Engineering and Technology Journal
Journal homepage: engtechjournal.org

Geometric Modeling of Compound NURBS Surfaces

Ali K. Alwan [a*], Wisam K. Hamdan[b]

[a] Department of Production Engineering and Metallurgy, University of Technology-Iraq.
alikarim4847@gmail.com

[b] Department of Biomedical Engineering, University of Technology-Iraq. wisamuot@yahoocom

*Corresponding author.

Submitted: 04/06/2019 Accepted: 31/08/2019 Published: 25/02/2020

K E Y W O R D S

CAD, Geometric modeling, NURBS, Compound surfaces, Car components design, Sculptured surfaces.

A B S T R A C T

The design of sculptured surfaces occupies an essential area in the field of modern industrial, aerospace, and medical applications. The challenge is to design products that have complex features efficiently with great flexibility of editing in certain regions without affecting other regions, which the designer has no intent to modify. In this paper, we propose a surface design method based on compound NURBS surface to model automotive parts with 400 control points. First, a Non-Uniform B-Spline basis function is derived with a cubic degree and 20 control points. This method is utilized to design car posterior door, car hood, and rear car door as case studies.

How to cite this article: A. K. Alwan and W. K. Hamdan, "Geometric modeling of compound NURBS surfaces," *Engineering and Technology Journal*, Vol. 38, Part A, No. 02, pp. 277-287, 2020.

DOI: https://doi.org/10.30684/etj.v38i2A.300

1. Introduction

Wide ranges of products are designed with free-form sculptured surfaces to enhance their aesthetic appeal, which plays an essential role in customer satisfaction, especially for automobile, aerospace, and electronic components [1]. The complex geometry of the designed surface often requires the application of multi-patch surfaces. A compound surface is designed by using a number of patches, and for example, a compound surface could be designed by assembling a group of Ruled, Bezier and NURBS surface Patches. The compound surface modeling offer significant advantages

• The designers can process the particular surface patches as separate features and model these patches individually depending on the required geometrical specifications of the surface.

• The utilization of compound surfaces improves design capabilities so that the designer has more freedom in the modeling of the distinct segments of a surface rather than designing the total surface as a single patch. Furthermore, sometimes, it is not practical to design the part surface as a single patch.

• Design alterations of each surface patch without changing other surface patches is applicable. Applying this property, the designer could suppress the dispensable processing of data, for example, the tool path generation for distinct patches without generating a tool path for the unnecessary patches.

• In addition to benefits associated with product design, compound surfaces also provide advantages in generating the tool path for the finishing process. Patch-by-patch, iso-parametric machining of compound surfaces can improve the tool path planning [2].

The modeling of typical 3D- objects that used in modern industrial products involve a combination of standard primitive surfaces such as (cubes, cylinders, cones, spheres) and sculptured surfaces like (Hermite, Bezier, B-spline) surfaces or the use of composite sculptured surface patches. Many problems encountered with this combination such as

• The free form surfaces are parametric, and standard primitive surfaces are not, which complicates the modeling process.

• Many intersection problems arise while combining free-form surfaces with the standard primitives and these intersection problems require a complex solution in most times.

2. Literature Review

There is much research in the field of the geometric modeling of sculptured surfaces, Michálková and Bastl [3] studies the construction of B-spline surfaces with the satisfaction of prescribed distribution of angles of specific angles for the modeling of Pelton turbine bucket. Hu et al. [4] investigated the shape modifications of NURBS surfaces by constrained optimization and energy minimization, so the modified surface satisfies the given constraints. Hamdan and Abdul-Razaq [5] build and develop an algorithm to model multiple patches of B-spline surfaces. Their work involves the assembling of patches by a method, which permits the association of the control vertices for the joined patches, and the conjunction of patches with correspondent weights sides due to the weight change of patch. Sun et al. [6] proposed a method to modify the NURBS surface to find surface offsets avoiding self-intersections. This algorithm composed of four steps, showed efficiency, and is capable of obtaining better-offset outcomes. Chen and Fu [7] proposed a steepest ascent path for machining free form surfaces by a 3-axis milling machine during the finishing process. In their work, simple formulas of the tool paths are generated, and an improved, effective method to produce steepest ascent machining paths on compound NURBS surfaces. Yang et al. [8] proposed an algorithm for the optimization of NURBS surface conformality with the application of bilinear transformations; this algorithm provided good results for the tessellation and rendering applications.

3. The Design Procedure

The B-Spline modeling technique is different from a Bezier or Hermite curve modeling technique in the case that the curves are frequently composed of more than one segment. All the segments are designed and affected by the small number of control mesh points. B- spline curves have the following advantages:

- Curve degree does not depend on the number of control points.
- Modifications in shape do not propagate more than one or only a few local segments. Most curve design techniques do not provide local control of shape. For example, the local changes (for example, a slight modification in the coordinates of a point on a spline curve or of control points of a Bezier curve) would propagate into the entire curve. That change is often depicted as a global overspread of a local change.
- The B-Spline curve is positioned within the convex hull of the control polygon. Also, it provides C^n continuity of curve segments.
- In general, uniform B-Splines do not necessarily interpolate their first and last points. However, non-uniform B-Spline curves allow this.

I. Non-Uniform B-spline curves

The general formula of non-rational B-spline curvesis defined by non-uniform basis functions. That is, each segment may be described by different basis functions from other segments, which permits the interpolation of many points of the control polygon. This operation is done by specifying a non-decreasing set of knot values. A B-Spline curve is analogous to a Bezier curve in that the effects of n + 1 control points are combined by the basic functions. The non-rational form is given by:

$$P(u) = \sum_{i=0}^{n} P_i N_{i,k}(u) \qquad (1)$$

A parameter k controls the degree of polynomials which equal to (k- 1). B-spline curve degree frequently independent of the number of control points. The basis functions are calculated recursively by the expressions below:

$$N_{i,1}(u) = 1 \quad if \quad t \leq u < t_{i+1} = 0 \qquad (2)$$

Otherwise:

$$N_{i,k} = \frac{(u - t_i) * N_{i,k-1}}{t_{i+k-1} - t_i} + \frac{(t_{i+k} - u)\, N_{i+1,k-1}}{t_{i+k} - t_{i+1}} \qquad (3)$$

The t_i are known as knot values, and a group of these values produces a knot vector. It ties parameter u to the control points where i = 0,. . . , n. A non-uniform curve that passes through the endpoints, the knot values are determined by the following formulas depending on n & k:

$$\begin{aligned} t_i &= 0 && \text{if } i < k \\ t_i &= i - k + 1 && \text{if } i \leq k \leq n \\ t_i &= n - k + 2 && \text{if } i > n \end{aligned} \qquad (4)$$

When the t_i are determined for n & k, it used to compute the basic functions $N_{i,k}$ for all k = 1,2, . . . k. The index i on the knot values t_i restricted between 0 to (n + k). The knot values t_i also restricted to values between 0 to (n - k + 2). The index i on the basis function $N_{i,k}$ takes values from 0 to n. There are constantly n + 1 basis functions for each k. Equation that comprises the number of control polygon vertices, knots values, and the degree of the curve are given by

$$n + k + l = T \qquad (5)$$

Where T is the number of elements of the knot vector. For non-uniform and open B-Spline curves, the knot vector T is given by

$$T = [\alpha, \alpha, \ldots, \alpha, t_k, \ldots\ldots, t_{T-k-1}, \beta, \ldots\ldots, \beta, \beta]$$

Where end knots α and β repeat with multiplicity k.
If the entire curve is parameterized over the unit interval, then, for most situations, α = 0 and β = 1. However, if we assign non-decreasing integer values of the parameter to the knots, then α = 0 and β = n - k + 2 [8].

II. NURBS curves

The cause of the exhausting of the B-spline curves al than the other curve generation approaches is using knot vectors. The *T* knots could be utilized as variables and be changed by the user to achieve the wanted design of the curve. The rational B-spline curves introduce a new set of n + 1 parameter *wi*, called *weights*, to provide more flexibility to the design of the curve. Perhaps the best way to introduce rational B-splines (and rational curves in general) is employing homogeneous coordinates. This method starts by adding an extra dimension to points, so a two-dimensional point becomes a triplet (x, y, w), and a three-dimensional point becomes a 4-tuple (x, y, z, w). After transforming or manipulating the point, it is projected back to its original number of dimensions by dividing its coordinates by w. Given four-dimensional control points $\mathbf{Q}i = (x_i, y_i, z_i, w_i)$, where we assume for convenience that the w_i coordinates are nonnegative, we can define a (non-rational) B-spline curve as

$$P_{nr}(u) = \sum_{i=0}^{n} P_i N_{i,k}(u) \qquad (6)$$

From this, we get the rational B-spline $\mathbf{P}_r(t)$ by isolating that part of $\mathbf{P}_{nr}(t)$ that depends on the fourth coordinates *wi* and dividing by this part.

$$P_r(u) = \frac{\sum_i^n P_i\, w_i\, N_{i,k}(u)}{\sum_i^n w_i\, N_{i,k}(u)} = \sum_i^n P_i\, R_{i,k} \qquad (7)$$

Where $\mathbf{P}_i = (x_i, y_i, z_i)$ are three-dimensional control points and $R_{ik}(t)$ are the new, rational blending functions defined by [9]

$$R_{i,k}(u) = \frac{w_i\, N_{i,k}(u)}{\sum_i^n w_i\, N_{i,k}(u)} \qquad (8)$$

III. Proposed compound surface design methodology

The authors propose using NURBS surfaces with 17*17 patches (400 control points) to design surfaces to be used in the experimental work of the thesis. First, we derive non-uniform B-spline curve basis function using k=4 (cubic degree) parametric range (0-17) and n=19 (20 control points) using equations 1,2,3,4 and 5) to obtain the following matrices:

$$M_1 = \begin{bmatrix} -1 & \frac{7}{4} & -\frac{11}{12} & \frac{1}{6} \\ 3 & -\frac{9}{2} & \frac{3}{2} & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \qquad M_2 = \begin{bmatrix} -\frac{1}{4} & \frac{7}{12} & -\frac{1}{2} & \frac{1}{6} \\ \frac{3}{2} & -3 & 2 & -\frac{1}{2} \\ -3 & \frac{9}{2} & -2 & \frac{1}{2} \\ 2 & -\frac{3}{2} & \frac{2}{3} & -\frac{1}{6} \end{bmatrix}$$

$$M_3 = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{3}{2} & -4 & \frac{7}{2} & -1 \\ -\frac{9}{2} & 10 & -\frac{15}{2} & 2 \\ \frac{9}{2} & -\frac{22}{3} & \frac{31}{6} & -\frac{4}{3} \end{bmatrix} \qquad M_4 = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ 2 & -\frac{11}{2} & 5 & -\frac{3}{2} \\ -8 & \frac{39}{2} & -16 & \frac{9}{2} \\ \frac{32}{3} & -\frac{131}{6} & \frac{50}{3} & -\frac{9}{2} \end{bmatrix}$$

$$M_5 = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{5}{2} & -7 & \frac{13}{2} & -2 \\ -\frac{25}{2} & 32 & -\frac{55}{2} & 8 \\ \frac{125}{6} & -\frac{142}{3} & \frac{229}{6} & -\frac{32}{3} \end{bmatrix} \qquad M_6 = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ 3 & -\frac{17}{2} & 8 & -\frac{5}{2} \\ -18 & \frac{95}{2} & -42 & \frac{25}{2} \\ 36 & -\frac{521}{6} & \frac{218}{3} & -\frac{125}{6} \end{bmatrix}$$

$$M_7 = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{7}{2} & -10 & \frac{19}{2} & -3 \\ -\frac{49}{2} & 66 & -\frac{119}{2} & 18 \\ \frac{343}{6} & -\frac{430}{3} & \frac{739}{6} & -36 \end{bmatrix} \qquad M_8 = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ 4 & -\frac{23}{2} & 11 & -\frac{7}{2} \\ -32 & \frac{175}{2} & -80 & \frac{49}{2} \\ \frac{256}{3} & -\frac{1319}{6} & \frac{578}{3} & -\frac{343}{6} \end{bmatrix}$$

$$M_9 = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{9}{2} & -13 & \frac{25}{2} & -4 \\ -\frac{81}{2} & 112 & -\frac{207}{2} & 32 \\ \frac{243}{2} & -\frac{958}{3} & \frac{1705}{6} & -\frac{256}{3} \end{bmatrix} \qquad M_{10} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ 5 & -\frac{29}{2} & 14 & -\frac{9}{2} \\ -50 & \frac{279}{2} & -130 & \frac{81}{2} \\ \frac{500}{3} & -\frac{2669}{6} & \frac{1202}{3} & -\frac{243}{2} \end{bmatrix}$$

$$M_{11} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{11}{2} & -16 & \frac{31}{2} & -5 \\ -\frac{121}{2} & 170 & -\frac{319}{2} & 50 \\ \frac{1331}{6} & -\frac{1798}{3} & \frac{3271}{6} & -\frac{500}{3} \end{bmatrix} \qquad M_{12} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ 6 & -\frac{35}{2} & 17 & -\frac{11}{2} \\ -72 & \frac{407}{2} & -192 & \frac{121}{2} \\ 288 & -\frac{4715}{6} & \frac{2162}{3} & -\frac{1331}{6} \end{bmatrix}$$

$$M_{13} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{13}{2} & -19 & \frac{37}{2} & -6 \\ -\frac{169}{2} & 240 & -\frac{455}{2} & 72 \\ \frac{2197}{6} & -\frac{3022}{3} & \frac{5581}{6} & -288 \end{bmatrix}$$

$$M_{14} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ 4 & -\frac{41}{2} & 20 & -\frac{13}{2} \\ -98 & \frac{559}{2} & -266 & \frac{169}{2} \\ \frac{1372}{3} & -\frac{7601}{6} & \frac{3530}{3} & -\frac{2197}{6} \end{bmatrix}$$

$$M_{15} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{15}{2} & -22 & \frac{43}{2} & -7 \\ -\frac{255}{2} & 322 & -\frac{615}{2} & 98 \\ \frac{1125}{2} & -\frac{4702}{3} & \frac{8779}{6} & -\frac{1372}{3} \end{bmatrix}$$

$$M_{16} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{2} & -\frac{7}{12} & \frac{1}{4} \\ 8 & -\frac{47}{2} & \frac{107}{4} & -\frac{45}{4} \\ -128 & \frac{735}{2} & -\frac{1633}{4} & \frac{675}{4} \\ \frac{2048}{3} & -\frac{11471}{6} & \frac{24887}{12} & -\frac{3375}{4} \end{bmatrix}$$

$$M_{17} = \begin{bmatrix} -\frac{1}{6} & \frac{11}{2} & -\frac{7}{4} & 1 \\ \frac{17}{2} & -\frac{181}{4} & \frac{339}{4} & -48 \\ -\frac{289}{2} & \frac{2975}{4} & -\frac{5469}{4} & 768 \\ \frac{4913}{6} & -\frac{48841}{12} & \frac{29393}{4} & -4096 \end{bmatrix}$$

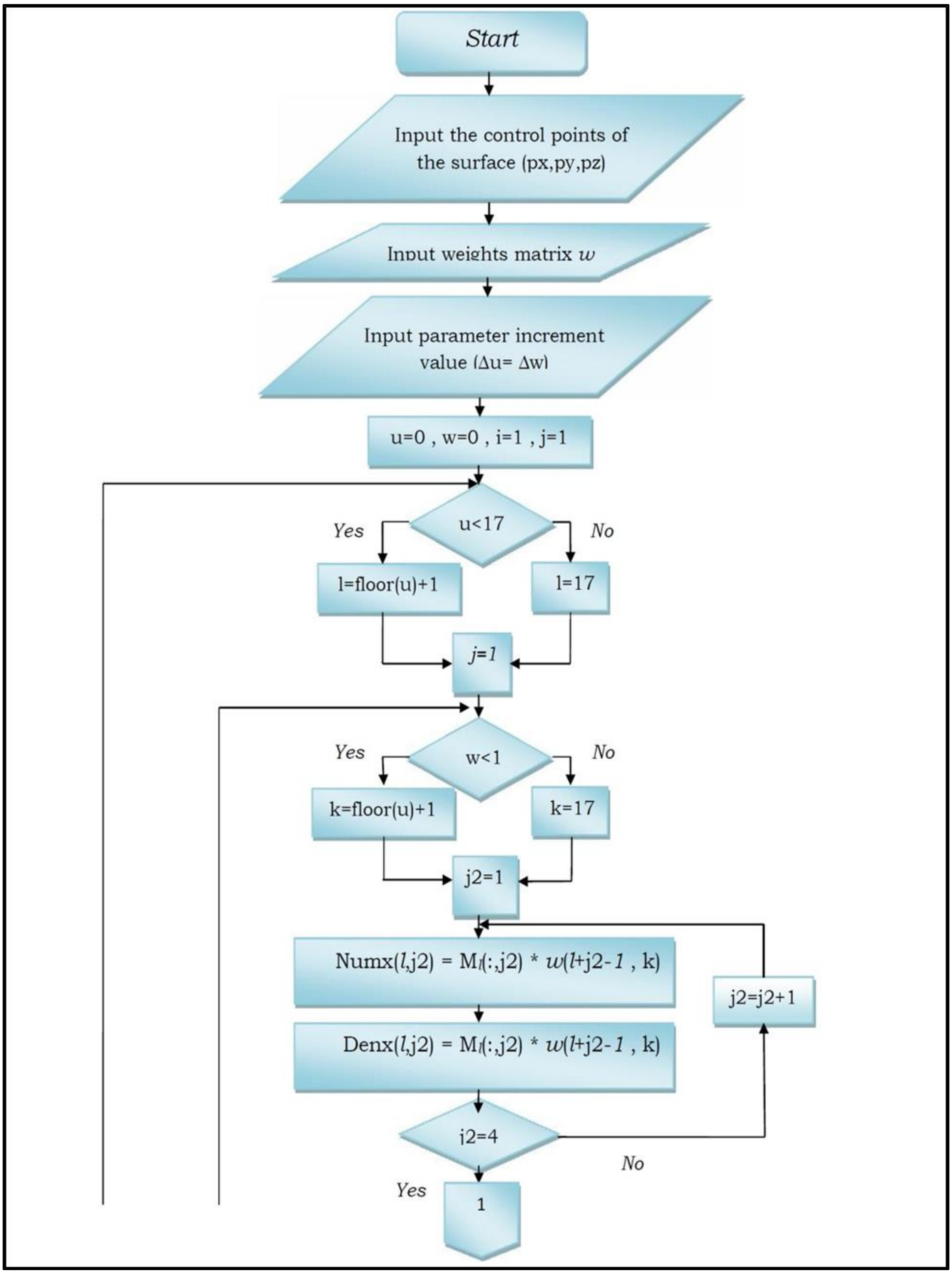
We can note that all derived matrices undergo the convex hull property ($\sum_{i=0}^{n} P_i(u) = 1$), which means that the curve lies within the control point's polygon, this property is beneficial in checking the derivation.

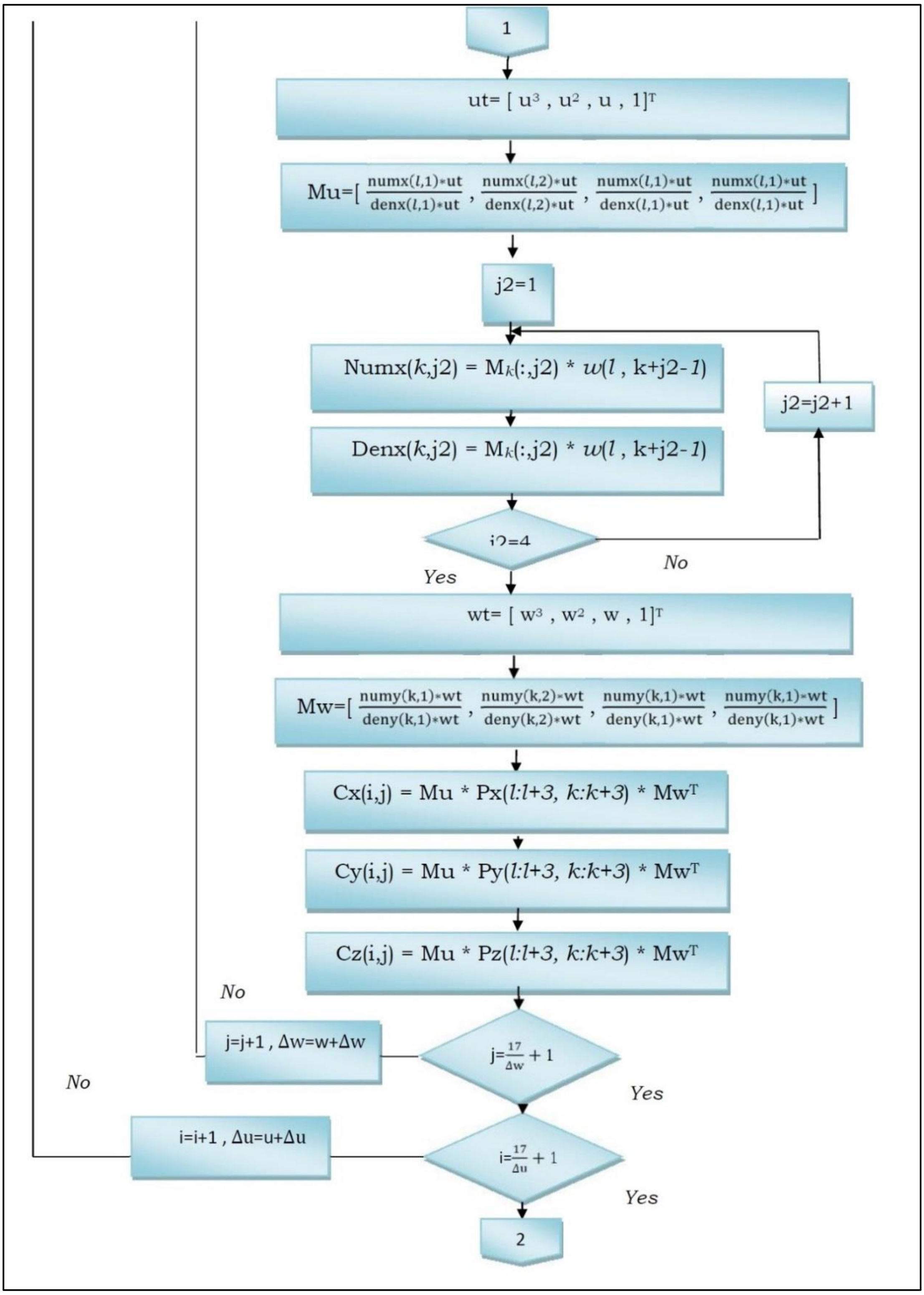
After derivation of basis functions of non-uniform a B-spline functions, we use these matrices in NURBS surface design algorithm

A NURBS surface is given by

$$S(u,w) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} W_{ij}P_{ij}\, N_{i,k}(u)\, N_{j,l}(w)}{\sum_{i=0}^{n}\sum_{j=0}^{m} W_{ij}N_{i,k}(u)\, N_{j,l}(w)} \quad (9)$$

The general discrimination between a NURBS and a B-spline modeling of surfaces is that the NURBS surface is capable of designing ordinary surfaces, like a sphere, ellipsoid, and so on.


Start
Input the control points of the surface (px,py,pz)
Input weights matrix w
Input parameter increment value (Δu= Δw)
u=0 , w=0 , i=1 , j=1
u<17
Yes
No
l=floor(u)+1
l=17
j=1
w<1
Yes
No
k=floor(u)+1
k=17
j2=1
Numx(l,j2) = Ml(:,j2) * w(l+j2-1 , k)
Denx(l,j2) = Ml(:,j2) * w(l+j2-1 , k)
j2=j2+1
j2=4
No
Yes
1


1
ut= [u3 , u2 , u , 1]T
Mu=[numx(l,1)*ut / denx(l,1)*ut , numx(l,2)*ut / denx(l,2)*ut , numx(l,1)*ut / denx(l,1)*ut , numx(l,1)*ut / denx(l,1)*ut]
j2=1
Numx(k,j2) = Mk(:,j2) * w(l , k+j2-1)
j2=j2+1
Denx(k,j2) = Mk(:,j2) * w(l , k+j2-1)
j2=4
No
Yes
wt= [w3 , w2 , w , 1]T
Mw=[numy(k,1)*wt / deny(k,1)*wt , numy(k,2)*wt / deny(k,2)*wt , numy(k,1)*wt / deny(k,1)*wt , numy(k,1)*wt / deny(k,1)*wt]
Cx(i,j) = Mu * Px(l:l+3, k:k+3) * MwT
Cy(i,j) = Mu * Py(l:l+3, k:k+3) * MwT
Cz(i,j) = Mu * Pz(l:l+3, k:k+3) * MwT
No
j=j+1 , Δw=w+Δw
j=17/Δw + 1
No
Yes
i=i+1 , Δu=u+Δu
i=17/Δu + 1
Yes
2

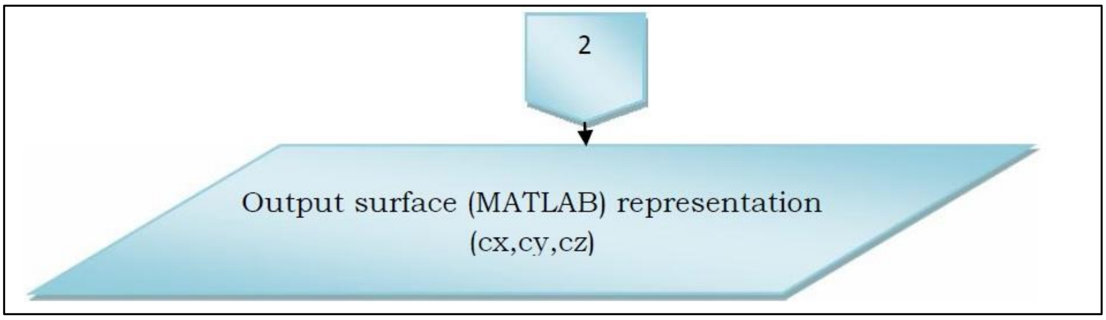


Figure 1: The proposed algorithm.

4. The Design Case Studies

We utilize the proposed algorithm for the designing of three automotive models

- First case study is the car door, and the control points mesh is shown in Figure 2, the designed surface is shown in Figure 3.

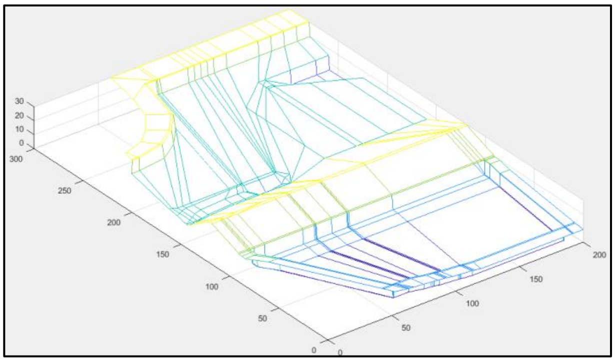


Figure 2: Control point mesh

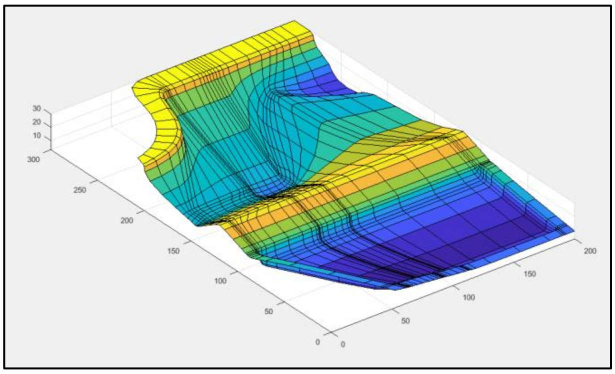


Figure 3: the designed surface of car exterior door (MATLAB).

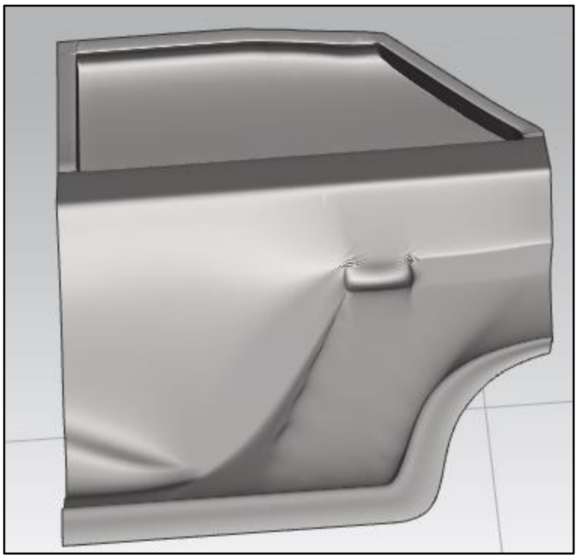


Figure 4: true shading of the designed surface (NX)

- The second case study is the design of the car hood. Half of the surface is designed using the NURBS surface, and the other half is found by the mirror function of the first half.

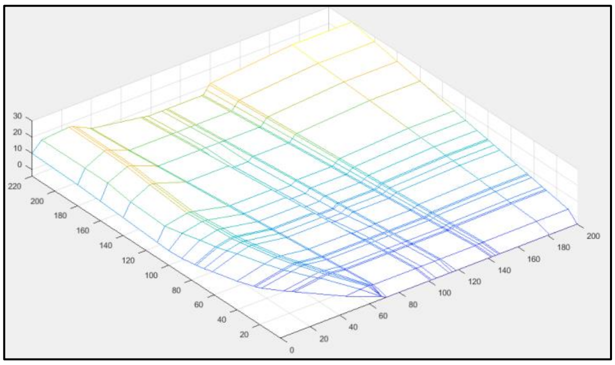


Figure 5: the control point mesh of car hood design

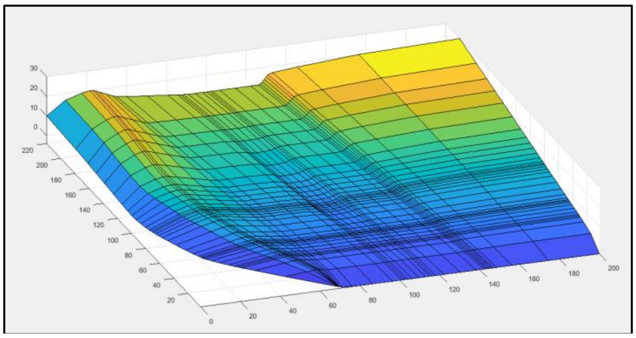


Figure 6: the designed surface of car hood design (half)

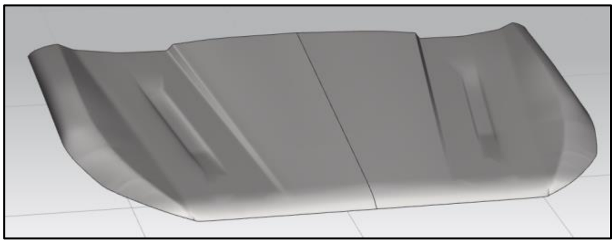


Figure 7: true shading of the car hood surface (NX).

- The third case study is the back door of the car. Same with case 2, half of the surface is designed using the NURBS surface, and the other half is found by the mirror function of the first half.

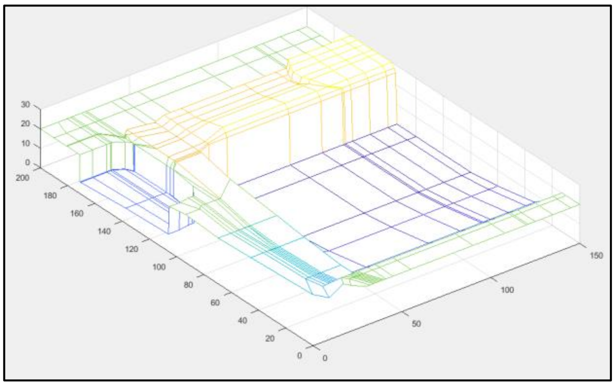


Figure 8: The mesh of control points of backdoor design

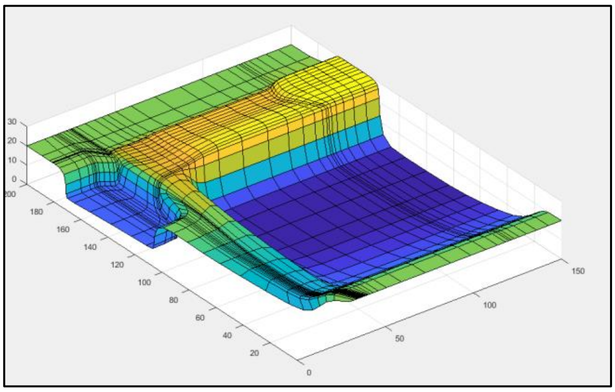


Figure 9: the designed surface of back car door design (half)


F

Figure 10: True shading of the back car door design (NX)

5. Discussion

- The proposed compound NURBS surface modeling proved to be useful in the designing of various sculptured surfaces without any need for joining the particular patches.
- The proposed NURBS surface design requires a careful selection of the control points; any distortion of the mesh strongly affects the shape of the surface.
- The straight portions of the surface can be modeled with 4-points occurring in the same line direction, which helps the designer to model the standard primitives that combined with free form entities.
- Unfortunately, the MATLAB programming does not store surface data transfer type to be processed in other CAD/CAM software. So, the data transferring method is critical for the preservation of surface design.

6. Conclusions

- Standard primitives and free form surfaces do not achieve smooth intersection or connections, which cause failure and weak locations in the design, but with the proposed method, a smooth C^2 continuity is provided between the surface patches.
- Design alterations in the designed surface algorithm are easy to be implemented while the alterations based on other approaches are difficult because any change in the control points of free form surfaces will change the outer profile of the surface, which leads to resolving the intersection problems.
- The weights of the NURBS surface provide more flexibility to the design, which helps the designed in changes in certain regions of the designed surface without modifying the control points of the surface.

References

[1] Byoung K. Choi and Robert B. Jerard, "Sculptured surface machining, theory and applications," Springer, 1998.

[2] D. Veeramani and Y. S. Gau, "Issues in patch-by-patch machining of compound sculptured surfaces," IIE Transactions, Vol. 30, pp. 341-355, 1998.

[3] K. Michálková and B. Bastl, "Imposing angle boundary conditions on B-spline/NURBS surfaces," Computer-Aided Design, vol. 62, pp. 1-9, 2015.

[4] S.M. Hu, Y.H. Li, T. Ju and X. Zhu, "Modifying the shape of NURBS surfaces with geometric constraints," Computer-Aided Design, vol. 33, pp. 903-912, 2001.

[5] W. K. Hamdan and M. M. Abdul-Razaq, "Constructing of multi-patches b-spline surfaces," Eng. & Tech. Journal, vol. 33, part A, No.4, 2015.

[6] F. Sun, A.Y.C. Nee and K.S. Lee, "Modifying free-formed NURBS curves and surfaces for offsetting without local self-intersection," Computer-Aided Design, vol. 36, pp. 1161-1169, 2004.

[7] Z. C. Chen and Q. Fu, "A practical approach to generating steepest ascent tool-paths for three-axis finish milling of compound NURBS surfaces," Computer-Aided Design, Vol. 39, pp. 964–974, 2007.

[8] Y. J. Yang, W. Zeng and T. Qi. Song, "Optimizing conformity of NURBS surfaces by general bilinear transformations," Computer-Aided Design, Vol. 63, pp. 12–25, 2015.

[9] M. E. Mortenson, "Geometric modeling," 2nd edition, Wiley, 1997.

[10] D. Salomon, "Curves and surfaces for computer graphics," Springer, 2006.