Original scientific paper
Accepted 21. 12. 2011.

MÁRTA SZILVÁSI-NAGY*
SZILVIA BÉLA**

B-spline Patches Fitting on Surfaces and Triangular Meshes

B-spline Patches Fitting on Surfaces and Triangular Meshes

ABSTRACT

In this paper a technique for the construction of quartic polynomial B-spline patches fitting on analytical surfaces and triangle meshes is presented. The input data are curvature values and principal directions at a given surface point which can be computed directly, if the surface is represented by a vector function.

In the case of discrete surface representation, i.e. on a triangle mesh the required input data are computed from a circular neighborhood of a specified triangle facet. Such a surface patch may replace a well defined region of the mesh, and can be used e.g. in re-triangulation, mesh-simplification and rendering algorithms.

Key words: B-spline surface, local surface approximation, principal curvatures, triangle mesh

MSC 2010: 65D17, 65D05, 65D07, 68U05, 68U07

B-splajn dijelovi koji pristaju na plohe i triangularne mreže

SAŽETAK

U ovom se radu prikazuje metoda za konstrukciju kvartnog polinoma B-splajn dijela podesnog za analitičke plohe i mreže trokuta. Ulazni podaci su vrijednosti zakrivljenosti i glavni smjerovi u danoj točki plohe, koji se mogu izravno računati za plohu zadanu vektorskom funkcijom.

Za slučj diskretne reprezentacije plohe, tj. za triangularnu mrežu, odgovarajući ulazni podaci računaju se iz kružne okoline određenog trokuta mreže. Takvi dijelovi mogu zamijeniti dobro definirano područje mreže, i mogu se upotrijebiti npr. u retriangulaciji, simplifikaciji mreže i renderiranju.

Ključne riječi: B-splajn ploha, localna aproksimacija plohe, glavne zakrivljenosti, mreža trokuta

1 Introduction

Surface patches matching free form surfaces, triangular meshes or point-based surfaces are widely used in computer graphics and in many applications. Different types of patches have been developed to reconstruct the surface geometry.

A technique in [16] which generates a hole-free, piecewise linear approximation to point-based surfaces uses circular and elliptical planar surface segments, so-called splats, for surface reconstruction and high-quality rendering. A circular splat is given by its center, its normal vector and its radius. Elliptical splats need two additional vectors to define the major and minor axes replacing the radius, they can adapt to the local curvature of the surface. Large number of linear splats are needed to represent the shape of most smooth models.

Quadrics are defined in [4] for mesh simplification algorithms which produce an approximation composed of fewer triangles that preserves surface shape. The quadrics characterize the local shape of the surface, they are elongated in directions of low curvature and thin in directions of high curvature. Minimization of a quadric error metric generates a triangulation with optimal triangle shape. In [2] quadratic and cubic splats are computed using a moving least squares procedure. They provide good quality and high rendering speed using fewer primitives than linear splats. In [5] a rendering primitive, called differential point is introduced with embedded curvature information in the vicinity of the actual point. The method leads to a more sparsely surface representation, to accelerated shading, to a point-based simplification technique, and to a better quality of rendering than a pure splat-based approach. The inputs are NURBS surface or polygonal mesh. A differential point is constructed from a sample point and principal curvatures and principal directions. Practically, a local surface is defined implicitly in the neighborhood of the point by a

* Supported by a joint project between the TU Berlin and the BUTE.
** Supported by the grant TÁMOP - 4.2.2.B-10/1–2010-0009.

set of osculating circle arcs in normal planes passing through given tangent vectors such that the distance of the circle arc from the surface or mesh is less than a given tolerance. The necessary curvature values for a mesh are estimated by the method of Taubin ([15]).

A method for fitting NURBS surfaces for cloud-of-points data representing rotational surfaces is shown in [1]. First, a scalar valued B-spline function is fitted to the data, then it is converted to a parametric NURBS. Three dimensional object matching is the tool for fitting NURBS surfaces to point based surfaces or to an other NURBS in [7]. Two intrinsic surface properties, the Gaussian and the mean curvatures are used for matching, and an optimal rigid body transformation is developed. A C^2-continuous spline surface is constructed to triangular meshes in [6]. The construction is made in two phases. First, a so called guide surface is constructed from vertices and boundary data, then it is modified such that the final surface has a good shape also in the case of triangulation with isolated extraordinary vertices.

Trigonometric surface patches are constructed from curvature data matching a neighborhood of a face of a triangular mesh in [14] and a neighborhood on an analytical surface [13]. The principal directions and curvature values of the meshed surface are estimated by the method developed in [11] and [12]. This is a face based method, different from the vertex based algorithms used in the most papers in the large literature dealing with discrete differential geometry ([8], [9], [10]).

In this paper the construction of a uniform polynomial B-spline surface patch of 4×4 degree from given curvature values is presented. These curvatures are the principal curvatures of a base surface at a given point. They determine its osculating circles in the two normal planes through the principal directions. The construction is made in two phases. First, the surface interpolates the two circle arcs by its middle parameter curves, and approximates four additional surface points at its corner points. Then the approximation is improved by correcting some boundary data while minimizing an error between the B-spline patch and the given base surface.

In Section 2 the approximation of a circular arc by a fourth degree B-spline curve is analyzed. In Section 3 the computation of a B-spline surface patch of 4×4 degree is presented from input data, which allow to fit the patch on a base surface. In Section 4 the generation of the input data from a base surface is shown in both, analytical and discrete representations. Examples are shown in Section 5.

2 Approximation of a circular arc by a fourth degree B-spline curve

The uniform polynomial B-spline curve is represented by the vector function

$$\mathbf{g}(t) = \left[t^4\ t^3\ t^2\ t\ 1\right]\mathbf{M}^4\left[\mathbf{p}_0\ \mathbf{p}_1\ \mathbf{p}_2\ \mathbf{p}_3\ \mathbf{p}_4\right]^T,\ 0\le t\le 1, \quad (1)$$

where the coefficient matrix is

$$\mathbf{M}^4 = \frac{1}{24}\begin{bmatrix} 1 & -4 & 6 & -4 & 1\\ -4 & 12 & -12 & 4 & 0\\ 6 & -6 & -6 & 6 & 0\\ -4 & -12 & 12 & 4 & 0\\ 1 & 11 & 11 & 1 & 0\end{bmatrix}. \quad (2)$$

Let the circular arc of radius ρ and central angle 2α be given in the xz coordinate plane parametrized as follows

$$\mathbf{c}(t) = \mathbf{i}\rho\sin(\alpha(2t-1)) + \mathbf{k}\rho\cos(\alpha(2t-1)),\ t\in[0,1]. \quad (3)$$

Three points $\mathbf{c}(0)$, $\mathbf{c}(0.5)$, $\mathbf{c}(1)$ and two tangent vectors at the end points $\dot{\mathbf{c}}(0)$ and $\dot{\mathbf{c}}(1)$ will be interpolated by solving the system of linear equations

$$\mathbf{c}(0)=\mathbf{g}(0),\ \mathbf{c}(0.5)=\mathbf{g}(0.5),\ \mathbf{c}(1)=\mathbf{g}(1),$$
$$\dot{\mathbf{c}}(0)=\dot{\mathbf{g}}(0),\ \dot{\mathbf{c}}(1)=\dot{\mathbf{g}}(1)$$

for the unknown control points $\mathbf{p}_i$, $i=0\ldots 4$, where $\dot{\mathbf{c}}$ and $\dot{\mathbf{g}}$ denote the derivatives according to the parameter t. A unique (symbolical) solution exists, and the B-spline curve with the computed control points approximates the given circular arc with an error

$$\int_0^1(\mathbf{c}(t)-\mathbf{g}(t))^2dt < 10^{-18},\ 2\alpha\le\frac{\pi}{3}. \quad (4)$$

In the examples the relative error with respect to the arc length is even smaller. The limit for the central angle is a usual limit also in classical approximations.

3 Computation of a B-spline surface patch of 4×4 degree from given geometric data (symbolical solution)

The polynomial B-spline surface patch of 4×4 degree with uniform knot vector is described by the vector function

$$\mathbf{r}(u,v) = \left[u^4\ u^3\ u^2\ u\ 1\right]\cdot\mathbf{M}\cdot\mathbf{B}\cdot\mathbf{M}^T\cdot\left[v^4\ v^3\ v^2\ v\ 1\right]^T, \quad (5)$$
$$(u,v)\in[0,1]\times[0,1].$$

The geometric data are the points of the control net:
$\mathbf{B} = [\mathbf{b}[i,j]]$, $\quad i=0,\ldots 4, \quad j=0,\ldots 4.$
The prescribed input data in our surface construction are the data of two circular arcs lying in two orthogonal planes, which will be interpolated by the middle parameter curves.

Four additional boundary data are four corner points of the required patch. For the parameter curve $\mathbf{r}(u,0.5)$ three points of a circular arc
$\mathbf{r}(0,0.5) = M11$, $\mathbf{r}(0.5,0.5) = M$, $\mathbf{r}(1,0.5) = M12$
and two tangent vectors at the starting and end points $M11$ and $M12$,
$\mathbf{r}_u(0,0.5) = T11$, $\mathbf{r}_u(1,0.5) = T12$
respectively, are given. Similarly, the other parameter curve $\mathbf{r}(0.5,v)$ is determined by three points of a circular arc
$\mathbf{r}(0.5,0) = M21$, $(\mathbf{r}(0.5,0.5) = M)$, $\mathbf{r}(0.5,1) = M22$
and by two prescribed tangent vectors at the points $M21$ and $M22$
$\mathbf{r}_v(0.5,0) = T21$, $\mathbf{r}_v(0.5,1) = T22$.

The parametrization of these arcs is the same as that of the curve $\mathbf{c}(t)$ in (3). $\mathbf{r}_u$ and $\mathbf{r}_v$ denote the partial derivatives of the function $\mathbf{r}(u,v)$ according to u and v, respectively. The position vectors of the points Mij are denoted by the same letter, Tij denote the corresponding tangent vectors.
The four corner points of the patch denoted by $P00$, $P10$, $P01$ and $P11$ according to their parameter values are also prescribed (Fig. 1). These are in all 13 input data.

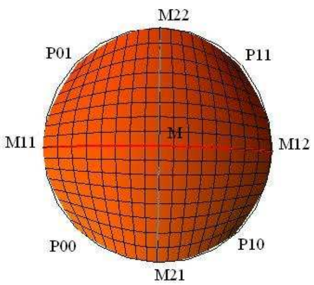


Figure 1: *The middle parameter curves of the surface patch pass through $M11$, M, $M12$ and $M21$, M, $M22$, respectively, the corner points are $P00$, $P10$, $P01$ and $P11$.*

The B-spline patch has 25 unknown control points therefore, 12 additional data are necessary for the computation. These are generated from the previous data in the following way.

$$
\begin{aligned}
\mathbf{r}_u(0,0) &= -\mathbf{r}_v(0,0) = \frac{1}{4}(M21 - M11)\\
\mathbf{r}_u(1,0) &= \mathbf{r}_v(1,0) = \frac{1}{4}(M12 - M21)\\
\mathbf{r}_u(1,1) &= -\mathbf{r}_v(1,1) = \frac{1}{4}(M12 - M22)\\
\mathbf{r}_u(0,1) &= \mathbf{r}_v(0,1) = \frac{1}{4}(M22 - M11)
\end{aligned} \tag{6}
$$

Four twist vectors at the corner points are determined by the change of the first partial derivatives while moving from a corner point into the corresponding midpoint along a boundary curve.

$$
\begin{aligned}
\mathbf{r}_{uv}(0,0) &= (\mathbf{r}_v(0.5,0) - \mathbf{r}_v(0,0) + \mathbf{r}_u(0,0.5) - \mathbf{r}_u(0,0)) \cdot \lambda\\
\mathbf{r}_{uv}(1,0) &= (-\mathbf{r}_v(0.5,0) + \mathbf{r}_v(1,0) + \mathbf{r}_u(1,0.5) - \mathbf{r}_u(1,0)) \cdot \lambda\\
\mathbf{r}_{uv}(1,1) &= (-\mathbf{r}_v(0.5,1) + \mathbf{r}_v(1,1) - \mathbf{r}_u(1,0.5) + \mathbf{r}_u(1,1)) \cdot \lambda\\
\mathbf{r}_{uv}(0,1) &= (\mathbf{r}_v(0.5,1) - \mathbf{r}_v(0,1) - \mathbf{r}_u(0,0.5) + \mathbf{r}_u(0,1)) \cdot \lambda
\end{aligned} \tag{7}
$$

Here the first partial derivatives on the right hand sides are expressed by the prescribed points Mij according to the equations (6). The free parameter λ changes the lengths of the twist vectors which strongly influence the shape of the surface patch. It will be determined by minimizing an error function. Putting all these conditions into a system of equations does not result in a solution for the unknown control points of the required patch however, all the equations are linear ones. Instead of this, the computation is organized according to the following strategy.
The matrix of the control points $\mathbf{B} = [\mathbf{b}[i,j]]$, $i = 0,\dots 4$, $j = 0,\dots 4$ will be partitioned, and the control points will be computed in three phases. Figure 2 shows which control points are computed in one step by marking them with the same symbol.

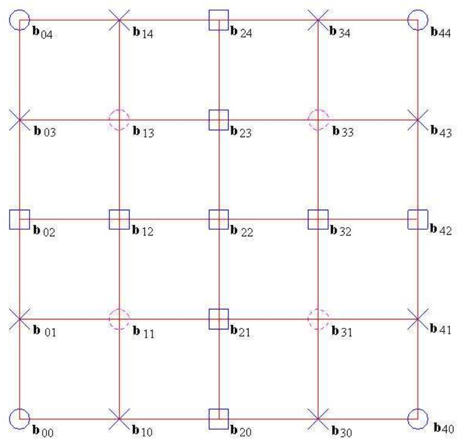


Figure 2: *Partition of the matrix of the control points.*

In the **first step** control points on the boundaries are computed from the data of four boundary curves as the solution of four systems of linear equations by the interpolation method of circular arcs in Section 2. Each boundary curve is determined by three points and two tangent vectors at the end points. Each system of linear equations result in five control points. The data and the corresponding solutions are as follows, while the boundary curves of the patch follow in counter clockwise direction.
For the boundary curve $v = 0$ the data are $P00$, $M21$, $P10$, $\mathbf{r}_u(0,0)$, $\mathbf{r}_u(1,0)$. The solutions are $\mathbf{p}_i^{u0}$, $i = 0,\dots 4$.

For the boundary curve $u = 1$ the data are $P10, M12, P11$, $\mathbf{r}_v(1,0), \mathbf{r}_v(1,1)$. The solutions are $\mathbf{p}_i^{1v}, i = 0,\dots 4$.
For the boundary curve $v = 1$ the data are $P01, M22, P11$, $\mathbf{r}_u(0,1), \mathbf{r}_u(1,1)$. The solutions are $\mathbf{p}_i^{u1}, i = 0,\dots 4$.
For the boundary curve $u = 0$ the data are $P00, M11, P01$, $\mathbf{r}_v(0,0), \mathbf{r}_v(0,1)$. The solutions are $\mathbf{p}_i^{0v}, i = 0,\dots 4$.
Not all these solutions will be placed into the matrix of the control points $\mathbf{B}$.
In the **second step** the system of linear equations from the interpolation conditions for the middle parameter curves is solved.
The data are
$\mathbf{r}(0.5,0) = M21$, $\mathbf{r}(0.5,0.5) = M$, $\mathbf{r}(0.5,1) = M22$,
$\mathbf{r}_v(0.5,0) = T21$, $\mathbf{r}_v(0.5,1) = T22$,
$\mathbf{r}(0,0.5) = M11$, $\mathbf{r}(1,0.5) = M12$,
$\mathbf{r}_u(0,0.5) = T11$, $\mathbf{r}_u(1,0.5) = T12$
The unique symbolical solution of this system are nine control points $\mathbf{b}[i,2], i = 0\dots 4$ and $\mathbf{b}[2, j], j = 0,1,3,4$ expressed with these data and the remaining sixteen control points.
Now, twelve from the sixteen control points will be replaced by the points of the solution in the first step as follows.

$$\mathbf{b}[1,0] = \mathbf{p}_1^{u0},\ \mathbf{b}[3,0] = \mathbf{p}_3^{u0},\ \mathbf{b}[4,1] = \mathbf{p}_1^{1v},\ \mathbf{b}[4,3] = \mathbf{p}_3^{1v}$$
$$\mathbf{b}[1,4] = \mathbf{p}_1^{u1},\ \mathbf{b}[3,4] = \mathbf{p}_3^{u1},\ \mathbf{b}[0,1] = \mathbf{p}_1^{0v},\ \mathbf{b}[0,3] = \mathbf{p}_3^{0v}$$
$$\mathbf{b}[0,0] = \frac{1}{2}(\mathbf{p}_0^{u0} + \mathbf{p}_0^{0v}),\ \mathbf{b}[4,0] = \frac{1}{2}(\mathbf{p}_4^{u0} + \mathbf{p}_0^{1v}),$$
$$\mathbf{b}[4,4] = \frac{1}{2}(\mathbf{p}_4^{u1} + \mathbf{p}_4^{1v}),\ \mathbf{b}[0,4] = \frac{1}{2}(\mathbf{p}_4^{0v} + \mathbf{p}_0^{u1}).$$

We note that the interpolation conditions for the corner points are not satisfied, but "relaxed" by the last four equations. They will be corrected by minimizing an error function in the last step.
In the **third step** the last four control points are computed from the system of four linear equations expressed by the twist vectors at the corner points which are computed earlier from the prescribed data in (7). The solution of the system of the linear equations are the control points
$\mathbf{b}[1,1], \mathbf{b}[3,1], \mathbf{b}[3,3], \mathbf{b}[1,3]$.
Finally, the matrix $\mathbf{B} = [\mathbf{b}[i, j]]\ ,\quad i = 0,\dots 4,\quad j = 0,\dots 4$ is expressed by the prescribed data and the free scalar parameter λ.
In the **fourth step** the free parameter λ is determined from an error function expressing the squared sum of distances between the prescribed and computed corner points.

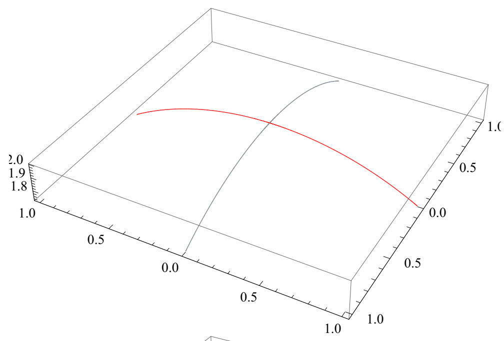
$$\begin{aligned} d(\lambda) =& (\mathbf{r}(0,0) - P00)^2 + (\mathbf{r}(1,0) - P10)^2 \\ &+ (\mathbf{r}(1,1) - P11)^2 + (\mathbf{r}(0,1) - P01)^2 \end{aligned} \tag{8}$$

This function is quadratic in λ, the minimization results in a unique value of it.

In **Example 1** two circular arcs of radius 2 and central angle $2\alpha = \pi/3$ are given in the xz and yz coordinate plane, respectively. The four corner points are rotated end points of these circular arcs around the axis z. The computed B-spline patch interpolates the given arcs within the integrated error (see (4)) of 10^{-8}, the interpolation error at the corner points (see (8)) is within 10^{-16}. We note that the interpolation error of the patch to the data of the given circular arcs is practically zero (less than 10^{-28}), and it is independent from the value of λ. The length of the twist vectors determined by λ effects on the error at the corner points very strongly. The computed value of λ by minimizing the distance function in (8) is 0.89, while the value $\lambda = 0$ results in a very large error of 1.2.

In **Example 2** the data are similar but they determine a hyperbolic surface patch. The analysis shows similar results to that in Example 1.

The Figures 3 and 4 show the middle parameter curves $\mathbf{r}(u,0.5)$ and $\mathbf{r}(0.5,v)$ of the resulting patches on the left hand side, and the patches with the corresponding control points on the right. The whole control net cannot be shown clearly, therefore they are omitted in the figures.



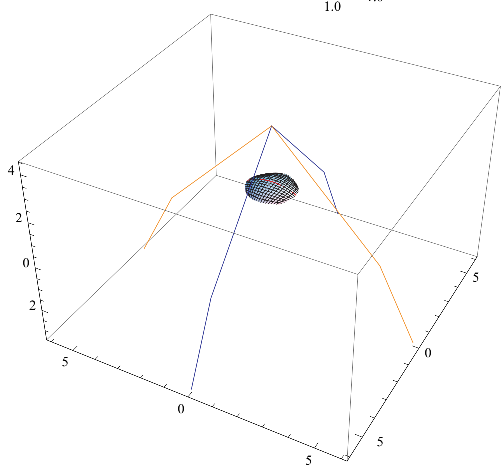


Figure 3: *Example 1.*

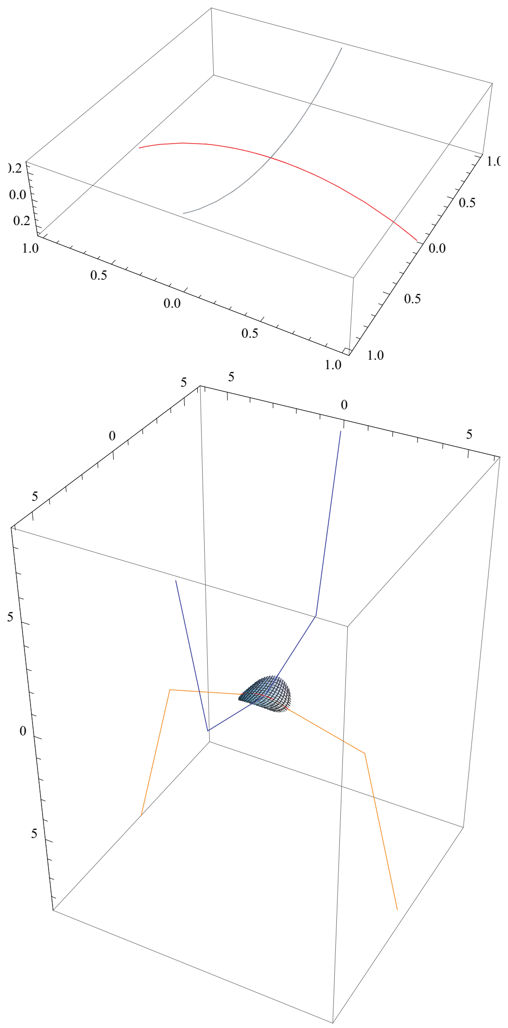


Figure 4: *Example 2*.

4 Solution of the fitting problem; computing the input data of the B-spline patch

A B-spline patch of 4×4 degree fitting on a base surface will be determined by two osculating circle arcs of the base surface lying in two normal planes through the principal directions at the actual surface point, and four additional surface points in a neighborhood of this point. The neighborhood is determined by the arc length of the osculating circle arcs given by the user. Then by measuring the given arc length on the surface from the point in normal planes rotating around the surface normal a circular neighborhood is constructed. In the case of a triangular mesh the neighborhood is constructed around the barycentric center of a specified face in the mesh.

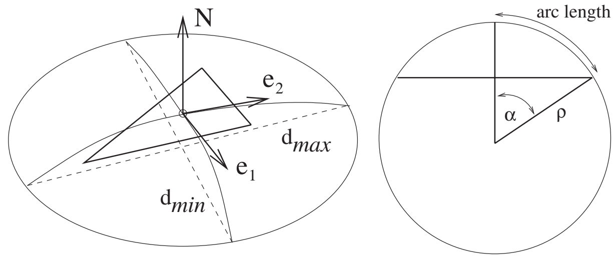


Figure 5: *Circular neighborhood and an osculating circle in a normal plane*.

In Figure 5 such a circular neighborhood is shown with the two osculating circle arcs of the shortest and largest chord lengths denoted by d_{min} and d_{max}, respectively. They determine at a regular surface point the orthogonal principal directions $\mathbf{e}_1$ and $\mathbf{e}_2$, if $d_{min} \neq d_{max}$. The central angle 2α and the radius ρ of an osculating circle arc in a normal plane is computed from the given arc length s and the corresponding chord length d in the following way ([11], [12]).
From $\rho\alpha = s$ and $\rho \sin\alpha = \frac{d}{2}$ and the approximation

$$\sin\alpha \approx \alpha - \frac{\alpha^3}{6}, 0 < \alpha << 1$$

follows that

$$\alpha \approx \sqrt{(1-\frac{d}{2s})6}, \; \rho \approx \frac{s}{\alpha} \text{ if } \alpha \neq 0, \; \kappa_n \approx \frac{\alpha}{s}.$$

The tangent vectors at the end points of the arc are computed with the parametrization in the local coordinate system determined by the vectors $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{N}$ according to the vector equation (3) in Section 2.
In the case of an analytical surface the principal curvatures and principal directions are computed directly from the known equation of the surface ([3]), though the construction of the circular neighborhood (more precisely the points of its boundary) is computed by a discrete method, while measuring the given arc length along a polygonal line which approximates the surface curve lying in an intersecting normal plane.
From the constructed circular neighborhood five interpolation points and four tangent vectors to the circular arcs (see the second step in the computation of the control points in Section 3) and four surface points on the boundary (determining the corner points of the patch) are used as direct input data in the matrix $\mathbf{B}$ of the control points. The control net is expressed symbolically by these input data and a scalar parameter λ. After replacing the actual numerical values into the matrix $\mathbf{B}$, only the computation of λ is necessary in the concrete examples (see the fourth step in Section 3).
Figure 6 shows the data of a B-spline patch matching a circular neighborhood on the base surface. The interpolation

points on the patch boundary Mij and Pij $(i = 0,1,\ j = 0,1)$ are end points of the surface curves in the corresponding normal planes computed as polygonal lines on the surface by measuring the given arc length along them. The tangent vectors Tij_u and Tij_v $(i = 0,1,\ j = 0,1)$ at the corner points are determined as described in the equations (6) in Section 3.

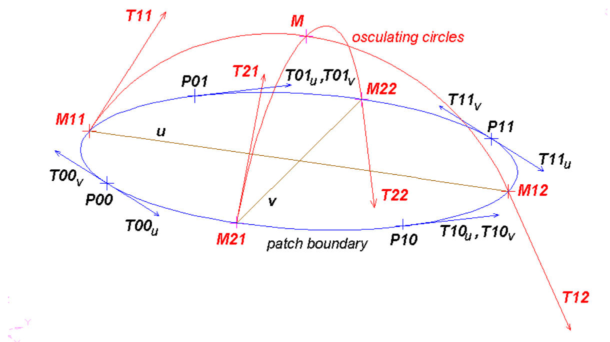


Figure 6: *Input data of a patch computed in a circular neighborhood of a base surface.*

5 Examples

The next two examples are computed with analytical base surfaces. In this case the principal directions are computed exactly from the vector functions representing the surfaces. The number of the computed points on the boundary of the circular patch is 72 in both examples.

In **Example 3** (Fig. 7) the base surface is a cylinder of radius 10, the circular neighborhood around a surface point is constructed with the arc length 5. The generated B-spline surface interpolates the data points of the two middle parameter curves $\mathbf{r}(u,0.5)$ and $\mathbf{r}(0.5,v)$ practically with zero error (less than 10^{-27}). The minimized error of the sum of squared distances between the prescribed and computed corner points of the patch (see in (4)) is within the relative error of 4% with respect to the given arc length measured on the surface around the given point.

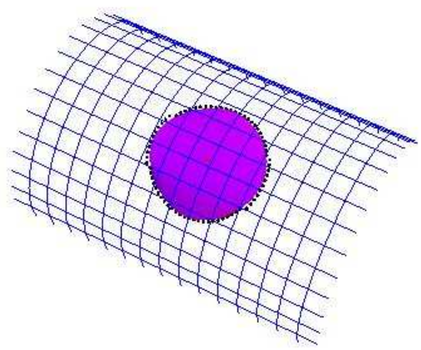


Figure 7: *Example 3.*

The results are similar in **Example 4** (Fig. 8) computed with a torus. The radius of its meridian circle is 10, the "radius" of the circular patch is 5, and the interpolation error at the corner points is here 4% too. Of course, the error is larger with growing neighborhoods to be approximated.

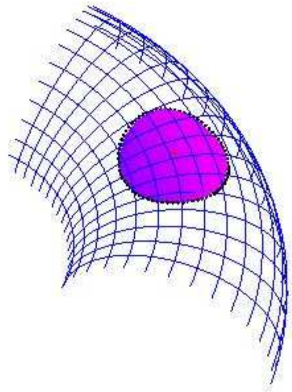


Figure 8: *Example 4.*

The B-spline patches in the next two examples are computed on triangulated surfaces, on so-called "synthetic meshes" generated by a triangulation of the parameter domain of the given surfaces. The principal curvatures and directions are estimated by the method described in Section 4. The circular neighborhood is computed around a chosen triangle face, more exactly around its barycentric center point in 48 normal planes.

In **Example 5** the mesh is a triangulated cylinder of radius 1 (Fig. 9). The circular neighborhood is constructed with the given arc length of 0.6. The estimation error in the computation of the principal curvatures is less than 10^{-3}. The constructed B-spline patch interpolates the data given in the principal normal sections for the middle parameter curves practically with zero error. The error at the corner points is approximately $2 \cdot 10^{-2}$.

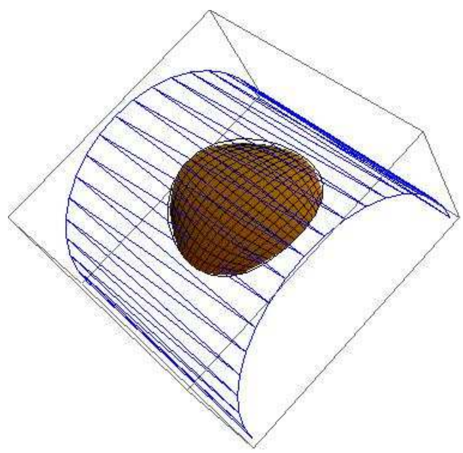


Figure 9: *Example 5.*

In the case of a synthetic mesh of a torus shown in **Example 6**, the results are better due to the dense triangulation. The error at the corner points of the patch is approximately 10^{-2}.

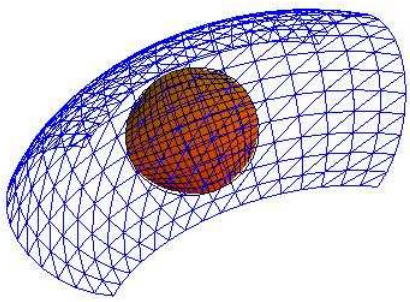


Figure 10: *Example 6*.

We note that the presented construction of a circular neighborhood is working also on a "bad triangulation" shown on the cylinder, where the long, thin triangles have no vertices in the actual neighborhood. This is due to the face based estimation of normal curvatures and to an appropriate polyhedral data structure representing the mesh, which provides effective computation of intersections with planes.

Example 7 shows a "real" mesh of a sphere, i.e. a triangulation generated from measured points on the surface of a sphere offered for test purposes. The computed B-spline patch matches very well a relatively large neighborhood shown in Figure 11.

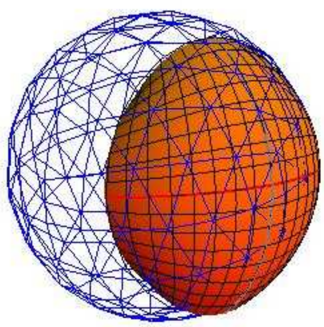


Figure 11: *Example 7*.

6 Conclusions

Construction of a uniform polynomial B-spline surface patch of 4×4 degree from geometric input data has been presented, which are suitable for the solution of a fitting problem, how a given neighborhood of a point on a given surface can be approximated by such a B-spline patch. The well chosen geometric data define the control points of the B-spline patch, and they can be computed on analytic and meshed surfaces by constructing a circular neighborhood around a specified point. The measurement of this neighborhood is a user specified arc length, which is measured on the surface around the surface point in normal sections resulting from a discrete description of the boundary of the neighborhood. In the examples the approximation of specified neighborhoods by the generated B-spline patches has been shown with error estimation. The figures, the symbolic computation of the control points of the B-spline patch and patch fitting with error estimation have been made with the algebraic program package Wolfram Mathematica. The curvature estimation and construction of the circular neighborhoods on triangular meshes have been computed with a program developed by the first author in the program language Java.
Further research is necessary for improving the approximation around the boundaries of the computed surface patches and for estimating the measurement of matched neighborhoods with prescribed error tolerances.

References

[1] S -H. BAE, B. K. CHOI, NURBS surface fitting using orthogonal coordinate transform for rapid product development. *Computer-Aided Design* 34 (2002) 683-690.

[2] N. K. BOLLA, P. J. NARAYANAN, Algebraic splats representation for point based models. *IEEE Sixth Indian Conference on Computer Vision, Graphics & Image Processing* 2008, 71-78.

[3] M. P. DO CARMO, 1976, *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, NJ.

[4] P. S. HECKBERT, M. GARLAND, Optimal triangulation on quadric-based surface simplification. *Computational Geometry* 14 (1999) 49-65.

[5] A. KALAIAH, A. VARSHNEY, Modeling and rendering of points with local geometry. *IEEE Transactions on Visualization and Computer Graphics* 9 (2003) 100-129.

[6] K. KARCIAUSKAS, J. PETERS, Assembling curvature continuous surfaces from triangular patches. *Computers & Graphics* 33 (2009) 204=210.

[7] K. H. KO, T. MAEKAWA, N. M. PATRIKALAKIS, An algorithm for optimal free-form object matching. *Computer-Aided Design* 35 (2003) 913-923.

[8] M. MEYER, M. DESBRUN, P. SCHRÖDER, A. H. BARR, Discrete differential-geometry operators for triangulated 2-manifolds, in *Visualization and Mathematics III*.(H.C. Hege and K. Polthier, Ed.) pp. 35-57, Springer Verlag 2003.

[9] K. POLTHIER, M. SCHMIES, Straightest geodesics on polyhedral surfaces, in *Mathematical visualization* (K. Polthier and H.C. Hege Ed.) pp.391-409, Springer Verlag 1998.

[10] S. RUSINKIEWICZ, Estimating curvatures and their derivatives on triangle meshes, *Proc. 3DPVT'04* 2004, 486-493.

[11] M. SZILVASI-NAGY, About curvatures on triangle meshes. *KoG* 10 (2006) 13-18.

[12] M. SZILVASI-NAGY, Face-based estimations of curvatures on triangle meshes, *Journal for Geometry and Graphics* 12(2008), No. 1, 63-73.

[13] M. SZILVÁSI-NAGY, Construction of circular splats on analytic surfaces. In *Fifth Hungarian Conference on Computer Graphics and Geometry*, Budapest 2010, 55-57.

[14] M. SZILVÁSI-NAGY, Surface patches constructed from curvature data. *KoG* 14 (2010) 29-34.

[15] G. TAUBIN, Estimating the tensor of curvature of a surface from a polyhedral approximation, in *ICCV'95 Proceedings of the Fifth International Conference on Computer Vision, IEEE Computer Society*, Washington DC, USA, 902.

[16] J. WU, L. KOBBELT, Optimized sub-sampling of point sets for surface splatting. *In: Eurographics 2004 (Eds. M.-P. Cani, M. Slater)* 23 (2004), 643-652.

Márta Szilvási-Nagy
e-mail: szilvasi@math.bme.hu
Szilvia Béla
email: belus@math.bme.hu

Dept. of Geometry
Budapest University of Technology and Economics
H-1521 Budapest, Hungary