Computer Aided Geometric Design ••• (••••) •••–•••

Contents lists available at ScienceDirect

Computer Aided Geometric Design

www.elsevier.com/locate/cagd

Matrix weighted rational curves and surfaces ☆

Xunnian Yang

School of Mathematical Sciences, Zhejiang University, Hangzhou 310027, China

ARTICLE INFO

Article history:
Received 18 January 2015
Received in revised form 20 July 2015
Accepted 24 November 2015
Available online xxxx

Keywords:
Rational curves and surfaces
Matrix weight
Control normals
Surface reconstruction

ABSTRACT

Rational curves and surfaces are powerful tools for shape representation and geometric modeling. However, the real weights are generally difficult to choose except for a few special cases such as representing conics. This paper presents an extension of rational curves and surfaces by replacing the real weights with matrices. The matrix weighted rational curves and surfaces have the same structures as the traditional rational curves and surfaces but the matrix weights can be defined in geometric ways. In particular, the weight matrices for the extended rational Bézier, NURBS or the generalized subdivision curves and surfaces are computed using the normal vectors specified at the control points. Similar to the effects of control points, the specified normals can be used to control the curve or the surface's shape efficiently. It is also shown that matrix weighted NURBS curves and surfaces can pass through their control points, thus curve or surface reconstruction by the extended NURBS model no longer needs solving any large system but just choosing control points and control normals from the input data.

1. Introduction

In Computer Aided Geometric Design, rational curves and surfaces like

$$R(\xi) = \frac{\sum_i \omega_i P_i \phi_i(\xi)}{\sum_i \omega_i \phi_i(\xi)}, \quad \xi \in \Xi \tag{1}$$

where P_i are the control points, ω_i are the scalar weights and $\phi_i(\xi)$ are the blending functions defined on a 1D or 2D domain, are powerful tools for shape representation, modeling and reconstruction. Particularly, polynomial or rational Bézier curves and surfaces, NURBS (non-uniform rational B-spline) curves and surfaces and the generalized subdivision surfaces are widely used in CAD or computer animation industry (Farin, 2001; DeRose et al., 1998; Müller et al., 2006). As its compact representation of freeform as well as analytical curves and surfaces, NURBS has even become de facto industry standard in CAD commercial software (Piegl and Tiller, 1997).

Except for representing typical curves or surfaces such as conics or rotational surfaces, the potentials of rational curves and surfaces have not been explored thoroughly for geometric modeling. One possible reason is due to the limitations of the current representation. Generally, the geometric meaning of weights of rational curves and surfaces is not as clear as their control points. High order rational curves and surfaces are useful for high quality shape modeling (Cashman et al., 2009), but how to construct them is not as clear as desired. Rational curves and surfaces are also promising for shape reconstruction from scanned data but the determination of the weights is not a simple task (Xie et al., 2012). If some

☆ This paper has been recommended for acceptance by Ron Goldman.

E-mail address: yxn@zju.edu.cn.

http://dx.doi.org/10.1016/j.cagd.2015.11.005
0167-8396/

differential quantities or sharp features should be considered, the reconstruction of curves and surfaces becomes nonlinear and optimization techniques have to be employed (Gofuku et al., 2009; Wu et al., 2013; Ma et al., 2015).

A geometric approach to construct rational curves and surfaces is the technique of dual Bézier curves and surfaces which treat the Bézier control points as control lines or control planes (Hoschek, 1983). Rational Bézier curves and surfaces are obtained as the envelopes of the dual Bézier curves or the dual Bézier surfaces. Even this is usually feasible, the denominator of an obtained rational Bézier curve or surface may vanish when the initial control lines or control planes have not been properly given. Another interesting extension of rational Bézier curves is complex rational Bézier curves (Sánchez-Reyes, 2009). Complex rational Bézier curves replace the scalar weights of the classical rational Bézier curves by complex numbers and several typical plane curves can be represented by complex rational Bézier curves of lower degrees. However, complex rational Bézier curves only lie in 2D space. Other interesting extensions of Bézier or NURBS curves and surfaces include plus curves and surfaces which use parameterized control lines (Goshtasby, 2005), curves and surfaces with basis functions that contain shape parameters (Juhász and Róth, 2013) or spline curves and surfaces with added geometric details controlled by vectors (Kosinka et al., 2015).

In this paper we present extended representation models of rational curves and surfaces with the following goals: (1) The new models maintain the same structures as the usual ones; (2) The new models are compatible with the traditional rational curves and surfaces; (3) The new weights permit geometric definitions and can be used for easy shape editing; (4) Curve or surface reconstruction using the extended rational models is simple and direct.

A rational curve or surface in $\mathbb{R}^d$ can be regarded as the perspective projection of a non-rational curve or surface in $\mathbb{R}^{d+1}$ (Farin, 1999). Similarly, we define rational curves and surfaces in $\mathbb{R}^d$ by the mapping of homogeneous curves or homogeneous surfaces in $\mathbb{R}^{(d+1)\times d}$. Concretely, we should only replace the scalar weights in Equation (1) with $d \times d$ matrices. If the denominator matrix function is non-singular over the whole parameter domain, a matrix weighted rational curve or surface will be obtained. It is clear that the extended rational curves and surfaces have the same structures as the traditional ones. If all the weight matrices are defined by multiplying a set of real numbers with the same non-singular matrix, the matrix weighted rational curve or surface will degenerate to a conventional rational curve or surface.

To achieve the third or even the fourth goal, we propose to compute the weight matrices based on normal vectors and real numbers specified at the control points of the curves or the surfaces. This is motivated by the observation that a rational curve or surface is in fact the solution to least-squares fitting to the control points with weights given by a set of basis functions. If we fit a curve or a surface to the points as well as to a set of lines or planes that pass through the points, we obtain a rational curve or a rational surface with matrix weights. These curves and surfaces can be controlled using control points together with control normals. Moreover, matrix weighted NURBS curves and surfaces can even pass through their control points, thus curve or surface reconstruction by the proposed models becomes easy because the control points of the reconstructed curve or surface should only be selected from the input data.

As the popular subdivision schemes proposed by Catmull and Clark (1978), Doo and Sabin (1978) or Loop (1987) are just the generalized B-spline surfaces with arbitrary topology control meshes, they can be extended to matrix weighted rational subdivision surfaces and the shapes of the extended subdivision surfaces can be controlled using control points and control normals. Similarly, other B-spline or NURBS compatible subdivision surfaces, for example the schemes in Sederberg et al. (1998), Stam (2001), Schaefer and Goldman (2009), can be generalized to matrix weighted rational subdivision surfaces too.

The paper is structured as follows. Section 2 presents the general definition and basic properties of matrix weighted rational curves and surfaces. More details about matrix weighted rational Bézier curves and matrix weighted NURBS curves are given in Section 3 and Section 4, respectively. Section 5 is devoted to matrix weighted rational parametric surfaces and matrix weighted rational subdivision surfaces. In Section 6 we present several interesting examples to show the potential applications of the proposed models. Section 7 concludes the paper.

2. Basics of matrix weighted rational curves and surfaces

In this section we present the definition of general matrix weighted rational curves and surfaces. A method for defining the weight matrices using specified normals and some basic properties of the obtained curves and surfaces will also be given.

2.1. Definition

Assume that $P_0, P_1, \ldots, P_n$ are a sequence or a set of points lying in $\mathbb{R}^d$ and $M_i \in \mathbb{R}^{d\times d}$, $i = 0, 1, \ldots, n$ are a set of known matrices. Suppose that $\phi_i(t) \geq 0$, $i = 0, 1, \ldots, n$ are a set of blending functions defined over a 1D or 2D domain Ξ. If the matrix function $M(\xi) = \sum_{i=0}^{n} M_i\phi_i(\xi)$ is non-singular for any $\xi \in \Xi$, a *valid* matrix weighted rational curve or surface is given by

$$Q(\xi) = \left[\sum_{i=0}^{n} M_i\phi_i(\xi)\right]^{-1} \sum_{i=0}^{n} M_i P_i\phi_i(\xi), \quad \xi \in \Xi \tag{2}$$

Particularly, if we choose $M_i = \omega_i W$, where $\omega_i \in \mathbb{R}$, $i = 0, 1, \ldots, n$ and W is a $d \times d$ matrix that satisfies $|W| \neq 0$, the matrix weighted rational curve or surface $Q(\xi)$ will degenerate to a conventional rational curve or surface as given by Equation (1).

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

Though there may exist many methods for defining the weight matrices for matrix weighted rational curves and surfaces, we present here a geometric approach to compute them. Suppose that $\mathbf{n}_i$, $i = 0, 1, \ldots, n$ are a set of unit normal vectors and $\omega_i > 0$, $\mu_i > -1$, $i = 0, 1, \ldots, n$ are a set of real numbers. The weight matrices for $Q(\xi)$ are defined as follows

$$M_i = \omega_i(I + \mu_i \mathbf{n}_i \mathbf{n}_i^T), \ i = 0, 1, \ldots, n \tag{3}$$

where I is the $d \times d$ identity matrix. Before we derive the condition when a matrix weighted rational curve or surface is valid, we have the following theorem.

Theorem 2.1. *Suppose that $\mathbf{n}_i \in \mathbb{R}^d$, $i = 0, 1, \ldots, n$ are a set of unit vectors, $\omega_i > 0$, $\mu_i > -1$, $i = 0, 1, \ldots, n$ are real numbers and M_i are the matrices defined by Equation (3). The matrix $M_s = \sum_{i=0}^{n} s_i M_i$, where the coefficients satisfy $s_i \geq 0$ and $\sum_{i=0}^{n} s_i > 0$, is non-singular.*

Proof. To prove the theorem we prove that the determinant of matrix M_s is greater than zero. Let $\theta_i = \omega_i s_i$, the matrix M_s can be reformulated as $M_s = \sum_{i=0}^{n} \theta_i(I + \mu_i \mathbf{n}_i \mathbf{n}_i^T)$. Assuming that λ is an arbitrary eigenvalue and $\mathbf{v}$ is the corresponding eigenvector of the matrix, it yields that

$$\lambda \mathbf{v} = \left[\sum_{i=0}^{n} \theta_i(I + \mu_i \mathbf{n}_i \mathbf{n}_i^T)\right]\mathbf{v} = \sum_{i=0}^{n} \theta_i(\mathbf{v} + \mu_i a_i \mathbf{n}_i),$$

where $a_i = \mathbf{n}_i^T \mathbf{v}$, $i = 0, 1, \ldots, n$. Dotting either side of above equation by vector $\mathbf{v}$, we have

$$\lambda = \sum_{i=0}^{n} \theta_i(1 + \mu_i a_i^2).$$

As $|a_i| = |\mathbf{n}_i^T \mathbf{v}| \leq 1$ and $\mu_i > -1$, $i = 0, 1, \ldots, n$, it can be concluded that $\lambda > 0$. Since the matrix M_s is symmetric and has d eigenvalues, namely $\lambda_1, \lambda_2, \ldots, \lambda_d$, we have $|M_s| = \prod_{i=1}^{d} \lambda_i > 0$. This proves the theorem. □

From Theorem 2.1 we know that if the blending functions $\phi_i(\xi)$ are non-negative and linear independent over the parameter domain, the matrix weighted rational curve or surface $Q(\xi)$ given by Equation (2) is valid.

From a geometric point of view, a rational curve or a rational surface is the solution to the weighted least squares fitting to the control points with weights chosen as the scaled blending functions. It can be shown that a matrix weighted rational curve or surface with weight matrices defined by Equation (3) is the solution to the weighted least squares fitting to the control points and lines/planes that pass through the points. For a set of given points P_i, normal vectors $\mathbf{n}_i$ and coefficients $\omega_i > 0$, $\mu_i > -1$, $i = 0, 1, \ldots, n$, let

$$F(Q(\xi)) = \sum_{i=0}^{n} \omega_i \phi_i(\xi)[Q(\xi) - P_i]^2 + \sum_{i=0}^{n} \omega_i \mu_i \phi_i(\xi)[(Q(\xi) - P_i) \cdot \mathbf{n}_i]^2. \tag{4}$$

By minimizing the functional $F(Q(\xi))$ we have

$$\frac{\partial F(Q(\xi))}{\partial Q(\xi)} = \sum_{i=0}^{n} \omega_i \phi_i(\xi)[Q(\xi) - P_i] + \sum_{i=0}^{n} \omega_i \mu_i \phi_i(\xi) \mathbf{n}_i \mathbf{n}_i^T (Q(\xi) - P_i) = 0. \tag{5}$$

Solving Equation (5) we obtain the matrix weighted rational curve or matrix weighted rational surface $Q(\xi)$ with weight matrices defined by Equation (3). Clearly, if all μ_i vanish, the solution to Equation (5) is a conventional rational curve or a conventional rational surface. Similar to the effects of control points P_i, the normals $\mathbf{n}_i$ can be used to control the curve or surface's shape efficiently. We therefore refer the normals as the *control normals* of the curves or surfaces.

2.2. Basic properties of matrix weighted rational curves and surfaces with control normals

Matrix weighted rational curves and surfaces maintain several fundamental properties of traditional rational curves and surfaces. These properties are useful for shape modeling using the proposed models of curves and surfaces. We state the following two properties for the matrix weighted rational curves but similar properties hold for the matrix weighted rational surfaces.

Property 2.2 *(Geometric invariance). The shape of a matrix weighted rational curve is invariant under the translation or uniform scaling of all control points. Moreover, if the weight matrices are defined by Equation (3), the shape of the curve is also invariant under the rotation of the coordinate system.*

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

Proof. We first prove the geometric invariance under the translation or uniform scaling of the control points. Suppose that a matrix weighted rational curve is represented by Equation (2). Since

$$Q_0 = \left[\sum_{i=0}^{n} M_i\phi_i(\xi)\right]^{-1} \sum_{i=0}^{n} M_i Q_0\phi_i(\xi),$$

we have

$$Q(\xi) + Q_0 = \left[\sum_{i=0}^{n} M_i\phi_i(\xi)\right]^{-1} \sum_{i=0}^{n} M_i(P_i + Q_0)\phi_i(\xi).$$

This proves that the shape of a matrix weighted rational curve is invariant under the translation of control points. For a nonzero real number s we have

$$sQ(\xi) = \left[\sum_{i=0}^{n} M_i\phi_i(\xi)\right]^{-1} \sum_{i=0}^{n} M_i sP_i\phi_i(\xi),$$

which proves the geometric invariance under the uniform scaling of the control points.

We next prove the geometric invariance under the rotation of the coordinate system. Suppose that $A \in \mathbb{R}^{d\times d}$ is a rotation matrix. It yields that $AA^T = I$ and $A^{-1} = A^T$. By multiplying A on either side of the expression of $Q(\xi)$, we have

$$\begin{aligned} AQ(\xi) &= A\left[\textstyle\sum_{i=0}^{n} \omega_i(I + \mu_i\mathbf{n}_i\mathbf{n}_i^T)\phi_i(\xi)\right]^{-1} A^{-1}A\textstyle\sum_{i=0}^{n} \omega_i(I + \mu_i\mathbf{n}_i\mathbf{n}_i^T)A^T AP_i\phi_i(\xi) \\ &= \left[A\textstyle\sum_{i=0}^{n} \omega_i(I + \mu_i\mathbf{n}_i\mathbf{n}_i^T)\phi_i(\xi)A^{-1}\right]^{-1} \textstyle\sum_{i=0}^{n} \omega_i(AA^T + \mu_i A\mathbf{n}_i\mathbf{n}_i^T A^T)AP_i\phi_i(\xi) \\ &= \left\{\textstyle\sum_{i=0}^{n} \omega_i[I + \mu_i A\mathbf{n}_i(A\mathbf{n}_i)^T]\phi_i(\xi)\right\}^{-1} \textstyle\sum_{i=0}^{n} \omega_i[I + \mu_i A\mathbf{n}_i(A\mathbf{n}_i)^T]AP_i\phi_i(\xi). \end{aligned}$$

Thus AP_i and $A\mathbf{n}_i$ are the control points and control normals of $AQ(\xi)$. The proposition is proven. □

Property 2.3 *(Effects of weights). Suppose that $Q(\xi)$ is a matrix weighted rational curve defined by Equation (2) and the weight matrices are defined by Equation (3). The curve $Q(\xi)$ will be approximate tangent to the line or the plane that passes through point P_{i_0} with normal vector $\mathbf{n}_{i_0}$ when the parameter μ_{i_0} approaches infinity. If ω_{i_0} approaches infinity, the points on $Q(\xi)$ will approach the control point P_{i_0}.*

Proof. From the definition of the functional $F(Q(\xi))$ we know that if μ_{i_0} approaches infinity the term $\phi_i(\xi)[(Q(t) - P_{i_0}) \cdot \mathbf{n}_{i_0}]^2$ will approach zero for any $\xi \in support(\phi_i(\xi)) \bigcap \Xi$ when the functional $F(Q(\xi))$ is minimized. As $(Q(\xi) - P_{i_0}) \cdot \mathbf{n}_{i_0}$ approaches zero, the curve $Q(\xi)$ will be approximate tangent to the line ($d = 2$) or to the plane ($d \geq 3$) that passes through point P_{i_0} with normal $\mathbf{n}_{i_0}$. Similarly, if ω_{i_0} approaches infinity, the term $(Q(\xi) - P_{i_0})^2$ with $\xi \in support(\phi_i(\xi)) \bigcap \Xi$ will approach zero too and the point on curve $Q(\xi)$ will approach the control point P_{i_0}. □

3. Matrix weighted rational Bézier curves

A matrix weighted rational Bézier curve of degree n is defined by

$$Q(t) = \left[\sum_{i=0}^{n} M_i B_{i,n}(t)\right]^{-1} \sum_{i=0}^{n} M_i P_i B_{i,n}(t), \quad t \in [0, 1] \tag{6}$$

where $B_{i,n}(t) = \frac{n!}{i!(n-i)!} t^i(1-t)^{n-i}$ are the Bernstein basis functions. Based on Theorem 2.1 we know that the matrix weighted rational Bézier curve is valid when the weight matrices are given by Equation (3).

Besides the basic properties stated in Section 2, matrix weighted rational Bézier curves have some other elegant properties due to the special basis functions.

Property 3.1 *(End property). A matrix weighted rational Bézier curve of degree n interpolates two end control points. If the weight matrices satisfy $M_0 = M_1$ and $M_{n-1} = M_n$, the derivatives of the curve at two ends are parallel to the two end control legs.*

Proof. Suppose that a matrix weighted rational Bézier curve $Q(t)$ is defined by Equation (6). Substituting $t = 0$ into the equation, we have

$$Q(0) = \left[\sum_{i=0}^{n} M_i B_{i,n}(0)\right]^{-1} \sum_{i=0}^{n} M_i P_i B_{i,n}(0) = M_0^{-1} M_0 P_0 = P_0.$$

Similarly, we have $Q(1) = P_n$.

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

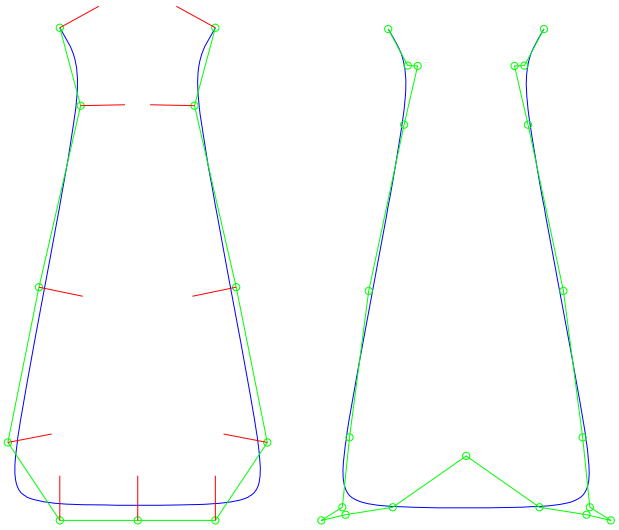


Fig. 1. (a) A matrix weighted rational Bézier curve generated by the control polygon and the control normals; (b) the converted rational Bézier curve and its control polygon.

Denoting $P_M(t) = \sum_{i=0}^{n} M_i P_i B_{i,n}(t)$ and $M(t) = \sum_{i=0}^{n} M_i B_{i,n}(t)$, the matrix weighted rational Bézier curve becomes $Q(t) = M^{-1}(t) P_M(t)$. Thus the derivative of $Q(t)$ can be obtained as

$$Q'(t) = M^{-1}(t)[P'_M(t) - M'(t)Q(t)].$$

If $M_0 = M_1$ and $M_{n-1} = M_n$, we have $M'(0) = M'(1) = 0$. Therefore, the derivatives of the matrix weighted rational Bézier curve at two ends are obtained as

$$Q'(0) = M^{-1}(0)[P'_M(0) - M'(0)Q(0)] = n(P_1 - P_0),$$
$$Q'(1) = M^{-1}(1)[P'_M(1) - M'(1)Q(1)] = n(P_n - P_{n-1}).$$

The proposition is proven. □

Property 3.2 *(Conversion). A matrix weighted Bézier curve of degree n in $\mathbb{R}^d$ can be converted to a rational Bézier curve of degree dn.*

Proof. Let $M(t)$ be the matrix function of degree n as defined in Equation (6). The inverse of $M(t)$ is computed by $[M(t)]^{-1} = M^{\star}(t)/|M(t)|$, where $M^{\star}(t)$ represents the adjoint matrix of $M(t)$ and $|M(t)|$ is the determinant of the matrix function. The matrix weighted rational Bézier curve $Q(t)$ can be reformulated as $Q(t) = M^{\star}(t) \sum_{i=0}^{n} M_i P_i B_{i,n}(t)/|M(t)|$. With simple computation, both the numerator and denominator can be formulated as Bézier functions of degree dn. Therefore, the matrix weighted rational Bézier curve $Q(t)$ is a rational Bézier curve of degree dn by reformulation. □

Even though matrix weighted rational Bézier curves can be reformulated as traditional rational Bézier curves, the former ones own the advantages of lower degrees and smaller numbers of control points. A curve with fewer control points can be edited more easily and the control normals can help to model straight or flat features very well. Fig. 1(a) illustrates a matrix weighted rational Bézier curve of degree 10 with weight matrices computed by the specified control normals. By choosing all $\omega_i = \mu_i = 1$ except for $\mu_2 = \mu_5 = \mu_8 = 20$ a bottle like shape with flat bottom and straight sides is generated. When the curve has been transformed into a rational Bézier curve of degree 20, the shape of the control polygon differs significantly with that of the curve; see Fig. 1(b). From the figure we also learn that a matrix weighted rational Bézier curve may not lie in the convex hull of its control polygon. The convex hull of a matrix weighted rational Bézier curve can be obtained by converting it to a rational Bézier curve or some other methods.

A point on a matrix weighted rational Bézier curve $Q(t) = M^{-1}(t) P_M(t)$ can be obtained by computing the matrix $M(t)$ and point $P_M(t)$ using the traditional de Casteljau algorithm. Alternatively, the point can also be obtained by applying a matrix rational de Casteljau algorithm which is similar to the rational algorithm for robust computation of points on rational Bézier curves (Farin, 2001). For a matrix weighted rational Bézier curve as defined by Equation (6), the matrix rational de Casteljau algorithm begins with the initialization $M_i^{[0]}(t) = M_i$ and $Q_i^{[0]}(t) = P_i$ for $i = 0, 1, \ldots, n$ and computes the intermediate matrices and points

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

$$M_i^{[r]}(t) = (1-t)M_i^{[r-1]}(t) + tM_{i+1}^{[r-1]}(t),$$
$$Q_i^{[r]}(t) = \left[M_i^{[r]}(t)\right]^{-1}\left[(1-t)M_i^{[r-1]}(t)Q_i^{[r-1]}(t) + tM_{i+1}^{[r-1]}(t)Q_{i+1}^{[r-1]}(t)\right]$$

for $r = 1, 2, \ldots, n$ and $i = 0, 1, \ldots, n-r$. Finally, the point is obtained as $Q(t) = Q_0^{[n]}(t)$. It is noticed that points $Q_i^{[r]}(t)$, $Q_i^{[r-1]}(t)$ and $Q_{i+1}^{[r-1]}(t)$ may not lie on the same line and the point evaluation by the matrix rational de Casteljau algorithm may no longer be a corner cutting process.

Degree elevation of a matrix weighted rational Bézier curve is also similar to that of the conventional rational Bézier curves. Assume that M_{-1} and M_{n+1} are zero matrices. Let $\overline{M}_i = \frac{i}{n+1}M_{i-1} + \left(1 - \frac{i}{n+1}\right)M_i$ and

$$\overline{P}_i = \left[\frac{i}{n+1}M_{i-1} + \left(1 - \frac{i}{n+1}\right)M_i\right]^{-1}\left[\frac{i}{n+1}M_{i-1}P_{i-1} + \left(1 - \frac{i}{n+1}\right)M_iP_i\right],$$

$i = 0, 1, \ldots, n+1$. A matrix weighted rational Bézier curve $Q(t)$ can be represented as

$$Q(t) = \left[\sum_{i=0}^{n} M_iB_{i,n}(t)\right]^{-1}\sum_{i=0}^{n} M_iP_iB_{i,n}(t) = \left[\sum_{i=0}^{n+1}\overline{M}_iB_{i,n+1}(t)\right]^{-1}\sum_{i=0}^{n+1}\overline{M}_i\overline{P}_iB_{i,n+1}(t).$$

By the same reason as the matrix rational de Casteljau algorithm, degree elevation of matrix weighted rational Bézier curves is generally not a corner cutting process either.

4. Matrix weighted NURBS curves

Suppose that $P_i \in \mathbb{R}^d$, $i = 0, 1, \ldots, n$ are a sequence of given points, $M_i \in \mathbb{R}^{d\times d}$, $i = 0, 1, \ldots, n$ are a set of given matrices. Let $\mathbf{t} = \{t_0, t_1, \ldots, t_{n+k}\}$ be a non-decreasing sequence and no k numbers are the same except for at two ends, a matrix weighted NURBS curve of order k (degree $k-1$) is defined by

$$Q(t) = \left[\sum_{i=0}^{n} M_iN_{i,k}(t)\right]^{-1}\sum_{i=0}^{n} M_iP_iN_{i,k}(t),\ t \in [t_{k-1}, t_{n+1}] \tag{7}$$

where $N_{i,k}(t)$, $i = 0, 1, \ldots, n$ are the B-splines defined on the knot vector $\mathbf{t}$. Particularly, the matrix weighted NURBS curve is valid when the weight matrices are defined by Equation (3).

Let $M(t) = \sum_{i=0}^{n} M_iN_{i,k}(t)$ and $P_M(t) = \sum_{i=0}^{n} M_iP_iN_{i,k}(t)$. Both the matrix $M(t)$ and the point $P_M(t)$ can be computed by the deBoor–Cox algorithm. Then, the corresponding point on the matrix weighted NURBS curve $Q(t)$ can be obtained as $Q(t) = M^{-1}(t)P_M(t)$ and the derivative of $Q(t)$ can be computed by $Q'(t) = M^{-1}(t)\left[P'_M(t) - M'(t)Q(t)\right]$, where $P'_M(t)$ and $M'(t)$ are the derivatives of the corresponding B-spline curve or the B-spline matrix function. Alternatively, the point $Q(t)$ can also be evaluated using the matrix rational deBoor–Cox algorithm, just as the rational deBoor–Cox algorithm for a NURBS curve (Farin, 2001).

In addition to the basic properties stated in Section 2, matrix weighted NURBS curves maintain a few similar properties as NURBS curves.

Property 4.1 *(Continuity order). A matrix weighted NURBS curve of order k is infinitely differentiable on the interior of knot spans and $k-1-p$ times differentiable at a knot of multiplicity p.*

Property 4.2 *(End interpolation). If the knot vector of a matrix weighted NURBS curve $Q(t)$ satisfies $t_0 = t_1 = \ldots = t_{k-1}$ and $t_{n+1} = t_{n+2} = \ldots = t_{n+k}$, we have $Q(t_{k-1}) = P_0$ and $Q(t_{n+1}) = P_n$. Moreover, if $M_0 = M_1$ and $M_{n-1} = M_n$, we also have $Q'(t_{k-1}) = \frac{k-1}{t_k - t_1}(P_1 - P_0)$ and $Q'(t_{n+1}) = \frac{k-1}{t_{n+k} - t_n}(P_n - P_{n-1})$.*

Property 4.3 *(Local approximation). The movement of control point P_i or the change of weight matrix M_i only changes the curve defined on interval $t \in [t_i, t_{i+k})$. If $t \in [t_l, t_{l+1})$, the expression for the matrix weighted NURBS curve is simplified as*

$$Q(t) = \left[\sum_{i=l-k+1}^{l} M_iN_{i,k}(t)\right]^{-1}\sum_{i=l-k+1}^{l} M_iP_iN_{i,k}(t).$$

Similar to the conversion of matrix weighted rational Bézier curves, matrix weighted NURBS curves can be converted to the traditional NURBS curves.

Property 4.4 *(Conversion to NURBS). A matrix weighted NURBS curve of order k in $\mathbb{R}^d$ can be converted to a NURBS curve of order $d(k-1)+1$.*

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

Besides matrix weighted NURBS curves of degree one that interpolate all of their control points, matrix weighted NURBS curves of higher degrees can interpolate selected control points when the control normals and scalar weights are properly given.

Property 4.5 *(Almost-interpolation). Assume that $k > 2$ is an integer and $l = \lfloor \frac{k-1}{2} \rfloor$, where $\lfloor\rfloor$ represents the integer part of a number. Let $\lambda_{q+j} = N_{q+j,k}(t_{q+\frac{k}{2}})$, $j = \pm 1, \pm 2, \ldots, \pm l$, where $N_{i,k}(t)$ are the uniform B-splines of order k and $t_{q+\frac{k}{2}} = \frac{1}{2}(t_q + t_{q+k})$. Suppose that L is the line that passes through point P_q with direction $\mathbf{n}_q$ on a plane, and points P_{q+j} and normals $\mathbf{n}_{q+j}$, $j = \pm 1, \ldots, \pm l$ symmetrically lie on two sides of the line L on the plane. If the weights μ_{q+j}, $j = \pm 1, \pm 2, \ldots, \pm l$ are all chosen as*

$$\mu = \frac{\sum_{|j|=1}^{l} \lambda_{q+j} \mathbf{n}_q^T (P_q - P_{q+j})}{\sum_{|j|=1}^{l} \lambda_{q+j} \mathbf{n}_q^T \mathbf{n}_{q+j} \mathbf{n}_{q+j}^T (P_{q+j} - P_q)}, \tag{8}$$

the matrix weighted NURBS curve with weight matrices $M_i = (I + \mu_i \mathbf{n}_i \mathbf{n}_i^T)$ passes through point P_q at $t = t_{q+\frac{k}{2}}$.

Proof. Suppose that a matrix weighted NURBS curve $Q(t)$ passes through point P_q at $t = t_{q+\frac{k}{2}}$, it requires that $Q(t_{q+\frac{k}{2}}) = P_q$. Since $Q(t)$ satisfies Equation (5), we have

$$\sum_{i=0}^{n} N_{i,k}(t_{q+\frac{k}{2}})(P_q - P_i) + \sum_{i=0}^{n} \mu_i N_{i,k}(t_{q+\frac{k}{2}}) \mathbf{n}_i \mathbf{n}_i^T (P_q - P_i) = 0.$$

By deleting all the vanishing terms and letting $\mu_i = \mu$, we have

$$\sum_{|j|=1}^{l} \lambda_{q+j}(P_q - P_{q+j}) + \mu \sum_{|j|=1}^{l} \lambda_{q+j} \mathbf{n}_{q+j} \mathbf{n}_{q+j}^T (P_q - P_{q+j}) = 0.$$

Dotting either side of the equation by $\mathbf{n}_q$ we have a unique solution of μ given by Equation (8). As points P_{q+j} and normals $\mathbf{n}_{q+j}$, $j = \pm 1, \ldots, \pm l$ symmetrically lie on two sides of the line L, the above steps can be reversed, which proves the proposition. □

We are interested in the computation of parameter μ for matrix weighted NURBS curves of order 3 or 4. If $k = 3$ or $k = 4$, we have $l = \lfloor \frac{k-1}{2} \rfloor = 1$ and $\lambda_{q-1} = \lambda_{q+1}$. Particularly, if points P_{q+j} and unit normals $\mathbf{n}_{q+j}$, $j = -1, 0, 1$ are uniformly sampled from a circular arc, it can be further verified that $\mathbf{n}_q^T (P_q - P_{q-1}) = \mathbf{n}_q^T (P_q - P_{q+1})$ and $\mathbf{n}_q^T (P_q - P_{q+j}) = \mathbf{n}_{q+j}^T (P_{q+j} - P_q)$ for $j = -1, 1$. As a result, the formula for parameter μ is simplified as

$$\mu = \frac{2}{\mathbf{n}_q^T \mathbf{n}_{q-1} + \mathbf{n}_q^T \mathbf{n}_{q+1}} = \frac{|N_1(q)|}{\sum_{j \in N_1(q)} \mathbf{n}_q^T \mathbf{n}_j}, \tag{9}$$

where $N_1(q)$ represents the index set of 1-ring neighbor points of point P_q and $|N_1(q)|$ is the valence of point P_q. We will use this latter formula to compute the local parameter μ for defining the weight matrices for surface reconstruction by low order matrix weighted NURBS surfaces or low order matrix weighted rational subdivision surfaces.

Fig. 2 illustrates an example of curve reconstruction by matrix weighted NURBS. The input 200 points were sampled from a spiral spline which has four pieces of monotone curvature plots. To reconstruct a smooth curve we first chose 30 points from the original data and compute normal vectors for the chosen points based on the input data. By using these points and normals as control points or control normals and computing the weight matrices by Equation (3) and Equation (8), matrix weighted NURBS curves of degree 1, 3 or 10 are generated. Even though the matrix weighted NURBS curve of degree 1 is only C^0 continuous at the joint points, it is still visually smooth; see Fig. 2(a). The matrix weighted NURBS curves of higher degrees can achieve higher orders of smoothness and improved fairness as well; see Figs. 2(b) and (c) for the reconstruction results.

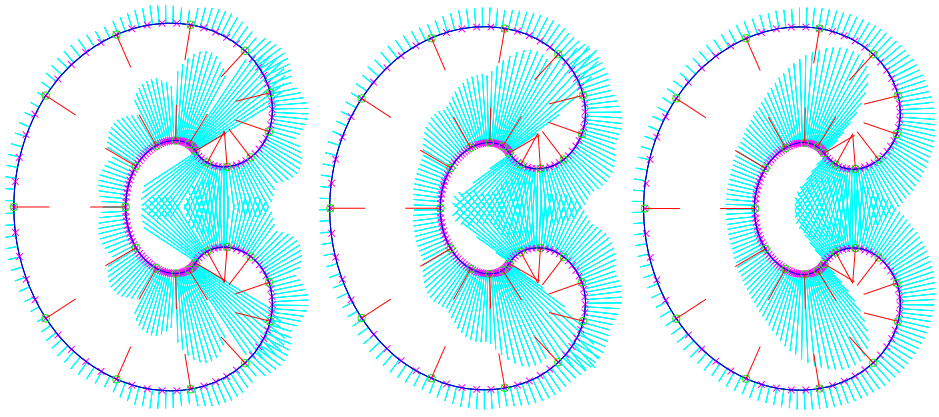
5. Matrix weighted rational surfaces

This section presents the extensions of Bézier surfaces and NURBS surfaces to matrix weighted rational surfaces. Several popular subdivision surfaces which are also the generalized B-spline surfaces with arbitrary topology control meshes will be extended to matrix weighted rational subdivision surfaces.

5.1. Matrix weighted rational parametric surfaces

Similar to the extensions of Bézier or NURBS curves, rectangle Bézier surfaces and NURBS surfaces can be extended to matrix weighted rational surfaces directly. Assuming P_{ij}, $i = 0, 1, \ldots, m$; $j = 0, 1, \ldots, n$ are a set of given points in $\mathbb{R}^d$, M_{ij},

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005



(a) (b) (c)

Fig. 2. Curve reconstruction by matrix weighted NURBS: (a) degree 1; (b) degree 3; (c) degree 10.

$i = 0, 1, \ldots, m;\ j = 0, 1, \ldots, n$ are a set of given matrices in $\mathbb{R}^{d\times d}$, a matrix weighted rational Bézier surface is defined by

$$Q_1(s,t) = \left[\sum_{i=0}^{m}\sum_{j=0}^{n} M_{ij}B_{i,m}(s)B_{j,n}(t)\right]^{-1} \sum_{i=0}^{m}\sum_{j=0}^{n} M_{ij}P_{ij}B_{i,m}(s)B_{j,n}(t), \quad (s,t)\in[0,1]^2 \tag{10}$$

With the same set of control points and weight matrices, a matrix weighted NURBS surface is obtained as

$$Q_2(u,v) = \left[\sum_{i=0}^{m}\sum_{j=0}^{n} M_{ij}N_{i,k_1}(u)N_{j,k_2}(v)\right]^{-1} \sum_{i=0}^{m}\sum_{j=0}^{n} M_{ij}P_{ij}N_{i,k_1}(u)N_{j,k_2}(v),$$
$$(u,v)\in[u_{k_1-1}, u_{m+1}]\times[v_{k_2-1}, v_{n+1}], \tag{11}$$

where $N_{i,k_1}(u)$ and $N_{j,k_2}(v)$ are the B-splines defined on knot vector $\mathbf{u} = \{u_0, u_1, \ldots, u_{m+k_1}\}$ or $\mathbf{v} = \{v_0, v_1, \ldots, v_{n+k_2}\}$, respectively.

Besides the rectangle Bézier patches, triangular Bézier patches can also be generalized to matrix weighted rational Bézier triangles. Suppose that P_{ijk}, $0 \le i, j, k \le n$, $i + j + k = n$ are the given control points and M_{ijk} are the corresponding weight matrices. A matrix weighted rational triangular Bézier patch of degree n is defined by

$$Q_3(u,v,w) = \left[\sum_{i+j+k=n} M_{ijk}B^n_{i,j,k}(u,v,w)\right]^{-1} \sum_{i+j+k=n} M_{ijk}P_{ijk}B^n_{i,j,k}(u,v,w),$$
$$0 \le u, v, w \le 1, u + v + w = 1, \tag{12}$$

where $B^n_{i,j,k}(u,v,w) = \frac{n!}{i!j!k!}u^i v^j w^k$ are the Bernstein basis functions defined on a triangular domain.

To control the surface shape using control points and control normals, the weight matrices M_{ij} or M_{ijk} for Equations (10), (11) or (12) can be computed by Equation (3) using normal vectors $\mathbf{n}_{ij}$ or $\mathbf{n}_{ijk}$ together with scalar weights ω_{ij}, μ_{ij} or ω_{ijk}, μ_{ijk} specified at the control points. As the matrix weighted rational surfaces have the same representation forms as the classical rational surfaces, they can be evaluated using similar algorithms for scalar weighted rational surfaces.

Fig. 3(a) illustrates an input control mesh and the traditional triangular Bézier patch generated by the control mesh. Under the same set of control points together with normals specified at the control points, a matrix weighted rational Bézier triangle is generated; see Fig. 3(b).

An original control mesh consisting of control points sampled from a twisted torus and a uniform B-spline surface of degree 3 in the u-direction (along the minor circle) and degree 6 in the v-direction (along the major circle) generated by the control mesh are illustrated in Fig. 4(a). We further model a matrix weighted NURBS surface of degree $(3, 6)$ by the control mesh together with a set of control normals specified at the control points. By computing all the weight matrices M_{ij} with $\omega_{ij} = 1$ and $\mu_{ij} = 10$, a twisted torus shape with salient local features modeled by a matrix weighted NURBS surface is obtained; see Fig. 4(b).

5.2. Matrix weighted rational subdivision surfaces

Just as the extensions of NURBS surfaces to matrix weighted NURBS surfaces, subdivision surfaces which are also the generalized B-spline surfaces with arbitrary topology control meshes can be extended to matrix weighted rational subdi-

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

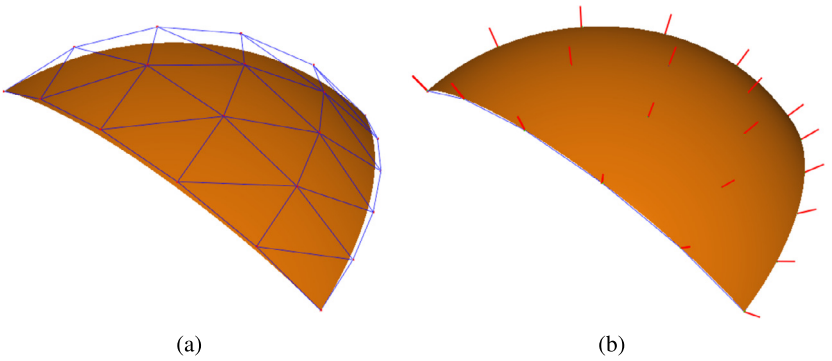


Fig. 3. (a) A Bézier triangle of degree 5 and the control mesh; (b) a matrix weighted rational Bézier triangle generated by the same control mesh as in (a) with additional control normals ($\omega_{ijk} = 1$, $\mu_{ijk} = 1.2$).

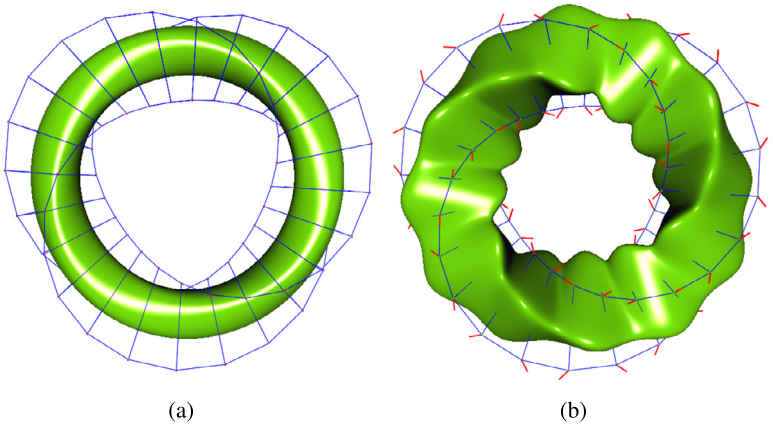


Fig. 4. Surface modeling using B-spline or matrix weighted NURBS surfaces: (a) a B-spline surface of degree $(3, 6)$; (b) a matrix weighted NURBS surface of degree $(3, 6)$ with control points and the control normals.

vision surfaces. Particularly, the weight matrices can be defined using the specified normals such that the shapes of the extended subdivision surfaces can be controlled by control points and control normals.

The control data for matrix weighted rational subdivision consists of an initial control mesh and a set of weight matrices specified at the initial control points. For each subdivision the control points and weight matrices will be refined and the generated points will be reconnected to form a new mesh by the same topology refinement rules of the traditional subdivision scheme. Suppose that mesh vertices in mesh $\mathcal{M}^k$ are refined by a non-rational subdivision scheme as

$$V_i^{k+1} = \sum_j s_{ij} V_j^k,$$

where the subdivision mask $\{s_{ij}\}$ satisfies $\sum_j s_{ij} = 1$ and $s_{ij} \geq 0$. The new weight matrices and new vertices by a matrix weighted rational subdivision scheme are computed by

$$\begin{aligned} M_i^{k+1} &= \textstyle\sum_j s_{ij} M_j^k, \\ V_i^{k+1} &= [M_i^{k+1}]^{-1} \textstyle\sum_j s_{ij} M_j^k V_j^k. \end{aligned} \tag{13}$$

Clearly, the matrix weighted rational subdivision will degenerate to the traditional subdivision when all the initial weight matrices are non-singular and are equal with each other.

As an example, we briefly present the algorithm steps and smoothness analysis for matrix weighted rational Catmull–Clark subdivision. Other popular subdivision schemes like Doo–Sabin subdivision and Loop subdivision can be extended similarly.

The first step to refine a mesh $\mathcal{M}^k$ is to compute new weight matrices. For a face F that consists of vertices V_i^k and weight matrices M_i^k, $i = 0, 1, \ldots, L-1$ in mesh $\mathcal{M}^k$, the face matrix corresponding to the face is computed by

$$M_F^{k+1} = \frac{1}{L} \sum_{i=0}^{L-1} M_i^k.$$

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

When all face matrices have been obtained, the edge matrix for each edge is computed by

$$M_E^{k+1} = \frac{1}{4}(M_0^k + M_1^k + M_{F_0}^{k+1} + M_{F_1}^{k+1}),$$

where M_0^k and M_1^k are the weight matrices at two ends of the edge and $M_{F_0}^{k+1}$ and $M_{F_1}^{k+1}$ are the face matrices computed for the two neighboring faces of the edge. After that, we refine the weight matrices corresponding to all vertices of $\mathcal{M}^k$. If a vertex V_V^k is of valence n, the new weight matrix for the vertex is obtained as

$$M_V^{k+1} = \frac{1}{n^2}\sum_{i=0}^{n-1} M_{F_i}^{k+1} + \frac{1}{n^2}\sum_{i=0}^{n-1} M_{E_i}^k + \frac{n-2}{n} M_V^k,$$

where $M_{F_i}^{k+1}$ are the face matrices of the neighboring faces and $M_{E_i}^k$ are the end weight matrices of neighboring edges other than M_V^k.

The second step is to compute new mesh vertices based on the control data of $\mathcal{M}^k$ and the newly obtained weight matrices as follows

$$\begin{aligned}
V_F^{k+1} &= \left[M_F^{k+1}\right]^{-1} \tfrac{1}{L}\textstyle\sum_{i=0}^{L-1} M_i^k V_i^k,\\
V_E^{k+1} &= \left[M_E^{k+1}\right]^{-1} \tfrac{1}{4}(M_0^k V_0^k + M_1^k V_1^k + M_{F_0}^{k+1} V_{F_0}^{k+1} + M_{F_1}^{k+1} V_{F_1}^{k+1}),\\
V_V^{k+1} &= \left[M_V^{k+1}\right]^{-1} (\tfrac{1}{n^2}\textstyle\sum_{i=0}^{n-1} M_{F_i}^{k+1} V_{F_i}^{k+1} + \tfrac{1}{n^2}\sum_{i=0}^{n-1} M_{E_i}^k V_{E_i}^k + \tfrac{n-2}{n} M_V^k V_V^k).
\end{aligned}$$

Lastly, connect all the refined vertices into a new mesh, just by the topology refinement rules of Catmull–Clark subdivision.

As for the smoothness of the subdivision surface we have the following result.

Theorem 5.1. *Suppose that $\mathcal{M}^0$ is an arbitrary topology control mesh and $\mathbf{n}_i$ are the unit normal vectors specified at the control points V_i^0 of the mesh. If the weight matrices for the points are defined by $M_i^0 = \omega_i(I + \mu_i \mathbf{n}_i \mathbf{n}_i^T)$, where $\omega_i > 0$ and $\mu_i > -1$, the mesh sequences generated by the matrix weighted rational Catmull–Clark subdivision converge and the limit surface is C^2 except for at the extraordinary points where the smoothness is C^1.*

Proof. By employing the homogeneous coordinates representation, the control data for the initial control mesh are $(M_i^0 V_i^0, M_i^0)$. Thus, the presented matrix weighted rational Catmull–Clark subdivision becomes non-rational subdivision in the homogeneous space $\mathbb{R}^{(d+1)\times d}$. Based on the smoothness analysis of Catmull–Clark subdivision (Doo and Sabin, 1978; Ball and Storry, 1988; Reif, 1995), the homogeneous meshes consisting of data $(M_i^k V_i^k, M_i^k)$ converge to a C^2 surface except for at extraordinary points where the smoothness becomes C^1 in $\mathbb{R}^{(d+1)\times d}$. On the other hand, from Equation (13) and Theorem 2.1 we know that each matrix M_i^k is non-singular. From Halstead et al. (1993) we know that the limit matrix of M_i^k is just a convex combination of the initial weight matrices even at an extraordinary vertex. By Theorem 2.1 once again, we conclude that the matrix component of the limit surface in the homogeneous space is nonsingular. Thus, the matrix weighted rational Catmull–Clark subdivision surface in Euclidean space is valid and has the same smoothness as the limit surface in the homogeneous space, i.e., the limit surface is C^2 except for at the extraordinary points where the smoothness is C^1. □

Fig. 5(a) illustrates a surface generated by Catmull–Clark subdivision from an initial control mesh. By using the same control mesh and computing the weight matrices based on the specified unit normal vectors, a matrix weighted rational Catmull–Clark subdivision surface is generated; see Fig. 5(b). Note that the control normals for vertices on each middle boundary control polygon are parallel, thus the four middle boundary curves are the same as those of the Catmull–Clark surface. Under the control of the specified normals, the surface becomes straight and flat in the vertical direction. Fig. 5(c) shows that if we choose some negative weights μ, the surface may shrink in the opposite directions of the control normals.

6. Examples

In this section we present a few more examples to demonstrate how to use the proposed models for shape modeling or surface reconstruction. For each example, the weight matrices are defined with normal vectors specified at the initial control points together with the default choice of $\omega_i = 1$ or $\omega_{ij} = 1$. If a matrix weighted NURBS surface is generated, the basis functions are defined on uniform knot vectors except for the multiple knots at the boundaries.

First, we present an example of surface design using Bézier or matrix weighted rational Bézier surfaces. Fig. 6(a) illustrates a control mesh that consists of 9×11 control points and a Bézier surface of degree $(8, 10)$ generated from the control mesh. To generate a matrix weighted rational Bézier surface from the control mesh we first assign a set of unit normals at

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

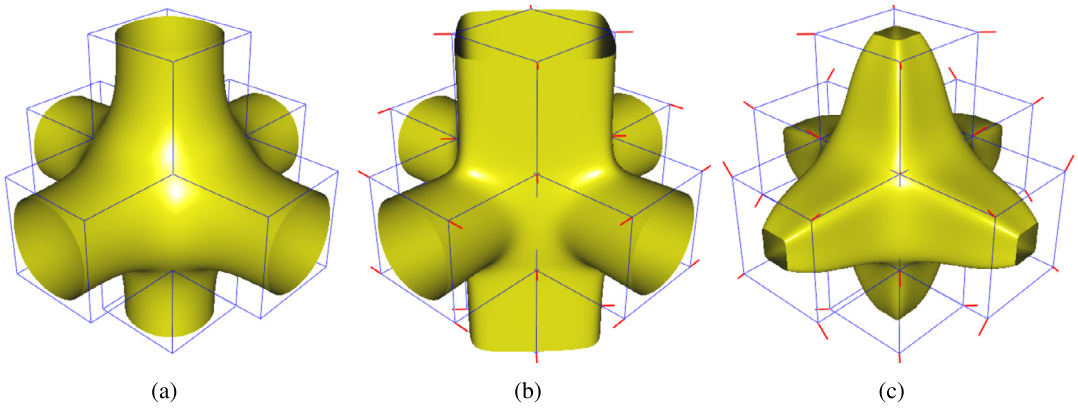


Fig. 5. (a) Catmull–Clark subdivision surface; (b) matrix weighted rational Catmull–Clark surface with $\omega_i = 1$ and $\mu_i = 10$; (c) matrix weighted rational Catmull–Clark surface with $\omega_i = 1$ and $\mu_i = -0.8$.

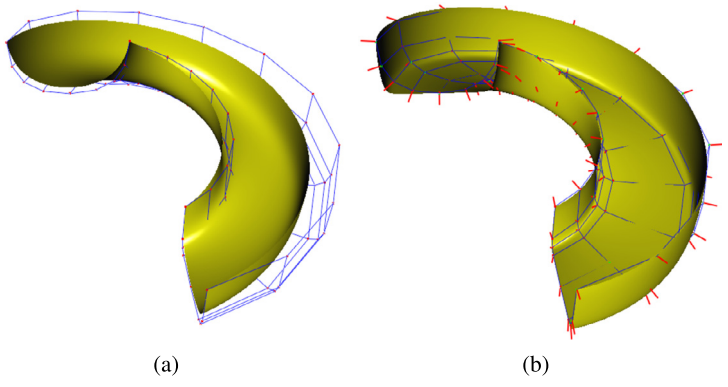


Fig. 6. (a) The input control mesh and the Bézier surface generated by the control mesh; (b) matrix weighted rational Bézier surface generated by the control mesh and the specified normals.

the control points. By choosing all parameters $\mu_{ij} = 1.35$ except for $\mu_{4,j} = 1000$, $0 \le j \le 10$, a sliding board like surface that has a flat bottom is generated; see Fig. 6(b).

Second, we present an example of surface reconstruction by a matrix weighted NURBS surface. Fig. 7(a) illustrates a screwdriver model from which 25×50 points, where 25 is the total row number, were sampled. A set of unit normal vectors at the sampled points have also been estimated from the surface. To reconstruct a bi-cubic matrix weighted NURBS surface with sharp features from the points and normals, we compute $\mu_{ij} = \frac{1}{d_{min}+\eta}$, where $d_{min} = \min\{\mathbf{n}_{ij}\mathbf{n}_{i+1,j}, \mathbf{n}_{ij}\mathbf{n}_{i-1,j}, \mathbf{n}_{ij}\mathbf{n}_{i,j+1}, \mathbf{n}_{ij}\mathbf{n}_{i,j-1}\}$ and $\eta = 0.15$. In case $i \pm 1$ or $j \pm 1$ is outside the range, the term $\mathbf{n}_{ij}\mathbf{n}_{i\pm1,j}$ or $\mathbf{n}_{ij}\mathbf{n}_{i,j\pm1}$ is chosen as 1 directly. Fig. 7(b) illustrates the matrix weighted NURBS surface and the control mesh together with the control normals scaled with weights μ_{ij}. Fig. 7(c) illustrates the final matrix weighted NURBS surface with clear sharp features. As a comparison, a bi-cubic B-spline surface can be used to fit the sampled points closely, but the original features have been blurred; see Fig. 7(d).

Third, we present two examples of surface modeling by matrix weighted rational subdivision schemes. Fig. 8(a) illustrates an initial control mesh and the Catmull–Clark subdivision surface generated by the control mesh. To model features like flat stairs on the surface a set of control normals are also assigned. Then a subdivision surface with salient features is obtained by applying the matrix weighted rational Catmull–Clark subdivision; see Fig. 8(b) for the control normals and the subdivision surface. Similarly, Figs. 8(c) and (d) are the surfaces generated by the Doo–Sabin subdivision or matrix weighted rational Doo–Sabin subdivision, respectively. The weight matrices for the latter one are computed using the specified normals and a fixed parameter $\mu = \sqrt{2}$ for all control points. As the boundary control points together with the control normals at these points lie in the same plane and the parameter μ satisfies Equation (9), the boundary curve by the matrix weighted rational subdivision interpolates exactly the control points.

Lastly, we present an example of surface generation using matrix weighted rational Loop subdivision. The comparison with the traditional Loop subdivision is also given. Fig. 9(a) illustrates the control mesh and the Loop subdivision surface. By choosing the control normals as the averages of normals of neighboring faces for all vertices of the control mesh and computing the scalar weight μ_i by Equation (9), a matrix weighted rational Loop subdivision surface that does not suffer shrinkage from the control mesh is generated; see Fig. 9(c). By employing the technique in Yang and Zheng (2013) we compute the curvatures of an intermediate subdivision mesh to check the smoothness of the subdivision surfaces. From

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

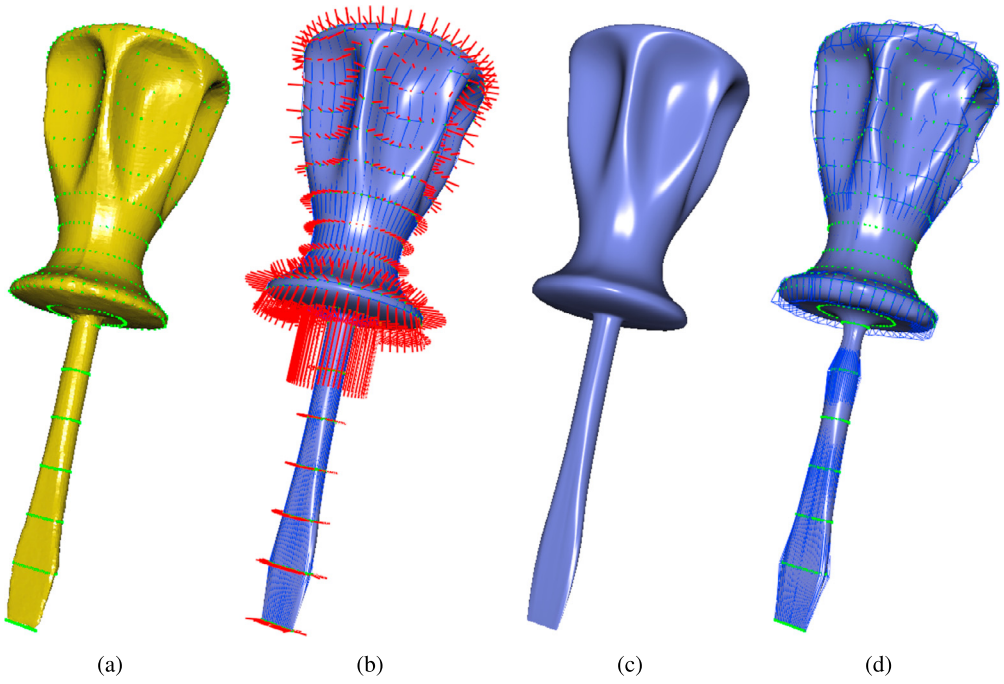


Fig. 7. Surface reconstruction from the sampled data: (a) points sampled from an input surface; (b) a bi-cubic matrix weighted NURBS surface with the control mesh and the scaled control normals; (c) the matrix weighted NURBS surface; (d) a bi-cubic B-spline surface fitting to the sampled points.

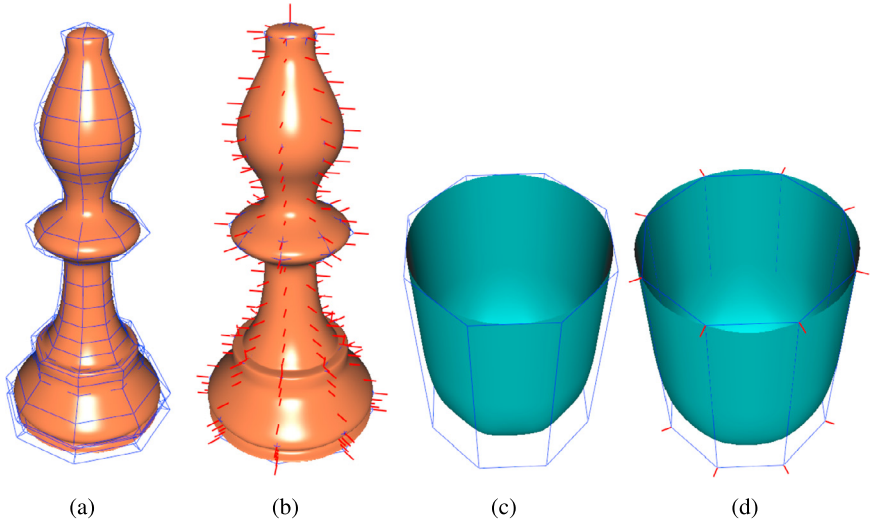


Fig. 8. Subdivision surfaces with or without control normals: (a) Catmull–Clark subdivision surface; (b) matrix weighted rational Catmull–Clark subdivision surface; (c) Doo–Sabin subdivision surface; (d) matrix weighted rational Doo–Sabin subdivision surface.

Figs. 9(b) and (d) we can see that the curvature artifacts near the extraordinary vertices within the Loop subdivision surface almost disappear in the matrix weighted rational subdivision surface.

7. Conclusion

We have generalized conventional rational curves and surfaces to matrix weighted rational curves and surfaces. The new models of rational curves and surfaces maintain the same structures and many similar properties as the traditional ones but the matrix weights permit clear shape parameters for geometric modeling. In addition to control points, control normals have been used to control the shapes of extended rational Bézier or extended NURBS curves and surfaces as well as several extended rational subdivision surfaces efficiently. Due to their nice properties such as almost-interpolation of control points, matrix weighted NURBS curves and surfaces are also suitable for curve or surface reconstruction from sampled points even

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005

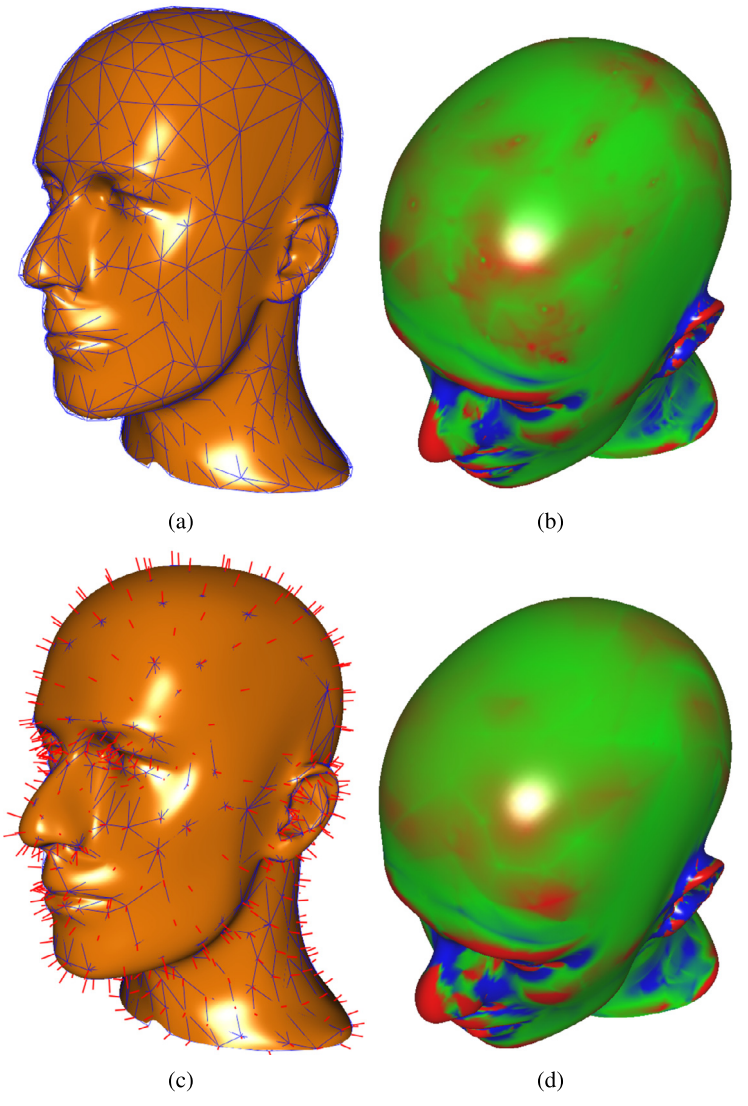


Fig. 9. Loop type subdivision surfaces: (a) the Loop subdivision surface; (b) the Gaussian curvature plot for the Loop subdivision surface; (c) the matrix weighted rational Loop subdivision surface; (d) the Gaussian curvature plot for the matrix weighted rational Loop subdivision surface. In (b) and (d) the color changes from red to blue when the curvature changes from high to low values. Readers can zoom in the figures to see the curvature artifacts. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with no need of solving large systems. As matrix weighted rational Bézier or NURBS curves and surfaces can be transformed from and to the traditional rational representations of curves and surfaces, it is convenient to plug the proposed models into Bézier or NURBS based modeling systems.

Several techniques need to be developed further in the future which can make the matrix weighted rational models even more useful for geometric modeling or surface reconstruction: (1) Define the weight matrices in some other geometric ways such that the obtained curves or surfaces can have other desired features; (2) Reconstruct curves and surfaces by adaptively choosing the control points and control normals from the input data; (3) Improve the smoothness of matrix weighted rational subdivision surfaces by applying geometric driven or non-stationary subdivision rules on the weight matrices.

Acknowledgements

We thank the editor and the reviewers for helpful comments and suggestions. This work is supported by the National Natural Science Foundation of China grants (11290142, 61272300).

References

Ball, A.A., Storry, D.J.T., 1988. Conditions for tangent plane continuity over recursively generated B-spline surfaces. ACM Trans. Graph. 7 (2), 83–102.

Cashman, T.J., Augsdörfer, U.H., Dodgson, N.A., Sabin, M.A., 2009. NURBS with extraordinary points: high-degree, non-uniform, rational subdivision schemes. ACM Trans. Graph. 28 (3), 46:1–46:9.
Catmull, E., Clark, J., 1978. Recursively generated B-spline surfaces on arbitrary topological meshes. Comput. Aided Des. 10 (6), 350–355.
DeRose, T., Kass, M., Truong, T., 1998. Subdivision surfaces in character animation. In: Proceedings of SIGGRAPH'98, pp. 85–94.
Doo, D., Sabin, M., 1978. Behaviour of recursive division surfaces near extraordinary points. Comput. Aided Des. 10 (6), 356–360.
Farin, G., 1999. NURBS: From Projective Geometry to Practical Use, second edition. A.K. Peters.
Farin, G., 2001. Curves and Surfaces for CAGD: A Practical Guide, fifth edition. Morgan Kaufmann.
Gofuku, S., Tamura, S., Maekawa, T., 2009. Point-tangent/point-normal B-spline curve interpolation by geometric algorithms. Comput. Aided Des. 41 (6), 412–422.
Goshtasby, A.A., 2005. Plus curves and surfaces. Vis. Comput. 21 (1–2), 4–16.
Halstead, M., Kass, M., DeRose, T., 1993. Efficient, fair interpolation using Catmull–Clark surfaces. In: Proceedings of SIGGRAPH'93, pp. 35–44.
Hoschek, J., 1983. Dual Bézier curves and surfaces. In: Barnhill, R., Boehm, W. (Eds.), Surfaces in Computer Aided Geometric Design. North-Holland, pp. 147–156.
Juhász, I., Róth, Á., 2013. A class of generalized B-spline curves. Comput. Aided Geom. Des. 30 (1), 85–115.
Kosinka, J., Sabin, M.A., Dodgson, N.A., 2015. Control vectors for splines. Comput. Aided Des. 58, 173–178.
Loop, C., 1987. Smooth subdivision surfaces based on triangles. Master's thesis. University of Utah, Salt Lake City.
Ma, X., Keates, S., Jiang, Y., Kosinka, J., 2015. Subdivision surface fitting to a dense mesh using ridges and umbilics. Comput. Aided Geom. Des. 32, 5–21.
Müller, K., Reusche, L., Fellner, D., 2006. Extended subdivision surfaces: building a bridge between NURBS and Catmull–Clark surfaces. ACM Trans. Graph. 25 (2), 268–292.
Piegl, L., Tiller, W., 1997. The NURBS Book, second edition. Springer.
Reif, U., 1995. A unified approach to subdivision algorithms near extraordinary vertices. Comput. Aided Geom. Des. 12 (2), 153–174.
Sánchez-Reyes, J., 2009. Complex rational Bézier curves. Comput. Aided Geom. Des. 26 (8), 865–876.
Schaefer, S., Goldman, R., 2009. Non-uniform subdivision for B-splines of arbitrary degree. Comput. Aided Geom. Des. 26 (1), 75–81.
Sederberg, T.W., Zheng, J., Sewell, D., Sabin, M., 1998. Non-uniform recursive subdivision surfaces. In: Proceedings of SIGGRAPH'98, pp. 387–394.
Stam, J., 2001. On subdivision schemes generalizing uniform B-spline surfaces of arbitrary degree. Comput. Aided Geom. Des. 18 (5), 383–396.
Wu, X., Cai, Y., Zheng, J., 2013. TV-L1 optimization for B-spline surface reconstruction with sharp features. In: 2013 International Conference on Computer-Aided Design and Computer Graphics, pp. 44–51.
Xie, W., Zou, X., Yang, J., Yang, J., 2012. Iteration and optimization scheme for the reconstruction of 3D surfaces based on non-uniform rational B-splines. Comput. Aided Des. 44 (11), 1127–1140.
Yang, X., Zheng, J., 2013. Curvature tensor computation by piecewise surface interpolation. Comput. Aided Des. 45 (12), 1639–1650.

Please cite this article in press as: Yang, X. Matrix weighted rational curves and surfaces. Comput. Aided Geom. Des. (2015), http://dx.doi.org/10.1016/j.cagd.2015.11.005