Least Square B-Spline Fitting For Surface Measurement

Jong-Yun Jung*, Lisheng Li+, Choon-Man Lee++, Won-Jee Chung++

(논문접수일 2003. 2. 6, 심사완료일 2003. 3. 10)

곡면 측정을 위한 최소 자승 비-스플라인 Fitting

정종윤*, 리리셍+, 이춘만++, 정원지++

Abstract

An algorithm for fitting with Least Square is a traditional and an effective method in processing with experimental data. Due to the lack of definite representation, it is difficult to fit measured data with free curves or surfaces. B-Spline is usefully utilized to express free curves and surfaces with a few parameters. This paper presents the combination of these two techniques to process the point data measured from CMM and other similar instruments. This research shows tests and comparison of the simulation results from two techniques.

Key Words : B-Spline, Least square fitting, Measurement

1. Introduction

A Coordinate Measuring Machine (CMM) is used to sample discrete data from a surface of a part. Since the sampled data are discrete, they do not give detail information for missed area. A mathematical function is required to supply enough information of a curve or a surface[1]. This can be completed with Least Square Best Fitting. A few parameters should be properly selected so that they can describe the characteristics of a geometrical feature, including its position, direction, size and even its shape and errors[2]. It is difficult to fit the measured data to free form curves and surfaces. B-spline, due to its excellent properties, has become one of the most popular methods to express free form curves.

The main concerning points in Computer Aided Geometric Design (CAGD) are smoothness, continuity, aesthetical shapes of curves and surface. But the main concerning points in metrology are how to eliminate or to reduce the influence of random factors from the

* 주저자, Changwon National University, Dept. of Industrial & Systems Engineering (jyjung@sarim.changwon.ac.kr)
주소: 641-773 경남 창원시 사림동 9
\+ Shanghai Machine Tool Research Center
++ Dept. of Mechanical Design & Manufacturing Engineering, Changwon National University

measured data as possible. This paper presents a method satisfying these two targets to build mathematical forms with the point data measured from CMM.

2. Background

2.1 Least Square

A linear equation

$$Ax=y \tag{1}$$

where $A=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{bmatrix}$ is a coefficient matrix;

$X=[x_1 \ \ x_2 \ \cdots \ x_n]^T$ is an n dimensional vector to be estimated; $Y=[y_1 \ \ y_2 \ \cdots \ y_m]^T$ is an m dimen sional observed vector. In metrology, X denotes the parameters to be estimated, and Y is measured data. When m is greater than n, that is, the number of measured data points is bigger than the number of parameters, X can be estimated by resolving the following normal equations:

$$A^TAX=A^TY \tag{2}$$

However, when **X** includes nonlinear equations instead of linear equation (1), the coefficient matrix **A** should be obtained with calculating partial derivates. In addition a suitable recursive-optimizing algorithm is required to acquire the derivates. This research introduces this method to find unknown n dimensional vector when the number of unknowns are greater than that of knowns in the linear systems.

2.2 B-spline curve fitting

Curves and surfaces are expressed with parametric formula in CAGD. One of them is based on the B-spline basis function. B-spline curve is expressed with the following equation:

$$P(u)=\sum_{i=0}^{n} d_i \ N_{i,k}(u) \tag{3}$$

where d_i, $i=0,1,\cdots,n$ are control vertices, or de Boor points; B-spline control polygon is lines connected through control vertices in order. $N_{i,k}(u)$, $i=0,1,\cdots,n$ are B-spline basis functions of k degree. P(u) is the point on the curve determined with the parameter value u in Fig. 1 The de Boor recursive formula [3] is

$$P(u)=\sum_{j=i-k+1}^{i} d_j^l \ N_{j,k-1}(u)=\cdots=d_j^k$$

$$d_j^l=\begin{cases} d_j & l=0 \\ (1-\alpha_j^l)\,d_{j-1}^{l-1}+\alpha_j^l\,d_j^{l-1} & l=1,2,\cdots,k \\ & j=i-k+l,\cdots,i \end{cases}$$

$$\alpha_j^l=\frac{u-u_j}{u_{j+k+1-l}-u_j} \tag{4}$$

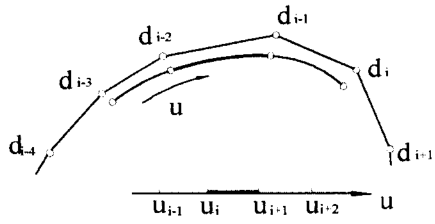

Fig. 1 B-spline and its parameters

where u with subscript is a knot value in parameter domain and u is a variable in the ith segment. The points on the curve can be expressed as the function of parameter $u\in[u_i \ , \ u_{i+1})\subset[u_k \ , \ u_{n+1}]$. P(u) is a recursive interpolation of control points d_j in Fig. 2[4].

When u is on the knot point u_i in the parameter domain, k control vertices d_j $j=i-k, i-k+1,\cdots,$

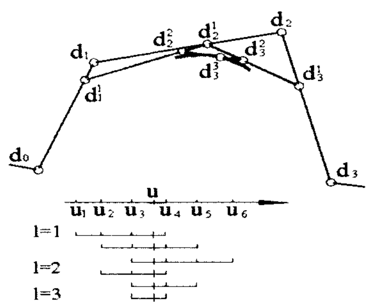

Fig. 2 Recursive proportional interpolation of B-Spline

$i-1$ are associated with the curve. For other u, k+1 control vertices are concerned. This is local property of B-spline.

In the application of metrology, the Ps are observed vectors; the control points d_j are the parameters to estimate. The combination of u with α_j^l in Equation (4) is the coefficient matrix. This is reverse calculation of B-spline. But the coefficient matrix for reverse calculation in CAGD is derived from the pure knot points.

2.3 B-spline surface fitting

Control grids should be constructed to perform surface fitting that is topologically homologous to data points in a matrix form, instead of the control polygon in a row for curve fitting. Surface fitting is more complex than curve fitting, since many control points are required in surface fitting. It can be simplified into double parameter B-spline function in a specific situation. The following equation is the double parameter form of the function[5].

$$P(u,v)=\sum_{i=0}^{m+k-1}\left(\sum_{j=0}^{n+l-1} d_{i,j}\, N_{j,l}(v)\right) N_{i,k}(u) \qquad (5)$$

Comparing the equation (5) to the equation (3), we can find that the d_i in equation (3) is replaced by $\sum_{j=0}^{n+l-1} d_{i,j}N_{j,l}(v)$ in equation (5).

It means that the control grids can be constructed in two steps. First, fixing *v* values which are homologous to the orthogonal to *u* direction in parameter domain to fit (n+l) control polygons of the control curves. For each *v* value, being iso-parametric, we can get (m+k) control points in *u* direction. Then fixing *u* values to fit the final (m+k)x(n+l) control vertices. These steps are represented in Fig. 3 In this procedure, some additional techniques are required to build control curves from section curves, such as inserting or deleting knot points in parameter domain, degree elevation of B-spline curve, etc. From Fig. 3, we can see that the section curves lay in *u* direction on the surface. But the temporary control vertices $\bar{d}_{i,j}$ with same *u* and several *v* values construct control curves that are not on the surface. In fact, the control curves are the enveloping curves of these temporary control vertices from the fitting of u direction.

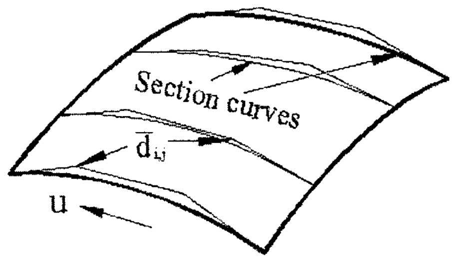

Fig. 3(a) Section curves and their control polynomials

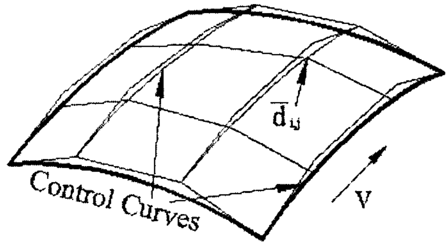

Fig. 3(b) Control curves and their control polynomials

The original data point array is not only limited in rectangular layout. It can be other type, for example, the (ρ, r) in a polar coordinate system is a nice combination to transform topologically into rectangular matrix format.

In engineering application, the cubic degree is popularly used to form B-spline surface, that is, k=l=3. The cubic degree B-spline curve and bi-cubic B-spline surface can represent inflection on curves and surfaces [6].

3. Least Square B-Spline fitting for cubic curves

According to equation (2) and its conditions, the number of measured points must be greater than the number of control points to be estimated. The curve can be divided into two segments arbitrary. Both ends of each segment cannot be on the surface. All the surface points are employed in estimating control vertices. From de Boor recursive formula in equation (4), surface points P are the linear function of *u*. We can therefore get the normal equations in the form of equation (2) and equation (1) without calculating partial derivates. Being different from reverse calculation of B-spline in CAGD, the u_i values that are both homologous to the surface points and located inside the segments are employed in calculating matrix A.

3.1 Naming conventions

Suppose the number of surface points P_j is m+1 (j=0,1,....,m); the curve is demarcated into n segments of which the boundary points are B_k (k=0,1,....,n). For cubic curve, n+3 control points d_i (i=0,1,....,n+2) are to be estimated; and the region of definition in parameter domain is $u \in [u_i, u_{i+1}) \subset [u_3, u_{n+3}]$. Being consistent with the conventions, the order of curve segments is numbered from 3 to n+2. In the i-th segment there are m_i (i=3,4,....,g_i,....,n+2) surface points. M_i is the number of surface points and g_i is the order of these points.

For open curve, the control points in both ends are conventionally located on the both ends of the curve, i.e., corresponding to u_3 and u_{n+3}, respectively. The multiplicity of both ends is 4 respectively. That is,

$$u_0 = u_1 = u_2 = u_3 = 0,$$
$$u_{n-6} = u_{n+5} = u_{n+4} = u_{n+3} = 1$$

If the segments are extended outside on both ends, then the m of the curve segment is,

$$m_0 = m_1 = m_2 = m_{n+3} = m_{n+4} = m_{n+5} = 0$$

The length of these segments is equal to 0. This is for unifying the expressions of formula addressed next section and its program is simple. For the open curve,

$$m+1 = \sum_{i=0}^{m+3} m_i.$$

From the similarity of cyclical close curve we draw the following equation:

$$\begin{cases} u_2 = u_{n+2} - 1, & u_{n+3} = u_3 + 1 \\ u_1 = u_{n+1} - 1, & u_{n+4} = u_4 + 1 \\ u_0 = u_n - 1, & u_{n+5} = u_5 + 1 \\ m_2 = m_{n+2}, & m_{n+3} = m_3 \\ m_1 = m_{n+1}, & m_{n+4} = m_4 \\ m_0 = m_n, & m_{n+5} = m_5 \end{cases}$$

3.2 Calculation of coefficients of reverse equations

Equation (4) is the same format with the following equation[7]

$$P(u_{g_i,i}) = d_{i-3} N_{i-3,3}(u_{g_i,i}) + d_{i-2} N_{i-2,3}(u_{g_i,i}) + d_{i-1} N_{i-1,3}(u_{g_i}, i) + d_i N_{i,3}(u_{g_i}, i) \quad (6)$$

where $u_{g_i,i}$ is the u value homologous to the gi-th point in i-th segment, or is simplified to u_{g_i}.

Set $C_{g_i}{}^l = N_{i-l,3}(u_{g_i}) / \triangle_i$ $1 = 3, 2, 0$

where $\triangle_i = u_{i+1} - u_i$.

The B-spline basis function can be developed further and $N_{i-l,3}(u_{g_i})$ are simplified, we can get the coefficients of reverse equations as in equation (7).

All the denominators are related to the knot points, and the numerators are concerned with u_{gi}. From the combination of equation (6) and (7), we get the coefficient equations in the equation (8). This coefficient matrix is (m+1)x(n+3) order which is A matrix in equation (1). The blank elements are all 0.

3.3 Calculation of coefficients of normal equations

From the equation (2), we get the normal equations for least square B-spline fitting. This is extended matrix of order (n+3)x(n+6). The last 3 columns in the matrix are observed vectors. The former n+3 columns belong to (n+3)x(n+3) symmetric coefficients matrix that is 7 diagonals matrix. The elements of q-th (q=1,2,.....,n+3) row of the whole extended matrix are listed in equation (9) on page 6.

The following equation is a supplementary to the equation (9).

$$A_{2,2} = \sum_{s=3}^{0}\sum_{k=1}^{m_{s+1}} C_{k,s+1}^{3-s} C_{k,s+1}^{3-s}$$
$$= \sum_{k=1}^{m_4} C_{k,4}^{0} C_{k,4}^{0} + \sum_{k=1}^{m_3} C_{k,3}^{1} C_{k,3}^{1}$$
$$+ \sum_{k=1}^{m_2} C_{k,2}^{2} C_{k,2}^{2} + \sum_{k=1}^{m_1} C_{k,1}^{3} C_{k,1}^{3}$$
$$= \sum_{k=1}^{m_4} C_{k,4}^{0} C_{k,4}^{0} + \sum_{k=1}^{m_3} C_{k,3}^{1} C_{k,3}^{1}$$

3.4 Parameterization

There are several different parameterization methods. The simplest one is named accumulated chord length parameterization. Applying least square B-spline, it needs to revise in small extent. According to equation (7), two sets of parameters are to be calculated, one on the boundary of segments, i.e., knot point in parameter domain; the other in the segments, i.e., u_{gi}. Then they are to be synthesized in one series. The following is procedure to calculate the u_i and $u_{j,i}$.

1) Inside the segments, calculate the distance from one points to the next point directly, and then store the result in a suitable element according to the number of points in the respective segment.
2) Interpolate linearly the u_i values on the knot points with $u_{j,i}$s. Fig. 4 shows the interpolation of u_i.
3) Normalize the u_i and $u_{j,i}$. The equation for this procedure is listed in equation (10).

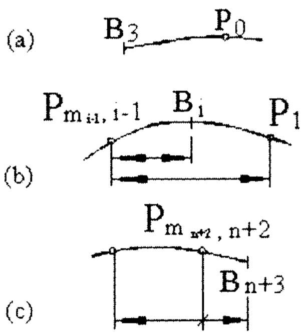

Fig. 4 Interpolation of u_i

4. Simulation Test

In order to certify the procedure listed in the previous paragraph, a test is conducted with data according to the following formula:

$$z_{i,j} = Ax_i^2 + By_j^2 + Cx_i^4 + Dx_i^2 y_j^2 + Ey_j^4 + Fx_i^4 y_j^2 + Gx_i^2 y_j^4 + Hx_i^4 y_j^2 + I + \delta_{i,j} \qquad (11)$$

where $x \in [-220, 220], y \in [-170, 170]$ and $\delta_{i,j}$ is artificial random errors which are subjected to normal distribution.

Coefficients are represented from A to I. These data construct a paraboloid. These data are fitted as a free surface with LS B-spline and ordinary B-spline respectively. The data are interpolated with the fitted surfaces, and then the interpolated data are fitted with ordinary least square paraboloid. Finally, the interpolated errors of data are calculated from the ordinary stimulation without random errors. The calculation results are listed in Table 1. Part of the original data, control vertices and interpolated points are listed in the appendix A. In Table 1, the values with percent mark in parentheses are the relative errors of the coefficients fitted to the corresponding coefficients simulated. The errors from the coefficients G and H look considerably large. But the analysis of errors statistically shows that the contribution of coefficients F, G and H to the surface fitting is smaller than the others'. It can be proved by the residual errors listed in Table 1.

When the number of data points is increased, the accuracy of least square B-spline is fairly improved while that of ordinary B-spline fitting keeps almost the same. This situation reflects the merit of least square fitting. Increasing the number of data makes the number of control vertices of ordinary B-spline increase also. However, equation (11) does not guarantee to give always good fitting for all types of curves.

5. Concluding Remarks

From Table 1 we can state the following statements:

$$
\begin{cases}
C_{g_i}^{0} = \dfrac{(u_{i+1}-u_{g_i})^3}{(\Delta_i+\Delta_{i-1}+\Delta_{i-2})(\Delta_i+\Delta_{i-1})} \\
C_{g_i}^{1} = \dfrac{(u_{i+1}-u_{g_i})^2(u_{g_i}-u_{i-2})}{(\Delta_i+\Delta_{i-1}+\Delta_{i-2})(\Delta_i+\Delta_{i-1})} + \dfrac{(u_{i+2}-u_{g_i})(u_{i+1}-u_{g_i})(u_{g_i}-u_{i-1})}{(\Delta_{i+1}+\Delta_i+\Delta_{i-1})(\Delta_i+\Delta_{i-1})} + \dfrac{(u_{i+2}-u_{g_i})^2(u_{g_i}-u_i)}{(\Delta_{i+1}+\Delta_i+\Delta_{i-1})(\Delta_{i+1}+\Delta_i)} \\
C_{g_i}^{2} = \dfrac{(u_{g_i}-u_{i-1})^2(u_{i+1}-u_{g_i})}{(\Delta_{i+1}+\Delta_i+\Delta_{i-1})(\Delta_i+\Delta_{i-1})} + \dfrac{(u_{i+2}-u_{g_i})(u_{g_i}-u_i)(u_{g_i}-u_{i-1})}{(\Delta_{i+1}+\Delta_i+\Delta_{i-1})(\Delta_{i+1}+\Delta_i)} + \dfrac{(u_{i+3}-u_{g_i})(u_{g_i}-u_i)^2}{(\Delta_{i+2}+\Delta_{i+1}+\Delta_i)(\Delta_{i+1}+\Delta_i)} \\
C_{g_i}^{3} = \dfrac{(u_{g_i}-u_i)^3}{(\Delta_{i+2}+\Delta_{i+1}+\Delta_i)(\Delta_{i+1}+\Delta_i)}
\end{cases}
\tag{7}
$$

$$
\begin{cases}
A_{q,r} = \sum_{s=3}^{0}\left[\Delta_{s+q-1}\sum_{k=1}^{m_{q+s-1}} C_{k,q+s-1}^{3-s}P_t\right] & r > (n+3), t = \sum_{t1=3}^{q+s-2} m_{t1} + k - 1 \\
A_{q,r} = \sum_{s=3}^{r-q}\sum_{k=1}^{m_{q+s-1}} C_{k,q+s-1}^{3-s}C_{k,q+s-1}^{3-s+r-q} & r \le (n+3), 0 \le r-q \le 3 \\
A_{q,r} = 0 & r \le (n+3), r-q \ge 4 \\
A_{q,r} = A_{r,q} & r \le (n+3), r \le q
\end{cases}
\tag{8}
$$

$$
\begin{pmatrix}
C_{1,3}^{0} & C_{1,3}^{1} & C_{1,3}^{2} & C_{1,3}^{3} & & & & & & & \\
C_{2,3}^{0} & C_{2,3}^{1} & C_{2,3}^{2} & C_{2,3}^{3} & & & & & & & \\
\vdots & \vdots & \vdots & \vdots & & & & & & & \\
C_{m_3}^{0} & C_{m_3}^{1} & C_{m_3}^{2} & C_{m_3}^{3} & & & & & & & \\
 & C_{1,4}^{0} & C_{1,4}^{1} & C_{1,4}^{2} & C_{1,4}^{3} & & & & & & \\
 & \vdots & \vdots & \vdots & \vdots & & & & & & \\
 & C_{m_4}^{0} & C_{m_4}^{1} & C_{m_4}^{2} & C_{m_4}^{3} & & & & & & \\
 & & C_{1,5}^{0} & C_{1,5}^{1} & C_{1,5}^{2} & C_{1,5}^{3} & & & & & \\
 & & \vdots & \vdots & \vdots & \vdots & & & & & \\
 & & & & & & & C_{1,n+2}^{0} & C_{1,n+2}^{1} & C_{1,n+2}^{2} & C_{1,n+2}^{3} \\
 & & & & & & & \vdots & \vdots & \vdots & \vdots \\
 & & & & & & & C_{m_{n+2}}^{0} & C_{m_{n+2}}^{1} & C_{m_{n+2}}^{2} & C_{m_{n+2}}^{3}
\end{pmatrix}
\begin{pmatrix} d_0 \\ d_1 \\ \vdots \\ \vdots \\ d_{n+2} \end{pmatrix}
=
\begin{pmatrix} \Delta_3 P_0 \\ \Delta_3 P_1 \\ \vdots \\ \Delta_3 P_{m_3} \\ \vdots \\ \Delta_{n+2} P_m \end{pmatrix}
\tag{9}
$$

$$
\begin{cases}
u_{1,3} = P_0 - B_0 \\
u_{j,i} = u_{j-1,i} + |P_t - P_{t-1}| & j \le m_i,\ t = \sum_{t1=3}^{i-1} m_{t1} + j - 1 \\
u_{j,i} = u_{m_{i-1},i-1} + |P_t - P_{t-1}| & j = 1,\ i > 3,\ t = \sum_{t1=3}^{i-1} m_{t1} + j - 1
\end{cases}
\tag{10}
$$

(1) Least Square B-spline and ordinary B-spline fitting (reverse calculation in CAGD) techniques are equally suitable for surface fitting.

(2) When we want to have response from all the measured data and to get a few parameters as possible to represent free curves and surfaces, the least square B-spline is a good choice.

(3) The number of real points must be larger than that of control vertices to be estimated. Otherwise wrong results would be obtained.

(4) The procedure mentioned in section 3 is used to curve fitting. In surface fitting, this procedure is repetitively applied to get control curves, and then a surface are constructed from these control curves with ordinary B-spline fitting technique according to the procedure mentioned in section 2.3.

(5) Since the number of real points is larger than the number of control vertices to estimate, boundary supplemental conditions such as ends tangent condition are unnecessary.

Acknowledgement

This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Machine Tool Research Center at Changwon National University.

References

(1) Chae, H. C., Myung, T. S. and Kim, T. K., 2000, "Development of Free-Form Surface Modeling System Using the Reverse Engineering Technology", *Journal of Korean Society of Machine Tool Engineers,* Vol. 9, No. 3, pp. 111~122.

(2) Goo, Y. H. and Lee, D. J., 1998, "A Study on the Generation of B-Spline Surface by 3D Measurement Data", *Proceedings of '98 Conference of Korean Society of Machine Tool Engineers,* pp. 76~81.

(3) Shi, F.Z., 1994, *Computer Aided Geometric Design and Non-uniform Rational B-Spline (CAGD & NURBS)*, Beijing University of Aeronautics and Astronautics Press, Chinese, pp. 266.

(4) Piegl, L. and Tiller, W., 1995, *The NURBS Book*, Springer, pp. 81~116.

(5) Lee, K. W., 1999, *Principles of CAD/CAM/CAE Systems*, Addison Wesley, pp. 507~511.

(6) Choi, B. K., 1991, *Surface Modeling for CAD/ CAM*, Elsevier, pp. 101~103.

(7) Zeid, I., 1991, *CAD/CAM Theory and Practice*, McGraw-Hill, Inc., pp. 226~228.

Appendix A

1) 285 points raw data (in part):			1) Control vertices (in part):			2) Interpolated Point (in part):		
Along X axis			Along X axis			Along X axis		
219.9879	0.0141	40.2984	-222.5000	0.0067	39.7587	-222.5000	0.0065	39.6765
200.5896	0.0163	44.6526	-207.7333	0.0059	43.4532	-178.2000	0.0070	48.8567
176.0324	0.0156	49.2223	-178.2000	0.0088	49.4785	-133.90	0.0072	54.8629
151.2628	0.0153	52.8360	-133.9000	0.0064	55.3476	-89.6000	0.0086	58.5514
126.4461	0.0149	55.6300	-89.6000	0.0096	58.8680	-45.3000	0.0094	60.5523
101.4921	0.0147	57.7683	-45.3000	0.0102	60.8680	-1.0000	0.0099	61.1932
76.5702	0.0140	59.3075	-1.0000	0.0097	61.4603	43.3000	0.0110	60.6014
51.5829	0.0133	60.3496	43.3000	0.0124	60.8720	87.6000	0.0118	58.6926
26.6282	0.0133	60.9740	87.6000	0.0115	59.0603	131.9000	0.0127	55.0787
1.6994	0.0127	61.1890	131.9000	0.0138	55.5130	176.2000	0.0132	49.1933
-23.3464	0.0112	61.0295	176.2000	0.0128	49.8575	220.5000	0.0116	40.1732
Along Y axis:			Along Y axis:			Along Y axis:		
1.2889	170.0117	50.2707	-1.0000	-169.9856	50.5775	-1.0000	-169.9876	50.2925
1.4625	150.0119	52.8459	-1.0000	-163.3301	51.4148	-1.0000	-153.0743	52.4567
1.4674	125.0125	55.4950	-1.0000	-148.3334	53.4064	-1.0000	-136.1334	54.3825
1.5931	100.0118	57.6060	-1.0000	-125.0059	55.7717	-1.0000	-119.1742	56.0180
1.6186	75.0121	59.1902	-1.0000	-100.0030	57.9134	-1.0000	-102.1898	57.4281
1.6609	50.0123	60.3015	-1.0000	-74.9994	59.4771	-1.0000	-85.1856	58.6031
1.6232	25.0124	60.9759	-1.0000	-49.9966	60.5654	-1.0000	-68.1654	59.5420
1.6994	0.0127	61.1890	-1.0000	-24.9927	61.2787	-1.0000	-51.1325	60.2616
1.6283	-24.9876	60.9807	-1.0000	0.0097	61.4603	-1.0000	-17.0425	61.0985
1.6745	-49.9875	60.3126	-1.0000	25.0192	61.2438	-1.0000	0.0099	61.1932
1.6269	-74.9876	59.1909	-1.0000	50.0212	60.5853	-1.0000	17.0641	61.0870
1.5945	-99.9876	57.5896	-1.0000	75.0245	59.4751	-1.0000	34.1140	60.7791
1.5537	-124.9872	55.4780	-1.0000	100.0285	57.9451	-1.0000	51.1560	60.2652
1.4481	-149.9872	52.8315	-1.0000	125.0263	55.7823	-1.0000	68.1887	59.5423
1.2445	-169.9858	50.2905	-1.0000	148.3579	53.4527	-1.0000	85.2092	58.6064

Segment boundary along X axis: (-222.5, -178.2, -133.9, -89.6, -45.3, -1, 43.3, 87.6, 131.9, 176.2, 220.5)

Notes: 1) Many data are concerned in surfaces fitting. Ones listed above are only small part of the whole data from 285 points set. The second set of data (555 points) is just derived from the first set by interpolating one point between every two points along the X axis, say, after (219.9879 0.0141 40.2984), point (210.2888 0.0152 42.5739) is added.

2) According to section 3.1 Convention and the data listed above, it is obvious that the number of original data along X axis is 19 (m=0, , 18); the X axis is divided into n=10 segments. The number of knot points concerned is n+1=11 (i=3, 4, .., n+3). There are 13 control vertices (j=0, 1, , n+2) to be estimated. The numbers of real points in each segment are $m_3=m_4== m_{12}=2$, but $m_8=1$. Being checked the real points and segment boundary, no real point is just on the boundary.