Quaternion Frame Approach to Streamline Visualization

Andrew J. Hanson, *Member, IEEE Computer Society,* and Hui Ma

Abstract—**Curves in space are difficult to perceive and analyze, especially when they form dense sets as in typical 3D flow and volume deformation applications. We propose a technique that exposes essential properties of space curves by attaching an appropriate moving coordinate frame to each point, reexpressing that moving frame as a unit quaternion, and supporting interaction with the resulting quaternion field. The original curves in three-space are associated with piecewise continuous four-vector quaternion fields, which map into new curves lying in the unit three-sphere in four-space. Since four-space clusters of curves with similar moving frames occur independently of the curves' original proximity in three-space, a powerful analysis tool results. We treat two separate moving-frame formalisms, the Frenet frame and the parallel-transport frame, and compare their properties. We describe several flexible approaches for interacting with and exploiting the properties of the four-dimensional quaternion fields.**

Index Terms—**Quaternion, Frenet frame, orientation frame.**

I. Introduction

We introduce techniques and tools for visualizing streamline data that are based on the differential geometry of 3D space curves. Intrinsic properties of space curves give rise to scalar fields over the curves such as the curvature and torsion. A moving coordinate frame on a curve is a tensor field that is equivalent to a quaternion field; either may be understood as the solution to a set of differential equations driven by the intrinsic scalar fields.

Our fundamental thesis is that quaternion frame coordinates are useful for exposing the similarities and differences of sets of streamlines. Good analytic and visual measures for revealing similarities of curve shapes are rare. Because of the existence of a uniform distance measure in the quaternion space that we use, orientation similarities in the evolution of flow fields appear automatically in meaningful spatial groups. Identification of these similarities is useful for applications such as finding repeating patterns and related curve shapes, both on single curves and within large collections of curves. Conversely, if a large set of nearly-identical curves contains a small number of significant curves that differ from their neighbors due to subtle changes in their frame orientations, our method will distinguish them.

We study two distinct moving coordinate frames that may be assigned to curves in three-space. One is the classic Frenet frame, also called the Frenet-Serret frame (see, e.g., [9], [10]), which is defined locally by the tangent, normal and binormal at each point of each curve; the other is the parallel-transport frame (see, e.g., Bishop [4]), which retains the tangent vector, but uses a nonlocal approach borrowed from the parallel-transport methods of differential geometry to compute the frame components in the plane perpendicular to the curve. All such frames can be recast into quaternion frame coordinates.

Orientation spaces and their relationship to quaternions are described in Altmann [2]; an interesting approach to the visualization of the properties of quaternions was recently given by Hart, Francis, and Kauffman [19]. Systematic approaches for representing clusters of orientations in 3D spaces of angles have been suggested, for example, by Alpern et al. [1]. Among previous approaches to visualizing the geometry of space curves, we note the work of Gray [7], [10] which exploits the curvature and torsion scalar fields on a curve for visualization purposes; this method extends naturally to higher-dimensional manifolds with well-defined local curvature. We will give some examples of the application of curvature and torsion fields for completeness here, but will not pursue this approach in detail.

The use of quaternion frames in a 4D display was proposed as a visualization technique for stream manifold characteristics in Hanson and Ma [17]. The current article is based on the concepts of the latter work, and includes additional results on the comparative properties of the Frenet and parallel-transport frames, as well as further work on interactive methods.

II. The Differential Geometry of Space Curves

Dense families of space curves can be generated by many applications, ranging from time-dependent particle flow fields, to static streamlines generated by integrating a volume vector field, to deformations of a solid coordinate grid. Our fundamental approach singles out space curves, although variations could be used to treat individual point frames (see [1]), stream surfaces (see [20]), and orientation differences (which are themselves orientation fields) as well. Thus, we begin with the properties of a curve $\vec{\mathbf{x}}(t)$ in 3D space parameterized by the unnormalized arc length t. If $\vec{\mathbf{x}}(t)$ is once-differentiable, then the tangent vector at any point is

$$\vec{\mathbf{T}}(t) = \frac{\vec{\mathbf{x}}'(t)}{\left\|\vec{\mathbf{x}}'(t)\right\|}.$$

The standard arc-length differential is typically expressed as

$$v(t)^2 = \left(\frac{ds}{dt}\right)^2 = \left(\frac{d\vec{\mathbf{x}}(t)}{dt}\cdot\frac{d\vec{\mathbf{x}}(t)}{dt}\right) = \left\|\vec{\mathbf{x}}'(t)\right\|^2.$$

A.J. Hanson and H. Ma are with the Department of Computer Science, Indiana University, Bloomington, IN 47405; e-mail: hanson@cs.indiana.edu.

IEEECS Log Number V95009.

In practice, we never have smooth curves in numerical applications, but only piecewise linear curves that are presumed to be approximations to differentiable curves; thus we might typically take, for a curve given by the set of points $\{\vec{\mathbf{x}}_i\}$,

$$\vec{\mathbf{T}}_i = \frac{\vec{\mathbf{x}}_{i+1} - \vec{\mathbf{x}}_i}{\left\|\vec{\mathbf{x}}_{i+1} - \vec{\mathbf{x}}_i\right\|},$$

or any corresponding formula with additional sampling points and desirable symmetries. We use a five-point formula to get a smoother result; one could also produce finer intermediate states by spline interpolation.

If the curve is locally straight, i.e., $\vec{\mathbf{x}}''(t) = 0$ or $\vec{\mathbf{T}}_{i+1} = \vec{\mathbf{T}}_i$, then there is no locally-determinable coordinate frame component in the plane normal to $\vec{\mathbf{T}}$; a nonlocal definition must be used to decide on the remainder of the frame once $\vec{\mathbf{T}}$ is determined. Below, we formulate our two alternate coordinate frames, one of which, the Frenet frame, is completely local, but is indeterminable where the curve is locally straight, and the other of which, the parallel transport frame, is defined everywhere but depends on a numerical integration over the whole curve.

A. Frenet Frames

The *Frenet frame* (see, e.g., [9], [10]) is defined as follows: If $\vec{\mathbf{x}}(t)$ is any thrice-differentiable space curve, its tangent, binormal, and normal vectors at a point on the curve are given by

$$\begin{aligned}
\vec{\mathbf{T}}(t) &= \frac{\vec{\mathbf{x}}'(t)}{\left\|\vec{\mathbf{x}}'(t)\right\|} \\
\vec{\mathbf{B}}(t) &= \frac{\vec{\mathbf{x}}'(t) \times \vec{\mathbf{x}}''(t)}{\left\|\vec{\mathbf{x}}'(t) \times \vec{\mathbf{x}}''(t)\right\|} \\
\vec{\mathbf{N}}(t) &= \vec{\mathbf{B}}(t) \times \vec{\mathbf{T}}(t).
\end{aligned} \tag{1}$$

We illustrate this standard frame configuration in Fig. 1. When the second derivative vanishes on some interval, the Frenet frame is temporarily undefined, as illustrated in Fig. 2. Attempts to work around this problem involve various heuristics (see, e.g., [5], [24]).

The Frenet frame obeys the following differential equation in the parameter t,

$$\begin{bmatrix} \vec{\mathbf{T}}'(t) \\ \vec{\mathbf{N}}'(t) \\ \vec{\mathbf{B}}'(t) \end{bmatrix} = v(t) \begin{bmatrix} 0 & \kappa(t) & 0 \\ -\kappa(t) & 0 & \tau(t) \\ 0 & -\tau(t) & 0 \end{bmatrix} \begin{bmatrix} \vec{\mathbf{T}}(t) \\ \vec{\mathbf{N}}(t) \\ \vec{\mathbf{B}}(t) \end{bmatrix} \tag{2}$$

where $v(t) = \|\mathbf{x}'(t)\|$ is the scalar magnitude of the curve derivative, $\kappa(t)$ is the scalar curvature, and $\tau(t)$ is the torsion. These quantities can in principle be calculated in terms of the parameterized or numerical local values of $\vec{\mathbf{x}}(t)$ and its first three derivatives as follows:

$$\begin{aligned}
\kappa(t) &= \frac{\left\|\vec{\mathbf{x}}'(t) \times \vec{\mathbf{x}}''(t)\right\|}{\left\|\vec{\mathbf{x}}'(t)\right\|^3} \\
\tau(t) &= \frac{\left(\vec{\mathbf{x}}'(t) \times \vec{\mathbf{x}}''(t)\right) \cdot \vec{\mathbf{x}}'''(t)}{\left\|\vec{\mathbf{x}}'(t) \times \vec{\mathbf{x}}''(t)\right\|^2}.
\end{aligned} \tag{3}$$

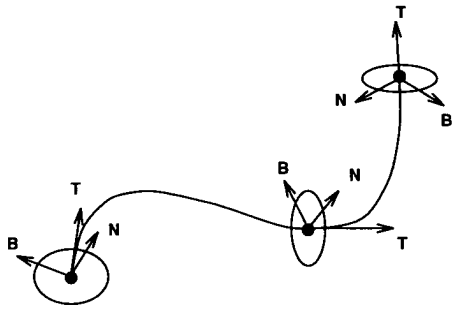


Fig. 1. The triad of orthogonal axes forming the Frenet frame for a curve with nonvanishing curvature.

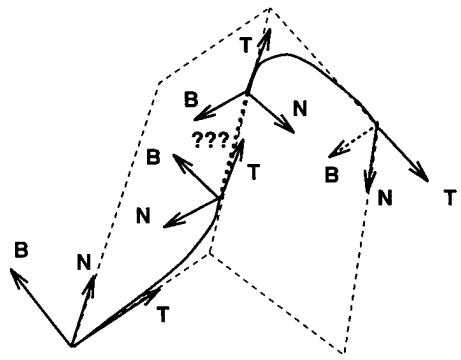


Fig. 2. The triad of orthogonal axes forming the Frenet frame for a curve with vanishing curvature on an interval; the frame is undefined on the interval.

If we are given a nonvanishing curvature and a torsion as smooth functions of t, we can theoretically integrate the system of equations to find the unique numerical values of the corresponding space curve $\vec{\mathbf{x}}(t)$ (up to a rigid motion).

B. Parallel Transport Frames

Bishop [4] noted that, while the Frenet frame has the advantage of consistent local computability at all points on a curve except those with vanishing second derivative, there is another natural frame, the *parallel transport frame*, that is well-defined everywhere; the distinguishing feature of the parallel transport frame is that it is essentially the solution to a differential equation, and thus depends on the initial conditions and is subject to numerical error for long curves. Operational methods of defining such frames have been noted (see, e.g., [5], [24]) but the underlying mathematical basis was not elaborated.

Geometrically, the parallel transport frame derives its name from the fact that it corresponds to the notion of moving a vector around a curved manifold in such a way that it remains as parallel to itself as possible. Its mathematical properties follow from the observation that, while $\vec{\mathbf{T}}(t)$ for a given curve model is unique, we may choose any convenient arbitrary basis $(\vec{\mathbf{N}}_1(t), \vec{\mathbf{N}}_2(t))$ for the remainder of the frame, so long as it is in the plane perpendicular to $\vec{\mathbf{T}}(t)$ at each point. If the derivatives of $(\vec{\mathbf{N}}_1(t), \vec{\mathbf{N}}_2(t))$ depend only on $\vec{\mathbf{T}}(t)$ and not each other, we can make $\vec{\mathbf{N}}_1(t)$ and $\vec{\mathbf{N}}_2(t)$ vary smoothly throughout the path regardless of the curvature. We may, therefore, choose the alternative frame equations

$$\begin{bmatrix} \vec{\mathbf{T}}' \\ \vec{\mathbf{N}}_1' \\ \vec{\mathbf{N}}_2 \end{bmatrix} = v \begin{bmatrix} 0 & k_1 & k_2 \\ -k_1 & 0 & 0 \\ -k_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \vec{\mathbf{T}} \\ \vec{\mathbf{N}}_1 \\ \vec{\mathbf{N}}_2 \end{bmatrix}, \tag{4}$$

illustrated in Fig. 3 for a curve with vanishing curvature on a segment. One can show that [4]

$$\kappa(t) = \left((k_1)^2 + (k_2)^2\right)^{1/2}$$

$$\theta(t) = \arctan\left(\frac{k_2}{k_1}\right)$$

$$\tau(t) = \frac{d\theta(t)}{dt},$$

so that k_1 and k_2 effectively correspond to a Cartesian coordinate system for the polar coordinates κ, θ with $\theta = \int \tau(t)dt$. A fundamental ambiguity in the parallel transport frame compared to the Frenet frame thus arises from the arbitrary choice of an integration constant for θ, which disappears from τ due to the differentiation.

A numerical method for computing the parallel transport frame with the desired properties is the following. Given a frame at $\vec{\mathbf{x}}_{i-1}$, compute two neighboring tangents $\vec{\mathbf{T}}_i$ and $\vec{\mathbf{T}}_{i-1}$ and their unit vectors $\hat{\mathbf{T}}_i = \vec{\mathbf{T}}_i / \| \vec{\mathbf{T}}_i \|$; find the angle $\theta = \arccos(\hat{\mathbf{T}}_i \cdot \hat{\mathbf{T}}_{i-1})$ between them and the perpendicular to the plane of the tangents given by $\vec{\mathbf{V}} = (\hat{\mathbf{T}}_{i-1} \times \hat{\mathbf{T}}_i)$; finally, rotate the frame at $\vec{\mathbf{x}}_{i-1}$ by θ about $\hat{\mathbf{V}}$ to get the frame at point $\vec{\mathbf{x}}_i$. Either 3D vector rotation or rotation by quaternion multiplication can be used to effect the rotation.

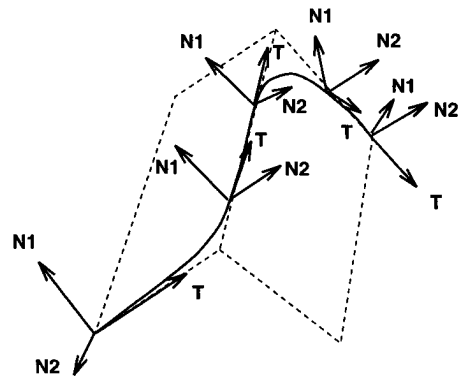


Fig. 3. The parallel-transport curve frame for the curve of Fig. 2 [4]. This frame, unlike the Frenet frame in Fig. 2, is continuous along the "roof peak" where the curvature vanishes.

Just as for the Frenet frame, one can begin with a curve $\vec{\mathbf{x}}(t)$ and an initial frame, or a pair of functions $(k_1(t), k_2(t))$ and an initial frame, or a frame over the entire curve, and then integrate where needed to compute the missing variables. It is also worthwhile noting that $(k_1(t), k_2(t))$ form a two-dimensional Cartesian vector field at each point of the curve, and thus allow a natural alternate characterization to Gray's (κ, τ) curve properties [7], [10].

III. Theory of Quaternion Frames

It is awkward to represent moving frames visually in high-density data because a frame consists of three 3D vectors, or nine components, yet it has only three independent degrees of freedom. Some approaches to representing these degrees of freedom in a three-dimensional space were suggested by Alpern et al. [1]. We propose instead to systematically exploit the representation of 3D orientation frames in four-dimensions using equivalent unit quaternions that correspond, in turn, to points on the three-sphere (see, e.g., [25]). A collection of oriented frames such as those of a crystal lattice can thus be represented by mapping their orientations to a point set in the 4D quaternion space. The moving frame of a 3D space curve can be transformed into a path in quaternion space corresponding pointwise to the 3D space curve.

The quaternion representation of rotations reexpressing a moving frame of a 3D space curve is an elegant unit four-vector field over the curve; the resulting quaternion frames can be displayed as curves in their own right, or can be used in combination with other methods to enrich the display of each 3D curve, e.g., by assigning a coded display color representing a quaternion component.

Properties. A quaternion frame is a unit-length four-vector $q = (q_0, q_1, q_2, q_3) = (q_0, \vec{q})$ that corresponds to exactly one 3D coordinate frame and is characterized by the following properties:

- *Unit Norm.* If we define the inner product of two quaternions as

  $$q \cdot p = q_0 p_0 + q_1 p_1 + q_2 p_2 + q_3 p_3,$$

  then the components of a unit quaternion obey the constraint

  $$q \cdot q = (q_0)^2 + (q_1)^2 + (q_2)^2 + (q_3)^2 = 1, \tag{5}$$

  and therefore lie on S^3, the three-sphere, which we will typically represent as embedded in four-dimensional Euclidean space R^4.
- *Multiplication Rule.* The quaternion product of two quaternions q and p, which we write as $q * p$, takes the form

  $$\begin{bmatrix} [q*p]_0 \\ [q*p]_1 \\ [q*p]_2 \\ [q*p]_3 \end{bmatrix} = \begin{bmatrix} q_0 p_0 - q_1 p_1 - q_2 p_2 - q_3 p_3 \\ q_0 p_1 + q_1 p_0 + q_2 p_3 - q_3 p_2 \\ q_0 p_2 + q_2 p_0 + q_3 p_1 - q_1 p_3 \\ q_0 p_3 + q_3 p_0 + q_1 p_2 - q_2 p_1 \end{bmatrix}$$

  This rule is isomorphic to multiplication in the group SU(2), the double covering of the ordinary 3D rotation group SO(3). If two quaternions a and b are transformed by multiplying them by the same quaternion q, their inner product $a \cdot b$ transforms as

  $$(q * a) \cdot (q * b) = (a \cdot b)(q \cdot q)$$

  and so is invariant if q is a unit quaternion.
- *Mapping to 3D rotations.* Every possible 3D rotation R (a 3×3 orthogonal matrix) can be constructed from either of two related quaternions, $q = (q_0, q_1, q_2, q_3)$ or $-q = (-q_0, -q_1, -q_2, -q_3)$, using the quadratic relationship

$$R = \begin{bmatrix} Q(+--) & D^-(123) & D^+(312) \\ D^+(123) & Q(-+-) & D^-(231) \\ D^-(312) & D^+(231) & Q(--+) \end{bmatrix} \quad (6)$$

where $Q(\pm\pm\pm) = q_0^2 \pm q_1^2 \pm q_2^2 \pm q_3^2$ and $D^{\pm}(ijk) = 2q_iq_j \pm 2q_0q_k$.

- *Rotation Correspondence.* When we substitute $q = (\cos\frac{\theta}{2}, \hat{\mathbf{n}}\sin\frac{\theta}{2})$, into (6), where $\hat{\mathbf{n}}\cdot\hat{\mathbf{n}} = 1$ is a unit three-vector lying on the two-sphere S^2, $R(\theta, \hat{\mathbf{n}})$ becomes the standard matrix for a rotation by θ in the plane perpendicular to $\hat{\mathbf{n}}$; the quadratic form ensures that the two distinct unit quaternions q and $-q$ in S^3 correspond to the *same* SO(3) rotation.

A. Quaternion Frenet Frames

All 3D coordinate frames can be expressed in the form of quaternions using (6). If we assume the columns of (6) are the vectors $((\vec{\mathbf{T}}, \vec{\mathbf{N}}, \vec{\mathbf{B}})$, respectively, one can show from (2) that $[q'(t)]$ takes the form (see [12])

$$\begin{bmatrix} q_0' \\ q_1' \\ q_2' \\ q_3' \end{bmatrix} = \frac{v}{2}\begin{bmatrix} 0 & -\tau & 0 & -\kappa \\ \tau & 0 & \kappa & 0 \\ 0 & -\kappa & 0 & \tau \\ \kappa & 0 & -\tau & 0 \end{bmatrix} \cdot \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}. \quad (7)$$

This equation has the following key properties:

- The matrix on the right hand side is antisymmetric, so that $q(t) \cdot q'(t) = 0$ by construction. Thus, all unit quaternions remain unit quaternions as they evolve by this equation.
- The number of equations has been reduced from nine coupled equations with six orthonormality constraints to four coupled equations incorporating a single constraint that keeps the solution vector confined to the three-sphere.

We verify that the matrices

$$A = \begin{bmatrix} q_0 & q_1 & -q_2 & -q_3 \\ q_3 & q_2 & q_1 & q_0 \\ -q_2 & q_3 & -q_0 & q_1 \end{bmatrix}$$

$$B = \begin{bmatrix} -q_3 & q_2 & q_1 & -q_0 \\ q_0 & -q_1 & q_2 & -q_3 \\ q_1 & q_0 & q_3 & q_2 \end{bmatrix}$$

$$C = \begin{bmatrix} q_2 & q_3 & q_0 & q_1 \\ -q_1 & -q_0 & q_3 & q_2 \\ q_0 & -q_1 & -q_2 & q_3 \end{bmatrix}$$

explicitly reproduce (2),

$$[A]\cdot[q'] = \vec{\mathbf{T}}' = v\kappa\vec{\mathbf{N}}$$
$$[B]\cdot[q'] = \vec{\mathbf{N}}' = -v\kappa\vec{\mathbf{T}} + v\tau\vec{\mathbf{B}}$$
$$[C]\cdot[q'] = \vec{\mathbf{B}}' = -v\tau\vec{\mathbf{N}},$$

where we have applied (7) to get the right-hand terms.

Just as the Frenet equations may be integrated to generate a unique moving frame with its space curve for nonvanishing $\kappa(t)$, we may integrate the much simpler quaternion equations (7).

B. Quaternion Parallel Transport Frames

Similarly, a parallel-transport frame system given by (4) with $(\vec{\mathbf{N}}_1, \vec{\mathbf{T}}, \vec{\mathbf{N}}_2)$ (in that order) corresponding to the columns of (6) is completely equivalent to the following parallel-transport quaternion frame equation for $[q'(t)]$:

$$\begin{bmatrix} q_0' \\ q_1' \\ q_2' \\ q_3' \end{bmatrix} = \frac{v}{2}\begin{bmatrix} 0 & -k_2 & 0 & k_1 \\ k_2 & 0 & -k_1 & 0 \\ 0 & k_1 & 0 & k_2 \\ -k_1 & 0 & -k_2 & 0 \end{bmatrix} \cdot \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (8)$$

where antisymmetry again guarantees that the quaternions remain constrained to the unit three-sphere. The correspondence to (4) is verified as follows:

$$[B]\cdot[q'] = \vec{\mathbf{T}}' = vk_1\vec{\mathbf{N}}_1 + vk_2\vec{\mathbf{N}}_2$$
$$[A]\cdot[q'] = \vec{\mathbf{N}}_1' = -vk_1\vec{\mathbf{T}}$$
$$[C]\cdot[q'] = \vec{\mathbf{N}}_2' = -vk_2\vec{\mathbf{T}}.$$

IV. Assigning Smooth Quaternion Frames

Given a particular curve, we are next faced with the task of assigning quaternion values to whatever moving frame sequence we have chosen.

A. Assigning Quaternions to Frenet Frames

The Frenet frame equations are pathological, for example, when the curve is perfectly straight for some distance or when the curvature vanishes momentarily. Thus, real numerical data for space curves will frequently exhibit behaviors that make the assignment of a smooth Frenet frame difficult, unstable, or impossible. In addition, since any given 3×3 orthogonal matrix corresponds to two quaternions that differ in sign, methods of deriving a quaternion from a Frenet frame are intrinsically ambiguous. Therefore, we prescribe the following procedure for assigning smooth quaternion Frenet frames to points on a space curve:

- Select a numerical approach to computing the tangent $\vec{\mathbf{T}}$ at a given curve point $\vec{\mathbf{x}}$; this typically depends on the chosen curve model and the number of points one wishes to sample.
- Compute the remaining numerical derivatives at a given point and use those to compute the Frenet frame according to (1). If any critical quantities vanish, tag the frame as undefined (or as needing a heuristic fix).
- Check the dot product of the previous binormal $\mathbf{B}(t)$ with the current value; if it is near zero, choose a correction procedure to handle this singular point. Among the correction procedures we have considered are: 1) simply jump discontinuously to the next frame to indicate the presence of a point with very small curvature; 2) create

an interpolating set of points and perform a geodesic interpolation [25]; or 3) deform the curve slightly before and after the singular point to "ease in" with a gradual rotation of the frame or apply an interpolation heuristic (see, e.g., [24]). Creating a jump in the frame assignment is our default choice, since it does not introduce any new information.

- Apply a suitable algorithm such as that of Shoemake [25] to compute a candidate for the quaternion corresponding to the Frenet frame.
- If the 3×3 Frenet frame is smoothly changing, make one last check on the 4D inner product of the quaternion frame with its own previous value: If there is a sign change, choose the opposite sign to keep the quaternion smoothly changing (this will have no effect on the corresponding 3×3 Frenet frame). If this inner product is near zero instead of ± 1, you have detected a radical change in the Frenet frame which should have been noticed in the previous tests.
- If the space curves of the data are too coarsely sampled to give the desired smoothness in the quaternion frames, but are still close enough to give consistent qualitative behavior, one may choose to smooth out the intervening frames using the desired level of recursive slerping [23], [25] to get smoothly splined intermediate quaternion frames.

In Fig. 4, we plot an example of a torus knot, a smooth space curve with everywhere nonzero curvature, together with its associated Frenet frames, its quaternion frame values, and the path of its quaternion frame field projected from four-space. Fig. 5 plots the same information, but this time for a curve with a discontinuous frame that flips too quickly at a zero-curvature point. This space curve has two planar parts drawn as though on separate pages of a partly-open book and meeting smoothly on the "crack" between pages. We see the obvious jump in the Frenet and quaternion frame graphs at the meeting point; if the two curves are joined by a long straight line, the Frenet frame is ambiguous and is essentially undefined in this segment. Rather than invent an interpolation, we generally prefer to use the parallel transport method described next.

B. Assigning Quaternions to Parallel Transport Frames

In order to determine the quaternion frames of an individual curve using the parallel transport method, we follow a similar, but distinct, procedure:

- Select a numerical approach to assigning a tangent at a given curve point as usual.
- Assign an initial reference orientation to each curve in the plane perpendicular to the initial tangent direction. The entire set of frames will be displaced from the origin in quaternion space by the corresponding value of this initial orientation matrix, but the shape of the entire curve will be the same regardless of the initial choice. This choice is intrinsically ambiguous and application dependent. However, one appealing strategy is to base the initial frame on the first well-defined Frenet frame, and then proceed from there using the parallel-transport frame evolution; this guarantees that identical curves have the same parallel-transport frames.

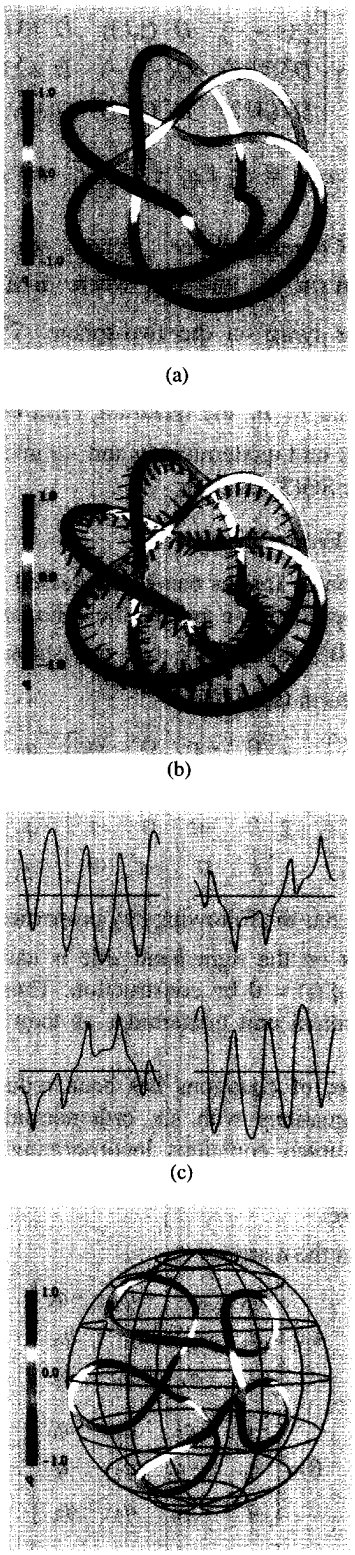


Fig. 4. (a) Projected image of a 3D (3, 5) torus knot. (b) Selected Frenet frame components displayed along the knot. (c) The corresponding smooth quaternion frame components. (d) The path of the quaternion frame components in the three-sphere projected from four-space. Color scales indicate the 0th component of the curve's four-vector frame (upper left graph in (c)).

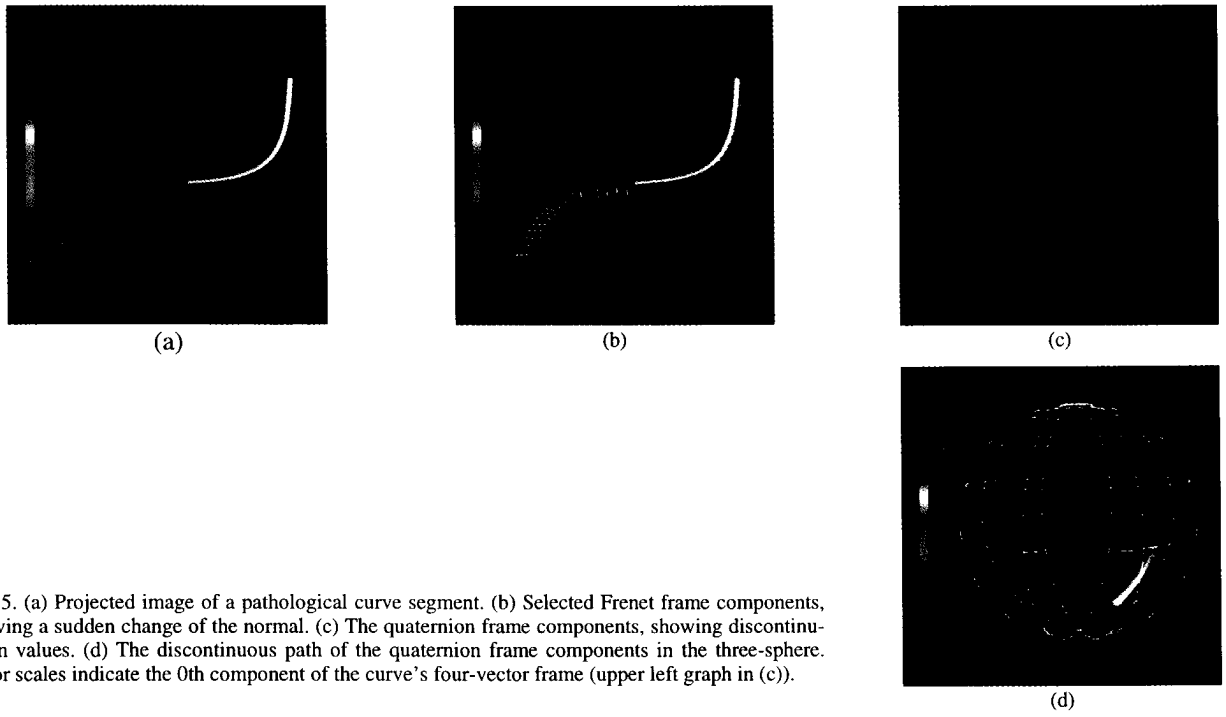


Fig. 5. (a) Projected image of a pathological curve segment. (b) Selected Frenet frame components, showing a sudden change of the normal. (c) The quaternion frame components, showing discontinuity in values. (d) The discontinuous path of the quaternion frame components in the three-sphere. Color scales indicate the 0th component of the curve's four-vector frame (upper left graph in (c)).

- Compute the angle between successive tangents, and rotate the frame by this angle in the plane of the two tangents to get the next frame value.
- If the curve is straight, the algorithm automatically makes no changes.
- Compute a candidate quaternion representation for the frame, applying consistency conditions as needed.

Note that the initial reference orientation and all discrete rotations can be represented *directly in terms of quaternions*, and thus quaternion multiplication can be used directly to apply frame rotations. Local consistency is then automatic.

An example is provided in Fig. 6, which shows the parallel transport analog of Fig. 4 for a torus knot. Fig. 7 is the parallel transport analog of the pathological case in Fig. 5, but this time the frame is continuous when the curvature vanishes.

V. Examples

We next present some typical examples of streamline data represented using the basic geometric properties we have described. Each data set is rendered in the following alternative modes: 1) as a 3D Euclidean space picture, pseudocolored by curvature value; 2) as a 3D Euclidean space picture, pseudocolored by torsion value; 3) as a four-vector quaternion Frenet frame field plotted in the three-sphere; and 4) as a four-vector quaternion parallel transport frame field plotted in the three-sphere.

- *Fig.* 8. A complicated set of streamlines derived from twisting a solid elastic Euclidean space as part of the process of tying a topological knot.
- *Fig.* 9. An AVS-generated streamline data set; the flow is obstructed somewhere in the center, causing sudden jumps of the streamlines in certain regions.

While our focus in this paper is specifically on the frames of space curves, we remark that collections of frames of isolated points, frames on stream surfaces [20], and volumetric frame fields could also be represented using a similar mapping into quaternion space.

VI. Visualization Methods

Once we have calculated the quaternion frames, the curvature, and the torsion for a point on the curve, we have a family of tensor and scalar quantities that we may exploit to expose the intrinsic properties of a single curve. Furthermore, and probably of greater interest, we also have the ability to make visual comparisons of the similarities and differences among families of neighboring space curves.

The moving frame field of a set of streamlines is potentially a rich source of detailed information about the data. However, the nine-component frame is unsuitable for direct superposition on dense data due to the high clutter resulting when its three orthogonal three-vectors are displayed; direct use of the frame is only practical at very sparse intervals, which prevents the viewer from grasping important structural details and changes at a glance. Displays based on 3D angular coordinates are potentially useful, but lack metric uniformity [1].

The four-vector quaternion frame is potentially a more informative and flexible basis for frame visualizations; below, we discuss several alternative approaches to the exploitation of quaternion frames for data consisting of families of smooth curves.

A. Direct Three-Sphere Plot of Quaternion Frame Fields

We now repeat the crucial observation: *For each 3D space curve, the moving quaternion frames define completely new 4D space curves lying on the unit three-sphere embedded in 4D Euclidean space.*

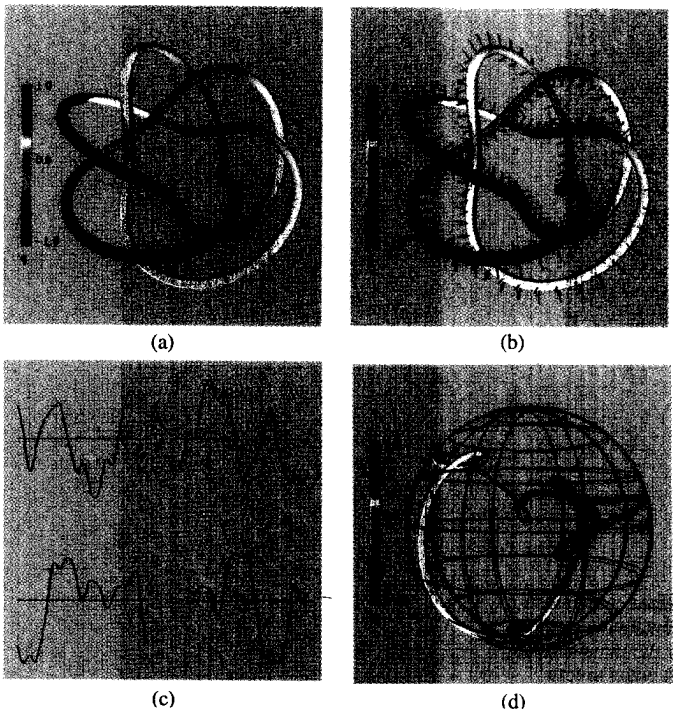


Fig. 6. (a) Projected image of a 3D (3, 5) torus knot. (b) Selected parallel-transport frame components displayed along knot. (c) The corresponding smooth quaternion frame components. (d) The path of the quaternion frame components in the three-sphere projected from four-space. Color scales indicate the 0th component of the curve's four-vector frame (upper left graph in (c)).

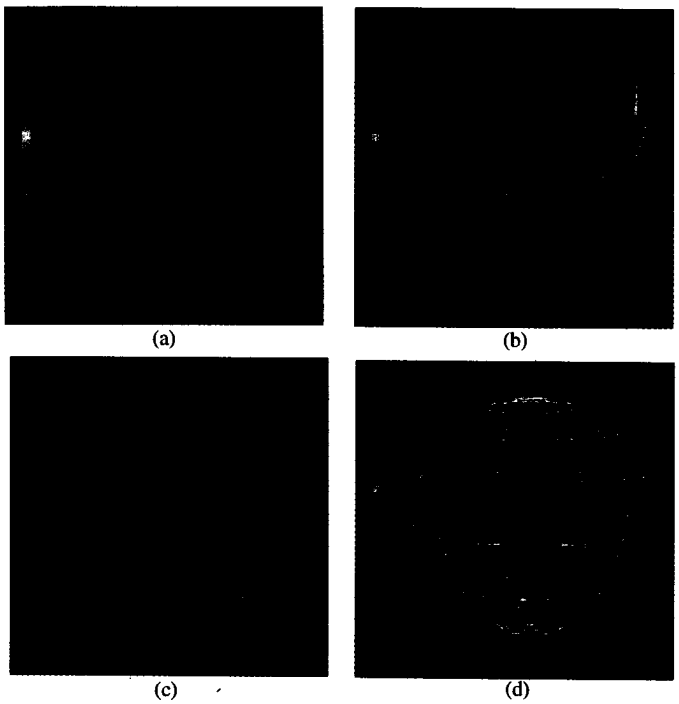


Fig. 7. (a) Projected image of a pathological curve segment. (b) Selected parallel transport frame components, showing smooth change of the normal. (c) The quaternion frame components, showing continuity in values. (d) The continuous path of the quaternion frame components in the three-sphere. Color scales indicate the 0th component of the curve's four-vector frame (upper left graph in (c)).

These curves can have *entirely different geometry* from the original space curve, since distinct points on the curve correspond to distinct orientations. Families of space curves with exactly the same shape will map to the *same* quaternion curve, while curves that fall away from their neighbors will stand out distinctly in the three-sphere plot. Regions of vanishing curvature will show up as discontinuous gaps in the otherwise continuous quaternion Frenet frame field curves, but will be well-behaved in the quaternion parallel transport frame fields. Straight 3D lines will of course map to single points in quaternion space, which may require special attention in the display.

Figs. 4d and 5d present elementary examples of the three-sphere plot for the Frenet frame, while Figs. 6d and 7d illustrate the parallel transport frame. Figs. 8c, 8d, 9c, and 9d present more realistic examples.

The quaternion frame curves displayed in these plots are 2D projections of two overlaid 3D solid balls corresponding to the "front" and "back" hemispheres of S^3. The three-sphere is projected from 4D to 3D along the 0th axis, so the "front" ball has points with $0 \le q_0 \le +1$, and the "back" ball has points with $-1 \le q_0 < 0$. The q_0 values of the frame at each point can be displayed as shades of gray or pseudocolor. In the default view projected along the q_0-axis, points that are projected from 4D to the 3D origin are in fact identity frames, since unit length of q requires $q = (\pm 1, 0, 0, 0)$ at these points. In Fig. 10, we show a sequence of views of the same quaternion curves from different 4D viewpoints using parallel projection; Fig. 11 shows the additional contrast in structure sizes resulting from a 4D perspective projection.

B. Scalar Geometric Fields

Gray [7], [10] has advocated the use of curvature and torsion-based color mapping to emphasize the geometric properties of single curves such as the torus knot. Since this information is trivial to obtain simultaneously with the Frenet frame, we also offer the alternative of encoding the curvature and torsion as scalar fields on a volumetric space populated either sparsely or densely with streamlines; examples are shown in Figs. 8a, 8b, 9a, and 9b.

C. Similarity Measures for Quaternion Frames

Quaternion frames carry with them a natural geometry that may be exploited to compute meaningful similarity measures. Rather than use the Euclidean distance in four-dimensional Euclidean space R^4, one may use the magnitude of the four-vector scalar product of unit quaternions

$$d(q, p) = |q \cdot p| = |\, q_0 p_0 + q_1 p_1 + q_2 p_2 + q_3 p_3 \,|,$$

or the corresponding angle,

$$\theta(q, p) = \arccos\,(d(q, p)),$$

which is the angular difference between the two 4D unit vectors and a natural measure of great-circle arc-length on S^3. Choosing this as a distance measure results in a quantity that is invariant under 4D rotations, invariant under 3D rotations represented by quaternion multiplication, and is also insensitive to the sign ambiguity in the quaternion representation for a given frame. Thus it may be used as a quantitative measure of the similarity of any two 3D frames. This is a natural way to compare either successive frames on a single streamline or pairs of frames on different streamlines.

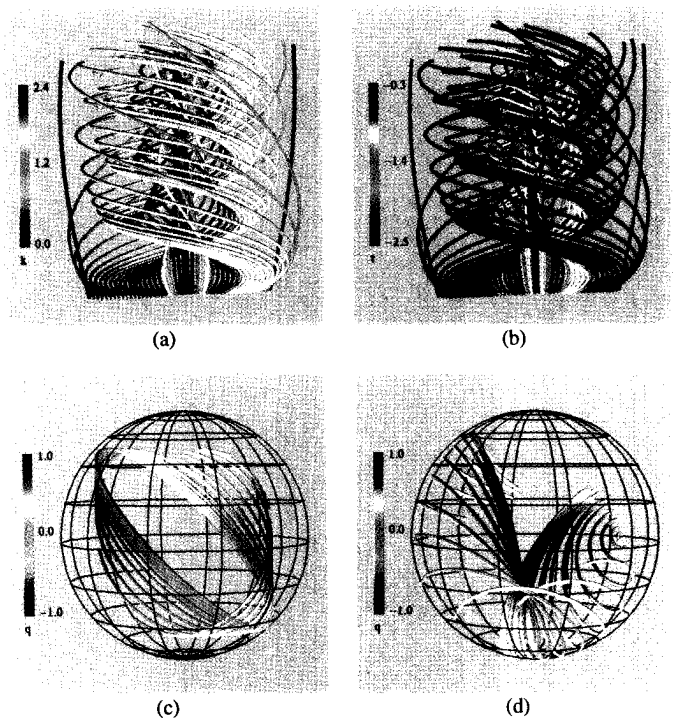


Fig. 8. (a) Deformed volume related to tying a knot, color coded by curvature. (b) Deformed volume related to tying a knot, color coded by torsion. (c) The corresponding quaternion field paths for the Frenet frames. (d) The corresponding quaternion field paths for the parallel transport frames. The color code is keyed to the value of the quaternion component q_0 that is collapsed in the projection from 4D to 3D.

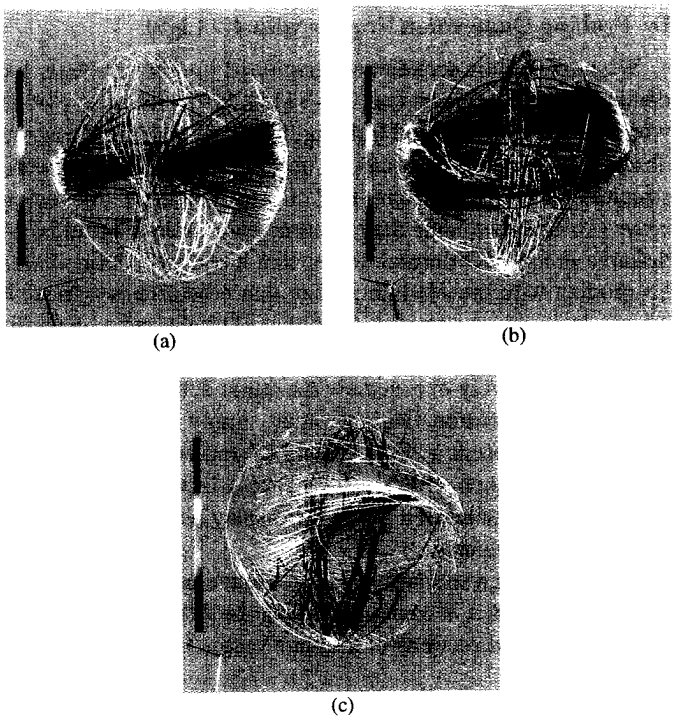


Fig.10. Successive frames in a 4D rotation of the parallel projected three-sphere display of the quaternion fields for a set of streamline data.

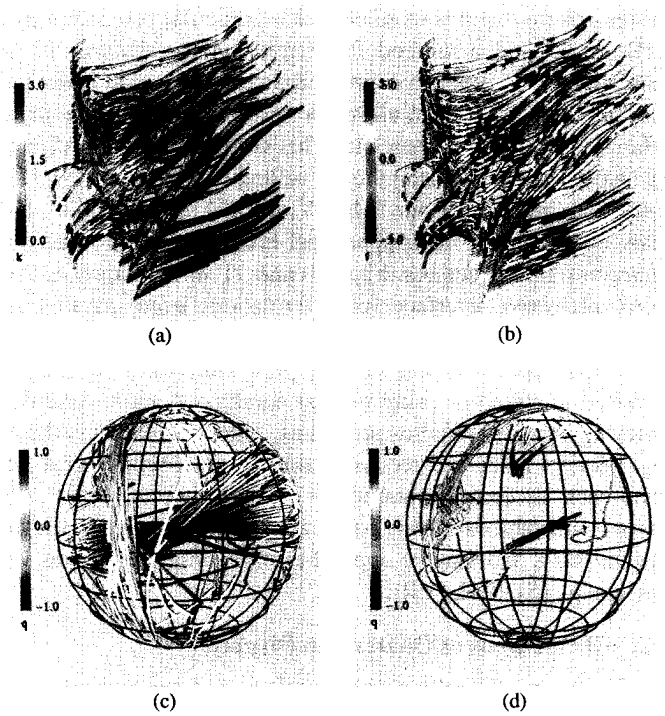


Fig. 9. (a) Vector field streamlines, color coded by curvature. (b) Vector field streamlines, color coded by torsion. (c) The corresponding quaternion field paths for the Frenet frames. (d) The corresponding quaternion field paths for the parallel transport frames. The color code is keyed to the value of the quaternion component q_0 that is collapsed in the projection from 4D to 3D. (AVS demonstration data set.)

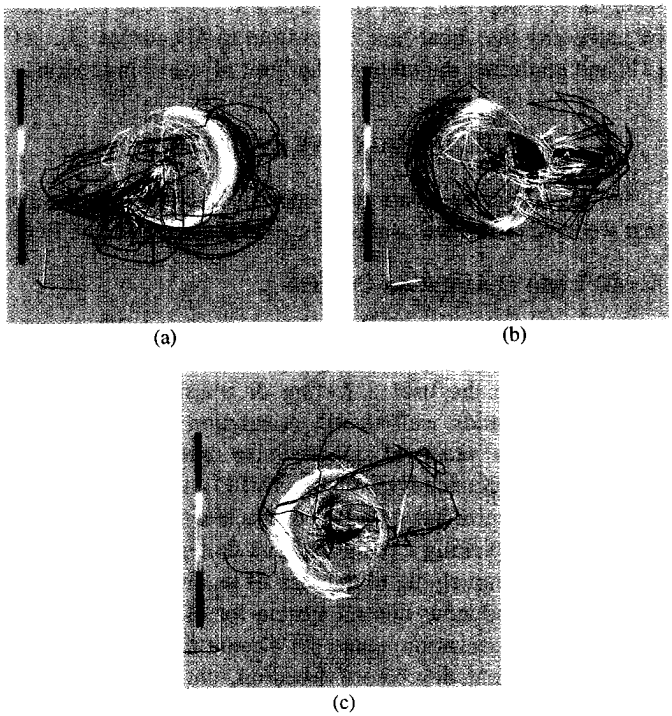


Fig. 11. Successive frames in a 4D rotation of the polar projected three-sphere display of the quaternion fields for a set of streamline data.

D. Probing Quaternion Frames with 4D Light

We next explore techniques developed in other contexts for dealing with 4D objects (see [14], [15], [16]). In our previous work on 4D geometry and lighting, the critical element was the observation that 4D light can be used as a probe of geometric structure provided we can find a way (such as thickening curves or surfaces until they become true three-manifolds) to define a unique 4D normal vector that has a well-defined scalar product with the 4D light; when that objective is achieved, we can interactively employ a moving 4D light and a generalization of the standard illumination equations to produce images that selectively expose new structural details.

Given a quaternion field, we may simply select a 4D unit vector L to represent a "light direction" and employ a standard lighting model such as $I(t) = L \cdot q(t)$ to select individual components of the quaternion fields for display using pseudo-color coding for the intensity.

Fig. 12 shows a streamline data set rendered by computing a pseudo-color index at each point using the 4D lighting formula and varying the directions of the four-vector L.

E. True 4D Illumination

The quaternion curves in 4D may also be displayed in an entirely different mode by thickening them to form three-manifolds using the method of Hanson and Heng [15], [16] and replacing $q(t)$ in the 4D lighting formula and its specular analogs by the 4D normal vector for each volume element or vertex. The massive expense of volume rendering the resulting solid tubes comprising the 4D projection to 3D can be avoided by extending the "bear-hair" algorithm to 4D curves [3], [14], [21] and rendering the tubes in the limit of vanishing radius.

VII. Interactive Interfaces

We next describe a variety of specific interactive techniques that we have examined as tools for exploring quaternion fields.

A. 4D Light Orientation Control

Direct manipulation of 3D orientation using a 2D mouse is typically handled using a rolling ball [11] or virtual sphere [6] method to give the user a feeling of physical control. This philosophy extends well to 4D orientation control (see [8], [13]), giving a practical approach to interacting with the visualization approaches of Sections VI.D and VI.E.

A 3D unit vector has only two degrees of freedom, and so is determined by picking a point within a unit circle to determine the direction uniquely up to the sign of its view-direction component. The analogous control system for 4D lighting is based on a similar observation: since the 4D normal vector has only three independent degrees of freedom, choosing an *interior point* in a solid sphere determines the vector uniquely up to the sign of its component in the unseen fourth dimension (the "4D view-direction component").

Fig. 12 shows an example with a series of snapshots of this interactive interface at work. An additional information display shows the components of the 4D light vector at any particular moment.

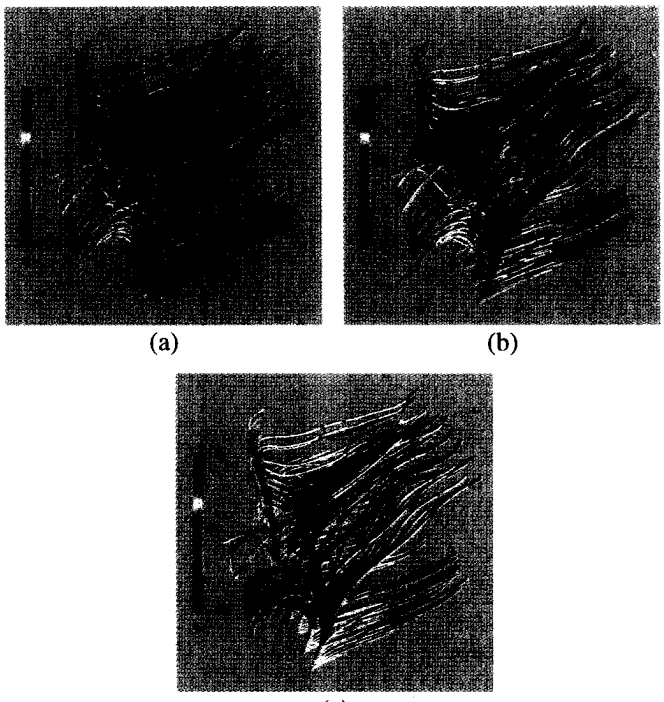


Fig. 12. Color coding a streamline data set using an interactively moving 4D "light" as a probe to isolate similar components of the quaternion fields associated to each point of each curve.

B. 4D Viewing and Three-Sphere Projection Control

Actually displaying quaternion field data mapped to the three-sphere requires us to choose a particular projection from 4D to 3D and a method for displaying the features of the streamlines. In order to expose all possible relevant structures, the user interface must allow the viewer to freely manipulate the 4D projection parameters. This control is easily and inexpensively provided using the 4D rolling ball interface [8], [13]. A special version of our "MeshView" 4D viewing utility [22] has been adapted to support real-time interaction with quaternion frame structures. Figs. 10 and 11 show snapshots from this interactive interface for 4D rotations using parallel and polar 4D projections, respectively.

The simplest viewing strategy plots wide lines that may be viewed in stereo or using motion parallax. A more expensive viewing strategy requires projecting a line or solid from the 4D quaternion space and reconstructing an ideal tube in real time for each projected streamline. The parallel transport techniques introduced in this paper are in fact extremely relevant to this task, and may be applied to the tubing problem as well (see, e.g., [5], [24], [18]).

C. 3D Rotations of Quaternion Displays

Using the 3D rolling ball interface, we can generate quaternion representations of 3D rotations of the form $q = (\cos\frac{\theta}{2}, \hat{\mathbf{n}}\sin\frac{\theta}{2})$, and transform the entire quaternion display by quaternion multiplication, i.e., by changing each point to $p' = q * p$. This effectively displaces the 3D identity frame in quaternion space from (1, 0, 0, 0) to q. This may be useful when trying to compare curves whose properties differ by a

rigid 3D rotation (a common occurrence in the parallel-transport frame due to the arbitrariness of the initial condition).

Other refinements might include selecting and rotating single streamlines in the quaternion field display to make interactive comparisons with other streamlines differing only by rigid rotations. One might also use automated tools to select rotationally similar structures based on minimizing the 4D scalar product between quaternion field points as a measure of similarity.

D. Exploiting or Ignoring Double Points

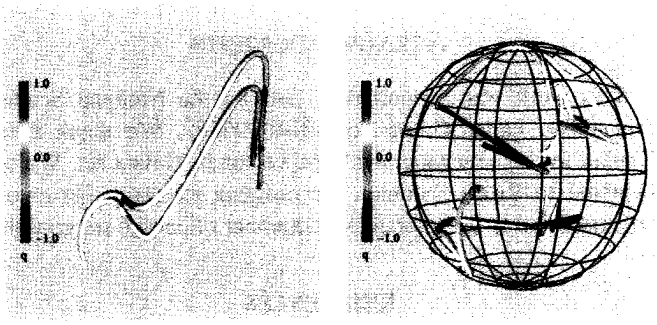
The unique feature of quaternion representations of orientation frames is that they are doubled. If we have a single curve, it does not matter which of the two points in S^3 is chosen as a starting point, since the others follow by continuously integrating small transformations. A collection of points with a uniform orientation as an initial condition similarly will evolve in tandem and normally requires only a single choice to see the pattern.

However, it is possible for a frame to rotate a full 2π radians back to its initial orientation, and be on the opposite side of S^3, or for a collection of streamlines to have a wide range of starting orientations that preclude a locally consistent method for choosing a particular quaternion q over its "neighbor" $-q$. We then have several alternatives:

- Include a reflected copy of every quaternion field in the display. This doubles the data density, but ensures that no two frame fields that are similar will appear diametrically opposite; the metric properties of similar curves will be easy to detect. In addition, 4D rotations will do no damage to the continuity of fields that are rotated to the outer surface and pass from the northern to the southern hyperhemisphere. If 4D depth is depicted by a color code, for example, a point that rotates up to the surface of the displayed solid ball will smoothly pass to the surface and then pass back towards the center while its color changes from positive to negative depth coding.
- Keep only one copy, effectively replacing q by $-q$ if it is not in the default viewing hyperhemisphere. This has the effect that each data point is unique, but that curve frames very near diametrically opposite points on the S^2 surface of the solid ball representing the north hyperhemisphere will be close in orientation but far away in the projection. In addition, when 4D rotations are applied, curves that reach the S^2 surface of the solid ball will jump to the diametrically opposite surface instead of passing smoothly "around" the edge to the southern hyperhemisphere.

E. Reciprocal Similarities and Differences

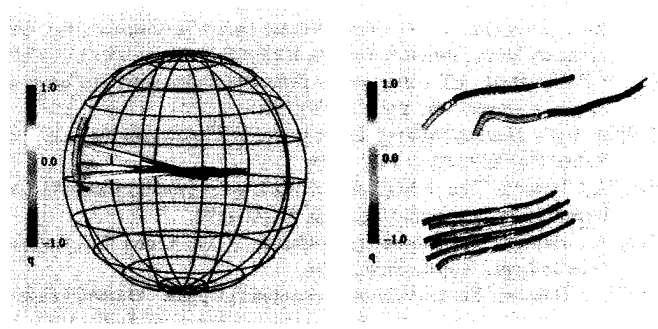
One of the most interesting properties of the quaternion frame method is the appearance of clusters of similar frame fields in the three-sphere display. Two reciprocal tools for exploring these properties immediately suggest themselves. In Fig. 13, we illustrate the effect of grabbing a cluster of streamlines that are spatially close in 3D space and then highlighting their counterparts in the 4D quaternion field space, thus allowing the separate study of their moving frame properties. This technique distinguishes curves that are similar in 3D space but have drastically different frame characteristics.



(a) (b)

Fig. 13. (a) Selecting stream fields that are close in the original 3D data display and (b) echoing them in the 4D quaternion Frenet frame display. The moving frames of these two curves are drastically different even though the curves appear superficially similar in 3D. The unseen component of 4D depth, with a range -1.0 to 1.0, is mapped to the color index.

Fig. 14, in contrast, shows the result of selecting a cluster of curves with similar frame-field properties and then highlighting the original streamlines back in the 3D space display. This method assists in the location of similar curves that could not be easily singled out in the original densely populated spatial display. We are examining a variety of alternative approaches to the design of such tools.



(a) (b)

Fig. 14. (a) Selecting stream fields that are close in the 4D quaternion Frenet frame display and (b) echoing them in the original 3D data display, thus showing the locations of similar curves that could not be easily singled out in the original 3D spatial display. The unseen component of 4D depth, with a range -1.0 to 1.0 is mapped to the color index.

VIII. Conclusion

In this paper, we have introduced a visualization method for distinguishing characteristic features of streamline-like volume data by assigning to each streamline a quaternion frame field derived from its moving Frenet or parallel-transport frame; curvature and torsion scalar fields may be incorporated as well. The quaternion frame is a four-vector field that is a piecewise smoothly varying map from each original space curve to a new curve in the three-sphere embedded in four-dimensional Euclidean space. This four-vector field can

be probed interactively using a variety of techniques, including 4D lighting, 4D view control, and interaction with selected portions of the data in tandem 3D streamline and 4D quaternion field displays.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under Grant No. IRI-91-06389. We thank Brian Kaplan for his assistance with the vector field data set. We are indebted to Bruce Solomon for bringing reference [4] to our attention, and to the referees for a number of helpful suggestions.

REFERENCES

[1] B. Alpern, L. Carter, M. Grayson, and C. Pelkie, "Orientation maps: Techniques for visualizing rotations (a consumer's guide)," *Proc. Visualization '93*, IEEE CS Press, pp. 183-188, 1993.

[2] S.L. Altmann, *Rotations, Quaternions, and Double Groups*. Oxford Univ. Press, 1986.

[3] D.C. Banks, "Illumination in diverse codimensions," *Computer Graphics, (Proc. SIGGRAPH '94)*, Ann. conf. series, pp. 327-334, 1994.

[4] R.L. Bishop, "There is more than one way to frame a curve," *Am. Math. Monthly*, vol. 82, no. 3, pp. 246-251, Mar. 1975.

[5] J. Bloomenthal, "Calculation of reference frames along a space curve," A. Glassner, ed., *Graphics Gems*, Cambridge, Mass.: Academic Press, pp. 567-571, 1990.

[6] M. Chen, S.J. Mountford, and A. Sellen, "A study in interactive 3d rotation using 2d control devices," *Computer Graphics, (Proc. SIGGRAPH '83)*, vol. 22, pp. 121-130, 1988.

[7] B. Cipra, "Mathematicians gather to play the numbers game," *Science 259*, pp. 894-895, 1993. Description of Alfred Gray's knots colored to represent variation in curvature and torsion.

[8] R.A. Cross and A.J. Hanson, "Virtual reality performance for virtual geometry" *Proc. Visualization '94*, IEEE CS Press, pp. 156-163, 1994.

[9] L.P. Eisenhart, *A Treatise on the Differential Geometry of Curves and Surfaces*. Dover, New York, (1909) 1960.

[10] A. Gray, *Modern Differential Geometry of Curves and Surfaces*. Boca Raton, Fla.: CRC Press, Inc., 1993.

[11] A.J. Hanson, "The rolling ball," D. Kirk, ed., *Graphics Gems III*, Cambridge, Mass.: Academic Press, pp. 51-60, 1992.

[12] A.J. Hanson, "Quaternion Frenet frames," Tech. Report 407, Computer Science Dept., Indiana Univ., 1994.

[13] A.J. Hanson, "Rotations for n-dimensional graphics," *Graphics Gems V*, A. Paeth, ed., Cambridge, Mass.: Academic Press, pp. 55-64, 1995.

[14] A.J. Hanson and R.A. Cross, "Interactive visualization methods for four dimensions," *Proc. Visualization '93*, IEEE CS Press, pp. 196-203, 1993

[15] A.J. Hanson and P.A. Heng, "Visualizing the fourth dimension using geometry and light," *Proc. Visualization '91*, IEEE CS Press, pp. 321-328, 1991.

[16] A.J. Hanson and P.A. Heng, "Illuminating the fourth dimension," *Computer Graphics and Applications*, vol. 12, no. 4, pp. 54-62, July 1992.

[17] A.J. Hanson and H. Ma, "Visualizing flow with quaternion frames," *Proc Visualization '94*, IEEE CS Press, pp. 108-115, 1994.

[18] A.J. Hanson and H. Ma, "Parallel transport approach to curve framing," Tech. Report 425, Computer Science Dept., Indiana Univ. 1995.

[19] J.C. Hart, G.K. Francis, and L.H. Kauffman, "Visualizing quaternion rotation," *ACM Trans. Graphics*, vol. 13, no. 3, pp. 256-276, 1994.

[20] J. Hultquist, "Constructing stream surfaces in steady 3d vector fields," *Proc. Visualization '92*, IEEE CS Press, pp. 171-178, 1992.

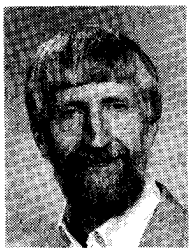
[21] J.T. Kajiya and T.L. Kay, "Rendering fur with three dimensional textures," J. Lane, ed., *Computer Graphics (Proc. SIGGRAPH '89)*, vol. 23, pp. 271-280, July 1989.

[22] H. Ma and A. Hanson, *Meshview*, A portable 4D geometry viewer written in OpenGL/Motif, available by anonymous ftp from geom.umn.edu, The Geometry Center, Minneapolis, Minn.

[23] J. Schlag, "Using geometric constructions to interpolate orientation with quaternions," J. Arvo, ed., *Graphics Gems II*, Academic Press, pp. 377-380, 1991.

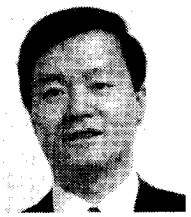
[24] U. Shani, and D.H. Ballard, "Splines as embeddings for generalized cylinders," *Computer Vision, Graphics and Image Processing*, vol. 27, pp. 129-156, 1984.

[25] K. Shoemake, "Animating rotations with quaternion curves," *Computer Graphics, (Proc. SIGGRAPH '85)*, vol. 19, pp. 245-254, 1985.



Andrew J. Hanson is a professor of computer science at Indiana University. Previously, he worked in theoretical physics and then with the perception research group at the SRI Artificial Intelligence Center. His research interests include scientific visualization with applications in mathematics and physics, machine vision, computer graphics, perception, and the design of interactive user interfaces for virtual reality and visualization applications.

Hanson was awarded his BA degree in chemistry and physics from Harvard College in 1966 and his PhD in theoretical physics from Massachusetts Institute of Technology in 1971. He is a member of AAAI, American Mathematical Society, American Physical Society, ACM Siggraph, IEEE Computer Society, and Sigma Xi.



Hui Ma was awarded the BS degree in applied mathematics from Zhejiang University, China in 1990. H. Ma is currently a doctoral candidate in computer science at Indiana University.

His research interests include computer graphics and visualization, user interface design, parallel and distributed systems.