Filling N-Sided Patch with a Trimmed Parametric Surface*

(N-Sided Surface Construction for Car Body Design)

Bingyan ZHAO** and Shigenori OKUBO**

In this paper, an algorithm of filling N-sided patch with a trimmed bipolynomial parametric surface is proposed. First, a virtual boundary is used and the N-sided curves are treated as some discrete points. Then using an algorithm of fitting arbitrary scattered points to parametric surface, a smoothly large piece of surface can be generated to approximate the N-sided curves within a given tolerance. In the end, the N-sided surface is produced by trim operation. Using this algorithm, the shape of the initial produced surface can be modified in an interactive way by pulling some points or curves on it. In order to approximate G^1 continuity with adjacent surfaces, Levenberg-Marquardt optimal method is used for final refinement. Several examples are given out to show the effectiveness of the proposed algorithm.

Key Words: Computer Aided Design (CAD), Modeling, Drawing, Surface Fitting, N-Sided Surface Patch

1. Introduction

A rectangular array of bipolynomial patches in the form of tensor product B-spline is widely used in the computer aided design of smooth surfaces. However, many surfaces that contain nonregular patches cannot be modelled parametrically by bipolynomial patches on a planar rectangular domain⁽¹⁾⁻⁽³⁾. To address these problems, various methods of modeling such surfaces have been suggested. Compared to the standard tensor-product B-spline representation, these constructions either sacrifice the low degree of the surfaces^{(4),(5)} or depart from the standard polynomial representation⁽⁶⁾⁻⁽⁸⁾. However the main drawback of the patching approach seems to arise from the very tool that allows it to model complex smooth surfaces, namely reparametrization when crossing from one patch to the next. As Ref.(9) pointed out, this shifts to focus from the geometry of

the modeling problem to clever use of the chain rule. Thus the algorithms of Goodman⁽¹⁰⁾, Hollig and Mogerle⁽¹¹⁾ fix the reparametrizations a priori dependent only on the number of the patches joining at a mesh points. In order to match geometric continuity, a large sparse linear system is solved. This makes it tricky to reason a priori about the shape of the surface⁽⁹⁾.

The general method for modeling N-sided holes in a known surface is known as the trimming of surface⁽¹²⁾. The trimmed surface is generated by mapping the intersections of surfaces in 3D space into 2D (u, v) space. Therefore it has the advantage that the surface patch maintains its former shape after the trimming operation. Consequently it is naturally to have the idea of fitting the N-sided curves to a large piece of surface and then performing trim operation using the N-sided curves as boundary to obtain the N-sided surface. Because the trimmed N-sided surface does not create triangular surfaces with degenerate sides, it is compatible with manufacturing systems where tool gouging as a result of a degenerate surface is avoided. Although the idea can be found in some modeler e.g. Alias Studio, few papers can be found.

* Received 19th February, 1999

** Department of Mechanical Systems Engineering,
Faculty of Engineering, Yamagata University, 4-3-16
Jonan, Yonezawa 992-8510, Japan. E-mail: tc026
@dip.yz.yamagata-u.ac.jp

In this paper we propose an algorithm to filling N-sided curves with a trimmed surface. This algorithm treats the N-sided curves as discrete points and then fitting the scatter points to B-spline surface. For a trimmed N-sided surface, optimal method is used to make G^1 continuity with adjacent surface patches.

The structure of this paper is as follows. Section 2 states the algorithm of fitting points to trimmed curve. Section 3 states the algorithm of filling the N-sided surface to with trimmed surface. Section 4 states the shape modification of N-sided trimmed surface. Section 5 states the using of Levenberg-Marquardt⁽¹³⁾ method to make G^1 continuity with adjacent surface patches. Section 6 gives out some examples and the last section summarizes the results.

2. Fitting the Points with Trimmed Curve

In this section we introduce an algorithm of fitting points to a trimmed curve and give out some basic concepts for the convenience of description in next section.

For a giving set of points P_j ($j=1, 2, \dots, M$) and parameter sequence u_j ($j=1, 2, \dots, M$), finding a parametric curve with control points C_i ($i=1, 2, \dots, m$). Then we have

$$P(u_j) = \sum_{i=1}^m B_i(u_j) \cdot C_i = P_j \quad (j=1, 2, \dots, M). \quad (1)$$

The matrix equation is

$$B \cdot C = P, \quad (2)$$

B is basic function matrix of Bezier or B-spline, C is control points matrix and P is the given points matrix. In the sense of least squares we have

$$\min (\|BC - P\|_a^2 + \|B''C\|_b^2). \quad (3)$$

Where $\|\cdot\|_a^2$ is denoted as $\|X\|_a^2 = \text{Tr}(X^T \cdot a \cdot X)$, $\text{Tr}(X)$ is the trace of matrix X .

Hence the control points of curve is

$$C = (B^T \cdot a \cdot B + B''^T \cdot b \cdot B'')^{-1} \cdot B^T \cdot a \cdot P. \quad (4)$$

Where $C \in R^{m \times 1}$, $B \in R^{M \times m}$, $B'' \in R^{M \times m}$, $P \in R^{M \times 1}$,

$$a = \text{diag}(a_j) \in R^{M \times M}, b = \text{diag}(b_i) \in R^{m \times m}.$$

a and b are approximation weight and smooth weight respectively. B'' is the second derivative of the basic function. $\|B''C\|$ plays a part of constraint. The number of constraints equals to the number of control points m . Test shows that there is no benefit of taking more constraints than the number of control points m .

It is well known that the parameter u_i of point P_i plays a very important part in curve fitting. As chord length parameterization usually produces a better result, here we adopt the chord length method.

Each measured point may have an error during CMM measurement, however, the first and last measured point P_0 and P_M are usually taken the same as first and last control points C_0 and C_m in curve design. Consequently point P_0 and P_M may have a bad effect

Fig. 1 Fitting points with trimmed parametric curve

on the shape of curve. This may be found in Fig. 1 (a). The curvature plot shows that the curvature is negative at the end point because of the Bessel end condition. Clearly, the shape is not satisfactory. For this reason we introduce a trimmed curve scheme to solve the above problem.

Firstly, two virtual end points P_{-1} and P_{M+1} are inserted into the given points set. The relative parameters are $t_{-1}=0.0$ and $t_{M+1}=1.0$ respectively. Then transform t_i into u_i by a linear map as following.

$$t_i = e + g \cdot u_i \quad (i=0, 1, \dots, m; 1 > e > 0, 1 > g > 0), \quad (5)$$

where e and g are coefficients which may take 0.1 and 0.8 in application. For the given points P_i ($i=-1, 0, \dots, M, m+1$) and parameters t_i ($i=-1, 0, \dots, M, m+1$), a new curve is obtained as shown in Fig. 1(b). Finally a better curve is produced by trim operation as shown in Fig. 1(c).

If we use the natural end condition as shown in Fig. 1, the curvature at the end is set to zero which is different from the inner part of the curve. Obviously it will improve the curve in some extent by the trimmed curve algorithm. In a word, the trimmed scheme can obtain a real natural end condition in some extent. We extend this method to the algorithm of filling N-sided surface with trimmed surface in next section.

3. Filling the N-Sided Patch with Trimmed Surface

Find a B-spline surface

$$S(u, v) = \sum_i \sum_j C_{ij} B_i(u) B_j(v), \quad (6)$$

that fitting the given B-spline curves P_l ($l=1, 2, \dots, h$) as

$$P_l(t) = \sum_i B_i(t) V_i \quad (l=1, 2, \dots, h). \quad (7)$$

Then we have the follows

$$S(u, v) = P_l(t), \quad (l=1, 2, \dots, h). \quad (8)$$

This becomes very complicated for numerical analysis. Reference (14) tried to give out an exact solution of Eq. (8) by representation of composition of splines and products of splines. However it turns to the least square method when the equation is singular. Consequently from view of practice and simplicity referring to Fig. 2, we discrete curves into many scatter points and fitting the scatter points to a surface,

Fig. 2 Filling the N-sided patch with trimmed parametric surface

and then by trim operation, produce a N-sided trimmed surface.

3.1 Fitting scatter points to a surface

To find a parameter surface with control points C_{ij} ($i=1, 2, \dots, m, j=1, 2, \dots, n$) to given scattered data points P_k and parameters (u_k, v_k) ($k=1, 2, \dots, L$), we get

$$S(u_k, v_k) = \sum_{i=1}^m \sum_{j=1}^n B_i(u_k) B_j(v_k) C_{ij} = P_k. \quad (9)$$

Changing Eq. (9) into matrix form yields

$$\mathbf{B}_{uk} \cdot \mathbf{C} \cdot \mathbf{B}_{vk}^T = \mathbf{P}_k. \quad (10)$$

Clearly Eq. (10) may be singular and unsolvable. Therefore we add some constraints to this equation and use the least square method. This yields

$$\text{Min} (\|\mathbf{B}_{uk} \mathbf{C} \mathbf{B}_{vk}^T - \mathbf{P}_k\|_2^2 + \|\mathbf{B}_{uk}' \mathbf{C} \mathbf{B}_{vk}^T\|_2^2 + \|\mathbf{B}_{uk} \mathbf{C} \mathbf{B}_{vk}''^T\|_2^2 + \|\mathbf{B}_{uk}' \mathbf{C} \mathbf{B}_{vk}'^T\|_2^2). \quad (11)$$

The last three items are the constraints of smoothness. Then the following equation holds.

$$2 \cdot \mathbf{B}_{uk}'^T \cdot \mathbf{a}_u \cdot (\mathbf{B}_{uk} \cdot \mathbf{C} \cdot \mathbf{B}_{vk}^T - \mathbf{P}_k) \cdot \mathbf{a}_v \cdot \mathbf{B}_{vk} + 2 \cdot \mathbf{B}_{uk}'^T \cdot \mathbf{b}_u \cdot (\mathbf{B}_{uk}' \cdot \mathbf{C} \cdot \mathbf{B}_{vk}^T) \cdot \mathbf{b}_v \cdot \mathbf{B}_{vk}' + 2 \cdot \mathbf{B}_{uk}'^T \cdot \mathbf{b}_u \cdot (\mathbf{B}_{uk} \cdot \mathbf{C} \cdot \mathbf{B}_{vk}''^T) \cdot \mathbf{b}_v \cdot \mathbf{B}_{vk}'' = 0 \quad (12)$$

Solving the linear Eq. (12) yields the control points of surface

$$\mathbf{y} = \left\{ \sum_{k=1}^L (\mathbf{A}_k \otimes \mathbf{B}_k^T) + \sum_{i=1}^3 (\mathbf{A}_{ei} \otimes \mathbf{B}_{ei}^T) \right\}^{-1} \sum_{k=1}^L \mathbf{q}(\mathbf{D}_k). \quad (13)$$

$\otimes$ is Kronecker product of matrices. $\mathbf{q}$ is a linear map that transforms a matrix into a vector, i.e.

$$\mathbf{q}: \mathbf{R}^{m \times n} \Rightarrow \mathbf{R}^{mn \times 1}, \text{ hence, } \mathbf{q}(\mathbf{C}) = \mathbf{y} \in \mathbf{R}^{mn \times 1},$$

$$\mathbf{A}_k = \mathbf{B}_{uk}'^T \cdot \mathbf{a}_{uk} \cdot \mathbf{B}_{uk}, \quad \mathbf{B}_k = \mathbf{B}_{vk}'^T \cdot \mathbf{a}_{vk} \cdot \mathbf{B}_{vk},$$

$$\mathbf{A}_{ei} = \mathbf{B}_{uei}'^T \cdot \mathbf{b}_{uei} \cdot \mathbf{B}_{uei}, \quad \mathbf{B}_{ei} = \mathbf{B}_{vei}'^T \cdot \mathbf{b}_{vei} \cdot \mathbf{B}_{vei}$$

$$(i=1, 2, 3)$$

$$\mathbf{B}_{ue1} = \mathbf{B}_{u'}', \quad \mathbf{B}_{ue2} = \mathbf{B}_u, \quad \mathbf{B}_{ue3} = \mathbf{B}_{u'}',$$

$$\mathbf{B}_{ve1} = \mathbf{B}_v, \quad \mathbf{B}_{ve2} = \mathbf{B}_{v'}', \quad \mathbf{B}_{ve3} = \mathbf{B}_{v'}',$$

$$\mathbf{D}_k = \mathbf{B}_{uk}'^T \cdot \mathbf{a}_{uk} \cdot \mathbf{P}_k \cdot \mathbf{a}_{vk} \cdot \mathbf{B}_{vk}, \quad \mathbf{B}_u \in \mathbf{R}^{M \times m},$$

$$\mathbf{B}_v \in \mathbf{R}^{N \times n}, \quad \mathbf{B}_{vei} \in \mathbf{R}^{n \times n}, \quad \mathbf{B}_{uei} \in \mathbf{R}^{m \times m},$$

$$\mathbf{a}_u = \text{diag}(\mathbf{a}_{ui}), \quad \mathbf{a}_v = \text{diag}(\mathbf{a}_{vi}),$$

$$\mathbf{b}_u = \text{diag}(\mathbf{b}_{ui}), \quad \mathbf{b}_v = \text{diag}(\mathbf{b}_{vi}).$$

Matrices $\mathbf{B}'$ and $\mathbf{B}''$ are first and second derivative with respect to parameter u and v respectively. The number of constraints is taken to number of control points of $n \times m$. It is easy to prove that Eq.

Fig. 3 Greville abscissae and control points of B-spline curve

Fig. 4 Local coordinate system (x', y', z')

(13) is solvable if a and b are positive.

3.2 Calculating the parameters of scatter points

It is very important to find suitable parameters of the given points for surface design. Numerous papers have been published on the selection of such parameters. All of these methods are based on the assumption that the points are arranged in a special pattern, i.e. chain points for curves, and grid points for surfaces. These methods fail completely if the given points are spaced as in Fig. 2. In this section we give out a simply method to compute the parameters on the hypothesis that all the points have unique projection points on its least square plane.

3.2.1 Sampling points and finding the least square plane Referring to Fig. 3, for a B-spline curve $P_l(t)$ ($l=1, 2, \dots, h$) with control points C_i^l ($i=1, 2, \dots, m_l$) there are m_l Greville abscissae on the curve which correspond to control vertices. Hence a given vertex will most strongly influence the curve in the neighborhood defined by its Greville abscissa. Consequently these m_l Greville abscissae and two more points between them are sampled. Therefore the total numbers of sampling points of N-sided curves are

$$m_{total} = \sum_{l=1}^h (3m_l - 2) = 3 \sum_{l=1}^h m_l - 2h. \quad (14)$$

Test shows that there is no advantage of sampling more than m points. Then find the least square plane H of the sampling points,

$$Ax + By + Cz + D = 0. \quad (15)$$

3.2.2 Construction of local coordinate system (X', Y', Z') Referring to Fig. 4, let us take one curve $P_l(t)$ from among the N-sided curves and draw a line L between the end point of $P_l(t)$. Then take middle point O' of line L as the origin point of local coordinate system (X', Y', Z') and make line L as X'

axis. Then a line is taken as Y' axis. This line passes through O' , keeps parallel with plane H and is vertical to $O'X'$. Finally, Z' axis is constructed from X' axis and Y' axis in that $Z' = X' \times Y'$.

3.2.3 Coordinate transformation First, we obtain the points $P(x'_k, y'_k, z'_k)$ by transforming the sampled points into coordinate system (X', Y', Z') . Then find a minimum rectangle that contains all points (x'_k, y'_k) as shown in Fig. 5 in $X'O'Y'$ plane. Furthermore move the coordinate system (X', Y', Z') to the left bottom point of the rectangle to construct a new coordinate system (X'', Y'', Z'') . Finally transform the sampled points into coordinate system (X'', Y'', Z'') to get the points $P(x''_k, y''_k, z''_k)$.

3.2.4 Computing the parameters of sampled points Now we take coordinate (x''_k, y''_k) as the parameters of given sampled points P_k .

We find the maximum values of x''_k and y''_k as follows in order to normalize the parameters.

$$X_{\max} = \max(x''_k), \quad (16)$$

$$Y_{\max} = \max(y''_k). \quad (17)$$

Then normalize the parameters (s_k, t_k) as follows

$$s_k = \frac{x''_k}{X_{\max}} \in [0.0, 1.0], \quad (18)$$

$$t_k = \frac{y''_k}{Y_{\max}} \in [0.0, 1.0]. \quad (19)$$

Finally as shown in Fig. 6, the parameters (u_k, v_k) of sampled points P_k ($k=1, 2, \dots, m_{\text{total}}$) are

$$u_k = e + g \cdot s_k \quad (k=1, 2, \dots, m_{\text{total}}; 1 > e > 0, 1 > g > 0), \quad (20)$$

$$v_k = e + g \cdot t_k \quad (k=1, 2, \dots, m_{\text{total}}; 1 > e > 0, 1 > g > 0), \quad (21)$$

where we take $e=0.1$ and $g=0.8$. In practical design application, the shape of N-sided curves is usually not so complex because the relationship between 2D $(u,$

Fig. 5 Parameter coordinate system

Fig. 6 Transform the parameter (s, t) to (u, v)

$v)$ space and 3D space is a one to one map in most cases. If it is not the one to one map, Eq.(13) can still be solvable but the shape of the surface becomes bad.

4. Modification of the N-Sided Surface

From Eq.(13), we can find that it is easy to add one point P_j as a constraint and to move the surface to this point within a given tolerance. Rewrite Eq.(13) into the following

$$y = (A_j \otimes B_j^T + E_1)^{-1} \cdot (q(D_j) + E_2). \quad (22)$$

Where

$$E_1 = \sum_{k=1}^L (A_k \otimes B_k^T) + \sum_{i=1}^3 (A_{ei} \otimes B_{ei}^T),$$

$$E_2 = \sum_{k=1}^L q(D_k)$$

$$A_j = B_u^T \cdot a_u \cdot B_u, \quad B_j = B_v^T \cdot a_v \cdot B_v$$

$$D_j = B_u^T \cdot a_{ju} \cdot P_j \cdot a_{jv} \cdot B_v.$$

After the N-sided trimmed surface being fitted, matrix E_1 and E_2 will keep constant in the modification operation. In this way we can move a point on the N-sided trimmed patch to a given point position interactively. Furthermore we can move a curve on surface to a given curve within a given tolerance. Generally speaking, it is easy to control the shape interactively if the point is moved along normal line.

5. Smoothing the Trimmed Surface to G^1 Continuity with Adjacent Surfaces

Since G^1 continuity of necessary and sufficient conditions of a trimmed B-spline surface have not been published, we have also used the discrete approximation to check for G^1 continuity. This is easy to use in practical application. The checkpoints are chosen according to the concept of section 3.2.1. Therefore a pair of surface satisfying this G^1 test will be approximate to G^1 continuity along the boundaries. See in Fig. 7, suppose that at all sampled points along the N-sided edge, the unit normal vectors of trimmed surface are n_{1i} and unit normal vectors of adjacent

Fig. 7 Tangent and normal vector notation of trimmed surface and adjacent surface

surface are $\mathbf{n}_{2i}$. Then error at these checkpoint are $\mathbf{r}_i = \mathbf{n}_{1i} - \mathbf{n}_{2i}$. Consequently the objective function is the sum of squares of the three components of error vectors $\mathbf{r}_i$:

$$f(Cx, Cy, Cz) = a \sum_{i=1}^{m_{total}} (r_{ix}^2 + r_{iy}^2 + r_{iz}^2) + b \|\Delta P\|^2, \quad (22)$$

where ΔP is the constraints that keep the optimal surface pass through the discrete points on N -sided curves, a and b are constants. This function formulation penalizes G^1 errors at the nodal vertices more heavily as these vertices are common to more than one boundary. This inconsistency can be safely ignored since f tends to zero in all the cases tried.

The optimization problem is a nonlinear least-square problem, and it is efficiently minimized using the Levenberg-Marquardt (LM) method. The LM method uses small, the steepest descent steps in the difficult, nonlinear regions of the objective function, and shifts to larger, quadratic steps as the solution is approached. This method requires the gradient of the objective function with all the available degree of the freedom. It is impractical to analytically compute the partial derivatives because of the complexity of the objective function. Instead we have discretely approximated the gradient using forward differences, where the derivative for $f(x)$ is

$$f'(x) = \frac{f(x+h) - f(x)}{h}. \quad (23)$$

The value of h must be carefully chosen. A small h increases the roundoff error and a larger h increases the discretization error. We chose the $h = 0.01 \|d_{avg}\|$ where d_{avg} is the average distance between vertices in control net.

6. Examples of Application

In this section we give out some examples to illustrate effectiveness of the proposed algorithm. Figure 8(a) gives out three faces of a 100 mm $\times$ 100 mm $\times$ 100 mm cube with 9-sided curves in the corner. We sampled 36 points totally. Figure 8(b) is a smoothly trimmed surface with 8 $\times$ 8 control points. During computation we take the weight $a = 100$ for all sampled points and $b = 0.1$ for all constraint points.

Fig. 8 Fitting a 9-sided surface without normal information of boundaries

The maximum distance from point to trimmed surface is 0.0035 mm. Because it does not take any normal information of the adjacent patches, referring in Fig. 8(c), shading image clearly shows that it is merely position continuity with adjacent surfaces.

Figure 9 demonstrates an optimal trimmed surface produced by LM method. In this example the number of control points is 8 $\times$ 8, therefore the number of freedoms is 192. Finishing the model takes 3 minutes 38 seconds in a Pentium II/233MHZ. The maximum error is 0.08 mm. The average error is 0.04 mm. The weight $a = 10$ for all points and $b = 1$ for sampling points in Eq.(22). Here we did not take account of smoothness, however the result is still relatively smooth. If it did not meet the smooth requirement, a smoothness constraint should be added to the Eq.(22). We take a small initial step = 0.05 for the LM method. The initial values are the control points of the produced trimmed surface as shown in Fig. 8. We can first modify the shape of surface interactively, and use the optimal method for refinement because the initial values play an important part in convergence of LM method.

Figure 10 illustrates the shape modification of a trimmed surface. In order to get a desired shape, one can first move points to a specified position to get a rough shape and then changing the value of approximation weight a to refine the shape. In this example, using $a = 50$, we take three points on surface and move each point in normal direction for 5 mm, one point in center ($u = 0.5, v = 0.5$) and move the center point in normal direction for 10 mm. Then we get shape of

Fig. 9 An optimal trimmed surface that has G^1 continuity with adjacent patches

Fig. 10 Modification of the trimmed surface by pulling points in normal direction

Fig. 11 Comparison with loft (b) and Coons (b) surface modeling methods

Fig. 12 A car body model finished using the proposed algorithm

Fig. 13 A shading image of the car body model

surface as shown in Fig. 10(b), (c).

Figure 11 is a corner part of a car trunk. Figure 11(a) is the given information. In order to illustrate the advantage of our algorithm, we use a loft, four edges Coons and our method to produce the surface. Figure 11(b) is a lofted surface of which the maximum distance is 0.88 mm. The average error is 0.11 mm. Figure 11(c) is a Coons surface of which the maximum distance is 0.98 mm, the average error is 0.37 mm. Figure 11(d) is a trimmed surface of which the maximum error is 0.725 mm and average error is 0.14 mm.

Figure 11(b) seems better, however it does not pass through the boundary completely. Only the trimmed one use the wholly information of points and curves. We find that the trimmed surface (d) is the best one of meeting the requirement of design.

Figure 12 gives out a model of an car body. Our main purpose of this paper is to develop a surface design algorithm for car body design. During car body

design, it is usually first measuring points from clay models and then fitting the measured points to surfaces of CAD model. There are many examples of fitting scatter data points to smooth surfaces and fitting N-sided curves to a surface in car body design process. For example the window surface, the fender surface, the corner part of a pillar and roof are all N-sided problems. Even though the model proposed here is not completely finished using our algorithm have drastically speeded up the design. For 3984 measured points, we took 2 days to finish the model as shown in Fig. 12 and Fig. 13. We used 850 curves and 84 surfaces in half model of the car.

6. Conclusion

A practical algorithm of filling N-sided patch with trimmed surface has been proposed in this paper. Through the application we found that the proposed method has more advantages than the Bézier or Gregory composite surfaces scheme :

- It is difficult to achieve high continuity G^n ($n > 1$) at a common vertex for the composite surfaces method, however the trimmed surface can achieve high geometric continuity by increase its degree.

- If the boundary curves of N-sided patch are trimmed curves, the composite surfaces method can hardly to use while the proposed method can do.

- The trimmed surface scheme can generate only one piece of surface while the composite surface scheme generate N pieces of surface that make the size of database increase.

- It can easily modify the shape of trimmed surface interactively by the proposed method while it is difficult to modify the composite surfaces interactively.

In fact this algorithm is a general method for bipolynomial surface design. It can not only fit grid or randomly distributed points to surface but also can fit arbitrary topological curves to surface. N-sided patch with trimmed surface can be approximated smoothly using this algorithm, and the shape of the initial produced surface can be modified latter in an interactive way by pulling some points on it along normal directions. LM method is used in order to maintain approximately G^1 continuity with adjacent surfaces. Through the application, we find that this algorithm is effective in car body design. The future work should be centered on finding an effective method to make two trimmed surfaces to a G^2 or higher degree of geometric continuity along a common trimmed curve.

References

- (1) Catmull, E. and Clark, J., Recursively Generated

- B-Spline Surface on Arbitrary Topological Meshes, *Computer-Aided Design*, Vol. 10, No. 6 (1978), p. 350-355.
- (2) Gregory, J.A., N-Sided Surface Patches, in Gregory, J.A., (Eds.), *The Mathematics of Surfaces*, (1986), p. 217-232, Clarendon Press, UK.
 - (3) Sabin, M.A., Non-Rectangular Surface Patches Suitable for Inclusion in a B-Spline Surface, *Proc. Eurographics '83*, North-Holland, (1983), p. 57-69.
 - (4) Sarraga, R.F., G^1 Interpolation of Generally Unrestricted Cubic Bezier Curves, *Computer Aided Geometric Design* 4, (1987), p. 23-40.
 - (5) Hahn, J.M., Filling Polygonal Holes with Rectangular Patches, in Straber, W. and Seidel, H.P. (Eds.), (1989), Springer-Verlag, New York.
 - (6) Chiyokura, H. and Kimura, F., Designs of Solids with Free Form Surfaces, *Computer Graphics*, Vol. 17(1983), p. 289-298.
 - (7) Loop, C. and DeRose, T., Generalized B-Spline Surfaces of Arbitrary Topology, *Computer Graphics*, Vol. 24, No. 4 (1990), p. 347-356.
 - (8) Kato, K., Generation of N-Sided Surface Patches with Holes, Vol. 23, No. 10 (1991), p. 676-683.
 - (9) Peters, J., Construction C^1 Surfaces of Arbitrary Topology Using Biquadratic and Bicubic Splines, in Sapidis, N.S., (Eds.), *SIAM*, (1994), p. 227-293, Philadelphia.
 - (10) Goodman, T.N.T., Closed Surfaces Defined from Biquadratic Splines, *Constructive Approximation* 7, (1991), p. 149-160.
 - (11) Hollig, K. and Mogerle, H., G-Splines, *Computer Aided Geometric Design* 7, (1989), p. 197-207.
 - (12) Casale, M.S., Free-Form Solid Modeling with Trimmed Surface Patches, *IEEE CG&A*, Vol. 7, No. 1 (1987), p. 33-43.
 - (13) More, J.J., The Levenberg-Marquardt Algorithm: Implementation and Theory, *Numerical Analysis*, in Watson, G.A. (Eds.), *Lecture Notes in Mathematics* 630, (1977), p. 105-116, Springer-Verlag.
 - (14) Hu, Y.P. and Sun, T.C., Moving a B-Spline Surface to a Curve—A Trimmed Surface Matching Algorithm, *Computer-Aided Design*, Vol. 29, No. 6 (1997), p. 449-455.
 - (15) Ma, W. and Kruth, J.P., Parameterization of Randomly Measured Points for Least Squares Fitting of B-Spline Curves and Surfaces, *Computer-Aided Design*, Vol. 27, No. 9 (1995), p. 663-675.
 - (16) Milory, M.J., Bradley, C., Vickers, G.W. and Weir, D.J., G^1 Continuity of B-Spline Surface Patches in Reverse Engineering, *Computer-Aided Design*, Vol. 27, No. 6 (1995), p. 471-478.