Rigid-Body Pose and Position Interpolation using Geometric Algebra

R. Wareham *, J. Lasenby

Cambridge University Engineering Department, Trumpington Street, Cambridge, CB2 1PZ, United Kingdom

Abstract

This paper presents a method of interpolating between two or more general displacements (rotation and translation). The resulting interpolated path is smooth and possesses a number of desirable properties. It differs from existing algorithms which require factorising the pose into separate rotation and translation components and is derived from an intuitively appealing framework – i.e. a natural extension of the standard interpolation scheme for pure rotations. The mathematical framework used for the derivation is that of geometric algebra. While this paper presents the theory behind the interpolation and its description as a tool, we also outline the possible advantages of using this technique for vision and graphics applications.

Key words: Geometric Algebra, keyframes, pose interpolation, SLERP, non-Euclidean geometry

1 Introduction

Geometric Algebra, the application of Clifford algebras to geometric problems (Hestenes and Sobczyk, 1984), has recently been found to have useful applications in many areas of Physics and Engineering (Hestenes and Sobczyk, 1984; Lasenby and Bayro-Corrochano, 1998; Lasenby, 2003), often resulting in a more compact and clear description of a problem whilst also providing significant geometric insight. Good introductory material may be found in Rockwood et al. (2001) but a brief introduction follows.

* Corresponding author

Email addresses: `rjw57@eng.cam.ac.uk` (R. Wareham), `jl@eng.cam.ac.uk` (J. Lasenby).

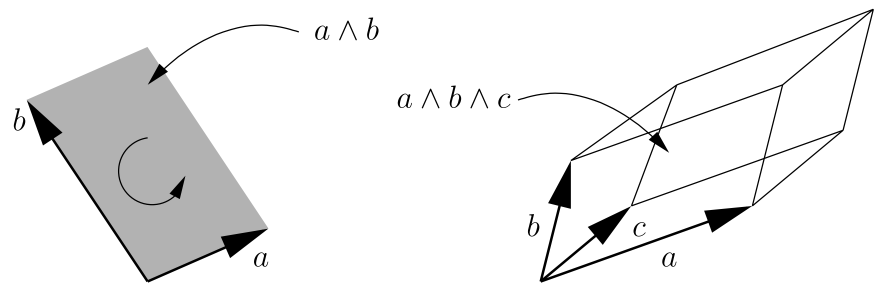

Fig. 1. Bivectors are directed area segments, trivectors are directed parallelepipeds.

Clifford algebras provide a natural method for forming a complete algebra over the vectors without resolving the vectors into some co-ordinate system. Vectors may be combined using either the *inner* or *outer* products. The inner-product $(a \cdot b)$ of two vectors is a scalar, analogous to the dot-product in traditional vector algebra. The outer-product $(a \wedge b)$ creates a directed area segment which we term a *bivector*. A bivector retains full information about the orientation of the vectors which created it but does not retain individual magnitudes (see figure 1). A bivector may further be combined with another vector to produce a directed volume, a trivector. This process may be repeated until the dimension of the space is exhausted.

We say that a scalar has a lower *grade* than a vector and, in turn, a bivector has a higher grade than a vector; scalars have grade zero, vectors have grade one, bivectors have grade two and so forth. When the inner product of a higher and a lower grade object is taken the result will be of lower grade than the original higher grade object. When the outer product is taken the orthogonal components of the original objects will result in a multivector composed of higher grade objects. The outer-product anti-commutes when dealing with vectors, i.e. $a \wedge b = -b \wedge a$. This can be interpreted as changing the 'handedness' of the directed area.

To unify the inner and outer products they are combined into one *geometric* product given, for vectors, by

$$ab = a \wedge b + a \cdot b.$$

Initially one may be concerned at the addition of different grade elements (if a and b are vectors, we are adding a bivector and scalar). The reader may be reassured by indicating the similarity to complex numbers, whereby a real number is added to an imaginary number. Just as some linear combination of real and imaginary numbers is termed a complex number, we shall term some linear combination of different-grade elements a *multivector*.

Notice that we have avoided the usual typographic convention of emboldening vectors. This is because vectors are now no longer special case elements in our algebra and may be considered with equal standing to scalars, bivectors and, in general, n-vectors.

In this paper we will consider the use of *Conformal Geometric Algebra* (Hestenes and Sobczyk, 1984; Lasenby et al., 2004) which maps n-dimensional vectors into an $(n+2)$-dimensional subspace of null-vectors. The extended space these null-vectors occupy is obtained by adding two more orthogonal basis vectors, e and $\bar{e}$ such that $e^2 = 1$, $\bar{e}^2 = -1$. The null-vectors themselves are mapped from the original vectors via an adaptation (Lasenby et al., 2004) of the so-called *Hestenes mapping* (Hestenes and Sobczyk, 1984):

$$x \mapsto \frac{1}{2\lambda^2}\left(x^2 n + 2\lambda x - \lambda^2 \bar{n}\right)$$

where $n = e + \bar{e}$ and $\bar{n} = e - \bar{e}$. The constant λ is an arbitrary length which is present to keep dimensional consistency. For Euclidean geometries its value is usually kept at unity to simplify further calculation. We often express this mapping via the function $F(\cdot)$ such that the n-dimensional vector x is represented by the $(n+2)$-dimensional null-vector $F(x)$.

We shall use the term *general displacement*, or simply *displacement*, to denote an arbitrary rigid-body transformation, combination of pure-rotation and translation, within a particular geometry. We shall also use, unless stated otherwise, Euclidean three-dimensional geometry although the techniques presented here readily generalise into higher-dimensions and even non-Euclidean geometries.

2 Bivector Exponentiation

2.1 Review of rotors

Consider three orthonormal basis vectors of $\mathbb{R}^3$, $\{e_1, e_2, e_3\}$. We can form 3 different bivectors from these vectors:

$$B_1 = e_2 e_3, \quad B_2 = e_3 e_1, \quad B_3 = e_1 e_2$$

Note that these are indeed bivectors since

$$e_i e_j = e_i \cdot e_j + e_i \wedge e_j = e_i \wedge e_j \quad \text{iff} \quad i \neq j$$

Now consider the effect of B_3 on the vectors e_1 and $e_1 + e_2$:

$$e_1 B_3 = e_1 e_1 e_2 = e_1^2 e_2 = e_2$$

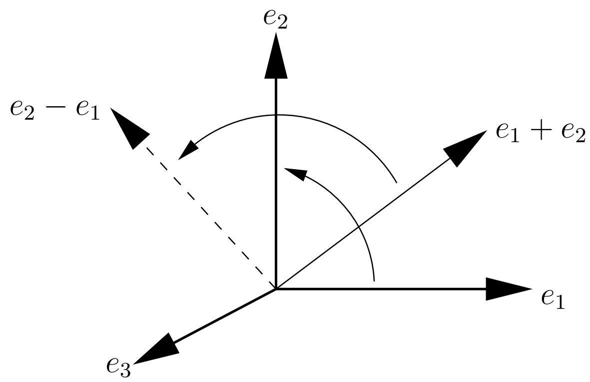

Fig. 2. The rotation effect of bivector $B_3 = e_1e_2$

$$(e_1 + e_2)B_3 = e_1B_3 + e_2B_3 = e_2 + e_2e_1e_2 = e_2 - e_1e_2^2 = e_2 - e_1$$

By looking at figure 2 it is clear that B_3 has the effect of rotating the vectors counter-clockwise by 90°. It is, in fact, a general property that the bivector e_ie_j will rotate a vector 90° in the plane defined by e_i and e_j. At first glance this seems to offer little more than quaternions but at no point have we assumed that we are working in 3-dimensional space; this method also works in higher-dimension spaces.

Now we extend to general rotations. Firstly it is trivial to show that B_3 squares to -1:

$$B_3^2 = e_1e_2e_1e_2 = -e_1e_2e_2e_1 = -1$$

We can represent any vector z in the plane defined by e_1 and e_2 using

$$z = r(e_1 \cos\theta + e_2 \sin\theta) = e_1r(\cos\theta + B_3 \sin\theta)$$

where r is the distance of the point z from the origin and θ is the angle z makes to e_1. By taking the Taylor expansion about the origin of cosine and sine and re-arranging the coefficients it can be shown that

$$e^{B_3\theta} = \cos\theta + B_3 \sin\theta$$

which is the analogous form of de Moivre's theorem for complex numbers.

We can thus represent any vector z which lies in the plane of the bivector B_3 by

$$z = e_1re^{B_3\theta}$$

From this, the same argument used for rotation in the complex plane can be used to show that rotation by ϕ radians in the plane of B_3 is accomplished by $z \mapsto z'$ where

$$z' = ze^{B_3\phi} = z(\cos\phi + B_3 \sin\phi)$$

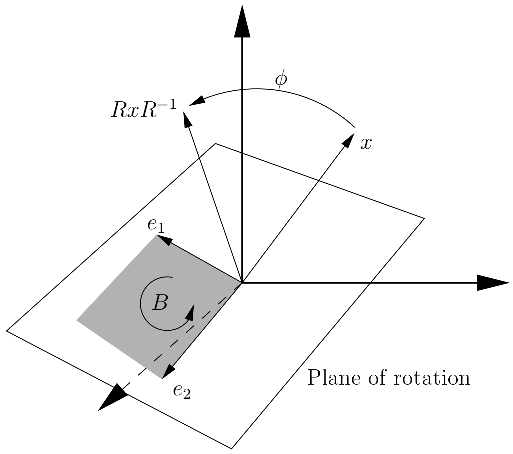

Fig. 3. Rotating vectors in arbitrary planes

This has all taken place in two dimensions for the moment but note that nothing in our development has assumed this. In fact we can define a bivector $B_3 = ab$ in three dimensions and rotate vectors in the plane defined by a and b using the expression above.

Careful consideration must be given to the case where the vector to be rotated, x, does not lie on the plane of rotation as in figure 3. Firstly decompose the vector into a component which lies in the plane $x_\parallel$ and one normal to the plane $x_\perp$

$$x = x_\parallel + x_\perp$$

Now consider the effect of the following

$$\begin{aligned} e^{-B_3\phi/2} x e^{B_3\phi/2} &= \left(\cos\frac{\phi}{2} - B_3 \sin\frac{\phi}{2}\right)(x_\parallel + x_\perp)\left(\cos\frac{\phi}{2} + B_3 \sin\frac{\phi}{2}\right) \\ &= x_\parallel(\cos\phi + B_3 \sin\phi) + x_\perp \end{aligned}$$

since bivectors anti-commute with vectors in their plane (e.g. $e_1(e_2e_1) = -e_2 = -(e_2e_1)e_1$) and commute with vectors normal to the plane (e.g. $e_1(e_2e_3) = (e_2e_3)e_1$). We have thus succeeded in rotating the component of the vector which lies in the plane without affecting the component normal to the plane — we have rotated the vector around an axis normal to the plane.

This leads to a general method of rotation in any plane; we form a bivector of the form $R = e^{-B\phi/2}$ for a given rotation ϕ in a plane specified by the bivector B and the transformation is given by

$$x \mapsto RxR^{-1}$$

We refer to these bivectors which have a rotational effect as *rotors*. Figure 3 shows the various objects used.

The computation of R^{-1} is rather difficult analytically (and indeed can require a full 2^n-dimension matrix inversion for a space of dimension n). To combat this we define the *reversion* of a n-vector $X = e_i e_j ... e_k$ as

$$\tilde{X} = e_k ... e_j e_i$$

i.e. the literal reversion of the components. By looking at the expression for R it is clear that $\tilde{R} \equiv R^{-1}$ for rotors. Computing $\tilde{R}$ is easier since it is simply a permutation.

Note that in spaces with dimension n the maximum grade object possible is an n-grade one. We denote the n-vector $e_1 \wedge ... \wedge e_n = I$ as the *pseudoscalar* and the product xI as the *dual* of a vector or x^*. The dual is also defined for general multivectors.

It is worth comparing this method of rotation to quaternions. The three bivectors B_1, B_2 and B_3 act identically to the three imaginary components of quaternions, $\mathbf{i}, -\mathbf{j}$ and $\mathbf{k}$ respectively. The sign difference between B_2 and $\mathbf{j}$ is due to the fact that the quaternions are not derived from the usual right-handed orthogonal co-ordinate system.

A particular rotation is represented via the quaternion q given by

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$$

where $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. Interpolation between rotations is then performed by interpolating each of q_i over the surface of a four-dimensional hyper-sphere. If we are interpolating between unit quaternions q_0 and q_1 the SLERP interpolation is

$$q = \begin{cases} q_0(q_0^{-1}q_1)^\lambda & \text{if } q_0 \cdot q_1 \geq 0 \\ q_0(q_0^{-1}(-q_1))^\lambda & \text{otherwise} \end{cases}$$

where λ varies in the range $(0, 1)$ (Lillholm et al., 1998).

Recalling that, in complex numbers, the locus of $\exp(i\theta)$ is the unit circle, it is somewhat simple to show that, for some normalised bivector B, the locus of the action of $\exp(B\theta)$ upon a point with respect to varying θ is also a circle in the plane of B. Therefore if we consider some rotations $R_1, R_2 = \exp(kB)R_1$, where k is a scalar and B is some normalised bivector, it is easy to see after some thought that the quaternionic interpolation is exactly given by

$$R_\lambda = (R_2\tilde{R}_1)^\lambda R_1 = \exp(\lambda k B)R_1$$

where λ, the interpolation parameter, varies in the range $(0, 1)$. A further moment's thought will reveal that this method is not confined to three dimensions, like quaternionic interpolation, but instead readily generalises to higher-dimensions.

2.2 Conformal Geometric Algebra (CGA)

In the Conformal Model (Hestenes and Sobczyk, 1984) we extend the space by adding two additional basis vectors. We first define the *signature*, (p, q) of a space $\mathcal{A}(p, q)$ with basis vectors, $\{e_i\}$, such that $e_i^2 = +1$ for $i = 1, ..., p$ and $e_j^2 = -1$ for $j = p+1, ..., p+q$. For example $\mathbb{R}^3$ would be denoted as $\mathcal{A}(3, 0)$. We extend $\mathcal{A}(3, 0)$ so that it becomes mixed signature and is defined by the basis

$$\{e_1, e_2, e_3, e, \bar{e}\}$$

where e and $\bar{e}$ are defined so that

$$e^2 = 1, \quad \bar{e}^2 = -1, \quad e \cdot \bar{e} = 0$$

$$e \cdot e_i = \bar{e} \cdot e_i = 0 \quad \forall\, i \in \{1, 2, 3\}$$

This space is denoted as $\mathcal{A}(4, 1)$. In general a space $\mathcal{A}(p, q)$ is extended to $\mathcal{A}(p+1, q+1)$. We may now define the vectors n and $\bar{n}$:

$$n = e + \bar{e}, \quad \bar{n} = e - \bar{e}$$

It is simple to show by direct substitution that both n and $\bar{n}$ are *null vectors* (i.e. $n^2 = \bar{n}^2 = 0$). It can be shown (Hestenes and Sobczyk, 1984; Lasenby et al., 2004) that general displacements may be represented conveniently via rotors in this algebra. Pure-rotation rotors are left unchanged whereas the translation rotor T_a is defined as

$$T_a = \exp\left(\frac{na}{2}\right) = 1 + \frac{na}{2}$$

and will transform a null-vector representation of the vector x to the null-vector representation of $x + a$ in the following manner:

$$F(x_a) = T_a F(x) \tilde{T}_a$$

2.3 Justification for using exponentiation

Referring to the displacement rotors presented above, we see that all of them have a common form; they are all exponentiated bivectors. Rotations are generated by bivectors with no component parallel to n and translations by a bivector with no components perpendicular to n. We may thus postulate that all displacement rotors [1] can be expressed as

$$R = \exp(B)$$

where B is the sum of two bivectors, one formed from two vectors which have no components parallel to e or $\bar{e}$. The other is formed from the outer product of vectors with no components parallel to e or $\bar{e}$ and n. The effect of this is to separate the basis bivectors of B into one with components of the form $e_i \wedge e_j$ and one with components of the form $e_i \wedge e$ and $e_i \wedge \bar{e}$.

We shall proceed assuming that all displacement rotors can be written as the exponentiation of a bivector of the form $B = ab + cn$ where a, b and c are independent of n, i.e. if $n \in \mathcal{A}(m+1,1)$ then $\{a, b, c\} \in \mathbb{R}^m$. It is clear that the set of all B is some linear sub-space of all the bivectors.

We now suppose that we may interpolate rotors by defining some function $\ell(R)$ which acts upon rotors to give the generating bivector element. We then perform direct interpolation of these generators. We postulate that direct interpolation of these bivectors, as in the reformulation of quaternionic interpolation above, will give some smooth interpolation between the displacements. It is therefore a defining property of $\ell(R)$ that

$$R \equiv \exp(\ell(R)) \tag{1}$$

and so $\ell(R)$ may be considered as to act as a logarithm-like function in this context. It is worth noting that $\ell(R)$ does not possess all the properties usually associated with logarithms, notably that, since $\exp(A)\exp(B)$ is not generally equal to $\exp(B)\exp(A)$ in non-commuting algebras, $\ell(\exp(A)\exp(B))$ cannot be equal to $A + B$ except in special cases.

To avoid the the risk of assigning more properties to $\ell(R)$ than we have shown, we shall resist the temptation to denote the function $\log(R)$. The most obvious property of $\log(\cdot)$ that $\ell(\cdot)$ doesn't possess is $\log(AB) = \log(A) + \log(B)$. This is clear since the geometric product is not commutative in general whereas addition is.

2.4 Form of $\exp(B)$ *in Euclidean space*

Lemma 1. *If B is of the form $B = \phi P + tn$ where $t \in \mathbb{R}^n$, ϕ is some scalar and P is a 2-blade where $P^2 = -1$ then, for any $k \in \mathbb{Z}^+$,*

$$B^k = \phi^k P^k + \alpha_k^{(1)} \phi P t n + \alpha_k^{(2)} \phi^2 P t n P + \alpha_k^{(3)} \phi t n P + \alpha_k^{(4)} t n$$

[1] There also exist dilation rotors but these will not be discussed in this paper.

with the following recurrence relations for $\alpha_k^{(\cdot)}$, $k > 0$

$$\begin{aligned} \alpha_k^{(1)} &= -\phi^2 \alpha_{k-1}^{(2)} & \alpha_k^{(2)} &= \alpha_{k-1}^{(1)} \\ \alpha_k^{(3)} &= \alpha_{k-1}^{(4)} & \alpha_k^{(4)} &= \phi^{k-1} P^{k-1} - \phi^2 \alpha_{k-1}^{(3)} \end{aligned}$$

with $\alpha_0^{(1)} = \alpha_0^{(2)} = \alpha_0^{(3)} = \alpha_0^{(4)} = 0$.

PROOF. Firstly note that the theorem is trivially provable by direct substitution for the cases $k = 0$ and $k = 1$. We thereafter seek a proof by induction.

Assuming the expression for B^{k-1} is correct, we post-multiply by $\phi P + tn$ to obtain

$$\begin{aligned} B^k = &\phi^k P^k + \alpha_{k-1}^{(1)} \phi^2 PtnP + \alpha_{k-1}^{(2)} \phi^3 PtnP^2 + \alpha_{k-1}^{(3)} \phi^2 tnP^2 + \\ &\alpha_{k-1}^{(4)} \phi tnP + \phi^{k-1} P^{k-1} tn + \alpha_{k-1}^{(1)} \phi P (tn)^2 + \alpha_{k-1}^{(2)} \phi^2 PtnPtn + \\ &\alpha_{k-1}^{(3)} \phi tnPtn + \alpha_{k-1}^{(4)} (tn)^2 \end{aligned}$$

Substituting $P^2 = -1$, $(tn)^2 = -tn^2 t = 0$ and noting that $nPt = -Ptn$ leading to $tnPtn = -tPtn^2 = 0$

$$\begin{aligned} B^k = &\phi^k P^k + \alpha_{k-1}^{(1)} \phi^2 PtnP - \alpha_{k-1}^{(2)} \phi^3 Ptn - \alpha_{k-1}^{(3)} \phi^2 tn + \\ &\alpha_{k-1}^{(4)} \phi tnP + \phi^{k-1} P^{k-1} tn \\ = &\phi^k P^k - (\alpha_{k-1}^{(2)} \phi^2) \phi Ptn + \alpha_{k-1}^{(1)} \phi^2 PtnP + \\ &\alpha_{k-1}^{(4)} \phi tnP + (\phi^{k-1} P^{k-1} - \alpha_{k-1}^{(3)} \phi^2) tn \end{aligned}$$

Equating like coefficients we obtain the required recurrence relations. □

Lemma 2. *Assuming the form of B given in lemma 1, for* $k \in \mathbb{Z}^+$,

$$B^{2k} = (-1)^k \phi^{2k} - k(-1)^k \phi^{2k-1} [tnP + Ptn]$$

and

$$B^{2k+1} = (-1)^k \phi^{2k+1} P + k\phi^{2k} (-1)^k [tn - PtnP] + (-1)^k \phi^{2k} tn$$

PROOF. Starting from $\alpha_0^{(\cdot)} = 0$ it is clear that the recurrence relations above imply that $\alpha_k^{(1)} = \alpha_k^{(2)} = 0 \; \forall \, k \geq 0$. Substituting $\alpha_k^{(3)} = \alpha_{k-1}^{(4)}$ it is trivial to show that the relation for $\alpha_k^{(4)}$ is satisfied by

$$\alpha_k^{(4)} = \begin{cases} \frac{k}{2}(\phi P)^{k-1} & k \text{ even,} \\ \frac{k+1}{2}(\phi P)^{k-1} & k \text{ odd.} \end{cases}$$

When substituted into the expression for B^k, we obtain the result stated above. □

Theorem 3. *If B is a bivector of the form given in theorem 1 then, defining $t_{\|}$ as the component of t lying in the plane of P and $t_{\perp} = t - t_{\|}$,*

$$\exp(B) = [\cos(\phi) + \sin(\phi)P]\,[1 + t_{\perp}n] + \operatorname{sinc}(\phi)t_{\|}n$$

PROOF. Consider the power series expansion of $\exp(B)$,

$$\exp(B) = \sum_{k=0}^{\infty} \frac{B^k}{k!} = \sum_{k=0}^{\infty} \left[\frac{B^{2k}}{(2k)!} + \frac{B^{2k+1}}{(2k+1)!}\right]$$

Substituting the expansion for B^{2k} and B^{2k+1} from lemma 2

$$\begin{aligned}\exp(B) = &\sum_{k=0}^{\infty} \left[\frac{(-1)^k\phi^{2k}}{(2k)!} - k\frac{(-1)^k\phi^{2k-1}}{(2k)!}(tnP + Ptn)\right] \\ &+ \sum_{k=0}^{\infty} \left[\frac{(-1)^k\phi^{2k}}{(2k+1)!}(\phi P + tn) + k\frac{(-1)^k\phi^{2k}}{(2k+1)!}(tn - PtnP)\right]\end{aligned}$$

We now substitute the following power-series representations

$$\cos(z) = \textstyle\sum_{k=0}^{\infty} \frac{(-1)^k z^{2k}}{(2k)!} \qquad \operatorname{sinc}(z) = \textstyle\sum_{k=0}^{\infty} \frac{(-1)^k z^{2k}}{(2k+1)!}$$

$$-z\sin(z) = \textstyle\sum_{k=0}^{\infty} 2k\frac{(-1)^k z^{2k}}{(2k)!} \qquad \cos(z) - \operatorname{sinc}(z) = \textstyle\sum_{k=0}^{\infty} 2k\frac{(-1)^k z^{2k}}{(2k+1)!}$$

to obtain

$$\begin{aligned}\exp(B) = &\cos\phi + \sin(\phi)\frac{1}{2}(tnP + Ptn) + \operatorname{sinc}(\phi)(\phi P + tn) \\ &+ \frac{1}{2}\left[\cos(\phi) - \operatorname{sinc}(\phi)\right](tn - PtnP)\end{aligned}$$

By considering parallel and perpendicular components of t with respect to the plane of P is easy to verify that $tnP + Ptn = 2Pt_{\perp}n$ and $PtnP = (t_{\|} - t_{\perp})n$

hence

$$\begin{aligned}\exp(B) &= \cos\phi + \sin(\phi)Pt_\perp n + \operatorname{sinc}(\phi)(\phi P + tn) + [\cos(\phi) - \operatorname{sinc}(\phi)]\, t_\perp n \\ &= \cos(\phi)\,[1 + t_\perp n] + \sin(\phi)P\,[1 + t_\perp n] + \operatorname{sinc}(\phi)t_\| n \\ &= [\cos(\phi) + \sin(\phi)P]\,[1 + t_\perp n] + \operatorname{sinc}(\phi)t_\| n\end{aligned}$$

□

Definition 4. A *twist* is a rotor whose action is to rotate by ψ in the plane of P whilst translating along a vector a perpendicular to the plane of P. It may therefore be defined by the rotor

$$\tau(\psi, P, a) = \left[\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\psi}{2}\right)P\right]\left[1 + \frac{na}{2}\right]$$

where ψ is a scalar, P is a 2-blade normalised such that $P^2 = -1$ and a is some vector satisfying $a \cdot n = a \cdot P = 0$.

Lemma 5. *The exponentiation function may be re-expressed using a twist*

$$\exp\left(\frac{\psi}{2}P + \frac{tn}{2}\right) = \left[1 + \operatorname{sinc}\left(\frac{\psi}{2}\right)\frac{t_\| n}{2}\tilde{\tau}(\psi, P, -t_\perp)\right]\tau(\psi, P, -t_\perp)$$

PROOF. We firstly substitute our definition of a twist into our form for the exponential

$$\exp\left(\frac{\psi}{2}P + \frac{tn}{2}\right) = \tau(\psi, P, -t_\perp) + \operatorname{sinc}\left(\frac{\psi}{2}\right)\frac{t_\| n}{2} \tag{2}$$

noting that, since twists are rotors, $\tau(\cdot)\tilde{\tau}(\cdot) = 1$, it is trivial to verify that the required expression is equivalent to this form of the exponential. □

Lemma 6. *The expression*

$$1 + \operatorname{sinc}\left(\frac{\psi}{2}\right)\frac{t_\| n}{2}\tilde{\tau}(\psi, P, -t_\perp)$$

is a rotor which acts to translate along a vector $t'_\|$ given by

$$t'_\| = -\operatorname{sinc}\left(\frac{\psi}{2}\right)t_\|\left(\cos\left(\frac{\psi}{2}\right) - \sin\left(\frac{\psi}{2}\right)P\right)$$

PROOF. The expression above may be obtained by substituting for the twist in the initial expression and simplifying. It is clearly a vector since multiplying $t_\|$ on the left by P is just a rotation by $\pi/2$ in the plane of P. □

We have now developed the required theorems and tools to discuss the action of the rotor

$$R = \exp\left(\frac{\psi}{2}P + \frac{tn}{2}\right)$$

It translates along a vector $t_\perp$ which is the component of t which does not lie in the plane of P, rotates by ψ in the plane of P and finally translates along $t'_\parallel$ which is given by

$$t'_\parallel = -\mathrm{sinc}\left(\frac{\psi}{2}\right) t_\parallel \left(\cos\left(\frac{\psi}{2}\right) - \sin\left(\frac{\psi}{2}\right) P\right)$$

which is the component of t lying in the plane of P, rotated by $\psi/2$ in that plane.

2.5 Checking $\exp(B)$ *is a rotor*

It is sufficient to check that $\exp(B)$ satisfies the following properties of a rotor R.

$$R\tilde{R} = 1, \quad Rn\tilde{R} = n$$

Theorem 7. *If $R = \exp(B)$ and B is a bivector of the form given in lemma 1 then $R\tilde{R} = 1$.*

PROOF. Consider the twist form of $\exp(B)$ from equation 2

$$R = \exp(B) = \tau(\psi, P, -t_\perp) + \mathrm{sinc}\left(\frac{\psi}{2}\right)\frac{t_\parallel n}{2}$$

and make use of our knowledge that $\tau(\psi, P, -t_\perp)$ is a rotor. Hence,

$$\begin{aligned} R\tilde{R} &= \tau(\psi, P, -t_\perp)\tilde{\tau}(\psi, P, -t_\perp) + \mathrm{sinc}^2\left(\frac{\psi}{2}\right)\frac{t_\parallel n^2 t_\parallel}{4} \\ &\quad + \mathrm{sinc}\left(\frac{\psi}{2}\right)\left[\tau(\psi, P, -t_\perp)nt_\parallel + t_\parallel n\tilde{\tau}(\psi, P, -t_\perp)\right] \\ &= 1 + 0 + \mathrm{sinc}\left(\frac{\psi}{2}\right)\left[T + \tilde{T}\right] \end{aligned}$$

where $T = \tau(\psi, P, -t_\perp)nt_\parallel$.

Looking at the definition of $\tau(\psi, P, -t_\perp)$, it is clear that it has only scalar, bivector and 4-vector components with the bivector components being parallel to P or $t_\perp n$ and the 4-vector components being parallel to $Pt_\perp n$. When post-multiplied by $nt_\parallel$ to form T, the 4-vector component goes to zero (since $n^2 = 0$) as does the bivector component parallel to $t_\perp n$ and so we are left with T having only components parallel to $nt_\parallel$ and $Pnt_\parallel$. We may now express T as

$$T = \alpha nt_\parallel + \beta Pnt_\parallel$$

where α and β are suitably valued scalars. Hence

$$T + \tilde{T} = \alpha \left[nt_\parallel + t_\parallel n\right] + \beta \left[Pnt_\parallel + t_\parallel n\tilde{P}\right] = 0 + \beta n \left[Pt_\parallel - t_\parallel \tilde{P}\right]$$

By considering two basis vectors of P, a and b, such that $P = ab$, $a \cdot b = 0$ and resolving $t_\parallel$ in terms of a and b it is easy to show that $Pt_\parallel - t_\parallel \tilde{P} = 0$ and hence $T + \tilde{T} = 0$ giving the required result. □

Theorem 8. *If $R = \exp(B)$ and B is a bivector of the form given in lemma 1 then $Rn\tilde{R} = n$.*

PROOF. Again using the twist form of R from equation 2 we have

$$\begin{aligned} Rn &= \tau(\psi, P, -t_\perp)n + \operatorname{sinc}\left(\frac{\psi}{2}\right)\frac{t_\parallel n^2}{2} \\ &= \tau(\psi, P, -t_\perp)n + 0 \end{aligned}$$

Defining the rotation rotor $R_{(P,\psi)}$ as

$$R_{(P,\psi)} = \cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\psi}{2}\right)P$$

and substituting for the definition of the twist above gives

$$Rn = R_{(P,\psi)}n$$

Similarly, again using the twist form of R we have

$$\begin{aligned} nR &= n\tau(\psi, P, -t_\perp) + \operatorname{sinc}\left(\frac{\psi}{2}\right)\frac{nt_\parallel n}{2} \\ &= n\tau(\psi, P, -t_\perp) + 0 \\ &= nR_{(P,\psi)}\left(1 + \frac{tn}{2}\right) \\ &= R_{(P,\psi)}n\left(1 + \frac{tn}{2}\right) \\ &= R_{(P,\psi)}n \end{aligned}$$

We now have that $Rn = nR$ and hence, using $R\tilde{R} = 1$ from the previous theorem, $Rn\tilde{R} = nR\tilde{R} = n$.

$\square$

2.6 Method for evaluating $\ell(R)$

We have found a form for $\exp(B)$ given that B is in a particular form. Now we seek a method to take an arbitrary displacement rotor, $R = \exp(B)$ and re-construct the original B. Should there exist a B for all possible R, we will show that our initial assumption that all displacement rotors can be formed from a single exponentiated bivector of special form is valid. We shall term this initial bivector the *generator* rotor (to draw a parallel with Lie algebras).

We can obtain the following identities for $B = (\psi/2)P + tn/2$ by simply considering the grade of each component of the exponential:

$$\begin{aligned}
\langle R\rangle_0 &= \cos\left(\frac{\psi}{2}\right)\\
\langle R\rangle_2 &= \sin\left(\frac{\psi}{2}\right)P + \cos\left(\frac{\psi}{2}\right)t_\perp n + \operatorname{sinc}\left(\frac{\psi}{2}\right)t_\parallel n\\
\langle R\rangle_4 &= \sin\left(\frac{\psi}{2}\right)Pt_\perp n
\end{aligned}$$

It is somewhat straightforward to reconstruct $\psi, t_\perp$ and $t_\parallel$ from these components by partitioning a rotor as above. Once we have a method which gives the generator B for any displacement rotor R we have validated our assumption.

Theorem 9. *The inverse-exponential function $\ell(R)$ is given by*

$$\ell(R) = ab + c_\perp n + c_\parallel n$$

where

$$\begin{aligned}
\|ab\| &= \sqrt{|(ab)^2|} = \cos^{-1}(\langle R\rangle_0)\\
ab &= \frac{(\langle R\rangle_2\, n)\cdot e}{\operatorname{sinc}(\|ab\|)}\\
c_\perp n &= -\frac{ab\,\langle R\rangle_4}{\|ab\|^2\operatorname{sinc}(\|ab\|)}\\
c_\parallel n &= -\frac{ab\,\langle ab\,\langle R\rangle_2\rangle_2}{\|ab\|^2\operatorname{sinc}(\|ab\|)}
\end{aligned}$$

PROOF. It is clear from the above that the form of $\|ab\|$ is correct. We thus proceed to show the remaining equations to be true

$$\begin{aligned}\langle R\rangle_2 &= \cos(\|ab\|)\, c_\perp n + \operatorname{sinc}(\|ab\|) \left[ab + c_\| n\right] \\ \langle R\rangle_2\, n &= \operatorname{sinc}(\|ab\|)\, abn \\ (\langle R\rangle_2\, n)\cdot e &= \operatorname{sinc}(\|ab\|)\, ab\end{aligned}$$

and hence the relation for ab is correct.

$$\begin{aligned}\langle R\rangle_4 &= \operatorname{sinc}(\|ab\|)\, abc_\perp n \\ ab\,\langle R\rangle_4 &= -\,\|ab\|^2 \operatorname{sinc}(\|ab\|)\, c_\perp n\end{aligned}$$

and hence the relation for $c_\perp n$ is correct.

$$\begin{aligned}\langle R\rangle_2 &= \cos(\|ab\|)\, c_\perp n + \operatorname{sinc}(\|ab\|) \left[ab + c_\| n\right] \\ ab\,\langle R\rangle_2 &= \cos(\|ab\|)\, abc_\perp n + \operatorname{sinc}(\|ab\|) \left[abc_\| n - \|ab\|^2\right] \\ \langle ab\,\langle R\rangle_2\rangle_2 &= \operatorname{sinc}(\|ab\|)\, abc_\| n\end{aligned}$$

and hence the relation for $c_\| n$ is correct. □

3 Interpolation via Logarithms

We have shown that any displacement of Euclidean geometry [2] may be mapped smoothly onto a linear sub-space of the bivectors. This immediately suggests applications to smooth interpolation of displacements. Consider a set of poses we wish to interpolate, $\{P_1, P_2, ..., P_n\}$ and a set of rotors which transform some origin pose to these target poses, $\{R_1, R_2, ..., R_n\}$. We may map these rotors onto the set of bivectors $\{\ell(R_1), \ell(R_2), ..., \ell(R_n)\}$ which are simply points in some linear subspace. We may now choose any interpolation of these bivectors which lies in this space and for any bivector on the interpolant, B'_λ, we can compute a pose, $\exp(B'_\lambda)$.

Another interpolation scheme is to have the poses defined by a set of chained rotors so that $\{P_1, P_2, ..., P_n\}$ is represented by

$$\{R_1, \Delta R_1 R_1, \Delta R_2 R_2, ..., \Delta R_n R_n\}$$

[2] Other geometries may be considered with appropriate modification of the rotors (Lasenby et al., 2004).

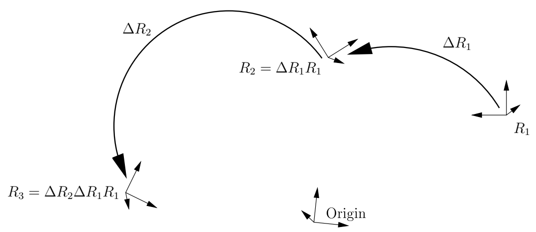

Fig. 4. Rotors used to piece-wise linearly interpolate between key-rotors.

where $R_i = \Delta R_{i-1} R_{i-1}$ as in figure 4. Using this scheme the interpolation between pose R_i and R_{i+1} involves forming the rotor $R_{i,\lambda} = \exp(B_{i,\lambda})R_{i-1}$ where $B_{i,\lambda} = \lambda\ell(\Delta R_{i-1})$ and λ varies between 0 and 1 giving $R_{i,0} = R_{i-1}$ and $R_{i,1} = R_i$.

We now investigate two interpolation schemes which interpolate through target poses, ensuring that each pose is passed through. This kind of interpolation is often required for key-frame animation techniques. The first form of interpolation is piece-wise linear interpolation of the relative rotors (the latter case above). The second is direct quadratic interpolation of the bivectors representing the final poses (the former case).

3.1 Piece-wise linear interpolation

Direct piece-wise linear interpolation of the set of bivectors is one of the simplest interpolation schemes we can consider. Consider the example shown in figure 4. Here there are three rotors to be interpolated. We firstly find a rotor, ΔR_n which takes us from rotor R_n to the next in the interpolation sequence, R_{n+1}.

$$\begin{aligned} R_{n+1} &= (\Delta R_n) R_n \\ \Delta R_n &= R_{n+1} \tilde{R}_n \end{aligned}$$

We then find the bivector, ΔB_n which generates $\Delta R_n = \exp(\Delta B_n)$. Finally we form a rotor interpolating between R_n and R_{n+1}:

$$R_{n,\lambda} = \exp(\lambda \Delta B_n) R_n$$

where λ is in the range $[0, 1]$ and $R_{n,0} = R_n$ and $R_{n,1} = R_{n+1}$. Clearly this interpolation scheme changes abruptly at interpolation points, something which is reflected in the resulting interpolation as shown in figure 5.

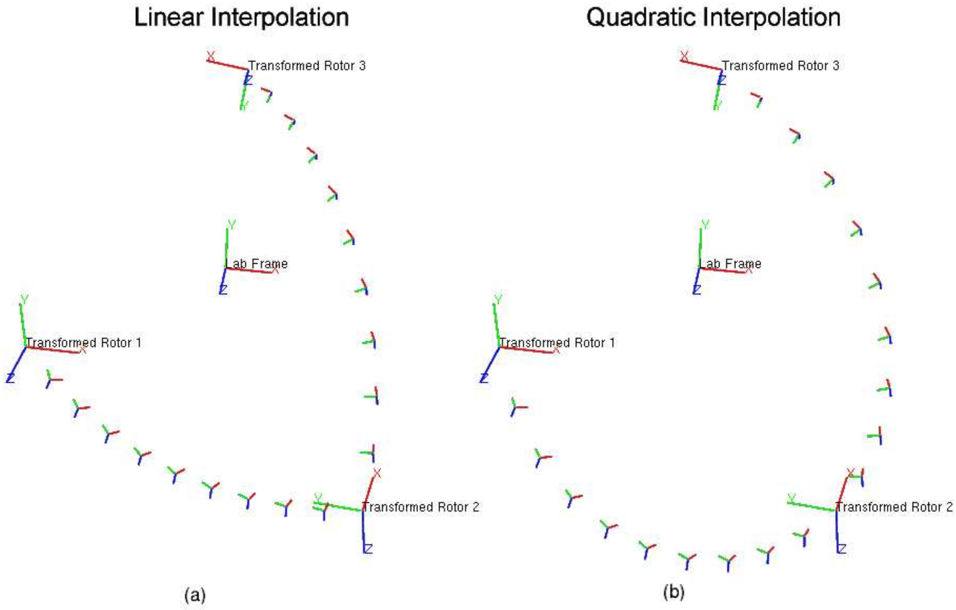

Fig. 5. Examples of a) piece-wise linear and b) quadratic interpolation for three representative poses.

3.2 Quadratic interpolation

Another simple form for interpolation is the quadratic interpolation where a quadratic is fitted through three interpolation points, $\{B_1, B_2, B_3\}$ with an interpolation parameter varying in the range $(-1, +1)$:

$$B'_\lambda = \left(\frac{B_3 + B_1}{2} - B_2\right)\lambda^2 + \frac{B_3 - B_1}{2}\lambda + B_2$$

giving

$$B'_{-1} = B_1, \quad B'_0 = B_2 \text{ and } B'_{+1} = B_3$$

This interpolation varies smoothly through B_2 and is reflected in the final interpolation, as shown in figure 5. Extensions to the quadratic interpolation for more than three interpolation points, such as smoothed quadratic interpolation (Cendes and Wong, 1987) are readily available.

3.3 Alternate methods

It is worth noting that each of the methods described above may be performed using either direct interpolation of the bivector $\ell(R)$ corresponding to a rotor R or by interpolating the relative rotors which take one frame to another. It is not yet clear which will give the best results and indeed it is probably application dependent.

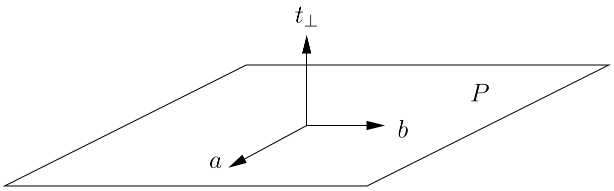

Fig. 6. Orthonormal basis resolved relative to P.

4 Form of the Interpolation

In this section we derive a clearer picture of the precise form of a simple linear interpolation between two frames in order to relate the interpolation to existing methods used in mechanics and robotics. We will consider the method used above whereby the rotor being interpolated takes one pose to another.

4.1 Path of the linear interpolation

Since we have shown that $\exp(B)$ is indeed a rotor, it follows that any Euclidean pure-translation rotor will commute with it. Thus we only need consider the interpolant path when interpolating from the origin to some other point since any other interpolation can be obtained by simply translating the origin to the start point. This location independence of the interpolation is a desirable property in itself but also provides a powerful analysis mechanism.

We have identified in section 2.4 the action of the $\exp(B)$ rotor in terms of $\psi, P, t_\parallel$ and $t_\perp$. We now investigate the resulting interpolant path when interpolating from the origin. We shall consider the interpolant $R_\lambda = \exp(\lambda B)$ where λ is the interpolation co-ordinate and varies from 0 to 1. For any values of $\psi, P, t_\parallel$ and $t_\perp$,

$$\lambda B = \frac{\lambda\psi}{2}P + \frac{\lambda(t_\perp + t_\parallel)n}{2}$$

from which we see that the action of $\exp(\lambda B)$ is a translation along $\lambda t_\perp$, a rotation by $\lambda\psi$ in the plane of P and finally a translation along

$$t'_\parallel = -\mathrm{sinc}\left(\frac{\lambda\psi}{2}\right)\lambda t_\parallel\left(\cos\left(\frac{\lambda\psi}{2}\right) - \sin\left(\frac{\lambda\psi}{2}\right)P\right)$$

We firstly resolve a three dimensional, orthonormal, basis relative to P as shown in figure 6. Here a and b are orthonormal vectors in the plane of P and hence $P = ab$. We may now express $t_\parallel$ as $t_\parallel = t^a a + t^b b$ where $t^{\{a,b\}}$ are suitably valued scalars.

The initial action of $\exp(B)$ upon a frame centred at the origin is therefore to

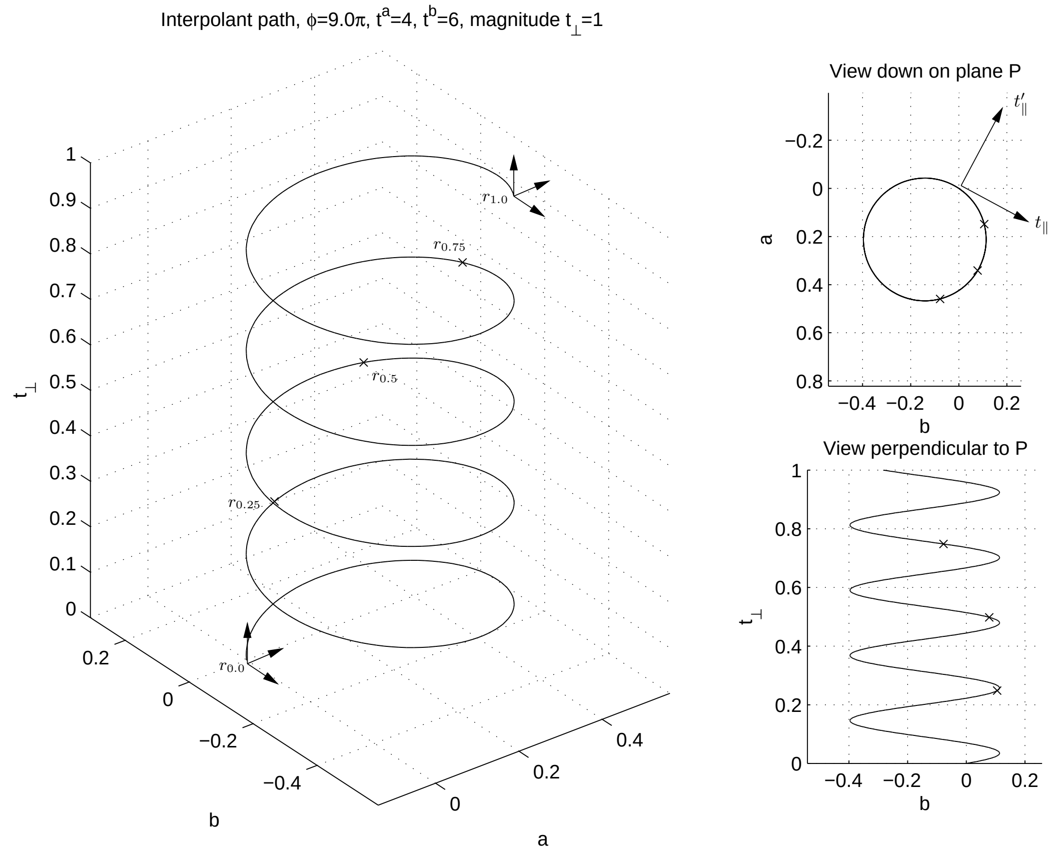

Fig. 7. Example of an interpolant path with the final location being given by $t_{\parallel} = 4a + 6b$, $\psi = 9\pi$ and $t_{\perp}$ having a magnitude of 1.

translate it to $\lambda t_{\perp}$ followed by a rotation in the plane of P. Due to our choice of starting point, this has no effect on the frame's location (but will have an effect on the pose, see the next section).

Finally there is a translation along $t'_{\parallel}$ which, using $c = \cos\left(\frac{\lambda\psi}{2}\right)$ and $s = \sin\left(\frac{\lambda\psi}{2}\right)$, can be expressed in terms of a and b as

$$\begin{aligned} t'_{\parallel} &= -\frac{2s}{\lambda\psi}\lambda(t^a a + t^b b)(c - sab) \\ &= -\frac{2s}{\psi}c(t^a a + t^b b) + s(t^b a - t^a b) \\ &\equiv -\frac{2s}{\psi}a(t^a c + t^b s) + b(t^b c - t^a s) \end{aligned}$$

The position, r_λ, of the frame at λ along the interpolation is therefore

$$r_\lambda = -\frac{2s}{\psi}(a(t^a c + t^b s) + b(t^b c - t^a s)) + \lambda t_{\perp}$$

which can easily be transformed via the harmonic addition theorem to

$$r_\lambda = -\frac{2s}{\psi}\alpha\left[a\cos\left(\frac{\lambda\psi}{2}+\beta_1\right)+b\cos\left(\frac{\lambda\psi}{2}+\beta_2\right)\right]+\lambda t_\perp$$

where $\alpha^2 = (t^a)^2 + (t^b)^2$, $\tan\beta_1 = -\frac{t^b}{t^a}$ and $\tan\beta_2 = -\frac{-t^a}{t^b}$. It is easy, via geometric construction or otherwise, to verify that this implies that $\beta_2 = \beta_1 + \frac{\pi}{2}$. Hence $\cos(\theta+\beta_2) = -\sin(\theta+\beta_1)$. We can now express the frame's position as

$$r_\lambda = -\frac{2\alpha}{\psi}\left[a\sin\left(\frac{\lambda\psi}{2}\right)\cos\left(\frac{\lambda\psi}{2}+\beta_1\right)-b\sin\left(\frac{\lambda\psi}{2}\right)\sin\left(\frac{\lambda\psi}{2}+\beta_1\right)\right]+\lambda t_\perp$$

which can be re-arranged to give

$$\begin{aligned}r_\lambda &= -\frac{\alpha}{\psi}\left[a\left(\sin\left(\lambda\psi+\beta_1\right)-\sin\beta_1\right)+b\left(\cos\left(\lambda\psi+\beta_1\right)-\cos\beta_1\right)\right]+\lambda t_\perp\\ &= -\frac{\alpha}{\psi}\left[a\sin\left(\lambda\psi+\beta_1\right)+b\cos\left(\lambda\psi+\beta_1\right)\right]+\frac{\alpha}{\psi}\left[a\sin\beta_1+b\cos\beta_1\right]+\lambda t_\perp\end{aligned}$$

noting that in the case $\psi \to 0$, the expression becomes $r_\lambda = \lambda t_\perp$ as one would expect. Since a and b are defined to be orthonormal, the path is clearly some cylindrical helix with the axis of rotation passing through $\alpha/\psi\left[a\sin\beta_1+b\cos\beta_1\right]$. An illustrative example, with a and b having unit magnitude, is shown in figure 7. It also clearly shows the relation between the direction of vector $t_\|$ and the final translation within the plane of P, $t'_\|$.

It is worth noting a related result in screw theory, Chasles' theorem, which states that a general displacement may be represented using a screw motion (cylindrical helix) such as we have derived. Screw theory is widely used in mechanics and robotics and the fact that the naïve linear interpolation generated by this method is indeed a screw motion suggests that applications of this interpolation method may be wide-ranging, especially since this method allows many other forms of interpolation, such as Bézier curves or three-point quadratic to be performed with equal ease. Also the pure rotation interpolation given by this method reduces exactly to the quaternionic or Lie group interpolation result allowing this method to easily extend existing ones based upon these interpolations.

4.2 Pose of the linear interpolation

The pose of the transformed frame is unaffected by pure translation and hence the initial translation by $\lambda t_\perp$ has no effect. The rotation by $\lambda\psi$ in the plane, however, now becomes important. The subsequent translation along $t'_\|$ also has no effect on the pose. We find, therefore, that the pose change λ along the interpolant is just the rotation rotor $R_{\lambda\psi,P}$.

5 Conclusions

This paper has shown how the application of Conformal Geometric Algebra to the problem of general displacement interpolation has provided a natural, intuitive method allowing for the application of traditional interpolation schemes to this area. The final displacement interpolation reflects the properties of the interpolation scheme used in the bivector sub-space, such as continuity, smoothness, etc.

A general method for evaluating rotor logarithms and bivector exponentiations has been developed and an analytical investigation of piece-wise linear interpolation of displacements has been performed. This investigation reveals that this simple scheme gives rise to cylindrical spiral shaped interpolants which have a pleasing, natural appearance. Crucially this approach is not limited to three dimensions and may readily be extended to form some interpolation within a higher parameter space when solving problems in Engineering.

A future item for investigation is the non-commutative nature of the exponentiation function, i.e. that $\exp(A)\exp(B) \neq \exp(B)\exp(A)$ generally. This implies that the piece-wise linear interpolation scheme above, where the rotors which moved from control-frame to control-frame were interpolated will generally be different to direct interpolation of the bivectors associated with the control-frames. An analytical description of the possible difference should be found.

References

Cendes, Z. J., Wong, S. H., Nov 1987. C1 quadratic interpolation over arbitrary point sets. IEEE Computer Graphics and Applications, 8–16.

Hestenes, D., Sobczyk, G., 1984. Clifford Algebra to Geometric Calculus: A unified language for mathematics and physics. Reidel.

Lasenby, A., 2003. Modeling the Cosmos: The Shape of the Universe, Keynote Address. SIGGRAPH, San Diego.

Lasenby, J., Bayro-Corrochano, E., 1998. Computing Invariants in Computer Vision using Geometric Algebra. Tech. Rep. CUED/F-INFENG/TR-244, Cambridge University Engineering Department.

Lasenby, J., Lasenby, A. N., Wareham, R. J., 2004. A Covariant Approach to Geometry using Geometric Algebra. Tech. Rep. CUED/F-INFENG/TR-483, Cambridge University Engineering Department.

Lillholm, M., Dam, E., Koch, M., July 1998. Quaternions, interpolation and animation. Tech. Rep. DIKU-TR-98/5, University of Copenhagen.

Rockwood, A., Hestenes, D., Doran, C., Lasenby, J., Dorst, L., Mann, S., 2001. Geometric Algebra, Course Notes. Course 31, SIGGRAPH, Los Angeles.