Spherical Splines in Modelling and Animation

Roman Ďurikovič
Department of Material Science end Engineering, Faculty of Engineering
Kyushu University, Japan

Abstract

A spherical spline is the smooth volume swept out by moving sphere whose centre follows a 3-D spline curve, surface or the volume of a volumetric spline. Spherical splines can be rendered rapidly, in composition they can form branched and highly concave objects. The spline in our method provides an underlying skeleton of the modelled object, making it easier to edit in an animation environment.

Key Words: blending, three-dimensional modelling, skeletal design

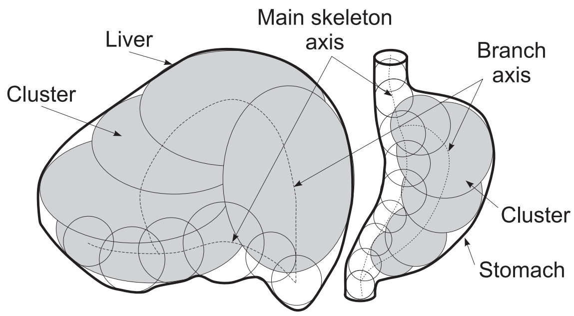

Figure 1: Object representation with cell clusters. Human embryo liver and stomach.

1 Introduction

Using a skeleton structure, we represent the topology of an object. Besides topology, the geometrical structure of objects is modelled using spheres and/or ellipses centred at nodes established on the skeleton. There are a number of applied problems where spherical representation of object shapes is more than appropriate. Published methods dealing with the spherical representation of 3-D objects [5] can be used to obtain the initial skeleton and spherical clusters for the approximation of object shape.

For the sake of clarity, in Fig. 1, we demonstrate the spherical representation in two dimensions. For a 3-D representation, the term "circle" should be substituted with "sphere" and "2-D skeleton" with "3-D skeleton". Figure 1 shows the skeleton and spherical clusters for a human embryo liver and stomach, respectively.

The main reason to use the spherical representation, is its stability. In other words, the perturbation of several boundary points results in very similar spherical representations. On the other hand, the disadvantage of this representation is that the sphere has very few degrees of freedom, only, limited types of shapes can be accurately represented by the union of a small number of spheres. Nevertheless, the spherical representation can accurately capture the shape of such objects as human organs or multiple branching structures.

As spheres are used in the modeling process, the surface generated from the union of original spheres is not smooth, a situation that unfortunately leads to an appearance that is not smooth. One effective way to smooth the surface without affecting the representation itself is to use scaled blended spheres [1] (also known as *blobbies*) during the process of rendering the organ. Unfortunately, the modeling an implicit surface defined by the blobies without any bulge or crease appears difficult.

Convolution surfaces [2] produced by a convoluting integral along the skeleton define the smooth resulting surface even for a non-convex skeleton. However, Bloomenthal showed that the convolution method generates bulges at a joint of one dimensional skeletal pieces.

The rest of this paper tries to overcome the following problems:

- description and handeling of branching and highly concave structures
- choice of an appropriate representation for objects such as human organs.

Here we are interested in representation of complex objects by continuous *spherical splines*. Volumetric interpolation splines introduced in Section 2 are used for definition of spherical splines introduced in next section. In Section 3 the spherical splines are defined in order to create a generalized cylinder of varying radius and a generalized plate with varying thickness.

2 Volumetric Spline - Variational Approach

Splines are well known in Computer Aided Geometric Design [3]. They have been widely used for modeling curves and surfaces that must pass exactly through individual points, so called *interpolation splines*. Let us first formulate the problem of interpolation with volumetric splines. Let Ω be an arbitrary shaped n-dimensional domain containing a set of points $P_i = (x_1^i, x_2^i, \ldots, x_n^i) : i = 1, 2, \ldots, N$ and r_i be a prescribed value of a spline at point P_i. The problem of finding the interpolation spline function

$$U(x) \in W_2^m(\Omega)$$

where $x = (x_1, x_2, \ldots, x_n)$ and W_2^m is the set of all functions whose squares of all derivatives of order $\leq m$ are integrable over $\mathbf{R}^n$, so that

$$U(P_i) = r_i, \qquad i = 1, \ldots, N,$$

and

$$\int_\Omega \sum_{|\alpha|=m} \frac{m!}{\alpha!}(D^\alpha U)^2 d\Omega \to \min$$

where m is a parameter of the variational functional, is considered in details in [6, 10].

The Green's function for the operator $T * T$ where

$$Tu = \sqrt{\frac{m!}{\alpha!}} D^\alpha U$$

has the form

$$G_{m,n}(x, P_i) = \begin{cases} \|x - P_I\|^{2m-n} \ln \|x - P_i\| & n \text{ even}, \\ \|x - P_I\|^{2m-n}, & n \text{ odd}, \end{cases}$$

where

$$\|x - P_I\| = \left[\sum_{j=1}^{n} (x_j - x_j^i)^2 \right]^{1/2}.$$

Finally, the spline has the form

$$U(x) = \sum_{i=1}^{N+k} \lambda_i g_i(x, P_i)$$

where

$$\begin{aligned} g_i(x, P_i) = G_{m,n}(x, P_i), \qquad & i = 1, \ldots, N \\ g_{N+1}(x, P_i) = 1 & \\ g_{N+1+j}(x, P_i) = x_j, \qquad & j = 1, \ldots, k-1, \\ k = (n + m - 1)!/n!(m-1)!. & \end{aligned}$$

If among the points $P_1, \ldots, P_N$ there are k points with the property that for any predefined set of values there is exactly one polynomial of n variables of degree $\leq m - 1$ that takes these values at these points, then the problem has a unique solution. If $m = n = 2$, then $k = 3$, and it is sufficient to find three points not lying in a straight line.

The minimization of the functional results in the following system of $N + k$ linear algebraic equations for the spline coefficients λ_i:

$$A \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \\ \lambda_{N+1} \\ \lambda_{N+2} \\ \vdots \\ \lambda_{N+k} \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

where $A = (A_{ij})$ is the matrix with the entries

$$\begin{aligned} A_{ij} = g_i(P_i, P_j), \qquad & i \leq N + k, j \leq N, i \neq j, \\ A_{ii} = 0 \qquad & i < N, \\ A_{ij} = g_i(P_i, P_j), \qquad & i \leq N, N < j \leq N + k, \\ A_{ij} = 0 \qquad & i > N, j > N. \end{aligned}$$

The matrix is the dense matrix of size N and above all often very ill-conditioned; we note for example that its diagonal elements are zero whereas an off-diagonal element can have a very large value. This matrix is symmetric, but not positive definite. Several attempts have been made to solve this system, the one method that can solve this system is the Householder method described in [4].

3 Spherical Spline

A *spherical spline* proposed here consists in extending the concept of splines applied on values to splines applied on spheres. A sphere is represented as a four dimensional vector, $s = (x, y, z, r)$, where the first three component represent the coordinate of the center of the sphere, and the last one its radius r.

3.1 Curve

The *spherical spline curve* should not be considered as a sweeping sphere [9] whose centre is moving along a spline curve while changing its radius, because radius is also involved in the minimisation of the spline functional. Let Ω be a 1-dimensional parametric domain that contains a set of nodal points $u_i : i = 1, \ldots, N$ and $s_i = (x_i, y_i, z_i, r_i)$ be a prescribed sphere at a node u_i. The volumetric spline interpolation described in Section 2 was employed on each single co-ordinate of a sphere independently, leading to spherical spline curve in the following form

$$U_{sc} = (U_x(u), U_y(u), U_z(u), U_r(u)),$$

where U_x, U_y, U_z, and U_r are the volumetric splines interpolating the set of values x_i, y_i, z_i and r_i, respectively.

3.2 Patch

The *spherical spline patch* shown in Fig. 2 is a 2-dimensional thin plate spline with continually varying thickness. In this case, Ω is a 2-dimensional domain containing a set of nodal points $(u_i, v_i) : i = 1, \ldots, N$ with prescribed sphere $s_i = (x_i, y_i, z_i, r_i)$ at a node (u_i, v_i). The spherical spline patch, introduced here in analogous way as spherical curve, has the following form

$$U_{sp} = (U_x(u,v), U_y(u,v), U_z(u,v), U_r(u,v)),$$

where U_x, U_y, U_z, and U_r are the volumetric splines interpolating the set of values x_i, y_i, z_i and r_i, respectively. The *spherical spline volume* is defined in a similar way by using a three-dimensional parametric domain Ω containing the points (u, v, w).

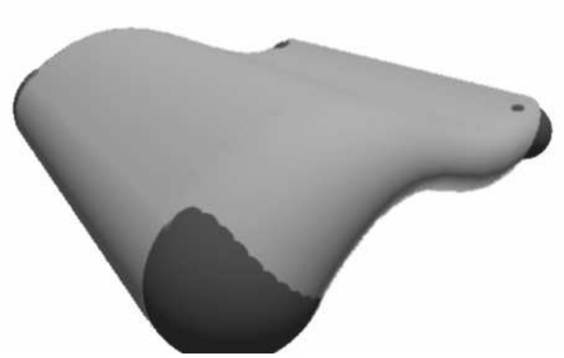

Figure 2: Two-dimensional spherical spline is a plate interpolating nodal spheres.

4 Control Spheres

The shape of an object is entirely given by the size and position of control spheres over the properly shaped domain Ω. Three problems are considered bellow: unknown shape of parametric domain, size and position of spheres. They can be solved using the spherical representation of an object as proposed here, see Fig. 1.

First, the parametric domain was determined based on a 3D geometric skeleton of an object. The *geometric skeleton* is the locus of the centres of spheres forming the spherical representation of an object. The tree structure of a skeleton can not handle a complex topology as many organs have, which can only be described by the skeletons with cycles. If the skeleton have no cycles then the domain is simply defined by the union of skeleton segments while the spherical spline curve will be used in interpolation process. The segments of a spherical spline curve are further joined by set-theoretic operations to form the object. A cycle present in a skeleton will represent a 2-dimensional part of parametric domain enclosed by the cycle while the spherical spline patch defined over the enclosed domain will be used in the interpolation. Thus, the formed domain consists of 2-dimensional patches and 1-dimensional curvilinear segments. Unfortunately, to work numerically with such a domain is not an easy task. Therefore, all 1-dimensional segments were transformed to thin 2-dimensional bands giving us a 2D parametric domain (see Fig. 3). The obtained 2D domain was further projected into a rectangular domain that is easier to implement and then the spherical spline patch was used in interpolation process.

Second, the size of a control sphere was selected to be the same as that in spherical representation of the object.

Third, the control spheres are distributed along the domain boundary $\partial\Omega$, implying that their position is close to that in spherical representation. Finally, the obtained control spheres were used to determine the interpolation with the spherical spline patch and consequently rendered with a raytracing algorithm.

5 Rendering

We use the raytracing approach to the rendering of spherical splines. The central computational task is twofold. First, one has to efficiently convert the equations of the surface and a given ray into a single equation, i.e., a polynomial in one variable. Second, a numerical procedure must be employed to compute the smallest positive root of the polynomial. This approach is supported by the public domain raytracer *rayshade* of Craig Kolb which requires supplying a software module for the ray-surface intersection calculations corresponding to spherical splines. The application of the system [7] allows to select a suitable view, lighting, material texture etc. of an object under interactive control and subsequently produces a corresponding input file for the *rayshade* program (see Fig. 4).

6 Implementation and Examples

The proposed method of modelling with spherical splines is demonstrated on an example of a human embryo stomach. The proposed method assumes a given set of control spheres selected from the spherical representation of an object. Construction of the spherical representation of human stomach is discussed in our paper [8], where the size and position of spheres along with the shape of the skeleton are determined automatically using an artificial life approach simulating the stomach growth. The simulation process calculates time variation of sphere in size and also its position along the skeleton. Additionally, skeleton undergoes the global and local transformations, while new spheres are created during the simulated growth. The drawback of our simulation method [8] was a huge number of generated spheres (say more than 300) making the rendering time extremely long. Unfortunately, also the shape, in our preliminary implementation approximated with union of metaballs, didn't look smooth during the animation. To speedup the rendering process the total number of spheres was rapidly reduced (to less than 20) and a spherical spline interpolation was employed to produce a smooth surface. The spherical splines proposed above have the advantage that fever control spheres can produce a smooth surface without any bulge or crease in a reasonable time.

The parametric domain was determined from a skeleton of the spherical representation shown in Fig. 3a that was actually an output from our growth simulation method. An example of the domain Ω consisting of a band and area patch for a certain stage of development is shown in Fig. 3b. Parametric domain was further projected into a rectangular domain for an easier manipulation, see Fig. 3c. Afterwards, spheres controlling the shape of spherical spline were distributed along the domain boundary $\partial\Omega$. Their, position and radius was determined automatically based on the simulated growth. Finally the shape of the stomach was calculated from spherical spline patch interpolating the control spheres over the given parametric domain Ω'.

Two stomach models with age 84 and 140 days using only 14 and 20 control spheres, respectively are shown in Fig. 4. The shape of models is represented by a spherical spline patch proposed in this paper. Table 1 compares both models from Fig. 4 and their total time needed for rendering and volumetric spline interpolation.

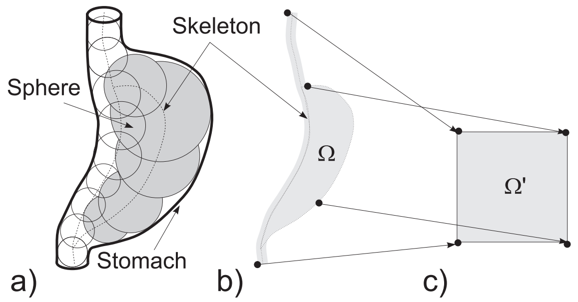

Figure 3: From spherical representation to parametric domain. a) Spherical representation of a stomach. b) Simple connected 2-dimesional parametric domain Ω with a band. c) Rectangular domain Ω' used for numerical implementations after the projection of Ω.

Two additional cases, one using a large number of metaballs generated automatically by simulated growth algorithm and the other one using few selected control spheres and the proposed spherical spline representation are summarised in table. Total rendering time of the proposed algorithm is applicable in rendering of an animation sequence while the former algorithm (using a large set of metaballs) is not.

Stomach model of age [day]	# of metaballs	Total time [min]
84	168	312
140	364	480
Stomach model of age [day]	**# of control spheres**	**Total time [min]**
84	14	37
140	20	53

Table 1: Total time for spherical spline interpolation and rendering.

7 Conclusions

This paper discussed the possibility to represent the shape of an object based on the volumetric spline interpolation of prescribed spheres. The method has the advantage of simple modelling of shapes as opposed to modelling using metaballs. The bulge created after the addition of a new metaball does not apply in representation with spherical splines. The spherical spline representation results in a smooth and controllable shape. Ability to determine control spheres automatically makes the proposed method also applicable in rendering of an animation sequence. The proposed

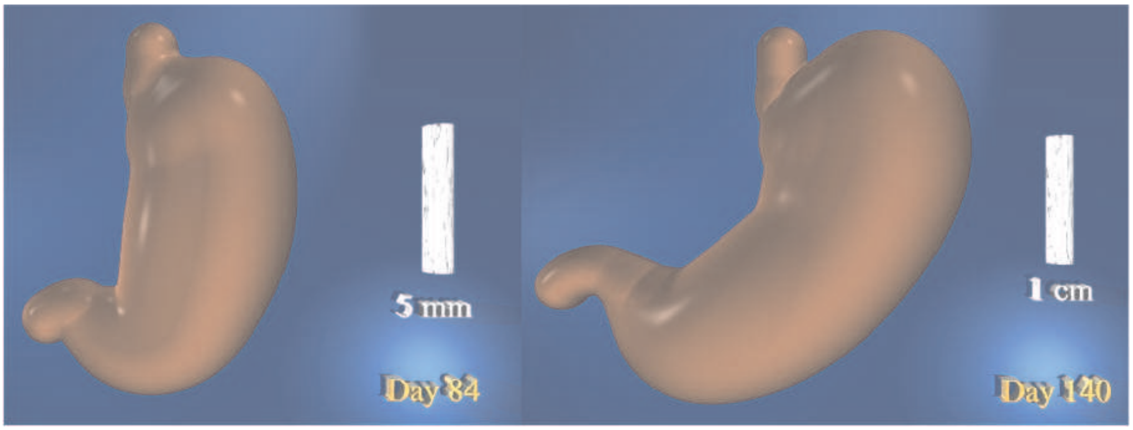

Figure 4: Two models of a stomach at development stages 84 and 140 days, respectively. The shape is approximated by spherical splines patches proposed above a) using 12 and b) 18 control spheres, respectively.

algorithm can generate highly concave objects. In addition, this technique allows generation of objects with complicated topology by application of set-theoretic operations.

The applicability of proposed approaches is limited if the number of spheres is large (say $N > 100$) in practical applications. In the case of a great number of spheres the volume interpolation requires heavy calculations because the time of calculation depends on N^3. To use a parallel computing for interpolation splines is a promising way of decreasing the calculation time.

8 Acknowledgment

This research was sponsored by grants from the Japanese Society for the Promotion of Science and Japanese Ministry of Education, Sport and Culture (MONBUSHO).

References

[1] J. F. Blinn. A generalization of algebraic surface drawing. *ACM Transactions on Graphics*, 1(3):235–256, 1982.

[2] J. Bloomenthal. Bulge elimination in convolution surfaces. *Computer Graphics Forum*, 16(1):31–41, 1997.

[3] C. De. Boor. *A practical guide to splines.* Springer-Verlag, 1978.

[4] G. H. Golub and C. F. van Loan. *Matrix computation.* The John Hopkins University Press, Baltimore, 1984.

[5] V. Ranjan and A. Fournier. Volume models for volumetric data. *IEEE Computer, special issue on volume visualization*, 27(7):28–36, 1994.

[6] V. A. Vasilenko. *Spline-functions: theory, algorithms, programs.* Nauka Publishers, Novosibirsk, 1983.

[7] R. Ďurikovič, K. Kaneda, and H. Yamashita. Visual modeling of stomach growth on the basis of L-systems. In *Proceedings of the Shape Modeling International*, pages 121–128, Aizu-Wakamatsu, Fukushima, Japan, March 1997.

[8] R. Ďurikovič, K. Kaneda, and H. Yamashita. Animation of biological organ growth based on l-systems. *Computer Graphics Forum (EUROGRAPHICS'98)*, 17(3), August 1998.

[9] J. J. Van Wijk. Ray tracing objects defined by sweeping a sphere. In *Proc. EUROGRAPHICS '84*, pages 73–82, held in Copenhagen, Denmark, 1984.

[10] F. Yamaguchi. *Curves and surfaces in computer aided geometric design.* Springer Verlag, Berlin, 1988.