Computer-Aided Design 45 (2013) 628–638

Contents lists available at SciVerse ScienceDirect

Computer-Aided Design

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

An algorithm to improve parameterizations of rational Bézier surfaces using rational bilinear reparameterization

Yi-Jun Yang [a,*], Wei Zeng [c], Cheng-Lei Yang [a], Bailin Deng [b], Xiang-Xu Meng [a], S. Sitharama Iyengar [c]

[a] School of Computer Science & Technology, Shandong University, Jinan, China

[b] Computer Graphics and Geometry Laboratory, EPFL, Switzerland

[c] School of Computing & Information Sciences, Florida International University, USA

ARTICLE INFO

Article history:
Received 8 September 2011
Accepted 4 October 2012

Keywords:
Rational Bézier surfaces
Rational bilinear reparameterization
Surface parameterization

ABSTRACT

The parameterization of rational Bézier surfaces greatly affects rendering and tessellation results. The uniformity and orthogonality of iso-parametric curves are two key properties of the optimal parameterization. The only rational Bézier surfaces with uniform iso-parametric curves are bilinear surfaces, and the only rational Bézier surfaces with uniform and orthogonal iso-parametric curves are rectangles. To improve the uniformity and orthogonality of iso-parametric curves for general rational Bézier surfaces, an optimization algorithm using the rational bilinear reparameterizations is presented, which can produce a better parameterization with the cost of degree elevation. Examples are given to show the performance of our algorithm for rendering and tessellation applications.

1. Introduction

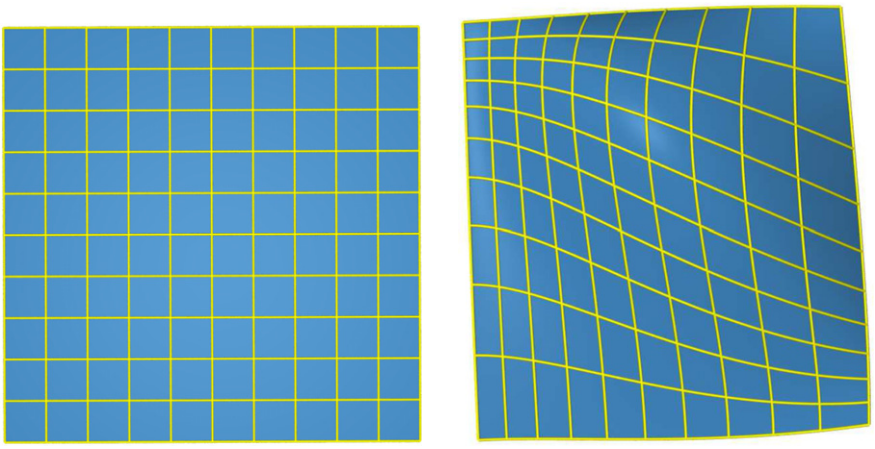
Freeform surfaces play an increasingly important role in contemporary Computer Aided Design (CAD). While the manufacturers are mainly concerned with the final geometric shapes, most algorithms [1–13] for surface rendering (e.g. texture mapping), tessellation and blending applications are highly dependent on the surface parameterization. To generate a target geometric shape, the control points and their weights of the rational Bézier surface are adjusted by the designer. Such modification may destroy some desirable properties of the surface parameterization such as the uniformity and orthogonality of iso-parametric curves (see Fig. 1), and affect the subsequent surface manipulations such as surface tessellation and surface rendering (see Fig. 2). As a result, either the designer is forced to make a conservative modification, or some reparameterization technique has to be introduced to improve the surface parameterization, which is the aim of our paper.

In the past twenty years, how to achieve optimal parameterization of Bézier curves has been studied extensively in the literature such as [14–24]. Farouki [15] identified arc-length parameterization as the optimal parameterization of Bézier curves. By minimizing an integral which measures the deviation from arc-length parameterization, the optimal representation is obtained by solving a quadratic equation. Jüttler [16] presented a simplified approach to Farouki's result by using a back substitution in the integral. Costantini et al. [14] obtained closer approximations to the arc-length parameterization by applying composite reparameterizations to Bézier curves.

To our knowledge, little attention has been paid to the Bézier surface reparameterization. The results of rendering and tessellation applications for Bézier surfaces largely depend on the parameterization quality. Moreover, a parameterization with uniform and orthogonal iso-parametric curves will lead to more robust and stable computations for derivative based algorithms such as surface intersection, curvature computation, and so on [1,6,9,11]. To perform texture mappings on a Bézier surface, the parametric coordinate of the surface is usually reused as the texture coordinate. If the iso-parametric curves are far from being uniform and orthogonal, there will be large distortion of the texture image on the surface (see Fig. 2(b)). To tessellate a Bézier surface, most existing algorithms [4,7,6] map a triangulation of the parameter domain onto the surface (see Fig. 2(c) and (d)). Similar to texture mapping, the final tessellation results largely depend on the surface parameterization. Yang et al. [25] presented an algorithm to improve the Bézier surface parameterization based on Möbius transformations [26,25], which can change only the distribution of iso-parametric curves, but not their shape. To obtain more uniform iso-parametric curves, a rational bilinear reparameterization algorithm was also presented in [25]. However, only the uniformity of iso-parametric curves was considered therein. Furthermore, the rational bilinear reparameterization coefficients are determined by a trivial interpolation method, which is only suitable for a

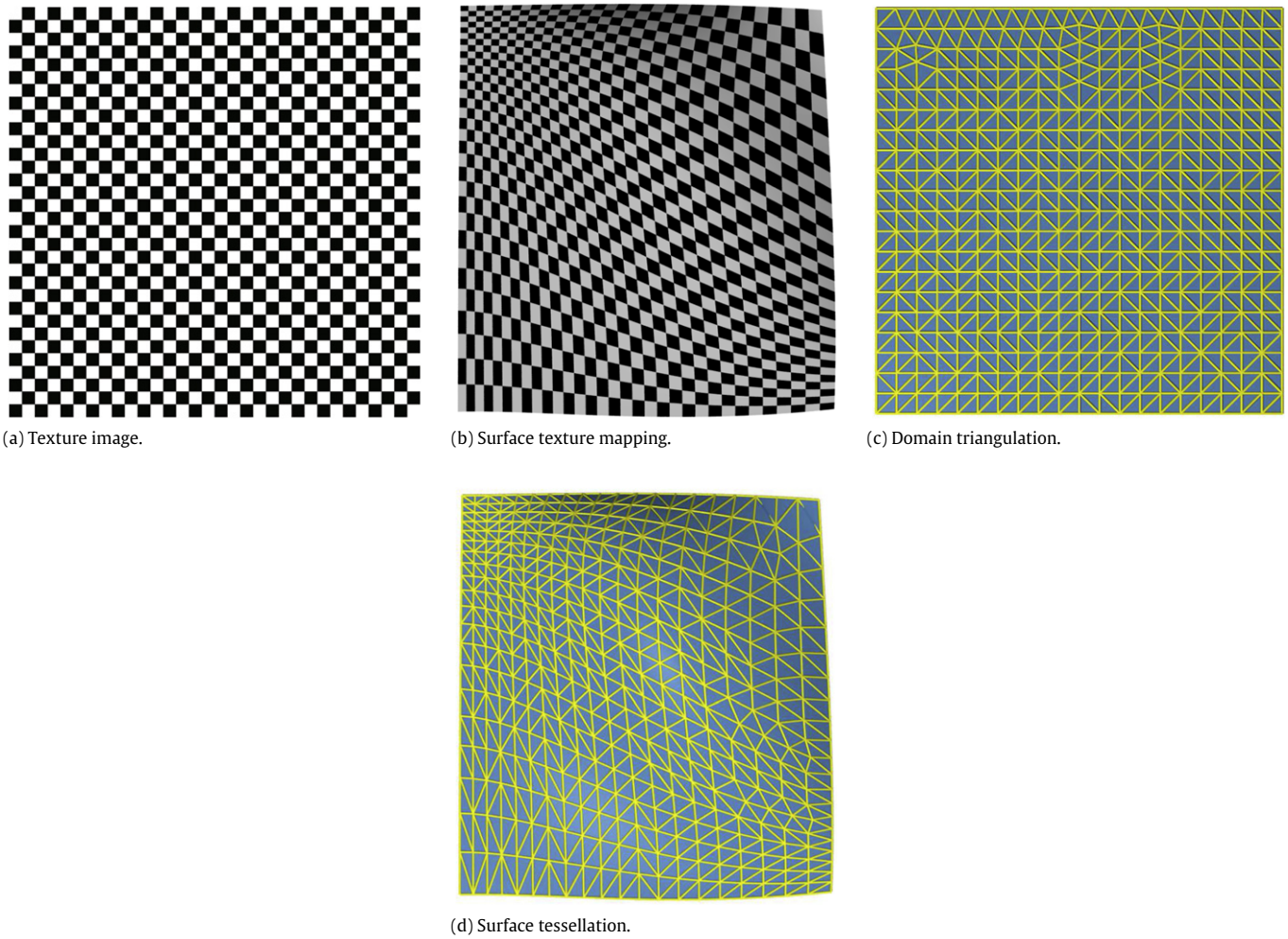
* Correspondence to: No. 1500, Shunhua Road, Gaoxin District, Shandong University, Jinan, Postal Code: 250100, China. Tel.: +86 15153126252.
E-mail address: yangyijuny@gmail.com (Y.-J. Yang).

0010-4485/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2012.10.048



(a) Original surface. (b) Modified surface.

Fig. 1. Surface modification: (a) a rational Bézier surface and its parameterization; (b) the modified Bézier surface and its parameterization after changing several control points and their weights.



(a) Texture image. (b) Surface texture mapping. (c) Domain triangulation.

(d) Surface tessellation.

Fig. 2. Texture mapping and tessellation results of the rational Bézier surface in Fig. 1(b): (a) texture image; (b) texture mapping result; (c) triangulation of the parameter domain; (d) tessellation result by mapping triangles in (c) onto the surface.

special surface case. Different from the curve cases, a surface parameterization with only uniform iso-parametric curves is usually not enough for CAD applications. Besides the uniformity of iso-parametric curves, orthogonality is also considered as an important factor in these applications [1–11]. A surface parameterization with uniform and orthogonal iso-parametric curves not only preserves the appearance of texture, but also avoids degenerate elements for the tessellation application. From our point of view, the lack of satisfying parameterizations with both uniform and orthogonal iso-parametric curves is the bottleneck for Bézier surface rendering and tessellation algorithms to achieve better quality results.

In this paper, we first study the differential geometry of rational Bézier surfaces and try to find out the surfaces with uniform and orthogonal iso-parametric curves. We conclude that the only rational Bézier surface with uniform iso-parametric curves is a bilinear surface, and the only rational Bézier surface with both uniform and orthogonal iso-parametric curves is a rectangle. As a general surface has no parameterizations with completely uniform and orthogonal iso-parametric curves, we then present an optimization algorithm to improve the uniformity and orthogonality of iso-parametric curves based on the rational bilinear reparameterizations. In the optimization procedure, a nonlinear energy measuring the uniformity and orthogonality deviations is formulated. To minimize the uniformity and orthogonality deviations, a discrete version of the formulated energy is presented and then minimized numerically. The examples indicate that, in practice, the algorithm produces significantly more uniform and orthogonal iso-parametric curves across the rational Bézier surfaces.

The paper is organized as follows. Section 2 discusses the rational Bézier surfaces with uniform and orthogonal iso-parametric curves and shows how to use the rational bilinear reparameterizations to achieve more uniform and orthogonal iso-parametric curves across the rational Bézier surfaces. In Section 3, three examples are given to show the performance of our algorithm and Section 4 concludes the paper.

2. An optimization method for minimizing uniformity and orthogonality energies

In this section, we first try to find the surfaces with uniform and orthogonal iso-parametric curves. Then an optimization algorithm based on rational bilinear reparameterizations is presented to improve the uniformity and orthogonality of iso-parametric curves for general rational Bézier surfaces.

2.1. Differential geometry of rational Bézier surfaces

A rational Bézier surface can be represented in the following form

$$\mathbf{X}(u,v)=\frac{\sum_{i=0}^{m}\sum_{j=0}^{n}B_i^m(u)B_j^n(v)\omega_{i,j}\mathbf{P}_{i,j}}{\sum_{i=0}^{m}\sum_{j=0}^{n}B_i^m(u)B_j^n(v)\omega_{i,j}},\quad u\in[0,1]\text{ and }v\in[0,1],\tag{1}$$

where $\mathbf{P}_{i,j}$ and $\omega_{i,j}$ are the control points and their weights, $B_i^m(u)$ and $B_j^n(v)$ are the Bernstein polynomials. The parameterization in Eq. (1) is characterized by its first fundamental form [27]

$$ds^2=\mathbf{X}_u\cdot\mathbf{X}_u(du)^2+2\mathbf{X}_u\cdot\mathbf{X}_v du dv+\mathbf{X}_v\cdot\mathbf{X}_v(dv)^2,$$

where $\mathbf{X}_u=\frac{\partial\mathbf{X}}{\partial u}$ and $\mathbf{X}_v=\frac{\partial\mathbf{X}}{\partial v}$ are the two partial derivative vectors of the surface $\mathbf{X}$. The first fundamental form describes a metric on the surface $\mathbf{X}$. Let

$$E=\mathbf{X}_u\cdot\mathbf{X}_u,\qquad F=\mathbf{X}_u\cdot\mathbf{X}_v,\qquad G=\mathbf{X}_v\cdot\mathbf{X}_v,$$

and rewrite the coefficients in a symmetric matrix

$$\mathbf{I}=\begin{pmatrix}E & F\\ F & G\end{pmatrix},$$

where E and G give the square length of the two partial derivatives and F measures the orthogonality of the two partial derivatives. Then, we have

$$ds^2=(du\ dv)\mathbf{I}\begin{pmatrix}du\\ dv\end{pmatrix}.$$

Here we are interested in the uniformity and orthogonality of iso-parametric curves, which are analyzed and summarized in the next subsection.

2.2. Rational Bézier surfaces with uniform and orthogonal iso-parametric curves

A rational Bézier surface has uniform iso-parametric curves if and only if

$$\begin{cases}E_u=\dfrac{\partial(\mathbf{X}_u\cdot\mathbf{X}_u)}{\partial u}=0,\\ G_v=\dfrac{\partial(\mathbf{X}_v\cdot\mathbf{X}_v)}{\partial v}=0,\end{cases}\quad\text{for all }(u,v)\in[0,1]\times[0,1],\tag{2}$$

which describes the uniform distribution of iso-parametric curves across the surface. For such a surface satisfying Eq. (2), each iso-parametric curve is arc-length parameterized. Each iso-parametric curve of a rational Bézier surface is obviously a rational Bézier curve [8]. From Farouki's result in [28], the only arc-length parameterized rational Bézier curve is a straight line. Thus the four boundaries as well as the inner iso-parametric curves of the surface are straight lines, from which we can conclude that it is a bilinear surface (see Fig. 3). Furthermore, for this bilinear surface to have orthogonal iso-parametric curves, the boundary curves must intersect orthogonally at the four corner points, which means that it is a rectangle. Therefore, we conclude that the only rational Bézier surfaces with uniform and orthogonal iso-parametric curves are rectangles (see Fig. 3(b)). As a general rational Bézier surface has no parameterization with uniform and orthogonal iso-parametric curves, we will present an optimization algorithm to improve the uniformity and orthogonality of iso-parametric curves for a general rational Bézier surface in the following subsections.

2.3. Rational bilinear reparameterizations of rational Bézier surfaces

In a rational bilinear reparameterization $(u,v)=\psi(s,t)$ of a rational Bézier surface, the parameters are subjected to the following transformations

$$u=u(s,t)=\frac{(\alpha-1)s}{2\alpha s-s-\alpha},\quad\alpha\in(0,1),\tag{3}$$

and

$$v=v(s,t)=\frac{(\beta-1)t}{2\beta t-t-\beta},\quad\beta\in(0,1),\tag{4}$$

where

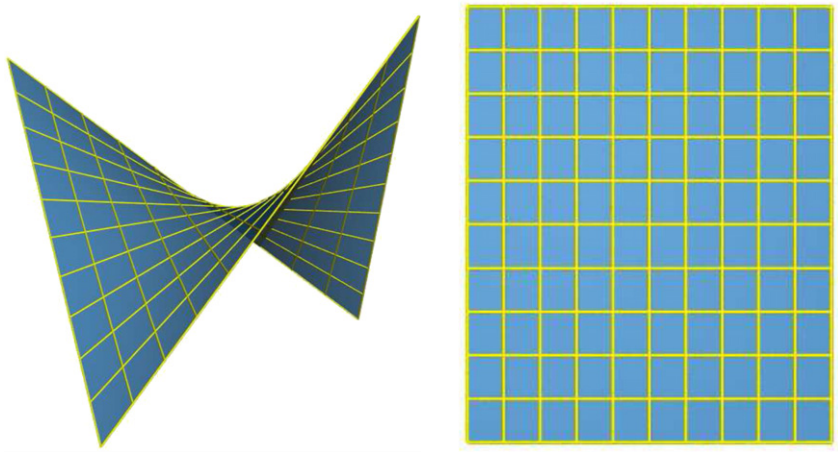
$$\alpha=\alpha_1 t+\alpha_2(1-t)\quad\text{and}\quad\beta=\beta_1 s+\beta_2(1-s),\quad\alpha_1,\alpha_2,\beta_1,\beta_2\in(0,1).\tag{5}$$

α in Eq. (3) is defined as a linear function of t, with coefficients α_1 and α_2, and β in Eq. (4) is defined as a linear function of s, with coefficients β_1 and β_2. Applying the rational bilinear transformations (3)–(5) to surface in Eq. (1) results in a rational Bézier surface

$$\mathbf{X}(s,t)=\frac{\sum_{k_1=0}^{m+n}\sum_{k_2=0}^{m+n}B_{k_1}^{m+n}(s)B_{k_2}^{m+n}(t)\widehat{\omega}_{k_1,k_2}\mathbf{Q}_{k_1,k_2}}{\sum_{k_1=0}^{m+n}\sum_{k_2=0}^{m+n}B_{k_1}^{m+n}(s)B_{k_2}^{m+n}(t)\widehat{\omega}_{k_1,k_2}},\quad s,t\in[0,1].$$

The new surface is of degree $(m+n)\times(m+n)$ where m and n are the degrees of the original Bézier surface in the u-direction and v-direction, respectively. The control points $\mathbf{Q}_{k_1,k_2}$ and their weights $\widehat{\omega}_{k_1,k_2}$ of the reparameterized surface are as follows

$$\mathbf{Q}_{k_1,k_2}=\frac{\sum_{i=\max(k_1-m,0)}^{\min(k_1,m)}\sum_{j=\max(k_2-n,0)}^{\min(k_2,n)}c_{k_2-j,i}d_{k_1-i,j}R_{i,j}\omega_{i,j}\mathbf{P}_{i,j}}{\sum_{i=\max(k_1-m,0)}^{\min(k_1,m)}\sum_{j=\max(k_2-n,0)}^{\min(k_2,n)}c_{k_2-j,i}d_{k_1-i,j}R_{i,j}\omega_{i,j}},$$



(a) A general surface. (b) A rectangle.

Fig. 3. Two examples of a bilinear Bézier surface (where each iso-parametric curve is a straight line): (a) a general surface whose four corner points are not lying in a same plane; (b) a rectangle.

and

$$\widehat{\omega}_{k_1,k_2} = \sum_{i=\max(k_1-m,0)}^{\min(k_1,m)} \sum_{j=\max(k_2-n,0)}^{\min(k_2,n)} c_{k_2-j,i} d_{k_1-i,j} R_{i,j} \omega_{i,j},$$

where

$$R_{i,j} = \frac{\binom{m}{i}\binom{n}{j}}{\binom{m+n}{k_1}\binom{m+n}{k_2}},$$

$$c_{l,k} = \sum_{e=\max(0,k+l-m)}^{\min(k,l)} (-1)^{m-k} \binom{k}{e} \times (\alpha_1 - 1)^e (\alpha_2 - 1)^{k-e} \binom{m-k}{l-e} \alpha_1^{l-e} \alpha_2^{m-k-l+e}$$

and

$$d_{l,k} = \sum_{e=\max(0,k+l-n)}^{\min(k,l)} (-1)^{n-k} \binom{k}{e} (\beta_1 - 1)^e (\beta_2 - 1)^{k-e} \times \binom{n-k}{l-e} \beta_1^{l-e} \beta_2^{n-k-l+e}.$$

We gave a detailed deduction of the rational bilinear reparameterization for Bézier surfaces in [25], where the coefficients $c_{l,k}$ and $d_{l,k}$ of the control points are deduced only for Bézier surfaces of degree 2×2 and 3×3. Explicit representations of $c_{l,k}$ and $d_{l,k}$ for Bézier surfaces of any degree are given in [12]. It should be mentioned that the explicit representation of rational bilinear reparameterizations for rational Bézier surfaces is first given in this paper. For the rational bilinear reparameterization, each iso-parametric curve is transformed by a Möbius transformation, which is a linear interpolation of the Möbius transformations imposed on the two corresponding opposite boundary curves. Moreover, for the reparameterized surface, the control points and their weights are expressed as linear interpolations of the original control points and weights, whose linear scaling factors are determined by the rational bilinear reparameterization coefficients $\alpha_1, \alpha_2, \beta_1$ and β_2. To give a better understanding of how the parameterization changes in a rational bilinear reparameterization, an example is given in Fig. 4.

2.4. An optimization method using rational bilinear reparameterizations

Given a rational Bézier surface, this subsection presents an optimization algorithm to improve the uniformity and orthogonality of iso-parametric curves using a rational bilinear reparameterization. Let $l_1(s)$ and $l_2(t)$ denote the curve length of the s and t iso-parametric curves, respectively, i.e.,

$$l_1(s) = \int_0^1 \|\mathbf{X}_t(s,t)\| dt, \qquad l_2(t) = \int_0^1 \|\mathbf{X}_s(s,t)\| ds.$$

To measure the deviation of the current parameterization from parameterizations with uniform and orthogonal iso-parametric curves, the following integral function is adopted

$$J(\alpha_1, \alpha_2, \beta_1, \beta_2) = \lambda J_{\text{orth}} + (1-\lambda) J_{\text{unif}}, \quad \lambda \in [0,1], \tag{6}$$

where

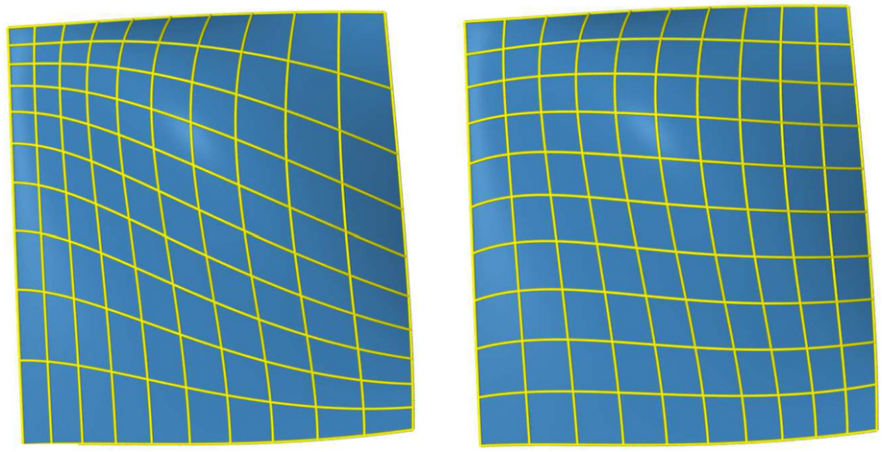
$$J_{\text{orth}} = \int_0^1 \int_0^1 F^2(s,t) ds dt,$$

$$J_{\text{unif}} = \int_0^1 \int_0^1 (E(s,t) - l_2^2(t))^2 ds dt + \int_0^1 \int_0^1 (G(s,t) - l_1^2(s))^2 dt ds,$$

and

$$E(s,t) = \mathbf{X}_s \cdot \mathbf{X}_s, \qquad F(s,t) = \mathbf{X}_s \cdot \mathbf{X}_t, \qquad G(s,t) = \mathbf{X}_t \cdot \mathbf{X}_t.$$

J_{orth} measures the deviation of the current parameterization from parameterizations with orthogonal iso-parametric curves, while J_{unif} measures the deviation of the current parameterization from parameterizations with uniform iso-parametric curves. To make both weights λ and $1-\lambda$ non-negative, λ is restricted to lie in $[0,1]$. λ represents the tradeoff between the improvements of the two desirable properties of the surface parameterization. The desirable value of λ depends on specific requirements of applications. When λ goes to zero, it will lead to a parameterization with more uniform iso-parametric curves. When λ goes to 1, it will lead to a parameterization with more orthogonal iso-parametric curves. For a general purpose, we can use $\lambda = 0.5$ trying to improve the uniformity and orthogonality of iso-parametric curves at the same time. The target function can also be written in the following



(a) The original surface. (b) The reparameterized surface.

Fig. 4. A rational bilinear reparameterization of a rational Bézier surface: (a) the original Bézier surface and its parameterization; (b) the reparameterized Bézier surface and its parameterization with the reparameterization coefficients $\alpha_1 = 0.69$, $\alpha_2 = 0.34$, $\beta_1 = 0.36$ and $\beta_2 = 0.40$.

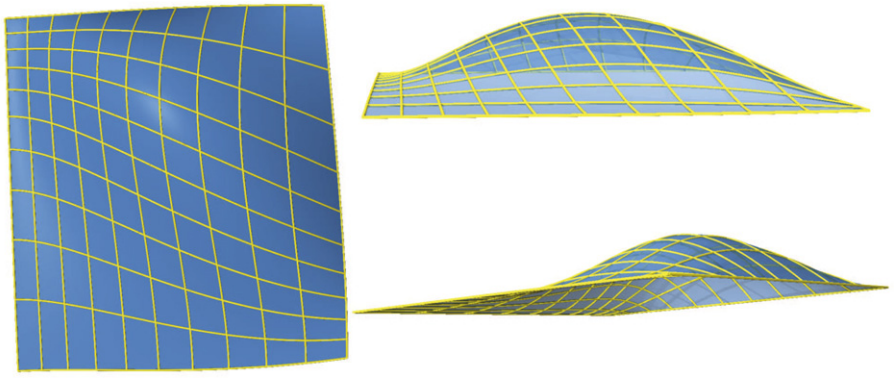


Fig. 5. Three different views of the first surface and its parameterization.

matrix form

$$J = \int_0^1 \int_0^1 \left\| \begin{bmatrix} \sqrt{1-\lambda}E(s,t) & \sqrt{\frac{\lambda}{2}}F(s,t) \\ \sqrt{\frac{\lambda}{2}}F(s,t) & \sqrt{1-\lambda}G(s,t) \end{bmatrix} - \begin{bmatrix} \sqrt{1-\lambda}l_2^2(t) & 0 \\ 0 & \sqrt{1-\lambda}l_1^2(s) \end{bmatrix} \right\|^2 dsdt,$$

where $\|\cdot\|$ is the Frobenius norm. In other words, J measures the difference between the weighted first fundamental form matrix of $\mathbf{X}(s,t)$ and the weighted first fundamental form matrix of the surface parameterization with uniform and orthogonal iso-parametric curves. In general, Eq. (6) is highly nonlinear, and there is no closed-form solution for α_1, α_2, β_1 and β_2. Thus we solve this optimization problem using the following discrete method. First we sample a grid of $(\iota+1)\times(\kappa+1)$ points at $\mathbf{X}(s_i,t_j)$, $i = 0,\ldots,\iota$ and $j = 0,\ldots,\kappa$. After the coefficients of the rational bilinear reparameterization are determined, the corresponding parameter values u_i and v_j in the parameter domain of the original surface can be computed from Eqs. (3) and (4) directly. We introduce auxiliary variables $\mathbf{e} = \{e_0,\ldots,e_\kappa\}$ and $\mathbf{g} = \{g_0,\ldots,g_\iota\}$ to represent the squares of the length of specific iso-parametric curves, i.e.,

$$e_j = l_2^2(t_j), \quad j = 0,\ldots,\kappa,$$
$$g_i = l_1^2(s_i), \quad i = 0,\ldots,\iota.$$

The energy function in Eq. (6) is then discretized as

$$\widetilde{J}(\alpha_1,\alpha_2,\beta_1,\beta_2,\mathbf{e},\mathbf{g}) = \lambda\widetilde{J}_{\text{orth}} + (1-\lambda)\widetilde{J}_{\text{unif}}, \quad \lambda\in[0,1], \tag{7}$$

where

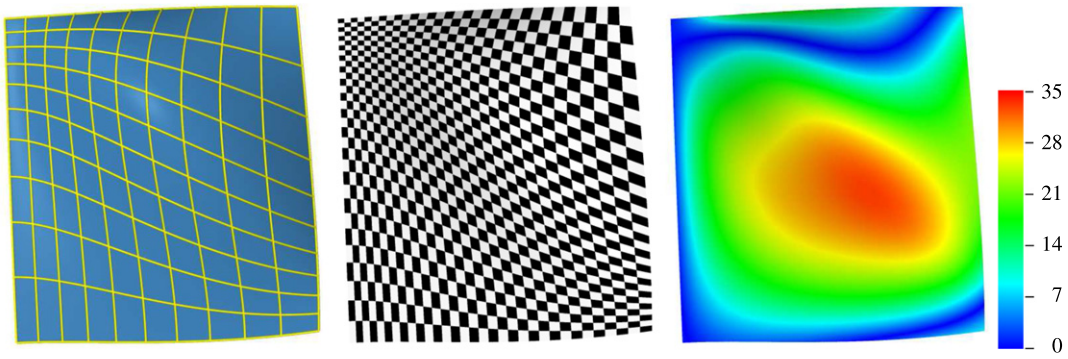
$$\widetilde{J}_{\text{orth}} = \sum_{i=0}^{\iota}\sum_{j=0}^{\kappa} F^2(s_i,t_j),$$
$$\widetilde{J}_{\text{unif}} = \sum_{i=0}^{\iota}\sum_{j=0}^{\kappa}(E(s_i,t_j)-e_j)^2 + \sum_{i=0}^{\iota}\sum_{j=0}^{\kappa}(G(s_i,t_j)-g_i)^2.$$

This leads to the following optimization problem

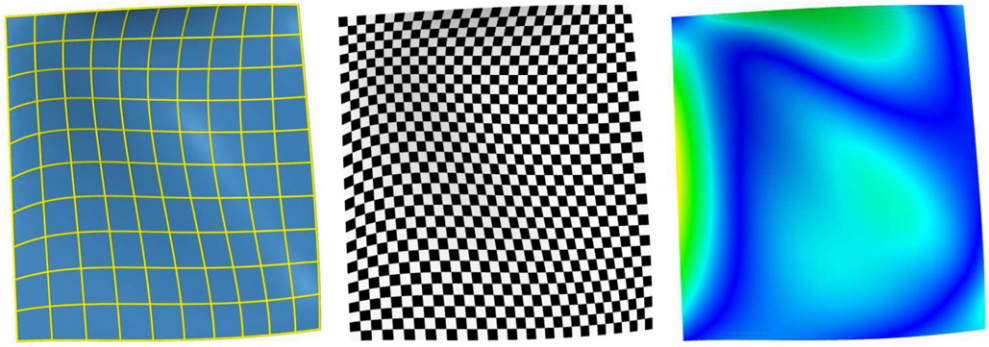
$$\min_{\alpha_1,\alpha_2,\beta_1,\beta_2,\mathbf{e},\mathbf{g}} \widetilde{J} \quad \text{s.t. } 0<\alpha_1,\alpha_2,\beta_1,\beta_2<1, \tag{8}$$

which is a constrained nonlinear least squares problem. This target function can be minimized numerically using the Levenberg–Marquardt method [29], which enjoys robust global convergence property to a local minimum. Each iteration of the Levenberg–Marquardt method needs to evaluate the gradient of functions $F(s_i,t_j)$, $E(s_i,t_j)-e_j$, $G(s_i,t_j)-g_i$ with respect to variables $\alpha_1,\alpha_2,\beta_1,\beta_2,\mathbf{e},\mathbf{g}$. The most involved part of the gradient evaluation is to compute the derivatives of $F(s_i,t_j)$, $E(s_i,t_j)$ and $G(s_i,t_j)$ with respect to $\alpha_1,\alpha_2,\beta_1,\beta_2$. These derivatives can be computed as follows. According to the chain rule, the partial derivatives $\mathbf{X}_s$ and $\mathbf{X}_t$ are

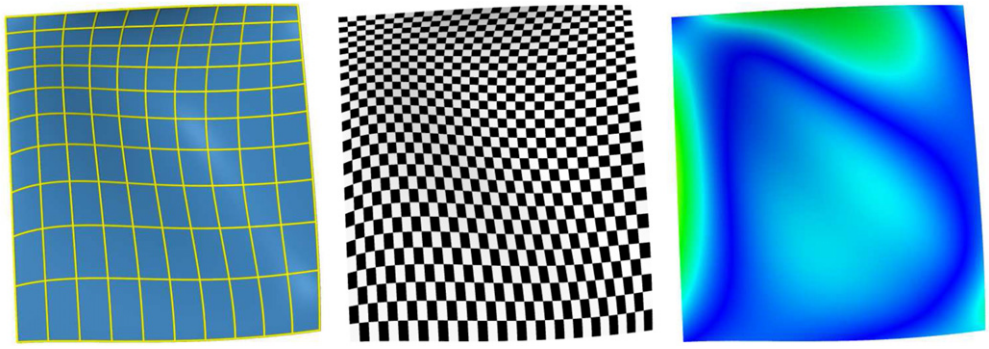
$$\begin{aligned}\mathbf{X}_s &= \mathbf{X}_u\cdot u_s + \mathbf{X}_v\cdot v_s\\ \mathbf{X}_t &= \mathbf{X}_u\cdot u_t + \mathbf{X}_v\cdot v_t.\end{aligned} \tag{9}$$



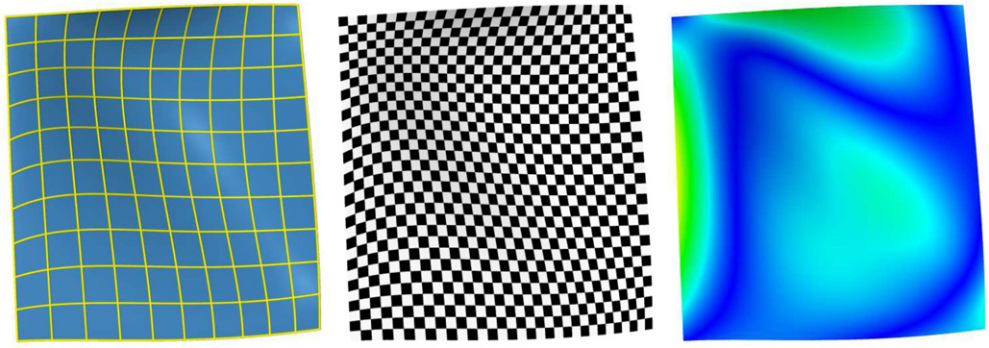
(a) The original surface.



(b) The resulting surface optimized for better uniformity ($\lambda = 0$).



(c) The resulting surface optimized for better orthogonality ($\lambda = 1$).



(d) The resulting surface optimized for better uniformity and orthogonality ($\lambda = 0.5$).

Fig. 6. Optimization for a rational Bézier surface of degree 3×3.

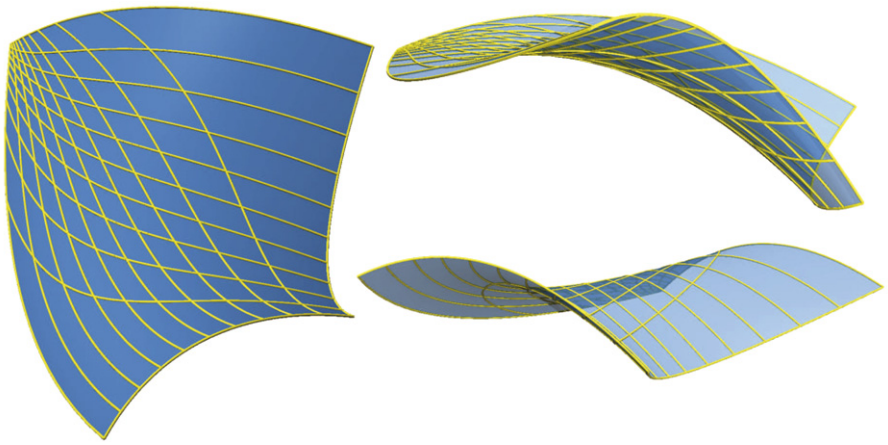


Fig. 7. Three different views of the second surface and its parameterization.

Let

$$\widehat{\mathbf{I}} = \begin{bmatrix} \widehat{E} & \widehat{F} \\ \widehat{F} & \widehat{G} \end{bmatrix}, \qquad \mathbf{S} = \begin{bmatrix} u_s \\ v_s \end{bmatrix} \quad \text{and} \quad \mathbf{T} = \begin{bmatrix} u_t \\ v_t \end{bmatrix}, \tag{10}$$

where

$$\widehat{E} = \mathbf{X}_u \cdot \mathbf{X}_u, \qquad \widehat{F} = \mathbf{X}_u \cdot \mathbf{X}_v, \qquad \widehat{G} = \mathbf{X}_v \cdot \mathbf{X}_v.$$

Then the first fundamental form terms E, F and G can be expressed as

$$E = \mathbf{S}^T \widehat{\mathbf{I}} \mathbf{S}, \qquad F = \mathbf{S}^T \widehat{\mathbf{I}} \mathbf{T}, \qquad G = \mathbf{T}^T \widehat{\mathbf{I}} \mathbf{T}.$$

For a variable $\gamma \in \{\alpha_1, \alpha_2, \beta_1, \beta_2\}$, we can compute the derivatives of E, F, G with respect to γ as

$$\begin{aligned} \frac{\partial E}{\partial \gamma} &= 2\mathbf{S}_\gamma^T \widehat{\mathbf{I}} \mathbf{S} + \mathbf{S}^T \widehat{\mathbf{I}}_\gamma \mathbf{S}, \\ \frac{\partial F}{\partial \gamma} &= \mathbf{S}_\gamma^T \widehat{\mathbf{I}} \mathbf{T} + \mathbf{S}^T \widehat{\mathbf{I}} \mathbf{T}_\gamma + \mathbf{S}^T \widehat{\mathbf{I}}_\gamma \mathbf{T}, \\ \frac{\partial G}{\partial \gamma} &= 2\mathbf{T}_\gamma^T \widehat{\mathbf{I}} \mathbf{T} + \mathbf{T}^T \widehat{\mathbf{I}}_\gamma \mathbf{T}, \end{aligned} \tag{11}$$

where

$$\widehat{\mathbf{I}}_\gamma = \begin{bmatrix} \dfrac{\partial \widehat{E}}{\partial \gamma} & \dfrac{\partial \widehat{F}}{\partial \gamma} \\ \dfrac{\partial \widehat{F}}{\partial \gamma} & \dfrac{\partial \widehat{G}}{\partial \gamma} \end{bmatrix}, \qquad \mathbf{S}_\gamma = \begin{bmatrix} \dfrac{\partial u_s}{\partial \gamma} \\ \dfrac{\partial v_s}{\partial \gamma} \end{bmatrix} \quad \text{and}$$

$$\mathbf{T}_\gamma = \begin{bmatrix} \dfrac{\partial u_t}{\partial \gamma} \\ \dfrac{\partial v_t}{\partial \gamma} \end{bmatrix}.$$

Matrices $\mathbf{S}_\gamma$, $\mathbf{T}_\gamma$ can be obtained easily from Eqs. (3) and (4) of the rational bilinear reparameterization. The entries of matrix $\widehat{\mathbf{I}}_\gamma$ can be computed as

$$\begin{aligned} \frac{\partial \widehat{E}}{\partial \gamma} &= 2\mathbf{X}_u \cdot \frac{\partial \mathbf{X}_u}{\partial \gamma}, \\ \frac{\partial \widehat{F}}{\partial \gamma} &= \frac{\partial \mathbf{X}_u}{\partial \gamma} \cdot \mathbf{X}_v + \mathbf{X}_u \cdot \frac{\partial \mathbf{X}_v}{\partial \gamma}, \\ \frac{\partial \widehat{G}}{\partial \gamma} &= 2\mathbf{X}_v \cdot \frac{\partial \mathbf{X}_v}{\partial \gamma}. \end{aligned}$$

The gradient of the target function $\widetilde{J}$ can be obtained from Eqs. (7) and (11). For numerical optimization, it is advisable to start from an initial point which is not far away from a local minimum. For initial values of coefficients $\alpha_1, \alpha_2, \beta_1, \beta_2$, we sample the coefficient space

$$\begin{aligned} &(\alpha_1 = i/\eta, \alpha_2 = j/\eta, \beta_1 = k/\eta, \beta_2 = l/\eta), \\ &i, j, k, l = 1, \ldots, \eta - 1, \end{aligned} \tag{12}$$

uniformly and select the coefficients with the minimal energy. To handle the bound constraints in (8), we solve the minimization problem (8) using the levmar package [30], which implements the Levenberg–Marquardt algorithm with bound constraints handled according to [31].

The Levenberg–Marquardt algorithm is a popular method for nonlinear least squares optimization. It is more robust than the Gauss–Newton algorithm, and is guaranteed to converge to a local minimum [29]. There is no guarantee that it will converge to a global minimum. To find the global minima, we need to use global optimization algorithms, which require a global search approach and are usually time-consuming [32]. Instead, our method selects a good initial point for the Levenberg–Marquardt algorithm, so that

Table 1
Optimization results for the rational Bézier surface given in Fig. 6.

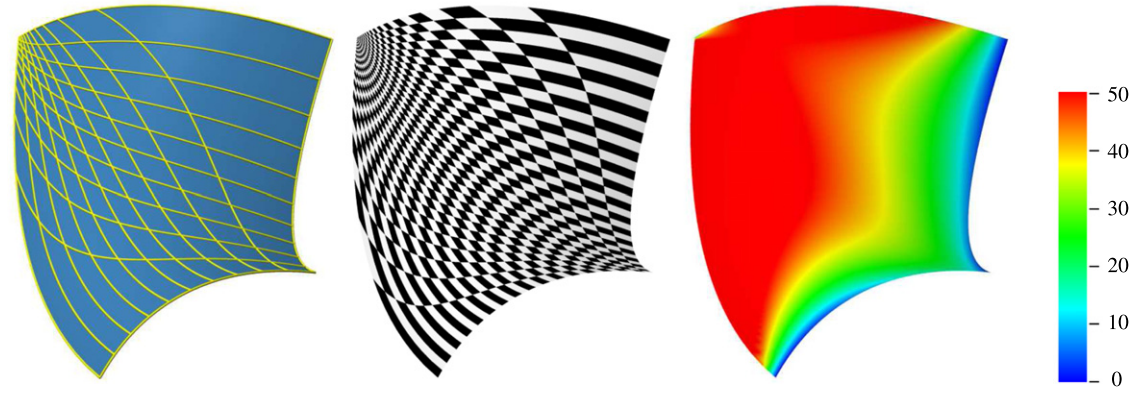
	Original surface	$\lambda = 0$	$\lambda = 1$	$\lambda = 0.5$
α_1	–	0.69	0.81	0.69
α_2	–	0.29	0.45	0.30
β_1	–	0.38	0.41	0.38
β_2	–	0.41	0.41	0.40
Orthogonality energy	25.17	4.21	3.13	4.05
Uniformity energy	241.85	9.22	249.99	9.29

Table 2
Optimization results for the surface in Fig. 8.

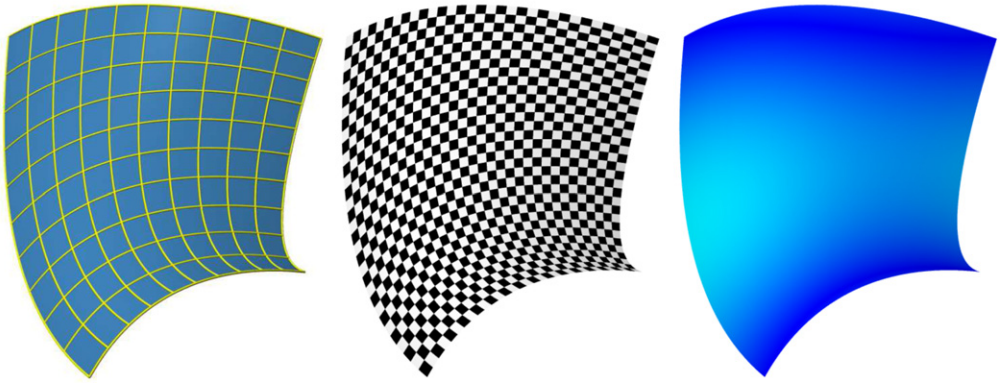
	Original surface	$\lambda = 0$	$\lambda = 1$	$\lambda = 0.5$
α_1	–	0.04	0.05	0.04
α_2	–	0.43	0.5	0.45
β_1	–	0.36	0.5	0.42
β_2	–	0.96	0.95	0.95
Orthogonality energy	780	1.36	1.5×10^{-11}	1.03
Uniformity energy	1.6×10^6	3.98	113	4.25

Table 3
Optimization results for the surface in Fig. 10.

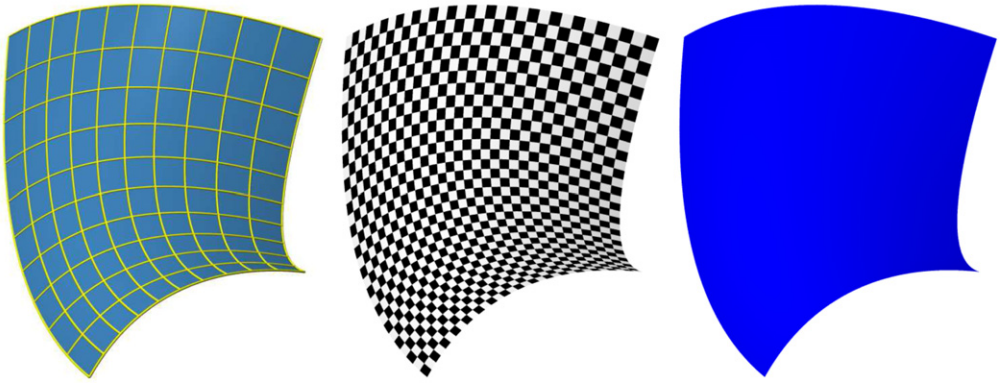
	Original surface	$\lambda = 0$	$\lambda = 1$	$\lambda = 0.5$
α_1	–	0.57	0.27	0.55
α_2	–	0.54	0.74	0.56
β_1	–	0.38	0.52	0.35
β_2	–	0.77	0.58	0.81
Orthogonality energy	226.34	127.38	63.19	107.21
Uniformity energy	316.46	23.90	1908	30.99



(a) The original surface.



(b) The resulting surface optimized for better uniformity ($\lambda = 0$).



(c) The resulting surface optimized for better orthogonality ($\lambda = 1$).

Fig. 8. Optimization for a rational Bézier surface of degree 4×4.

it can converge to a local minimum nearby. Such a strategy has been successfully applied to other geometric modeling problems involving nonlinear least squares optimization [33,34]. In our method, the quality of the initial point can be improved by increasing the sample density factor η in Eq. (12). We suggest using $\eta \geq 10$ for a good optimization result.

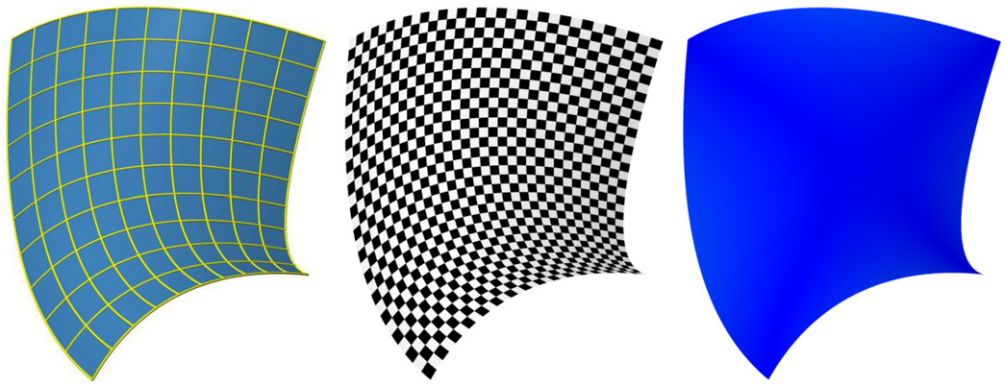
3. Experimental results

Our algorithm is implemented on a PC with an Intel 3.06 GHZ CPU, 2G Memory and Microsoft Visual Studio 2008. To show the performance of our algorithm, some examples are given below. In all examples, we perform the optimization using 11×11 sample points, and three different orthogonality weights $\lambda = 0, 1, 0.5$.

Fig. 6 illustrates the optimization results of a rational Bézier surface of degree 3×3 shown in Fig. 5. To compare different results, each surface is illustrated using three methods: iso-parametric curves, texture mapping, and color-coding of angles between partial derivative vectors. In the color-coding images, we measure for a parameterization $\mathbf{X}(\delta, \zeta)$ the angle (in degrees) between the partial derivative vectors $\mathbf{X}_\delta$ and $\mathbf{X}_\zeta$, and use colors to illustrate the absolute value of the difference between this angle and 90°. In other words, the color-coding images illustrate the following function across the surface

$$\theta(\delta, \zeta) = \left| \arccos\left(\frac{\mathbf{X}_\delta \cdot \mathbf{X}_\zeta}{\|\mathbf{X}_\delta\| \, \|\mathbf{X}_\zeta\|} \right) \cdot \frac{180}{\pi} - 90 \right|$$

iso-parametric curves of surface $\mathbf{X}(\delta, \zeta)$ are orthogonal at a point (δ, ζ) if and only if $\theta(\delta, \zeta) = 0$.



(d) The resulting surface optimized for better uniformity and orthogonality ($\lambda = 0.5$).

Fig. 8. (*continued*)

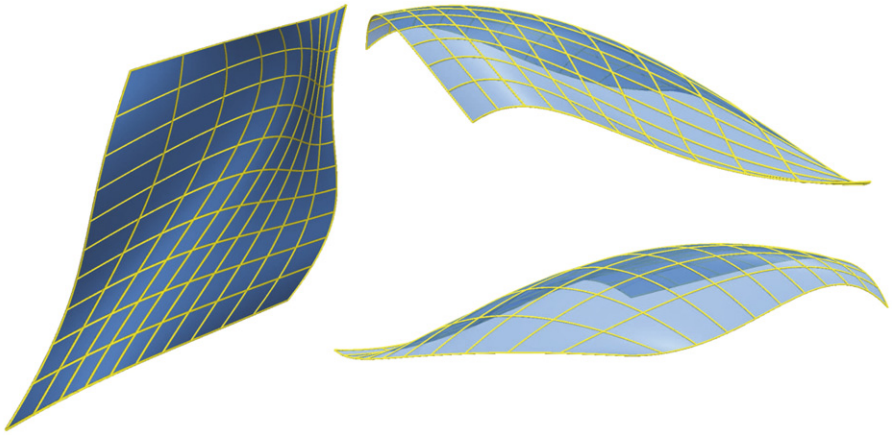


Fig. 9. Three different views of the third surface and its parameterization.

As shown in Fig. 6, iso-parametric curves of the original surface parameterization are neither uniform nor orthogonal. With $\lambda = 0$, only the uniformity property of the surface parameterization is considered in the optimization. Iso-parametric curves of the optimized surface are more uniform than the original ones. With $\lambda = 1$, only the orthogonality of the iso-parametric curves is considered in the optimization. The color-coding image shows that the new iso-parametric curves are more orthogonal than the original ones. Finally with $\lambda = 0.5$, both the uniformity and orthogonality of iso-parametric curves are considered in the optimization. We obtain a parameterization where both the uniformity and orthogonality are improved over the original one. The rational bilinear reparameterization coefficients and the energies are given in Table 1.

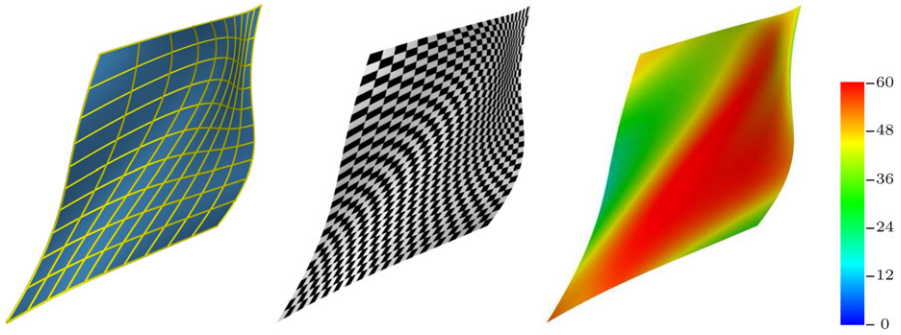
Another example is given in Fig. 8, where we optimize a rational Bézier surface shown in Fig. 7. The input surface is obtained by reparameterizing a Bézier surface where the original iso-parametric curves are orthogonal. The rational bilinear reparameterization coefficients and the corresponding energies are given in Table 2. The orthogonality energy values indicate that we recover the orthogonality of the iso-parametric curves with $\lambda = 1$. Even when $\lambda = 0$, we significantly improve the orthogonality.

In Fig. 10, a rational Bézier surface of degree 3×3 shown in Fig. 9 is optimized. The rational bilinear reparameterization coefficients and the energies are given in Table 3. Note that we cannot improve the orthogonality of iso-parametric curves at the four corner points. This is because our rational bilinear reparameterization always maps the iso-parametric lines $s = 0$, $s = 1$, $t = 0$ and $t = 1$ to the boundary curves of the surface. So the angles between partial derivative vectors at the corner points $\mathbf{X}(0, 0)$, $\mathbf{X}(0, 1)$, $\mathbf{X}(1, 0)$, $\mathbf{X}(1, 1)$ will remain unchanged.

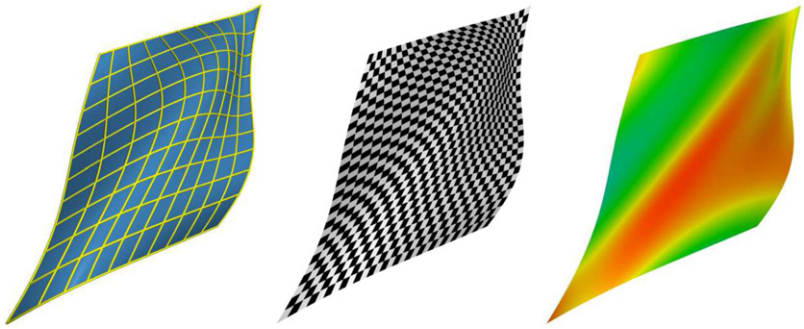
The above examples show that our method can improve the surface parameterization according to the given weights. For some cases, the optimizations for the uniformity and orthogonality coincide (orthogonality is improved even if $\lambda = 0$). For some other cases, the optimization for these two properties conflicts with each other, and the user should determine the orthogonality weight more carefully. Sometimes, the parameterization can only be improved to a limited degree. The reason is as follows. First of all, with rational bilinear reparameterizations, the additional degrees of freedom that we gain may still be not enough. Besides, we cannot improve the orthogonality of iso-parametric curves at the four corner points. In order to achieve better results, we may compute another rational Bézier surface patch $\overline{\mathbf{X}}$ which has the desirable parameterization, and contains the original surface $\mathbf{X}$. Then the reparameterization of $\mathbf{X}$ can be obtained by trimming the larger patch $\overline{\mathbf{X}}$ using the boundary curves of $\mathbf{X}$ (see Fig. 11).

4. Conclusions and future work

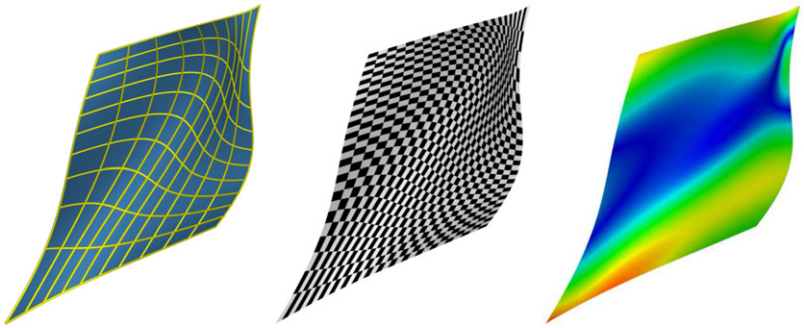
In this paper, we conclude that the only rational Bézier surfaces with uniform iso-parametric curves are bilinear surfaces, and the only rational Bézier surfaces with uniform and orthogonal iso-parametric curves are rectangles. Moreover, to improve the uniformity and orthogonality of iso-parametric curves for general rational Bézier surfaces, an optimization algorithm using the rational bilinear reparameterizations is presented. The coefficients



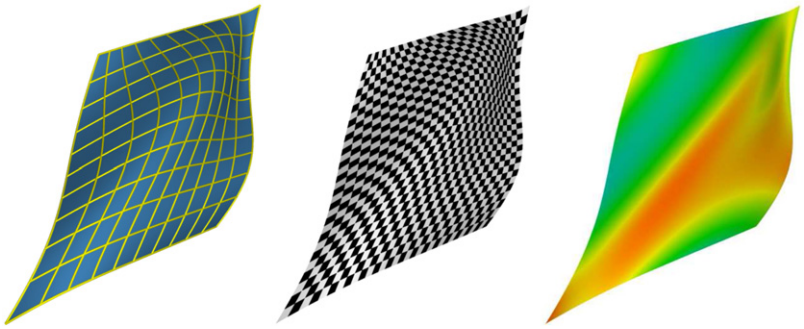
(a) The original surface.



(b) The resulting surface optimized for better uniformity ($\lambda = 0$).



(c) The resulting surface optimized for better orthogonality ($\lambda = 1$).



(d) The resulting surface optimized for better uniformity and orthogonality ($\lambda = 0.5$).

Fig. 10. Optimization for a rational Bézier surface of degree 3×3.

of the rational bilinear reparameterization are obtained by minimizing a discrete nonlinear energy measuring the deviation of the current parameterization from parameterizations with uniform and orthogonal iso-parametric curves. The examples indicate that, in practice, the algorithm produces significantly more uniform and orthogonal iso-parametric curves across the rational Bézier surfaces, which is indispensable for many CAD applications such as rendering, tessellation and surface blending.

The rational bilinear reparameterizations presented in this paper can be applied to the whole NURBS surface and its parametric continuity is preserved automatically. A more challenging and meaningful task is to reparameterize each patch of a NURBS surface with different rational bilinear transformations and still preserve the parametric continuity, which could produce better results than reparameterizations with only one rational bilinear transformation. This is left as our future work.

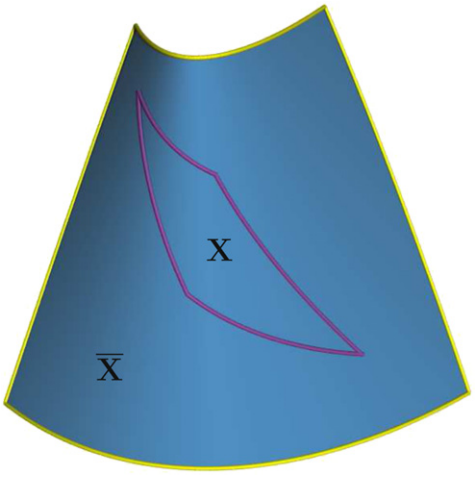


Fig. 11. A rational Bézier surface **X** (with purple boundary curves) contained in another rational Bézier surface $\overline{\mathbf{X}}$ (with yellow boundary curves). The reparameterization of **X** can be done by trimming a parameterization of $\overline{\mathbf{X}}$ using the boundary curves of **X**. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Acknowledgments

The authors thank the anonymous reviewers for their valuable suggestions. This work was supported by the China national natural science foundation (61202146, 61272243, 61070093 and U1035004), Shandong province outstanding young scientist research award fund (BS2012DX014 and BS2009DX026), independent innovation foundation of Shandong university, IIFSDU (11150072614024) and the ERC starting grant 257453 COSYM.

References

[1] Cheung G, Lau R, Li F. Incremental rendering of deforming NURBS surfaces. In: Proceedings of the ACM symposium on virtual reality software and technology. VRST 03, New York, NY, USA: ACM Press; 2003. p. 48–55.
[2] Chuang JH, Lin CH, Hwang WC. Variable-radius blending of parametric surfaces. The Visual Computer 1995;11(10):513–25.
[3] Chuang JH, Hwang WC. Variable-radius blending by constrained spine generation. The Visual Computer 1997;13(7):316–29.
[4] Hamann B, Tsai PY. A tessellation algorithm for the representation of trimmed nurbs surfaces with arbitrary trimming curves. Computer-Aided Design 1996; 28(6–7):461–72.
[5] Kumar S, Manocha D. Efficient rendering of trimmed NURBS surfaces. Computer-Aided Design 1995;27(7):509–21.
[6] Ng WMM, Tan ST. Incremental tessellation of trimmed parametric surfaces. Computer-Aided Design 2000;32(4):279–94.
[7] Piegl LA, Richard AM. Tessellating trimmed NURBS surfaces. Computer-Aided Design 1995;27(1):15–26.
[8] Piegl LA, Tiller W. The NURBS book. 2nd ed. New York: Springer; 1997.
[9] Piegl LA, Tiller W. Filling *n*-sided regions with NURBS patches. The Visual Computer 1999;15(2):77–89.
[10] William LL. Tessellation of trimmed NURB surfaces. Computer Aided Geometric Design 1996;13(2):163–77.
[11] Yang YJ, Yong JH, Zhang H, Paul JC, Sun JG. A rational extension of Piegl's method for filling *n*-sided holes. Computer-Aided Design 2006;38(11): 1166–78.
[12] Yang YJ, Zeng W, Zhang H, Paul JC, Yong JH. Projection of curves on *B*-spline surfaces using quadratic reparameterization. Journal of Graphical Models 2010;72(5):47–59.
[13] Yang YJ, Zeng W, Yang CL, Meng XX, Yong JH, Deng BL. G1 continuous approximate curves on NURBS surfaces. Computer-Aided Design 2012;44(9): 824–34.
[14] Costantini P, Farouki RT, Manni C, Sestini A. Computation of optimal composite re-parameterizations. Computer Aided Geometric Design 2001;18(9):875–97.
[15] Farouki RT. Optimal parameterizations. Computer Aided Geometric Design 1997;14(2):153–68.
[16] Jüttler B. A vegetarian approach to optimal parameterizations. Computer Aided Geometric Design 1997;14(9):887–90.
[17] Ong BH. An extraction of almost arc-length parameterization from parametric curves. Annals of Numerical Mathematics 1996;3:305–16.
[18] Shpitalni M, Koren Y, Lo CC. Realtime curve interpolators. Computer-Aided Design 1994;26(11):832–8.
[19] Wang FC, Yang DCH. Nearly arc-length parameterized quintic spline interpolation for precision machining. Computer-Aided Design 1993;25(5): 281–8.
[20] Wang FC, Wright PK, Barsky BA, Yang DCH. Approximately arc-length parameterized C^3 quintic interpolatory splines. ASME Journal of Mechanical Design 1999;121:430–9.
[21] Wever U. Optimal parameterization for cubic spline. Computer-Aided Design 1991;23(9):641–4.
[22] Yang DCH, Kong T. Parametric interpolator versus linear interpolator for precision CNC machining. Computer-Aided Design 1994;26(3):225–34.
[23] Yang DCH, Wang FC. A quintic spline interpolator for motion command generation of computer-controlled machines. ASME Journal of Mechanical Design 1994;116:226–31.
[24] Yeh S-S, Hsu P-L. The speed-controlled interpolator for machining parametric curves. Computer-Aided Design 1999;31(5):349–57.
[25] Yang YJ, Yong JH, Zhang H, Paul JC, Sun JG. Optimal parameterizations of Bézier surfaces. In: 2nd international symposium on visual computing. ISVC06. 2006. p. 672–81.
[26] Farin G. NURBS from projective geometry to practical use. 2nd ed. A.K. Peters; 1999.
[27] Do Carmo MP. Differential geometry of curves and surfaces. Prentice Hall; 1976.
[28] Farouki RT, Sakkalis T. Real rational curves are not 'unit speed'. Computer Aided Geometric Design 1991;8(2):151–7.
[29] Nocedal J, Wright SJ. Numerical optimization. 2nd ed. Springer; 2006.
[30] Lourakis M. levmar: Levenberg–Marquardt nonlinear least squares algorithms in C/C++, http://www.ics.forth.gr/~lourakis/levmar/.
[31] Kanzow C, Yamashita N, Fukushima M. Levenberg–Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints. Journal of Computational and Applied Mathematics 2004; 173(2):321–43.
[32] Horst R, Pardalos PM, Thoai NV. Introduction to global optimization. 2nd ed. Springer; 2000.
[33] Deng B, Pottmann H, Wallner J. Functional webs for freeform architecture. Computer Graphics Forum 2011;30(5):1369–78.
[34] Liu Y, Pottmann H, Wallner J, Yang Y-L, Wang W. Geometric modeling with conical meshes and developable surfaces. ACM Transactions on Graphics 2006; 25(3):681–9.