\mathbf{e}_2\mathbf{e}_3$, and the unit pseudoscalar of $\mathcal{G}_{3,0,1}$ is $\mathbf{I}_4 = \mathbf{e}_0\mathbf{I}_3$. In the subalgebra $\mathcal{G}_{3,0,0}$, we define the dual of any element $A \in \mathcal{G}_{3,0,0}$ as $A^* = A/\mathbf{I}_3$. In $\mathcal{G}_{3,0,1}$, the dual of $A \in \mathcal{G}_{3,0,1}$ is $A^\star = J_e(A)$ as discussed in Section 2.4.1.

In PGA, there is a Point-based Geometric Algebra and a Plane-based Geometric Algebra, both having point, line, and plane geometric entities. The two PGA algebras are related to each other through the geometric entity dualization operation J_e.

The point-based algebra is similar to the "Algebra in Projective Space" (§7.4 in [1]) that uses $\mathcal{G}_{1,3,0}$. In the point-based geometric algebra, the point entity $\mathbf{P}_\mathbf{t} = \mathbf{e}_0 + \mathbf{t}$ embeds the vector (1-blade) point $\mathbf{t} = x\mathbf{e}_1 + y\mathbf{e}_2 + z\mathbf{e}_3$, the line entity $\mathbf{L} = \mathbf{P}_2 \wedge \mathbf{P}_1$ is the join of two points as a bivector (2-blade), and the plane entity $\mathbf{\Pi} = \mathbf{P}_3 \wedge \mathbf{P}_2 \wedge \mathbf{P}_1$ is the join of three points as a trivector (3-blade). The algebra is called point-based since the entities can be formed as the join of points using the outer (wedge) product $\wedge$. The join of points represents their geometrical span. The span of two points $\{\mathbf{P}_1, \mathbf{P}_2\}$ represents the line through the two points and includes all points $\mathbf{P} = (1-t)\mathbf{P}_1 + t\mathbf{P}_2$ with real parameter t such that $\mathbf{P} \wedge \mathbf{L} = 0$. A point $\mathbf{P}$ is on line $\mathbf{L}$ if and only if $\mathbf{P} \wedge \mathbf{L} = 0$, and we call $\mathbf{L}$ an outer product null space (OPNS) line entity. The span of three points $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3\}$ represents the plane through the three points and includes all points $\mathbf{P} = (1-s)((1-t)\mathbf{P}_1 + t\mathbf{P}_2) + s\mathbf{P}_3$ with real parameters s and t such that $\mathbf{P}\wedge\mathbf{\Pi}=0$. A point $\mathbf{P}$ is on plane $\mathbf{\Pi}$ if and only if $\mathbf{P}\wedge\mathbf{\Pi}=0$, and we call $\mathbf{\Pi}$ an OPNS plane entity. Two homogeneous point entities $\{\mathbf{P}_1,\mathbf{P}_2\}$ represent the same 3D point if and only if $\mathbf{P}_1\wedge\mathbf{P}_2=0$, and so we also call the points OPNS points. Since the point entity $\mathbf{P}$ is homogeneous, then the line $\mathbf{L}$ and plane $\mathbf{\Pi}$ entities are also homogeneous, which means that the entities can be multiplied by any non-zero scalar $a\neq 0$ without affecting the geometry that is represented by an entity. If point $\mathbf{P}_{\mathbf{t}}$ has been scaled as $a\mathbf{P}_{\mathbf{t}}=a\mathbf{e}_0+a\mathbf{t}$, then we must divide by a to recover $\mathbf{t}$. The 3D space is embedded in the hyperplane $\mathbf{e}_0$ ($w=1$). Since the metric of $\mathcal{G}_{3,0,1}$ is degenerate and $\mathbf{e}_0^2=0$, we cannot extract the scalar a from $a\mathbf{e}_0$ by contraction or inner product. The scalar a and vector $\mathbf{t}$ can be obtained from $a\mathbf{P}_{\mathbf{t}}$ by methods using $J_e(\mathbf{A})=\mathbf{A}^\star$.

The plane-based algebra is similar to CGA. In the plane-based geometric algebra, which is dual to the point-based algebra through the entity dualization operation J_e, the homogeneous plane entity $\boldsymbol{\pi}_{\mathbf{p},\mathbf{n}}=\mathbf{n}+(\mathbf{p}\cdot\mathbf{n})\mathbf{e}_0=\mathbf{\Pi}^\star$ is a vector (1-blade), the homogeneous line entity $\boldsymbol{l}=\boldsymbol{\pi}_2\wedge\boldsymbol{\pi}_1=\mathbf{L}^\star$ is the meet of two planes as a bivector (2-blade), and the homogeneous point entity $\boldsymbol{p}=\boldsymbol{\pi}_1\wedge\boldsymbol{\pi}_2\wedge\boldsymbol{\pi}_3=\mathbf{P}^\star$ is the meet of three planes as a trivector (3-blade). The algebra is called plane-based since the entities can be formed as the meet (intersection) of planes using the outer (wedge) product $\wedge$. Each unit plane $\hat{\boldsymbol{\pi}}_{d,\mathbf{n}}=\hat{\mathbf{n}}+d\mathbf{e}_0$ can be seen as fixing one coordinate at $d\hat{\mathbf{n}}$ and leaving a plane in the other two free coordinates perpendicular to and through $d\hat{\mathbf{n}}$, or through point $\mathbf{p}$ with normal $\hat{\mathbf{n}}$. The line $\boldsymbol{l}$ fixes two coordinates, leaving one free. The point $\boldsymbol{p}$ fixes all three coordinates. For example, using $\boldsymbol{\pi}_x=\mathbf{e}_1+x\mathbf{e}_0$, $\boldsymbol{\pi}_y=\mathbf{e}_2+y\mathbf{e}_0$, and $\boldsymbol{\pi}_z=\mathbf{e}_3+z\mathbf{e}_0$, then the point $[1,x,y,z]\hat{=}(\boldsymbol{p}_{\mathbf{t}}=\mathbf{e}_0+\mathbf{t})$ is represented as $\boldsymbol{p}_{\mathbf{t}}=\boldsymbol{\pi}_x\wedge\boldsymbol{\pi}_y\wedge\boldsymbol{\pi}_z$. As it turns out, point $\boldsymbol{p}$ is a point of the line $\boldsymbol{l}$ or plane $\boldsymbol{\pi}$ if and only if $\boldsymbol{p}\times\boldsymbol{l}=0$ or $\boldsymbol{p}\times\boldsymbol{\pi}=0$, respectively. Two points $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ represent the same point if and only if $\boldsymbol{p}_1\times\boldsymbol{p}_2=0$. The product $\times$ is the commutator product, defined as $A\times B=(AB-BA)/2$ for any multivectors A and B. Therefore, we call the plane-based entities commutator product null space (CPNS) entities.

Although there are homogeneous coordinates $[w,\ x,\ y,\ z]$ embedded within the PGA algebra, the geometric entities of PGA allow to work with the geometry more abstractly as geometrical objects (entities) in what is called the coordinate-free geometry of geometric entities and operators. In the point-based algebra, there is the join operation on points. In the plane-based algebra, there is the meet operation on planes. The join of points and the meet of planes are coordinate-free geometrical concepts. The line $\boldsymbol{l}$ is usually seen not as fixing two coordinates but as the meet (intersection) of two planes in a line. The point $\boldsymbol{p}$ is usually seen not as fixing all three coordinates but as the meet of three planes in a point. For both algebras, there is a common rotation operator $R=\exp\left(\frac{\theta}{2}(\hat{\mathbf{n}}/\mathbf{I}_3)\right)=\exp(\theta\hat{\mathbf{n}}^*/2)=\cos(\theta/2)+\sin(\theta/2)\hat{\mathbf{n}}^*$ for rotating any entity $\mathbf{A}$ around axis $\hat{\mathbf{n}}$ by angle θ as the versor operation $\mathbf{A}'=R\mathbf{A}R^{-1}$. Only for entities in the plane-based algebra, there is also the translation operator $T = \exp(\mathbf{e}_0\mathbf{d}/2) = 1 + \mathbf{e}_0\mathbf{d}/2$ for translating any plane-based entity $\boldsymbol{a}$ by displacement $\mathbf{d}$ as the versor operation $\boldsymbol{a}' = T\boldsymbol{a}T^{-1}$. As will be discussed further, the translation operator T represents a homogeneous point entity in the dual quaternion geometric algebra (DQGA). Since the plane-based geometric algebra of PGA includes the translation operator T and also supports many more other operations than the point-based algebra, the plane-based algebra has been defined in most prior literature as being *the* PGA algebra, and the point-based algebra has not been used as much.

Part of the difficulty with trying to use both the plane-based and point-based algebras, often having to choose one or the other (the plane-based being the usual choice), is that the dualization operation, denoted J in the prior literature, was only abstractly described or not implemented correctly in the prior literature, leading to problems with sign changes and changes in orientation when attempting to use the tables of duals or other dualization methods as given in the prior literature or in prior software implementations. The new entity dualization operation J_e resolves these problems, and now we may freely dualize entities between the two algebras without concern about sign or orientation changes. The point-based and plane-based algebras are related to each other as duals through the new geometric entity dualization operation J_e. Using J_e, any point, line, or plane entity can now be transformed (dualized) between the two different point-based or plane-based forms as duals that represent the same geometric entity with the same orientation.

In the following sections, we briefly give the entities and operations of the point-based and plane-based algebras of PGA. More details about how these entities are derived are in [5].

2.2. OPNS PGA Geometric Entities in Point-based Algebra

We refer to the entities of the point-based algebra as the OPNS PGA geometric entities. In this section, we briefly give the three OPNS PGA homogeneous geometric entities: OPNS PGA 1-blade point $\mathbf{P}$, OPNS PGA 2-blade line $\mathbf{L}$, and OPNS PGA 3-blade plane $\mathbf{\Pi}$.

2.2.1. OPNS PGA 1-blade Point Entity. The OPNS PGA 1-blade point entity $\mathbf{P_t}$, embedding vector point $\mathbf{t} = x\mathbf{e}_1 + y\mathbf{e}_2 + z\mathbf{e}_3$, is defined in standard form and orientation as

$$\mathbf{P_t} = \mathbf{e}_0 + \mathbf{t} = D_e(\boldsymbol{p}_\mathbf{t}) = \boldsymbol{p}_\mathbf{t}^{-\star}. \tag{1}$$

The point $\mathbf{P_t}$ represents the homogeneous coordinates $[1, x, y, z]$. The dual is $J_e(\mathbf{P_t}) = \mathbf{P}_\mathbf{t}^\star = \boldsymbol{p}_\mathbf{t}^{-\star\star} = \boldsymbol{p}_\mathbf{t}$, which is the plane-based point $\boldsymbol{p}_\mathbf{t}$ representing the same point with the same orientation. A directed point at infinity is defined as

$$\mathbf{P}_{\infty\hat{\mathbf{t}}} = \lim_{\|\mathbf{t}\|\to\infty} \frac{\mathbf{P_t}}{\|\mathbf{t}\|} = \hat{\mathbf{t}}. \tag{2}$$

More generally, $\mathbf{P}_{\infty\mathbf{t}} = \mathbf{t}$, since points are homogeneous and can be scaled by any non-zero scalar $\|\mathbf{t}\| \neq 0$. For finite point $\mathbf{P_t}$, vector $\mathbf{t}$ can be projected as

$$\mathbf{t} = \mathbf{I}_3(\mathbf{P_t} \wedge \mathbf{e}_0)^\star / (\mathbf{I}_3 \wedge \mathbf{P_t})^\star. \tag{3}$$

2.2.2. OPNS PGA 2-blade Line Entity. The OPNS PGA 2-blade line $\mathbf{L}$ spanning $\{\mathbf{P}_1, \mathbf{P}_2\}$, or through $\mathbf{P} = \mathbf{P}_1$ with direction $\mathbf{d} = \mathbf{P}_2 - \mathbf{P}_1$, is

$$\mathbf{L} = \mathbf{L}_{\mathbf{p},\mathbf{d}} = \mathbf{d} \wedge \mathbf{P} = \mathbf{P}_2 \wedge \mathbf{P}_1 = D_e(\boldsymbol{l}_{\mathbf{p},\mathbf{d}}) = \boldsymbol{l}^{-\star}_{\mathbf{p},\mathbf{d}}. \tag{4}$$

$\mathbf{L}_{\mathbf{p},\mathbf{d}}$ and $J_e(\mathbf{L}_{\mathbf{p},\mathbf{d}}) = \mathbf{L}^{\star}_{\mathbf{p},\mathbf{d}} = \boldsymbol{l}_{\mathbf{p},\mathbf{d}}$ represent the same line with same orientation. If $\mathbf{d} = \hat{\mathbf{d}}$, then $J_e(\mathbf{L}_{\mathbf{p},\hat{\mathbf{d}}}) = \hat{\boldsymbol{l}}_{\mathbf{p},\mathbf{d}} = \hat{\boldsymbol{l}}$ is a plane-based unit line, where $\hat{\boldsymbol{l}}^2 = -1$.

2.2.3. OPNS PGA 3-blade Plane Entity. The OPNS PGA 3-blade plane entity $\mathbf{\Pi}$ through point $\mathbf{P} = \mathbf{e}_0 + \mathbf{p}$ with normal $\mathbf{n}$ is

$$\mathbf{\Pi} = \mathbf{\Pi}_{\mathbf{p},\mathbf{n}} = \mathbf{P} \wedge \mathbf{n}^* = D_e(\boldsymbol{\pi}_{\mathbf{p},\mathbf{n}}) = \boldsymbol{\pi}^{-\star}_{\mathbf{p},\mathbf{n}}. \tag{5}$$

$\mathbf{\Pi}_{\mathbf{p},\mathbf{n}}$ and $J_e(\mathbf{\Pi}_{\mathbf{p},\mathbf{n}}) = \mathbf{\Pi}^{\star}_{\mathbf{p},\mathbf{n}} = \boldsymbol{\pi}_{\mathbf{p},\mathbf{n}}$ represent the same plane with same orientation. If $\mathbf{n} = \hat{\mathbf{n}}$, then $J_e(\mathbf{\Pi}_{\mathbf{p},\hat{\mathbf{n}}}) = \hat{\boldsymbol{\pi}}_{\mathbf{p},\mathbf{n}} = \hat{\boldsymbol{\pi}}$ is a plane-based unit plane, where $\hat{\boldsymbol{\pi}}^2 = 1$.

The plane spanning three points $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3\}$ arranged *clockwise* on the plane is

$$\mathbf{\Pi}_{\circlearrowright} = \mathbf{P}_1 \wedge \mathbf{P}_2 \wedge \mathbf{P}_3. \tag{6}$$

If the points are arranged *counterclockwise* on the plane, then the plane is

$$\mathbf{\Pi}_{\circlearrowleft} = \mathbf{P}_3 \wedge \mathbf{P}_2 \wedge \mathbf{P}_1. \tag{7}$$

The plane with normal $\hat{\mathbf{n}}$ through $\mathbf{p}$, or distance $d = \mathbf{p} \cdot \hat{\mathbf{n}}$ from origin, is

$$\mathbf{\Pi}_{d,\hat{\mathbf{n}}} = \mathbf{e}_0 \wedge \hat{\mathbf{n}}^* + \mathbf{p} \wedge \hat{\mathbf{n}}^* = \mathbf{e}_0 \wedge \hat{\mathbf{n}}^* - (\mathbf{p} \cdot \hat{\mathbf{n}})\mathbf{I}_3 = D_e(\hat{\boldsymbol{\pi}}_{d,\mathbf{n}}) = \hat{\boldsymbol{\pi}}^{-\star}_{d,\mathbf{n}}. \tag{8}$$

2.3. CPNS PGA Geometric Entities in Plane-based Algebra

We refer to the entities of the plane-based algebra as the CPNS PGA geometric entities. In this section, we briefly give the three CPNS PGA homogeneous geometric entities: CPNS PGA 1-blade plane $\boldsymbol{\pi}$, CPNS PGA 2-blade line $\boldsymbol{l}$, and CPNS PGA 3-blade point $\boldsymbol{p}$.

2.3.1. CPNS PGA 1-blade Plane Entity. The CPNS PGA 1-blade plane entity $\boldsymbol{\pi} = \boldsymbol{\pi}_{\mathbf{p},\mathbf{n}}$ with normal $\mathbf{n}$ through $\mathbf{p}$ is

$$\boldsymbol{\pi} = \boldsymbol{\pi}_{\mathbf{p},\mathbf{n}} = \mathbf{n} + (\mathbf{p} \cdot \mathbf{n})\mathbf{e}_0 = J_e(\mathbf{\Pi}_{\mathbf{p},\mathbf{n}}) = \mathbf{\Pi}^{\star}_{\mathbf{p},\mathbf{n}}. \tag{9}$$

If $\mathbf{n} = \hat{\mathbf{n}}$, then $d = \mathbf{p} \cdot \mathbf{n}$ is the distance from the origin and $\boldsymbol{\pi} = \hat{\boldsymbol{\pi}} = \hat{\boldsymbol{\pi}}_{d,\mathbf{n}}$ is a unit plane, where $\hat{\boldsymbol{\pi}}^2 = 1$ and $D_e(\hat{\boldsymbol{\pi}}) = D_e(\hat{\boldsymbol{\pi}}_{d,\mathbf{n}}) = \hat{\boldsymbol{\pi}}^{-\star}_{d,\mathbf{n}} = \mathbf{\Pi}_{d,\hat{\mathbf{n}}}$.

The join of three points is $\boldsymbol{\pi} = (\boldsymbol{p}_3^{-\star} \wedge \boldsymbol{p}_2^{-\star} \wedge \boldsymbol{p}_1^{-\star})^{\star} = \boldsymbol{p}_3 \vee \boldsymbol{p}_2 \vee \boldsymbol{p}_1$.

2.3.2. CPNS PGA 2-blade Line Entity. The CPNS PGA 2-blade line entity $\boldsymbol{l}$ is the meet of two planes as

$$\boldsymbol{l} = \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1. \tag{10}$$

The line through $\mathbf{p}$ in direction $\mathbf{d}$ is

$$\boldsymbol{l} = \boldsymbol{l}_{\mathbf{p},\mathbf{d}} = \mathbf{d}^* - (\mathbf{p}\cdot\mathbf{d}^*)\mathbf{e}_0 = J_e(\mathbf{L}_{\mathbf{p},\mathbf{d}}) = \mathbf{L}^\star_{\mathbf{p},\mathbf{d}}. \tag{11}$$

If $\mathbf{d} = \hat{\mathbf{d}}$, then $\boldsymbol{l} = \hat{\boldsymbol{l}}$ is a unit line, where $\hat{\boldsymbol{l}}^2 = -1$ and $D_e(\hat{\boldsymbol{l}}) = \hat{\boldsymbol{l}}^{-\star} = \mathbf{L}_{\mathbf{p},\hat{\mathbf{d}}}$.

The join of two points is $\boldsymbol{l} = (\boldsymbol{p}_2^{-\star} \wedge \boldsymbol{p}_1^{-\star})^\star = \boldsymbol{p}_2 \vee \boldsymbol{p}_1$.

2.3.3. CPNS PGA 3-blade Point Entity. The CPNS PGA 3-blade point entity $\boldsymbol{p}_{\mathbf{t}}$, embedding $\mathbf{t} = x\mathbf{e}_1 + y\mathbf{e}_2 + z\mathbf{e}_3$, is

$$\boldsymbol{p}_{\mathbf{t}} = (1 + \mathbf{e}_0\mathbf{t})\mathbf{I}_3 = \mathbf{I}_3 + \mathbf{I}_4\mathbf{t} = \mathbf{I}_3 - \mathbf{e}_0\mathbf{t}^* = J_e(\mathbf{P}_{\mathbf{t}}) = \mathbf{P}^\star_{\mathbf{t}}. \tag{12}$$

In dual quaternions, $1 + \mathbf{e}_0\mathbf{t} = 1 + \mathbf{t}^*\mathbf{I}_4 = p_{\mathbf{t}}$ is a homogeneous point (75). The product of two points $\boldsymbol{p}_{\mathbf{a}}\boldsymbol{p}_{\mathbf{b}} = -p_{\mathbf{a}}\overline{p}_{\mathbf{b}} = -p_{\mathbf{a}-\mathbf{b}} = -(1 + \mathbf{e}_0(\mathbf{a} - \mathbf{b}))$ represents their difference. Points of form $\boldsymbol{p}_{\infty\mathbf{t}} = \mathbf{I}_4\mathbf{t} = J_e(\mathbf{t}) = J_e(\mathbf{P}_{\infty\mathbf{t}})$ represent directed infinite points at infinity.

The meet of three planes is the point

$$\boldsymbol{p} = \boldsymbol{\pi}_3 \wedge \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1 = \boldsymbol{\pi} \wedge \boldsymbol{l}. \tag{13}$$

For finite point $\boldsymbol{p}_{\mathbf{t}}$, vector $\mathbf{t}$ can be projected as

$$\mathbf{t} = \mathbf{I}_3(\boldsymbol{p}_{\mathbf{t}}^{-\star} \wedge \mathbf{e}_0)^\star / (\mathbf{I}_3 \wedge \boldsymbol{p}_{\mathbf{t}}^{-\star})^\star. \tag{14}$$

2.4. PGA Operations

In this section, we give the PGA operations for dualization, translation, rotation, join, meet, reflections, projections, and rejections.

2.4.1. Geometric Entity Dualization Operation. Table 1[2] defines the geometric entity dualization operation $J_e(\mathbf{A})$ and gives the dual $J_e(\mathbf{A})$ for each basis blade $\mathbf{A}$ in $\mathcal{G}_{3,0,1}$. The operation $J_e(\mathbf{A})$ is defined to dualize any OPNS PGA grade k entity $\mathbf{A}$, in the point-based algebra, to its dual CPNS PGA grade $4-k$ entity $J_e(\mathbf{A}) = \mathbf{A}^\star = \boldsymbol{a}$, in the plane-based algebra.

$\mathbf{A}$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_0\mathbf{e}_1\mathbf{e}_2$	$\mathbf{e}_0\mathbf{e}_1\mathbf{e}_3$	$\mathbf{e}_0\mathbf{e}_2\mathbf{e}_3$	$\mathbf{I}_3 = \mathbf{e}_1\mathbf{e}_2\mathbf{e}_3$
$J_e(\mathbf{A})$	$\mathbf{I}_3 = \mathbf{e}_1\mathbf{e}_2\mathbf{e}_3$	$\mathbf{e}_0\mathbf{e}_2\mathbf{e}_3$	$-\mathbf{e}_0\mathbf{e}_1\mathbf{e}_3$	$\mathbf{e}_0\mathbf{e}_1\mathbf{e}_2$	$-\mathbf{e}_3$	$\mathbf{e}_2$	$-\mathbf{e}_1$	$-\mathbf{e}_0$

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

TABLE 1. Geometric Entity Dualization Operation $J_e(\mathbf{A}) = \mathbf{A}^\star$ on OPNS PGA grade k basis blade $\mathbf{A}$ (in the point-based algebra) to its dual CPNS PGA grade $4-k$ basis blade $J_e(\mathbf{A}) = \mathbf{A}^\star$ (in the plane-based algebra).

Since J_e is an anti-involution, the inverse (undual) is $J_e^{-1}(\mathbf{A}^\star) = -J_e(\mathbf{A}^\star) = D_e(\mathbf{A}^\star) = \mathbf{A}^{-\star\star} = \mathbf{A}$, dualizing any CPNS PGA grade $4-k$ element

[2] Table 1 was found empirically in [5] to give dual entities with the same orientation.

$\mathbf{A}^{\star} = \boldsymbol{a}$, in the plane-based algebra, to its dual OPNS PGA grade k element $\mathbf{A}$, in the point-based algebra.

We use a form of Hodge star $\star$ notation[3] as superscripts since J_e can be realized or implemented according to the following geometric algebra definition of the Hodge star $\star$ dual in non-degenerate metric algebras that correspond to PGA $\mathcal{G}_{3,0,1}$, which is a degenerate-metric algebra.

The Hodge star $\star$ dualization operation is defined in a non-degenerate geometric algebra $\mathcal{G}_{p,q,0}$, $n = p+q$, by the relation $\boldsymbol{A}\wedge\boldsymbol{B}^{\star} = \boldsymbol{B}\wedge\boldsymbol{A}^{\star} = (\boldsymbol{A}\cdot\boldsymbol{B})^{\star}$ with $\boldsymbol{A}^{\star} = \pm\boldsymbol{A}\boldsymbol{I}_n$, or $\boldsymbol{A}^{\star}\wedge\boldsymbol{B} = \boldsymbol{B}^{\star}\wedge\boldsymbol{A} = (\boldsymbol{A}\cdot\boldsymbol{B})^{\star}$ with $\boldsymbol{A}^{\star} = \pm\boldsymbol{I}_n\boldsymbol{A}$, for k-vectors $\{\boldsymbol{A},\boldsymbol{B}\} \in \mathcal{G}^k_{p,q,0}$, where $\boldsymbol{I}_n$ is the non-degenerate unit pseudoscalar. The dualization $\boldsymbol{A}^{\star}$ takes $\boldsymbol{A}$ of grade k to $\boldsymbol{A}^{\star}$ of grade $n-k$. Using the unit pseudoscalar $\boldsymbol{I}_s$ of any subalgebra of $\mathcal{G}_{p,q,0}$, we may apply sandwiching outermorphism as $\boldsymbol{I}_s\boldsymbol{A}^{\star}(\pm\boldsymbol{I}_s)= \boldsymbol{A}^{\star\prime}$ and $\boldsymbol{I}_s(\boldsymbol{A}\wedge\boldsymbol{B}^{\star})(\pm\boldsymbol{I}_s) = \boldsymbol{I}_s(\boldsymbol{B}\wedge\boldsymbol{A}^{\star})(\pm\boldsymbol{I}_s) = \boldsymbol{I}_s(\boldsymbol{A}\cdot\boldsymbol{B})^{\star}(\pm\boldsymbol{I}_s)$ without breaking the defining relation. The modified dual $\boldsymbol{A}^{\star\prime}$ may also be found as just $\boldsymbol{A}^{\star}$ in a different algebra $\mathcal{G}_{p,q,0}$. We can use $\boldsymbol{A}^{\star}$ or $\boldsymbol{A}^{\star\prime}$ (then renaming $\boldsymbol{A}^{\star\prime}$ as $\boldsymbol{A}^{\star}$) as the dual, whichever fits the application. In our application, we want to find $\boldsymbol{A}^{\star}$ to match Table 1. The operation $\boldsymbol{A}^{\star}$ may be an involution $\boldsymbol{A}^{\star\star} = \boldsymbol{A}$, or it may be an anti-involution $\boldsymbol{A}^{\star\star} = -\boldsymbol{A}$ with undual (inverse) $\boldsymbol{A}^{-\star} = -\boldsymbol{A}^{\star}$ and $\boldsymbol{A}^{-\star\star} = \boldsymbol{A}$, depending on how $\boldsymbol{A}^{\star}$ is formed.

$\mathcal{G}_{p,q,0}$	$\mathcal{G}_{4,0,0}$	$\mathcal{G}_{3,1,0}$	$\mathcal{G}_{1,3,0}$
$\mathcal{J}_e(\boldsymbol{A})$	$\boldsymbol{I}_3\boldsymbol{I}_4\boldsymbol{A}\boldsymbol{I}_3 = \boldsymbol{e}_0\boldsymbol{A}\boldsymbol{I}_3$	$\boldsymbol{I}_4\boldsymbol{A}$	$\boldsymbol{A}\boldsymbol{I}_4$

TABLE 2. Entity dualization $\mathcal{J}_e(\boldsymbol{A})$ in non-degenerate geometric algebras $\mathcal{G}_{p,q,0}$.

Table 2 gives the dualization operations $\mathcal{J}_e(\boldsymbol{A}) = \boldsymbol{A}^{\star} \in \mathcal{G}_{p,q,0}$ that have been found to implement $J_e(\mathbf{A}) = \mathbf{A}^{\star} \in \mathcal{G}_{3,0,1}$ in non-degenerate algebras $\mathcal{G}_{p,q,0}$ that correspond to $\mathcal{G}_{3,0,1}$. The coefficients on the basis blades in the dual $\boldsymbol{A}^{\star} \in \mathcal{G}_{p,q,0}$ are transferred onto corresponding basis blades in $\mathbf{A}^{\star} \in \mathcal{G}_{3,0,1}$. The complete geometric entity dualization operation is $J_e(\mathbf{A}) = \mathbf{A}^{\star} = \mathcal{G}_{3,0,1}(\mathcal{J}_e(\mathcal{G}_{p,q,0}(\mathbf{A})))$, where $\mathcal{G}_{p,q,0}(\mathbf{A}) = \boldsymbol{A} \in \mathcal{G}_{p,q,0}$ and $\mathcal{G}_{3,0,1}(\boldsymbol{A}^{\star}) = \mathbf{A}^{\star} \in \mathcal{G}_{3,0,1}$ are the operations that transfer coordinates between algebras.

The following three Python functions using $\mathcal{GA}$lgebra [2] for SymPy each implement $J_e(\mathbf{A})$ of Table 1 in one of the non-degenerate algebras $\mathcal{G}_{p,q,0} \in \{\mathcal{G}_{4,0,0}, \mathcal{G}_{3,1,0}, \mathcal{G}_{1,3,0}\}$ of Table 2. Only one function is needed, and all three produce Table 1 as required.

```
# Create the algebras.
g301 = Ga('e*0|1|2|3',g=[ 0, 1, 1, 1])
g400 = Ga('e*0|1|2|3',g=[ 1, 1, 1, 1])
g310 = Ga('e*0|1|2|3',g=[-1, 1, 1, 1])
g130 = Ga('e*0|1|2|3',g=[ 1,-1,-1,-1])
```

[3] In other literature, the notation is $\star\mathbf{A}$ as the dual of $\mathbf{A}$. In this paper, the notation is $\mathbf{A}^{\star}$.

```
# Get the basis for PGA G(3,0,1).
(e0,e1,e2,e3) = g301.mv()
# Create the unit pseudoscalars.
I3 = e1^e2^e3; I4 = e0^I3
```

```
# Entity Dualization Operation Je in G(4,0,0)
def Je_g400(A):
  EA = g400.mv(A); EI3 = g400.mv(I3); EI4 = g400.mv(I4)
  return g301.mv(EI3*EI4*EA*EI3)
```

```
# Entity Dualization Operation Je in G(3,1,0)
def Je_g310(A):
  EA = g310.mv(A); EI4 = g310.mv(I4)
  return g301.mv(EI4*EA)
```

```
# Entity Dualization Operation Je in G(1,3,0)
def Je_g130(A):
  EA = g130.mv(A); EI4 = g130.mv(I4)
  return g301.mv(EA*EI4)
```

2.4.2. Plane-based PGA 2-versor Translation Operator. The plane-based PGA 2-versor translation operator, called a translator, is defined as

$$T = T_{\mathbf{d}} = \exp(\mathbf{e}_0\mathbf{d}/2) = 1 + \mathbf{e}_0\mathbf{d}/2, \tag{15}$$

for translation of any plane-based geometric entity $\boldsymbol{a}$ by displacement $\mathbf{d}$ as $\boldsymbol{a}' = T\boldsymbol{a}T^{-1}$, where $T^{-1} = T^{\sim}$ (the reverse). The translator is the same as a dual quaternion point (75) $T = p_{\mathbf{d}/2}$, where $T^{-1} = \overline{T} = T^{\dagger} = p_{-\mathbf{d}/2}$.

The translation operator can also be formed as successive reflections in two parallel planes, in $\boldsymbol{\pi}_1$ and then in $\boldsymbol{\pi}_2$, that are separated by $\mathbf{d}/2$ so that $\boldsymbol{\pi}_1 = \hat{\mathbf{d}} + d_1\mathbf{e}_0$ and $\boldsymbol{\pi}_2 = \hat{\mathbf{d}} + (d_1 + \|\mathbf{d}\|/2)\mathbf{e}_0$. The translator $T_{\mathbf{d}}$ is then

$$T_{\mathbf{d}} = \boldsymbol{\pi}_2\boldsymbol{\pi}_1 = \boldsymbol{\pi}_2 \cdot \boldsymbol{\pi}_1 + \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1 = 1 + \mathbf{e}_0\mathbf{d}/2 = \exp(\mathbf{e}_0\mathbf{d}/2). \tag{16}$$

The translator T cannot be used on any of the point-based entities. To translate a point-based entity $\mathbf{A}$, we must dualize it using $J_e(\mathbf{A}) = \mathbf{A}^{\star}$ into a plane-based entity $\mathbf{A}^{\star}$, translate it as a plane-based entity $\mathbf{A}^{\star\prime} = T\mathbf{A}^{\star}T^{-1}$, and then dualize it back to a point-based entity using $-J_e\left(\mathbf{A}^{\star\prime}\right) = D_e\left(\mathbf{A}^{\star\prime}\right)$ $= \mathbf{A}'$, which is done as $\mathbf{A}' = D_e(TJ_e(\mathbf{A})T^{-1})$.

2.4.3. PGA 2-versor Rotation Operator. The 2-versor rotation operator, called a rotor, is defined as

$$R = \exp(\theta\hat{\mathbf{n}}^*/2), \tag{17}$$

for rotation centered on the origin, around axis $\hat{\mathbf{n}}$, by angle θ counterclockwise by right-hand rule, where $\hat{\mathbf{n}}^* = \hat{\mathbf{n}}/\mathbf{I}_3 = -\hat{\mathbf{n}}\mathbf{I}_3$. Any PGA entity $\mathbf{A}$ can be rotated using R as a versor as $\mathbf{A}' = R\mathbf{A}R^{-1}$, where $R^{-1} = R^{\sim}$ (the reverse).

Using the unit line $\hat{\boldsymbol{l}} = \hat{\boldsymbol{l}}_{\mathbf{p},\mathbf{d}} = \mathbf{L}^{\star}_{\mathbf{p},\hat{\mathbf{d}}}$, we form the rotor $R_{\boldsymbol{l}} = \exp(\theta\hat{\boldsymbol{l}}/2)$ for rotation of any *plane-based* PGA entity $\boldsymbol{a}$ around the line $\hat{\boldsymbol{l}} = \hat{\mathbf{d}}^* - (\mathbf{p} \cdot \hat{\mathbf{d}}^*)\mathbf{e}_0$ by angle θ as $\boldsymbol{a}' = R_{\boldsymbol{l}}\boldsymbol{a}R_{\boldsymbol{l}}^{-1}$, where $R_{\boldsymbol{l}}^{-1} = R_{\boldsymbol{l}}^{\sim}$ (the reverse). The rotation is around the direction $\hat{\mathbf{d}}$ of the line by right-hand rule. The rotor $R_{\boldsymbol{l}}$ cannot be used on any of the point-based PGA entities since the point-based entities cannot be translated using T. The rotor $R_{\boldsymbol{l}}$ is the same as the translated rotor $R_{\boldsymbol{l}} = T_{\mathbf{p}} \exp(\theta\hat{\mathbf{d}}^*/2)T_{\mathbf{p}}^{-1}$.

The rotor $R_{\boldsymbol{l}}$ can also be formed as successive reflections in two non-parallel CPNS PGA unit planes $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$ that meet in line $\boldsymbol{l} = \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1$ with angle $\theta/2$ between the planes from $\boldsymbol{\pi}_1$ toward $\boldsymbol{\pi}_2$ such that $R_{\boldsymbol{l}} = \boldsymbol{\pi}_2\boldsymbol{\pi}_1 = \boldsymbol{\pi}_2 \cdot \boldsymbol{\pi}_1 + \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1 = \cos(\theta/2) + \sin(\theta/2)\hat{\boldsymbol{l}}$.

2.4.4. Join Operation in the Point-based Algebra. In the point-based algebra, points can be spanned by wedge product, which is called join product. A line is $\mathbf{L} = \mathbf{P}_2 \wedge \mathbf{P}_1$, with orientation from $\mathbf{P}_1$ toward $\mathbf{P}_2$, or $\mathbf{L}_{\mathbf{p},\hat{\mathbf{d}}} = \hat{\mathbf{d}} \wedge \mathbf{P}$, where $\mathbf{d} = \mathbf{P}_2 - \mathbf{P}_1$. The dual $\hat{\boldsymbol{l}}_{\mathbf{p},\mathbf{d}} = J_e(\mathbf{L}_{\mathbf{p},\hat{\mathbf{d}}})$ is a unit line with axis direction $\hat{\mathbf{d}}$. A plane is $\boldsymbol{\Pi} = \mathbf{P}_3 \wedge \mathbf{P}_2 \wedge \mathbf{P}_1$, where $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3\}$ are arranged geometrically counterclockwise on the plane. Using $\{\mathbf{d}_2 = \mathbf{P}_3 - \mathbf{P}_1,\ \mathbf{d}_1 = \mathbf{P}_2 - \mathbf{P}_1,\ \mathbf{P} = \mathbf{P}_1\}$ then $\boldsymbol{\Pi} = \mathbf{d}_2 \wedge \mathbf{d}_1 \wedge \mathbf{P}$. Using $\mathbf{D} = \mathbf{d}_2 \wedge \mathbf{d}_1$ and $\hat{\mathbf{D}} = \mathbf{D}/\sqrt{|\mathbf{D}^2|} = \hat{\mathbf{n}}^*$, then $\boldsymbol{\Pi} = \hat{\mathbf{n}}^* \wedge \mathbf{P} = \mathbf{P} \wedge \hat{\mathbf{n}}^* = D_e(\hat{\boldsymbol{\pi}})$ is the dual of $\hat{\boldsymbol{\pi}}$.

In the plane-based algebra, we can form join products using dualization. A line is $\boldsymbol{l} = (\boldsymbol{p}_2^{-\star} \wedge \boldsymbol{p}_1^{-\star})^\star = \boldsymbol{p}_2 \vee \boldsymbol{p}_1$ using the join product notation $\vee$, where $\boldsymbol{p}^{-\star} = D_e(\boldsymbol{p}) = -J_e(\boldsymbol{p})$. A plane is $\boldsymbol{\pi} = (\boldsymbol{p}_3^{-\star} \wedge \boldsymbol{p}_2^{-\star} \wedge \boldsymbol{p}_1^{-\star})^\star = \boldsymbol{p}_3 \vee \boldsymbol{p}_2 \vee \boldsymbol{p}_1$.

2.4.5. Meet Operation in the Plane-based Algebra. In the plane-based algebra, planes can be intersected by wedge product, which is called meet product. A line is $\boldsymbol{l} = \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1$, and a point is $\boldsymbol{p} = \boldsymbol{\pi}_3 \wedge \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1$.

In the point-based algebra, we can form meet products using dualization. A line is $\mathbf{L} = (\boldsymbol{\Pi}_2^\star \wedge \boldsymbol{\Pi}_1^\star)^{-\star}$, and a point is $\mathbf{P} = (\boldsymbol{\Pi}_3^\star \wedge \boldsymbol{\Pi}_2^\star \wedge \boldsymbol{\Pi}_1^\star)^{-\star}$.

2.4.6. Reflections in Planes. The reflection of $\boldsymbol{\pi}_1$ in $\boldsymbol{\pi}_2$ is

$$\boldsymbol{\pi}_1' = -\boldsymbol{\pi}_2\boldsymbol{\pi}_1\boldsymbol{\pi}_2^{-1}. \tag{18}$$

The reflection of line $\boldsymbol{l} = \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1$ in plane $\boldsymbol{\pi}_3$ is

$$\boldsymbol{l}' = (-\boldsymbol{\pi}_3\boldsymbol{\pi}_2\boldsymbol{\pi}_3^{-1}) \wedge (-\boldsymbol{\pi}_3\boldsymbol{\pi}_1\boldsymbol{\pi}_3^{-1}) = (-1)^2\boldsymbol{\pi}_3\boldsymbol{l}\boldsymbol{\pi}_3^{-1} = \boldsymbol{\pi}_3\boldsymbol{l}\boldsymbol{\pi}_3^{-1}. \tag{19}$$

The reflection of an oriented point $\boldsymbol{p} = \boldsymbol{\pi}_3 \wedge \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_1$ in plane $\boldsymbol{\pi}_4$ is

$$\boldsymbol{p}' = (-1)^3\boldsymbol{\pi}_4\boldsymbol{p}\boldsymbol{\pi}_4^{-1} = -\boldsymbol{\pi}_4\boldsymbol{p}\boldsymbol{\pi}_4^{-1} \quad \text{(oriented point reflection)}. \tag{20}$$

To keep the reflected point $\boldsymbol{p}'$ in standard form and orientation as a non-oriented point, then reflect as

$$\boldsymbol{p}' = \boldsymbol{\pi}_4\boldsymbol{p}\boldsymbol{\pi}_4^{-1} \quad \text{(non-oriented point reflection)}. \tag{21}$$

We do not actually reflect in lines, but we can in general rotate any plane-based entity $\boldsymbol{a}$ around line $\boldsymbol{l}$ by $180°$ as $\boldsymbol{a}' = \boldsymbol{l}\boldsymbol{a}\boldsymbol{l}^{-1}$, since $\hat{\boldsymbol{l}} = \exp(\pi\hat{\boldsymbol{l}}/2)$. Orientation is preserved by the rotation.

2.4.7. Projections onto Planes and Lines. A point $\boldsymbol{p}$ is projected onto plane $\boldsymbol{\pi}$ as

$$\boldsymbol{p}' = (\boldsymbol{p} \cdot \boldsymbol{\pi})\boldsymbol{\pi}^{-1}. \tag{22}$$

A line $\boldsymbol{l}$ is projected onto plane $\boldsymbol{\pi}$ as

$$\boldsymbol{l}' = (\boldsymbol{l} \cdot \boldsymbol{\pi})\boldsymbol{\pi}^{-1}. \tag{23}$$

A point $\boldsymbol{p}$ is projected onto line $\boldsymbol{l}$ as

$$\boldsymbol{p}' = (\boldsymbol{p} \cdot \boldsymbol{l})\boldsymbol{l}^{-1}. \tag{24}$$

2.4.8. Rejections from Planes and Lines. The rejection of line $\boldsymbol{l}$ from plane $\boldsymbol{\pi}$ is

$$\boldsymbol{l}' = (\boldsymbol{l} \wedge \boldsymbol{\pi})\boldsymbol{\pi}^{-1}. \tag{25}$$

The rejection of plane $\boldsymbol{\pi}$ from line $\boldsymbol{l}$ is

$$\boldsymbol{\pi}' = (\boldsymbol{\pi} \wedge \boldsymbol{l})\boldsymbol{l}^{-1}. \tag{26}$$

2.5. Conclusion to 3D PGA

In the previous sections, we have introduced or reviewed enough about 3D PGA for our needs. In the next section, we begin to discuss the main contribution of this paper, about the details of the dual quaternion geometric algebra in PGA.

Using a complete set of identities that we newly derive that relate the plane-based PGA to the dual quaternions, we will be able to convert nearly all PGA plane-based entities and operations into dual quaternion forms. We can also take advantage of the PGA entity dualization operation J_e and other algebraic abilities of PGA to implement every operation needed to work effectively with dual quaternions as a complete geometric algebra for points, lines, and planes.

3. Dual Quaternion Geometric Algebra in PGA

This section is about the Dual Quaternion Geometric Algebra (DQGA), which is the even-grades subalgebra $\mathcal{G}^+_{3,0,1}$ in PGA $\mathcal{G}_{3,0,1}$. In the prior sections, we reviewed and introduced PGA $\mathcal{G}_{3,0,1}$ enough that we can now use it to further examine the even-grades subalgebra $\mathcal{G}^+_{3,0,1}$ that represents, or emulates faithfully, the Dual Quaternion Algebra (DQA).

As this paper's main contribution, expanding on [5], we introduce DQGA as an algebra for points, lines, and planes with many useful and new operations that we could not find in any prior literature[4] during the time in which we performed the research [5]. All of our results in dual quaternions can be implemented purely in DQA without PGA. Using the larger PGA algebra, we have the PGA entity dualization operation J_e for dualizing nilpotent elements, and we implement a complete set of special operations that

[4] The recent book [6], only noticed by the authors after completing the research in [5], does not have all of the same results as contributed in this paper.

any DQA implementation would also need, such as conjugates and operations for taking various parts from a dual quaternion. DQGA is itself a complete implementation of DQA that is also extended further by its seemless orientation-preserving integration with PGA and its entities, allowing all PGA and DQGA entities and operations to be used together through identities and dualizations between the entity forms in DQGA $\{p, l, \pi\}$, plane-based PGA $\{\boldsymbol{p}, \boldsymbol{l}, \boldsymbol{\pi}\}$, and point-based PGA $\{\mathbf{P}, \mathbf{L}, \mathbf{\Pi}\}$.

3.1. Introduction to Dual Quaternion Geometric Algebra

In DQA, the dual quaternion basis elements are $\{1, \varepsilon, \mathbf{i}, \mathbf{j}, \mathbf{k}\}$. The element ε is the nilpotent scalar, where $\varepsilon^2 = 0$. The elements $\{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$ are quaternion unit vectors with the quaternion product rule $\mathbf{ijk} = \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -1$ and the vector calculus dot (bold dot $\boldsymbol{\cdot}$) and cross (bold cross $\boldsymbol{\times}$) products.

In DQA, a homogeneous point is $p_{\mathbf{p}} = 1 + \varepsilon\mathbf{p}$, embedding vector point $\mathbf{p}$. There is a rotation operator (rotor) $R = \exp(\theta\hat{\mathbf{n}}/2)$ and a translation operator (translator) $T = \exp(\varepsilon\mathbf{d}/2)$, which act on a point as $p'_{\mathbf{p}} = Rp_{\mathbf{p}}R^{-1}$ and $p'_{\mathbf{p}} = Tp_{\mathbf{p}}T = p_{\mathbf{p}+\mathbf{d}}$, respectively. The rotor R acts as a versor on point p, just as it does on quaternions. The translator T is not acting as a versor, but it acts as just another homogeneous point $T = \exp(\varepsilon\mathbf{d}/2) = 1 + \varepsilon\mathbf{d}/2 = p_{\mathbf{d}/2}$. For points, we can compose translation and rotation to rotate relative to center $\mathbf{d}$ as $p'_{\mathbf{t}} = (TRT^{-1})p_{\mathbf{t}}(T^{-1}R^{-1}T) = R_{\bar{l}}p_{\mathbf{t}}\overline{R}^{\dagger}_{\bar{l}}$, which we discuss further in Section 3.8.2. For lines and planes, we will derive other formulas for them and their translations.

In the following sections, we review dual numbers, quaternions, and dual quaternions and then show how they are represented in DQGA, which is the even-grades subalgebra $\mathcal{G}^+_{3,0,1}$ of PGA with basis elements $\{1, \mathbf{I}_4, \mathbf{e}^*_1 = \mathbf{e}_3\mathbf{e}_2, \mathbf{e}^*_2 = \mathbf{e}_1\mathbf{e}_3, \mathbf{e}^*_3 = \mathbf{e}_2\mathbf{e}_1, \mathbf{e}^*_1\mathbf{I}_4 = \mathbf{e}_0\mathbf{e}_1, \mathbf{e}^*_2\mathbf{I}_4 = \mathbf{e}_0\mathbf{e}_2, \mathbf{e}^*_3\mathbf{I}_4 = \mathbf{e}_0\mathbf{e}_3\}$. Then, we derive the dual quaternion point p, line l, and plane π and their rotation, translation, and other operations.

3.2. Dual Numbers in PGA

In this section, we discuss Dual Number Algebra (DNA) and then its representation in geometric algebra $\mathcal{G}^+_{3,0,1}$, which we call Dual Number Geometric Algebra (DNGA).

3.2.1. Dual Number Algebra. In DNA, a dual number z is

$$z = x + y\varepsilon, \tag{27}$$

where x and y are real numbers and ε is the nilpotent scalar, $\varepsilon^2 = 0$. A dual number z has a complex conjugate

$$\overline{z} = x - y\varepsilon, \tag{28}$$

magnitude

$$|z| = \sqrt{z\overline{z}} = \sqrt{x^2} = |x|, \tag{29}$$

inverse

$$z^{-1} = \overline{z}/z\overline{z} = \overline{z}/|z|^2, \tag{30}$$

and exponential form

$$z = x\exp(y\varepsilon/x) = x(1 + y\varepsilon/x). \tag{31}$$

A unit dual number is $\hat{z} = z/|z|$, or $\exp(y\varepsilon) = 1 + y\varepsilon$. The product of two unit dual numbers is

$$(1+y_1\varepsilon)(1+y_2\varepsilon) = \exp(y_1\varepsilon)\exp(y_2\varepsilon) = 1+(y_1+y_2)\varepsilon = \exp((y_1+y_2)\varepsilon). \tag{32}$$

Addition is performed as multiplication, which is used for translation operations.

3.2.2. Dual Number Geometric Algebra. In $\mathcal{G}^+_{3,0,1}$, $\{1, \varepsilon\} \hat{=} \{1, \mathbf{I}_4\}$. We refer to dual numbers on the basis $\{1, \varepsilon\}$ as dual number algebra (DNA), and dual numbers on the basis $\{1, \mathbf{I}_4\}$ as dual number geometric algebra (DNGA). In DNGA, a dual number $z = x + y\varepsilon$ is represented as

$$z = x + y\mathbf{I}_4. \tag{33}$$

The complex conjugate is implemented as

$$\overline{z} = \mathbf{I}_3 z \mathbf{I}_3^{-1}. \tag{34}$$

The real part of z is

$$\Re(z) = \mathrm{X}(z) = x = (z + \overline{z})/2. \tag{35}$$

The imaginary part of z is

$$\Im(z) = y\mathbf{I}_4 = (z - \overline{z})/2. \tag{36}$$

The real number y is taken as

$$\mathrm{Y}(z) = y = -J_e(\Im(z)). \tag{37}$$

3.3. Quaternions in PGA

In this section, we review the Quaternion Algebra (QA) in its original form [11], and then we discuss its representation in the even-grades subalgebra $\mathcal{G}^+_3$, which we will call Quaternion Geometric Algebra (QGA).

3.3.1. Quaternion Algebra. In QA, the basis elements $\{1, \mathbf{i}, \mathbf{j}, \mathbf{k}\}$ have the defined product rule

$$\mathbf{ijk} = \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -1. \tag{38}$$

A quaternion q is a linear combination of the basis elements as

$$q = q_w + q_x\mathbf{i} + q_y\mathbf{j} + q_z\mathbf{k} = q_w + \mathbf{q}. \tag{39}$$

The units $\{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$ are a vector basis for $\mathbb{R}^3$ and are the same as the unit vectors used in Vector Calculus. Using the product rule, we can derive the ratios $\{\mathbf{i} = \mathbf{k}/\mathbf{j},\ \mathbf{j} = \mathbf{i}/\mathbf{k},\ \mathbf{k} = \mathbf{j}/\mathbf{i}\}$ which are the same as the cross products $\{\mathbf{i} = \mathbf{j}\times\mathbf{k},\ \mathbf{j} = \mathbf{k}\times\mathbf{i},\ \mathbf{k} = \mathbf{i}\times\mathbf{j}\}$ that define the *right-hand rule*. The product of two quaternions, p and q, is

$$pq = p_w q_w - \mathbf{p}\cdot\mathbf{q} + p_w\mathbf{q} + q_w\mathbf{p} + \mathbf{p}\times\mathbf{q}, \tag{40}$$

where the *dot product* is

$$\mathbf{p}\cdot\mathbf{q} = -(\mathbf{pq} + \mathbf{qp})/2 = p_x q_x + p_y q_y + p_z q_z \tag{41}$$

and the *cross product* is

$$\mathbf{p}\times\mathbf{q} = (\mathbf{pq}-\mathbf{qp})/2 = (p_y q_z - p_z q_y)\mathbf{i} + (p_z q_x - p_x q_z)\mathbf{j} + (p_x q_y - p_y q_x)\mathbf{k}. \quad (42)$$

The product of two vectors is

$$\mathbf{pq} = -\mathbf{p}\cdot\mathbf{q} + \mathbf{p}\times\mathbf{q}. \quad (43)$$

The *conjugate* of q is

$$\mathrm{K}(q) = q^{\dagger} = q_w - \mathbf{q}, \quad (44)$$

where $\mathrm{K}(q_1 q_2 \ldots) = \ldots q_2^{\dagger} q_1^{\dagger}$ in the reverse order. The scalar part of q is

$$\mathrm{S}(q) = q_w = (q + q^{\dagger})/2. \quad (45)$$

The vector part is

$$\mathrm{V}(q) = \mathbf{q} = (q - q^{\dagger})/2. \quad (46)$$

The *tensor* (or magnitude) of q is

$$\mathrm{T}(q) = |q| = \sqrt{qq^{\dagger}} = \sqrt{q_w^2 + q_x^2 + q_y^2 + q_z^2}. \quad (47)$$

A unit quaterion $\hat{q}$, also called a versor, is

$$\hat{q} = \mathrm{U}(q) = q/\mathrm{T}(q) = \cos(\theta/2) + \sin(\theta/2)\hat{\mathbf{q}} = \exp(\theta\hat{\mathbf{q}}/2) = R \quad (48)$$

and represents a rotation operator R for rotation by angle θ around unit vector axis $\hat{\mathbf{q}}$, where

$$\hat{\mathbf{q}} = \mathbf{q}/\mathrm{T}(\mathbf{q}) = \mathbf{q}/\sqrt{\mathbf{q}\cdot\mathbf{q}} = \mathbf{q}/\|\mathbf{q}\|. \quad (49)$$

The square of any unit vector is $\hat{\mathbf{q}}^2 = -1$. Any quaternion q can be written as the product of its unit $\mathrm{U}(q)$ and tensor $\mathrm{T}(q)$ as

$$q = \mathrm{T}(q)\mathrm{U}(q). \quad (50)$$

The inverse of q is

$$q^{-1} = q^{\dagger}/qq^{\dagger} = q^{\dagger}/\mathrm{T}(q)^2 = \mathrm{U}(q)^{\dagger}\mathrm{T}(q)^{-1}, \quad (51)$$

where $\mathrm{U}(q)^{\dagger} = \mathrm{U}(q)^{-1}$. Quaternion rotation of a vector $\mathbf{p}$ is

$$\mathbf{p}' = R\mathbf{p}R^{\dagger}, \quad (52)$$

rotating $\mathbf{p}$ around axis $\hat{\mathbf{q}}$ by angle θ centered on the origin.

3.3.2. Quaternion Geometric Algebra. In QGA, $\{1, \mathbf{i}, \mathbf{j}, \mathbf{k}\} \hat{=} \{1, \mathbf{e}_1^*, \mathbf{e}_2^*, \mathbf{e}_3^*\}$. QGA is the subalgebra $\mathcal{G}_3^+$. Any vector $\mathbf{v} \in \mathcal{G}_3^1$ is transformed into its quaternion vector form $\mathbf{v}^*$ in QGA by the dualization

$$\mathbf{v}^* = \mathbf{v}/\mathbf{I}_3 \in \mathcal{G}_3^2. \quad (53)$$

For example, in QA we have $\mathbf{k} = \mathbf{j}/\mathbf{i}$, and in QGA we have the corresponding ratio $\mathbf{k}\hat{=}\mathbf{e}_2^*/\mathbf{e}_1^* = \mathbf{e}_2/\mathbf{e}_1 = \mathbf{e}_2\mathbf{e}_1$. This is simpler to express as the dual $\mathbf{k}\hat{=}\mathbf{e}_3^* = \mathbf{e}_3/\mathbf{I}_3 = \mathbf{e}_3\mathbf{e}_3\mathbf{e}_2\mathbf{e}_1 = \mathbf{e}_2\mathbf{e}_1$, and similarly for $\mathbf{i}$ and $\mathbf{j}$. In QGA, a quaternion q is

$$q = q_w + q_x\mathbf{e}_1^* + q_y\mathbf{e}_2^* + q_z\mathbf{e}_3^* = q_w + \mathbf{q}^*. \quad (54)$$

The quaternion conjugate is implemented as

$$\mathrm{K}(q) = q^{\dagger} = q^{\sim} = q_w - \mathbf{q}^*, \quad (55)$$

where $q^{\sim}$ is the geometric algebra *reverse* operation on q. The notation $q^{\dagger}$ is also a geometric algebra notation for reverse [12], so there is no conflict with notation.

The dot product $\boldsymbol{\cdot}$ (bold dot) is implemented as

$$\mathbf{p}^* \boldsymbol{\cdot} \mathbf{q}^* = -\mathbf{p}^* \cdot \mathbf{q}^* = -(\mathbf{p}^*\mathbf{q}^* + \mathbf{q}^*\mathbf{p}^*)/2. \tag{56}$$

The cross product $\boldsymbol{\times}$ (bold cross) is implemented as

$$\mathbf{p}^* \boldsymbol{\times} \mathbf{q}^* = \mathbf{p}^* \times \mathbf{q}^* = (\mathbf{p}^*\mathbf{q}^* - \mathbf{q}^*\mathbf{p}^*)/2. \tag{57}$$

The scalar part $\mathrm{S}(q)$, vector part $\mathrm{V}(q)$, tensor $\mathrm{T}(q)$, unit $\mathrm{U}(q)$, and inverse q^{-1} are all implemented the same as in QA by using the conjugate $q^{\dagger}$. The QGA quaternion vector units still obey the quaternion product rule:

$$\mathbf{e}_1^*\mathbf{e}_2^*\mathbf{e}_3^* = \mathbf{e}_1^{*2} = \mathbf{e}_2^{*2} = \mathbf{e}_3^{*2} = -1. \tag{58}$$

It should be understood that, in QA we have vector $\mathbf{v}$ on the basis $\{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$, in PGA we have $\mathbf{v}$ on the basis $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$, and in QGA we have $\mathbf{v}$ on the basis $\{\mathbf{e}_1^*, \mathbf{e}_2^*, \mathbf{e}_3^*\}$ as $\mathbf{v}^*$. Each represents the same vector $\mathbf{v}$ on a different basis in a different algebra. We will refer to a dualized vector $\mathbf{v}^* = \mathbf{v}/\mathbf{I}_3$ as just being the quaternion vector $\mathbf{v}^*$.

3.4. Dual Quaternions in PGA

In the prior sections, we reviewed dual number algebra (DNA) and quaternion algebra (QA) in their original forms, and then how they are represented or emulated in the geometric algebra $\mathcal{G}^+_{3,0,1}$ as subalgebras that we have called dual number geometric algebra (DNGA) and quaternion geometric algebra (QGA). In this section, we discuss dual quaternion algebra (DQA) and its representation in $\mathcal{G}^+_{3,0,1}$ as dual quaternion geometric algebra (DQGA).

3.4.1. Dual Quaternion Algebra. In DQA, dual quaternions are very similar to quaternions, but instead of using only the real numbers, we extend the real numbers to dual numbers. The DQA basis elements are $\{1, \varepsilon, \mathbf{i}, \mathbf{j}, \mathbf{k}\}$. A dual quaternion d has the general form $d = q_1 + q_2\varepsilon$, where $q_1 = q_{1w} + \mathbf{q}_1 = q_{1w} + q_{1x}\mathbf{i} + q_{1y}\mathbf{j} + q_{1z}\mathbf{k}$ and $q_2 = q_{2w} + \mathbf{q}_2$ are quaternions and ε is the nilpotent scalar, where $\varepsilon^2 = 0$.

3.4.2. Dual Quaternion Geometric Algebra. In DQGA, $\{1,\ \varepsilon,\ \mathbf{i},\ \mathbf{j},\ \mathbf{k}\} \hat{=} \{1,\ \mathbf{I}_4,\ \mathbf{e}_1^*,\ \mathbf{e}_2^*,\ \mathbf{e}_3^*\}$, and a dual quaternion is

$$d = q_1 + q_2\mathbf{I}_4 = q_{1w} + \mathbf{q}_1^* + q_{2w}\mathbf{I}_4 + \mathbf{q}_2^*\mathbf{I}_4. \tag{59}$$

The *complex conjugate* $\overline{d} = q_1 - q_2\mathbf{I}_4$ is implemented as

$$\overline{d} = \mathbf{I}_3 d \mathbf{I}_3^{-1}, \tag{60}$$

which is the same as in DNGA, where $(d_1d_2\ldots)^- = \overline{d}_1\overline{d}_2\ldots$ in the same order.

The *quaternion conjugate* $\mathrm{K}(d) = q_{1w} - \mathbf{q}_1^* + q_{2w}\mathbf{I}_4 - \mathbf{q}_2^*\mathbf{I}_4$ is implemented as

$$\mathrm{K}(d) = d^{\dagger} = d^{\sim}, \tag{61}$$

which is the same as in QGA, where $(d_1 d_2 \ldots)^\dagger = \cdots d_2^\dagger d_1^\dagger$ in the reverse order.

We can compose $\overline{d}$ and $d^\dagger$ as the "*dual conjugate*"

$$\overline{d}^\dagger = q_1^\dagger - q_2^\dagger \mathbf{I}_4 = q_{1w} - \mathbf{q}_1^* - q_{2w}\mathbf{I}_4 + \mathbf{q}_2^*\mathbf{I}_4. \tag{62}$$

The *real part* is

$$\Re(d) = \mathrm{X}(d) = (d + \overline{d})/2 = q_1. \tag{63}$$

The *imaginary part* is

$$\Im(d) = (d - \overline{d})/2 = q_2 \mathbf{I}_4. \tag{64}$$

The real component of the imaginary part is

$$\mathrm{Y}(d) = -J_e(\Im(d)) = q_2. \tag{65}$$

The dual number-valued *scalar part* is

$$\mathrm{S}(d) = (d + d^\dagger)/2 = q_{1w} + q_{2w}\mathbf{I}_4. \tag{66}$$

The *vector part* is

$$\mathrm{V}(d) = (d - d^\dagger)/2 = \mathbf{q}_1^* + \mathbf{q}_2^*\mathbf{I}_4. \tag{67}$$

The dual number-valued *tensor* (or magnitude) is

$$\mathrm{T}(d) = \sqrt{dd^\dagger} = \sqrt{|q_1|^2 \exp(2(q_{1w}q_{2w} + \mathbf{q}_1^* \cdot \mathbf{q}_2^*)\mathbf{I}_4/|q_1|^2)} \tag{68}$$

$$= |d|_{\mathbb{D}} = |q_1|(1 + ((q_{1w}q_{2w} + \mathbf{q}_1^* \cdot \mathbf{q}_2^*)/|q_1|^2)\mathbf{I}_4), \tag{69}$$

for $|q_1| \neq 0$. For $q_1 = 0$, $dd^\dagger = 0$ and $\mathrm{T}(d) = 0$. The notation $\mathrm{T}(d) = |d|_{\mathbb{D}}$ indicates that the tensor is dual number-valued and $\mathrm{T}(z) = |x| \exp(y\mathbf{I}_4/x) = \pm z$. The inverse tensor is

$$\mathrm{T}(d)^{-1} = |q_1|^{-1}(1 - ((q_{1w}q_{2w} + \mathbf{q}_1^* \cdot \mathbf{q}_2^*)/|q_1|^2)\mathbf{I}_4) = \overline{\mathrm{T}}(d)/|\mathrm{T}(d)|^2. \tag{70}$$

Using $\mathrm{T}(d)^{-1}$, a unit dual quaternion is

$$\hat{d} = \mathrm{U}(d) = d\mathrm{T}(d)^{-1}, \tag{71}$$

which is also called *normalizing* the dual quaternion d. We define the following three new part operators. The *point part* is [for $p_{\mathbf{p}}$ (75) with $\mathbf{p}^* = \mathbf{q}_2^*/q_{1w}$]

$$\mathrm{P}(d) = (d + \overline{d}^\dagger)/2 = q_{1w} + \mathbf{q}_2^*\mathbf{I}_4. \tag{72}$$

The *plane part* is [for $\hat{\pi}_{d,\mathbf{n}}$ (81) with $-\hat{\mathbf{n}}^* = \mathbf{q}_1^*/\|\mathbf{q}_1^*\|$ and $-d = q_{2w}/\|\mathbf{q}_1^*\|$]

$$\Pi(d) = (d - \overline{d}^\dagger)/2 = \mathbf{q}_1^* + q_{2w}\mathbf{I}_4. \tag{73}$$

The *line part* is [for $l_{\mathbf{p},\mathbf{d}}$ (78) when $\mathbf{q}_1^* \cdot \mathbf{q}_2^* = 0$ with $\mathbf{d}^* = \mathbf{q}_1^*$ and $\mathbf{m}^* = \mathbf{p}^* \times \mathbf{d}^* = -\mathbf{q}_2^*$]

$$\mathrm{L}(d) = \mathrm{V}(q) = \mathbf{q}_1^* + \mathbf{q}_2^*\mathbf{I}_4. \tag{74}$$

3.5. DQGA Geometric Entities

In this section, we derive the DQGA point entity $p = p_{\mathbf{t}}$ (embedding $\mathbf{t}^*$), DQGA plane entity $\pi = \pi_{\mathbf{p},\mathbf{n}} = \pi_{d,\mathbf{n}}$ (through $\mathbf{p}^*$ with normal $\mathbf{n}^*$), and the DQGA line entity $l = l_{\mathbf{p},\mathbf{d}}$ (through $\mathbf{p}^*$ in direction $\mathbf{d}^*$).

Each entity $A \in \{p, l, \pi\}$ represents a *null space set* of points, $\mathcal{N}_A = \{p_{\mathbf{t}} : \mathrm{N}_A(p_{\mathbf{t}}A) = 0\}$. The product $p_{\mathbf{t}}A$ is the test of point $p_{\mathbf{t}}$ for coincidence with the surface represented by A. Points in the null space set $\mathcal{N}_A$ are on the surface represented by A. The *null space entity part* $\mathrm{N}_A(p_{\mathbf{t}}A)$ for the product $p_{\mathbf{t}}A$ is the part that represents the null space of entity A. For a point, $A = p$ and $\mathrm{N}_p(p_{\mathbf{t}}p_{\mathbf{a}}) = \mathrm{V}(\Im(p_{\mathbf{t}}p_{\mathbf{a}}))$ with set $\mathcal{N}_p = \{p_{\mathbf{t}} : \mathrm{Y}(\mathrm{N}_p(p_{\mathbf{t}}p_{\mathbf{a}})) = \mathbf{t} + \mathbf{a} = 0\} = \{p_{-\mathbf{a}}\}$. For a line, $A = l$ and $\mathrm{N}_l(p_{\mathbf{t}}l) = \mathrm{V}(\Im(p_{\mathbf{t}}l))$ with set $\mathcal{N}_l = \{p_{\mathbf{t}} : \mathrm{Y}(\mathrm{N}_l(p_{\mathbf{t}}l)) = \mathbf{t}^* \times \mathbf{d}^* - \mathbf{p}^* \times \mathbf{d}^* = 0\}$. For a plane, $A = \pi$ and $\mathrm{N}_\pi(p_{\mathbf{t}}\pi) = \mathrm{S}(\Im(p_{\mathbf{t}}\pi))$ with set $\mathcal{N}_\pi = \{p_{\mathbf{t}} : \mathrm{Y}(\mathrm{N}_\pi(p_{\mathbf{t}}\pi)) = \mathbf{t}^* \cdot \mathbf{n}^* - \mathbf{p}^* \cdot \mathbf{n}^* = 0\}$.

In CPNS PGA, the commutator product $\times$ always gives the null space entity part. In DQGA, there is not a single product that always gives the null space entity part. In DQGA, we have to compute the dual quaternion product $p_{\mathbf{t}}A$ and then take part $\mathrm{N}_A(p_{\mathbf{t}}A)$. Point $p_{\mathbf{t}}$ is a point of A if and only if $\mathrm{N}_A(p_{\mathbf{t}}A) = 0$.

For each kind of entity A, there are also entity-specific formulas for their reflections in planes, translations, and other operations. Rotation is performed the same on all entities using the rotor R in a versor sandwich product of the entity.

3.5.1. DQGA Point Entity. In DQA, a quaternion vector $\mathbf{t}$ is embedded as a homogeneous point $p_{\mathbf{t}} = 1 + \mathbf{t}\varepsilon$. In DQGA, a quaternion vector $\mathbf{t}^* = (x\mathbf{e}_1 + y\mathbf{e}_2 + z\mathbf{e}_3)/\mathbf{I}_3$ is embedded as the homogeneous DQGA point

$$p_{\mathbf{t}} = 1 + \mathbf{t}^*\mathbf{I}_4 = \exp(\mathbf{t}^*\mathbf{I}_4) = 1 + \mathbf{e}_0\mathbf{t} = 1 + x\mathbf{e}_0\mathbf{e}_1 + y\mathbf{e}_0\mathbf{e}_2 + z\mathbf{e}_0\mathbf{e}_3. \quad (75)$$

Two points $p_{\mathbf{t}}$ and $p_{\mathbf{p}}$ represent the same point if and only if $\mathrm{N}_p(p_{\mathbf{t}}\overline{p}_{\mathbf{b}}) = \mathrm{V}(\Im(p_{\mathbf{t}}\overline{p}_{\mathbf{b}})) = 0$ or $\mathrm{Y}(\mathrm{V}(\Im(p_{\mathbf{t}}\overline{p}_{\mathbf{b}}))) = 0$. The identity between $p_{\mathbf{t}}$ and $\boldsymbol{p}_{\mathbf{t}}$ (12) is

$$p_{\mathbf{t}} = \boldsymbol{p}_{\mathbf{t}}\mathbf{I}_3^{-1} \text{ or } \boldsymbol{p}_{\mathbf{t}} = p_{\mathbf{t}}\mathbf{I}_3. \quad (76)$$

Using (76) and $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$ (60), two points $p_{\mathbf{t}}$ and $p_{\mathbf{p}}$ represent the same point if and only if $\boldsymbol{p}_{\mathbf{t}} \times \boldsymbol{p}_{\mathbf{p}} = (\boldsymbol{p}_{\mathbf{t}}\boldsymbol{p}_{\mathbf{p}} - \boldsymbol{p}_{\mathbf{p}}\boldsymbol{p}_{\mathbf{t}})/2 = 0$, or

$$(-p_{\mathbf{t}}\overline{p}_{\mathbf{p}} + p_{\mathbf{p}}\overline{p}_{\mathbf{t}})/2 = 0. \quad (77)$$

We can save computation by using $\mathrm{N}_p(p_{\mathbf{t}}\overline{p}_{\mathbf{p}}) = 0$.

3.5.2. DQGA Line Entity. In CPNS PGA, $\boldsymbol{p}_{\mathbf{t}} \times \boldsymbol{l} = 0$ is the test of $\boldsymbol{p}_{\mathbf{t}}$ (12) with $\boldsymbol{l}$ (11), which is a grade 3 test. We switch to geometric product $\boldsymbol{p}_{\mathbf{t}}\boldsymbol{l}$ and use (76) to obtain $p_{\mathbf{t}}\mathbf{I}_3\boldsymbol{l}$, which we rewrite as $p_{\mathbf{t}}\mathbf{I}_3\boldsymbol{l}\mathbf{I}_3^{-1}\mathbf{I}_3$ and abridge the RHS $\mathbf{I}_3$ to take the DQGA line entity as

$$l = \mathbf{I}_3\boldsymbol{l}\mathbf{I}_3^{-1} = \overline{\boldsymbol{l}} = \mathbf{d}^* + (\mathbf{p} \cdot \mathbf{d}^*)\mathbf{e}_0 = \mathbf{d}^* - (\mathbf{p}^* \times \mathbf{d}^*)\mathbf{I}_4, \quad (78)$$

where we have used the identity $\mathbf{p}\cdot\mathbf{d}^* = (\mathbf{p}^*\times\mathbf{d}^*)\mathbf{I}_3$. For $\mathbf{d}^* = \hat{\mathbf{d}}^*$, then $l = \hat{l}$ is a unit line. The identity between l and $\boldsymbol{l}$ (11) using $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$ (60) is

$$l = \bar{\boldsymbol{l}} \text{ or } \boldsymbol{l} = \bar{l}. \tag{79}$$

Point $p_\mathbf{t}$ is on l if and only if $\mathrm{N}_l(p_\mathbf{t}l) = \mathrm{Y}(\mathrm{V}(\Im(p_\mathbf{t}l))) = \mathbf{t}^*\times\mathbf{d}^* - \mathbf{p}^*\times\mathbf{d}^* = 0$, which is the Plücker coordinates $(\mathbf{d}^* : \mathbf{m}^*)$ condition for the line with $\mathbf{m}^* = \mathbf{p}^*\times\mathbf{d}^*$. Using (79), (76), and (60), $p_\mathbf{t}$ is on l if and only if $\boldsymbol{p}_\mathbf{t}\times\boldsymbol{l} = (\boldsymbol{p}_\mathbf{t}\boldsymbol{l} - \boldsymbol{l}\boldsymbol{p}_\mathbf{t})/2 = 0$, $(p_\mathbf{t}l - \bar{l}p_\mathbf{t})\mathbf{I}_3/2 = 0$, or

$$(p_\mathbf{t}l - \bar{l}p_\mathbf{t})/2 = 0. \tag{80}$$

We may save computation by using $\mathrm{N}_l(p_\mathbf{t}l) = 0$.

3.5.3. DQGA Plane Entity. In CPNS PGA, $\boldsymbol{p}_\mathbf{t}\times\boldsymbol{\pi} = \boldsymbol{p}_\mathbf{t}\wedge\boldsymbol{\pi} = 0$ is the test of $\boldsymbol{p}_\mathbf{t}$ (12) with $\boldsymbol{\pi}$ (9). We switch to geometric product $\boldsymbol{p}_\mathbf{t}\boldsymbol{\pi}$ and use (76) to obtain $\boldsymbol{p}_\mathbf{t}\boldsymbol{\pi} = p_\mathbf{t}\mathbf{I}_3\boldsymbol{\pi}$. We take the DQGA plane entity as

$$\pi = \mathbf{I}_3\boldsymbol{\pi} = \pi_{\mathbf{p},\mathbf{n}} = \mathbf{I}_3(\mathbf{n} + (\mathbf{p}\cdot\mathbf{n})\mathbf{e}_0) = -(\mathbf{n}^* + (\mathbf{p}^*\cdot\mathbf{n}^*)\mathbf{I}_4). \tag{81}$$

The identity between π and $\boldsymbol{\pi}$ is

$$\pi = \mathbf{I}_3\boldsymbol{\pi} \text{ or } \boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi. \tag{82}$$

For $\mathbf{n}^* = \hat{\mathbf{n}}^*$, then $d = \mathbf{p}^*\cdot\mathbf{n}^*$ is the distance from origin and $\pi = \hat{\pi}_{d,\mathbf{n}}$ is a unit plane. Point $p_\mathbf{t}$ is on π if and only if $\mathrm{N}_\pi(p_\mathbf{t}\pi) = \mathrm{S}(\Im(p_\mathbf{t}\pi)) = 0$ or $\mathrm{Y}(\mathrm{N}_\pi(p_\mathbf{t}\hat{\pi})) = x\hat{n}_x + y\hat{n}_y + z\hat{n}_z - d = 0$. Using (82), (76), and $\overline{d}$ (60), $p_\mathbf{t}$ is on π if and only if $\boldsymbol{p}_\mathbf{t}\times\boldsymbol{\pi} = (\boldsymbol{p}_\mathbf{t}\boldsymbol{\pi} - \boldsymbol{\pi}\boldsymbol{p}_\mathbf{t})/2 = 0$, or

$$(p_\mathbf{t}\pi - \overline{\pi p}_\mathbf{t})/2 = 0. \tag{83}$$

We may save computation by using $\mathrm{N}_\pi(p_\mathbf{t}\pi) = 0$.

3.6. DQGA Reflection Operations

3.6.1. Reflection of a plane in another plane. DQGA plane π_1 (81) reflected in unit plane $\pi_2 = \hat{\pi}_2$ is

$$\pi_1' = \pi_2\overline{\pi}_1\pi_2. \tag{84}$$

Proof. In CPNS PGA, the reflection is $\boldsymbol{\pi}_1' = -\boldsymbol{\pi}_2\boldsymbol{\pi}_1\boldsymbol{\pi}_2$ (18). Using identities $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$ (82) and $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$ (60), then $\boldsymbol{\pi}_1' = \mathbf{I}_3^{-1}\pi_1' = -\mathbf{I}_3^{-1}\pi_2\mathbf{I}_3^{-1}\pi_1\mathbf{I}_3^{-1}\pi_2 = \mathbf{I}_3^{-1}\pi_2\overline{\pi}_1\pi_2$. □

3.6.2. Reflection of a line in a plane. DQGA line l (78) is reflected in unit plane $\pi = \hat{\pi}$ as

$$l' = -(\overline{\pi}l\pi)^- = -\pi\bar{l}\overline{\pi}. \tag{85}$$

Proof. In CPNS PGA, the reflection is $\boldsymbol{l}' = \boldsymbol{\pi}\boldsymbol{l}\boldsymbol{\pi}$ (19). Using identities $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$ (82), $\boldsymbol{l} = \bar{l}$ (79), and $\overline{d}$ (60), then $\boldsymbol{l}' = \bar{l}' = \mathbf{I}_3^{-1}\pi\bar{l}\mathbf{I}_3^{-1}\pi = -\overline{\pi}l\pi$ and $l' = -(\overline{\pi}l\pi)^- = -\pi\bar{l}\overline{\pi}$. □

3.6.3. Reflection of a point in a plane. Using $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$ (82), $\boldsymbol{p} = p\mathbf{I}_3$ (76), $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$ (60), and $\boldsymbol{p}' = -\boldsymbol{\pi}\boldsymbol{p}\boldsymbol{\pi}$ (20) for oriented point reflection, p reflected in unit plane $\pi = \hat{\pi}$ is

$$p' = \overline{\pi p \pi} = (\pi p \pi)^{-}. \quad (86)$$

This reflection changes the orientation of p'. By this method, reflecting three points and then joining them in the *same order* will produce the correctly reflected plane.

Using $\boldsymbol{p}' = \boldsymbol{\pi}\boldsymbol{p}\boldsymbol{\pi}$ (21) for non-oriented point reflection, p reflected in $\pi = \hat{\pi}$ is

$$p' = -\overline{\pi p \pi} = -(\pi p \pi)^{-}. \quad (87)$$

This reflection maintains point p' in standard form orientation. By this method, reflecting three points and then joining them in the *reverse order* will produce the correctly reflected plane. We can compose two reflections to generate rotation or translation.

3.7. DQGA Translation Operations

3.7.1. Translation of a point. DQGA point $p_{\mathbf{p}}$ is translated by $\mathbf{d}^*$ using $T = p_{\mathbf{d}/2}$ (15) as

$$p_{\mathbf{p}'} = p_{\mathbf{p}+\mathbf{d}} = T p_{\mathbf{p}} T. \quad (88)$$

Proof. In CPNS PGA, point $\boldsymbol{p}_{\mathbf{p}}$ is translated as $\boldsymbol{p}_{\mathbf{p}'} = \boldsymbol{p}_{\mathbf{p}+\mathbf{d}} = T\boldsymbol{p}_{\mathbf{p}}T^{-1}$ (15). Using identities $\boldsymbol{p}_{\mathbf{p}} = p_{\mathbf{p}}\mathbf{I}_3$ (76), $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$ (60), and $T^{-1} = \overline{T}$, then $\boldsymbol{p}_{\mathbf{p}'} = p_{\mathbf{p}'}\mathbf{I}_3 = T p_{\mathbf{p}} T\mathbf{I}_3$. Since points have commutative multiplication, $p_{\mathbf{p}'} = T^2 p_{\mathbf{p}} = p_{\mathbf{d}} p_{\mathbf{p}} = p_{\mathbf{p}+\mathbf{d}}$. □

3.7.2. Translation of a plane. DQGA plane π is translated by $\mathbf{d}^*$ using $T = p_{\mathbf{d}/2}$ (15) as

$$\pi' = \overline{T}\pi\overline{T}. \quad (89)$$

Proof. In CPNS PGA, plane $\boldsymbol{\pi}$ is translated as $\boldsymbol{\pi}' = T\boldsymbol{\pi}T^{-1}$. Using identities $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$ (82), $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$ (60), $T^{-1} = \overline{T}$, and $T\mathbf{I}_3^{-1} = \mathbf{I}_3^{-1}\overline{T}$, then $\boldsymbol{\pi}' = \mathbf{I}_3^{-1}\pi' = T\mathbf{I}_3^{-1}\pi T^{-1} = \mathbf{I}_3^{-1}\overline{T}\pi\overline{T}$. □

3.7.3. Translation of a line. DQGA line l is translated by $\mathbf{d}^*$ using $T = p_{\mathbf{d}/2}$ (15) as

$$l' = \overline{T} l T. \quad (90)$$

Proof. In CPNS PGA, line $\boldsymbol{l}$ is translated as $\boldsymbol{l}' = T\boldsymbol{l}T^{-1}$. Using identities $\boldsymbol{l} = \overline{l}$ (79), $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$ (60), and $T^{-1} = \overline{T}$, then $\boldsymbol{l}' = \overline{l}' = T\overline{l}T^{-1}$ and $l' = (T\overline{l}\overline{T})^{-} = \overline{T} l T$. □

3.8. DQGA Rotation Operations

3.8.1. Rotation centered around the origin. In DQA, we can rotate all dual quaternions and dual quaternion geometric entities using the quaternion algebra rotor $R = \exp(\theta\hat{\mathbf{n}}/2)$. In DQGA, the DQA rotor $R = \exp(\theta\hat{\mathbf{n}}/2)$ corresponds to the DQGA rotor

$$R = \exp(\theta\hat{\mathbf{n}}^*/2), \tag{91}$$

where $\hat{\mathbf{n}}^* = \hat{\mathbf{n}}/\mathbf{I}_3$. By outermorphism, R rotates any quaternion vector $\mathbf{v}^*$ within DQGA expressions, thereby rotating them as a whole rigid body. The rotation is centered on the origin, around the axis $\hat{\mathbf{n}}^*$, by angle θ using R as a versor sandwich product on any DQGA element A as $A' = RAR^{-1}$.

This versor operation is valid on all DQGA entities for rotation centered on the origin, but most of the other DQGA operations, for translation and rotation around lines, are not versor sandwich products, but are instead entity-specific special sandwich products that are derived for each entity.

We can take the plane-based PGA rotor $R_{\boldsymbol{l}} = \exp(\theta\hat{\boldsymbol{l}}/2)$ and convert it to the DQGA rotor $R_l = \overline{R_{\boldsymbol{l}}} = R_{\bar{\boldsymbol{l}}} = \exp(\theta\hat{l}/2)$ for rotation around the line l by angle θ, but we cannot apply R_l to all DQGA entities using a single form of versor sandwich product. However, the rotation of a point p, line l, or plane π can each use R_l in a specific formula for each entity type as we derive in the following sections.

3.8.2. Rotation of a point around a line. In CPNS PGA, a point $\boldsymbol{p}$ is rotated around the unit line $\boldsymbol{l} = \hat{\mathbf{d}}^* - (\mathbf{p}\cdot\hat{\mathbf{d}}^*)\mathbf{e}_0$ using the rotor $R_{\boldsymbol{l}} = \exp(\theta\boldsymbol{l}/2) = \cos(\theta/2) + \sin(\theta/2)\boldsymbol{l}$ as $\boldsymbol{p}' = R_{\boldsymbol{l}}\boldsymbol{p}R_{\boldsymbol{l}}^{-1}$. This rotor can also be formed as a composition of rotation and translation as a translated rotor $R_{\boldsymbol{l}} = TRT^{-1}$ or as reflection in two non-parallel planes $R_{\boldsymbol{l}} = \boldsymbol{\pi}_2\boldsymbol{\pi}_1$. We will just use $\exp(\theta\boldsymbol{l}/2)$, which is the easier and more intuitive form.

We use identities $\boldsymbol{p} = p\mathbf{I}_3$, $\boldsymbol{l} = \bar{l}$, $\mathbf{I}_3\bar{l} = l\mathbf{I}_3$. Then, $\boldsymbol{p}' = p'\mathbf{I}_3 = \exp(\theta\bar{l}/2)p\mathbf{I}_3\exp(-\theta\bar{l}/2)$. Therefore, the rotation of point p around line l by angle θ is

$$p' = \exp(\theta\bar{l}/2)p\exp(-\theta l/2) = \overline{R}_l p R_l^{\dagger}. \tag{92}$$

It is important that $l = \hat{l} = \mathrm{U}(l)$ be a unit line, or else the angle θ will be scaled incorrectly by any magnitude $\mathrm{T}(l)$ on l. The sense of rotation is by right-hand rule around the line through axis of rotation direction $\hat{\mathbf{d}}^*$. This is not a versor operation; it is a special dual quaternion sandwich product that is entity-specific, for rotating a DQGA point around a DQGA line. Each DQGA entity has a different formula for this operation in DQGA.

3.8.3. Rotation of a plane around a line. In CPNS PGA, the plane $\boldsymbol{\pi}$ is rotated around the unit line $\boldsymbol{l}$ using the rotor $R_{\boldsymbol{l}} = \exp(\theta\boldsymbol{l}/2)$ as $\boldsymbol{\pi}' = R_{\boldsymbol{l}}\boldsymbol{\pi}R_{\boldsymbol{l}}^{-1}$. We use identities $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$, $\boldsymbol{l} = \bar{l}$, $\bar{l}\mathbf{I}_3^{-1} = \mathbf{I}_3^{-1}l$. Then, $\boldsymbol{\pi}' = \mathbf{I}_3^{-1}\pi' = \exp(\theta\bar{l}/2)\mathbf{I}_3^{-1}\pi\exp(-\theta\bar{l}/2)$. Therefore, the rotation of plane π around line l by angle θ is

$$\pi' = \exp(\theta l/2)\pi\exp(-\theta\bar{l}/2) = R_l\pi\overline{R}_l^{\dagger}. \tag{93}$$

3.8.4. Rotation of a line around another line. In CPNS PGA, the line $\boldsymbol{l}_1$ is rotated around the unit line $\boldsymbol{l}_2$ using the rotor $R_{\boldsymbol{l}_2} = \exp(\theta \boldsymbol{l}_2/2)$ as $\boldsymbol{l}'_1 = R_{\boldsymbol{l}_2}\boldsymbol{l}_1 R_{\boldsymbol{l}_2}^{-1}$. We use identities $\boldsymbol{l} = \bar{l}$, $\bar{l}\mathbf{I}_3^{-1} = \mathbf{I}_3^{-1}l$. Then, $\boldsymbol{l}'_1 = \bar{l}'_1 = \exp(\theta\bar{l}_2/2)$ $\bar{l}_1 \exp(-\theta\bar{l}_2/2)$. Therefore, the rotation of line l_1 around line l_2 by angle θ is

$$l'_1 = \exp(\theta l_2/2) l_1 \exp(-\theta l_2/2) = R_{l_2} l_1 R_{l_2}^{\dagger}. \tag{94}$$

3.9. DQGA Intersection Operations

In this section, we discuss the intersection operations in DQGA. The intersection of two geometrical entities, where they have common points, is also called their meet. Intersection operations are also called meet operations. We give the operations for testing the meet of a point with a plane, line, or another point. We then give operations for forming entities that represent the meet of lines and planes.

3.9.1. Testing a point for intersection with a plane, line, or other point. We have defined the DQGA surface entities $A \in \{\overline{p}_{\mathbf{p}}, \pi_{\mathbf{p},\mathbf{n}}, l_{\mathbf{p},\mathbf{d}}\}$ with respect to testing a DQGA point $p_{\mathbf{t}}$ against them, to determine if $p_{\mathbf{t}}$ is in the *null space set* $\{p_{\mathbf{t}} : \mathrm{N}_A(p_{\mathbf{t}}A) = 0\}$ of the *null space entity part* $\mathrm{N}_A(p_{\mathbf{t}}A)$ of the test product $p_{\mathbf{t}}A$. The null space entity part $\mathrm{N}_A(p_{\mathbf{t}}A)$ depends on type of surface A.

For a point surface $p = \overline{p}_{\mathbf{p}}$, $A = p$, and

$$\mathrm{N}_p(p_{\mathbf{t}}\overline{p}_{\mathbf{b}}) = \mathrm{V}(\Im(p_{\mathbf{t}}\overline{p}_{\mathbf{b}})) = 0 \tag{95}$$

for the null space set of a single point $\{p_{\mathbf{t}} : \mathrm{N}_p(p_{\mathbf{t}}\overline{p}_{\mathbf{b}}) = (\mathbf{t}^* - \mathbf{b}^*)\mathbf{I}_4 = 0\} = \{p_{\mathbf{t}} = p_{\mathbf{b}}\}$.

For a plane surface $\pi = \pi_{\mathbf{p},\mathbf{n}}$, $A = \pi$, and

$$\mathrm{N}_\pi(p_{\mathbf{t}}\pi_{\mathbf{p},\mathbf{n}}) = \mathrm{S}(\Im(p_{\mathbf{t}}\pi_{\mathbf{p},\mathbf{n}})) = 0 \tag{96}$$

for the null space set of the entire plain of points $\{p_{\mathbf{t}} : \mathrm{N}_\pi(p_{\mathbf{t}}\pi_{\mathbf{p},\mathbf{n}}) = (\mathbf{t}^* \cdot \mathbf{n}^* - \mathbf{p}^* \cdot \mathbf{n}^*)\mathbf{I}_4 = 0\}$.

For a line surface $l = l_{\mathbf{p},\mathbf{d}}$, $A = l$, and

$$\mathrm{N}_l(p_{\mathbf{t}}l_{\mathbf{p},\mathbf{d}}) = \mathrm{V}(\Im(p_{\mathbf{t}}l_{\mathbf{p},\mathbf{d}})) = 0 \tag{97}$$

for the null space set of the entire line of points $\{p_{\mathbf{t}} : \mathrm{N}_l(p_{\mathbf{t}}l_{\mathbf{p},\mathbf{d}}) = (\mathbf{t}^* \times \mathbf{d}^* - \mathbf{p}^* \times \mathbf{d}^*)\mathbf{I}_4 = 0\}$, where $\mathbf{p}^* \times \mathbf{d}^* = \mathbf{m}^*$ is called the *moment* in Plücker coordinates $(\mathbf{d}^* : \mathbf{m}^*)$ for the line.

If we do not want to use the null space entity part operators N_A, then we can reformulate the point-surface intersection tests as full geometric products and make substitutions with identities to derive the full point-surface tests.

For the point-point test, in CPNS PGA we test $\boldsymbol{p}_{\mathbf{t}} \times \mathbf{p}_{\mathbf{p}} = (\boldsymbol{p}_{\mathbf{t}}\boldsymbol{p}_{\mathbf{p}} - \boldsymbol{p}_{\mathbf{p}}\boldsymbol{p}_{\mathbf{t}})/2 = 0$. Using identities $\boldsymbol{p} = p\mathbf{I}_3$ and $\overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}$, the test becomes $(p_{\mathbf{t}}\mathbf{I}_3 p_{\mathbf{p}}\mathbf{I}_3 - p_{\mathbf{p}}\mathbf{I}_3 p_{\mathbf{t}}\mathbf{I}_3)/2 = 0$. Therefore, two points $p_{\mathbf{t}}$ and $p_{\mathbf{p}}$ represent the same point if and only if

$$-(p_{\mathbf{t}}\overline{p}_{\mathbf{p}} - p_{\mathbf{p}}\overline{p}_{\mathbf{t}})/2 = 0. \tag{98}$$

We may save computation by using (95).

For the point-plane test, in CPNS PGA we test $\boldsymbol{p}_{\mathbf{t}} \times \boldsymbol{\pi} = (\boldsymbol{p}_{\mathbf{t}}\boldsymbol{\pi} - \boldsymbol{\pi}\boldsymbol{p}_{\mathbf{t}})/2 = 0$. Using identities $\{\boldsymbol{p} = p\mathbf{I}_3,\ \boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi,\ \overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}\}$, the test becomes $(p_{\mathbf{t}}\mathbf{I}_3\mathbf{I}_3^{-1}\pi - \mathbf{I}_3^{-1}\pi p_{\mathbf{t}}\mathbf{I}_3)/2 = 0$. Therefore, point $p_{\mathbf{t}}$ is on plane π if and only if

$$(p_{\mathbf{t}}\pi - \overline{\pi p_{\mathbf{t}}})/2 = 0. \tag{99}$$

We may save computation by using (96).

For the point-line test, in CPNS PGA we test $\boldsymbol{p}_{\mathbf{t}} \times \boldsymbol{l} = (\boldsymbol{p}_{\mathbf{t}}\boldsymbol{l} - \boldsymbol{l}\boldsymbol{p}_{\mathbf{t}})/2 = 0$. Using identities $\{\boldsymbol{p} = p\mathbf{I}_3,\ \boldsymbol{l} = \bar{l},\ \overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1},\ \mathbf{I}_3\bar{l} = l\mathbf{I}_3\}$, the test becomes $(p_{\mathbf{t}}\mathbf{I}_3\bar{l} - \bar{l}p_{\mathbf{t}}\mathbf{I}_3)/2 = 0$. Therefore, abridging $\mathbf{I}_3$ factored to the RHS, point $p_{\mathbf{t}}$ is on line l if and only if

$$(p_{\mathbf{t}}l - \bar{l}p_{\mathbf{t}})/2 = 0. \tag{100}$$

We may save computation by using (97).

3.9.2. Intersection of two planes as a line. In CPNS PGA, plane $\boldsymbol{\pi}_1$ and plane $\boldsymbol{\pi}_2$ intersect as the line $\boldsymbol{l} = \boldsymbol{\pi}_1 \wedge \boldsymbol{\pi}_2$. We use the identities $\{\boldsymbol{l} = \bar{l},\ \boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi,\ \overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}\}$ and switch to geometric products. Then, the line is $l = \bar{\boldsymbol{l}} = \mathrm{L}(\mathbf{I}_3\mathbf{I}_3^{-1}\pi_1\mathbf{I}_3^{-1}\pi_2\mathbf{I}_3^{-1})$. Therefore, the intersection of plane π_1 and plane π_2 is the line

$$l = \mathrm{L}(-\pi_1\overline{\pi}_2). \tag{101}$$

Notice that, we have to take the *line part* using the $\mathrm{L}(d) = \mathrm{V}(d)$ operator, which replaces the wedge product (a geometric product part operator) used in the corresponding CPNS PGA.

If we do not want to use $\mathrm{L}(d)$, then we can fully reformulate into geometric products as $\boldsymbol{l} = \boldsymbol{\pi}_1 \wedge \boldsymbol{\pi}_2 = (\boldsymbol{\pi}_1\boldsymbol{\pi}_2 - \boldsymbol{\pi}_2\boldsymbol{\pi}_1)/2$ for two vectors. Then, $l = \bar{\boldsymbol{l}} = \mathbf{I}_3(\mathbf{I}_3^{-1}\pi_1\mathbf{I}_3^{-1}\pi_2 - \mathbf{I}_3^{-1}\pi_2\mathbf{I}_3^{-1}\pi_1)\mathbf{I}_3^{-1}/2$. Therefore, the intersection of plane π_1 and plane π_2 is the line

$$l = -(\pi_1\overline{\pi}_2 - \pi_2\overline{\pi}_1)/2. \tag{102}$$

Using $\mathrm{L}(d)$ may save computation. Orientation is reversed as $-l$.

When forming a line l as intersection of two planes, we cannot be sure of its scale, so it may need to be normalized as $\hat{l} = \mathrm{U}(l)$, but it will have the same scale and orientation as the corresponding CPNS PGA line $\boldsymbol{l} = \bar{l}$.

3.9.3. Intersection of a line and plane as a point. In CPNS PGA, non-parallel line $\boldsymbol{l}$ and plane $\boldsymbol{\pi}$ intersect as the point $\boldsymbol{p} = \boldsymbol{l} \wedge \boldsymbol{\pi}$. We use the identities $\{\boldsymbol{l} = \bar{l},\ \boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi,\ \overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1},\ \boldsymbol{p} = p\mathbf{I}_3\}$ and switch to geometric products. Then, the point is $p = \boldsymbol{p}\mathbf{I}_3^{-1} = \mathrm{P}(\bar{l}\mathbf{I}_3^{-1}\pi\mathbf{I}_3^{-1})$. Therefore, the intersection of line l and plane π is the point

$$p = \mathrm{P}(-\bar{l}\overline{\pi}). \tag{103}$$

The *point part* $\mathrm{P}(d)$ operator replaces the wedge product (a geometric product part operator) used in the corresponding CPNS PGA.

If we do not want to use $\mathrm{P}(d)$, then we can fully reformulate into geometric products as $\boldsymbol{p} = \boldsymbol{l} \wedge \boldsymbol{\pi} = (\boldsymbol{l}\boldsymbol{\pi} + \boldsymbol{\pi}\boldsymbol{l})/2$. Then, $p = \boldsymbol{p}\mathbf{I}_3^{-1} = (\bar{l}\mathbf{I}_3^{-1}\pi + \mathbf{I}_3^{-1}\pi\bar{l})\mathbf{I}_3^{-1}/2$. Therefore, the intersection of line l and plane π is the point

$$p = -(\bar{l}\overline{\pi} + \overline{\pi}l)/2. \tag{104}$$

Using $\mathrm{P}(d)$ may save computation.

When forming a point p as intersection of line and plane, we cannot be sure of its scale, so it may need to be normalized as $\hat{p} = \mathrm{U}(p)$, but it will have the same scale and orientation as the corresponding CPNS PGA point $\boldsymbol{p} = p\mathbf{I}_3$.

3.9.4. Intersection of three planes as a point. In CPNS PGA, three non-parallel planes $\{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \boldsymbol{\pi}_3\}$ intersect as the point $\boldsymbol{p} = \boldsymbol{\pi}_1 \wedge \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_3$. We use the identities $\{\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi, \boldsymbol{p} = p\mathbf{I}_3, \overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}\}$ and switch to geometric products. Then, the point is $p = \boldsymbol{p}\mathbf{I}_3^{-1} = \mathrm{P}(\mathbf{I}_3^{-1}\pi_1\mathbf{I}_3^{-1}\pi_2\mathbf{I}_3^{-1}\pi_3\mathbf{I}_3^{-1})$. Therefore, the intersection of three planes, π_1, π_2, and π_3, is the point

$$p = \mathrm{P}(\overline{\pi}_1\pi_2\overline{\pi}_3) = \mathrm{P}(-\bar{l}_{12}\overline{\pi}_3) = \mathrm{P}(-\overline{\pi}_1 l_{23}). \tag{105}$$

The *point part* $\mathrm{P}(d)$ operator replaces the wedge products (a geometric product part operator) used in the corresponding CPNS PGA.

If we do not want to use $\mathrm{P}(d)$, then we can fully reformulate into geometric products as $p = \boldsymbol{p}\mathbf{I}_3^{-1} = (\boldsymbol{\pi}_1 \wedge \boldsymbol{\pi}_2 \wedge \boldsymbol{\pi}_3)\mathbf{I}_3^{-1} = ((\boldsymbol{\pi}_1\boldsymbol{\pi}_2 - \boldsymbol{\pi}_2\boldsymbol{\pi}_1)\boldsymbol{\pi}_3 + \boldsymbol{\pi}_3(\boldsymbol{\pi}_1\boldsymbol{\pi}_2 - \boldsymbol{\pi}_2\boldsymbol{\pi}_1))\mathbf{I}_3^{-1}/4$. Then, we make the substitutions and get $p = ((-\overline{\pi}_1\pi_2+\overline{\pi}_2\pi_1)(-\overline{\pi}_3)+(-\overline{\pi}_3)(-\pi_1\overline{\pi}_2+\pi_2\overline{\pi}_1))/4$. Therefore, the intersection of three planes, π_1, π_2, and π_3, is the point

$$p = (\overline{\pi}_1\pi_2\overline{\pi}_3 - \overline{\pi}_2\pi_1\overline{\pi}_3 + \overline{\pi}_3\pi_1\overline{\pi}_2 - \overline{\pi}_3\pi_2\overline{\pi}_1)/4. \tag{106}$$

Using $\mathrm{P}(d)$ may save computation, but both ways give the same point entity.

When forming a point p as the intersection of three planes, we cannot be sure of its scale, so it may need to be normalized as $\hat{p} = \mathrm{U}(p)$, but it will have the same scale and orientation as the corresponding CPNS PGA point $\boldsymbol{p} = p\mathbf{I}_3$.

3.10. DQGA Projection Operations

In CPNS PGA, we only make projections of a smaller-dimensional geometric entity $a \in \{\boldsymbol{p}, \boldsymbol{l}, \boldsymbol{\pi}\}$ onto a subspace of a larger-dimensional geometric entity $A \in \{\boldsymbol{p}, \boldsymbol{l}, \boldsymbol{\pi}\}$. A point is 0-dimensional, a line is 1-dimensional, and a plane is 2-dimensional in terms of geometric degrees of freedom. The general projection operation in CPNS PGA is $a' = (a \cdot A)A^{-1}$ (same as in CGA) and results in orthographic projection of a onto A. Therefore, we have three projections: $(\boldsymbol{p} \cdot \boldsymbol{\pi})\boldsymbol{\pi}^{-1}$, $(\boldsymbol{l} \cdot \boldsymbol{\pi})\boldsymbol{\pi}^{-1}$, $(\boldsymbol{p} \cdot \boldsymbol{l})\boldsymbol{l}^{-1}$. In this section, we use identities to convert these projections into DQGA forms.

3.10.1. Projection of a point onto a plane. In CPGA PGA, the projection $\boldsymbol{p}'$ (22) of point $\boldsymbol{p}$ onto unit plane $\boldsymbol{\pi}$ is $\boldsymbol{p}' = (\boldsymbol{p} \cdot \boldsymbol{\pi})\boldsymbol{\pi}$. We use the identities $\{\boldsymbol{p} = p\mathbf{I}_3, \boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi, \overline{d} = \mathbf{I}_3 d\mathbf{I}_3^{-1}, \pi\overline{\pi} = -1, \boldsymbol{p} \cdot \boldsymbol{\pi} = (\boldsymbol{p}\boldsymbol{\pi} + \boldsymbol{\pi}\boldsymbol{p})/2\}$ and switch to geometric products. Then, $p' = \boldsymbol{p}'\mathbf{I}_3^{-1} = ((p\mathbf{I}_3\mathbf{I}_3^{-1}\pi + \mathbf{I}_3^{-1}\pi p\mathbf{I}_3)\mathbf{I}_3^{-1}\pi/2)\mathbf{I}_3^{-1}$. Therefore, the projection of point p onto plane π is the point

$$p' = -(p\pi + \overline{\pi p})\overline{\pi}/2. \tag{107}$$

The projected point p' is the point on the plane π that is closest to the point p.

For unit plane π, we have $\pi\overline{\pi} = -1$ and $p' = (p - \overline{\pi p \pi})/2$. We notice that, $-\overline{\pi p \pi}$ (87) is p reflected in the plane π as a non-oriented point, and that $(p - \overline{\pi p \pi})/2$ is the average point between p and $-\overline{\pi p \pi}$, which is the point p' on the plane between them.

3.10.2. Projection of a line onto a plane. In CPNS PGA, the projection $\boldsymbol{l}'$ (23) of line $\boldsymbol{l}$ onto unit plane $\boldsymbol{\pi}$ is $\boldsymbol{l}' = (\boldsymbol{l} \cdot \boldsymbol{\pi})\boldsymbol{\pi}$. We use the identities $\{\boldsymbol{l} = \bar{l}$, $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$, $\overline{d} = \mathbf{I}_3 d \mathbf{I}_3^{-1}$, $\pi\overline{\pi} = -1$, $\boldsymbol{l} \cdot \boldsymbol{\pi} = (\boldsymbol{l}\boldsymbol{\pi} - \boldsymbol{\pi}\boldsymbol{l})/2\}$ and switch to geometric products. Then, $l' = \overline{\boldsymbol{l}'} = \mathbf{I}_3((\bar{l}\mathbf{I}_3^{-1}\pi - \mathbf{I}_3^{-1}\pi\bar{l})\mathbf{I}_3^{-1}\pi/2)\mathbf{I}_3^{-1}$. Therefore, the projection of line l onto plane π is the line

$$l' = (-l\pi + \pi\bar{l})\overline{\pi}/2. \tag{108}$$

For unit plane π, we have $\pi\overline{\pi} = -1$ and then $l' = (l + \pi\bar{l}\overline{\pi})/2$. We notice that, $-\pi\bar{l}\overline{\pi}$ (85) is l reflected in the plane π. Reflection in a plane causes an orientation reversal in the reflected entity (except for the non-oriented point reflection). Then, $\pi\bar{l}\overline{\pi}$ restores orientation to that of l, and then they are averaged as l'. The average line l' is on the plane between them.

3.10.3. Projection of a point onto a line. In CPNS PGA, the projection $\boldsymbol{p}'$ (24) of point $\boldsymbol{p}$ onto the unit line $\boldsymbol{l}$ is $\boldsymbol{p}' = (\boldsymbol{p} \cdot \boldsymbol{l})\boldsymbol{l}^{-1} = -(\boldsymbol{p} \cdot \boldsymbol{l})\boldsymbol{l}$. We use the identities $\{\boldsymbol{p} = p\mathbf{I}_3$, $\boldsymbol{l} = \bar{l}$, $\overline{d} = \mathbf{I}_3 d \mathbf{I}_3^{-1}$, $\mathbf{I}_3\bar{l} = l\mathbf{I}_3$, $\boldsymbol{p} \cdot \boldsymbol{l} = (\boldsymbol{p}\boldsymbol{l} + \boldsymbol{l}\boldsymbol{p})/2\}$ and switch to geometric products. Then, $\boldsymbol{p}' = p'\mathbf{I}_3 = -(p\mathbf{I}_3\bar{l} + \bar{l}p\mathbf{I}_3)\bar{l}/2$. Therefore, the projection of point p onto line l is the point

$$p' = -(pl + \bar{l}p)l/2. \tag{109}$$

For unit line l, we have $ll = l^2 = -1$ and $p' = (p - \bar{l}pl)/2$. We notice that, $-\bar{l}pl$ (92) is p "reflected" in l, which is actually an orientation-preserving rotation around l by 180°. Then, p' is the average point between p and $-\bar{l}pl$, which is the projected point on the line.

3.11. DQGA Rejection Operations

3.11.1. Rejection of a line from a plane. In CPNS PGA, the rejection $\boldsymbol{l}'$ (25) of line $\boldsymbol{l}$ from unit plane $\boldsymbol{\pi}$ is $\boldsymbol{l}' = (\boldsymbol{l} \wedge \boldsymbol{\pi})\boldsymbol{\pi}$. We use the identities $\{\boldsymbol{l} = \bar{l}$, $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$, $\boldsymbol{l} \wedge \boldsymbol{\pi} = (\boldsymbol{l}\boldsymbol{\pi} + \boldsymbol{\pi}\boldsymbol{l})/2$, $\overline{d} = \mathbf{I}_3 d \mathbf{I}_3^{-1}\}$ and switch to geometric products. Then, $\boldsymbol{l}' = \bar{l}' = (\bar{l}\mathbf{I}_3^{-1}\pi + \mathbf{I}_3^{-1}\pi\bar{l})\mathbf{I}_3^{-1}\pi/2$. Therefore, the rejection of line l from plane π is the line

$$l' = -(l\pi + \pi\bar{l})\overline{\pi}/2. \tag{110}$$

3.11.2. Rejection of a plane from a line. In CPNS PGA, the rejection $\boldsymbol{\pi}'$ (26) of plane $\boldsymbol{\pi}$ from unit line $\boldsymbol{l}$ is $\boldsymbol{\pi}' = (\boldsymbol{\pi} \wedge \boldsymbol{l})\boldsymbol{l}^{-1} = -(\boldsymbol{\pi} \wedge \boldsymbol{l})\boldsymbol{l}$. We use the identities $\{\boldsymbol{l} = \bar{l}$, $\boldsymbol{\pi} = \mathbf{I}_3^{-1}\pi$, $\boldsymbol{\pi} \wedge \boldsymbol{l} = (\boldsymbol{\pi}\boldsymbol{l} + \boldsymbol{l}\boldsymbol{\pi})/2$, $\overline{d} = \mathbf{I}_3 d \mathbf{I}_3^{-1}\}$ and switch to geometric products. Then, $\boldsymbol{\pi}' = \mathbf{I}_3^{-1}\pi' = -(\mathbf{I}_3^{-1}\pi\bar{l} + \bar{l}\mathbf{I}_3^{-1}\pi)\bar{l}/2$. Therefore, the rejection of plane π from line l is the plane

$$\pi' = -(\pi\bar{l} + l\pi)\bar{l}/2. \tag{111}$$

3.12. Conclusion to Dual Quaternion Geometric Algebra

We reviewed dual number algebra (DNA), quaternion algebra (QA), and dual quaternion algebra (DQA) in their original notations, and then provided the details on how each algebra is represented in the even-grades subalgebra $\mathcal{G}^+_{3,0,1}$ of PGA. The representations in geometric algebra are called the dual number geometric algebra (DNGA), quaternion geometric algebra (QGA), and dual quaternion geometric algebra (DQGA) to distinguish the different notations and implementations of certain special operations by taking advantage of the larger $\mathcal{G}_{3,0,1}$ and PGA. We have discussed the Dual Quaternion Geometric Algebra $\mathcal{G}^+_{3,0,1}$ in PGA $\mathcal{G}_{3,0,1}$ (DQGA/PGA) in much detail that seemed to be missing in the prior literature that we are aware of, though we acknowledge that dual quaternions are an old subject and some results may be found somewhere in the published literature.

We arrive at our view of dual quaternions by a method that may be new in the literature, by using PGA. We derived identities that convert the CPNS PGA entities $\{\boldsymbol{p}, \boldsymbol{l}, \boldsymbol{\pi}\}$ between their corresponding DQGA entities $\{p, l, \pi\}$ without change of orientation. By using the identities, we also converted CPNS PGA operations on entities into their corresponding operations on the DQGA entities, including reflections, translations, rotations, intersections, projections, and rejections. Nearly anything that can be done in CPNS PGA can also be done in DQGA (dual quaternions). All of the DQGA entities and operations could be implemented in a pure DQA implementation that may be more efficient than the full PGA algebra.

We borrow from PGA the entity dualization operation J_e, pseudoscalar $\mathbf{I}_3$, and certain geometric algebra operators (reverse $\dagger$ or $\tilde{}$, inner product $\cdot$, commutator product $\times$) to implement a complete set of special dual quaternion operations including conjugates (dual number conjugate $\overline{z}$, quaternion conjugate $\mathrm{K}(q) = q^\dagger$, dual conjugate $\overline{d}^\dagger$), dual quaternion part extraction operators (real $\Re$, imaginary $\Im$, scalar S, vector V, point P, plane Π, and line L parts), the dual number-valued tensor T (magnitude) of a dual quaternion, a normalization operation $\hat{d} = \mathrm{U}(d)$ of a dual quaternion d to a unit dual quaternion $\hat{d}$, the vector calculus dot product $\mathbf{a}^* \bullet \mathbf{b}^* = -\mathbf{a}^* \cdot \mathbf{b}^*$ and cross product $\mathbf{a}^* \times \mathbf{b}^* = \mathbf{a}^* \times \mathbf{b}^*$, and an operation Y (using J_e and $\Im$) to extract the quaternion $q_2 = \mathrm{Y}(d)$ from $d = q_1 + q_2\mathbf{I}_4$ ($\varepsilon \hat{=} \mathbf{I}_4$). Using the point P, plane Π, and line L parts operations, we improve the computational efficiency of intersection operations. DQGA is a complete implementation of DQA, and it is also extended in PGA to other algebraic forms of the entities and operations in CPNS PGA and OPNS PGA.

It is possible to convert between the DQGA entities $\{p, l, \pi\}$ and the CPNS PGA entities $\{\boldsymbol{p}, \boldsymbol{l}, \boldsymbol{\pi}\}$ by using simple identities between them. Using the PGA entity dualization operation J_e, we can convert between CPNS PGA entities $\{\boldsymbol{p}, \boldsymbol{l}, \boldsymbol{\pi}\}$ and OPNS PGA entities $\{\mathbf{P}, \mathbf{L}, \mathbf{\Pi}\}$. Therefore, we can freely convert a point, line, or plane entity into three different forms within PGA $\mathcal{G}_{3,0,1}$ without orientation change, and take advantage of each form of entity and operations in the three different algebras in PGA.

4. Conclusion

In Section 1, we introduced the subject of this paper, which is about the geometric algebra PGA $\mathcal{G}_{3,0,1}$ for points, lines and planes, and its Dual Quaternion Geometric Algebra within its even-grades subalgebra $\mathcal{G}^+_{3,0,1}$. The main contribution of this paper is the detailed development of the Dual Quaternion Geometric Algebra (DQGA), providing more geometric entities and operations in dual quaternions than we were able to find in any prior literature, including recent literature such as [6].

In Section 2 and its subsections, we provided a detailed review and introduction to 3D PGA $\mathcal{G}_{3,0,1}$ to provide the foundation for Section 3 on the Dual Quaternion Geometric Algebra.

In Section 3, as the main contribution of this paper, we explored the details of the Dual Quaternion Geometric Algebra (DQGA) within the even-grades subalgebra $\mathcal{G}^+_{3,0,1}$ of PGA. In DQGA, we rediscovered many results that may be known in older published literature, while we may have contributed some new results on representing lines and planes and various operations on them that are derived through identities to the CPNS PGA entities and operations.

The details of the Dual Quaternion Geometric Algebra may contribute to the literature on Dual Quaternions, showing that much more can be done with dual quaternions than seems to be commonly known. Some old literature could turn up with many of the results, but we are not aware of it. The dual quaternions could eventually be superseded by the Plane-based algebra of PGA that has a nicer and simpler form. There is also the point-based algebra of PGA through the dualization J_e that offers the ability to join points, which we did not find in dual quaternions.

In comparing CGA $\mathcal{G}_{4,1}$ to PGA $\mathcal{G}_{3,0,1}$, PGA has some advantages. In CGA, the embedding of a vector point $\mathbf{t}$ is the conformal embedding $\mathbf{P} = \mathbf{P_t} = \mathbf{t} + \mathbf{t}^2\mathbf{e}_\infty/2 + \mathbf{e}_o$, where $\mathbf{P}^2 = 0$. The square $\mathbf{t}^2$ increases numerical error, and then the condition $\mathbf{P}^2 = 0$ can fail, leading to $\mathbf{P}^2 \neq 0$ and further instability or errors. In PGA, the embedding of a vector $\mathbf{t}$ is the homogeneous embedding $\mathbf{P} = \mathbf{P_t} = \mathbf{e}_0 + \mathbf{t}$ (or its dual $\boldsymbol{p}_\mathbf{t}$), where there is no $\mathbf{t}^2$ or condition on the square $\mathbf{P}^2$. The homogeneous embedding is more stable and less sensitive to numerical error. PGA is a smaller algebra than CGA and is likely to be faster. Compared to CGA for points, lines, and planes, DQGA (and DQA) also has the same advantages as PGA. A further advantage of dual quaternions may be existing software implementations of DQA, where dual quaternions are already used for some algorithms. The DQA entities and operations introduced in this paper may enable new applications of dual quaternions using existing software. For those who are wanting to try to get the most out of an efficient dual quaternion implementation, the DQGA entities and operations for points, lines, and planes may be of interest.

References

[1] Bayro-Corrochano, E., Lasenby, J.: The Geometry Algebra of Computer Vision. In: E. Bayro-Corrochano, G. Sobczyk (eds.) Geometric Algebra with Applications in Science and Engineering, pp. 123–146. Birkhäuser Boston, MA (2001). doi:10.1007/978-1-4612-0159-5_7

[2] Bromborsky, A., Song, U., Wieser, E., Hadfield, H., The Pygae Team: pygae/-galgebra: v0.5.0 (2020). doi:10.5281/zenodo.3875882

[3] Dorst, L., De Keninck, S.: PGA4CS: A Guided Tour to the Plane-Based Geometric Algebra PGA, Version 2.0 – March 14, 2022. https://bivector.net/PGA4CS.html. Accessed: 2023-11-16

[4] Dorst, L., Fontijne, D., Mann, S.: Geometric Algebra for Computer Science (Revised Edition). The Morgan Kaufmann Series in Computer Graphics. Morgan Kaufmann (2009). doi:10.1016/B978-0-12-374942-0.X0000-0

[5] Easter, R.B., Pimchangthong, D.: 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: 2023-12-20

[6] Goldman, R.: Dual Quaternions and Their Associated Clifford Algebras, 1st edn. CRC Press (2023). doi:10.1201/9781003398141

[7] Gunn, C.: Geometry, Kinematics, and Rigid Body Mechanics in Cayley-Klein Geometries. Ph.D. thesis, Technical University Berlin (2011)

[8] Gunn, C.: On the Homogeneous Model of Euclidean Geometry. In: L. Dorst, J. Lasenby (eds.) Guide to Geometric Algebra in Practice, chap. 15, pp. 297–327. Springer, London (2011). doi:10.1007/978-0-85729-811-9_15

[9] Gunn, C.: Doing Euclidean Plane Geometry Using Projective Geometric Algebra. Advances in Applied Clifford Algebras **27**(2), 1203–1232 (2017). doi:10.1007/s00006-016-0731-5

[10] Gunn, C.: Geometric Algebras for Euclidean Geometry. Advances in Applied Clifford Algebras **27**(1), 185–208 (2017). doi:10.1007/s00006-016-0647-0

[11] Hamilton, W.R.: Lectures on Quaternions. Dublin: Hodges and Smith, Grafton-Street, Booksellers to the University. London: Whittaker & Co., Ave-Maria Lane. Cambridge: Macmillan & Co. (1853). URL https://archive.org/details/lecturesonquater1853hami

[12] Hestenes, D., Sobczyk, G.: Clifford Algebra to Geometric Calculus, *Fundamental Theories of Physics*, vol. 5. Springer Dordrecht (1984). doi:10.1007/978-94-009-6292-7

[13] Perwass, C.: Geometric Algebra with Applications in Engineering, *Geometry and Computing*, vol. 4. Springer Berlin, Heidelberg (2009). doi:10.1007/978-3-540-89068-3

[14] Selig, J.: Clifford algebra of points, lines and planes. Robotica **18**(5), 545–556 (2000). doi:10.1017/S0263574799002568

Robert Benjamin Easter
Independent Researcher,
Bangkok, Thailand
e-mail: reaster2015@gmail.com

Daranee Pimchangthong
Rajamangala University of Technology Krungthep
Institute of Science Innovation and Culture (UTKISIC),
Bangkok, Thailand
e-mail: daranee.p@mail.rmutk.ac.th