3D ball skinning using PDEs for generation of smooth tubular surfaces

Greg Slabaugh [a,*], Brian Whited [b], Jarek Rossignac [b], Tong Fang [c], Gozde Unal [d]

[a] *Research and Development Department, Medicsight, London, UK*
[b] *School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA, USA*
[c] *Real-Time Vision Department, Siemens Corporate Research, Princeton, NJ, USA*
[d] *Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey*

ARTICLE INFO

ABSTRACT

We present an approach to compute a smooth, interpolating skin of an ordered set of 3D balls. By construction, the skin is constrained to be C^1 continuous, and for each ball, it is tangent to the ball along a circle of contact. Using an energy formulation, we derive differential equations that are designed to minimize the skin's surface area, mean curvature, or convex combination of both. Given an initial skin, we update the skin's parametric representation using the differential equations until convergence occurs. We demonstrate the method's usefulness in generating interpolating skins of balls of different sizes and in various configurations.

1. Introduction

In this paper, we consider the geometric problem of *ball skinning*, which we define to be the computation of a continuous interpolation of a discrete set of balls; an example appears in Fig. 1. This problem arises in numerous applications, including character skinning, molecular surface model generation, and in our primary application, the modeling of tubular structures. The balls are ordered, can have different radii, can be configured in different positions, and may or may not overlap. In our formulation of the problem, we require that the skin contacts each ball along a circle, and is tangent to the ball along this circle of contact. The skin then rests on and interpolates the underlying balls.

For a given configuration of balls, there exist an infinite number of possible solutions to this problem as expressed above. To formulate the problem so that it is well-posed, we seek the skin that has minimal surface area, mean curvature, or combination of both. We achieve this by deriving, and solving, differential equations that minimize an energy, composed of surface area and mean curvature terms, based on this variational problem. By minimizing this energy, the method provides an optimal constrained interpolation of the balls.

1.1. Related work

The problem of skinning appears in various contexts. In computer animation, often an articulated object or character is constructed using a layered representation consisting of a skeletal structure and a corresponding geometric skin [1]. The skeleton has fewer degrees of freedom and is simpler to adjust by an animator. Given a new skeletal pose, the skinning algorithm is responsible for deforming the geometric skin to respond to the motion of the underlying skeleton. The skinning problem is a special case of the problem of computing the envelopes of families of quadrics, which have been investigated by Peternell [2] via the use of cyclographic maps. Rossignac and Schaefer [3] present J-splines, which produce smooth curves from a set of ordered points using a subdivision framework.

The problem of ball skinning appears frequently in the context of computational chemistry and molecular biology, when generating surface meshes for molecular models [4–6]. Several algorithms exist to skin a molecular model to produce a C^1 continuous surface that is tangent smooth and has high mesh quality. These methods are typically either based on Delaunay triangulation [4] or by finding the isosurface of an implicit function [6]. The work of [6] derives a special subset of skins that is piece-wise quadratic. When dealing with a continuous family of balls, the skin may be computed as the envelope of the infinite union of the circles of intersection of two consecutive balls of infinitely close center. While the surfaces generated by these methods are tangent to the balls and have smoothness at the point of tangency, none of these existing methods provides an optimally smooth skin, unlike the method we present here.

In our application, we are interested in modeling the geometry of a blood vessel that has been identified using Pearling [7], a ball packing algorithm that places numerous balls of different radii so that they fit snugly inside an imaged blood vessel. Given these balls, we would like to find a C^1 skin that smoothly interpolates the balls.

* Corresponding author. Tel.: +44 0794 8308316.
E-mail address: greg.slabaugh@gmail.com (G. Slabaugh).
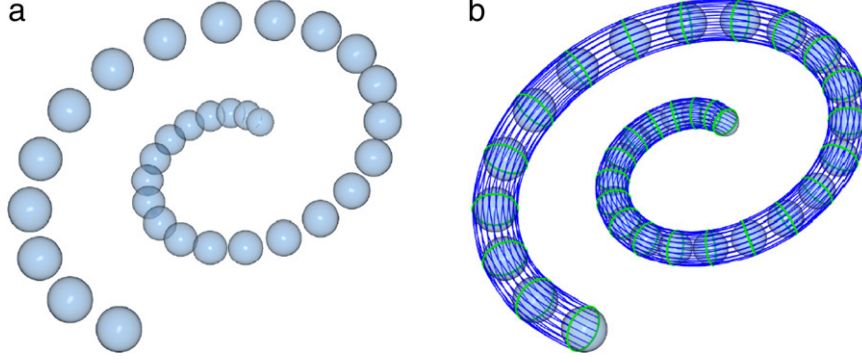*URL:* http://www.gregslabaugh.name (G. Slabaugh).

Fig. 1. An example ball skinning. Given an ordered sequence of balls (a), we produce a skin that optimally interpolates the balls (b). This skin is a surface that consists of splines (rendered in blue) and is computed using differential equations.

This surface can then be used for visualization of the blood vessel, simulation of blood flows using computational fluid dynamics, as well as measurements such as volume or surface area. We note that the problem of 2D ball skinning was addressed in our previous work [8]; in this paper, we extend the methodology to skinning 3D balls. In this case, the problem has a similar conceptual formulation based on differential equations; however, the geometry is notably different: instead of minimizing the arc length and curvature of a curve, we minimize the surface area and mean curvature of a surface. Furthermore, the geometry of contact between the skin and the balls, as well as the differential geometry of the skin is significantly different in the 3D case.

Our approach produces a surface that minimizes a convex combination of surface area and squared mean curvature terms. Such surfaces are loosely referred to as *minimal surfaces* in the literature. It has been shown in the surface evolution literature that, for unconstrained surfaces, these terms result in curvature and Willmore flows [9,10], respectively. By adding the additional constraint that the skin must pass through a circle of intersection with each ball, we significantly reduce the dimensionality of the optimization.

1.2. Our contribution

We model the skin as a C^1 surface, which, by construction, must touch each ball along a circle that is tangent to the ball. We then provide two novel derivations, one for deforming this constrained surface to minimize its surface area; and a second derivation for minimizing its squared mean curvature. The result of these derivations are differential equations, which we then solve to update a given surface to its optimal position. We then show experimental examples of how these differential equations are used to perform *optimally smooth* skinning of balls.

2. Methodology

2.1. Representation

We require the skin to pass through a circle on the ith ball $\mathbf{B}^i$, as depicted in Fig. 2 (a). This circle resides in a plane with normal $\mathbf{N}^i = [\cos\theta^i \sin\phi^i, \sin\theta^i \sin\phi^i, \cos\phi^i]^T$ that passes through the ball center $\mathbf{c}^i$ and intersects $\mathbf{B}^i$.

Let $\mathbf{q}(u)$ be a point on the unit circle in the $z = 0$ plane, as shown in Fig. 2(b). We can parameterize $\mathbf{q}(u)$ as

$$\mathbf{q}(u) = \hat{\mathbf{x}}\cos u + \hat{\mathbf{y}}\sin u, \tag{1}$$

where u is a parametrization angle. We can then express $\mathbf{p}^i(u)$, a point on the circle of $\mathbf{B}^i$ as

$$\mathbf{p}^i(u) = \mathbf{c}^i + a^i R^i \mathbf{q}(u), \tag{2}$$

where r^i is the radius of the ith ball, and R^i is a rotation matrix specified by $\mathbf{N}^i$ as

$$R^i = R(\theta^i, \phi^i) = \begin{bmatrix} \cos\theta^i\cos\phi^i & -\sin\theta^i & \cos\theta^i\sin\phi^i \\ \sin\theta^i\cos\phi^i & \cos\theta^i & \sin\theta^i\sin\phi^i \\ -\sin\phi^i & 0 & \cos\phi^i \end{bmatrix} \tag{3}$$

$\mathbf{p}^i(u)$ provides a way to parameterize the circle on each ball. We would like to form a parametric skin $\mathbf{S}(u, v)$ that satisfies several geometric criteria:

(1) The skin should be modeled by a circle that contacts each ball.
(2) The skin should be tangent to each ball along the circle of contact.
(3) The skin should optimize an energy functional composed of surface area and mean curvature terms.

We compose the skin $\mathbf{S}(u, v)$ as a set of segments $\mathbf{S}^i(u, v)$ for $i = 1 \ldots N - 1$, where N is the total number of balls, as depicted in Fig. 3. To form the segment $\mathbf{S}^i(u, v)$, we would like to generate a spline-based surface that connects the circles on adjacent balls. Various spline representations (such as Catmull–Rom, 4-point, etc.) are possible for modeling segments using a set of splines. Each spline starts at point $\mathbf{p}^i(u)$ in direction $\mathbf{N}^i$, and ends at point $\mathbf{p}^{i+1}(u)$ in direction $\mathbf{N}^{i+1}$, with

$$\mathbf{S}^i(u, v) = \mathbf{A}^i(u)v^3 + \mathbf{B}^i(u)v^2 + \mathbf{C}^i(u)v + \mathbf{D}^i(u), \tag{4}$$

since the four constraints require four degrees of freedom. For the ith segment, $\mathbf{A}^i(u)$, $\mathbf{B}^i(u)$, $\mathbf{C}^i(u)$, and $\mathbf{D}^i(u)$ are coefficients, and $v \in [0, 1]$ is a parametrization variable.

Each segment $\mathbf{S}^i(u, v)$ of the skin is defined by the Hermite interpolation of the boundary conditions, specifically, $\mathbf{S}^i(u, v)|_{v=0} = \mathbf{p}^i(u)$, $\frac{\partial \mathbf{S}^i(u,v)}{\partial v}|_{v=0} = t^i\mathbf{N}^i$, $\mathbf{S}^i(u, v)|_{v=1} = \mathbf{p}^{i+1}(u)$, and $\frac{\partial \mathbf{S}^i(u,v)}{\partial v}|_{v=1} = t^{i+1}\mathbf{N}^{i+1}$, where, for each i, t^i is a stiffness coefficient that controls the influence of the normal $\mathbf{N}^i$. Each t^i is fixed to be half the distance between the next and previous ball centers (for the first and last balls, it is the distance between the ball center and its neighbor ball center) for all examples in this paper. Such a stiffness encourages smoothness of the connecting segments at a circle, and is based on the central difference approximation of the first derivative computed using ball centers. We note however that straighter segments can be achieved by scaling the stiffness by a coefficient less than one.

With these constraints, and the derivative of the segment,

$$\frac{\partial \mathbf{S}^i(u, v)}{\partial v} = 3\mathbf{A}^i(u)v^2 + 2\mathbf{B}^i(u)v + \mathbf{C}^i(u), \tag{5}$$

we obtain a system of four equations for the four coefficients:
$\mathbf{D}^i(u) = \mathbf{p}^i(u), \mathbf{C}^i(u) = t^i\mathbf{N}^i, \mathbf{A}^i(u) + \mathbf{B}^i(u) + \mathbf{C}^i(u) + \mathbf{D}^i(u) = \mathbf{p}^{i+1}(u),$
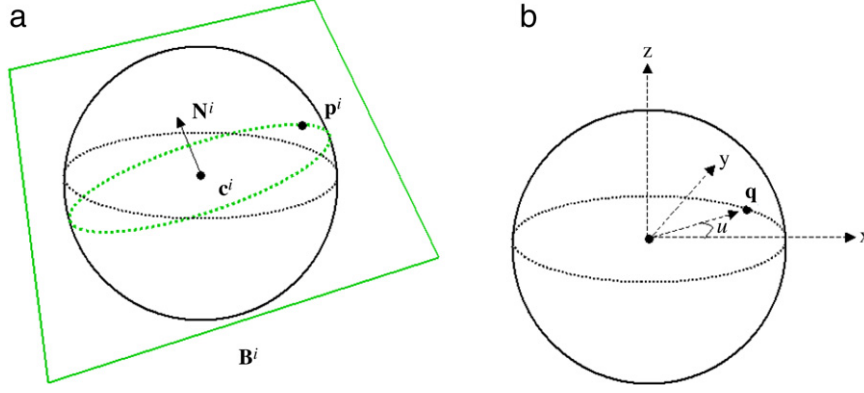
Fig. 2. Representation of the circle of contact. We show how point $\mathbf{q}(u)$ on the unit circle is mapped to point $\mathbf{p}^i(u)$, which is a point on the circle of contact where the skin is tangent to the ball.
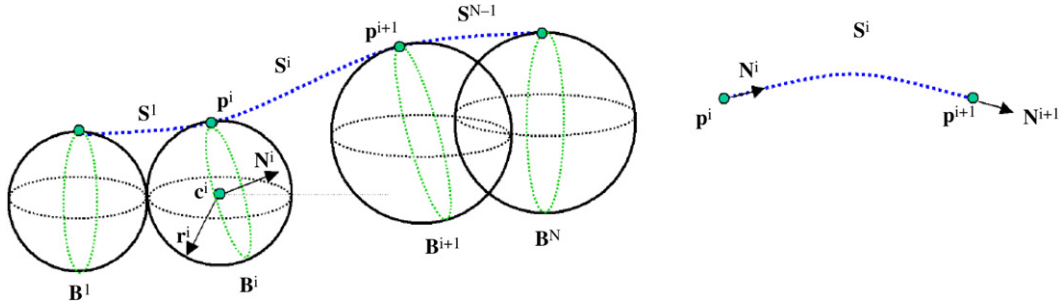

Fig. 3. Segments and spline.

and $3\mathbf{A}^i(u) + 2\mathbf{B}^i(u) + \mathbf{C}^i(u) = t^{i+1}\mathbf{N}^{i+1}$, which is easily solved, yielding

$$\mathbf{A}^i(u) = -2\mathbf{p}^{i+1}(u) + 2\mathbf{p}^i(u) + t^i\mathbf{N}^i + t^{i+1}\mathbf{N}^{i+1}$$

$$\mathbf{B}^i(u) = 3\mathbf{p}^{i+1}(u) - 3\mathbf{p}^i(u) - 2t^i\mathbf{N}^i - t^{i+1}\mathbf{N}^{i+1}$$

$$\mathbf{C}^i(u) = t^i\mathbf{N}^i \tag{6}$$

$$\mathbf{D}^i(u) = \mathbf{p}^i(u).$$

Thus, we have a way of describing the skin $\mathbf{S}(u, v)$ as a collection of $N - 1$ C^1 continuous segments $\mathbf{S}^i(u, v)$. Segment $\mathbf{S}^i(u, v)$ in turn is specified by the coefficients $\mathbf{A}^i(u)$, $\mathbf{B}^i(u)$, $\mathbf{C}^i(u)$, $\mathbf{D}^i(u)$, and these coefficients are functions of the circles $\mathbf{p}^i(u)$, $\mathbf{p}^{i+1}(u)$, and the normals $\mathbf{N}^i$ and $\mathbf{N}^{i+1}$. Finally, the circles and normals are, in turn, functions of the angles ϕ^i, ϕ^{i+1}, θ^i, θ^{i+1}.

3. Surface area minimization

In this section, we derive differential equations to evolve the parameters of the skin to minimize the skin's surface area.

From differential geometry, it is well known that the surface area is given by

$$J^a = \iint \sqrt{EG - F^2}\,du dv, \tag{7}$$

where

$$E = \mathbf{S}_u \cdot \mathbf{S}_u \tag{8}$$

$$F = \mathbf{S}_u \cdot \mathbf{S}_v \tag{9}$$

$$G = \mathbf{S}_v \cdot \mathbf{S}_v \tag{10}$$

are coefficients of the first fundamental form.

Since $\mathbf{S}(u, v)$ is expressed as a sum of $N - 1$ segments, this is equivalent to

$$J^a = \sum_{i=1}^{N-1} \iint \sqrt{E^i G^i - (F^i)^2}\,du dv. \tag{11}$$

We would like to take the derivative of Eq. (11) with respect to the parameter w^k, where $w^k \in [\theta^i, \phi^i]\forall i$. This will give us a gradient direction that we can use in a numerical gradient descent procedure to find the angles that minimize the surface area of the skin. Since θ^i and ϕ^i affect only the ith and $i - 1$st segments, we can replace the summation with two surface integrals,

$$\frac{\partial J^a}{\partial w^k} = \frac{\partial}{\partial w^k} \iint \sqrt{E^i G^i - (F^i)^2}\,du dv$$

$$+ \frac{\partial}{\partial w^k} \iint \sqrt{E^{i-1} G^{i-1} - (F^{i-1})^2}\,du dv. \tag{12}$$

Propagating the derivative through the integrals gives

$$\frac{\partial J^a}{\partial w^k} = \frac{1}{2} \iint \frac{\frac{\partial E^i}{\partial w^k}G^i + E^i\frac{\partial G^i}{\partial w^k} - 2F^i\frac{\partial F^i}{\partial w^k}}{\left[E^i G^i - (F^i)^2\right]^{\frac{1}{2}}}\,du dv$$

$$+ \frac{1}{2} \iint \frac{\frac{\partial E^{i-1}}{\partial w^k}G^{i-1} + E^{i-1}\frac{\partial G^{i-1}}{\partial w^k} - 2F^{i-1}\frac{\partial F^{i-1}}{\partial w^k}}{\left[E^{i-1} G^{i-1} - (F^{i-1})^2\right]^{\frac{1}{2}}}\,du dv. \tag{13}$$

In Appendix A, derivations for the derivatives of the coefficients of the first fundamental form with respect to w^k are given.

4. Curvature minimization

We would additionally like to derive differential equations for updating the skin to minimize its curvature. In 3D there are several potential curvatures one could employ, including mean and Gaussian curvatures. In this paper we focus on the mean curvature, which is closely related to the first variation of surface area. The mean curvature is given by

$$H = \frac{eG - 2fF + gE}{2(EG - F^2)}, \tag{14}$$

where E, F, and G are given in the previous section, and e, f, and g

Box I.

come from the second fundamental form,

$$e = \mathbf{M} \cdot \mathbf{S}_{uu} \tag{15}$$

$$f = \mathbf{M} \cdot \mathbf{S}_{uv} \tag{16}$$

$$g = \mathbf{M} \cdot \mathbf{S}_{vv} \tag{17}$$

where $\mathbf{M}$ is the surface normal (not to be confused with $\mathbf{N}$, which is the normal of the plane that intersects a ball).

Our energy is

$$J^c = \iint H^2 dudv. \tag{18}$$

Since $\mathbf{S}(u, v)$ is expressed as a sum of $N - 1$ segments, this is equivalent to

$$J^c = \sum_{i=1}^{N-1} \iint (H^i)^2 dudv. \tag{19}$$

As before, we would like to take the derivative of Eq. (19) with respect to the parameter w^k, where $w^k \in [\theta^i, \phi^i] \forall i$. This will give us a gradient direction which we can use in a numerical gradient descent procedure to find the angles that minimize the curvature of the skin. Since θ^i and ϕ^i affect only the ith and $i-1$st segments, we can replace the summation with two surface integrals,

$$\frac{\partial J^c}{\partial w^k} = \frac{\partial}{\partial w^k} \iint \left[\frac{e^iG^i - 2f^iF^i + g^iE^i}{2(E^iG^i - (F^i)^2)} \right]^2 dudv$$

$$+ \frac{\partial}{\partial w^k} \iint \left[\frac{e^{i-1}G^{i-1} - 2f^{i-1}F^{i-1} + g^{i-1}E^{i-1}}{2(E^{i-1}G^{i-1} - (F^{i-1})^2)} \right]^2 dudv. \tag{20}$$

Propagating the derivative through the integrals gives where the derivatives of the coefficients from the first fundamental form (i.e., E, F, and G) are given in Appendix A and the derivatives of the coefficients from the second fundamental form (i.e., e, f, and g) are analytically derived in Appendix B.

5. Implementation

We combine the energies J^a and J^c together, as

$$J = (1 - k)J^a + kJ^c, \tag{21}$$

where k is a constant used to weight the surface area minimization relative to the curvature minimization. Convex combinations of the two can be selected using $k \in [0, 1]$. Therefore, the combined energy minimization is given by

$$\frac{\partial J}{\partial w^k} = (1 - k)\frac{\partial J^a}{\partial w^k} + k\frac{\partial J^c}{\partial w^k}, \tag{22}$$

where $\frac{\partial J^a}{\partial w^k}$ is given in Eq. (13) and $\frac{\partial J^c}{\partial w^k}$ is provided in Box I. In all of the experiments in this paper, we fix $k = 0.9$, to encourage smoother solutions.

These equations are a set of differential equations that can be used in a gradient descent procedure to optimize the skin by manipulating the parameters $\mathbf{w}^i = [\theta^i, \phi^i]^T$ of each ball i. Let $\mathbf{w}^i(n)$ be the ith ball's parameters at iteration n. We can then update the parameters by moving them in the negative gradient direction, i.e.,

$$\mathbf{w}^i(n + 1) = \mathbf{w}^i(n) - \Delta t \nabla J_{\mathbf{w}^i(n)}, \forall i, \tag{23}$$

where Δt is a time step.

The computational complexity of the algorithm depends on the number of balls N and the number of points on the surface where the points and derivatives are evaluated. The number of points is given by LM, where L is the number of sampling points on each spline, and M is the number of splines on a segment. For each iteration of the gradient descent procedure, the computational complexity is $O(NLM)$. The number of iterations required depends on the time step Δt as well as how close the initial skin is to the final solution.

6. Experimental results

A simple example is provided in Fig. 4. Here, four balls of radius 3, 2, 2, and 3 units, respectively were set in the xz-plane, at points $\mathbf{c}^1 = [0, 0, 0]^T$, $\mathbf{c}^2 = [5, 0, 5]^T$, $\mathbf{c}^3 = [10, 0, 0]^T$ and $\mathbf{c}^4 = [17, 0, -5]^T$. The initial parameters for this experiment were $\mathbf{w}^1 = [0, 0]^T$, $\mathbf{w}^2 = [0, \pi/4]^T$, $\mathbf{w}^3 = [0, \pi/2]^T$, and $\mathbf{w}^4 = [0, \pi/4]^T$ respectively; the initial skin is shown in part (a) of the figure. The parameters were iteratively updated using Eq. (23), with $L = 50$ and $M = 20$ (these values for L and M are used for all experiments in this paper). An intermediate solution after 20 iterations in shown in (b); at this stage, the skin is considerably smoother while still satisfying the constraints of the problem. We show the result after 40 iterations in (c), at which point the energy has reached a minimum and the parameters have converged. The solution (all 40 iterations) is computed in 4.3 s using C++ code on a machine with a 2.0 GHz processor. We render the surface as a collection of splines in blue, and additionally show the circle of intersection on each ball in green. The energy of the surface, as measured using Eq. (21), drops from 7.36×10^9 in (a) to 670 in (c).

Fig. 5 shows a slightly more complicated example for which some balls overlap and others do not. The initial skin is shown in (a), an intermediate result after 25 iterations in (b), and the final result upon convergence after 50 iterations in (c). The solution (all 50 iterations) is computed in 10.4 s. The energy of the surface, as measured using Eq. (21), drops from 1.86×10^{10} in (a) to 1400 in (c).

Fig. 6 shows another example for a symmetric configuration of balls, but asymmetric initial conditions. The initial skin is shown in (a), an intermediate result after 30 iterations in (b), and the final result upon convergence after 60 iterations in (c). The solution (all 60 iterations) is computed in 8.3 s. The energy of the surface, as measured using Eq. (21), drops from 2.79×10^7 in (a) to 1287 in (c). Note that due to the symmetry of the balls, the skin itself is symmetric upon convergence.
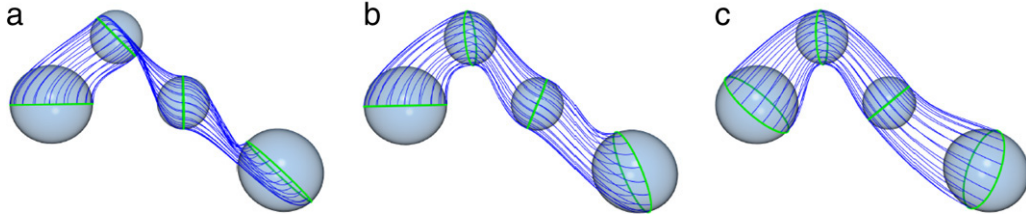
Fig. 4. Simple example demonstrating ball skinning. The initialization is shown in (a), and the result after 20 iterations is shown in (b), and the converged result after 40 iterations is shown in (c). The skin is rendered in a blue color.
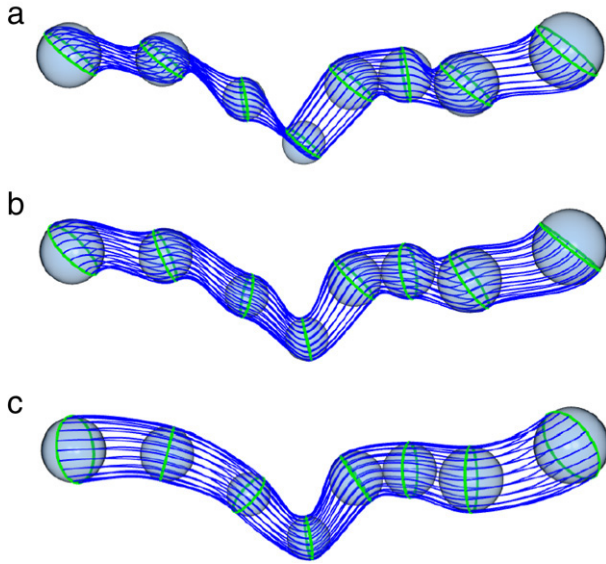

Fig. 5. Another ball skinning. The initialization is shown in (a), and the result after 25 iterations is shown in (b), and the converged result after 50 iterations is shown in (c).

Fig. 7. Generating a smooth interpolating region between a set of balls. Initialization (a), intermediate result (b) after 25 iterations, and final result upon convergence (c) after 50 iterations.

More examples are provided in Figs. 1 and 7. In Fig. 7, the balls are arranged on a sine wave and have a variable radius. In addition, some of the balls overlap while others do not. Convergence of the skinning algorithm, starting from a set of angles far from the optimal result, takes 11.4 s, and reduces the energy from 9.67×10^5 to 11,471. In Fig. 1, the variable radius balls are arranged in a spiral. The skin is generated in 15.3 s.

In Fig. 8, we show plot of the energy J of the surface vs. the iteration number. Note that initially, the energy is high and successive iterations reduce the energy until convergence occurs around the 30th iteration. Upon convergence, the energy oscillates around its minimal value. This fact can be exploited as an automatic convergence criterion.

6.1. Comparison

We have implemented the J-splines technique [3] to which we compare our method. This approach outlines a general subdivision algorithm for producing smooth curves (J-splines) from a set of ordered points. Iterative applications of the subdivision algorithm yield a family of limit curves, one of which is a quintic b-spline (C^4). This quintic b-spline will no longer interpolate the input points, but can be "retrofitted" as the paper describes. This retrofitting process iteratively offsets the input control points until the final curve interpolates the input, thus resulting in an interpolating C^4 spline. As a comparison, we used this subdivision approach to subdivide the series of balls as a 4D curve ($x, y, z + $ radius). The cross sections of the skin are then computed as circles which lie on the surface of the convex hull of every consecutive pair of balls and are also orthogonal to the line connecting their centers.

Table 1 presents the results of the comparison. For each skin, we compute the energy J as described in Eq. (21). While the result for the spiral and sine wave are slightly lower for the J-spline method, the difference is not significant (i.e., the difference is less than 1%) in both cases and the surfaces have an identical appearance. The PDE method demonstrates a significant improvement however for the other surfaces. The primary reason for this is that with J-splines, a self-intersection occurs that results in high local curvatures and a fatter object in general, as demonstrated in Fig. 9 below. The PDE method will penalize such local self-intersections and deform the spline surface so that the individual splines are smooth upon convergence.
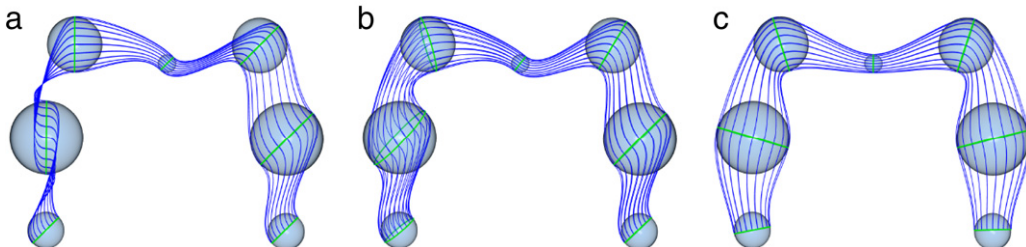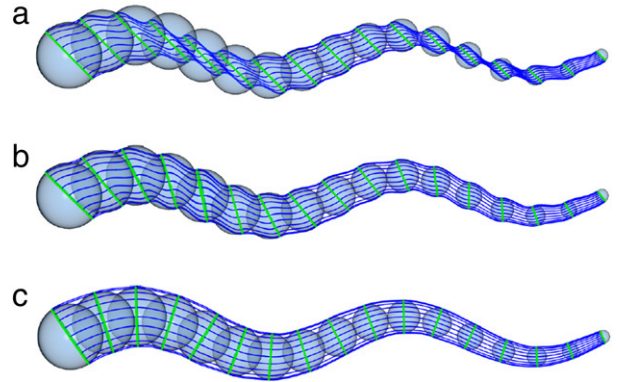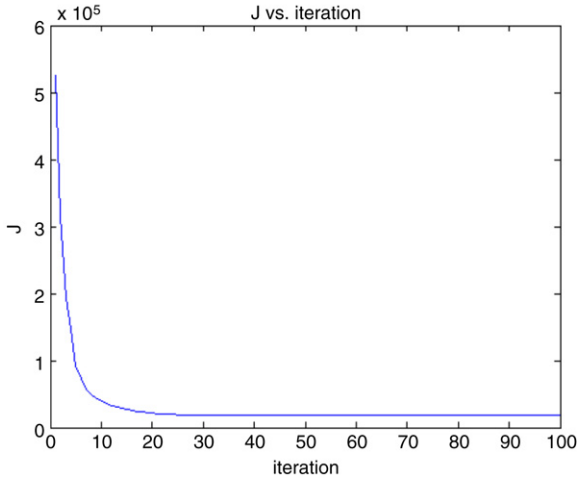

Fig. 6. An example for a symmetric configuration of balls. The initialization is shown in (a), and the result after 30 iterations is shown in (b), and the converged result after 60 iterations is shown in (c).

Fig. 8. Convergence plot for the sine wave example from Fig. 7.

convexity of the energy functional as well as the initial condition. In the examples shown in this paper, the initial skins are chosen to be far from the final solution to demonstrate the effect and robustness of the differential equations. Fig. 10, part (a) and a zoom in view (b), show an example with a very poor initialization that has a strong self-intersection. Despite the undesirable initialization, the algorithm is able to untangle the self-intersection and produce a smooth interpolation of the balls, shown in (c). In (d) and and a zoom-in view (e) of the same figure, we show an example of a severe self-intersection where the surface has completely folded in on itself. This initialization is not in the basin of attraction of the desired solution, so the skin upon convergence, shown in (f), is not the desired solution. In practice, it is typically easy to determine a good initialization by choosing an initialization for each ball such that the normal of the intersection plane points along the vector that connects adjacent ball centroids.
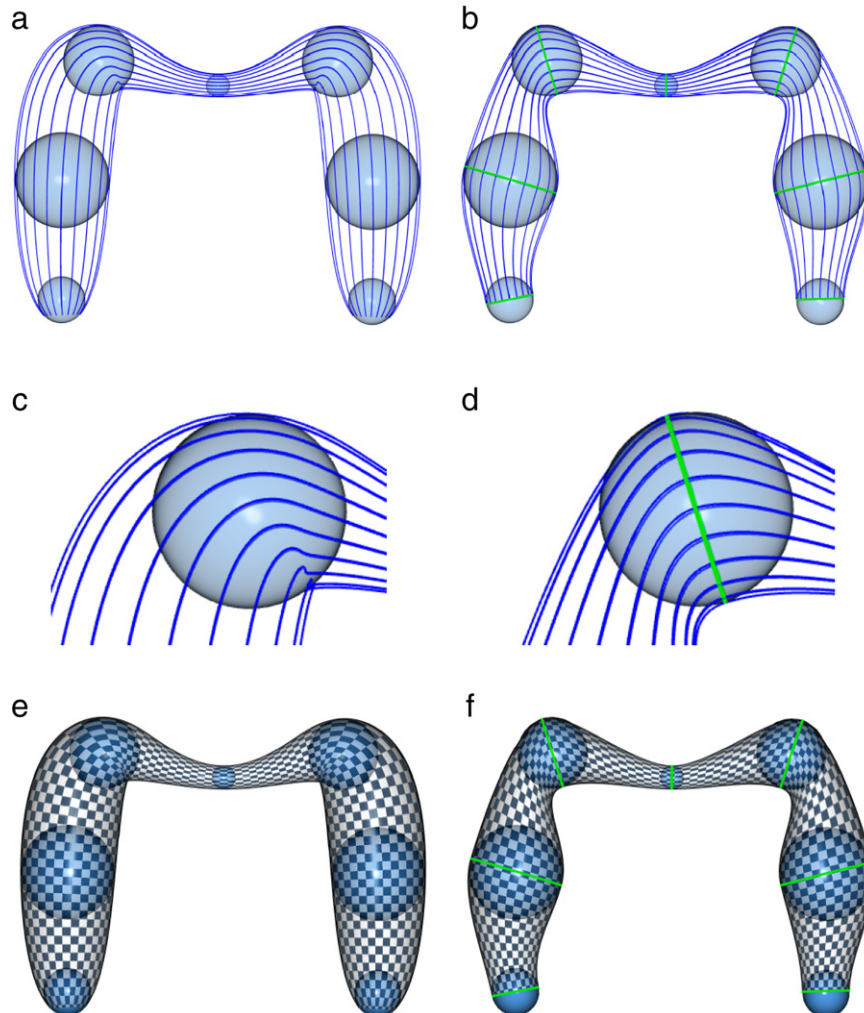
6.2. Discussion

We note that our gradient descent approach only guarantees a locally optimal solution; the particular solution depends on the

Fig. 9. PDE method produces smoother results. On the left, is the J-spline skin for the example of 6. The are two self-intersections; one is shown in a zoomed view of (c), with the green circle of contact removed for clarity. In (a) we show the wireframe model of the splines, and in (c) a texture-mapped view. On the right, is the result of the PDE method, which does not have a self-intersection.
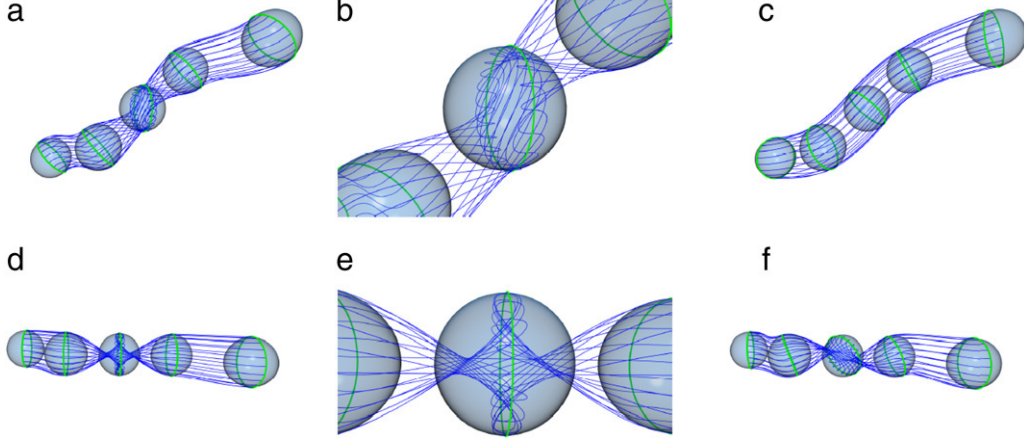
Fig. 10. Convergence to desired and undesired minima based on poor and severely poor initializations. The PDE approach is able to untangle the difficult case in (a), resulting in (c). However, the completely folded initialization of (d) results in the skin (f), which is not in the basin of attraction of the desired solution.
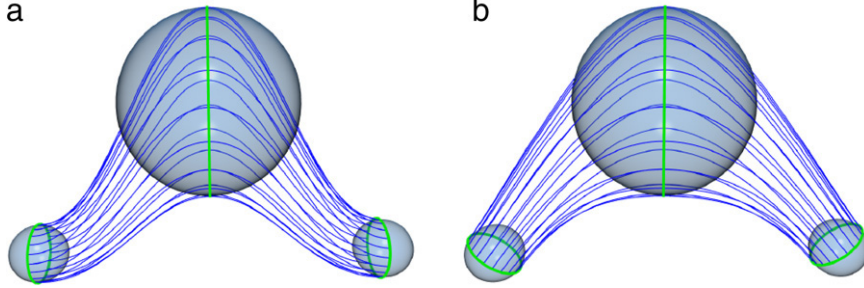

Fig. 11. Skin may pass through a ball. Initialization (a) and final result (b).

Note that the skin our method generates may pass through a ball (shown in Fig. 11(b)) since it is only constrained to be tangent to the ball at the circle of intersection. For points not in the circle's plane, the skin may be larger or smaller than the ball. Thus, the skin does not provide an *exact* envelope of the balls, but rather, an *approximating* envelope. In our application of modeling blood vessels, this is an acceptable solution since ball itself is a geometric proxy of the local vessel geometry.

7. Conclusion

In this paper, we presented a method for optimally skinning an ordered set of 3D balls. Our formulation of the problem requires that the skin be modeled by a circle of contact with each ball, and be tangent to the ball along this circle. We have presented novel derivations resulting in differential equations that minimize a convex combination of the surface area and mean curvature of a third order polynomial spline surface subject to these constraints. Starting with an initial skin, we evolve the skin's parameters until convergence. Experimental results demonstrate the viability of the method.

Acknowledgement

Appendix A. I: Derivatives of the Coefficients of the First Fundamental Form

The derivatives for the jth ball's ($j \in [i-1, i]$) coefficients of the first fundamental form with respect to $w^k \in \mathbf{w}^i = [\theta^i, \phi^i, d^i]^{\mathrm{T}}$ are given by

$$\frac{\partial E^j}{\partial w^k} = 2\mathbf{S}_u^j \cdot \frac{\partial \mathbf{S}_u^j}{\partial w^k} \tag{A.1}$$

$$\frac{\partial F^j}{\partial w^k} = \mathbf{S}_u^j \cdot \frac{\partial \mathbf{S}_v^j}{\partial w^k} + \mathbf{S}_v^j \cdot \frac{\partial \mathbf{S}_u^j}{\partial w^k} \tag{A.2}$$

$$\frac{\partial G^j}{\partial w^k} = 2\mathbf{S}_v^j \cdot \frac{\partial \mathbf{S}_v^j}{\partial w^k} \tag{A.3}$$

where

$$\mathbf{S}_u^j = \mathbf{A}_u^j v^3 + \mathbf{B}_u^j v^2 + \mathbf{C}_u^j v + \mathbf{D}_u^j \tag{A.4}$$

$$\frac{\partial \mathbf{S}_u^j}{\partial w^k} = \frac{\partial \mathbf{A}_u^j}{\partial w^k} v^3 + \frac{\partial \mathbf{B}_u^j}{\partial w^k} v^2 + \frac{\partial \mathbf{C}_u^j}{\partial w^k} v + \frac{\partial \mathbf{D}_u^j}{\partial w^k} \tag{A.5}$$

$$\mathbf{S}_v^j = 3\mathbf{A}^j v^2 + 2\mathbf{B}^j v + \mathbf{C}^j \tag{A.6}$$

$$\frac{\partial \mathbf{S}_v^j}{\partial w^k} = 3\frac{\partial \mathbf{A}^j}{\partial w^k} v^2 + 2\frac{\partial \mathbf{B}^j}{\partial w^k} v + \frac{\partial \mathbf{C}^j}{\partial w^k} \tag{A.7}$$

and

$$\frac{\partial \mathbf{A}^j}{\partial w^k} = \begin{cases} 2\dfrac{\partial \mathbf{p}^j}{\partial w^k} + t^j \dfrac{\partial \mathbf{N}^j}{\partial w^k}, & j = i \\ -2\dfrac{\partial \mathbf{p}^{j+1}}{\partial w^k} + t^{j+1}\dfrac{\partial \mathbf{N}^{j+1}}{\partial w^k}, & j = i-1 \end{cases} \tag{A.8}$$

$$\frac{\partial \mathbf{B}^j}{\partial w^k} = \begin{cases} -3\dfrac{\partial \mathbf{p}^j}{\partial w^k} - 2t^j \dfrac{\partial \mathbf{N}^j}{\partial w^k}, & j = i \\ 3\dfrac{\partial \mathbf{p}^{j+1}}{\partial w^k} - t^{j+1}\dfrac{\partial \mathbf{N}^{j+1}}{\partial w^k}, & j = i-1 \end{cases} \tag{A.9}$$

$$\frac{\partial \mathbf{C}^j}{\partial w^k} = \begin{cases} t^j \dfrac{\partial \mathbf{N}^j}{\partial w^k}, & j = i \\ \mathbf{0}, & j = i-1 \end{cases} \tag{A.10}$$

$$\frac{\partial \mathbf{D}^j}{\partial w^k} = \begin{cases} \frac{\partial \mathbf{p}^j}{\partial w^k}, & j = i \\ \mathbf{0}, & j = i - 1 \end{cases} \tag{A.11}$$

$$\mathbf{A}_u^j = -2\mathbf{p}_u^{j+1} + 2\mathbf{p}_u^j \tag{A.12}$$

$$\mathbf{B}_u^j = 3\mathbf{p}_u^{j+1} - 3\mathbf{p}_u^j \tag{A.13}$$

$$\mathbf{C}_u^j = \mathbf{0} \tag{A.14}$$

$$\mathbf{D}_u^j = \mathbf{p}_u^j \tag{A.15}$$

$$\frac{\partial \mathbf{A}_u^j}{\partial w^k} = \begin{cases} 2\frac{\partial \mathbf{p}_u^j}{\partial w^k}, & j = i \\ -2\frac{\partial \mathbf{p}_u^{j+1}}{\partial w^k}, & j = i - 1 \end{cases} \tag{A.16}$$

$$\frac{\partial \mathbf{B}_u^j}{\partial w^k} = \begin{cases} -3\frac{\partial \mathbf{p}_u^j}{\partial w^k}, & j = i \\ 3\frac{\partial \mathbf{p}_u^{j+1}}{\partial w^k}, & j = i - 1 \end{cases} \tag{A.17}$$

$$\frac{\partial \mathbf{C}_u^j}{\partial w^k} = \mathbf{0} \tag{A.18}$$

$$\frac{\partial \mathbf{D}_u^j}{\partial w^k} = \begin{cases} \frac{\partial \mathbf{p}_u^j}{\partial w^k}, & j = i \\ \mathbf{0}, & j = i - 1 \end{cases} \tag{A.19}$$

$$\frac{\partial \mathbf{p}^i}{\partial \theta^i} = a^i \frac{\partial R^i}{\partial \theta^i}(\mathbf{x}\cos u + \mathbf{y}\sin u) \tag{A.20}$$

$$\frac{\partial \mathbf{p}^i}{\partial \phi^i} = a^i \frac{\partial R^i}{\partial \phi^i}(\mathbf{x}\cos u + \mathbf{y}\sin u) \tag{A.21}$$

$$\mathbf{p}_u^i = a^i R^i(-\mathbf{x}\sin u + \mathbf{y}\cos u) \tag{A.22}$$

$$\frac{\partial \mathbf{p}_u^i}{\partial \theta^i} = a^i \frac{\partial R^i}{\partial \theta^i}(-\mathbf{x}\sin u + \mathbf{y}\cos u) \tag{A.23}$$

$$\frac{\partial \mathbf{p}_u^i}{\partial \phi^i} = a^i \frac{\partial R^i}{\partial \phi^i}(-\mathbf{x}\sin u + \mathbf{y}\cos u) \tag{A.24}$$

and

$$\frac{\partial R^i}{\partial \theta^i} = \begin{bmatrix} -\sin\theta^i\cos\phi^i & -\cos\theta^i & -\sin\theta^i\sin\phi^i \\ \cos\theta^i\cos\phi^i & -\sin\theta^i & \cos\theta^i\sin\phi^i \\ 0 & 0 & 0 \end{bmatrix} \tag{A.25}$$

$$\frac{\partial R^i}{\partial \phi^i} = \begin{bmatrix} -\cos\theta^i\sin\phi^i & 0 & \cos\theta^i\cos\phi^i \\ -\sin\theta^i\sin\phi^i & 0 & \sin\theta^i\cos\phi^i \\ -\cos\phi^i & 0 & -\sin\phi^j \end{bmatrix} \tag{A.26}$$

and

$$\frac{\partial \mathbf{N}^i}{\partial \theta^i} = [-\sin\theta^i\sin\phi^i, \cos\theta^i\sin\phi^i, 0]^\mathrm{T} \tag{A.27}$$

$$\frac{\partial \mathbf{N}^i}{\partial \phi^i} = [\cos\theta^i\cos\phi^i, \sin\theta^i\cos\phi^i, -\sin\phi^i]^\mathrm{T}. \tag{A.28}$$

Appendix B. II: Derivatives of the coefficients of the second fundamental form

The derivatives for the jth ball's ($j \in [i-1, i]$) coefficients of the second fundamental form with respect to $w^k \in \mathbf{w}^i = [\theta^i, \phi^i, d^i]^\mathrm{T}$ are given by

$$\frac{\partial e^j}{\partial w^k} = \frac{\partial \mathbf{M}^j}{\partial w^k} \cdot \mathbf{S}_{uu}^j + \mathbf{M}^j \cdot \frac{\partial \mathbf{S}_{uu}^j}{\partial w^k} \tag{B.1}$$

$$\frac{\partial f^j}{\partial w^k} = \frac{\partial \mathbf{M}^j}{\partial w^k} \cdot \mathbf{S}_{uv}^j + \mathbf{M}^j \cdot \frac{\partial \mathbf{S}_{uv}^j}{\partial w^k} \tag{B.2}$$

$$\frac{\partial g^j}{\partial w^k} = \frac{\partial \mathbf{M}^j}{\partial w^k} \cdot \mathbf{S}_{vv}^j + \mathbf{M}^j \cdot \frac{\partial \mathbf{S}_{vv}^j}{\partial w^k} \tag{B.3}$$

where

$$\mathbf{M}^j = \frac{\mathbf{S}_u^j \times \mathbf{S}_v^j}{||\mathbf{S}_u^j \times \mathbf{S}_v^j||} \tag{B.4}$$

$$\frac{\partial \mathbf{M}^j}{\partial w^k} = \frac{\frac{\partial \mathbf{S}_u^j}{\partial w^k} \times \mathbf{S}_v^j}{||\mathbf{S}_u^j \times \mathbf{S}_v^j||} + \frac{\mathbf{S}_u^j \times \frac{\partial \mathbf{S}_v^j}{\partial w^k}}{||\mathbf{S}_u^j \times \mathbf{S}_v^j||} - \frac{\left[(\mathbf{S}_u^j \times \mathbf{S}_v^j) \cdot \left(\frac{\partial \mathbf{S}_u^j}{\partial w^k} \times \mathbf{S}_v^j + \mathbf{S}_u^j \times \frac{\partial \mathbf{S}_v^j}{\partial w^k}\right)\right] \mathbf{S}_u^j \times \mathbf{S}_v^j}{||\mathbf{S}_u^j \times \mathbf{S}_v^j||^3} \tag{B.5}$$

where $\mathbf{S}_u^j$, $\mathbf{S}_v^j$, $\frac{\partial \mathbf{S}_u^j}{\partial w^k}$, and $\frac{\partial \mathbf{S}_v^j}{\partial w^k}$ are given in Appendix A, and

$$\mathbf{S}_{uu}^j = \mathbf{A}_{uu}^j v^3 + \mathbf{B}_{uu}^j v^2 + \mathbf{C}_{uu}^j v + \mathbf{D}_{uu}^j \tag{B.6}$$

$$\mathbf{S}_{uv}^j = 3\mathbf{A}_u^j v^2 + 2\mathbf{B}_u^j v + \mathbf{C}_u^j \tag{B.7}$$

$$\mathbf{S}_{vv}^j = 6\mathbf{A}^j v + 2\mathbf{B}^j \tag{B.8}$$

$$\frac{\partial \mathbf{S}_{uu}^j}{\partial w^k} = \frac{\partial \mathbf{A}_{uu}^j}{\partial w^k} v^3 + \frac{\partial \mathbf{B}_{uu}^j}{\partial w^k} v^2 + \frac{\partial \mathbf{C}_{uu}^j}{\partial w^k} v + \frac{\partial \mathbf{D}_{uu}^j}{\partial w^k} \tag{B.9}$$

$$\frac{\partial \mathbf{S}_{uv}^j}{\partial w^k} = 3\frac{\partial \mathbf{A}_u^j}{\partial w^k} v^2 + 2\frac{\partial \mathbf{B}_u^j}{\partial w^k} v + \frac{\partial \mathbf{C}_u^j}{\partial w^k} \tag{B.10}$$

$$\frac{\partial \mathbf{S}_{vv}^j}{\partial w^k} = 6\frac{\partial \mathbf{A}^j}{\partial w^k} v + 2\frac{\partial \mathbf{B}^j}{\partial w^k} \tag{B.11}$$

and

$$\frac{\partial \mathbf{A}_{uu}^j}{\partial w^k} = \begin{cases} 2\frac{\partial \mathbf{p}_{uu}^j}{\partial w^k}, & j = i \\ -2\frac{\partial \mathbf{p}_{uu}^{j+1}}{\partial w^k}, & j = i - 1 \end{cases} \tag{B.12}$$

$$\frac{\partial \mathbf{B}_{uu}^j}{\partial w^k} = \begin{cases} -3\frac{\partial \mathbf{p}_{uu}^j}{\partial w^k}, & j = i \\ 3\frac{\partial \mathbf{p}_{uu}^{j+1}}{\partial w^k}, & j = i - 1 \end{cases} \tag{B.13}$$

$$\frac{\partial \mathbf{C}_{uu}^j}{\partial w^k} = \mathbf{0} \tag{B.14}$$

$$\frac{\partial \mathbf{D}_{uu}^j}{\partial w^k} = \begin{cases} \frac{\partial \mathbf{p}_{uu}^j}{\partial w^k}, & j = i \\ \mathbf{0}, & j = i - 1 \end{cases} \tag{B.15}$$

where

$$\mathbf{p}_{uu}^i = a^i R^i(-\mathbf{x}\cos u - \mathbf{y}\sin u) \tag{B.16}$$

$$\frac{\partial \mathbf{p}_{uu}^i}{\partial \theta^i} = a^i \frac{\partial R^i}{\partial \theta^i}(-\mathbf{x}\cos u - \mathbf{y}\sin u) \tag{B.17}$$

$$\frac{\partial \mathbf{p}_{uu}^i}{\partial \phi^i} = a^i \frac{\partial R^i}{\partial \phi^i}(-\mathbf{x}\cos u - \mathbf{y}\sin u). \tag{B.18}$$

References

[1] Singh K, Kokkevis E. Skinning characters using surface oriented free-form deformations. In: Graphics interface. 2000. p. 35–42.
[2] Peternell M. Rational parametrizations for envelopes of quadric families. Ph.D. thesis. Vienna (Australia): University of Technology; 1997.
[3] Rossignac J, Schaefer S. J-splines. Computer Aided Design 2008;40(10–11): 1024–32.
[4] Cheng H, Shi X. Quality mesh generation for molecular skin surfaces using restricted union of balls. In: IEEE visualization. 2005.

[5] Edelsbrunner H. Deformable smooth surface design. Discrete and Computational Geometry 1999;21(1):87–115.

[6] Kruithov N, Vegter G. Envelope surfaces. In: Annual symposium on computational geometry. 2006. p. 411–20.

[7] Whited B, Rossignac J, Slabaugh G, Fang T, Unal G. Pearling: 3D interactive extraction of tubular structures from volumetric images. In: Interaction in Medical Image Analysis and Visualization, held in conjunction with MICCAI. 2007.

[8] Slabaugh G, Unal G, Fang T, Rossignac J, Whited B. Variational skinning of an ordered set of discrete 2D balls. In: Geometric modeling and processing. 2008.

[9] Sethian JA. Level set methods and fast marching methods evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science. Cambridge University Press; 1999.

[10] Schatzle KER. Gradient flow for the Willmore functional. Communications in Analysis and Geometry 10 (5).