Graphical Models 71 (2009) 107–112

Contents lists available at ScienceDirect

Graphical Models

journal homepage: www.elsevier.com/locate/gmod

Computing the minimum distance between a point and a clamped B-spline surface

Xiao-Diao Chen [a,d,*], Gang Xu [a,c], Jun-Hai Yong [b], Guozhao Wang [c], Jean-Claude Paul [b,d]

[a] Hangzhou Dianzi University, Hangzhou 310018, PR China
[b] School of Software, Tsinghua University, Beijing 100084, PR China
[c] Institute of Computer Graphics and Image Processing and the Department of Mathematics, Zhejiang University, Hangzhou 310027, PR China
[d] INRIA, Le Chesnay Cedex 78153, France

ARTICLE INFO

Article history:
Received 10 February 2008
Received in revised form 28 May 2008
Accepted 9 January 2009
Available online 24 January 2009

Keywords:
Minimum distance
B-spline surface
Newton's method
Exclusion criterion

ABSTRACT

The computation of the minimum distance between a point and a surface is important for the applications such as CAD/CAM, NC verification, robotics and computer graphics. This paper presents a spherical clipping method to compute the minimum distance between a point and a clamped B-spline surface. The surface patches outside the clipping sphere which do not contain the nearest point are eliminated. Another exclusion criterion whether the nearest point is on the boundary curves of the surface is employed, which is proved to be superior to previous comparable criteria. Examples are also shown to illustrate efficiency and correctness of the new method.

1. Introduction

The computation problem of the minimum distance between a point and a surface is also called the point projection problem of the surface. The distance information between a point and a surface is very important for interactively selecting surfaces and surface construction in geometric modeling [3,5], for collision detection and physical simulation in computer graphics and computer vision, for interference avoidance in CAD/CAM and NC verification [4,8]. It is also essential for the B-spline surface fitting problem [8,12].

Suppose that a surface has a parametric form $\mathbf{S}(u,v)$, and that the squared distance from $\mathbf{S}(u,v)$ to a test point $\mathbf{p}$ is $H(u,v) = (\mathbf{S}(u,v) - \mathbf{p})^2$. Then the following system of equations

$$\begin{cases} H_u(u,v) = (\mathbf{S}(u,v) - \mathbf{p}) \cdot \mathbf{S}_u = 0 \\ H_v(u,v) = (\mathbf{S}(u,v) - \mathbf{p}) \cdot \mathbf{S}_v = 0 \end{cases} \tag{1}$$

is satisfied at the nearest point on the surface $\mathbf{S}(u,v)$ to the test point $\mathbf{p}$. Thus, the point projection problem is turned into a root-finding problem of the equation system (1) [2,7,10,13]. There are two main steps in the above methods. The first step is to remove the regions or patches which contain no solutions, and the second one is to compute local extrema with numerical methods such as the Newton method. Zhou et al. use both the Projected-Polyhedron and Linear Programming methods for the first step [13], while Johnson utilizes the tangent cone method instead [2].

However, not all of the solutions of the equation system (1) need to be computed. In the worst case, as shown in Fig. 1, the nearest point $\mathbf{q}$ is a corner control point of the surface, and it is not mapped to any solution of the equation system (1). Thus, all of the computation on solving the solutions of the equation system (1) is unnecessary.

Recently, instead of analyzing the solutions of the equation system (1), several geometric methods combining elimination technique with subdivision technique are used for the point projection problem of NURBS surfaces [5,8,11]. In these methods, the surface may be iteratively subdivided into surface patches, and most of the surface

* Corresponding author. Address: Hangzhou Dianzi University, Hangzhou 310018, PR China.
E-mail address: xiaodiao@nit.zju.edu.cn (X.-D. Chen).

1524-0703/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.gmod.2009.01.001

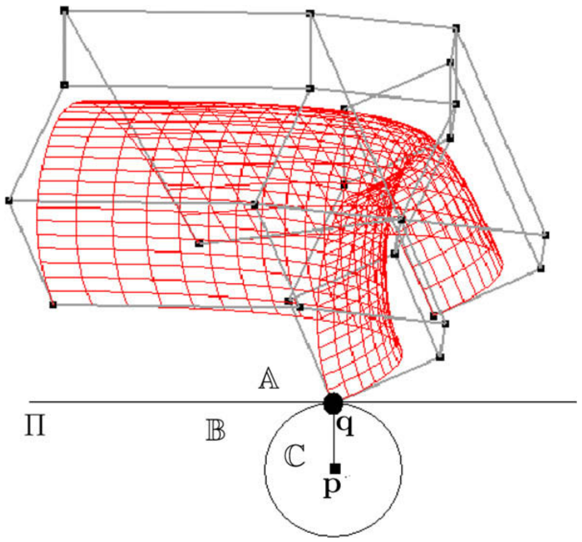

Fig. 1. An example shows that the nearest point is not the solution of Eq. (1). **p** is a test point, and the nearest point **q** is one of the corner points of the surface. All the other control points are above the plane Π, which passes through **q** and is perpendicular to the direction vector from **p** to **q**.

patches not containing the nearest point are eliminated by analyzing the relationship between these surface patches and the test point. When the remaining surface patches are small or flat enough, then the Newton's method is used to solve the nearest point for each remaining patch. The subdivision process of the NURBS surfaces is robust, and these methods seem geometrically more intuitive, reasonable and robust. The key technique of these geometric methods is how to eliminate the removable surface patches. Piegl and Tiller directly decompose the base surface into quadrilaterals within a relative geometric tolerance [8]. However, such a decomposition process is pointed out to be time-consuming [5]. To reduce the computation time of decomposition, Ma and Hewitt present a control polygon approach for the same problem [5]. They provide a condition that the nearest point is on one of the four boundary curves. Selimovic provides an improved algorithm by using two exclusion criteria [11]. The first one is a sufficient condition that the nearest point is one of the four corner control points, and the second one is another sufficient condition that the nearest point is on one of the four boundary curves. For the case shown in Fig. 1, the method of [11] prunes the surface patch directly, and costs no computation on solving the solutions of Eq. (1).

This paper presents a spherical clipping method for the point projection problem of clamped B-spline surfaces. A clipping sphere with its center point being the test point **p** is introduced, and the radius of the sphere is set by the Euclidean distance between the test point **p** and a point of the surface. Thus, any surface patch outside the clipping sphere can be eliminated. Another exclusion criterion is also employed by using a new sufficient condition whether the nearest point is on one of the boundaries curves of the surface. The algorithms in [5] have an overlooked flaw and may fail in some cases [1]. For clamped B-spline surfaces, we will prove that the criterion in our algorithm is superior to the comparable criterion in [11].

We assume that the clamped B-spline surface is defined by

$$\mathbf{S}(u,v)=\sum_{i=0}^{n}\sum_{j=0}^{m}B_{i,p}(u)B_{j,q}(v)\mathbf{P}_{i,j},\quad u\in[u_0,u_r],\ v\in[v_0,v_k],\tag{2}$$

where u and v are the parameters of the surface, p and q are the degrees of the surface in u and v directions, $\{\mathbf{P}_{i,j}\}$ are the control points in the space $\mathbb{R}^3$, $\{B_{i,p}(u)\}$ and $\{B_{j,q}(v)\}$ are the pth-degree and the qth-degree B-spline basis functions based on the knot vectors $U=(u_0,\ldots,u_0,u_1,\ldots,u_{r-1},u_r,\ldots,u_r)$ and $V=(v_0,\ldots,v_0,v_1,\ldots,v_1,\ldots,v_k,\ldots,v_k)$, respectively.

This paper is organized as follows. Section 2 presents the outline of the new algorithm. Section 3 compares the new method with the method in [11]. Analysis and examples are also shown in this section. Some conclusions are drawn at the end of this paper.

2. Algorithm for the point projection problem for B-spline surfaces

The spherical clipping method is explained in this section. The basic idea is as follows. Suppose that the point **p** is a test point, the point **q** is a point on the surface, and $\mathbf{O}(\mathbf{p},||\mathbf{pq}||)$ is a sphere with the center point **p** and its radius $||\mathbf{pq}||$. The nearest point to the point **p** must be inside the sphere $\mathbf{O}(\mathbf{p},||\mathbf{pq}||)$. Thus, any surface patch outside the sphere $\mathbf{O}(\mathbf{p},||\mathbf{pq}||)$ can be directly eliminated. We also call sphere $\mathbf{O}(\mathbf{p},||\mathbf{pq}||)$ the clipping sphere. During the subdivision process, the point **q** becomes more and more close to the test point **p** and the radius of the clipping sphere becomes smaller and smaller, which will lead to higher and higher elimination efficiency.

One of the key issues for the spherical clipping method is to judge whether a surface patch is outside a sphere. If all the control points of a surface patch are inside a sphere, then the patch must be inside the sphere too. Thus, the clipping sphere seems to be useful to compute the maximum distance between a point and a surface. Unfortunately, even if all the control points of a surface are outside a sphere, we can't ensure that the surface is outside the sphere. It seems not easy to judge whether a surface is outside a sphere directly by its control net.

To overcome this problem, we introduce the objective squared distance function for judging whether a surface is outside a sphere. By using the product formula of two B-spline basis functions in [6], the objective square distance function

$$F(u,v)=(\mathbf{p}-\mathbf{S}(u,v))^2$$

can be turned into B-spline form

$$F(u,v)=\sum_{i=0}^{\kappa}\sum_{j=0}^{\iota}\hat{B}_{i,2p}(u)\hat{B}_{j,2q}(v)f_{i,j},\tag{3}$$

where the new knot vectors $\hat{U}$ and $\hat{V}$ for $\hat{B}_{i,2p}(u)$ and $\hat{B}_{j,2q}(v)$ are

$$\hat{U} = (\underbrace{u_0, \ldots, u_0}_{2p+1}, \underbrace{u_1, \ldots, u_1}_{a_1+p}, \ldots, \underbrace{u_r, \ldots, u_r}_{2p+1}) \quad \text{and}$$

$$\hat{V} = (\underbrace{v_0, \ldots, v_0}_{2q+1}, \underbrace{v_1, \ldots, v_1}_{b_1+q}, \ldots, \underbrace{v_k, \ldots, v_k}_{2q+1}),$$

and

$$f_{ij} = \sum_a \sum_b \sum_c \sum_d \hat{\mathbf{P}}_{ab} \hat{\mathbf{P}}_{cd} \left\{ \sum_\beta \lambda^1_{a,p,\beta,i} \lambda^2_{c,p,\beta,i} / C^{2p}_p \right\}$$
$$\left\{ \sum_\gamma \chi^1_{b,q,\gamma,j} \chi^2_{d,q,\gamma,j} / C^{2q}_q \right\}. \tag{4}$$

Here C^j_i denotes the binomial coefficient $\binom{j}{i}$, $\hat{\mathbf{P}}_{ij} = \mathbf{P}_{ij} - \mathbf{p}$, $\lambda^1_{a,p,\beta,i}$ and $\lambda^2_{c,p,\beta,i}$ are non-negative real numbers depending on $i, a, c, \beta, \mathbf{U}$ and $\hat{\mathbf{U}}$, $\chi^1_{b,q,\gamma,j}$ and $\chi^2_{d,q,\gamma,j}$ are non-negative real numbers depending on $j, b, d, \gamma, \mathbf{V}$ and $\hat{\mathbf{V}}$. If j is equal to zero, we have

$$f_{i0} = \sum_a \sum_c \hat{\mathbf{P}}_{a0} \hat{\mathbf{P}}_{c0} \left\{ \sum_\beta \lambda^1_{a,p,\beta,i} \lambda^2_{c,p,\beta,i} / C^{2p}_p \right\}. \tag{5}$$

When $\mathbf{S}(u, v)$ is a Bézier surface, we have

$$f_{i,j} = \sum_{k=0}^{i} \sum_{l=0}^{j} C^p_k C^{2p}_{i-k} C^q_l C^q_{j-l} \hat{\mathbf{P}}_{k,l} \cdot \hat{\mathbf{P}}_{i-k,j-l} / (C^{2p}_i C^{2q}_j) \tag{6}$$

and

$$f_{i,0} = \sum_{k=0}^{i} (C^p_k C^p_{i-k} \hat{\mathbf{P}}_{k,0} \cdot \hat{\mathbf{P}}_{i-k,0}) / C^{2p}_i,$$

where $\hat{\mathbf{P}}_{i,j} = \mathbf{0}$, forall $i > p$ or $j > q$. The spherical clipping method may be mathematically described as the following theorem, which can be obtained directly from the convex hull property of B-spline basis functions.

Theorem 1. *Suppose that $\alpha \in \mathbb{R}^+$. If $f_{i,j} \geqslant \alpha^2$ for all $0 \leqslant i \leqslant \kappa$ and $0 \leqslant j \leqslant \imath$, then $F(u, v) \geqslant \alpha^2$, which means that the surface is outside the clipping sphere $\mathbf{O}(\mathbf{p}, \alpha)$.*

At the beginning of the spherical clipping method, α can be set as $||\mathbf{pq}||$, where $\mathbf{q}$ is one of the four corner control points of the surface. If Theorem 1 is satisfied, then all points on the surface except $\mathbf{q}$ are outside of the sphere with the center point $\mathbf{p}$ and its radius $||\mathbf{pq}||$. So the point $\mathbf{q}$ is the nearest point. During the subdivision process, the radius of the sphere α becomes smaller and smaller, which will lead to better elimination efficiency.

We also provide another sufficient condition whether the nearest point is on one of the four boundary curves of the surface.

Theorem 2. *If there exists an integer $k = 0$ or $k = \kappa$, such that for all $i \in \{0, 1, \ldots, \kappa\}$ and $j \in \{0, 1, \ldots, \imath\}$, $f_{i,j} \geqslant f_{k,j}$, then $F(u, v)$ must reach the minimum value at the boundary $u = u_0$ or $u = u_r$. Similarly, if there exists an integer $k = 0$ or $k = \imath$, such that for all $i \in \{0, 1, \ldots, \kappa\}$ and $j \in \{0, 1, \ldots, \imath\}$, $f_{i,j} \geqslant f_{i,k}$, then $F(u, v)$ must reach the minimum value at the boundary $v = v_0$ or $v = v_k$.*

Proof 1. We only prove the former sufficient condition. Without loss of generality, we may assume that $f_{i,j} \geqslant f_{0,j}$, for all $0 \leqslant i \leqslant \kappa$ and $0 \leqslant j \leqslant \imath$. For all $u \in [u_0, u_r], v \in [v_0, v_k]$, we have

$$\begin{aligned}
F(u, v) &= \sum_{i=0}^{\kappa} \sum_{j=0}^{\imath} \hat{B}_{i,2p}(u) \hat{B}_{j,2q}(v) f_{i,j} \\
&= \sum_{j=0}^{\imath} \left\{ \sum_{i=0}^{\kappa} \hat{B}_{i,2p}(u) f_{i,j} \right\} \hat{B}_{j,2q}(v) \\
&\geqslant \sum_{j=0}^{\imath} \left\{ \sum_{i=0}^{\kappa} \hat{B}_{i,2p}(u) f_{0,j} \right\} \hat{B}_{j,2q}(v) \\
&= \sum_{j=0}^{\imath} \left\{ \sum_{i=0}^{\kappa} \hat{B}_{i,2p}(u) \right\} f_{0,j} \hat{B}_{j,2q}(v) = \sum_{j=0}^{\imath} f_{0,j} \hat{B}_{j,2q}(v) \\
&= F(u_0, v).
\end{aligned}$$

Thus, $F(u, v)$ must reach the minimum value at the boundary $u = u_0$, and we complete the proof. □

If the nearest point is detected to be a corner point or on the boundary of the original surface by using Theorems 1 and 2, we return the corner point or the nearest point on the boundary curve immediately. Without loss of generality, we assume that the nearest point is neither a corner point nor on the boundary of the original surface. The outline of our algorithm is described as follows. Firstly, we calculate the objective function $F(u, v)$, which describes a clamped B-spline surface $\mathbf{S}^F$ in $\mathbb{R}^1$ space. Secondly, we split the surface $\mathbf{S}^F$, and after each split, we try to eliminate each of the parts by analyzing the control net of $\mathbf{S}^F$ with Theorems 1 and 2. The subdivision process stops as soon as a flatness condition is satisfied or the remaining region is within a given tolerance. Finally, we apply the globalized Newton method as described in [9] for each of the remaining patches. Another similar outline can be found in [11].

3. Analysis and examples

Selimovic provides two exclusion criteria for the point projection problem of NURBS surfaces [11]. The first exclusion criterion is based on the plane clipping method. As shown in Fig. 1, the plane and the sphere separate the whole space into three parts $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{C}$. From the geometrical intuition, the plane clipping method in [11] can eliminate the surface patches in part $\mathbb{A}$, while our spherical clipping method may eliminate both parts $\mathbb{A}$ and $\mathbb{B}$. Moreover, the plane clipping method in [11] seems unable to compute the farthest point problem, while our spherical clipping method can be used to compute the farthest point problem by changing "outside" into "inside" in the corresponding algorithm.

Theorem 3 will show that the second exclusion criterion in Theorem 2 extends the comparable one in [11].

Theorem 3. *Given a clamped B-spline surface, if the second exclusion criterion in [11] that the nearest point is on one of the four boundary curves is satisfied, then the criterion in Theorem 2 is satisfied.*

Proof 2. Suppose that the constructive tangent cone $\mathbf{T}_c$ of the surface $\mathbf{S}(u, v)$ is given by the axis $\mathbf{Z}$ and the angle β, and that the $\mathbf{p}$ is a test point. If the exclusion criterion in [11] that the nearest point is on one of the four boundary curves is satisfied, i.e.,

$$\angle(\mathbf{P}_{i,0}-\mathbf{p},\mathbf{Z}) < \pi/2-\beta, \quad \text{for all } 0 \leqslant i \leqslant n, \tag{7}$$

we will prove that $f_{i,0} \leqslant f_{i,j}$ for all i and j, which is the criterion in Theorem 2. From Eq. (7), any angle between two vectors in $\mathbf{T}_c$ is less than $\pi/2$. The direction vector $\overrightarrow{\mathbf{P}_{i,k}\mathbf{P}_{i,k+1}}$ is in the tangent cone $\mathbf{T}_c$, so we have

$$(\mathbf{P}_{i,k+1}-\mathbf{P}_{i,k})\cdot(\mathbf{P}_{j,l+1}-\mathbf{P}_{j,l}) \geqslant 0. \tag{8}$$

Combining with Eqs. (7) and (8), we have

$$\begin{aligned}\hat{\mathbf{P}}_{a,b}\cdot\hat{\mathbf{P}}_{c,d} &= \left\{\hat{\mathbf{P}}_{a,0}+\sum_{k=1}^{b}(\mathbf{P}_{a,k}-\mathbf{P}_{a,k-1})\right\}\\ &\quad\cdot\left\{\hat{\mathbf{P}}_{c,0}+\sum_{h=1}^{d}(\mathbf{P}_{c,h}-\mathbf{P}_{c,h-1})\right\}\\ &\geqslant \hat{\mathbf{P}}_{a,0}\cdot\hat{\mathbf{P}}_{c,0}.\end{aligned}$$

Since $1=\sum_b B_{b,q}(v)$ and $1=\sum_j \hat{B}_{j,2q}(v)$, we have

$$\sum_b B_{b,q}(v)\sum_d B_{d,q}(v) = \sum_j \hat{B}_{j,2q}(v).$$

From the formula (3.2) in [6], we have

$$\sum_b B_{b,q}(v)\sum_d B_{d,q}(v) = \sum_j\left(\sum_b\sum_d\left\{\sum_\gamma \chi^1_{b,q,\gamma,j}\chi^2_{d,q,\gamma,j}/C^{2q}_q\right\}\right)\hat{B}_{j,2q}(v).$$

Thus, we obtain

$$\sum_j \hat{B}_{j,2q}(v) = \sum_j\left(\sum_b\sum_d\left\{\sum_\gamma \chi^1_{b,q,\gamma,j}\chi^2_{d,q,\gamma,j}/C^{2q}_q\right\}\right)\hat{B}_{j,2q}(v).$$

From the linear independent property of the B-spline basis functions, we have

$$\sum_b\sum_d\left\{\sum_\gamma \chi^1_{b,q,\gamma,j}\chi^2_{d,q,\gamma,j}/C^{2q}_q\right\} = 1. \tag{9}$$

Combining with Eqs. (4), (5) and (9), we have

$$\begin{aligned}f_{i0} &= \sum_a\sum_c \hat{\mathbf{P}}_{a0}\hat{\mathbf{P}}_{c0}\left\{\sum_\beta \lambda^1_{a,p,\beta,i}\lambda^2_{c,p,\beta,i}/C^{2p}_p\right\}\\ &= \sum_a\sum\nolimits_c \hat{\mathbf{P}}_{a0}\hat{\mathbf{P}}_{c0}\left\{\sum_\beta \lambda^1_{a,p,\beta,i}\lambda^2_{c,p,\beta,i}/C^{2p}_p\right\}\\ &\qquad\times\sum_b\sum_d\left\{\sum_\gamma \chi^1_{b,q,\gamma,j}\chi^2_{d,q,\gamma,j}/C^{2q}_q\right\}\\ &= \sum_a\sum_b\sum_c\sum_d \hat{\mathbf{P}}_{a0}\hat{\mathbf{P}}_{c0}\left\{\sum_\beta \lambda^1_{a,p,\beta,i}\lambda^2_{c,p,\beta,i}/C^{2p}_p\right\}\\ &\qquad\times\left\{\sum_\gamma \chi^1_{b,q,\gamma,j\chi^2_{d,q,\gamma,j}}/C^{2q}_q\right\}\\ &\leqslant \sum_a\sum_b\sum_c\sum_d \hat{\mathbf{P}}_{ab}\hat{\mathbf{P}}_{cd}\left\{\sum_\beta \lambda^1_{a,p,\beta,i}\lambda^2_{c,p,\beta,i}/C^{2p}_p\right\}\\ &\qquad\times\left\{\sum_\gamma \chi^1_{b,q,\gamma,j\chi^2_{d,q,\gamma,j}}/C^{2q}_q\right\}\\ &= f_{ij}.\end{aligned}$$

So we complete the proof. □

Fig. 2 shows two examples. In Fig. 2a, with our spherical clipping method, we can judge that the point $\mathbf{P}_{0,0}$ is the nearest point. But both exclusion criteria of [11] are not satisfied. In Fig. 2b, the shape of the surface is very complicated. After one step of subdivision, one of the two surface patches is outside the sphere $\mathbf{O}(\mathbf{p},||\mathbf{P}_{00}\mathbf{p}||)$ and can be eliminated by the spherical clipping method, the remaining surface patch is verified to satisfy the exclusion criterion in Theorem 2. However, it can be verified that both exclusion criteria of [11] are not satisfied with both of the two surface patches, and it needs more subdivision steps in the method in [11].

Fig. 3 shows three more examples. The test surfaces in Fig. 3a and b are Bézier surfaces, and Fig. 3c shows a clamped B-spline surface case. The test points are sampled from a d-offset surface of the given surface with a small d. Table 1 shows the corresponding comparison results between the method in [11] and our new method. Both the average subdivision time and the average computation time of the new method are less than those of the method of [11]. Usually, it is very easy to compute f_{ij} in Eq. (4) for Bézier surfaces by using Formula (6), while it seems to be a bit time-consuming to compute the objective squared distance function in B-spline form for B-spline surfaces.

Fig. 4 shows two examples comparing both efficiency and correctness between the new method and the method in [11]. There are altogether 441 test points sampled from a self-intersected offset surface of the original surface at parameters $\{(i/20, j/20)\}$, where $i,j=0,\ldots,20$. The corresponding results are shown in Table 2. In Table 2, T_s and T_c denote the subdivision time and the computation time, $M_{\text{Selimovic}}$ and M_{new} denote the method in [11] and the new method, N_i denotes the number of the resulting nearest points which are within the tolerance 10^{-i}, $i=3,4,5$, and N_2 denotes the number of the points which are beyond the tolerance 10^{-2}. As shown in Table 2, all of the resulting nearest points in the new method are within the tolerance 10^{-5}. While in the method of [11], about 2% points are not within the tolerance 10^{-5}, and some of them are even not within the tolerance 10^{-2}. All these above examples are implemented in Visual C++ on Windows PC with 1.7 G CPU and 1.0 G memory.

4. Conclusions

A spherical clipping method on computing the point projection problem for clamped B-spline surfaces is presented. Compared with the root-finding method, the new method can geometrically eliminate most of the roots not mapping to the nearest point. Compared with the geometrical method in [11], as shown from geometrical intuition, our spherical clipping method is able to eliminate more surface patches than the plane clipping method of [11]. The new exclusion criterion in Theorem 2 is proved to extend the comparable one in [11]. Examples are also shown to illustrate efficiency of the new method. Our future work will be to extend the new method to the point projection problem for rational B-spline surfaces.

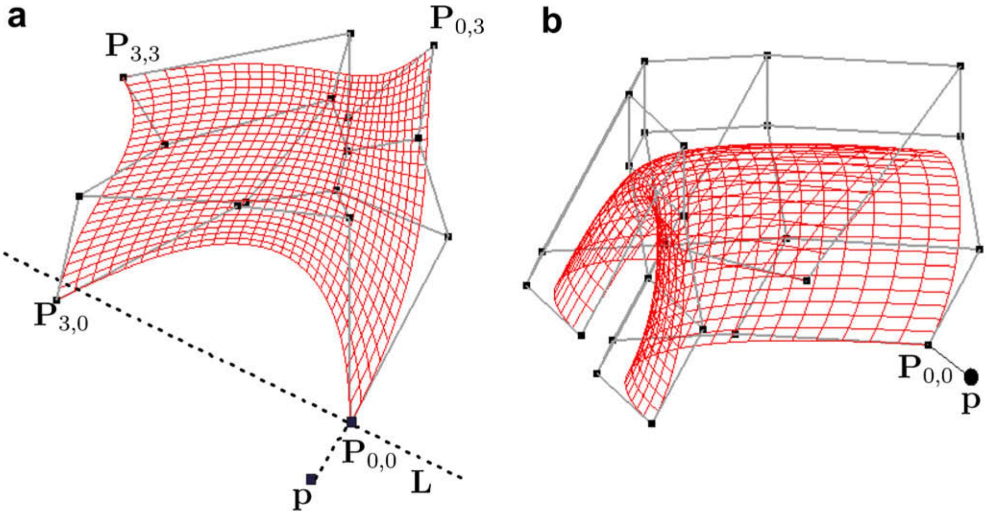

Fig. 2. Example showing the efficiency of the exclusion criteria for B-spline surfaces.

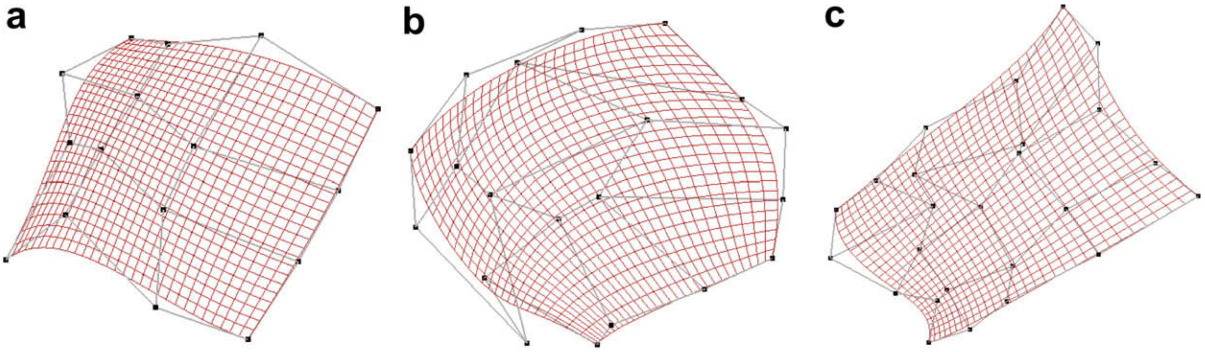

Fig. 3. Tested surfaces: (a) a Bézier surface, (b) a Bézier surface, (c) a B-spline surface.

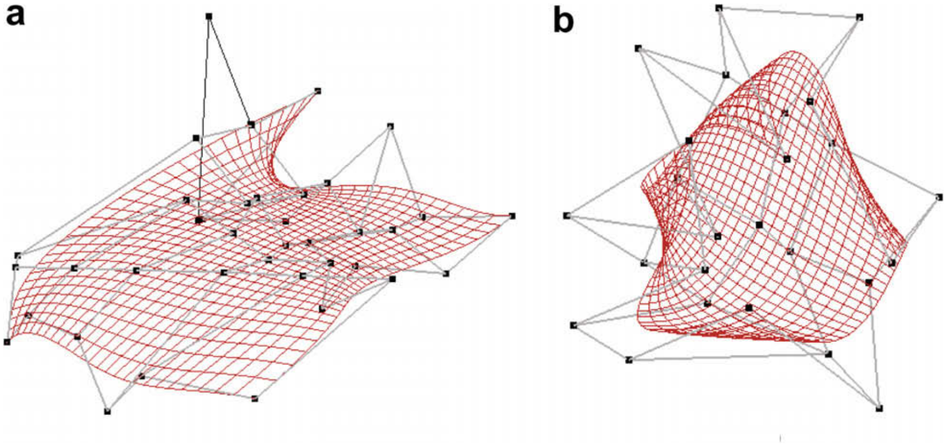

Fig. 4. Examples for illustrating both efficiency and correctness.

Table 1
Comparison on subdivision time and computation time.

Method	Subdivision time		Computation time	
	Selimovic	Chen et al.	Selimovic	Chen et al.
Fig. 3a	109.1	56.2	1.029 ms	0.317 ms
Fig. 3b	125.9	67.0	1.127 ms	0.484 ms
Fig. 3c	111.4	66.3	1.262 ms	0.653 ms

Table 2
Comparison on both efficiency and correctness.

Method	T_s		T_c (ms)		$M_{\text{Selimovic}}$				M_{new}	
	M_s	M_n	M_s	M_n	N_5	N_4	N_3	N_2	N_5	N_4
Fig. 4a	621.4	102.7	9.140	1.274	433	3	3	2	441	0
Fig. 4b	816.4	100.9	7.015	0.816	429	4	4	4	441	0

Acknowledgments

The research was supported by the Chinese 973 Program (2004CB719403, 2004CB318000) and the National Science Foundation of China (60803076, 60625202, 60533070). The third author was supported by a Foundation for the Author of National Excellent Doctoral Dissertation of PR China (200342) and Fok Ying Tung Education Foundation (111070). The authors would like to thank the reviewers for their helpful comments.

References

[1] X.D. Chen, H. Su, J.H. Yong, J.C. Paul, J.G. Sun, A counterexample on point inversion and projection for nurbs curve, Computer Aided Geometric Design 24 (5) (2007) 302.

[2] E. Cohen, D.E. Johnson, Distance extrema for spline models using tangent cones, In: Proceedings of the 2005 Conference on Graphics Interface, 2005, pp. 169–175.

[3] S.M. Hu, J. Wallner, A second order algorithm for orthogonal projection onto curves and surfaces, Computer Aided Geometric Design 22 (3) (2005) 251–260.

[4] K.J. Kim, Minimum distance between a canal surface and a simple surface, Computer-Aided Design 35 (10) (2003) 871–879.

[5] Y.L. Ma, W.T. Hewitt, Point invers\ion and projection for nurbs curve and surface: control polygon approach, Computer Aided Geometric Design 20 (2) (2003) 79–99.

[6] K. Morken, Some identities for products and degree raising of splines, Constructive Approximation 7 (2) (1991) 195–208.

[7] M.E. Mortenson, Geometric Modeling, Wiley, New York, 1985.

[8] L. Piegl, W. Tiller, Parametrization for surface fitting in reverse engineering, Computer-Aided Design 33 (8) (2001) 593–603.

[9] E. Polak, Optimization, Algorithms and Consistent Approximations, Springer-Verlag, Berlin, Heidelberg, New York, 1997.

[10] E. Polak, J.O. Royset, Algorithms with adaptive smoothing for finite minimax problems, Journal of Optimization: Theory and Applications 119 (3) (2003) 459–484.

[11] I. Selimovic, Improved algorithms for the projection of points on nurbs curves and surfaces, Computer Aided Geometric Design 23 (5) (2006) 439–445.

[12] H.P. Yang, W.P. Wang, J.G. Sun, Control point adjustment for B-spline curve approximation, Computer-Aided Design 36 (7) (2004) 639–652.

[13] J.M. Zhou, E.C. Sherbrooke, N. Patrikalakis, Computation of stationary points of distance functions, Engineering with Computers 9 (1993) 231–246.