Eng.& Tech. Journal,Vol.28, No.8, 2010

A New Method For Three Dimensional Cubic Bezier Surface Reconstruction Based On Matching The Surface Framework

Dr. Wissam K. Hamdan*

Received on: 30/9/2009
Accepted on: 7/1/2010

Abstract

Numerous efforts have been directed to convert the physical model (in-hand model) to a computer model. This work is dedicated for the cubic Bezier surface reconstruction based on finding the positions of the sixteen control points that form the surface of the product. The idea is based on inverse progressive search (IPS) method rather than the approximate surface fitting method already used in previous researches. The presented method is based on three successive steps:(a) converting the continuous coordinate measuring machine (CMM) data to discrete data,(b) estimating the positions of the 12 boundary control points and (c) estimating the positions of the 4 intermediate control points to generate the intended surface. To show the feasibility of the suggested method two experimental examples are conducted. The results show the validity and effectiveness of the method from the accuracy and computation time point of view.

Keywords: Surface Reconstruction, Reverse Engineering, CMM, 3D Parametric Surfaces, Surface Fitting.

طريقة جديدة لإعادة تركيب سطح بيزير ثلاثي الإبعاد من الدرجة الثالثة بالاعتماد على طريقة مطابقة مضلع السطح

الخلاصة

هنالك الكثير من الجهود التي بذلت من قبل الباحثين والرامية إلى تحويل النماذج او المنتجات النهائية الى نماذج مرنة (نماذج حاسوبية). يهدف البحث الحالي الى اعادة انشاء سطح بييزر من الدرجة الثالثة بالاعتماد على ايجاد مواقع نقاط السيطرة الستة عشر المكونة لسطح النموذج النهائي. الفكرة تعتمد على طريقة البحث العكسي التقادمي(IPS) بدلا من طريقة إلباس الأسطح(Surface Fitting) التقريبية المستخدمة في البحوث السابقة. الطريقة المقترحة تعتمد على ثلاثة مراحل متوالية: (أ) تحويل البيانات المستمرة لماكنة قياس الاحداثيات الى بيانات متقطعة, (ب) تخمين مواقع نقاط السيطرة الاثنى عشر المحيطية للسطح, (ج) تخمين مواقع نقاط السيطرة الاربعة الداخلية لتوليد السطح المطلوب. لغرض معرفة امكانية تطبيق الطريقة المقترحة تم اعادة توليد سطحين لنموذجين مختلفين. اثبتت النتائج مشروعية الطريقة المقترحة وكفائتها العالية من حيث الدقة والسرعة

*Production Engineering & Metallurgy Department, University of Technology/Baghdad

https://doi.org/10.30684/etj.28.8.11
2412-0758/University of Technology-Iraq, Baghdad, Iraq

1. Introduction

The objective of reverse engineering (RE) is to convert the physical prototype (PP) to a virtual prototype (VP). The physical prototype may be in the form of clay or wooden prototype produced by artisan. The mass production of such prototypes can be done by copy milling machine. Accordingly, this method of production become fruitless from the economical point of view. If the PP is converted to VP using RE, the copy milling machine can be dispensed with a CNC milling machine. Consequently, the production can be conducted by integration between CAD/CAM to speed up the machining cycle. Meanwhile, the PP may be a final product converted from its original VP conducted by a somewhat manufacturing process such as casting, machining, forming, …etc.

A very important problem arise when the manufacturing engineer want to compare between the produced physical prototype and its virtual prototype (CAD model). The exception for this when the physical prototypes have a primitive shapes (cube, cylinder, hemisphere , ..etc.). On the other hand, when the physical prototypes have complex free from shapes, the comparison between physical prototypes and virtual prototypes become a tricky matter from both topological and geometrical point of view. Undoubtedly, the comparison between physical prototypes and virtual prototypes of free shapes could not be achieved directly. The reason is that the VP described mathematically as a function of two independent variables [1]:

$$S(u,v) = \left[S_x(u,v) \quad S_y(u,v) \quad S_z(u,v) \right] \quad (u,v) \in [0,1] \qquad \ldots\ldots(1)$$

in the u and v domain, while the physical prototypes have no mathematical description just the chaotic points of specific coordinate in the Cartesian domain. As a consequence, for any point $P(x, y, z)$ relates to the physical prototype one can't know the specific value of u and v parameter which leads to its corresponding point in the virtual prototype. In other word, the digitization of virtual prototype is based on u and v values, while the digitization of physical prototype is based on the x and y range which in turn depend on the size and shape of the object itself and the increment of measurements. These reasons reflect the importance of RE in converting the measured points of physical prototype to its original virtual prototype. The surface fitting and interpolation techniques are widely used methods to convert the 3D measured points to parametric surfaces[2]. Many considerable efforts discussed the problem of converting the know 3D data to its virtual prototype [3-6].The synopsis of these works is that they depend on the iteration method for surfaces fitting and interpolation. An approach to represent the Euclidean distance between the virtual prototype and physical prototype was presented by Pratt [3], the parametric variables u and v are solved iteratively using two nonlinear equations. Rogers and Fig [4] utilizes the first order Taylor series to optimize the parametric variables. Sarkar and Menq [5] proposed an explicit expression as an optimization method. Bing Li et.al [6] proposed a non contact method to acquire the 3D large profile information based on laser scanning measurements technology. This work is dedicated to present a reverse surface approximation algorithm (RSA) for the reconstruction of complex 3D Bezier surfaces from 3D measured sparse points. The Bezier surface technique is nominated in this work since it is become a famous tool in computer aided geometric design CAGD [7,8] . The proposed reverse method is based on estimation the position of the sixteen control points of the Bezier polygon from the given measured 3D data points based on an inverse iterative procedure. Figure(1) shows the relation between virtual prototype and physical prototype, the Figure also depicts the proposed RE procedure (the material of the present work).

2. Problem Statements and Solution Approach

Mainly, there are three problems concerning with conversion of known 3D data- produced by CMM- to surface model:

1. Converting the 3D CMM data to successive sets of points.
2. Finding the control points of the boundary curves.
3. Finding the intermediate control points.

Table 1 can be used as an aid to understand the solution of the first problem. This table represent an assumed CMM data of the surface presented in Figure2.

The vector modulus L appears in previous table 1 is used as a criterion for the distinguishing between each two successive sets of points which can be calculated as follows;

$$L=\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2+(z_{i+1}-z_i)^2},\ i=1,2,3,\ldots,n \qquad \ldots\ldots.(2)$$

where n is the number of data points.

From the previous table 1, it is observed that the vector modulus seams to be uniform for some points and then increased sharply at another points. This increases in the value of vector modulus takes place when the CMM probe's transfer from curve to another or when change its motion from forward feed direction to cross feed direction.

As a consequence , in the present work, the length of the first vector was used as a threshold (T_h) value to sort the successive sets of points. Any successive points will represent the same set of points if the following condition has been satisfied:

$$\frac{L_{i+1}}{L_i} >> T_h \qquad \forall\, i \in n \ \ldots\ldots.(3)$$

Figure 3 shows a plot of the vector modulus ratios resembled in table 1.

The sharp change in the vector modulus ratio represent the change from curve to another one which are indicated by red cells in table 1. Figure 4,a shows the plot of the proposed 3D CMM data of table 1, while Figure 4,b shows the converted or processed 3D CMM data according to Eq.2. The 2nd and 3rd problem are solved later on in the next section.

3.The reverse surface interpolation algorithm

The idea of reverse surface interpolation algorithm is based on finding the positions of the16 control points of the Bezier polygon and its four boundary curves. According to the properties of the Bezier surface [9,10], only the four corner points joined with the control polygons as figure5 depicts. Accordingly, for each boundary set of points, there are two know control points (flagged by fill red circles in figure5) and two unknown control points (fill black squares in the same figure5). The procedure adopted to fined the unknown control points is detailed in the following subsection.

3.1 Boundary Curve Approximation

Figure 6 shows the graphical steps of constructing Bezier curve.

Given P_1 & P_4

Wanted P_2 & P_3

Using the parametric linear interpolation, the following sets of equations can be obtained:

$\mathbf{P}_5 = \mathbf{P}_1 + u(\mathbf{P}_2 + \mathbf{P}_1)$

$\mathbf{P}_6 = \mathbf{P}_2 + u(\mathbf{P}_3 + \mathbf{P}_2)$

$\mathbf{P}_7 = \mathbf{P}_3 + u(\mathbf{P}_4 + \mathbf{P}_3)$

$\mathbf{P}_8 = \mathbf{P}_5 + u(\mathbf{P}_6 + \mathbf{P}_5)$

$\mathbf{P}_9 = \mathbf{P}_6 + u(\mathbf{P}_7 + \mathbf{P}_6)$

$\mathbf{P} = \mathbf{P}_8 + u(\mathbf{P}_9 + \mathbf{P}_8)$ ……(4)

Where **P** is any point on the Bezier curves. After some substitutions between the terms of Eq.4, the following equation can be obtained:

$$P(u) = P_1U_1 + P_2U_2 + P_3U_3 + P_4U_4 \quad \text{……(5)}$$

Where

$U_1 = 1 - 3u + 3u^2 - u^3$,

$U_2 = 3u - 6u^2 + 3u^3$

$U_3 = 3u^2 - 3u^3$, $U_4 = u^3$;

$u \in [0,1]$ ……(6)

when $u=0$

$U_1 = 1$, $U_2 = U_3 = U_4 = 0$

so, $P(o) = P_1$

when $u = 1$

$U_1 = U_2 = U_3 = 0$; $U_4 = 1$

So, $P(1) = P_4$

Therefore, the point P moves from P_1 to P_4 along the curves as the u increased from $0 \rightarrow 1$. Equation 5 represents the target of the present proposed reverse algorithm.

3.2 The Proposed Reverse Algorithm

Given: a set of boundary points (recognized CMM data)

Wanted : four control points of the boundary Bezier polygon.

Procedure:

1. set $P_1 = P^1$, $P_4 = P^n$

2. Draw tangent line from P^1 and passing through P_2 (see fig. 7)
3. Draw tangent line from P^n passing through P^{n-1}.
4. Extended the two tangents to met at P^c.
5. Fined the parametric equations of the two tangents $L_1(t)$ and $L_2(t)$

$$V_1(t) = P^1 + t(P^c - P^1)$$
$$V_2(t) = P^n + t(P^c - P^n) \quad \text{....(7)}$$

Undoubtedly, the 2nd control point P_2 will be along $V_1(t)$ and 3rd control point along $V_1(t)$. An iterative scheme with the aid of Eq.5 will used here to fined P_2 and P_3 as follows :

6. set t = 0 , u = 0
7. t = 0
8. Let $P_2(t) = P_1 + t(P^c - P_1)$ ...(8)
9. $P_3(t) = P^c + s(P_4 - P^c)$.(9)

$$P(u) = P_1U_1 + P_2U_2 + P_3U_3 + P_4U_4 \quad \forall P_2, P_3 \quad \text{.....(10)}$$

11. $u = u + \Delta u$

If $u < 1$ Go to 10

Otherwise

$$e_{av} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(e_i)^2} \quad \text{......(11)}$$

$$e_i = \left|P - P^i\right| \quad \text{.....(12)}$$

where e_{av} is the root-mean squares error (RMS), P^i represent the measured points and P represents the corresponding point on the interpolated curves.

12. If $e_{av} > e_{\max}$ Go to 9

Otherwise

$P_2(t) = P_4$; $P_3(t) = P_3$

13. If $t<1$

$t = t + \Delta t$ Go to 8

Otherwise

14. End

3.3 An Example

In this section, and to avoid future ambiguity, the following example has been introduced, to demonstrate in detail the proposed algorithm.

Table 2 shows the x, y, z coordinate of the assumed CMM data for a certain set of points. The plot of these data is depicted in Fig.8

According to data presented in table 2 and referring to Fig.8, the following linear equations can be evaluated:

$$L_1 : Z = Z_1 + \frac{z_2 - z_1}{x_2 - x_1}(x - x_1) \quad \text{..(13)}$$

$$L_2 : Z = Z_4 + \frac{z_{n-1} - z_4}{x_2 - x_1}(x - x_4) \quad \text{.....(14)}$$

To fined the intersection point, the two lines should be equated:

$L_1 = L_2$ at Pc

$$\therefore \; m_1(x - x_1) + z_1 = m_2(x - x_4) + z_4$$
$$m_1x - m_2x = z_4 - z_1 + m_1x_1 - m_2x_2$$
$$x(m_1 - m_2) = z_4 - z_1 + m_1x_1 - m_2x_2$$

$$x^c = x = \frac{z_4 - z_1 + m_1 x_1 - m_2 x_4}{m_1 - m_2} \quad ..(15)$$

Substituting Eq.15 in Eq.13 gives:

$$Z^c = m_1(x^c - x_1) + z_1 \quad .(16)$$

Accordingly , the parametric equation of V_1 & V_2 is (See Fig.9)

$$V_1(t) = P^1 + t(P^c - P^1)$$

$$V_2(t) = P^4 + t(P^c - P^4)$$

The objective of primary and secondary search is to find the second and the third control points of each boundary curve. Figure 10 shows a schematic representation of the iteration procedure which is coded in MATLAB programming language[11,12] as follows:

```
px2 =[5.0000    7.1500    9.3000 ];
px3 =[13.6000   15.2000   ];
pz2 =[ 6.0000  7.7500  9.5000];
pz3 =[13.0000   11.7500  ];
k=1;
for s=1:3  % the number of elements in px2 vector
for t=1:2  %the number of elements in px3 vector
      px1=[5 px2(s) px3(t) 20];
      pz1=[6 pz2(s) pz3(t) 8];
M=[1 0 0 0;-3 3 0 0;3 -6 3 0;-1 3 -3 1]; % the Bezier curve Matrix
I=1;
for u=0:0.1:1
      U=[1 u u^2 u^3];      % the vector of the independent parameter u of Bezier curve
         x(I)=U*M*px1';    % the Cartesian valued function of Bezier curve
         z(I)=U*M*pz1';    % the Cartesian valued function of Bezier curve
         I=I+1;
end
subplot(3,2,k);plot(x,z,'r')
hold on
subplot(3,2,k);plot(px1,pz1)
k=k+1;
end
end
```

The aforementioned progressive search procedure should be employed for each boundary curve. Up to now, the four control points of each boundary curves has been specified as shown in Fig.11. In other word, 12 control point out of 16 control points have been specified.

The remaining four control points are called intermediate points as show in fig.12.The procedure adopted to fined the two control points of each boundary curves is also adopted here to fined the intermediate control points. The exception for this that the two tangent vectors are passed between P_{11} & P_{44} to fined P_{22} & P_{33} , and between P_{14} & P_{41} to fined P_{23} & P_{32} control points (see Fig.12).

4. Experimental Work

Two experiment examples are demonstrated to show the validity of the proposed inverse algorithm. The first example deals with a perfume canister with a concave surface. Fig.13 shows an image of this canister . The dial gauge of a movable table in two dimensions. This arrangement versatile the measurements along x & y direction for any increment in the two directions. The isoparametric digitization approach of 1 mm increment was used to digitize this complex object in both x-and y-direction. Fig. 14 shows the x,y and z coordinates of the digitized canister. The sixteen control points of the Bezier surface which are generated by means of the proposed inverse search method are shown in Fig. 15, Fig.16 shows the fitted final surface. Figures 17 a,b and c show the average error each between the measured data points and the predicted surface along with x, y and z axis direction respectively. The second example is a light cover, this shape was selected to show the capability of the present method in extracting the concave shapes. Figures 18 to 22 show the results of this example.

5.1 Conclusions

A method for free-form surfaces reconstruction has been presented. It is based on an inverses progressive search for best control points that give the intended surface or near net surface. The procedure for the presented surface reconstruction method comprises three modules: the transformation module, the iteration module and the matching module. In the first module, the continuous space CMM data points are transformed into discrete sets of points. The second module is responsible for estimating the position of the surface control points using a multi steps iteration. This module may be regarded as the target of the proposed method. The latest module judge whether the estimated control points produce an error exceeds the allowable deviation limit. The results obtained from the error surface distribution shows that there is no error near the corners of the both shapes being tested since that the corner control points of Bezier surface lie exactly on it at these positions. Meanwhile, the produced error between the developed method and the exact shapes is with god agreement for most CAD and CAGD applications.

5.2. Contribution

The proposed method can yields a free-form surfaces without the needs for well known surface fitting or surface lofting methods. Therefore, the proposed method provides an additional tool for surface reconstruction.

5.3. Future Work

There are other types of free form surface which involves combined curvature profiles. A new modification to the current algorithm should be developed for such problems which make the iteration problem more difficult to solve.

5.4. Epilogue

Before now, there is no any previous work deals with the reconstruction of Bezier surfaces throughout looking for the control points. Accordingly, the proposed method open a new area in the RE context. The presented method for parametric surface reconstruction seems to be feasible, the feasibility is demonstrated by means of computer simulation and experimental runs.

References

[1] Jinming Wu, Renhong Wang "Approximate implicitization of parametric surfaces by using compactly supported radial basis functions". Computers & Mathematics with Applications, 56, 12, (2008) 3064-3069

[2] Simon Flöry, Michael Hofer "Surface fitting and registration of point clouds using approximations of the unsigned distance function" Computer Aided Geometric Design, in Press, (2009).

[3] M.J.Partt "Smooth Parametric Surface Approximation to Discrete Data" Computer Aided Design 2 (1985). Pp. 165-171.

[4] D.F. Rogers and N.G.Fig "Constrained B-Spline Curve and Surface Fitting" Computer Aided Design 21, 10 (1989). Pp. 641-648.

[5] B.Sarkar and C.H.Menq "Parameter Optimization in Approximation Curves and Surfaces to Measurements Data " Computer Aided Design 8 (1991). Pp. 267-290.

[6] Yingfu Guo; Zhuangde Jiang; Xiaoqiang Wang; Bing Li "Technology of freeform surface reconstruction from laser scanning point cloud basedon nonuniform B-spline" Third International Symposium on Precision Mechanical Measurements (2006).

[7] Ding Qiulin and Davies B.J. " Surface engineering geometry for Computer aided design and Manufacturing" New York : John Wiley &Sons , 1987 .

[8] R.M.C. Bodduluri, B. Ravani, "Geometric design and fabrication of developable Bezier and B-spline surfaces" Trans. ASME Journal of Mechanical Design 116 (1994) 1024-1048.

[9] Charles T.Loop and Tony D.Derose "A Multisided Generalization of Bezier Surfaces " ACM Trans.on Graphics. 8, 3 (1989). Pp. 204-234.

[10] E. Michael, Mortensor " Geometric modelling " John Wiely and Sons, Inc.new York, 1997.

[11] Rudar Pratap " Getting started with MATLAB" New York, Oxford University Press, 2002.

[12] MATLAB help, Mathworks, Massachusetts, 2004.

Table (1) CMM data of the surface shown in Fig.2.

#	X	Y	Z	L	#	X	Y	Z	L
1	5.0000	5.0000	5.0000		25	16.9896	17.7021	11.1169	3.4710
2	7.2500	5.6503	7.0205	3.0932	26	20.6146	17.9060	11.2942	3.6351
3	9.5000	5.8820	8.4740	2.6887	27	24.6004	17.5030	10.7287	4.0458
4	11.7500	5.8167	9.1985	2.3647	28	28.9793	16.4049	9.3102	4.7321
5	14.0000	5.5760	9.0320	2.2690	29	8.6000	18.3200	7.1600	20.5817
6	16.2500	5.2813	7.8125	2.5762	30	11.3228	19.7638	8.9207	3.5494
7	18.5000	5.0540	5.3780	3.3228	31	14.4718	20.9432	10.2854	3.6290
8	6.6594	7.5706	6.1475	12.1295	32	18.0456	21.7740	11.1563	3.7710
9	9.1631	8.6528	8.0374	3.3184	33	22.0427	22.1723	11.4356	4.0266
10	11.6299	9.3622	9.4464	2.9281	34	26.4616	22.0539	11.0255	4.4395
11	14.1209	9.6941	10.2174	2.6286	35	31.3010	21.3347	9.8280	5.0369
12	16.6971	9.6437	10.1930	2.5769	36	9.9219	22.5781	6.6875	21.6443
13	19.4199	9.2063	9.2161	2.9256	37	12.5433	23.7778	8.5389	3.4262
14	22.3503	8.3771	7.1294	3.6917	38	15.6470	24.8941	10.0235	3.6170
15	7.4750	10.7150	6.8900	15.0598	39	19.2215	25.8519	11.0279	3.8345
16	10.1409	12.0841	8.6838	3.4927	40	23.2554	26.5764	11.4384	4.1189
17	12.8864	13.0904	10.0422	3.2243	41	27.7373	26.9924	11.1414	4.5110
18	15.7744	13.6657	10.8304	3.0484	42	32.6559	27.0251	10.0233	5.0442
19	18.8674	13.7417	10.9136	3.0951	43	12.4250	27.0050	5.8100	20.6650
20	22.2282	13.2503	10.1569	3.4798	44	14.8611	27.7516	7.8421	3.2591
21	25.9196	12.1232	8.4254	4.2301	45	17.6414	28.7455	9.5666	3.4193
22	7.9531	14.3319	7.2275	18.1413	46	20.7963	29.9064	10.8110	3.5847
23	10.6914	15.8268	8.9736	3.5751	47	24.3566	31.1541	11.4030	3.8187
24	13.6927	16.9797	10.3067	3.4805	48	28.3527	32.4082	11.1702	4.1948
					49	32.8155	33.5887	9.9402	4.7773

Table (2) Assumed CMM Data Points

#	X	Z	Y	#	X	Z	Y
1	5.00	6.00	Constant	26	12.50	10.00	.
2	5.30	6.24	=	27	12.80	10.06	.
3	5.60	6.47	.	28	13.10	10.10	.
4	5.90	6.70	.	29	13.40	10.14	.
5	6.20	6.92	.	30	13.70	10.16	.
6	6.50	7.14	.	31	14.00	10.18	.
7	6.80	7.35	.	32	14.30	10.18	.
8	7.10	7.55		33	14.60	10.17	.
9	7.40	7.75	.	34	14.90	10.16	.
10	7.70	7.94	.	35	15.20	10.13	.
11	8.00	8.13	.	36	15.50	10.09	.
12	8.30	8.31	.	37	15.80	10.04	.
13	8.60	8.48	.	38	16.10	9.97	.
14	8.90	8.64	.	39	16.40	9.90	.
15	9.20	8.80	.	40	16.70	9.81	.
16	9.50	8.95	.	41	17.00	9.71	.
17	9.80	9.09	.	42	17.30	9.60	.
18	10.10	9.23	.	43	17.60	9.48	.
19	10.40	9.36	.	44	17.90	9.34	.
20	10.70	9.47	.	45	18.20	9.19	.
21	11.00	9.58	.	46	18.50	9.02	.
22	11.30	9.69	.	47	18.80	8.85	.
23	11.60	9.78	.	48	19.10	8.66	.
24	11.90	9.86	.	49	19.40	8.45	.
25	12.20	9.94	.	50	19.70	8.23	.
					20.00	8.00	.

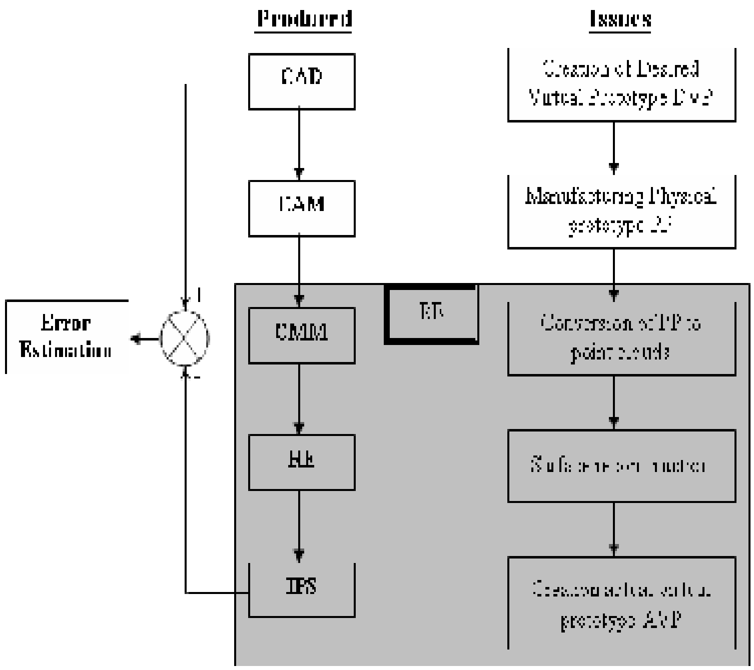

Figure (1) The proposed reverse engineering approach

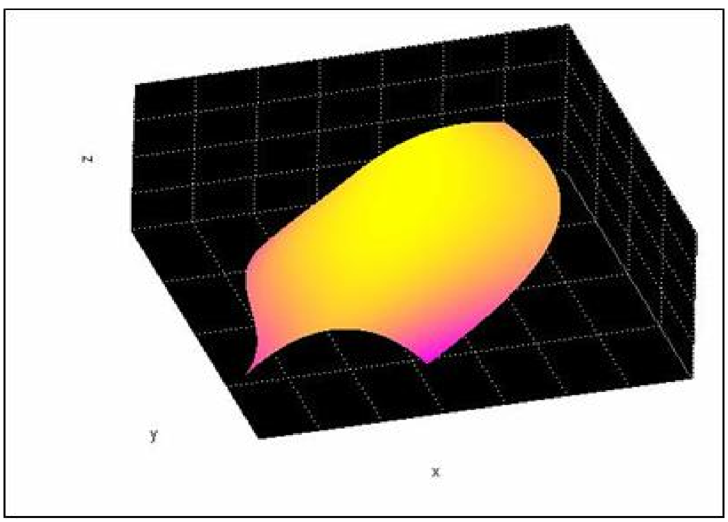

Figure (2) The assumed 3D surface.

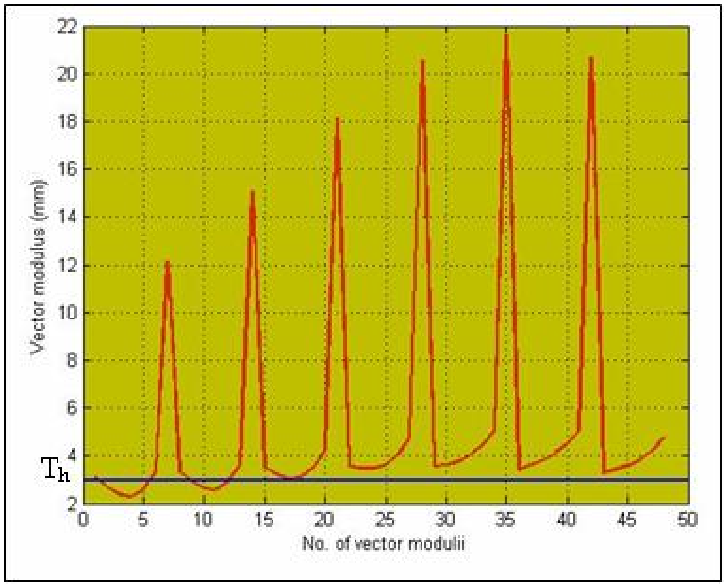

Figure (3) Distribution the vector modules over the whole surface shows the sudden change in its modulus at some points.

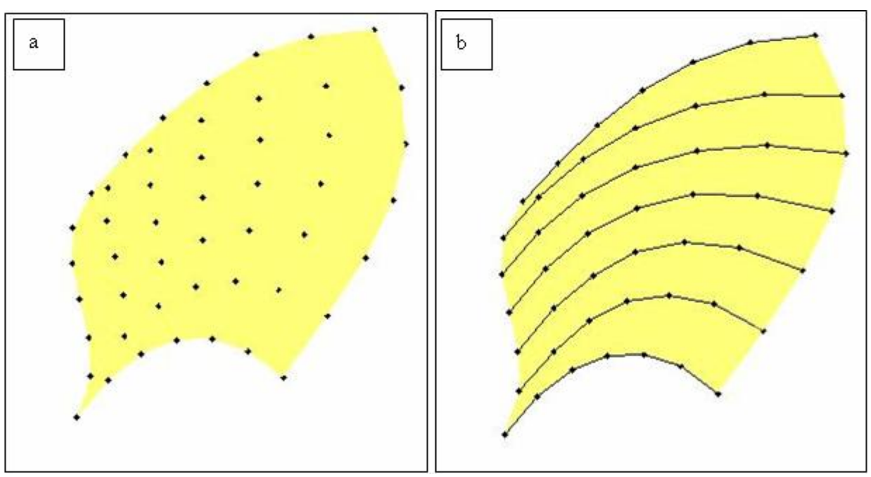

Figure (4) The difference between chaotic CMM data points and the processing data.

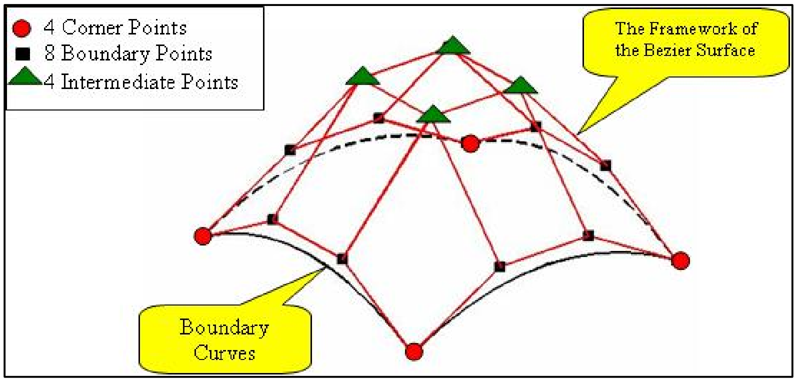

Figure (5) Properties and ingredients of a Bezier surface

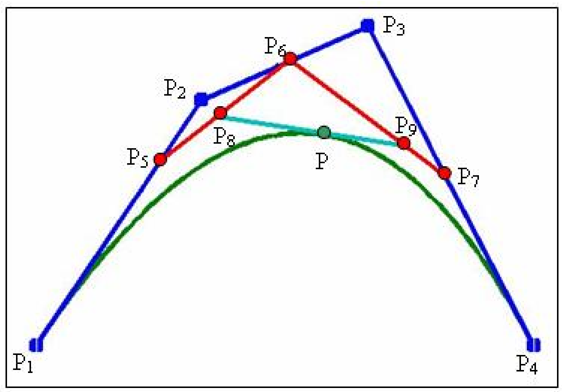

Figure (6) Schematic Representation of Constructing Bezier Curve

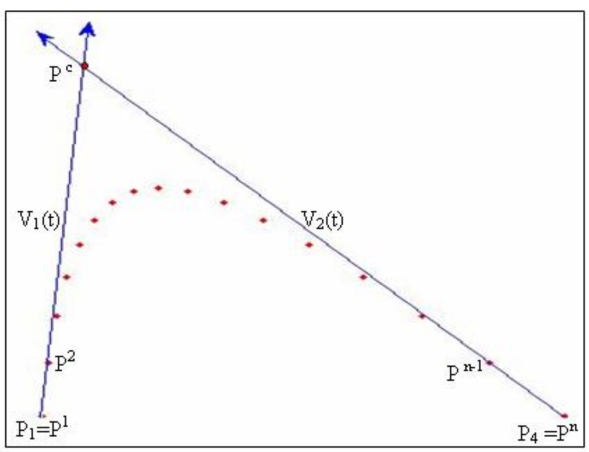

Figure (7) Method of Determining the Boundary Control Points

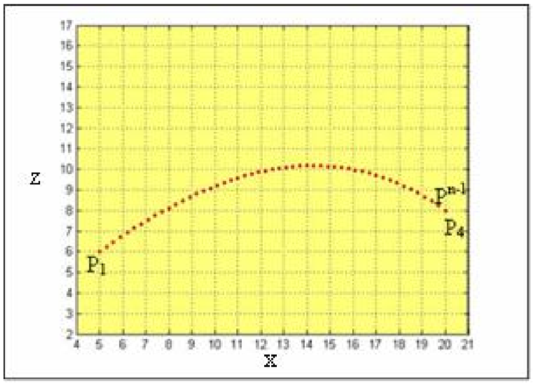

Figure (8) The Plot of Data Points Listed in Table 2.

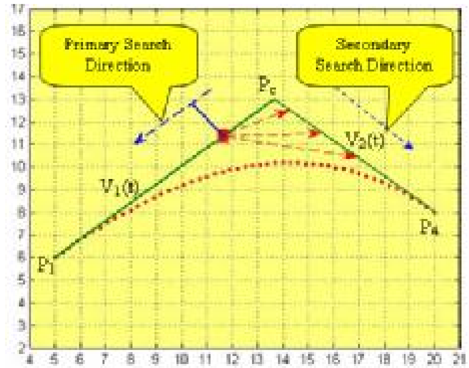

Figure (9) The Adopted Progressive Search Method.

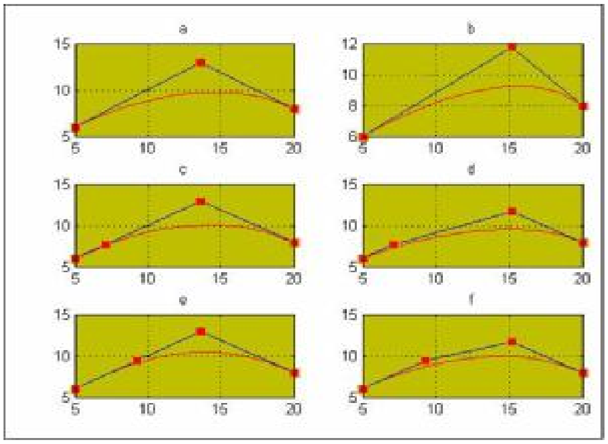

Figure (10) The Iteration Scheme to Find the Second and Third Control Points.

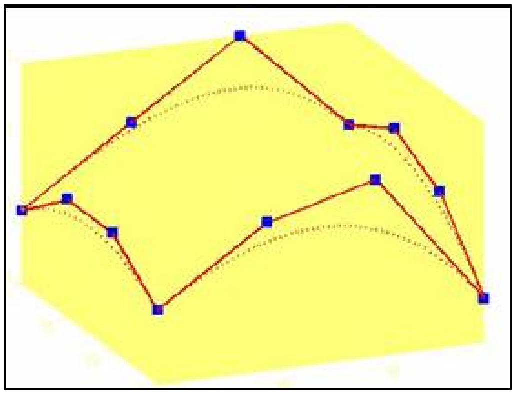

Figure (11) The Estimated 12 Boundary Control Points of the Bezier Surface Using Proposed Method.

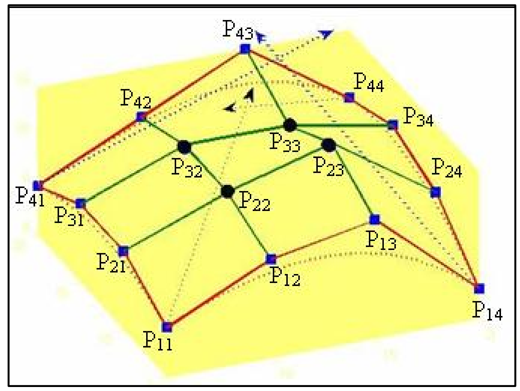

Figure (12) The Intermediate Control Points of the Bezier Surface (Black Circles).

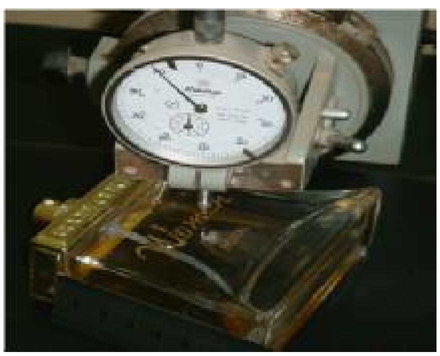

Figure (13) The Measurement Setup of the First Example.

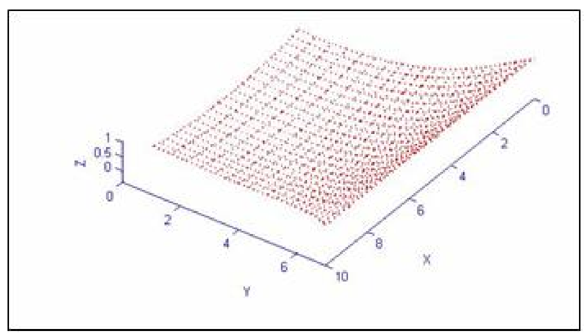

Figure (14) The Measured Data Points of the First Example.

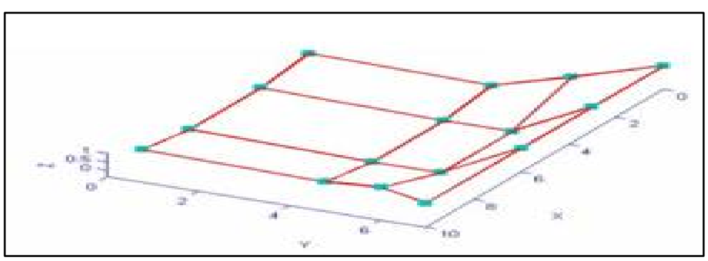

Figure (15) The Estimated Control Points of the intended Surface.

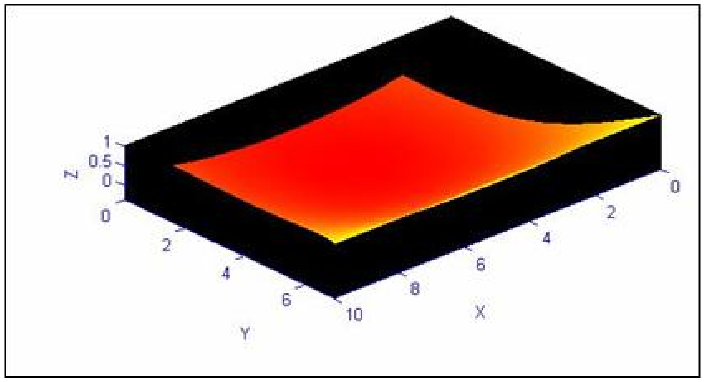

Figure (16) The Estimated Surface.

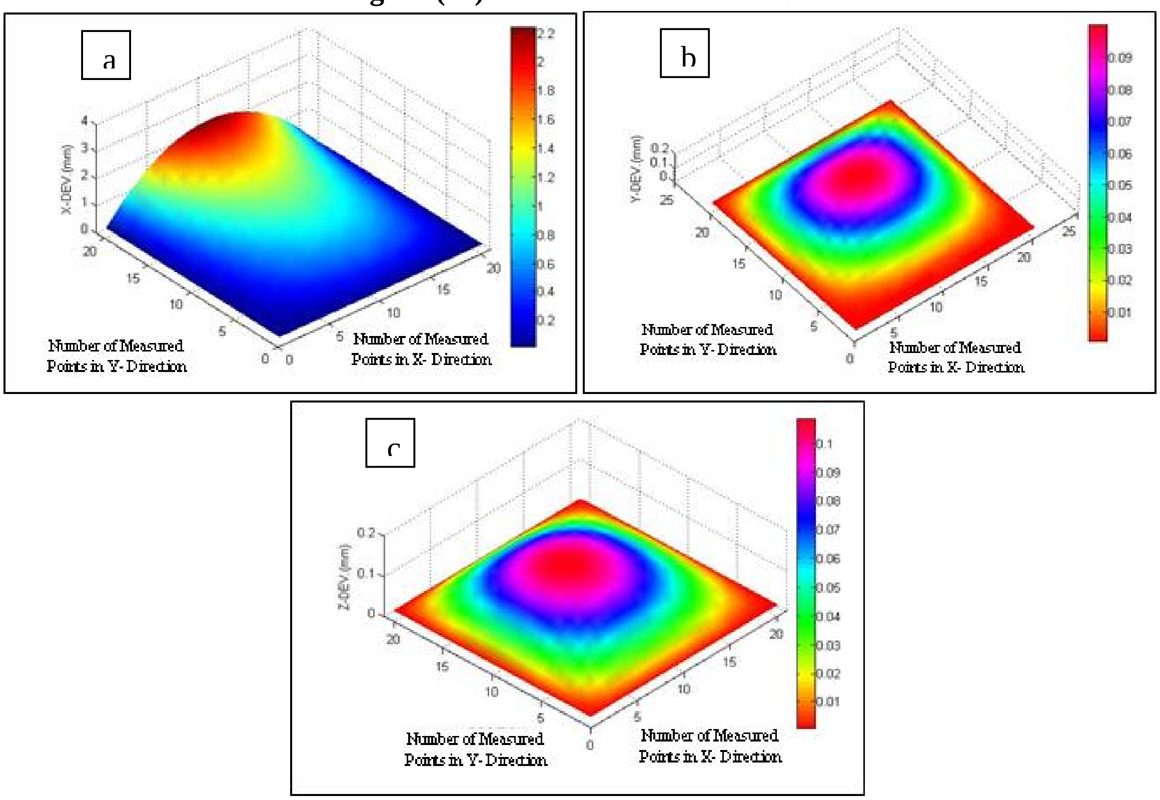

Figure (17) Error Distribution Between the Measured and Predicted surfaces, (a) X-axis Error, (b) Y-axis Error, (c) Z-axis Error

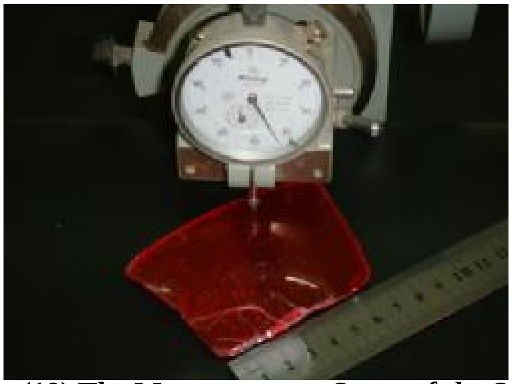

Figure (18) The Measurements Setup of the Second Example.

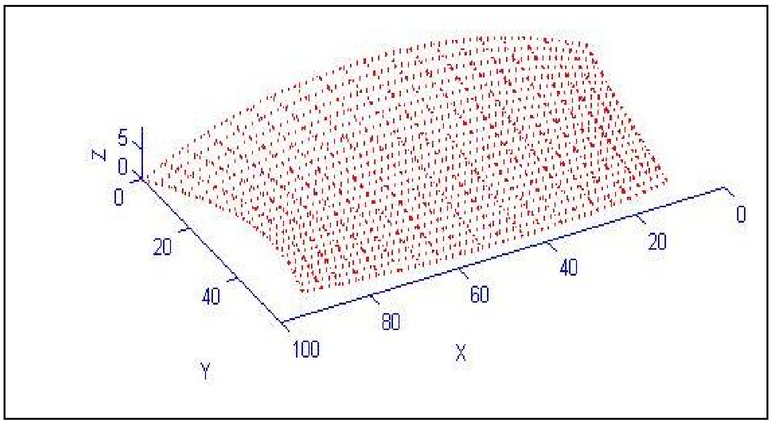

Figure (19) The Measured Data Points of the Second Example.

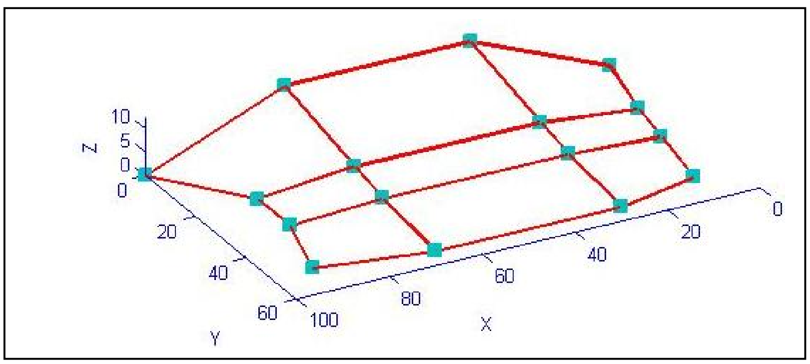

Figure (20) The Estimated Control Points of the intended Surface

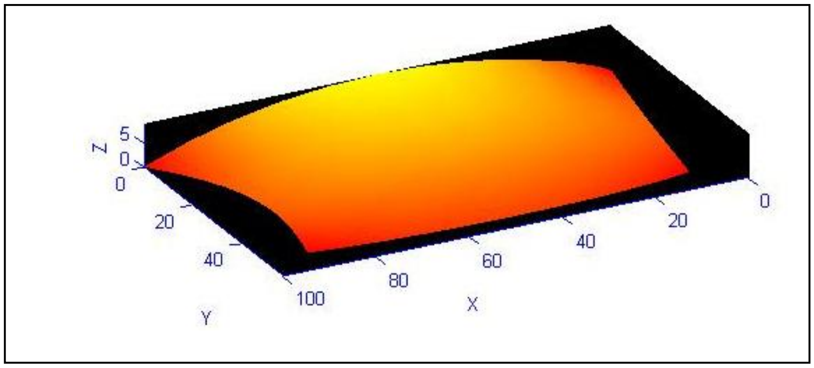

Figure (21) The Estimated Surface of a Light Cover.

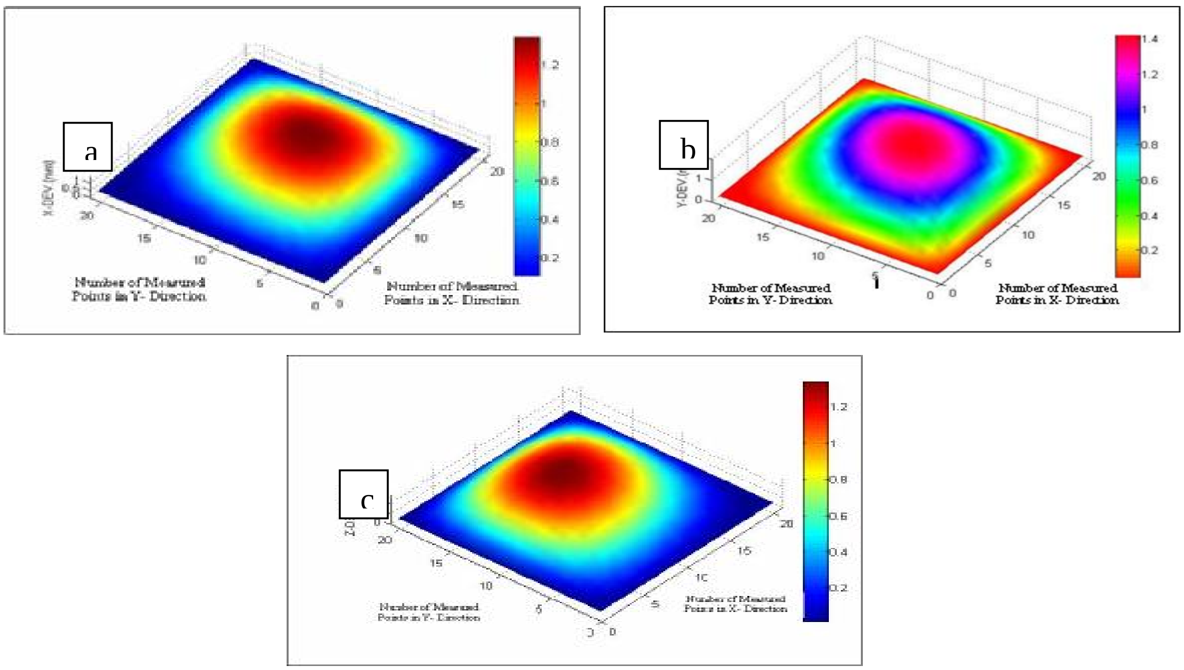

Figure (22) Error Distribution Between the Measured and Predicted surfaces, (a) X- axis Error, (b) Y-axis Error, (c) Z-axis Error.