

## An interpolation method of b-spline surface for hull form design

Hyung-Bae Jung

Mokpo National University, Korea

**ABSTRACT:** *This paper addresses the problem of B-spline surface interpolation of scattered points for a hull form design, which are not arbitrarily scattered, but can be arranged in a series of contours permitting variable number of points in the contours. A new approach that allows different parameter value for each point on the same contour has been adopted. The usefulness and quality of the interpolation has been demonstrated with some experimental results.*

**KEY WORDS:** Parameterization; Knot insertion; Degree elevation.

### INTRODUCTION

A common problem encountered in geometrical design is interpolation of scattered data points  $P_{ij}$  with corresponding parameter values  $t_{ij}$  to find a surface that passes through the points  $P_{ij}$ . Skinning is a special process of surface generation based on the interpolation of given set of curves. In one sense, it is very similar to lofting where the surface interpolates a family of predefined curves (Woodward, 1988). But in another, it is more specific method for B-spline representations. Recently, the skinning based on Non-uniform Rational Basis Spline (NURBS) is commonly used to utilize the advantages of NURBS.

A traditional skinning approach first creates a set of B-spline curves in u-direction through B-spline curve interpolation to each contour, and makes them compatible via degree elevation and knot insertion. Secondly, it performs B-spline surface skinning in v-direction to get a B-spline surface by making all points on each contour to have same parameter value using averaging method. The present paper overcomes the problem of B-spline surface interpolation to non-arbitrarily scattered points by arranging them in series of contours. The new approach permits different parameter value for each point on the same contour by using the compatible process in v-direction for the surface interpolation. But this method can result in an astonishing number of control points in the resulting B-spline surface. This situation arises due to the inevitable process of progressive merging of different knot vectors to make the B-spline curves compatible. The novelty of the present approach lies in its use of universal parameterization, allowing the knots to be selected

freely (Jung, 1998; Lim, 1999; Jung et al., 2000).

Since the universal parameterization permits that the number of control points in v-direction is equal to the number of contours, the proposed approach can realize efficient data reduction and provide a compact representation of B-spline surface, retaining the desired surface shape.

In order to skin across curves of various types, they need to be made compatible, i.e. they all need to (i) be rational or non-rational; (ii) have the same degree; (iii) be defined over the same knot vector. The present approach satisfies the first two conditions without any problem, i.e. we can unify the representation form and use the algorithms of degree elevation. The third condition can be fulfilled through knot insertion. Although the present approach tends to result in an astonishing number of control points of the B-spline surface, it processes in both u- and v-direction, to minimize the number of control points.

It is important to obtain a compact B-spline representation with small number of control points. Such a representation speeds up most of the downstream processes and leads to a decrease in storage requirements. The approximate approaches have been proposed to fix this redundancy problem (Park et al., 1996; Piegl et al., 1996; Park et al., 2000; Park, 2001; Piegl et al., 2002).

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 (Piegl et al., 2000). However, it requires a control parameter for assuring a numerically stable interpolation.

This paper resolves the problem of B-spline surface interpolation to non-arbitrarily scattered points by arranging them in serial contours.

---

Corresponding author: *Hyung Bae Jung*  
e-mail: [hbjung@mokpo.ac.kr](mailto:hbjung@mokpo.ac.kr)

**B-SPLINE CURVE INTERPOLATION**

B-spline curve interpolation can be stated as the problem of constructing a B-spline curve passing through a set of distinct points  $Q_i$  ( $i = 0, \dots, m$ ). When parameter values  $u_i$  of  $Q_i$ , the order  $p$  of B-spline functions, and a knot vector  $T$  of a B-spline curve  $C(u)$  are given, the problem leads to solving a set of linear equations:

$$Q_i = C(u_i) = \sum_{k=0}^m P_k N_{k,p}(u_i) \quad (i = 0, \dots, m) \tag{1}$$

where  $P_k$  ( $k=0, \dots, m$ ) are unknown control points and the system matrix is a  $(m+1) \times (m+1)$  square matrix of scalars.

$$\begin{bmatrix} N_{0,p}(\bar{u}_0) & \dots & N_{m,p}(\bar{u}_0) \\ N_{0,p}(\bar{u}_1) & \dots & N_{m,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N_{0,p}(\bar{u}_m) & \dots & N_{m,p}(\bar{u}_m) \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_m \end{bmatrix} = \begin{bmatrix} Q_0 \\ Q_1 \\ \vdots \\ Q_m \end{bmatrix} \tag{2}$$

The choice of the parameter values  $u_i$  and the knot vector  $U$  affects the shape and the parameterization of the curve  $C(u)$ . Four common methods of choosing the  $u_i$  are

① Equally spaced:

$$\begin{aligned} u_0 &= 0, u_n = 1 \\ u_k &= \frac{k}{n}, k = 1, \dots, n-1 \end{aligned} \tag{3}$$

② Chord length :

$$\begin{aligned} u_0 &= 0, u_n = 1 \\ u_i &= u_{i-1} + \frac{|Q_i - Q_{i-1}|}{\sum_{j=1}^{n-1} |Q_j - Q_{j-1}|} \end{aligned} \tag{4}$$

③ Centripetal method :

$$\begin{aligned} u_0 &= 0, u_n = 1 \\ u_i &= u_{i-1} + \frac{|Q_i - Q_{i-1}|^{\frac{1}{2}}}{\sum_{j=1}^{n-1} |Q_j - Q_{j-1}|^{\frac{1}{2}}} \end{aligned} \tag{5}$$

④ Universal method:

The parameter value at each data point is chosen to be the parameter value at the maximum of the associated rational basis function  $u_i = N_{i,p}(u)$

**B-SPLINE SURFACE INTERPOLATION**

When the curves are represented in B-spline, a B-spline

surface  $S(u, v)$  can be easily computed. 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  $P_{i,j}$  of the surface  $S(u,v)$  by applying B-spline curve interpolation  $(m+1)$  times which is presented in following equation.

$$\begin{aligned} Q_{k,l} &= \sum_{i=0}^m \sum_{j=0}^n P_{i,j} N_{i,p}(u) N_{j,q}(v) \\ &= \sum_{i=0}^m [ \sum_{j=0}^n P_{i,j} N_{i,p}(v_l) ] N_{j,q}(u_k) \\ &= \sum_{i=0}^m C_i(v_l) N_{j,q}(u_k) \end{aligned} \tag{6}$$

Each contour is considered to be open and the number of contour points varies from one contour to the other. B-spline surface interpolation to the contours can be accomplished by the traditional skinning approach.

**Traditional approach**

Two traditional skinning procedures are available in the literature. Procedure 1, proposed by Hohmeyer and Barsky (1991) is presented below.

**Procedure 1:**

(1) Choose an order  $p$  of B-splines and create a set of  $p^{\text{th}}$  order B-spline curves by doing the following steps for each contour  $P_i$ :

- [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.
- [b] Otherwise, construct a path order B-spline curve via B-spline curve interpolation to the contour points.

(2) By merging the knot vectors of the curves through knot insertion, make the curves compatible, i.e. defined on a common knot vector  $U = \{u_0, \dots, u_{\hat{m}+p}\}$  with the order  $p$ .

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

Procedure 2, proposed by Park et al. (2004) is presented below.

**Procedure 2:**

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

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

(3) Create a set of compatible B-spline curves defined on the knot vector  $\mathbf{U}$  with the order  $\rho$  by doing the following steps for each contour:

- [a] If  $(m_i+1) < \rho$  construct an  $(m_i+1)$  order B-spline curve segment passing through the contour points, raise its order to  $p$  via degree elevation, and make the curve defined on the knot vector  $\mathbf{U}$  via knot insertion.
- [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  $\xi_j^i (j = 0, \dots, m_i-p+2)$  by appropriately selecting  $(m_i-p+3)$  knots from the knots  $\xi_k^i (k = 0, \dots, \hat{m}-p+2)$ . 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 the  $(\hat{m} - m_i)$  complementary knots  $\xi_k^i \in (\mathbf{U}-\mathbf{U}_i)$  into the knot vector  $\mathbf{U}_i$ .

(4) Generate a B-spline surface, interpolating the compatible B-spline curves, through B-spline surface skinning.

Procedure1 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 increase the number of common knots in Step 2, and consequently caused a bulky and redundant representation of the skinned B-spline surface in Step 3. Procedure 2 is proposed for reducing the number of control points significantly without sacrificing the resultant surface shape. But being the latest approach, it needs further experimental verification.

**Proposed approach**

(1) Choose an order  $p$  of B-splines and create a set of  $p$ th order B-spline curves by doing the following steps for each contour:

- [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  $p$  through degree elevation .
- [b] Otherwise, construct a  $p$ th order B-spline curve via B-spline curve interpolation to the contour points.

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

(3) Calculate control points for each curve.

(4) In  $v$ -direction with each control points, choose an order  $q$  of B-splines, and determine a common knot vector  $\mathbf{V} = \{v_0, \dots, v_{n+q}\}$  with its appropriate domain knots  $\xi_k^i (k = 0, \dots, n-$

$q+2)$  and with the universal parameterization of equally spaced points.

(5) Generate a B-spline surface.

The universal parameterization is applied to determine the knot vector  $\mathbf{V}$  and the parameter values  $v_k$ , i.e. the knots are freely selected, whereas the parameter values correspond to the maximum value of B-spline basis function.

This approach has advantage in skinning of relatively scattered data in  $v$ -direction. The averaging method has problem to reflect the characteristics of scattered data. In the case of serial contours, the number of contour points varies from one contour to the other but the distance between points in  $v$ -direction doesn't vary. This method has no additional advantage compared to traditional approach.

**EXPERIMENTAL RESULTS**

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

This approach is tested for generating the hull form surface of ship. The design data for ship hull form, obtained using SIKOB program, after initial design of a real ship is shown in Table 1. The same data can be arranged as shown in Fig. 1.

Table 1 Design data from SIKOB.

1	215	32.2	17.8	107.5	24	0	0
\$							
\$	frame	Space					
	11	42	242				
0.75	0.8	0.86	0.75				
\$							
\$	ship	Contour					
1	0	-4.5	2	19			
1	0	-4.5	2	10.95			
1	0	0	2	9.91			
1	0	3.501	2	9			
1	0	75	2	8.207			
1	0	5.702	2	8			
1	0	6.579	2	7			
1	0	6.68	2	6			
1	0	6.08	2	5			
1	0	4.9	2	4.39			
1	0	4.9	2	3.34			
1	0	5.375	2	3.09			
1	0	6.503	2	2			
1	0	8.25	2	1			
.....							

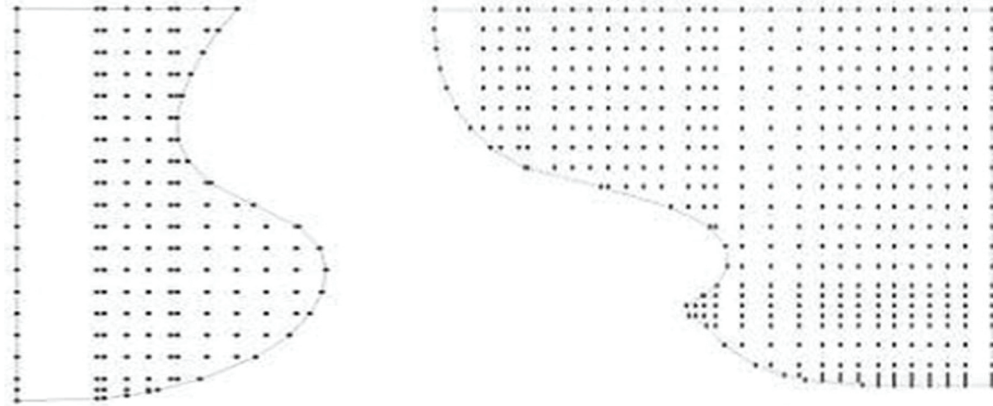


Fig. 1 Data for skinning.

Two surfaces as shown in Fig. 2 are generated with the data of Fig. 1. The left side represents fore-body of a ship and right side is the after-body. We omitted the procedure for mid-body because its contour is regular and the surface for midbody can be easily generated. Especially, the fore-body and after-body are compound surfaces, and a lot of difficulties arise during generation of smooth surface for these areas. For generating Fig. 2, we used chord length method for parameterization and knot insertion procedure

for compatibility in u-direction. We used universal method for parameterization in for v-direction.

The geometry of ship hull surfaces is traditionally checked by a set of curves, called ship lines plan. The lines plan primarily consists of planar curves, waterlines, sections and buttock lines, lying in orthogonal planes parallel to the coordinate axes. Fig. 4 is the lines plan for Fig. 2. This lines plan demonstrates that the surface fairness is good.

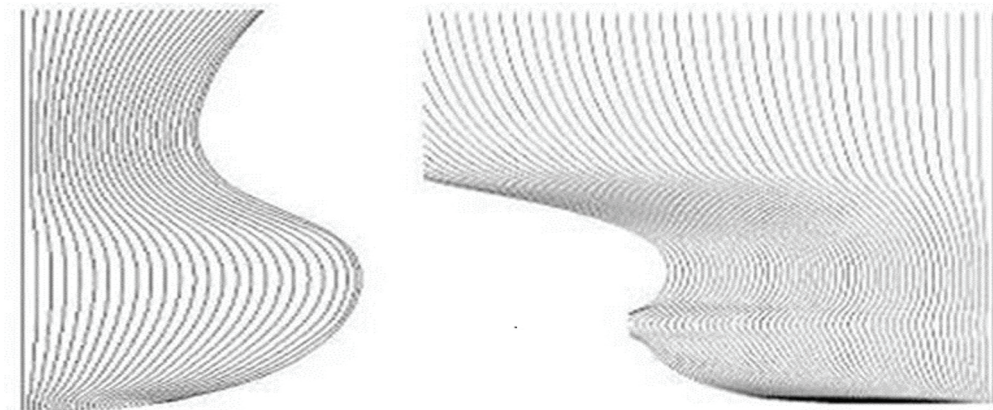


Fig. 2 Iso-parametric plan with u-constant at xz-plane.

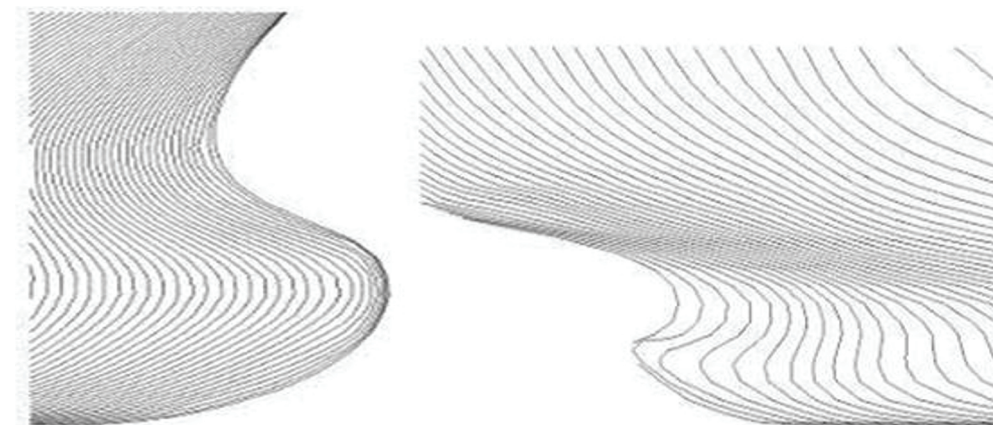


Fig. 3 Buttock lines (Sheer plan).



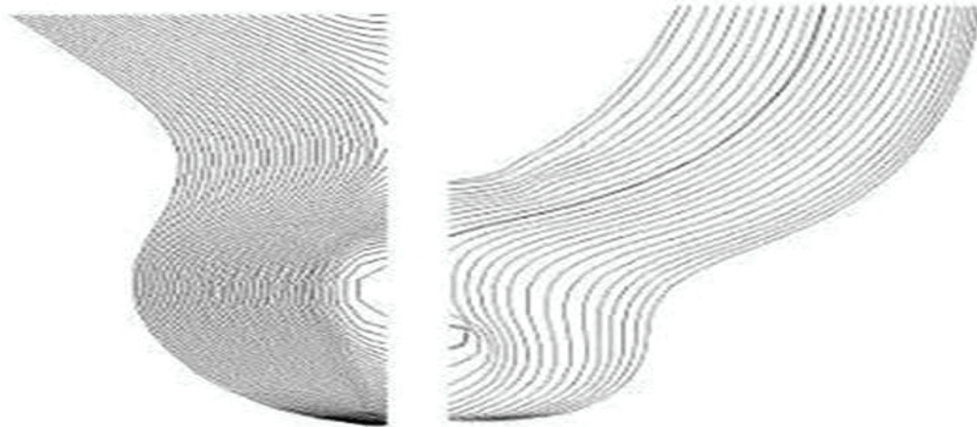


Fig. 4 Sections lines (Body plan).

## CONCLUSION

This paper presents a new skinning approach for ship hull form. The proposed approach requires reducing the control points than previous ones and providing a compact representation of a B-spline surface, while maintaining the desired surface shape. However, there is room for improving this method through intensive investigation of the effects of knot vector selection.

## REFERENCES

- Hohmeyer, M. and Barsky, B., 1991. Skinning rational B-spline curves to construct an interpolatory surfaces. *Comput. Vis. Graph and Image Processing*, 53, pp. 511-521.
- Jung, H. B., 1998. A new method of the global interpolation in NURBS surface: II. *Journal of Korean CAD/CAM engineers*, 3(4), pp. 243-250.
- Jung, H. B. and Kim, K., 2000. A new parameterization method for NURBS surface interpolation. *International Journal of Advanced Manufacturing Technology*, 16, pp. 784-790.
- Park, H. and Kim, K., 1996. Smooth surface approximation to serial cross-sections. *Computer-Aided Design*, 28(12), pp. 995-1005.
- Park, H. Kim, K. and Lee, S.C., 2000. A method for approximate NURBS curve compatibility based on multiple curve refitting. *Computer-Aided Design*, 32(4), pp. 237-252.
- Piegl, L. and Tiller, W., 1996. Algorithm for approximate NURBS skinning. *Computer-Aided Design*, 28(9), pp. 699-706.
- Piegl, L. and Tiller, W., 2000. Reducing control points in surface interpolation. *IEEE Computer Graphics and Applications*, 20, pp. 70-74.
- Piegl, L. and Tiller, W., 2002. Surface skinning revisited. *Visual Computer*, 18, pp. 273-283.
- Woodward, C., 1988. Skinning techniques for interactive B-spline surface interpolation. *Computer-Aided Design*, 20(8), pp. 441-451.
- Lim, C. G., 1999. A universal parameterization in B-spline curve and surface interpolation. *Computer Aided Geometric Design*, 16, pp. 407-422.
- Park, H., 2001. An approximate lofting approach for B-spline surface fitting to functional surfaces. *International Journal of Advanced Manufacturing Technology*, 18(7), pp. 474-482.
- Park, H. Jung, H. B. and Kim, K., 2004. A new approach for lofted B-spline surface interpolation to serial contours. *International Journal of Advanced Manufacturing Technology*, 23, pp. 889-895.