Computation of Rotation Minimizing Frame in Computer Graphics

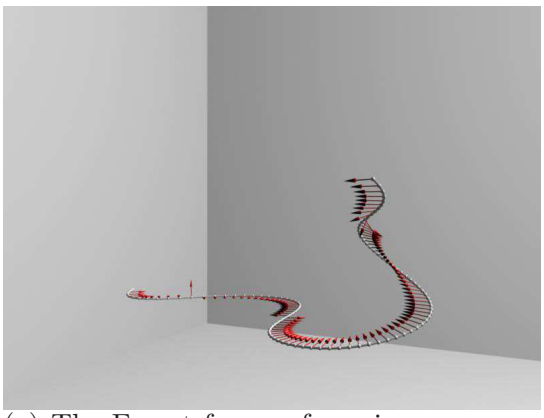
Wenping Wang [a] Bert Jüttler [b] Dayue Zheng [a] Yang Liu [a]

[a] *Department of Computer Science, University of Hong Kong*
[b] *Johannes Kepler University*

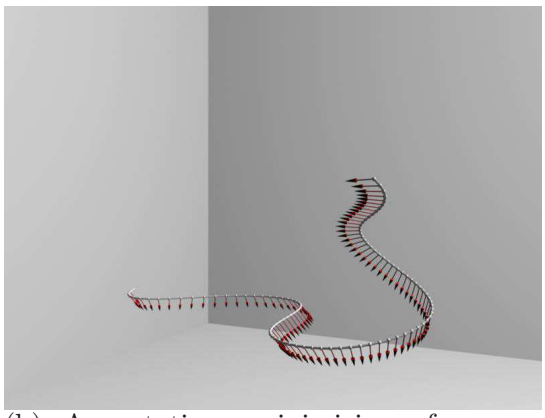
Abstract

We investigate the computation and properties of rotation minimizing frame (RMF), which is a moving orthonormal frame $U(u)$ attached to a smooth curve $\mathbf{x}(u)$, called the *spine curve*, in 3D such that $U(u)$ does not rotate about the instantaneous tangent of $\mathbf{x}(u)$. Due to its minimal-twist property, the RMF is widely used in computer graphics, including sweep or blending surface modeling, motion design and control in computer animation and robotics, streamline visualization, and tool path planning in CAD/CAM. In general, the RMF cannot be computed exactly and therefore one often needs to approximate the exact RMF by a sequence of orthonormal frames at sampled points on the spine curve. We present a novel simple and efficient method for accurate and stable computation of an RMF for any C^1 regular curve in 3D. This method, called the *double reflection method*, uses two reflections to compute each frame from its preceding one to yield a sequence of frames to approximate an exact RMF. The double reflection method is highly accurate – it has the global fourth order approximation error, thus comparing favorably to the second order approximation error of two currently prevailing methods – the projection method by Klok and the rotation method by Bloomenthal, while all these methods have comparable per-frame computational cost. Furthermore, the double reflection method is much simpler and faster than using the standard 4-th order Runge-Kutta method to integrate the defining ODE of the RMF, which yields the same accuracy as the double reflection method. We also present further properties and extensions of the double reflection method for various application scenarios. Finally, we discuss the variational principles in design moving frames with boundary conditions, based on the RMF.

Key words: curve, motion, rotation minimizing frame



(a) The Frenet frame of a spine curve. Only the normal vector is shown.



(b) A rotation minimizing frame (RMF) of the same curve in (a). Only the reference vector is shown.

(c) A snake modeled using the RMF in (b).

Fig. 1. An example of using the RMF in shape modeling.

1 Introduction

1.1 Background

Let $\mathbf{x}(u) = (x(u), y(u), z(u))^T$ be a C^1 regular curve in $\mathbb{E}^3$, the 3D Euclidean space. Denote $\mathbf{x}'(u) = \mathrm{d}\mathbf{x}(u)/\mathrm{d}u$ and $\mathbf{t}(u) = \mathbf{x}'(u)/||\mathbf{x}'(u)||$, which is the unit tangent vector of the curve $\mathbf{x}(u)$. We define a *moving frame* associated with $\mathbf{x}(u)$ to be a right-handed orthonormal system composed of an ordered triple of vectors $U(u) = (\mathbf{r}(u), \mathbf{s}(u), \mathbf{t}(u))$ satisfying $\mathbf{r}(u) \times \mathbf{s}(u) = \mathbf{t}(u)$. The curve $\mathbf{x}(u)$ in this context will be called a *spine curve*. Since $\mathbf{t}(u)$ is known and $\mathbf{s}(u) = \mathbf{t}(u) \times \mathbf{r}(u)$, a moving frame is uniquely determined by the unit normal

vector $\mathbf{r}(u)$. Thus $\mathbf{r}$ is called the *reference vector* of a moving frame.

From the differential geometry point of view, a readily available moving frame of a curve in 3D is the Frenet frame, whose three orthogonal axis vectors are defined as

$$\mathbf{t} = \frac{\mathbf{x}'(u)}{\|\mathbf{x}'(u)\|}, \quad \mathbf{s} = \frac{\mathbf{x}'(u) \times \mathbf{x}''(u)}{\|\mathbf{x}'(u) \times \mathbf{x}''(u)\|}, \quad \mathbf{r} = \mathbf{s} \times \mathbf{t}.$$

Although the Frenet frame can easily be computed, its rotation about the tangent of a general spine curve often leads to undesirable twist in motion design or sweep surface modeling. Moreover, the Frenet frame is not continuously defined for a C^1 spine curve, and even for a C^2 spine curve the Frenet frame becomes undefined at an inflection point (i.e., curvature $\kappa = 0$), thus causing unacceptable discontinuity when used for sweep surface modeling [6].

A moving frame that does not rotate about the instantaneous tangent of the curve $\mathbf{x}(u)$ is called a *rotation minimizing frame* of $\mathbf{x}(u)$, or RMF, for short. It can be shown that the RMF is defined continuously for any C^1 regular spine curve. Because of its minimal-twist property and stable behavior in the presence of inflection points, the RMF is preferred to the Frenet frame in many applications in computer graphics, including free-form deformation with curve constraints [3,26,23,25,24], sweep surface modeling [7,28,31,32], modeling of generalized cylinders and tree branches [30,5,8,29], visualization of streamlines and tubes [1,16,14], simulation of ropes and strings [2], and motion design and control [18]. Discussion of the RMF and its applications can be found in the recent book by Hanson [15], where the RMF is treated using a parallel transport approach.

A typical application of RMF in shape modeling is shown in Figure 1. Here a canonical snake surface model is first defined along a straight line axis possessing an RMF generated by translation along the line. Then a new axis curve (i.e., a spine curve) is designed to produce a novel pose of the snake. For comparison, both Frenet frame and RMF of this same axis curve are shown in Figures 1(a) and 1(b). The RMF determines a mapping from the space of the canonical model of the snake to the space around the new axis curve in Figure 1(b); this mapping produces the snake in Figure 1(c). Note that the Frenet frame in this case exhibits excessive rotation compared with the RMF, so it is less appropriate for shape modeling.

Next consider moving frames of a deforming spine curve $\mathbf{x}(u;t)$, as frequently encountered in computer animation (see Figure 2). While the Frenet frame does not always experience abrupt twist for a given static spine curve, the Frenet frame of the deforming spine curve often suddenly exhibits a radical twist at an instant during deformation, especially when the spine curve has a nearly curvature vanishing point (i.e., an inflection point). In contrast, the RMF of the deforming spine curve $\mathbf{x}(u;t)$ always varies smoothly and stably

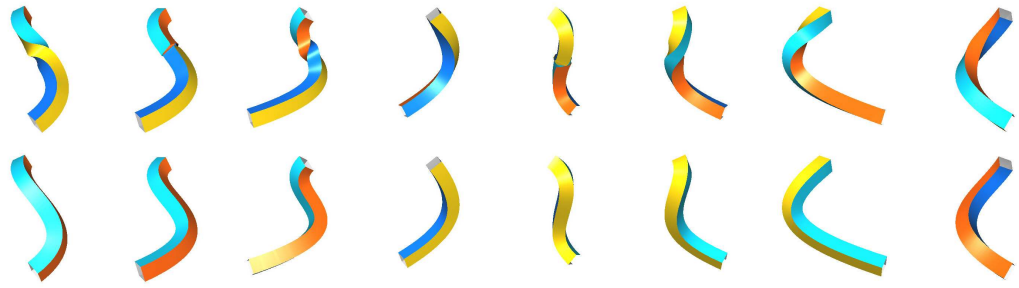


Fig. 2. Sweep surfaces showing moving frames of a deforming curve: the Frenet frames in the first row and the RMF in the second row.

over time as well as along the spine curve. The different behaviors of these two moving frames are illustrated in Figure 2, visualized as sweep surfaces, through a sequence of snapshots of a deforming spine curve. Here by continuous deformation we mean that the rate of change in both position (i.e., $\partial \mathbf{x}(u;t)/\partial t$) and unit tangent (i.e., $\partial \mathbf{t}(u;t)/\partial t$) are bounded for any (u,t) in their finite intervals of definition. Note that, this assumption is reasonable in practical application but does not imply that the normal vector of $\mathbf{x}(u;t)$ changes continuously with respect to time t, thus explaining the potential instability of the Frenet frame.

Computation of the RMF is more involved than that of the Frenet frame. The RMF is first proposed and formulated as the solution of an ordinary differential equation in [4] and later in [30,21]. Exact (i.e., closed form) RMF computation is either impossible or very involved for a general spine curve. Hence, a number of approximation methods have been proposed for RMF computation. These methods fall under three categories: 1) *discrete approximation*; 2) *spine curve approximation*; and 3) *numerical integration*. The discrete approximation approach is versatile for various applications in computer graphics and computer animation, even when only a sequence of points on a path (i.e., spine curve) is available, while the approach based on spine curve approximation is useful for surface modeling in CAGD applications. We will see that direct numerical integration of the defining ODE of RMF is relatively inefficient and therefore not well suited for RMF computation. The new method we are going to propose is based on discrete approximation.

1.2 Problem formulation

The RMF computation problem as solved by the discrete approximation approach is formulated as follows. Let $U(u)$ denote an exact RMF of a C^1 regular spine curve $\mathbf{x}(u)$ in 3D, $u \in [0, L]$, with the initial condition $U(0) = U_0$, which is some fixed orthonormal frame at the initial point $\mathbf{x}(0)$. Suppose that a sequence of points $\mathbf{x}_i = \mathbf{x}(u_i)$ and the unit tangent vectors $\mathbf{t}_i$ at $\mathbf{x}_i$ are sampled

on the curve $\mathbf{x}(u)$, with $u_i = i * h$, $i = 0, 1, \ldots, n$, where $h = L/n$ is called the *step size*. The goal of discrete approximation is to compute a sequence of orthonormal frames U_i at $\mathbf{x}_i$ that approximates the exact RMF frame $U(u)$ at the sampled points, i.e., each U_i is an an approximation to $U(u_i)$, $i = 0, 1, 2 \ldots, n$.

Error measurement is needed to evaluate and compare different approximation schemes. Suppose that the exact RMF $U(u)$ has the same initial frame as the approximating frame sequence at $\mathbf{x}(u_0)$, i.e., $U(0) = U_0$. Then the approximation error between U_1 and $U(h)$ is called the *one-step error*. The approximation errors at intermediate sampled points are normally accumulated to give a large error at the end of the spine curve. However, due to error fluctuation, the maximum error may not always occur at the endpoint $\mathbf{x}(L)$. Therefore, we define the *global error* E_g to be the maximum error of frame approximation over all the sampled points $\mathbf{x}(u_i)$, i.e.,

$$E_g = \max_{i=0}^{n}\{|\angle(U_i, U(u_i))|\}, \tag{1}$$

where $|\angle(U_i, U(u_i))|$ measures the magnitude of the angle between the reference vectors $\mathbf{r}_i$ and $\mathbf{r}(u_i)$ of the frames U_i and $U(u_i)$.

We shall present a new discrete approximation method, called *double reflection method*, for RMF computation. The main idea is based on the observation that the rigid transformation between two consecutive frames for RMF approximation can be realized by two reflections, each being a reflection in a plane. The resulting method is simple, fast, and highly accurate – its global approximation error is of order $\mathcal{O}(h^4)$, where $h = L/n$ is the step size. This compares favorably with the second order (i.e., $\mathcal{O}(h^2)$) approximation error of two prevailing discrete approximation methods, i.e., the rotation method [6] and the projection method [21]. The accuracy of the double reflection method matches that of using the standard 4-th order Runge-Kutta method to integrate the defining differential equation of RMF, but is much simpler and faster than the latter.

In the following we will first review related works in Section 2 and present necessary preliminaries in 3. The double reflection method is presented and analyzed in Section 4. Then we present experimental verifications in Section 5, discuss extensions in Section 6 and conclude the paper in Section 7.

Readers interested only in implementation may skip to Section 4.1 for a simple description of the double reflection method; the pseudo code is given in Table 1 in Section 4.

2 Related Work

2.1 Discrete approximation

In discrete approximation an RMF is approximated by a sequence of orthogonal frames located at sampled points $\mathbf{x}_i$ on the spine curve $\mathbf{x}(u)$. The projection method, as originally proposed in [21], computes an approximate RMF for modeling a sweep surface. Suppose that the the sampled points $\mathbf{x}_i$ and the unit tangent vectors $\mathbf{t}_i$ of $\mathbf{x}(u)$ at the sampled points $\mathbf{x}_i$ are provided as input. For RMF computation, the projection method projects, along the direction $\mathbf{x}_1 - \mathbf{x}_0$, an initial reference vector $\mathbf{r}_0$ in the normal plane of the spine curve at $\mathbf{x}_0$ to the next reference vector $\mathbf{r}_1$ on the normal plane at $\mathbf{x}_1$. Then this step is repeated to generate on the subsequent normal planes a sequence of reference vectors $\mathbf{r}_i$, which, together with the tangent vectors $\mathbf{t}_i$, define a sequence of orthonormal frames that approximate an exact RMF. The projection method is empirically demonstrated to have the second order of approximation error [10]. Note that the above projection between normal planes is not length preserving. Therefore the reference vectors $\mathbf{r}_i$ need to be normalized to give unit vectors.

Another popular discrete approximation method is the *rotation method* [6,31,27]. The rotation method also needs as input the sampled points $\mathbf{x}_i$ on the spine curve and the unit tangent vectors $\mathbf{t}_i$ of the spine curve at $\mathbf{x}_i$. Consider the first two sampled points $\mathbf{x}_0$ and $\mathbf{x}_1$. Given the initial frame U_0 at $\mathbf{x}_0$, suppose that we need to compute the next frame U_1 at $\mathbf{x}_1$ from the *boundary data* $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$. To minimize the rotation about the tangent of the spine curve, this method rotates U_0 into U_1 about an axis $\mathbf{b}_0$ perpendicular to $\mathbf{t}_0$ and $\mathbf{t}_1$, that is, $\mathbf{b}_0 = \mathbf{t}_0 \times \mathbf{t}_1$; the rotation angle θ is such that the frame vector $\mathbf{t}_0$ of U_0 is brought into alignment with the frame vector $\mathbf{t}_1$ of U_1, i.e., $\theta = \arccos(\mathbf{t}_0 \cdot \mathbf{t}_1)$. Here, for frame computation, we ignore the translational difference between the origins of U_0 and U_1. The rotation method has the second order global approximation error [27].

A major problem with the rotation method is its lack of robustness for nearly collinear data. When the two consecutive tangent vectors $\mathbf{t}_0$ and $\mathbf{t}_1$ are collinear, the rotation axis becomes undefined, since $\mathbf{b}_0 = \mathbf{t}_0 \times \mathbf{t}_1 = 0$; but, since no rotation is needed in this case, we just need to set $U_1 := U_0$. However, numerical problems will be experienced when $\mathbf{t}_0$ and $\mathbf{t}_1$ approach each other, i.e., becoming closer and closer to being collinear; this happens, for example, when the spine curve is densely sampled for high accuracy RMF computation. In this case some threshold value has to be used to avoid the degeneracy of the rotation vector $\mathbf{b}_0$ by treating nearly collinear data as collinear data. But if a spine curve is so densely sampled that all consecutive data segments are deemed as

collinear due to thresholding, then there will be a large accumulated error in the computed RMF, because the spine curve will be treated as a straight line and all the frames U_i will be set to be identical to the initial frame U_0. We note that this numerical problem for nearly collinear data does not exist with the double reflection method we are going to propose.

2.2 Methods based on spine curve approximation

If a spine curve is first approximated by some simple curves whose RMF can be computed exactly or more accurately, then the RMF of this simple approximating curve can be taken as an approximation to the RMF of the original spine curve. An intuitive argument for this idea is that if two spine curves are close to each other, then their RMFs should also be. This type of intuition lacks rigorous justification and could be unreliable for moving frames defined by differential properties; recall that the Frenet frames of two spine curves close to each other can be radically different. A related result by Poston et al [27] basically states that the RMF of a spine curve $\tilde{\mathbf{x}}(u)$ approaches the RMF of another spine $\mathbf{x}(u)$ if and only if the unit tangent vector $\tilde{\mathbf{t}}(u)$ of $\tilde{\mathbf{x}}(u)$ approaches the unit tangent vector $\mathbf{t}(u)$ of $\mathbf{x}(u)$.

Discrete approximation methods, such as the projection method or the rotation method, can be regarded as the simplest methods based on spine curve approximation, using a polygon to approximate the spine curve. A G^1 spline curve composed of circular arcs is used to approximate an input spine curve in [32] to compute an approximate RMF for modeling sweep surfaces in NURBS form. The spine curve is approximated by PH curves using Hermite interpolation in [19] for generating sweep surfaces in rational representation. Exact description of the RMF of a PH curve and its rational approximation are provided in [18,11,12,9]. A closely related technique is to approximate the rotation minimizing motions (RMM) by affine motions (cf. [28]) and rational motions from the point of view of spherical kinematics [17].

2.3 Numerical integration

Since the RMF is defined by a vector-valued ODE of the type $\mathbf{y}' = \mathbf{f}(\mathbf{x}, \mathbf{y})$ [4,30,21,28], naturally one may consider computing the RMF using a numerical method to directly solve this ODE. Suppose that the classical fourth order Runge-Kutta method is used. Then the RMF thus computed has the 4-th order global approximation error, which is the same as that of the double reflection method that we are to propose. However, this general approach to solving the ODE does not take into account the special geometric property of the problem of RMF computation and therefore has severe drawbacks.

Firstly, the Runge-Kutta method requires the spine curve $\mathbf{x}(u)$ to be C^2, since the right hand side $\mathbf{f}$ of the ODE is a function of the second derivative of $\mathbf{x}(u)$ (cf. Eqn. (5) in Section 3). This requirement is unnecessarily restrictive, since the RMF is continuously defined for any C^1 spine curve. Secondly, deriving and evaluating the second derivative of $\mathbf{x}(u)$ can be tedious and costly, rendering the method inefficient. In the RMF computation problem under consideration, the sampled points $\mathbf{x}_i$ and the tangent vectors $\mathbf{t}_i$ are available as input. But both first and second derivatives of the spine $\mathbf{x}(u)$ are required by the Runge-Kutta method. This mismatch between the input data of the RMF computation problem and the data it requires makes the Runge-Kutta method not well suited for RMF computation.

Another problem is that the Runge-Kutta method does not strictly enforce the orthogonality between the solved reference vectors $\mathbf{r}_i$ and the tangent vectors $\mathbf{t}_i$, even though in the initial conditions $\mathbf{r}_0 = \mathbf{r}(0)$ is orthogonal to $\mathbf{t}_0 = \mathbf{t}(0)$. Therefore each $\mathbf{r}_i$ has to be projected onto the normal plane of the spine curve to make it perpendicular to $\mathbf{t}_i$; this adds further to the cost of the method.

Another method is based on the observation that the RMF and the Frenet frame differ by a rotation determined by the torsion in the normal plane of the spine curve. Let $\theta(u)$ be the angle of this rotation. Let $\tau(u)$ be the torsion of the spine curve $\mathbf{x}(u)$. Then $\theta(u)$ is given by [13]

$$\theta(u) = -\int_{u_0}^{u} \tau(v)\|x'(v)\|\mathrm{d}v. \tag{2}$$

With this formula, $\theta(u)$ may be computed with some quadrature rule and used to compute the RMF by compensating the rotation of the Frenet frame. However, at inflection points of a spine curve, the Frenet frame itself becomes discontinuous and exhibits abrupt change, and the torsion $\tau(u)$ becomes ill-defined (i.e., unbounded), making it difficult to evaluate the integration (2) accurately; therefore in this case the method becomes unstable. This problem is further discussed with an example in Appendix II.

3 Preliminaries

3.1 Definition by differential equations

First we introduce the rotation minimizing frame under weak assumptions on a spine curve, using differential equations. These results will later be connected to the classical results from differential geometry. Generally, we assume the spine curve $\mathbf{x}(u)$ to be a C^1 regular curve, i.e., $\mathbf{x}'(u) \neq 0$ in its domain of

definition, but higher differentiability is needed for analysis of approximation orders. Again we use $\mathbf{t}(u) = \mathbf{x}'(u)/||\mathbf{x}'(u)||$ to denote the unit tangent vector.

Consider a one-parameter family of unit vectors $\mathbf{f}(u)$ perpendicular to the tangent vector $\mathbf{t}(u)$. Such a vector function $\mathbf{f}(u)$ is said to exhibit the *minimal rotation*, and therefore called a *rotation minimizing vector*, if it is a solution to the following system of differential–algebraic equations (DAE)

$$\left.\begin{aligned} \mathbf{f}'(u) - \phi(u)\,\mathbf{t}(u) &= 0 \\ \mathbf{f}(u) \cdot \mathbf{t}(u) &= 0 \end{aligned}\right\} \tag{3}$$

for the functions $\mathbf{f}(u) = (f_1(u), f_2(u), f_3(u))^\top$ and some function $\phi(u)$. Here the first equation (in vector form) constrains the evolution of $\mathbf{f}(u)$ to be parallel to the tangent, and the second equation serves to preserve orthogonality.

A rotation minimizing vector $\mathbf{f}(u)$ is not necessarily differentiable for a C^1 spine curve $\mathbf{x}(u)$; (e.g., consider the case of a C^1 curve composed of a circular arc and a straight line segment). In view of this, one may adopt the following *weak form* of the DAE (3)

$$\left.\begin{aligned} \mathbf{f}(u) - \int_0^u \phi(v)\,\mathbf{t}(v)\,\mathrm{d}v &= 0 \\ \mathbf{f}(u) \cdot \mathbf{t}(u) &= 0 \end{aligned}\right\} \tag{4}$$

which does not involve any derivative of $\mathbf{f}(u)$.

If the spine curve is of the C^2 class, then the above DAE is equivalent to the ODE

$$\mathbf{f}'(u) = [\mathbf{t}(u) \times \mathbf{t}'(u)] \times \mathbf{f}(u) \tag{5}$$

since

$$\phi\,\mathbf{t} = (\mathbf{f}' \cdot \mathbf{t})\mathbf{t} = (-\mathbf{f} \cdot \mathbf{t}')\mathbf{t} = [\mathbf{t}(u) \times \mathbf{t}'(u)] \times \mathbf{f}(u) \tag{6}$$

A rotation minimizing frame (RMF) is determined by a rotation minimizing vector. Specifically, we have

Definition 1: [Rotation minimizing frame] *Given a C^1 curve $\mathbf{x}(u) \subset \mathbb{E}^3$, $u \in [0, L]$, a moving orthonormal frame $U(u) = (\mathbf{r}(u), \mathbf{s}(u), \mathbf{t}(u))$, where $\mathbf{r}(u) \times \mathbf{s}(u) = \mathbf{t}(u)$, is called a* rotation minimizing frame *(RMF) of $\mathbf{x}(u)$ if $\mathbf{t}(u) = \mathbf{x}'(u)/||\mathbf{x}'(u)||$ and $\mathbf{r}(u)$ is a solution of Eqn. (4) (or Eqn.(3) if $\mathbf{x}(u)$ is C^2) for some initial condition $U(0) = U_0$. Here $\mathbf{r}(u)$ is called the* reference vector *of the RMF $U(u)$.*

Since the frame vector $\mathbf{t}(u)$ of $U(u)$ is always constrained to be the unit tangent vector of $\mathbf{x}(u)$, $U(u)$ is uniquely determined by its *reference vector* $\mathbf{r}(u)$, which

is a rotation minimizing vector. The third frame vector is given by $\mathbf{s}(u) = \mathbf{t}(u) \times \mathbf{r}(u)$.

The evolution defined by DAE (3) preserves the inner product of two vectors. Indeed, if vectors $\mathbf{f}(u)$ and $\mathbf{g}(u)$ both satisfy Eqn.(3) with associated functions $\phi(u)$ and $\psi(u)$, then

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{f} \cdot \mathbf{g}) = \mathbf{f}' \cdot \mathbf{g} + \mathbf{f} \cdot \mathbf{g}' = (\phi\,\mathbf{t}) \cdot \mathbf{g} + \mathbf{f} \cdot (\psi\,\mathbf{t}) = 0 \tag{7}$$

Hence, the inner product $(\mathbf{f} \cdot \mathbf{g})$ is a constant. From this we have the following observations:

Corollary 3.1 *If two vectors $\mathbf{f}_1(u)$ and $\mathbf{f}_2(u)$ satisfy Eqn. (3) and the three vectors $\mathbf{f}_1(0)$, $\mathbf{f}_2(0)$ and $\mathbf{t}(0)$ form a right–handed orthonormal frame, then $\mathbf{f}_1(u)$, $\mathbf{f}_2(u)$ and $\mathbf{t}(u)$ define an RMF of the spine curve $\mathbf{x}(u)$.*

Corollary 3.2 *Suppose that $\mathbf{r}(u)$ is a rotation minimizing vector of a spine curve $\mathbf{x}(u)$. Then another normal vector $\tilde{\mathbf{r}}(u)$ of $\mathbf{x}(u)$ is a rotation minimizing vector of $\mathbf{x}(u)$ if and only if $\tilde{\mathbf{r}}(u)$ keeps a constant angle with $\mathbf{r}(u)$.*

Or, equivalently,

Corollary 3.3 *Suppose that $U(u) = (\mathbf{r}(u), \mathbf{s}(u), \mathbf{t}(u))$ is an RMF of a spine curve $\mathbf{x}(u)$. Then another right-handed orthonormal moving frame $\tilde{U}(u) = (\tilde{\mathbf{r}}(u), \tilde{\mathbf{s}}(u), \mathbf{t}(u))$ of $\mathbf{x}(u)$ is an RMF of $\mathbf{x}(u)$ if and only if $\tilde{U}(u)$ keeps a constant angle with $U(u)$.*

Finally, we note that the RMF is determined only by the geometry of a spine curve and independent of any particular parameterization $\mathbf{x}(u)$ of it.

3.2 Some differential geometry

In this subsection we shall use the arc-length parameterization $\mathbf{x}(s)$ of the spine curve. Using the Frenet formulas one may express (5) as

$$\mathbf{f}'(s) = \kappa(s)\mathbf{b}(s) \times \mathbf{f}(s), \tag{8}$$

where $\kappa(s)$ and $\mathbf{b}(s)$ are the curvature and the binormal vector of $\mathbf{x}(s)$. The vector

$$\omega_{\mathrm{RMF}}(s) = \kappa(s)\mathbf{b}(s) \tag{9}$$

is the angular velocity of the RMF.

The angular velocity of the Frenet frame is the so–called Darboux vector [22]

$$\omega_{\mathrm{Frenet}}(s) = \kappa(s)\mathbf{b}(s) + \tau(s)\mathbf{t}(s) \tag{10}$$

This shows that, compared to the RMF, the Frenet frame involves an additional rotation around the tangent, whose speed equals the torsion τ. This observation explains the integral formula (2) for computing the RMF by correcting the "unwanted" rotation of the Frenet frame. The Frenet frame coincides with the RMF for planar curves, for which $\tau \equiv 0$.

The RMF is also closely related to developable surfaces and principal curvature lines of a surface. Suppose that $U(u) = (\mathbf{r}(u), \mathbf{s}(u), \mathbf{t}(u))$ is an RMF of a curve $\mathbf{x}(u)$. Then the surface $D(u, v) = \mathbf{x}(u) + v\mathbf{r}(u)$ is developable. Let $\mathbf{g}(u)$ be the edge of regression of the developable surface $D(u, v)$. Then the spine curve is an involute of the curve $\mathbf{g}(u)$. This observation suggests a natural (but restrictive) way of modeling a developable ribbon surface along a spine curve using the RMF.

Suppose that $\mathbf{x}(u)$ is a principal curvature line of a surface S. Then the consistent unit normal vector of S along the curve $\mathbf{x}(u)$ is a rotation minimizing vector of $\mathbf{x}(u)$, thus determining an RMF of $\mathbf{x}(u)$. This follows from the well known fact that the normals of S along $\mathbf{x}(u)$ form developable surface if and only if $\mathbf{x}(u)$ is a principal curvature line of S. It therefore also follows that the spine curve $\mathbf{x}(u)$ is a principal curvature line of the developable $D(u, v)$ defined in the last paragraph.

Another important property of the RMF is its preservation under conformal transformation of $\mathbb{E}^3$ [20]. This means that, given a spine curve $\mathbf{x}(u) \subset \mathbb{E}^3$ and a conformal mapping $\mathcal{C}$ of $\mathbb{E}^3$, the RMF of $\mathbf{x}(u)$ is mapped by $\mathcal{C}$ to the RMF of the transformed spine curve $\mathcal{C}(\mathbf{x}(u))$. In other words, the operation of computing RMF of a curve and a conformal transformation commute. This property will be needed later in the analysis of the approximation order of our new method for computing the RMF.

Note that the group of conformal mappings in 3D is exactly the group generated by translations, rotations, uniform scalings and sphere inversions (reflections with respect to spheres). Since a straight line is mapped to a circle by a sphere inversion, in the above the transform of a unit vector $\mathbf{v}$ is defined by the unit tangent vector of the circle which is the image of the straight line associated with $\mathbf{v}$.

4 Double reflection method

In this section we will first give an outline of the double reflection method, and, through a study of the RMF of a spherical curve, explain why the method works well. Then we will give a procedural description of the method that has an optimized number of arithmetic operations, and finally present an analysis

of the approximation order of the method. The double reflection method is straightforward and can very easily be described; however, its justification takes interesting geometric arguments that do not appear to be trivial.

4.1 Outline of method

Given boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$ and an initial right-handed orthonormal frame $U_0 = (\mathbf{r}_0, \mathbf{s}_0, \mathbf{t}_0)$ at $\mathbf{x}_0$, the next frame $U_1 = (\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1)$ at $\mathbf{x}_1$ for RMF approximation is computed by the double reflection method in the following two steps.

Step 1 : Let $\mathcal{R}_1$ denote the reflection in the bisecting plane of the points $\mathbf{x}_0$ and $\mathbf{x}_1$ (see Figure 3). Use $\mathcal{R}_1$ to map U_0 to a left-handed orthonormal frame $U_0^L = (\mathbf{r}_0^L, \mathbf{s}^L, \mathbf{t}_0^L)$.

Step 2 : Let $\mathcal{R}_2$ denote the reflection in the bisecting plane of the points $\mathbf{x}_1 + \mathbf{t}_0^L$ and $\mathbf{x}_1 + \mathbf{t}_1$ (see Figure 4). Use $\mathcal{R}_2$ to map U_0^L to a right-handed orthonormal frame $U_1 = (\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1)$. Output U_1.

An efficient implementation of the above steps is given by the pseudo code in Table 1.

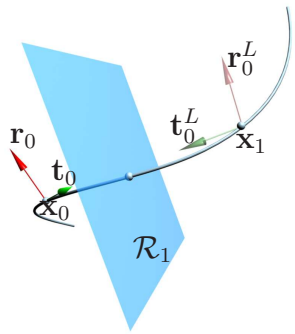


Fig. 3. The first reflection $\mathcal{R}_1$ of the double reflection method.

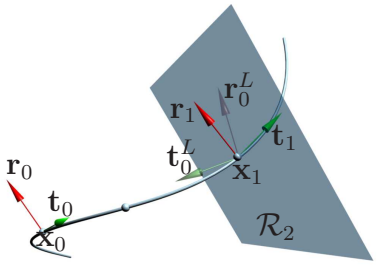


Fig. 4. The second reflection $\mathcal{R}_2$ of the double reflection method.

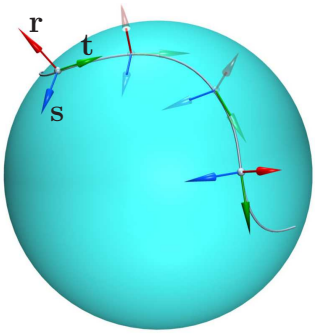


Fig. 5. An RMF of a spherical curve.

4.2 Geometric interpretation

In the following we are going to provide an explanation for why the double reflection method described above computes an accurate approximation of an RMF, based on two key observations: 1) the double reflection method computes an exact RMF of any spherical curve; and 2) a spine curve $\mathbf{x}(u)$ with boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$ is well approximated by a spherical curve $\hat{\mathbf{x}}(u)$ interpolating the same boundary data.

First consider the RMF of a spherical curve. The next lemma indicates that there is a simple explicit characterization of the RMF of a spherical curve. (We will treat a planar curve as special case of a spherical curve where the radius is infinite.)

Lemma 4.1 *Let $\mathbf{x}(u)$, $u \in [0, h]$, be a curve segment lying on a sphere S or a plane P (see Figure 5). Let $\mathbf{n}(u)$ be the outward unit normal vector of the sphere S along the curve $\mathbf{x}(u)$ or a unit (constant) normal vector of the plane P. Then an RMF of $\mathbf{x}(u)$ is given by $\bar{U}_1 = (\bar{\mathbf{r}}, \bar{\mathbf{s}}, \mathbf{t}_1)$, where*

$$\bar{\mathbf{r}}(u) = \mathbf{n}(u) \ \text{ and } \ \bar{\mathbf{s}}(u) = \mathbf{t}(u) \times \mathbf{n}(u). \tag{11}$$

Proof. First consider the case of $\mathbf{x}(u)$ being on a sphere. Without loss of generality, suppose that the sphere S is centered at the origin and has radius r. It is clear that $\mathbf{r}(u) = \mathbf{n}(u)$, $\mathbf{s}(u) = \mathbf{t}(u) \times \mathbf{n}(u)$ and $\mathbf{t}(u)$ form a right-handed orthonormal moving frame. Since $\mathbf{n}(u) = \frac{1}{r}\mathbf{x}(u)$, $\mathbf{r}' = \mathbf{n}' = \frac{1}{r}\mathbf{x}'$, which is parallel to $\mathbf{t}(u)$. Therefore, $\mathbf{r}$ satisfies Eqn. (3), i.e., it is a rotational minimizing vector. Hence, by Definition 1, $U(u) = (\mathbf{r}, \mathbf{s}, \mathbf{t})$ is an RMF of $\mathbf{x}(u)$.

The proof is similar when $\mathbf{x}(u)$ is a plane curve. □.

Lemma 4.1 suggests that, given the initial frame U_0 at $\mathbf{x}_0$, the RMF U_1 of a spherical curve $\mathbf{x}(u)$ at the point $\mathbf{x}_1$ does not depend on the in-between shape of $\mathbf{x}(u)$, but depends only on the boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$. This will be referred to as the *path independence property*, as stated below.

Lemma 4.2 [Path independence property] [1] *Let $\mathbf{x}(u)$ and $\mathbf{y}(v)$ be two curve segments, $u \in [0, h_1]$ and $v \in [0, h_2]$, on a sphere (or a plane) sharing the same boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$. Let $U(u)$ and $V(v)$ denote the RMFs of $\mathbf{x}(u)$ and $\mathbf{y}(v)$, having the same initial frame U_0, i.e., $U(0) = V(0) = U_0$. Then $U(h_1) = V(h_2)$.*

Proof. We will only consider the case of $\mathbf{x}(u)$ and $\mathbf{y}(u)$ being on a sphere

[1] This property is equivalent to the fact that the integral $\int_a^b \tau(s)\mathrm{d}s$ vanishes for closed spherical curves [22].

S; the case of their being on a plane can be proved in a similar way. First suppose that the initial frame U_0 is the special frame $\bar{U}_0 = (\bar{\mathbf{r}}_0, \bar{\mathbf{s}}_0, \mathbf{t}_0)$ where $\bar{\mathbf{r}}_0$ is the unit outward normal vector of the sphere S at $\mathbf{x}_0$ and $\bar{\mathbf{s}}_0 = \mathbf{t}_0 \times \bar{\mathbf{r}}_0$. Then, by Lemma 4.1, the RMFs $\bar{U}_1$ and $\bar{V}_1$ of $\mathbf{x}(u)$ and $\mathbf{y}(u)$ at $\mathbf{x}_1$ are the same, i.e., $\bar{U}_1 = \bar{V}_1 = (\bar{\mathbf{r}}_1, \bar{\mathbf{s}}_1, \mathbf{t}_1)$, where $\bar{\mathbf{r}}_1$ is the unit outward normal vector $\mathbf{n}_1$ of the sphere S at $\mathbf{x}_1$ and $\bar{\mathbf{s}}_1 = \mathbf{t}_1 \times \bar{\mathbf{r}}_1$.

Now suppose that the initial frame $U_0 = (\mathbf{r}_0, \mathbf{s}_0, \mathbf{t}_0)$ is arbitrary. Let α_0 be the angle between U_0 and $\bar{U}_0$. Then, by Corollary 3.3, $U(h_1)$ and $\bar{U}_1$, as two RMFs of $\mathbf{x}(u)$ at the endpoint $\mathbf{x}_1$, keep the same angle α_0. Similarly, the angle between the $V(h_2)$ and $\bar{V}_1$, as two RMFs of $\mathbf{y}(v)$ at the endpoint $\mathbf{x}_1$, is also α_0. It follows that $U(h_1) = V(h_2)$, since $\bar{U}_1 = \bar{V}_1$. $\square$.

Next we show that the double reflection method yields the exact RMF for a spherical curve.

Theorem 4.3 *Let $\mathbf{x}(u)$ be a curve segment, $u \in [0, h]$, on a sphere or a plane with boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$. Let $U(u)$ be an RMF of $\mathbf{x}(u)$. Let $U_0 = U(0)$ and $U_1 = U(h)$. Then, given boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$ and the initial frame U_0, the double reflection method produces the frame U_1.*

Proof. Again we will only consider the case of the curve $\mathbf{x}(u)$ being on a sphere S; the case of a plane can be proved similarly. First consider the special case of $U_0 = \bar{U}_0 = (\bar{\mathbf{r}}_0, \bar{\mathbf{s}}_0, \mathbf{t}_0)$, as defined in the proof of Lemma 4.2. Then, by Lemma 4.1, $U_1 = \bar{U}_1 = (\bar{\mathbf{r}}_1, \bar{\mathbf{s}}_1, \mathbf{t}_1)$. Here, $\bar{\mathbf{r}}_0$ and $\bar{\mathbf{r}}_1$ are unit outward normal vectors of the sphere S at $\mathbf{x}_0$ and $\mathbf{x}_1$, respectively. Recall that in the double reflection method (cf. Section 4.1) the first reflection $\mathcal{R}_1$ is in the bisecting plane (denoted as H_1) of $\mathbf{x}_0$ and $\mathbf{x}_1$, and $\mathcal{R}_1$ maps $\bar{U}_0$ to a left-handed frame $\bar{U}_0^L = (\bar{\mathbf{r}}_0^L, \bar{\mathbf{s}}_0^L, \mathbf{t}_0^L)$. Because the two normals $\bar{\mathbf{r}}_0$ and $\bar{\mathbf{r}}_1$ of S at $\mathbf{x}_0$ and $\mathbf{x}_1$ are symmetric about the plane H_1, we have $\bar{\mathbf{r}}_0^L = \bar{\mathbf{r}}_1$.

Let H_2 denote the bisecting plane of the two points $\mathbf{x}_1 + \mathbf{t}_0^L$ and $\mathbf{x}_1 + \mathbf{t}_1$. Clearly, $\bar{\mathbf{r}}_0^L$ (or $\bar{\mathbf{r}}_1$) is contained in H_2. Since the second reflection $\mathcal{R}_2$ is in the plane H_2, it preserves $\bar{\mathbf{r}}_0^L = \bar{\mathbf{r}}_1$. Furthermore, by its construction, $\mathcal{R}_2$ maps $\mathbf{t}_0^L$ to $\mathbf{t}_1$. Therefore, $\mathcal{R}_2$ maps $\bar{U}_0^L$ to $\bar{U}_1 = (\bar{\mathbf{r}}_1, \bar{\mathbf{s}}_1, \mathbf{t}_1)$. Hence, the theorem holds in the special case of $U_0 = \bar{U}_0$.

Now consider an arbitrary initial frame $U_0 = (\mathbf{r}_0, \mathbf{s}_0, \mathbf{t}_0)$. Let α_0 denote the angle between U_0 and $\bar{U}_0$. Let $\mathcal{R}$ denote the composition of $\mathcal{R}_1$ and $\mathcal{R}_2$, i.e., the total rotation effected by the double reflection method. Clearly, $\mathcal{R}$ maps U_0 to a right-handed orthonormal frame $\hat{U}_1 = (\hat{\mathbf{r}}_1, \hat{\mathbf{s}}_1, \hat{\mathbf{t}}_1)$ such that $\hat{\mathbf{t}}_1 = \mathbf{t}_1$. Therefore, $\hat{U}_1$ and $\bar{U}_1$ differ by a rotation in the normal plane of $\mathbf{x}(u)$ at $\mathbf{x}_1$. Furthermore, since the rotation $\mathcal{R}$ is angle-preserving, the angle between $\hat{U}_1$ and $\bar{U}_1$ is also α_0, since $\mathcal{R}$ maps $\bar{U}_0$ to $\bar{U}_1$, and U_0 to $\hat{U}_1$. On the other hand, by Corollary 3.3, the angle between $U_1 = U(h)$ and $\bar{U}_1$ is also α_0. It follows that $\hat{U}_1 = U_1$, i.e., the exact RMF U_1 of the curve $\mathbf{x}(u)$ at $\mathbf{x}_1$ is generated by

the double reflection method. $\square$.

Not only the RMF of a spherical or plane curve $\mathbf{x}(u)$ is computed exactly by the double reflection method, but also this computation does not refer to the sphere or the plane containing $\mathbf{x}(u)$. That is possible because of the path independence property of the RMF of a spherical curve (cf. Lemma 4.2). Note that when the curve segment $\mathbf{x}(u)$ is C^1 regular and parameterizes a line segment, since $\mathbf{x}(u)$ is a plane curve, its RMF is computed exactly by the double reflection method, with no need of threshold as in the projection method to avoid numerical instability.

Now consider applying the double reflection method to computing the RMF of a general spine curve $\mathbf{x}(u) \subset \mathbb{E}^3$, $u \in [0, h]$, which has boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$ and is not necessarily spherical or planar. In general, there is a unique sphere S such that $\mathbf{x}_0$ and $\mathbf{x}_1$ are on S and $\mathbf{t}_0$ and $\mathbf{t}_1$ are tangent to S at $\mathbf{x}_0$ and $\mathbf{x}_1$. Let $\hat{\mathbf{x}}(u)$ denote the projection of the curve $\mathbf{x}(u)$ onto the sphere S through the center of S. Then it is easy to see that the curve $\hat{\mathbf{x}}(u)$ shares the same boundary data $(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$ with $\mathbf{x}(u)$ and that $\hat{\mathbf{x}}(u)$ approximates $\mathbf{x}(u)$ with an approximation error of order $\mathcal{O}(h^4)$. Since $\mathbf{x}(u)$ is well approximated by $\hat{\mathbf{x}}(u)$ and the double reflection method computes an exact RMF of the spherical curve $\hat{\mathbf{x}}(u)$, it is reasonable to believe that the double reflection method computes an accurate approximation to the RMF of the original spine curve $\mathbf{x}(u)$.

Note that the above argument does not constitute a formal analysis of the approximation accuracy of the double reflection method; it merely provides a geometric and intuitive understanding of why the method is expected to work well for RMF computation. It will be proved in Section 4.6 that the global approximation error of the double reflection method has the order $\mathcal{O}(h^4)$.

4.3 Procedural description

The description of the double reflection method in Section 4.1, though simple in geometric terms, is not for efficient implementation. In this section we will give a procedural description of the method, aiming at minimizing the number of arithmetic operations required.

Since only transformation of vectors matters in RMF computation, we may just use the linear parts, denoted by matrices R_1 and R_2, of the two reflections $\mathcal{R}_1$ and $\mathcal{R}_2$. Since $\mathcal{R}_1$ is a reflection in a plane with normal vector $\mathbf{v}_1 \equiv \mathbf{x}_1 - \mathbf{x}_0$, it can be shown that its linear part is

$$R_1 = I - 2(\mathbf{v}_1 \mathbf{v}_1^T)/(\mathbf{v}_1^T \mathbf{v}_1), \tag{12}$$

Table 1
Algorithm — *Double Reflection*

Input: Points $\mathbf{x}_i$ and associated unit tangent vectors $\mathbf{t}_i$, $i = 0, 1, \ldots, n$.
An initial frame $U_0 = (\mathbf{r}_0, \mathbf{s}_0, \mathbf{t}_0)$.
Output: $U_i = (\mathbf{r}_i, \mathbf{s}_i, \mathbf{t}_i)$, $i = 0, 1, 2, \ldots, n$, as approximate RMF.

```
Begin
for i = 0 to n − 1 do
      Begin
1)    v1 := x_{i+1} − x_i;              /*compute reflection vector of R1. */
2)    c1 := v1 · v1;
3)    r_i^L := r_i − (2/c1) ∗ (v1 · r_i) ∗ v1;     /*compute r_i^L = R1 r_i. */
4)    t_i^L := t_i − (2/c1) ∗ (v1 · t_i) ∗ v1;       /*compute t_i^L = R1 t_i. */
5)    v2 := t_{i+1} − t_i^L;            /*compute reflection vector of R2. */
6)    c2 := v2 · v2;
7)    r_{i+1} := r_i^L − (2/c2) ∗ (v2 · r_i^L) ∗ v2;    /*compute r_{i+1} = R2 r_i^L. */
8)    s_{i+1} := t_{i+1} × r_{i+1};       /*compute vector s_{i+1} of U_{i+1}. */
9)    U_{i+1} := (r_{i+1}, s_{i+1}, t_{i+1});
      End
End
```

where I is the 3×3 identity matrix. We will call $\mathbf{v}_1$ the *reflection vector* of R_1. (Note that R_1 is none other than the Householder transform used for QR matrix decomposition.)

The reflection $\mathcal{R}_2$ has the reflection vector $\mathbf{v}_2 \equiv (\mathbf{x}_1 + \mathbf{t}_1) - (\mathbf{x}_1 + \mathbf{t}_0^L) = \mathbf{t}_1 - \mathbf{t}_0^L$. So its linear part is

$$R_2 = I - 2(\mathbf{v}_2\mathbf{v}_2^T)/(\mathbf{v}_2^T\mathbf{v}_2). \tag{13}$$

Let $\mathbf{r}_0$ be the reference vector of U_0. Then $\mathbf{r}_1 = R_2R_1\mathbf{r}_0$ is the reference vector $\mathbf{r}_1$ of the next frame U_1. With the known tangent vector $\mathbf{t}_1$, the remaining vector $\mathbf{s}_1$ of $U_1 = (\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1)$ is given by $\mathbf{s}_1 = \mathbf{t}_1 \times \mathbf{r}_1$.

The procedure of the double reflection method is given in Table 1. For a given sequence of sampled points $\mathbf{x}_i$ and associated unit tangent vectors $\mathbf{t}_i$, with an initial frame U_0 defined at $\mathbf{x}_0$, one just needs to apply the two reflections R_1 and R_2 to successively generate the approximate RMF U_i at $\mathbf{x}_i$. In each step, from the current frame U_i, we form the first reflection R_1 following Eqn.(12) and use R_1 to map the reference vector $\mathbf{r}_i$ to $\mathbf{r}_i^L$, and also the tangent vector $\mathbf{t}_i$ to $\mathbf{t}_i^L$. Then we use $\mathbf{t}_i^L$ and $\mathbf{t}_{i+1}$ to form the second reflection R_2 following Eqn. (13) and use R_2 to map $\mathbf{r}_i^L$ to the reference vector $\mathbf{r}_{i+1}$ of the next frame U_{i+1}.

4.4 Degenerate cases and symmetry

By degeneracy we mean that either of the reflections $\mathcal{R}_1$ and $\mathcal{R}_2$ becomes undefined. Clearly, $\mathcal{R}_1$ is undefined if and only if $\mathbf{x}_1 - \mathbf{x}_0 = 0$, and $\mathcal{R}_2$ is undefined if and only if $\mathbf{x}_1 + \mathbf{t}_0^L = \mathbf{x}_1 + \mathbf{t}_1$, i.e., the two points $\mathbf{x}_0 + \mathbf{t}_0$ and $\mathbf{x}_1 + \mathbf{t}_1$ are symmetric about the bisecting plane of $\mathbf{x}_0$ and $\mathbf{x}_1$; this is equivalent to $(\mathbf{x}_1 - \mathbf{x}_0) \cdot (\mathbf{t}_1 + \mathbf{t}_0) = 0$ and $(\mathbf{x}_1 - \mathbf{x}_0) \times (\mathbf{t}_1 - \mathbf{t}_0) = 0$. Hence, for proper application of the double reflection method, we need to ensure that the following two conditions are satisfied: (1) $\mathbf{x}_1 - \mathbf{x}_0 \neq 0$; and (2) $(\mathbf{x}_1 - \mathbf{x}_0) \cdot (\mathbf{t}_1 + \mathbf{t}_0) \neq 0$ or $(\mathbf{x}_1 - \mathbf{x}_0) \times (\mathbf{t}_1 - \mathbf{t}_0) \neq 0$. Both conditions are simple to test and can easily be satisfied provided that the spine curve is sufficiently subdivided or sampled.

The double reflection method is symmetric in the following sense. Given a sequence of sampled points $\mathbf{x}_i$, $i = 0, 1, \ldots, n$, on a spine curve $\mathbf{x}(u)$, suppose that the U_i are the frames computed by the double reflection method applied to $\mathbf{x}(u)$ with U_0 as the initial frame. Then the same sequence of frames in the reversed order, i.e., U_{n-i}, $i = 0, 1, \ldots, n$, will be generated by applying the double reflection method starting from $\mathbf{x}_n$, using U_n as the initial frame. This symmetry property can be proved by examining the basic steps of the double reflection method, but we will skip the proof. The projection method and the rotation method also possess this symmetry property, while the Runge–Kutta method does not.

4.5 Invariance under conformal mappings

We have seen that conformal mappings in 3D preserve the RMF of a space curve (cf. Section 3.2). It turns out that the approximate RMF computed with the double reflection method is also preserved by conformal mappings, in the following sense. Suppose that the sampled points $\mathbf{x}_i$ of a spine curve $\mathbf{x}(u)$ are used to compute the approximate RMF U_i of $\mathbf{x}(u)$. Then the images of U_i under a conformal mapping $\mathcal{C}$ are the same as the approximate RMF of the curve $\mathcal{C}(\mathbf{x}(u))$ that are computed by the double reflection method using the sampled points $\mathcal{C}(\mathbf{x}_i)$.

This property follows easily from the fact that the basic step of the double reflection method is performed on the sphere S_i touching the two ends of the data $(\mathbf{x}_i, \mathbf{t}_i; \mathbf{x}_{i+1}, \mathbf{t}_{i+1})$ and this sphere is preserved by any conformal mapping $\mathcal{C}$ (which are a sequence of sphere inversions), i.e., the image $\mathcal{C}(S_i)$ is the sphere touching the transformed data $(\mathcal{C}(\mathbf{x}_i), \mathcal{C}(\mathbf{t}_i); \mathcal{C}(\mathbf{x}_{i+1}), \mathcal{C}(\mathbf{t}_{i+1}))$.

Since both exact RMF and approximate RMF computed with the double reflection method are preserved by conformal mappings, and the conformal mapping is angle preserving, we conclude that the approximation error of the

double reflection method is invariant under conformal mappings.

The double reflection method is an ideal method from the viewpoint of discrete differential geometry. Because the exact RMF of a smooth curve is preserved by conformal mappings, we naturally expect that a good method acting on a discretization of the curve for computing its approximate RMF is invariant under the same group of transformations. The double reflection method indeed satisfies this property. We note that the projection method, the rotation method and the Runge–Kutta method do not possess this property.

4.6 Order of approximation

First consider an analytic curve segment with the arc length parameterization $\mathbf{x}(s)$, $s \in [0, h]$, of length h. Suppose that the initial frame $U(0) = U_0 \equiv (\mathbf{r}_0, \mathbf{s}_0, \mathbf{y}_0)$ of an RMF $U(s)$ of $\mathbf{x}(s)$ is given. We approximate the frame $U(h)$ at $\mathbf{x}_1 = \mathbf{x}(h)$ by the frame U_1 computed with the the double reflection method.

Theorem 4.4 *The one-step error $U(h) - U_1$ in RMF computation introduced by the double reflection method has the order of $\mathcal{O}(h^5)$. Specifically,*

$$\|\mathbf{r}(h) - \mathbf{r}_1\| = \tfrac{1}{720} K h^5 + \mathcal{O}(h^6). \tag{14}$$

Here $K = 2\,{\kappa_1}^2\tau_0 + {\kappa_0}^2{\tau_0}^3 + \kappa_1\kappa_0\tau_1 - \kappa_2\kappa_0\tau_0$ is a bounded constant for a smooth curve, where $\kappa_i = (\mathrm{d}/\mathrm{d}s)^i\kappa(s)|_{s=0}$, $\tau_i = (\mathrm{d}/\mathrm{d}s)^i\tau(s)|_{s=0}$ are the curvature, torsion and their respective derivatives at $s = 0$.

The proof of Theorem 4.4 is given in Appendix I. The constant K in Eqn.(14) has an interesting geometric interpretation. A spherical curve $\mathbf{x}(s)$ is characterized by the differential equation [22]

$$\frac{\tau}{\kappa} - \frac{\mathrm{d}}{\mathrm{d}s}\left\{\frac{\kappa'}{\kappa^2\tau}\right\} = 0.$$

It is easy to verify that the numerator of this equation is

$$K(s) = 2\,\kappa_1(s)^2\tau_0(s) + \kappa_0(s)^2\tau_0(s)^3 + \kappa_1(s)\kappa_0(s)\tau_1(s) - \kappa_2(s)\kappa_0(s)\tau_0(s).$$

Therefore, $K(s) = 0$ if and only if $\mathbf{x}(s)$ is a spherical curve. Hence, intuitively speaking, $K = K(0)$ measures how close $\mathbf{x}(s)$ is to a spherical curve at $s = 0$.

As an obvious corollary of Theorem 4.4, we have the next theorem that the RMF computation by the double reflection method applied to a general regularly parameterized spine curve has the fourth order global approximation error.

Theorem 4.5 *Given a regularly parametrized spine curve* $\mathbf{x}(u)$, $u \in [0, M]$, *let* $\mathbf{x}_i = \mathbf{x}(u_i)$, $i = 0, 1, \ldots, n$, *be points sampled on* $\mathbf{x}(u)$ *with equally spaced parameter values, i.e.,* $u_i = i * h$ *and* $h = M/n$. *Then the global error of the approximate RMF of* $\mathbf{x}(u)$ *computed by the double reflection method applied to the sequence* $\{\mathbf{x}_i\}$ *has the order* $\mathcal{O}(h^4)$.

5 Comparison and Experiments

5.1 Computational cost

We need to count the numbers of operations in order to compare the efficiency of different methods for RMF computation. First consider the operation cost of the double reflection method for computing each new frame U_{i+1} from U_i, following the procedure in Table 1. We will count a subtraction as equivalent to an addition. Step (1) uses 3 *adds*. Step (2) uses 2 *adds*, 3 *mults*. After evaluating c_1 with 1 *div*, step (3) can be completed using 5 *adds*, 7 *mults* and 1 *div*. Similarly, step (4) can be done using, 5 *adds*, 7 *mults* and 1 *div*. Step (5) uses only 3 *adds*. Step (6) uses 2 *adds*, and 3 *mults*. Step (7) uses 5 *adds*, 6 *mults* and 1 *div*. Finally, step (8) uses 3 *adds* and 6 *mults*. Hence, in total, the per frame computation of the double reflection method costs, 28 additions, 32 multiplications and 2 divisions.

As comparison, we next give the operation counts of the projection method and the rotation method. In the projection method [21], the new reference vector $\mathbf{r}_1$ can be computed from $\mathbf{r}_0$ by

$$\mathbf{r}_1 = \mathbf{r}_0 - \frac{\mathbf{r}_0 \cdot \mathbf{t}_1}{(\mathbf{x}_1 - \mathbf{x}_0) \cdot \mathbf{t}_1}(\mathbf{x}_1 - \mathbf{x}_0).$$

This evaluation takes 9 *mults* and 1 *div*. Since $\mathbf{r}_1$ thus derived is in general not a unit vector, 6 *mults*, 1 *div* and one square root are needed to normalize $\mathbf{r}_1$. Then another 6 *mults* are needed to compute the third frame vector $\mathbf{s}_1 = \mathbf{t}_1 \times \mathbf{r}_1$. Hence, in total, the projection method needs 15 additions, 21 multiplications, 2 divisions and 1 square root to compute a new frame. This is less than, but comparable to, the cost of the double reflection method.

A procedure of the rotation method is given in [27]. Given the two consecutive unit tangent vectors $\mathbf{t}_0$ and $\mathbf{t}_1$, the rotation axis is computed as $(a, b, c) = \mathbf{t}_0 \times \mathbf{t}_1$ and the cosine of rotation angle is $\cos\alpha = \mathbf{t}_0 \cdot \mathbf{t}_1$. Then the rotation

Method	# of *adds*	# of *mults*	# of *divs*	# of *sqrt*
Projection	15	21	2	1
Rotation	26	36	1	0
Double reflection	28	32	2	0

Table 2
The operations counts of the three methods.

matrix is given by

$$R = \begin{bmatrix} \cos\alpha & -c & b \\ c & \cos\alpha & -a \\ -b & a & \cos\alpha \end{bmatrix} + \frac{1-\cos\alpha}{a^2+b^2+c^2} \begin{bmatrix} a^2 & ab & ac \\ ab & b^2 & bc \\ ac & bc & c^2 \end{bmatrix}.$$

Therefore, 21 *mults* and 1 *div* are needed to obtain R from $\mathbf{t}_0$ and $\mathbf{t}_1$; (note that 27 *mults* are claimed in [27]). In addition, 9 *mults* are needed for computing the next reference vector $\mathbf{r}_1 = R\mathbf{r}_0$, and 6 *mults* for computing the remaining vector $\mathbf{s}_1 = \mathbf{t}_1 \times \mathbf{r}_1$. Hence, in total, the rotation method needs 26 additions, 36 multiplications and 1 division.

The number of operations for the three methods are listed in Table 2. The three methods have similar computational costs, as our tests show that a *sqrt* or a division is is about six times more time consuming than a multiplication. The actual timing comparison will be given in the next subsection.

It is worth mentioning that another procedure of the rotation method is given in [6], which uses 19 *mults* and a square root to compute the rotation matrix R after using 6 *mults* to get the rotation axis $\mathbf{t}_0 \times \mathbf{t}_1$. Hence, that version of the rotation method requires in total 40 multiplications and a square root to compute a new frame, assuming that the $\mathbf{t}_i$ are unit tangent vectors. In the subsequent experimental comparisons involving the rotation method we will refer to the faster implementation in [27].

5.2 Experimental results

We will use two examples to compare the double reflection method with the following existing methods: the projection method, the rotation method and the 4-th order Runge-Kutta method, in terms of efficiency and accuracy. All test cases were run on a PC with Intel Xeon 2.66 GHz CPU and 2.00 GB RAM.

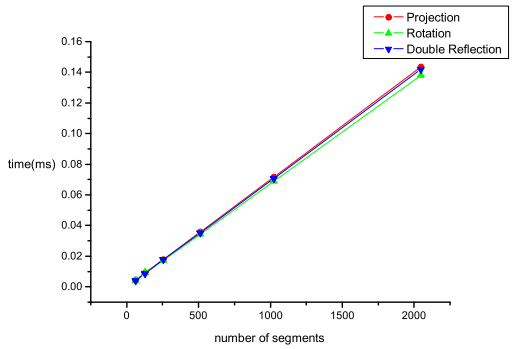


Fig. 6. Timings of the double reflection method, the projection method and the rotation method.

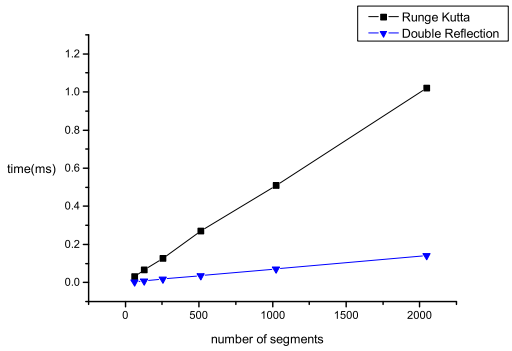


Fig. 7. Timings of Runge-Kutta method and the double reflection method.

Example 1: In the first example we use the four methods to compute the RMF of the spine curve, which is a torus knot, given by

$$\mathbf{x}(u) = [(0.6{+}0.3\cos(7u))\cos(2u), (0.6{+}0.3\cos(7u))\sin(2u), 0.3\sin(7u)]^T, \quad u \in [0, L] \tag{15}$$

We compute the RMF using different step sizes $h = 0.01 * 2^{-k}$, $k = 0, 1, \ldots$; that is, for each fixed step size h, the sampled points are $\mathbf{x}(i * h)$, $i = 0, 1, \ldots, L/h$.

The timings of computing the sequence of frames by the four methods are shown in Figures 6 and 7. We see that the projection method, the rotation method and the double reflection method have similar time costs. The Runge-Kutta method costs much more time than the double reflection method, since it needs more function evaluations in each step than the other three methods.

To observe approximation errors, we need an exact RMF of the spine curve or an approximate RMF of very high accuracy against which the computed approximate RMF by the four methods can be compared. Since the exact RMF of the torus knot given by Eqn.(15) is difficult to obtain, we use the integration function provided in Maple to get an approximate RMF of $\mathbf{x}(u)$ whose approximation error is known to be less than 10^{-16}. This highly accurate RMF is used in place of an exact RMF to measure the global approximation error E_g defined in (1).

The global approximation errors e_k of the four methods are shown in Figure 8 and also in Tables 3 and 4, where e_k is the error of using 2^k segments, $k = 6, 7, \ldots, 11$. These data confirm that the projection method and the rotation method have the second order of global approximation error $\mathcal{O}(h^2)$, and the Runge-Kutta method and the double reflection method have the fourth order of global approximation error $\mathcal{O}(h^4)$. We use the computed sequences of frames

	Double reflection	Runge-Kutta
# of segments	error e_k, ratio e_k/e_{k-1}	error e_k, ratio e_k/e_{k-1}
2^6	5.10E−3, N.A.	3.58E−2, N.A.
2^7	3.24E−4, 0.063577	2.32E−3, 0.064846
2^8	2.03E−5, 0.062776	1.46E−4, 0.062737
2^9	1.27E−6, 0.062571	9.10E−6, 0.062408
2^{10}	7.95E−8, 0.062578	5.68E−7, 0.062422
2^{11}	4.97E−9, 0.062575	3.55E−8, 0.062438

Table 3
Global approximation errors e_k of the double reflection method and by using the 4-th order Runge-Kutta method for the torus knot in Example 1. The error ratios e_k/e_{k-1} show that the approximation orders of these two methods are both $\mathcal{O}(h^4)$.

by the four methods to generate ribbon-like sweep surfaces with the torus knot $\mathbf{x}(u)$ as the spine curve, and show the four surfaces in Figures 17 through 20 using color coding to indicate the magnitude of the approximation errors.

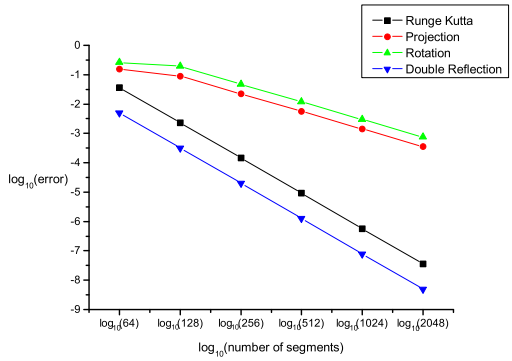


Fig. 8. Global errors of the four methods for the torus knot in Example 1.

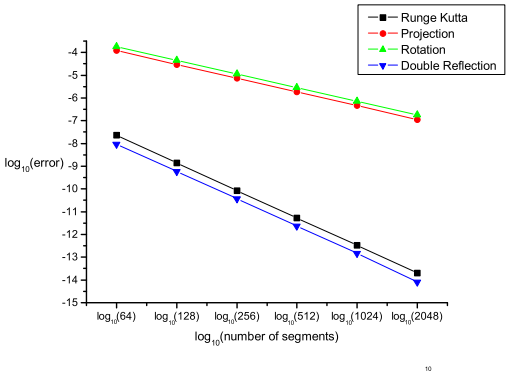


Fig. 9. Global errors of the four methods for the PH curve in Example 2.

Example 2: In the second example we use the double reflection method to approximate the RMF of a PH (Pythagorean-hodograph) curve, whose RMF can be computed exactly by a closed-form formula [11]. Given two points $\mathbf{x}_0 = (1000, 0, 0)^T$ and $\mathbf{x}_1 = (1000, 2000, 4000)^T$ with associated un-normalized tangent vectors $\hat{\mathbf{t}}_0 = (1, 5, -1)^T, \hat{\mathbf{t}}_1 = (-3, 2, 5)^T$, we obtain a cubic PH curve $\mathbf{x}(u)$ as the spine curve using G^1 Hermite interpolation, following [19]. Let the Frenet frame of $\mathbf{x}(u)$ at $u = 0$ be the initial frame U_0. Compared with the exact RMF of $\mathbf{x}(u)$ at the endpoint $\mathbf{x}_1 = \mathbf{x}(1)$, we obtain the errors of the approximate RMF computed by the four methods. These errors are shown in Figure 9. The errors of the double reflection method and the rotation method are also given in Table 5 and their color coded surface representations in

	Projection method	Rotation method
# of segments	error e_k, ratio e_k/e_{k-1}	error e_k, ratio e_k/e_{k-1}
2^6	1.56E−1, N.A.	2.60E−1, N.A.
2^7	9.03E−2, 0.579295	1.91E−1, 0.736606
2^8	2.26E−2, 0.249757	4.76E−2, 0.248776
2^9	5.64E−3, 0.249939	1.19E−2, 0.249668
2^{10}	1.41E−3, 0.249983	2.97E−3, 0.249906
2^{11}	3.52E−4, 0.249995	7.42E−4, 0.249971

Table 4
Global approximation errors e_k of the projection method and the rotation method for the torus knot in Example 1. The error ratios e_k/e_{k-1} show that the approximation orders of these two methods are both $\mathcal{O}(h^2)$.

	Double reflection	Rotation method
# of segments	error e_k, ratio e_k/e_{k-1}	error e_k, ratio e_k/e_{k-1}
2^6	9.29E−9, N.A.	1.78E−4, N.A.
2^7	5.94E−10, 0.063919	4.47E−5, 0.250721
2^8	3.75E−11, 0.063181	1.12E−5, 0.250321
2^9	2.36E−12, 0.062926	2.80E−6, 0.250151
2^{10}	1.48E−13, 0.062789	7.00E−7, 0.250073
2^{11}	9.25E−15, 0.062521	1.75E−7, 0.250036

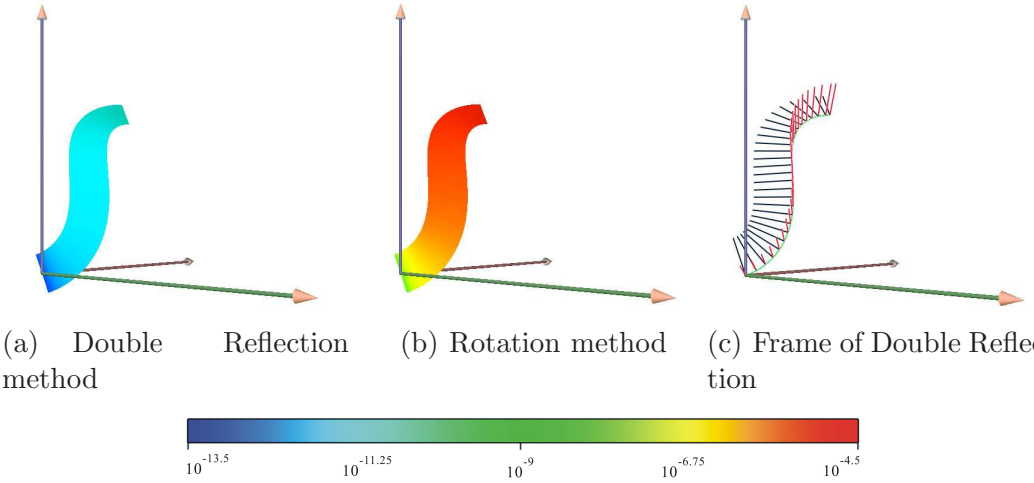
Table 5
Global approximation errors e_k of the double reflection method and the rotation method for the PH curve. The error ratios e_k/e_{k-1} confirm again the $\mathcal{O}(h^4)$ approximation order of the double reflection method and the $\mathcal{O}(h^2)$ approximation order of the rotation method.

Figure 10. These data confirm again the fourth order approximation error $\mathcal{O}(h^4)$ of the double reflection method.

6 Extensions

6.1 Spine curve defined by a sequence of points

In some applications a spine curve is specified by a sequence of points $\mathbf{x}_i$ in 3D, which we may assume to lie on some unknown regularly parameterized spine



(a) Double Reflection method (b) Rotation method (c) Frame of Double Reflection

(d) Error coding bar

Fig. 10. The color coded sweep surfaces showing the errors of the double reflection method and the rotation method for the PH curve in Example 2, with 256 segments.

curve, and we need to compute a sequence of frames U_i which has minimal rotation about the spine curve. In order to apply the double reflection method in this case, we need to furnish each data point $\mathbf{x}_i$ with a unit tangent vector $\mathbf{t}_i$.

One possible way is to define $\mathbf{t}_i$ to be unit tangent vector at $\mathbf{x}_i$ to the circle passing through the three consecutive points $\mathbf{x}_{i-1}$, $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$. Denote $\mathbf{a} = \mathbf{x}_i - \mathbf{x}_{i-1}$ and $\mathbf{b} = \mathbf{x}_{i+1} - \mathbf{x}_i$. It is straightforward to show that $\mathbf{t}_i = \mathbf{w}/\|\mathbf{w}\|$, where $\mathbf{w} = \|\mathbf{b}\|\mathbf{a} + \|\mathbf{a}\|\mathbf{b}$. The tangent vectors $\mathbf{t}_i$ thus defined possess the *co-sphere* property that $\mathbf{t}_i$ and $\mathbf{t}_{i+1}$ at the points $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ are tangential to the sphere S determined by the four consecutive points $\mathbf{x}_{i-1}$, $\mathbf{x}_i$, $\mathbf{x}_{i+1}$ and $\mathbf{x}_{i+2}$. This co-sphere property appears desirable because, by Theorem 4.3, the high accuracy of the double reflection method comes from the very same idea of computing the exact RMF of a spherical curve on a sphere determined by the data $(\mathbf{x}_i, \mathbf{t}_i; \mathbf{x}_{i+1}, \mathbf{t}_{i+1})$.

However, assuming that the $\mathbf{x}_i$ are sampled from an underlying regular curve $\mathbf{x}(u)$ with step size h, it can be shown that the global approximation error of this scheme has the order $\mathcal{O}(h^2)$; this loss of accuracy is due that the above estimated tangent vectors t_i are only $\mathcal{O}(h^2)$ approximation to the exact unit tangent of $\mathbf{x}(u)$ at $\mathbf{x}_i$. So, an alternative is to first obtain better estimates of the unit tangent vectors t_i by using more sample points around x_i.

6.2 Using only tangent vectors

According to its defining equation (3), the RMF of a spine curve $\mathbf{x}(u)$ is entirely determined by the unit tangent vector $\mathbf{t}(u)$. Thus it is natural to consider computing the RMF of $\mathbf{x}(u)$ using only the sampled tangent vector $\mathbf{t}_i = \dot{\mathbf{x}}(u_i)$. From a practical point of view, this treatment is also desirable when the points $\mathbf{x}(u_i)$ are overly densely sampled, which may make the first reflection vector $\mathbf{v}_1 = \mathbf{x}_{i+1} - \mathbf{x}_i$ too small and therefore computation of the reflection R_1 unstable.

In order to apply the double reflection method in this case, all we need to do is provide a reflection vector for the first reflection R_1. Our analysis shows that the global approximation order $\mathcal{O}(h^4)$ to the true RMF of $\mathbf{x}(u)$ is preserved if the first reflection vector is chosen to be

$$\mathbf{v}_1 = 13(\mathbf{t}_i + \mathbf{t}_{i+1}) - (\mathbf{t}_{i-1} + \mathbf{t}_{i+2}). \tag{16}$$

Then the remaining steps of the double reflection method are the same. This assertion can be proved in a similar way to that of proving Theorem 4.6. Note that the computation of $\mathbf{v}_1$ in Eqn. (16) does not involve subtraction between two close quantities, and therefore is numerically robust. Note, however, a different treatment is needed to compute $\mathbf{v}_0$ and $\mathbf{v}_{n-1}$, such that an order $\mathcal{O}(h^4)$ approximations to $\mathbf{x}_1 - \mathbf{x}_0$ and $\mathbf{x}_n - \mathbf{x}_{n-1}$ are achieved. We skip the details here.

6.3 Variational principles for RMF with boundary conditions

In general, the RMF of a closed smooth spine curve does not form a closed moving frame. Therefore, when a closed moving frame with least rotation is needed, it can be generated by adding a gradual rotation to the RMF along the closed spine curve to make the resulting moving frame closed. Even for an open spine curve, it is often required that its moving frame meet given end conditions while having a natural distribution of rotation along the spine curve. So an appropriate additional rotation to the RMF needs to be computed in this case. We study in this section how this additional rotation can properly be determined.

More specifically, consider a curve segment $\mathbf{x}(s)$, $s \in [\,0, L\,]$, in arc-length parameterization. We would like to find a one-parameter family of unit vectors $\mathbf{g}(s)$ orthogonal to the tangent vector $\mathbf{t}(s)$ and satisfying the boundary conditions

$$\mathbf{g}(0) = \mathbf{g}_0 \;\text{ and }\; \mathbf{g}(L) = \mathbf{g}_1 \tag{17}$$

The vector $\mathbf{g}(s)$ defines an orthonormal frame $M = (\mathbf{t}, \mathbf{g}, \mathbf{t} \times \mathbf{g})$ along the spine curve.

We compare the frame M with the RMF generated by a vector $\mathbf{r}(s)$ satisfying $\mathbf{r}(0) = \mathbf{g}(0)$. Let $\alpha(s) = \angle(\mathbf{f}(s), \mathbf{g}(s))$ be the angle between the two frames, where the sign is chosen such that it corresponds to a rotation around the oriented line determined by the tangent vector $\mathbf{t}(s)$. In addition, assume that $\alpha(s)$ is continuous and satisfies $\alpha(0) = 0$. We will call $M(s)$ the *modified frame*, since it is obtained by adding a rotation of angle $\alpha(s)$ to the RMF. In this sense the RMF serves as a reference frame with respect to which another moving frame is specified.

The boundary conditions (17) imply that

$$\alpha(0) = 0 \quad \text{and} \quad \alpha(L) = \angle(\mathbf{f}(L), \mathbf{g}_1) + 2k\pi \tag{18}$$

for a some fixed integer k. The angular velocity vector of the modified frame $M(s)$ is

$$\omega_{\text{modified}}(s) = \kappa(s)\mathbf{b}(s) + \alpha'(s)\mathbf{t}(s). \tag{19}$$

The function $s \mapsto \alpha'(s)$ specifies the angular speed of the rotation of $M(s)$ around the tangent of the curve $\mathbf{x}(u)$. We now consider two possible ways of choosing $\alpha(s)$.

Minimum total angular speed One may choose $\alpha(s)$ that minimizes the functional

$$\int_0^L ||\omega_{\text{modified}}|| \, \mathrm{d}s = \int_0^L \sqrt{\kappa(s)^2 + \alpha'(s)^2} \, \mathrm{d}s \quad \to \quad \text{Min} \tag{20}$$

and satisfies the boundary conditions (18). Let $F(s, \alpha, \alpha') = \sqrt{\kappa^2 + \alpha'^2}$. Then we have at hand a functional of the angular function $\alpha(s)$. The moving frame $M(s)$ corresponds to a curve on the unit quaternion sphere, and minimizing the functional in (20) amounts to minimizing the length of this curve subject to that $\mathbf{g}(s)$ is perpendicular to $\mathbf{t}(s)$; this is the computational approach taken in [14].

Here we will analyze this variational problem to give it a simple geometric interpretation as well as an easy computational method. Solving Euler's equation of the functional (20) using calculus of variations yields

$$0 = F_\alpha - \frac{\mathrm{d}}{\mathrm{d}s} F_{\alpha'} = -\frac{\kappa}{(\kappa^2 + \alpha'^2)^{3/2}}(\kappa\alpha'' - \alpha'\kappa') = -\frac{\kappa^3}{(\kappa^2 + \alpha'^2)^{3/2}} \left(\frac{\alpha'}{\kappa}\right)', \tag{21}$$

assuming $\kappa \neq 0$. It follows that

$$\alpha'(s) = C\kappa(s) \tag{22}$$

for some constant C, which can be determined from the boundary conditions and the total curvature. *Consequently, the angular speed of the additional rotation around the tangent is proportional to the curvature of the curve.* Hence, minimizing (20) makes the additional rotation more concentrated on curve segments of higher curvatures.

The above analysis is only valid for curved segments with $\kappa(s) \not\equiv 0$. For straight line segments, the variational problem (20) does not have a unique solution. In fact, the integrand in this case simplifies to $|\alpha'|$, and any monotonic function $\alpha(s)$ which satisfies the boundary conditions is a solution. Because of this non-uniqueness of solution, optimization methods as used in [14] for minimizing (20) will experience numerical problems with a spine curve that is close to a straight line. Based on our analysis, a more efficient method is to compute the curvatures at sampled points of the spine curve, and then distribute the additional rotation proportional to the curvatures along the curve, with respect to the RMF.

Minimum total squared angular speed One may also choose $\alpha(s)$ that minimizes

$$\int_0^L ||\omega_{\text{modified}}||^2 \mathrm{d}s = \int_0^L (\kappa(s)^2 + \alpha'(s)^2)\mathrm{d}s \quad \rightarrow \quad \text{Min} \tag{23}$$

and satisfies the boundary conditions (18). Now, with $F = \kappa^2 + \alpha'^2$, Euler's equation gives $\alpha'' = 0$, or $\alpha(s) = as$ for some constant a; that is, *the rotation of M is linearly proportional to the arc length parameter s.*

This choice of the additional rotation is not only easy to implement, but also free of the numerical problem with the method based on minimizing (20); so it is recommended over the first one based on minimizing the total angular speed. Note that this means of applying the additional rotation as proportional to arc-length has been suggested in the literature (e.g. [6,32]), but here we have provided a theoretical justification from the viewpoint of the variational principle through minimization of the total squared angular speed.

Efficient implementation of the above methods of computing a moving frame with boundary conditions is based on angle adjustment to the RMF, either according to curvature or arclength. When the RMF is computed approximately, the resulting solution is only an approximate one. In this regard, the higher accuracy of the double reflection method makes this solution more accurate than using the projection method or the rotation method.

One may choose the integer k in (18) to minimize the rotation if the least deviation to the RMF is desired, or choose k to add a moving frame with a specified amount of total twist along the spine curve. Figure 11 shows compar-

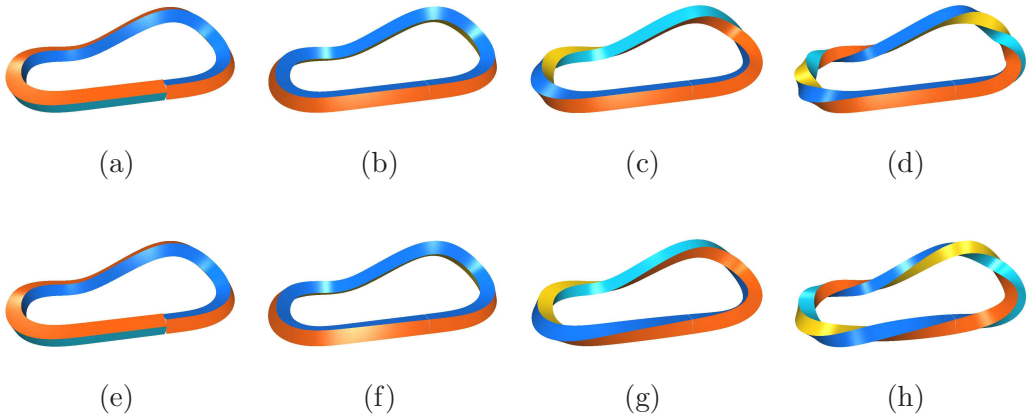


Fig. 11. Comparison in computing a closed moving frame. Minimization of total angular speed is shown in row one. Minimization of total *squared* minimization is shown in row two. In each row, from left to right, the four figures are for the case of RMF computed by the double reflection method, the case of using the minimal twist to close the frame, the case of an additional twist of 2π, and the case of an additional twist of 4π.

ison of the two methods above for computing frames meeting certain boundary conditions. The method based on total angular speed minimization (i.e., rotation proportional to curvature) and the method based in total squared angular speed minimization (i.e., rotation proportional to arclength) are shown in the first row and the second row, respectively. In each row, the four figures are for the case of using RMF computed by the double reflection method with no twist adjustment, the case of using the minimal twist to close the frame, the case of a twist of 2π, and the case of a twist of 4π. We see that the twist is more concentrated in high curvature parts of the spine curve in the first row, while it is distributed more uniformly along the curve in the second row.

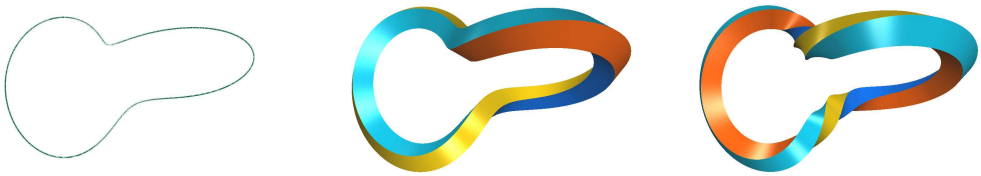
A closed moving frame is useful in visualization of closed space curve, such as knots. Figures 12 and 13 show two such examples, where the closed frame is computed by adjusting an RMF by an additional rotation linearly proportional to the arclength. The curve in Figure 12, a cinquefoil knot, is given by

$$\mathbf{x}(t) = [\cos(t)(2-\cos(2t/(2a^2+1))), \sin(t)(2-\cos(2t/(2a^2+1))), -\sin(2t/(2a^2+1))] \tag{24}$$

The curve in Figures 13, a trefoil knot, is given by

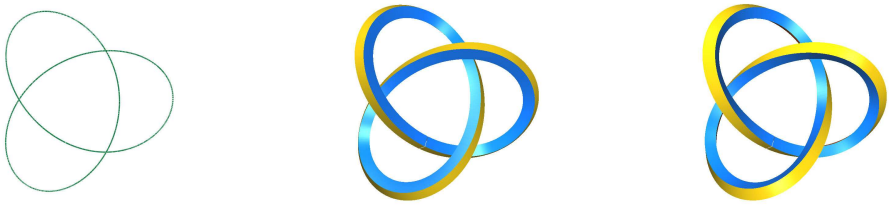
$$\begin{aligned}\mathbf{x}(t) = [&41\cos(t) - 18\sin(t) - 83\cos(2t) - 83\sin(2t) - 11\cos(3t) + 27\sin(3t),\\ &36\cos(t) + 27\sin(t) - 113\cos(2t) + 30\sin(2t) + 11\cos(3t) - 27\sin(3t),\\ &45\sin(t) - 30\cos(2t) + 113\sin(2t) - 11\cos(3t) + 27\sin(3t)]\end{aligned} \tag{25}$$

Note that in these two examples the Frenet frame exhibits noticeable rotation about the curves to be visualized.



(a) (b) (c)

Fig. 12. (a) A cinquefoil knot, given by Eqn. (24), is shown without visualization cue; (b) Visualization of the curve $\mathbf{x}(t)$ is enhanced by a closed sweep surface modeled using an adjusted RMF; (c) another sweep surface modeled using the Frenet frame.



(a) (b) (c)

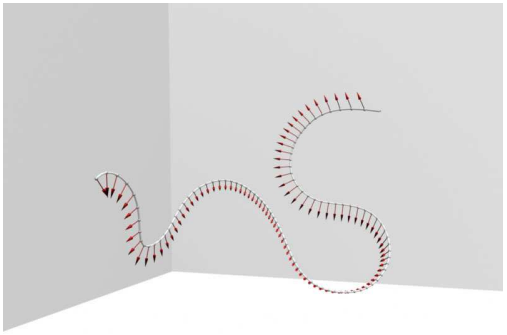
Fig. 13. (a) A trefoil knot, given by Eqn. (25), is shown without visualization cue; (b) Visualization of the curve $\mathbf{x}(t)$ is enhanced by a closed sweep surface modeled using an adjusted RMF; (c) another sweep surface modeled using the Frenet frame.

We next give two more examples of RMF based moving frame design with boundary conditions in shape modeling. In Figure 14, the main body of a dragon along a spine curve is modeled with a moving frame meeting user specified boundary conditions. The frame is computed using arclength twist adjustment of the approximate RMF computed by the double reflection method.

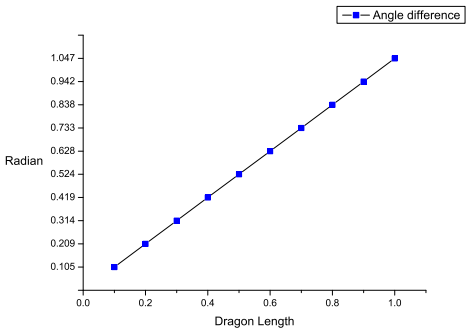
In Figure 15, the support structure of a glass table, as a closed sweep surface, is modeled with a moving frame meeting six conditions to make the surface have proper contact (i.e., along a line segment) with the table at three locations and with the ground at three locations. These conditions are met by adjusting an RMF by a twist linearly proportional to arclength between every two consecutive contact locations.

7 Concluding remarks

We have presented a new discrete approximation method for computing the rotation minimizing frame of a space curve. The method uses two reflections in a plane to compute the next frame from the current frame, and is therefore called the *double reflection method*. This method is simple, fast, and more accurate than the projection method and the rotation method, that are cur-



(a) RMF based moving frame by boundary condition.



(b) Angle difference between the RMF and the frame in (a).

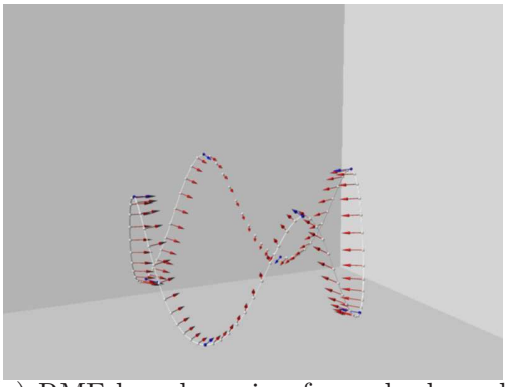
(c) A dragon modeled with the moving frame in (a).

Fig. 14. Modeling of an oriental dragon.

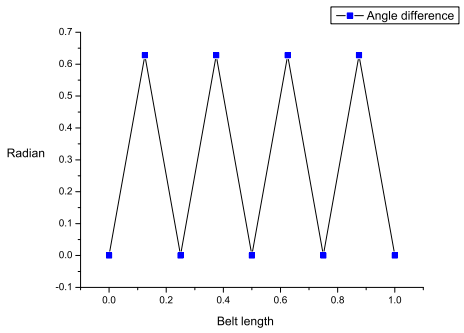
rently often used in practice. We have shown that the approximation error of the double reflection method is $\mathcal{O}(h^4)$, while the errors of the other two methods are $\mathcal{O}(h^2)$, where h is the step size used to sample points on a spine curve of fixed length.

The double reflection method is also much superior to direct application of the standard 4-th order Runge-Kutta method. Although the two methods have the same order of approximation error, the double reflection method is simpler and faster, and requires only C^1 information of a spine curve, while the Runge-Kutta method needs C^2 information. We have also discussed the applications of RMF in modeling moving frames meeting boundary conditions.

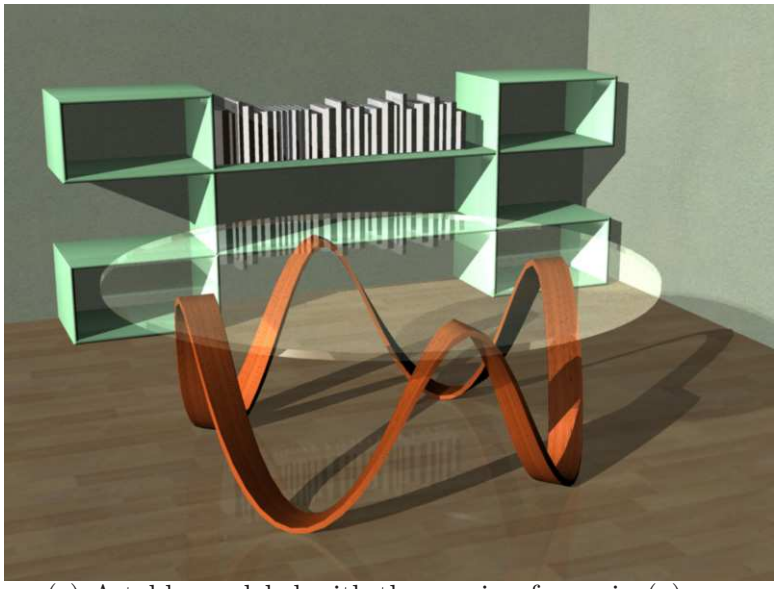
We conjecture that $\mathcal{O}(h^4)$ is the maximum accuracy that can be achieved in RMF computation using only the sampled positional and tangent data



(a) RMF based moving frame by boundary condition.



(b) Angle difference between the RMF and the frame in (a).



(c) A table modeled with the moving frame in (a).

Fig. 15. An RMF based moving frame is used to design the supporting structure of a glass table as a sweep surfaces

$(\mathbf{x}_0, \mathbf{t}_0; \mathbf{x}_1, \mathbf{t}_1)$ of a curve segment.

References

[1] D. C. Banks and B. A. Singer. A predictor-corrector technique for visualizing unsteady flows. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):151–163, 1995.

[2] R. Barzel. Faking dynamics of ropes and springs. *IEEE Computer Graphics and Applications*, 17(3):31–39, 1997.

[3] D. Bechmann and D. Gerber. Arbitrary shaped deformation with dogme. *The Visual Computer*, 19(2-3):175–186, 2003.

[4] R. L. Bishop. There is more than one way to frame a curve. *American Mathematics Monthly*, 82(3):246–251, 1975.

[5] J. Bloomenthal. Modeling the mighty maple. In *Proceedings of SIGGRAPH 1985*, pages 305–311, 1985.

[6] J. Bloomenthal. *Caculation of reference frames along a space curve*. 1990.

[7] M. Bloomenthal and R. F. Riesenfeld. Approximation of sweep surfaces by tensor product NURBS. In *SPIE Proceedings: Curves and Surfaces in Computer Vision and Graphics II*, volume 1610, pages 132–154, 1991.

[8] W. F. Broonsvoort and F. Flok. Ray tracing generalized cylinders. *ACM Transactions on Graphics*, 4(4):291–302, 1985.

[9] H. I. Choi, S-H Kwon, and N-S. Wee. Almost rotation-minimizing rational parametrization of canal surfaces. *Gomputer Aided Geometric Design*, 21(9):859–881, 2004.

[10] T. L. Chung and W. Wang. Discrete moving frames for sweep surface modeling. In *Proceedings of Pacific Graphics'96*, pages 159–173, 1996.

[11] R. Farouki. Exact rotation-minimizing frames for spatial Pythagorean-hodograph curves. *Graphical Models*, 64:382–395, 2002.

[12] R. Farouki and C. Y. Han. Rational approximation schemes for rotation-minimizing frames on Pythagorean-hodograph curves. *Computer Aided Geometric Design*, 20(7):435–454, 2003.

[13] H. W. Guggenheimer. Computing frames along a trajectory. *Gomputer Aided Geometric Design*, 6:77–78, 1989.

[14] A. Hanson. Constrained optimal framing of curves and surfaces using quaternion gauss map. In *Proceedings of Visulization'98*, pages 375–382, 1998.

[15] A. Hanson. *Visualizing Quaternions*. Morgan Kaufmann, 2005.

[16] A. J. Hanson and H. Ma. A quaternion approach to streamline visualization. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):164–174, 1995.

[17] B. Jüttler. Rotational minimizing spherical motions. In *Advacnes in Robotics: Analysis and Control*, pages 413–422. Kluwer Dordrecht, 1998.

[18] B. Jüttler. Rational approximation of rotation minimizing frames using Pythagorean-hodograph cubics. *Journal of Geometry and Graphics*, 3:141–159, 1999.

[19] B. Jüttler and C. Mäurer. Cubic Pythagorean hodograph spline curves and applications to sweep surface modeling. *Computer-aided Design*, 31:73–83, 1999.

[20] B. Jüttler, M. Peternell, and W. Wang. Möbius invariant frames of space curves. in preparation, 20xx.

[21] F. Klok. Two moving frames for sweeping along a 3D trajectory. *Computer Aided Geometric Design*, 3(1):217–229, 1986.

[22] E. Kreyszig. *Differential Geometry*. Dover, 1991.

[23] F. Lazarus, S. Coquillart, and P. Jancène. Interactive axial deformations. In *Modeling in Computer Graphics*, pages 241–254. Springer Verlag, 1993.

[24] F. Lazarus and A. Verroust. Feature-based shape transformation for polyhedral objects. In *Proceedings of the 5th Eurographics Workshop on Animation and Simulation*, pages 1–14, 1994.

[25] S. Coquillart Lazarus and P. Jancene. Axial deformation: an intuitive technique. *Computer-Aided Design*, 26(8):607–613, 1994.

[26] Q. Peng, X. Jin, and J. Feng. Arc-length-based axial deformation and length preserving deformation. In *Proceedings of Computer Animation 1997*, pages 86–92, 1997.

[27] T. Poston, S. Fang, and W. Lawton. Computing and approximating sweeping surfaces based on rotation minimizing frames. In *Proceedings of the 4-th International Conference on CAD/CG, Wuhan, China*, 1995.

[28] H. Pottmann and M. Wagner. Contributions to motion based surface design. *International Journal of Shape Modeling*, 4(3&4):183–196, 1998.

[29] S. K. Semwal and J. Hallauer. Biomedical modeling: implementing line-of-action algorithm for human muscles and bones using generalized cylinders. *Computers and Graphics*, 18(1):105–112, 1994.

[30] U. Shani and D. H. Ballard. Splines as embeddings for generalized cylinders. *Computer Vision, Graphics, and Image Processing*, 27:129–156, 1984.

[31] P. Siltanen and C. Woodward. Normal orientation methods for 3D offset curves, sweep surfaces, skinning. In *Proceedings of Eurographics'92*, pages 449–457, 1992.

[32] W. Wang and B. Joe. Robust computation of rotation minimizing frame for sweep surface modeling. *Gomputer-Aided Design*, 29:379–391, 1997.

8 Appendix I

Proof of Theorem 4.4. There are two parts in this proof. In the first part we derive an expression of the order $\mathcal{O}(h^5)$ term of the one-step error. In the second part we show that coefficient of this error term is bounded for a regular curve, thus yielding the claimed order of magnitude.

We will obtain the error expression using the canonical Taylor expansion of the curve $\mathbf{x}(s)$ at $\mathbf{x}(0)$, which can be derived from the Frenet formulas [22]. In

a neighborhood of $\mathbf{x}(0)$, $\mathbf{x}(s)$ is approximated by the series

$$\mathbf{x}(s) = \begin{pmatrix} s & -\frac{1}{6}\kappa_0^2 s^3 & -\frac{1}{8}\kappa_0\kappa_1 s^4 + \cdots \\ \frac{1}{2}\kappa_0 s^2 & +\frac{1}{6}\kappa_1 s^3 & +\frac{1}{24}(\kappa_2 - \kappa_0^3 - \tau_0^2\kappa_0)s^4 + \cdots \\ & +\frac{1}{6}\kappa_0\tau_0 s^3 & +\frac{1}{24}(\kappa_0\tau_1 + 2\kappa_1\tau_0)s^4 + \cdots \end{pmatrix}, \tag{26}$$

where the Frenet frame at $s = 0$ is aligned with the axes of the Cartesian coordinates, and $\kappa_i = (\mathrm{d}/\mathrm{d}s)^i\kappa(s)|_{s=0}$, $\tau_i = (\mathrm{d}/\mathrm{d}s)^i\tau(s)|_{s=0}$. With the help of computer algebra tools, we generate Taylor series for all quantities needed for computing the variables listed in the procedure of the double reflection method (Table 1). Due to space limitation, only an outline of the derivation will be given.

Consider a segment of $\mathbf{x}(s)$ of length h starting at the origin, i.e.,

$$(0,0,0)^\top = \mathbf{x}_0 = \mathbf{x}(0),\ \mathbf{x}_1 = \mathbf{x}(h),\ (1,0,0)^\top = \mathbf{t}_0 = \dot{\mathbf{x}}(0),\ \mathbf{t}_1 = \dot{\mathbf{x}}(h). \tag{27}$$

Let $\mathbf{r}_0 = (0, C, S)$, where $C^2 + S^2 = 1$, be the reference vector of U_0 at $\mathbf{x}_0$. We compute the new reference vector $\mathbf{r}_1$ using steps from (1) to (7) of the algorithm *Double Reflection* (see Table 1):

$$\begin{aligned}
&\mathbf{v}_1 = (h + \mathcal{O}(h^3),\ \tfrac{1}{2}\kappa_0 h^2 + \mathcal{O}(h^3),\ \mathcal{O}(h^3))^\top \\
&c_1 = h^2 - \tfrac{1}{12}{\kappa_0}^2 h^4 + \mathcal{O}(h^5) \\
&\mathbf{r}_0^L = (-C\kappa_0 h - \tfrac{1}{3}(C\kappa_1 + \kappa_0\tau_0 S)h^2 + \mathcal{O}(h^3),\ C - \tfrac{1}{2}{\kappa_0}^2 C h^2 + \mathcal{O}(h^3), S + \mathcal{O}(h^3))^\top \\
&\mathbf{t}_0^L = (-1 + \tfrac{1}{2}{\kappa_0}^2 h^2 + \mathcal{O}(h^3), -\kappa_0 h - \tfrac{1}{3}\kappa_1 h^2 + \mathcal{O}(h^3), -\tfrac{1}{3}\kappa_0\tau_0 h^2 + \mathcal{O}(h^3))^\top \\
&\mathbf{v}_2 = (2 - {\kappa_0}^2 h^2 + \mathcal{O}(h^3), 2\kappa_0 h + \tfrac{5}{6}\kappa_1 h^2 + \mathcal{O}(h^3), \tfrac{5}{6}\kappa_0\tau_0 h^2 + \mathcal{O}(h^3))^\top \\
&c_2 = 4 - \tfrac{1}{36}({\tau_0}^2{\kappa_0}^2 + {\kappa_1}^2)h^4 + \mathcal{O}(h^5) \\
&\mathbf{r}_1 = (-C\kappa_0 h - \tfrac{1}{2}(C\kappa_1 + \kappa_0\tau_0 S)h^2 + \mathcal{O}(h^3), C - \tfrac{1}{2}{\kappa_0}^2 C h^2 + \mathcal{O}(h^3), S + \mathcal{O}(h^3))^\top
\end{aligned}$$

On the other hand, using the angular velocity of the RMF (Eqn. (9)) we generate the Taylor expansion of the reference vector $\mathbf{r}(h)$ of the exact RMF $U(h)$,

$$\mathbf{r}(h) = \mathbf{r}(s)\Big|_{s=0} + \underbrace{\kappa(s)\mathbf{b}(s)\times\mathbf{r}(s)}_{=\mathbf{r}'(0)}\Big|_{s=0} h + \underbrace{\frac{\mathrm{d}}{\mathrm{d}s}(\kappa(s)\mathbf{b}(s)\times\mathbf{r}(s))}_{=\mathbf{r}''(0)}\Big|_{s=0} \frac{h^2}{2} + \ldots$$

Using the Frenet formulas and the fact that the derivatives of $\mathbf{r}(s)$ are given by the previously generated terms of the Taylor expansion, $\mathbf{r}(h)$ can be expressed solely by using derivatives of curvature and torsion at $s = 0$, and by the initial value $\mathbf{r}(0) = (0, C, S)^\top$. Finally, we compare the Taylor expansions of $\mathbf{r}(h)$ and

$\mathbf{r}_1$ to obtain

$$\mathbf{r}(h) - \mathbf{r}_1 = (\mathcal{O}(h^6), -\tfrac{1}{720}\, S\, K\, h^5 + \mathcal{O}(h^6), \tfrac{1}{720}\, C\, K\, h^5 + \mathcal{O}(h^6))^\top,$$

where

$$K = 2\,{\kappa_1}^2\tau_0 + {\kappa_0}^2{\tau_0}^3 + \kappa_1\kappa_0\tau_1 - \kappa_2\kappa_0\tau_0 \tag{28}$$

Hence,

$$\|\mathbf{r}(h) - \mathbf{r}_1\| = \frac{1}{720} K h^5 + \mathcal{O}(h^6)$$

Next, we need to show that the coefficient K in the $\mathcal{O}(h^5)$ term above is finite for a regular smooth curve. This is a concern because the torsion τ_0 appearing in K (Eqn. (28)) and τ_0 can become unbounded for a regular curve; such an example is given in Appendix II. Note that only the curvature κ_0, torsion τ_0 and their derivatives are present in K. Since

$$\kappa(s) = \|\ddot{\mathbf{x}}(s)\|, \quad \tau(0) = \frac{\|(\dot{\mathbf{x}}(s) \times \ddot{\mathbf{x}}(s)) \cdot \dddot{\mathbf{x}}(s)\|}{\|\ddot{\mathbf{x}}(s)\|^3}$$

it is easy to see that, if a spine curve has non-vanishing curvature, then $\kappa_0 = \kappa(0)$ is bounded from zero, and $\tau_0 = \tau(0)$ and its derivative are finite; consequently, K will be finite in this case.

We will use a conformal mapping to turn an arbitrary curve segment $\mathbf{x}(s)$, $s \in [0, h]$, possibly with vanishing curvature, into another curve segment with curvature bounded from zero. First take the osculating plane of $\mathbf{x}(s)$ at $s = 0$. With a rigid motion we take this plane to be the x-y plane and have the point $\mathbf{x}(0)$ positioned at the origin $(0, 0, 0)$. Let $\mathcal{C}_s$ denote the inversion with respect to the sphere S_1 of radius 1 and centered at $(0, 0, 1)$. Then the plane x-y is mapped by $\mathcal{C}_s$ to the sphere S_2 of radius $1/2$ and centered at $(0, 0, 1/2)$. Clearly, $\mathcal{C}_s$ is conformal, and the point $\mathbf{x}(0) = (0, 0, 0)$ is fixed by $\mathcal{C}_s$.

Let κ_0 be the curvature of $\mathbf{x}(s)$ at $s = 0$. Let $\mathbf{x}_c(s)$ denote the transformed curve $\mathcal{C}_s(\mathbf{x}(s))$. With a bit abuse of notation, we use $\mathbf{x}_c(t)$, $t \in [0, h_c]$, to denote arclength parameterization of the segment $\mathbf{x}_c(s)$. At $t = 0$, the curve $\mathbf{x}_c(t)$ has the normal curvature equal to 2, which is the reciprocal of the radius of S_2, and the geodesic curvature equal to κ_0, which is the curvature of $\mathbf{x}(s)$ at $s = 0$. (The curve $\mathbf{x}_c(s)$ has the same normal curvature and geodesic curvature at $\mathbf{x}_c(0)$ as any spherical curve on S_2 that has the second order contact with $\mathbf{x}_c(s)$ at $\mathbf{x}_c(0)$.) It follows that the curvature of $\mathbf{x}_c(u)$ at $\mathbf{x}_c(0)$ is $\kappa_c = (\kappa_0^2 + 4)^{1/2}$.

Clearly, κ_c is bounded away from zero. Hence, if we apply the double reflection method to the transformed curve segment $\mathbf{x}_c(t)$, $t \in [0, h_c]$, according to the preceding analysis, the fifth order term of the approximation error takes the form $\frac{1}{720} K_c h_c^5$; here K_c is finite, since κ_c is bounded away from zero. On the other hand, because the approximation error produced by the double reflection

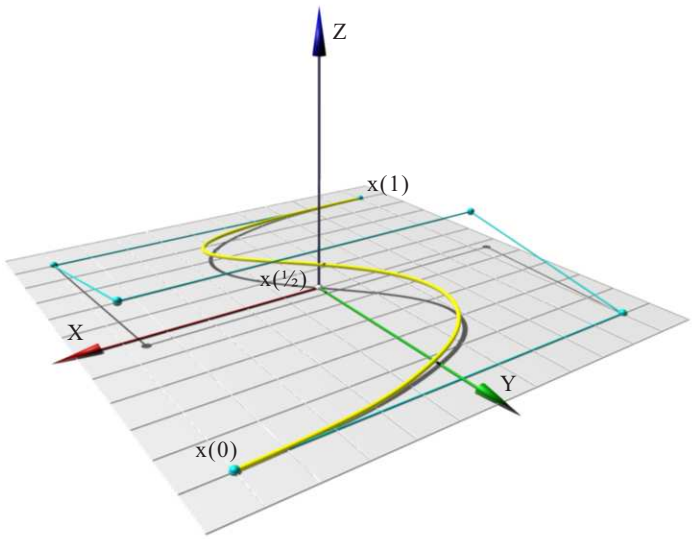


Fig. 16. S-shaped sixth order Bézier curve

method is invariant under a conformal mapping (cf. Section 4.5), in the limit we have

$$\frac{K}{720}h^5 = \frac{K_c}{720}h_c^5$$

When h is sufficiently small, due to the regular nature of the mapping $\mathcal{C}_s$ in the neighborhood of $\mathbf{x}(0)$, there exists a constant $d > 0$ such that $h_c < dh$. It follows that

$$K = \frac{h_c^5}{h^5}K_c < d^5 K_c$$

Hence, K is also finite. This completes the proof that the local one-step error of the double reflection method is of the order of $\mathcal{O}(h^5)$. □

9 Appendix II

It is well known that a curve is planar if and only if its torsion is zero everywhere. However, the torsion has certain peculiar behavior which makes it an "unstable" characteristic of a planar curve. Below we use an example to show that a nearly planar curve can have arbitrarily large torsion. This example also serves two further purposes. First, the numerical integration method in [13] will experience severe difficulty in computing the RMF for the spine curve in this example. Second, although the torsion τ is unbounded in this example, we will see that the constant K in the fifth order error term in Theorem 4.4 is still finite, thus showing that the second part of the proof of Theorem 4.4 is warranted (cf. Appendix I).

Consider the S-shaped sixth order Bézier curve $\mathbf{x}(t; h)$ in 3D defined by the

control points $P_0 = (1,1,0)^T$, $P_1 = (-1,1,0)^T$, $P_2 = (-1,0,h)^T$, $P_3 = (1,0,h)^T$, $P_4 = (1,-1,0)^T$ and $P_5 = (-1,-1,0)^T$. The curve has the parametric equation

$$\mathbf{x}(t;h) = \begin{pmatrix} 8t^5 - 20t^4 + 20t^2 - 10t + 1 \\ 8t^5 - 20t^4 + 20t^3 - 10t^2 + 1 \\ 10t^4 h - 20t^3 h + 10t^2 h \end{pmatrix}, \quad t \in [0,1]$$

We consider the behavior of $\mathbf{x}(t;h)$ as $h \to 0$. When $h = 0$, $\mathbf{x}(t;0)$ becomes planar and $\mathbf{x}(1/2;0) = (0,0,0)$ is an inflection point.

Let us first check $\tau_0, \tau_1, \kappa_0, \kappa_1, \kappa_2$ at the point $\mathbf{x}(1/2;h)$.

$$\tau_0 = -\frac{12}{5h}, \quad \tau_1 = 0, \quad \kappa_0 = \frac{4}{5}|h|, \quad \kappa_1 = 0, \quad \kappa_2 = \frac{192(h^4 - 3h^2 - 3)}{125|h|}$$

Clearly, τ_0 and κ_2 are not bounded as $h \to 0$.

Now we check K at $\mathbf{x}(1/2;h)$. The four terms of the expression of K in Eqn. (28) are

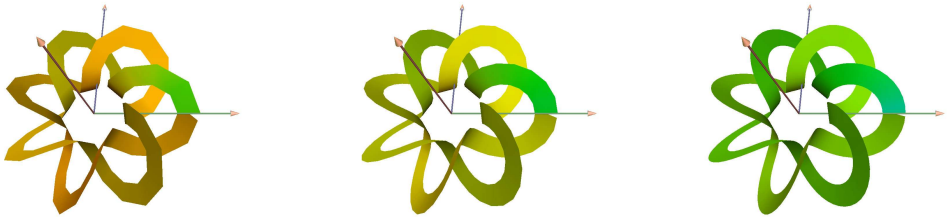
$$2\kappa_1^2\tau_0 = 0, \quad \kappa_0^2\tau_0^3 = -\frac{27648}{3125h}, \quad \kappa_1\kappa_0\tau_1 = 0, \quad \kappa_2\kappa_0\tau_0 = \frac{9216(h^4 - 3h^2 - 3)}{3125h}$$

Then

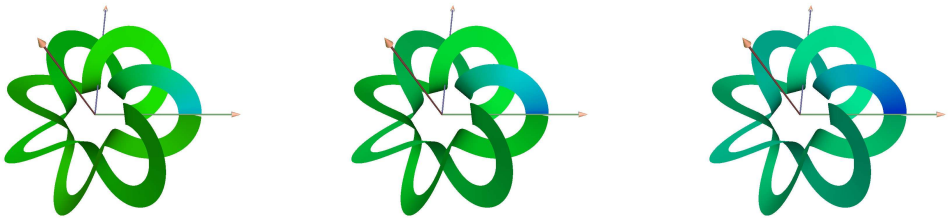
$$K = 2\,{\kappa_1}^2\tau_0 + {\kappa_0}^2{\tau_0}^3 + \kappa_1\kappa_0\tau_1 - \kappa_2\kappa_0\tau_0 = \frac{9216h(3 - h^2)}{3125}$$

and $\lim_{h\to 0} K = 0$. Hence, K is finite for all finite values of h in this example.

In this example, numerical integration of (2) becomes difficult as the integrand $\tau(t)$ becomes unbounded at $t = 1/2$ for arbitrarily small h. Furthermore, even if this integration can be done, the Frenet frame of $\mathbf{x}(t)$ becomes increasingly unstable at $t = 1/2$ as $h \to 0$. All this makes it difficult to apply Eqn. (2) to computing the RMF of $\mathbf{x}(t)$.

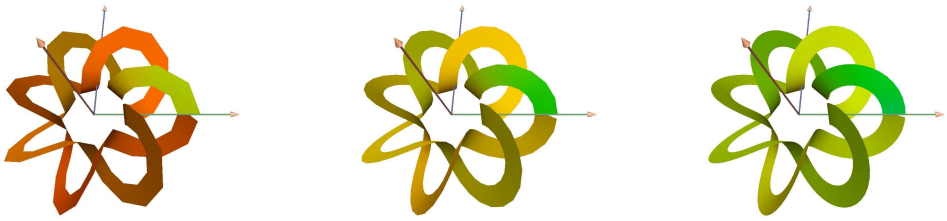


(a) 2^6 segments. (b) 2^7 segments. (c) 2^8 segments.

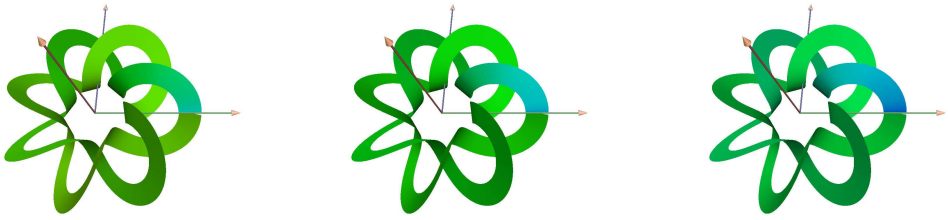


(d) 2^9 segments. (e) 2^{10} segments. (f) 2^{11} segments.

Fig. 17. Color coded sweep surfaces showing the errors of double reflection method

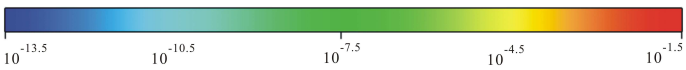


(a) 2^6 segments. (b) 2^7 segments. (c) 2^8 segments.

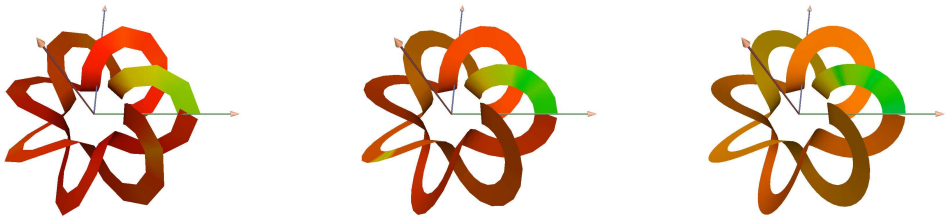


(d) 2^9 segments. (e) 2^{10} segments. (f) 2^{11} segments.

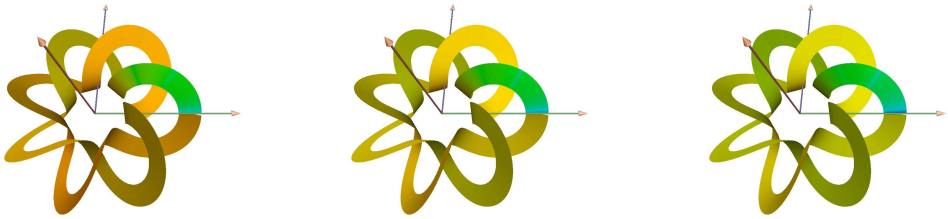
Fig. 18. Color coded sweep surfaces showing the errors of 4-th order Runge-Kutta method



(a) Error coding bar

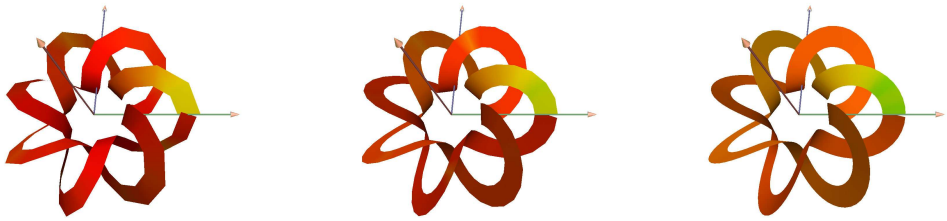


(b) 2^6 segments. (c) 2^7 segments. (d) 2^8 segments.

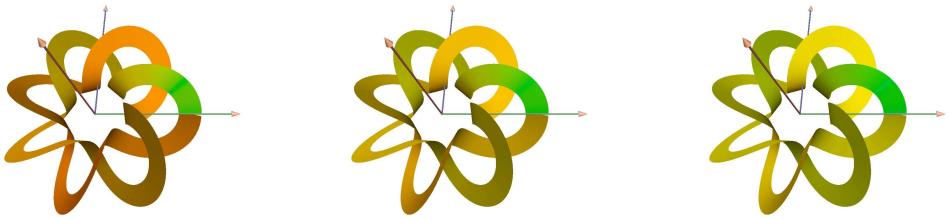


(e) 2^9 segments. (f) 2^{10} segments. (g) 2^{11} segments.

Fig. 19. Color coded sweep surfaces showing the errors of the projection method.

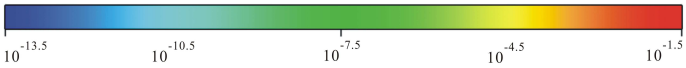


(a) 2^6 segments. (b) 2^7 segments. (c) 2^8 segments.



(d) 2^9 segments. (e) 2^{10} segments. (f) 2^{11} segments.

Fig. 20. Color coded sweep surfaces showing the errors of the rotation method.



(a) Error coding bar