한국경영과학회/대한산업공학회 2003 춘계공동학술대회
2003 년 5 월 16 일-17 일 한동대학교(포항)

윤곽선들의 B-spline 곡면 보간을 위한 새로운 방식
A new approach for B-spline surface interpolation to contours

박형준 [1], 정형배 [2], 김광수 [3]
[1] 조선대학교 산업공학과
[2] 목포대학교 선박해양공학과
[3] 포항공과대학교 산업공학과

Hyungjun Park[1], Hyung Bae Jung[2], and Kwangsoo Kim[3]
[1] Department of Industrial Engineering, Chosun University
[2] Department of Naval Architecture and Ocean Engineering, Mokpo National University
[3] Department of Industrial Engineering, Pohang University of Science and Technology

ABSTRACT

This paper addresses the problem of B-spline surface interpolation to serial contours, where the number of points varies from contour to contour. A traditional lofting approach creates a set of B-spline curves via B-spline curve interpolation to each contour, makes them compatible via degree elevation and knot insertion, and performs B-spline surface lofting to get a B-spline surface interpolating them. The approach tends to result in an astonishing number of control points in the resulting B-spline surface. This situation arises mainly from the inevitable process of progressively merging different knot vectors to make the B-spline curves compatible. This paper presents a new approach for avoiding this troublesome situation. The approach includes a novel process of getting a set of compatible B-spline curves from the given contours. The process is based on the *universal* parameterization [1,2] allowing the knots to be selected freely but leading to a more stable linear system for B-spline curve interpolation. Since the number of control points in each compatible B-spline curve is equal to the highest number of contour points, the proposed approach can realize efficient data reduction and provide a compact representation of a B-spline surface while keeping the desired surface shape. Some experimental results demonstrate its usefulness and quality.

1. INTRODUCTION

B-spline surface interpolation determines a B-spline surface passing through a set of data points, and has been intensively investigated [1-6]. The data points can be classified into three according to their structure: rectangular arranged points, serial contours, and scattered points. In this paper, we address the problem of B-spline surface interpolation to serial contours, where the number of contour points varies from contour to contour. Various devices for data acquisition may be used to return a sequence of contours. This task can be accomplished by a traditional lofting approach [5,6] which takes three main steps: B-spline curve interpolation to each contour; making the B-spline curves exactly compatible via degree elevation and knot insertion; getting a B-spline surface interpolating the curves via B-spline surface lofting. The approach is straightforward but tends to result in an astonishing number of control points in the B-spline surface. See Figure 1. This situation arises mainly from the inevitable process of progressively merging different knot vectors to make the B-spline curves defined on a common knot vector. It becomes more obvious as the number of the contours increases and the number of points varies severely from contour to contour.

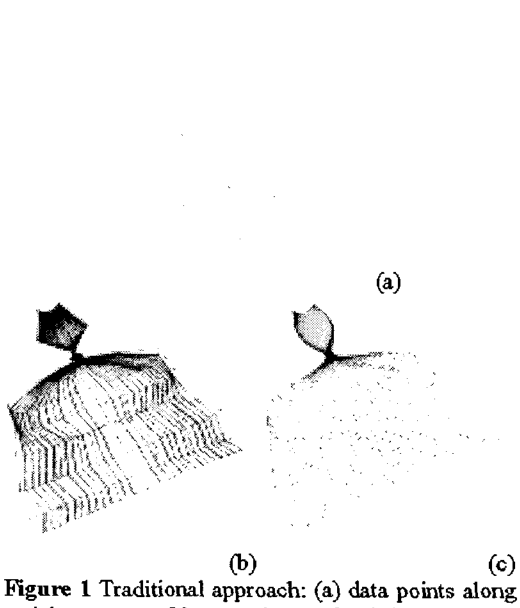


Figure 1 Traditional approach: (a) data points along serial contours; (b) control net of a lofted B-spline surface; (c) wireframe of the surface.

It is important to obtain a compact B-spline representation whose control points are kept small in their number. Such a representation speeds up most of downstream processes and leads to a decrease in storage requirements. The approximate approaches have been proposed for fixing this redundancy problem [7-11]. But they require much computation and result in a B-spline surface approximating the given contours. An approach was recently proposed to reduce the number of control points while maintaining precise interpolation [12]. It however requires a control parameter for assuring a numerically stable interpolation.

This paper presents a new approach providing a compact B-spline representation. The approach includes a novel process of creating a set of compatible B-spline curves from the given contours. The process is based on the *universal* parameterization [1,2] allowing the knots to be selected freely but leading to a more stable linear system for B-spline curve interpolation. Once determining a common knot vector, the approach does not need to increase the number of common knots afterwards. Since the number of control points in each compatible B-spline curve is equal to the highest number of contour points, the approach can thus realize efficient data reduction and provide a compact representation of the lofted B-spline surface without sacrificing the desired surface shape.

The rest of the paper is organized as follows. In Section 2, B-spline curves and surfaces, curve interpolation, and surface lofting are briefly described. In Section 3, the proposed approach is described in detail. In Section 4, experimental results are given to demonstrate the usefulness and quality of the approach. Section 5 closes the paper.

2. PRELIMINARIES

A parametric B-spline curve of order p can be defined as follows:

$$\mathbf{C}(t)=\sum_{i=0}^{m}\mathbf{b}_i N_{i,p}(t) \qquad (1)$$

where $\mathbf{b}_i$ are control points and $N_{i,p}(t)$ are the normalized B-spline functions of order p defined on a knot vector $\mathbf{T}=\{t_0,t_1,\ldots,t_{m+p-1},t_{m+p}\}$. Clamped knot vectors are prevalently used in most applications and defined as

$$\mathbf{T}=\{\underbrace{a,\ldots,a=\xi_0}_{p},\xi_1,\ldots,\xi_{m-p+1},\underbrace{\xi_{m-p+2}=b,\ldots b}_{p}\}$$

where the knots ξ_k $(k=0,\ldots,m-p+2)$ are called domain knots. The values a, b are given as 0 and 1, respectively. A $(p\times q)$th order B-spline surface is defined as follows:

$$\mathbf{S}(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}\mathbf{v}_{i,j}M_{j,q}(v)N_{i,p}(u) \qquad (2)$$

where $\mathbf{v}_{i,j}$ are control points and the B-spline functions $N_{i,p}(u)$ and $M_{j,q}(v)$ are defined respectively over $\mathbf{U}=\{u_0,u_1,\ldots,u_{m+p-1},u_{m+p}\}$ and $\mathbf{V}=\{v_0,v_1,\ldots,v_{n+q-1},v_{n+q}\}$.

B-spline curve interpolation [1,2,4,5,13] can be stated as the problem of constructing a B-spline curve passing through a set of distinct points $\mathbf{p}_i$ $(i=0,\ldots,m)$. When parameter values $\bar{t}_i$ of $\mathbf{p}_i$, the order p of B-spline functions, and a knot vector $\mathbf{T}$ of a B-spline curve $\mathbf{C}(t)$ are given, the problem leads to solving a set of linear equations:

$$\mathbf{p}_i=\mathbf{C}(\bar{t}_i)=\sum_{k=0}^{m}\mathbf{b}_k N_{k,p}(\bar{t}_i) \quad (i=0,\ldots,m) \qquad (3)$$

where $\mathbf{b}_k$ $(k=0,\ldots,m)$ are unknown control points and the system matrix is a $(m+1)\times(m+1)$ square matrix of scalars. The values $\bar{t}_i$ are ranged from a to b. The choice of the parameter values $\bar{t}_i$ and the knot vector $\mathbf{T}$ affects the shape and the parameterization of the curve $\mathbf{C}(t)$. Three common methods of choosing the $\bar{t}_i$ are equally spaced, chordal, and centripetal parameterizations [4,5,13]. Knots can be equally spaced, but it has been recommended to determine them based on the distribution of the parameter values $\bar{t}_i$. A common method is to space the domain knots ξ_i $(i=0,\ldots,m-p+2)$ by the averaging technique [4,5].

A different method for choosing the parameter values $\bar{t}_i$ has been proposed [1,2], where the knots are first selected and the parameter value at each data point is then determined to be the parameter value at the maximum of the associated B-spline function. The knots, for example, can be determined by applying the averaging technique to the chordal or centripetal parameters of the contour. The method, called *universal* parameterization, has a good advantage over the conventional one. It allows the knots to be selected freely but leads to a more stable linear system that is positive definite, banded, and non-singular. This parameterization is fully utilized in the proposed lofting approach.

Surface lofting, also known as skinning, is a process of passing a smooth surface through a set of cross-sectional curves [5,6]. When quite a number of curves are given, it might be preferable to approximate the curves [8-11]. We herein consider the task of interpolating through them. When the curves are represented in B-spline and made compatible beforehand, we can compute a B-spline surface $S(u,v)$ with ease as follows. After determining an order q, a knot vector **V**, and the parameter values of the curves in the v direction, compute the $(m+1)\times(n+1)$ control points $\mathbf{v}_{i,j}$ of the surface $S(u,v)$ by applying B-spline curve interpolation $(m+1)$ times [4,5].

3. B-SPLINE SURFACE INTERPOLATION

Let $\mathbf{P}_i$ $(i=0,..,n)$ denote the ith contour whose points are given as $\mathbf{p}_{i,j}$ $(j=0,..,m_i)$. Each contour is considered to be open and the number of contour points varies from contour to contour. B-spline surface interpolation to the contours can be accomplished by the traditional lofting approach [4-12]. Summarized below are its overall steps.

(Procedure 1) Traditional approach

(1) Choose an order p of B-splines and create a set of pth order B-spline curves by doing the following steps for each contour $\mathbf{P}_i$:
   [a] If $(m_i+1)<p$, construct an (m_i+1) order B-spline curve segment passing through the points and raise its order to p via degree elevation [4,5].
   [b] Otherwise, construct a pth order B-spline curve via B-spline curve interpolation to the contour points.

(2) By merging the knot vectors of the curves via knot insertion [4,5], make the curves compatible, that is, defined on a common knot vector $\mathbf{U}=\{u_0,...,u_{\hat{m}+p}\}$ with the order p.

(3) Generate a B-spline surface interpolating the compatible B-spline curves via B-spline surface lofting.

Any method for choosing knots and parameter values can be used in B-spline curve interpolation. This approach is straightforward but has a critical shortcoming. The B-spline curves are likely to be defined on different knot vectors in Step 1. Thus, the process of progressively merging the knot vectors tends to result in a significant increase in the number of common knots in Step 2 and, correspondingly, a bulky and redundant representation of the lofted B-spline surface in Step 3 [8,9,11]. A new approach is proposed for reducing the number of control points significantly while not sacrificing the resultant surface shape. Summarized below are its overall steps.

(Procedure 2) Proposed approach

(1) Choose an order p of B-splines and find the highest index $\hat{m}$ of contour points among m_i $(i=0,..,n)$.

(2) According to the order p and the index $\hat{m}$, determine a common knot vector $\mathbf{U}=\{u_0,...,u_{\hat{m}+p}\}$ with its appropriate domain knots $\hat{\xi}_k$ $(k=0,..,\hat{m}-p+2)$.

(3) Create a set of compatible B-spline curves defined on the knot vector $\mathbf{U}$ with the order p by do the following steps for each contour $\mathbf{P}_i$:
   [a] If $(m_i+1)<p$, construct an (m_i+1) order B-spline curve segment passing through the contour points, raise its order to p via degree elevation [4,5], and make the curve defined on the knot vector $\mathbf{U}$ via knot insertion [4.5].
   [b] If $(m_i=\hat{m})$, apply the universal parameterization and perform B-spline curve interpolation to get a B-spline curve of an order p defined on the knot vector $\mathbf{U}$.
   [c] If $(m_i<\hat{m})$, set up a knot vector $\mathbf{U}_i$ with domain knots ξ_j $(j=0,..,m_i-p+2)$ by appropriately selecting (m_i-p+3) knots from the knots $\hat{\xi}_k$ $(k=0,..,\hat{m}-p+2)$. Apply the universal parameterization and perform B-spline curve interpolation to get a B-spline curve of an order p defined on the knot vector $\mathbf{U}_i$. Make the curve defined on the knot vector $\mathbf{U}$ by inserting into the knot vector $\mathbf{U}_i$ the $(\hat{m}-m_i)$ complementary knots $\hat{\xi}_k\in(\mathbf{U}-\mathbf{U}_i)$.

(4) Generate a B-spline surface interpolating the compatible B-spline curves via B-spline surface lofting.

In Step 2, the common knots can be spaced uniformly. However, they can be determined by the following steps: Compute chordal or centripetal parameter values for each of the contours with the index $\hat{m}$, average these parameter values, and determine the common knots for order p via the averaging technique [4,5].

Once the common knot vector has been determined, no more increase occurs in the number of common knots afterwards. This is supported by the universal parameterization method allowing the knots to be selected freely but leading to a more stable linear system for B-spline curve interpolation [2]. It means that each B-spline curve can be created on a knot vector that is the subset of the predetermined common knot vector. The curve can be made compatible with the common knot vector by inserting the complementary knots into the knot vector. Note that the number of control points in each compatible B-spline curve is equal to the highest number of contour points. The proposed approach can thus avoid the data redundancy residing in the traditional approach.

In Step 3(c), it is required to determine local knots ξ_j by choosing $(m_i - p+3)$ knots from the $(\hat{m} - p+3)$ knots $\hat{\xi}_k$ of the common knot vector **U** in a proper way. Consider temporary knots $\tilde{\xi}_j$ $(j = 0,..,m_i - p+2)$ for the contour $\mathbf{P}_i$. They can be spaced uniformly or determined by averaging the chordal or centripetal parameter values of the $\mathbf{P}_i$. The local knots ξ_j are determined by the following procedure with $\hat{\xi}_k$ $(k = 0,..,\hat{m} - p+2)$ and $\tilde{\xi}_j$ $(j = 0,..,m_i - p+2)$.

(Procedure 3) Local knot spacing

Input: common knots $\hat{\xi}_k$ $(k = k_s,..,k_e)$ and temporary knots $\tilde{\xi}_j$ $(j = j_s,..,j_e)$

Output: local knots ξ_j

(1) $\hat{n} = k_e - k_s + 1$ and $\tilde{n} = j_e - j_s + 1$. If $\tilde{n} = \hat{n}$, $\xi_j \leftarrow \hat{\xi}_j$ for all j and quit the procedure.

(2) $d \leftarrow \hat{n} - \tilde{n}$, $h \leftarrow j_s + \text{int}\left(\frac{\tilde{n}-1}{2}\right)$, and $k_m \leftarrow k_s + \text{int}\left(\frac{\tilde{n}-1}{2}\right)$. Among the knots $\hat{\xi}_k$ $(k = k_m,..,k_m + d)$, find the knot $\hat{\xi}_g$ closest to the knot $\tilde{\xi}_h$.

(3) $\xi_h \leftarrow \hat{\xi}_g$. If $\tilde{n} = 1$, quit the procedure.

(4) If $h-1 \ge j_s$, invoke the procedure with $\hat{\xi}_k$ $(k = k_s,..,g-1)$ and $\tilde{\xi}_j$ $(j = j_s,..,h-1)$.

(5) If $h+1 \le j_e$, invoke the procedure with $\hat{\xi}_k$ $(k = g+1,..,k_e)$ and $\tilde{\xi}_j$ $(j = h+1,..,j_e)$.

The procedure progressively divides two sets of common and temporary knots nearly at the connection between the median temporary knot and its closest common knot such that the gap between temporary knots and their corresponding common knots should be possibly minimized. Though this procedure does not guarantee the global optimal solution, it provides a reasonable one adequate for the proposed lofting approach. Figure 2 shows an example of local knot spacing that has occurred during B-spline curve interpolation to the 9th contour displayed in Figure 1(a). The contour consists of 9 points ($m_i = 8$).

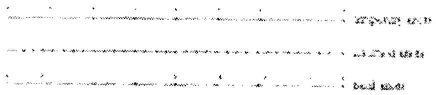


Figure 2 Example of local knot spacing.

The temporary knots are determined by averaging the centripetal parameter values, and the common knots are computed from the contours with $\hat{m} = 34$. The local knot vector is computed via Procedure 3 and the parameter values are determined by the universal parameterization.

Figure 3 shows an application of cubic B-spline curve interpolation to the 9th contour. The universal parameter values and the local knot vector are displayed at the lower part of each figure. Figure 4 shows an application of knot insertion to the B-spline curve. Note that 26 complementary knots are inserted into the local knot vector. The universal parameter values and the common knot vector are displayed at the lower part of each figure.

(a) (b)

Figure 3 Curve interpolation using local knots: (a) B-spline curve; (b) its control polygon.

(a) (b)

Figure 4 Knot insertion: (a) B-spline curve compatible with the common knot vector; (b) its control polygon.

4. EXPERIMENTAL RESULT

The proposed approach has been tested for various sets of contour data. In this paper, two sets are included to demonstrate the usefulness and quality of the approach. In the test, bicubic $(p = q = 4)$ B-spline surfaces are used for interpolating the contours.

Firstly, 81 contours were scanned from the surface of a dust pan. Table 1 shows the comparison of the proposed approach with the traditional approach for four different contour numbers. Centripetal parameterization and the averaging technique are applied in the traditional approach and, partially, in the proposed approach. Figure 1 shows the application of the traditional approach to 11 contours. The number of points per contour ranges from 5 to 35. Figure 5 shows the application of the proposed approach to the same contour data. Compare Figure 5 with Figure 1. The proposed approach requires 35×11 control points to interpolate the data, whereas the traditional approach needs 187×11 control points.

Table 1 Results of test examples for the surface of a dustpan.

n	10	20	40	80
min m_i	4	4	4	4
max m_i	34	34	34	34
Traditional approach $\hat{m}$	186	361	713	1410
Proposed approach $\hat{m}$	34	34	34	34

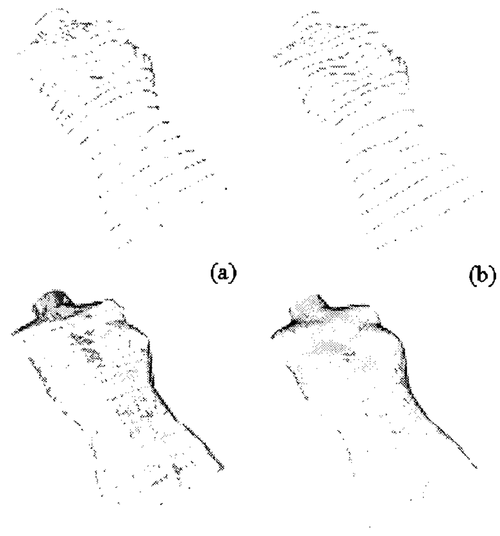
(a) (b)

Figure 5 Proposed approach: (a) control net of a lofted B-spline surface; (b) wireframe of the surface.

Secondly, 81 contours were scanned from the front surface of a woman body. With the traditional approach and Piegl and Tiller's approach [12], the proposed approach was tested for four different contour numbers. Table 2 summarizes the results. Note that the scale factor *per* is used to tune the number of knots in a common knot vector. A large value *per* tends to decrease the number of knots and a small value tends to increase their number. Chordal parameterization and the averaging technique are applied in the previous approaches and, partially, in the proposed approach. Figure 6 shows the application of the proposed approach to 21 contours. The number of points per contour ranges from 27 to 79. The proposed approach requires 79×21 control points to interpolate the data, whereas the traditional approach needs 1081×21 control points as shown in Table 2.

Table 2 Results of test examples for the front surface of a woman body

n		10	20	40	80
min m_i		29	26	26	26
max m_i		78	78	78	78
Traditional approach $\hat{m}$		566	1080	2127	4212
Piegl and Tiller's approach $\hat{m}$	$per = 1.00$	84	89	93	99
	$per = 0.75$	113	127	133	140
	$per = 0.50$	149	168	177	188
	$per = 0.25$	246	292	332	359
Proposed approach $\hat{m}$		78	78	78	78



(c) (d)

Figure 6 Lofted B-spline surface interpolation applied to serial contours of a human body: (a) data points along the contours; (b) B-spline curves made compatible; (c) control net of a lofted B-spline surface; (d) wireframe of the surface.

5. CONCLUSION

This paper has proposed a new approach for lofted B-spline surface interpolation to serial contours, where the number of points varies from contour to contour. The approach includes a novel process of getting a set of compatible B-spline curves from the given contours. The process is supported by the universal parameterization [1,2] allowing the knots to be selected freely but leading to a more stable linear system for B-spline curve interpolation. The number of control points in each compatible B-spline curve is equal to the highest number of contour points. From the experimental results, the proposed approach requires much fewer control points than previous approaches and provides a compact representation of a lofted B-spline surface while keeping the desired surface shape.

REFERENCES

[1] C. G. Lim, "A universal parameterization in B-spline curve and surface interpolation", Computer Aided Geometric Design, 16, pp. 407-422, 1999.

[2] H. B. Jung and K. Kim, "A new parameterisation method for NURBS surface interpolation", International Journal of Advanced Manufacturing Technology, 16, pp. 784-790, 2001.

[3] G. Farin, Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide, Academic Press, 1993.

[4] J. Hoschek and D. Lasser, Fundamentals of Computer Aided Geometric Design, A K Peters, 1993.

[5] L. Piegl and W. Tiller, The NURBS Book, Springer-Verlag, New York, 1995.

[6] C. Woodward, "Skinning techniques for interactive B-spline surface interpolation", Computer-Aided Design, 20 (8), pp. 441-451, 1988.

[7] H. Park and K. Kim, "Smooth surface approximation to serial cross-sections", Computer-Aided Design, 28 (12), pp. 995-1005, 1996.

[8] L. Piegl and W. Tiller, "Algorithm for approximate NURBS skinning", Computer-Aided Design, 28 (9), pp. 699-706, 1996.

[9] H. Park, K. Kim, and S. C. Lee, "A method for approximate NURBS curve compatibility based on multiple curve refitting", Computer-Aided Design, 32 (4), pp. 237-252, 2000.

[10] H. Park, "An approximate lofting approach for B-spline surface fitting to functional surfaces", International Journal of Advanced Manufacturing Technology, 18 (7), pp. 474-482, 2001.

[11] L. Piegl and W. Tiller, "Surface skinning revisited", Visual Computer, 18, pp. 273-283, 2002.

[12] L. Piegl and W. Tiller, "Reducing control points in surface interpolation", IEEE Computer Graphics and Applications, 20, pp. 70-74, 2000.

[13] E. Y. T. Lee, "Choosing nodes in parametric curve interpolation", Computer-Aided Design, 21(6), pp. 363-370, 1989.