Journal of
Mathematics, Statistics and Allied Fields
Volume 2, Issue 1, 2008

Direct extending of NURBS surfaces

David Prochazka, Assistant Professor, Faculty of Business and Economics, Mendel University of Agriculture and Forestry in Brno, prochazka@pef.mendelu.cz
Jana Prochazkova, Assistant Professor, Faculty of Mechanical Engineering, Brno University of Technology, jana.prochazkova@ieee.org

Abstract

Merging NURBS surfaces together and extending existing ones are common problems in CAD systems. Extending existing surface is usually done by creating a new separate surface and applying some merging function – result is one continuous surface. In this paper, full description of a method of direct extending existing surfaces is introduced. The extension of the surface is recomputed during its construction, and the NURBS surfaces are immediately C^1 connected.

1 Introduction

NURBS (Non-Uniform Rational B-splines) are frequently used for construction purposes – bodies of cars, aircrafts, etc. – because of their large construction possibilities and stable, quick and well known computing algorithms. Models are usually composed as a set of surfaces which are merged together in one at the end of the work. In this paper, a method which allows continuous extension of a single surface, is proposed. New part of the surface is in-time merged with the existing hull. Among the advantages of this approach belong the negligible change of input surfaces as well as straightforward design of complex objects.

1.1 Related work

The theoretical base of NURBS is outlined in books (4) and (5). The most used principle of computing NURBS surfaces is based on properties of B-spline functions – (2), (3).

Surface merging is commonly used during construction processes. Eck and Hoppe in (1) use surface merging for automatic reconstruction of B-spline surfaces of arbitrary topological type. Source (8) presents a fast method of collecting fitted surfaces to obtain C^0 continuous model of a given surface. New possibility of merging surfaces is the use of T-splines, which allow merging of arbitrary surfaces with irregular net of control points. The conception of T-spline and a method of merging is described in (6) and (7).

2 NURBS surface

Arbitrary NURBS surface is defined by a regular net of control points with their weights, two knot vectors and two degrees. A correct definition can be written as:

Definition 1. Let P_{ij} be the control points of a $(m+1)\times(n+1)$ net $(i=0,\dots,m, j=0,\dots,n)$ with $(m+1)\times(n+1)$ positive real numbers w_{ij} called weights. Let us denote p the row degree and q column degree of the surface. Further let **u** $=(u_0,u_1,\dots u_{m+p+1})$ be a row knot vector and **v** $=(v_0,v_1,\dots v_{n+q+1})$ a column knot vector. Then the **NURBS surface** is defined by:

$$S\left(u,v\right)=\frac{\sum_{i=0}^{m}\sum_{j=0}^{n}w_{ij}P_{ij}N_{j}^{p}\left(u\right)N_{i}^{q}\left(v\right)}{\sum_{i=0}^{m}\sum_{j=0}^{n}N_{j}^{p}\left(u\right)N_{i}^{q}\left(v\right)},\tag{1}$$

where $(u,v)\in\langle u_0,u_{p+n+1})\times\langle v_0,v_{m+q+1})$ and N_i^p,N_j^q are the B-spline functions (defined in (4)). If

$$R_{i}^{j}\left(u,v\right)=\frac{N_{j}^{p}\left(u\right)N_{i}^{q}\left(v\right)w_{ij}}{\sum_{k=0}^{n}\sum_{l=0}^{m}N_{k}^{p}\left(u\right)N_{l}^{q}\left(v\right)w_{kl}},\tag{2}$$

then Eq. (1) can be written as:

$$S\left(u,v\right)=\sum_{i=0}^{m}\sum_{j=0}^{n}R_{i}^{j}\left(u,v\right)P_{ij}.\tag{3}$$

3 C^1 extension of NURBS surfaces

The first step of surface extension is formulation of presumptions for its continuous connection with a new part. In this section, a theorem which determines all sufficient and necessary conditions for such extension is formulated and proved.

Theorem 3.1. *Let S,S' be two arbitrary NURBS surfaces.*
***Surface* S :**
net $(m+1)\times(n+1)$ of control points P_{ij} with weights w_{ij}, $i=0,1,\dots,m,\ \ j=0,1,\dots,n,$
row and column knot vectors $\mathbf{u},\mathbf{v}$,
row and column degrees p,q,

***Surface* S' :**
net $(l+1)\times(r+1)$ of control points Q_{ij} with weights w'_{ij}, $\ i=0,1,\dots,l,\ \ j=0,1,\dots,r,$
row and column knot vectors $\mathbf{u}',\mathbf{v}'$,
row and column degrees p',q'.

NURBS surfaces S and S' are C^1 continuously merged if:

1. $\mathbf{v}'=\mathbf{v}$
2. $q'=q$
3. $w_{i,n-1}=w_{i,n}=w'_{i,0}=w'_{i,1}$ *for* $i=0,1,\dots,m$
4. $Q_{i,0}=P_{i,n}$ *for* $i=0,1,\dots,m$
5. *for any $k\neq 0,k\in\mathbb{R}$*

$$P_{i,n-1}Q_{i,1}=k.P_{i,n-1}Q_{i,0}=k.P_{i,n-1}P_{i,n}\quad \textit{for}\ i=0,1,\dots,m$$

Proof:
First, the conditions in Theorem 3.1 will be explained. Condition 1 and 2 say that the column knot vectors and column degrees of both surfaces are the same. Condition 3 says that corresponding points in last two columns of S and first column of surface S' have same weights.
The condition 4

$$Q_{i,0}=P_{i,n},\ \ \text{for}\ \ i=0,1,\dots,m\tag{4}$$

means that control points of the last column of the surface S are equal to the control points of the first column which belongs to the surface S'.

The condition 5

$$P_{i,n-1}Q_{i,1} = k.P_{i,n-1}Q_{i,0} = k.P_{i,n-1}P_{i,n} \text{ for any } k\ \neq 0, k\ \in \mathbb{R}. \tag{5}$$

indicates that corresponding points $P_{i,n-1}$, $P_{i,n}$ and $Q_{i,0}$ lie on the line. We can rewrite this expression as:

$$Q_{i,1} = (1-k)P_{i,n-1} + kQ_{i,0}. \tag{6}$$

Now it will be proven that in case that the conditions from Theorem 3.1 are fulfilled, NURBS surfaces are C^1 continuously connected. The gist of the proof is that in case that two surfaces are continuously connected, they have common tangent plane in every point on boundary curve. This tangent plane is defined by partial derivatives in directions u and v. We must prove that these partial derivatives are linearly dependent for both surfaces and common tangent plane thus exists here.

Obviously, if C^1 continuous connection is required, there has to exist the same partial derivatives for the points on the boundary curve. The boundary curve is shaped by the control points $Q_{i,0} = P_{i,n}$ with weights $w_{i,n} = w'_{i,0}$(for $i = 0, 1, \ldots, m$), knot vector $\mathbf{v} = \mathbf{v}'$ and degree $q = q'$. The partial derivatives in the points on the boundary curve (in direction of columns) are apparently the same.

We prove that for an arbitrary point on the boundary curve, row partial derivatives (direction u) are linearly depend. This indicates that common tangent plane exists for both surfaces on the boundary curve. So the surfaces are C^1 merged (Fig. 1).

Now, the well-known construction of NURBS surface is used. The computation is done by twice application of de Boor's algorithm (see (4)). First, de Boor's algorithm is applied to every column with the same column knot vector, column degree and parameter v. These computed points are the input to the second run of de Boor's algorithm with row knot vector and row degree and parameter u. The output is a point on NURBS surface. The points on the NURBS curve from the first cycle of de Boor's algorithm are used to compute partial derivatives in direction u.

Note the computation of derivatives of boundary points on NURBS curve. Let us suppose that the knot vector lies in interval $\langle 0, 1\rangle$. Derivatives of the NURBS curve of degree q with m control points P_i, weights w_i and knot vector $\mathbf{v}$ are of the form:

$$C'(0) = \frac{m}{1 - v_{p-1}}(P_1 - P_0)\frac{w_1}{w_0} \tag{7}$$

$$C'(1) = \frac{m}{m - v_{m-p-1}}(P_n - P_{n-1})\frac{w_{n-1}}{w_n} \tag{8}$$

Obviously, derivatives depend only on points P_0, P_1 and P_{n-1}, P_n.

Thus, we need the derivatives for row points which are the output from the first cycle of de Boor's algorithm. We express the form of the corresponding arbitrary points in the last column of surface S ($C^S_{n-1}(v)$), boundary curve ($C^S_n(v)$) and the first column on surface S' ($C^{S'}_1(v)$) after the first application of de Boor's algorithm. We obtain:

$$C^S_{n-1}(v) = \sum_{i=0}^{m} w_{i,n-1}P_{i,n-1}N^q_i(v) \tag{9}$$

$$C^S_n(v) = \sum_{i=0}^{m} w_{i,n}P_{i,n}N^q_i(v) = \sum_{i=0}^{m} w_{i,n}Q_{i,0}N^q_i(v) \tag{10}$$

$$C_1^{S'}(v) = \sum_{i=0}^{m} w'_{i,1} Q_{i,1} N_i^q(v) \tag{11}$$

With Eq. (6) we can rewrite Eq. (11) as:

$$C_1^{S'}(v) = \sum_{i=0}^{m} w_{i,n}((1-k)P_{i,n-1} + kQ_{i,0}) N_i^q(v) \tag{12}$$

We have to prove that points $C_{n-1}^S(v), C_n^S(v), C_1^{S'}(v)$ for any $v \in \mathbf{v}$ lie on a line. In the case this condition holds, vectors of partial derivatives (generally done by Eq. (7), (8)) are linearly depend, common derivatives exist, and the surfaces are C^1 continuously merged.

If points $C_{n-1}^S(v), C_n^S(v), C_1^{S'}(v)$ for any $v \in \mathbf{v}$ lie on a line then for $c \neq 0, c \in \mathbb{R}$ is:

$$c.\overrightarrow{C_{n-1}^S(v) C_n^S(v)} = \overrightarrow{C_n^S(v) C_1^{S'}(v)}. \tag{13}$$

We can rewrite Eq.(13) as:

$$c(C_n^S(v) - C_{n-1}^S(v)) = C_1^{S'}(v) - C_n^S(v). \tag{14}$$

Now, the left and the right side of Eq.(14) are expressed using Eq.(9), (10), (11) and (12) :

$$C_n^S(v) - C_{n-1}^S(v) = \sum_{i=0}^{m} w_{i,n} Q_{i,0} N_i^q(v) - \sum_{i=0}^{m} w_{i,n} P_{i,n-1} N_i^q(v) =$$

$$\sum_{i=0}^{m} w_{i,n}(Q_{i,0} - P_{i,n-1}) N_i^q(v) \tag{15}$$

and

$$C_1^{S'}(v) - C_n^S(v) = \sum_{i=0}^{m} w_{i,n}(P_{i,n-1}(1-k) + kQ_{i,0}) N_i^q(v) - \sum_{i=0}^{m} w_{i,n} Q_{i,0} N_i^q(v)) =$$

$$= (k-1) \sum_{i=0}^{m} w_{i,n}(Q_{i,0} - P_{i,n-1}) N_i^q(v) \tag{16}$$

Expressions (15), (16) are the same except for multiple $k-1$ (real number). Thus, the vectors in Eq. (13) are linearly depend except a multiples and they determine partial derivatives in direction u. Both partial derivatives are linear depend, on the boundary curve common tangent plane exists and this proves the Theorem 3.1. On the Fig. 1 there is an example of a simply surface extension.

4 Implementation

Method of direct extension of NURBS surfaces will be a part of the future release of engineering CAD system RFEM 3D. Arbitrary NURBS surface is defined by four close boundary NURBS curves. The boundary curve of the existing surface and other curves, which define the new part, are entered for the new surface.

There exist two possibilities of such direct extension. First approach is that the program will change the position of the control points in the first column of the new surface to fulfill the condition in Theorem 3.1. This method could cause notable changes in the shape of the surface. For this reason, we based our method on addition of new column of control points near the boundary curve. This approach results from the proof of Theorem 3.1. The basic idea is in Eq.(13). For suitable c, new points on the connected surface are computed.

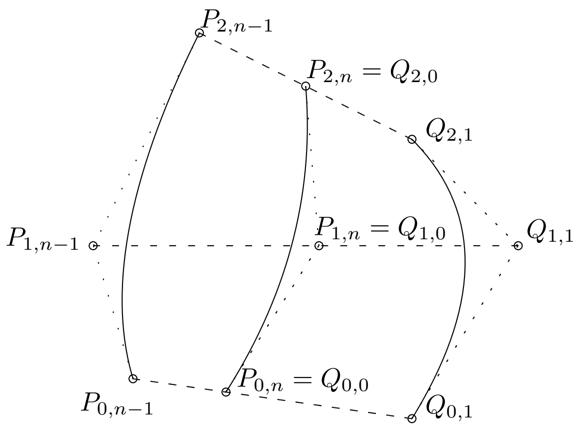

Figure 1: Scheme of continuous extension of NURBS surfaces

Let us suppose that we need to connect two surfaces along a column boundary curve. The algorithm for one row will be shown. For all others, same principle is used. We have last two point A, B on the first surface, thus we are searching for point C on the adjoint surface which satisfies:

$$(1+c)\overrightarrow{AB} = \overrightarrow{AC} \rightarrow C = (1+c)B - cA \tag{17}$$

It is necessary to choose a good value for constant c. It depends on the size of the surface and the distance between control points. With Eq.(17) we generate new column of control points and construct the resultant surface with them.

4.1 Direct extension algorithm

Let us have two NURBS surfaces defined in Theorem 3.1. The algorithm for recomputing the extended surface can be written as:

1. We release first column of surface S'.

```
for (j=1; j< numberOfColumn S'; j++)
  for (i=0; i< numberOfRows S'; i++)
    Q_i,j+1 = Q_i,j;
```

2. Initial value of constant c is chosen.

3. Computation of adding column of surface S' to be added.

```
for (i=0; i<numberOfRows S; i++)
  Q_i,1=(1+c)P_i,n-1 - cP_i,n-1;
```

4. Checking phase:

```
s1 = computeSizeOfVector(Q_i,0 Q_i,1);
s2 = computeSizeOfVector(Q_i,1 Q_i,2);
```

```
if  (s2 < 1/4*s1) stop;
else
  repeat{
    c = c/2;
    Step 3;}
  until  s2 < 1/4*s1;
```

We tested our algorithm on arbitrary surfaces. We chose the constant $c = 0.25$. Surfaces without and with C^1 connection done by our algorithm are on Fig. 2. On Fig. 3 is the connection line in detail. On Fig. 4, 5 is a connection between a cone and a cylinder.

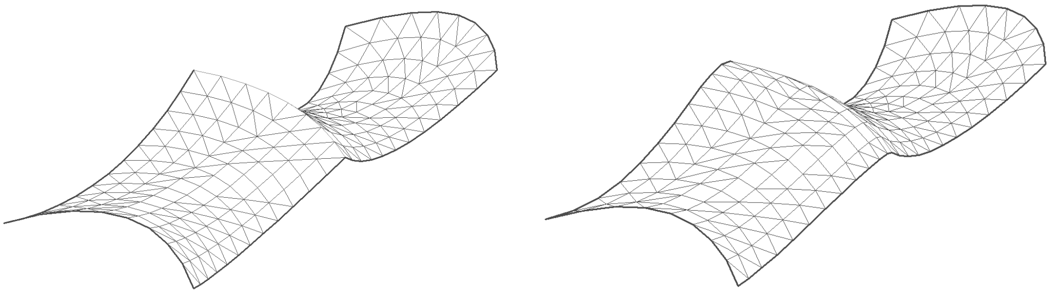

Figure 2: Connection of arbitrary surfaces without and with application of our algorithm

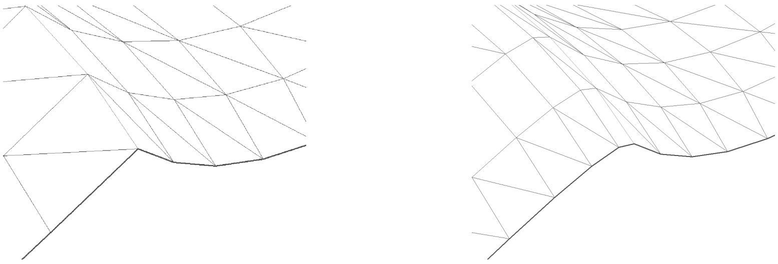

Figure 3: Detail of the connection (C^0, C^1 continuity)

5 Conclusion

This article deals with the sufficient and necessary conditions for C^1 direct extension of NURBS surfaces. In literature, these conditions are not clearly defined, therefore we outlined them here in detail and proved. Application of these conditions is shown in the second part of this article – in direct extension algorithm. As it was presented, the main advantage of this approach is that no auxiliary surfaces nor subsequent merging is needed. All extensions are done directly. This feature simplifies its implementation in different CAD systems.

This method is suitable for surfaces with approximately same proportions. For connection of very different surfaces, T-spline approach, which is based on irregular net of control points, is preferred (see (6), (7)).

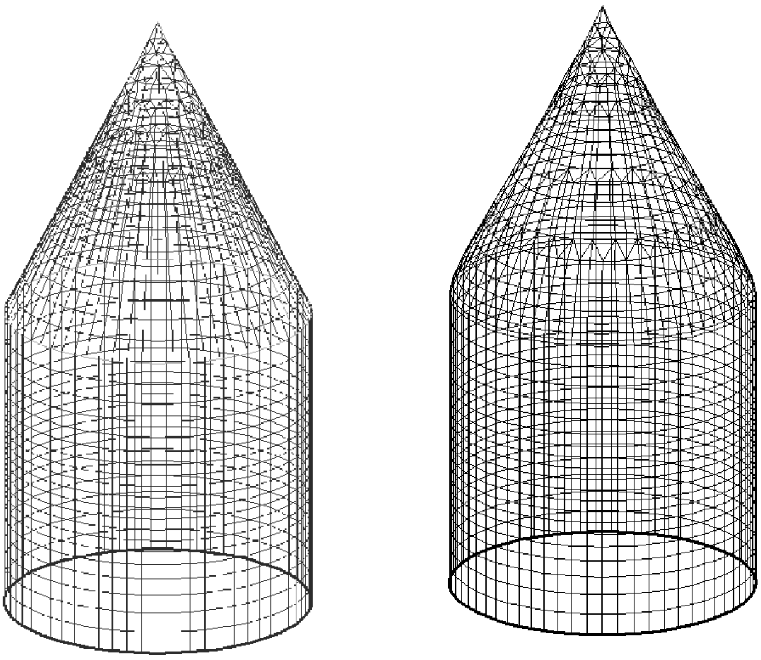

Figure 4: Cone and cylinder C_0 merged, cone extended by a cylinder using our algorithm (C_1 continuous)

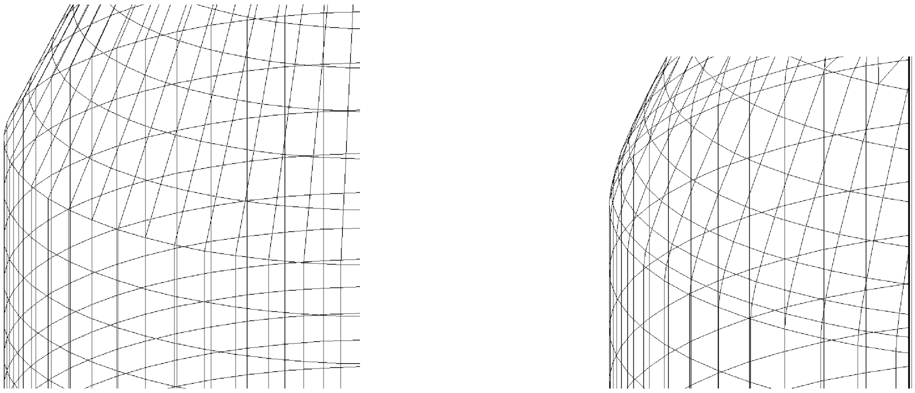

Figure 5: Merged and extended surfaces – detail

References

[1] Eck, M. and Hoppe, H. Automatic Reconstruction of B-spline Surfaces of Arbitrary Topology. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, (New Orleans, LA, USA), ACM, New York, 1996, pp. 325–334.

[2] Farin, G. *Curves and Surfaces for Computer-Aided Geometric Design: A Practical Guide*, Academic Press, San Diego, USA, 1990.

[3] Hoffmann C.M. *Geometric and solid modeling: An introduction*. Morgan Kaufmann, San Mateo, California, USA, 1989.

[4] Piegl L. and Tiller, W. *NURBS Book*. Springer-Verlag Berlin, Germany, 1997.

[5] Rogers, F.D. *An Introduction to NURBS*. Morgan Kaufman Publisher, 2000.

[6] Sederberg, T. W. Zheng, J. Bakenov, A. and Nasri, A., T-splines and T–NURCCS. *ACM Transactions on Graphics* 22, 3 (2003), pp. 477–484.

[7] Sederberg, T.W. Zheng, J. Cardon, D.L. and Lyche, T. T-splines Simplification and Local Refinement. *ACM Trans. Graphic* 23, 3 (2004), pp. 276–283.

[8] Yang, J. Speidel, U. Zanaty, E.A. and Girgis M.R. Collect the Fitted Surfaces into Complex Based on C^0 Continuity. *J.UCS*, 11, 6 (2005), pp. 1102 – 1115.