Blending using ODE swept surfaces with shape control and C^1 continuity

L.H. You · H. Ugail · B.P. Tang · Xiaogang Jin · X.Y. You · J.J. Zhang

Abstract Surface blending with tangential continuity is most widely applied in computer aided design, manufacturing systems, and geometric modeling. In this paper, we propose a new blending method to effectively control the shape of blending surfaces, which can also satisfy the blending constraints of tangent continuity exactly. This new blending method is based on the concept of swept surfaces controlled by a vector-valued fourth order ordinary differential equation (ODE). It creates blending surfaces by sweeping a generator along two trimlines and making the generator exactly satisfy the tangential constraints at the trimlines. The shape of blending surfaces is controlled by manipulating the generator with the solution to a vector-valued fourth order ODE. This new blending methods have the following advantages: 1). exact satisfaction of C^1 continuous blending boundary constraints, 2). effective shape control of blending surfaces, 3). high computing efficiency due to explicit mathematical representation of blending surfaces, and 4). ability to blend multiple (more than two) primary surfaces.

Keywords surface blending, C^1 continuity, shape control, fourth order ordinary differential equations, analytical solution, swept surfaces

L.H. You · J.J. Zhang (Corresponding author) ⁂
National Centre for Computer Animation, Bournemouth University, UK
Email: lyou@bournemouth.ac.uk

⁂ Traction Power State Key Laboratory, Southwest Jiaotong Univerzity, China
Email: jzhang@bournemouth.ac.uk

H. Ugail
Center for Visual Computing, University of Bardford, UK
Email: H.Ugail@Bradford.ac.uk

B.P. Tang
College of Mechanical Engineering, Chongqing University, China
Email: bptang@cqu.edu.cn

Xiaogang Jin
State Key Lab of CAD & CG, Zhejiang University, China
Email: jin@cad.zju.edu.cn

X.Y. You
Faculty of Engineering and Computing, Coventry University, UK
Email: lawrenceyoux@googlemail.com

1 Introduction

Surface blending is widely used in many applications such as computer-aided design, manufacturing systems, and geometric modeling to achieve a smooth transition between two or more separate primary surfaces for strength, manufacturing, aesthetic and usage purposes. Due to its wide range of applications, various surface blending approaches have been proposed.

Among existing surface blending approaches, tangent continuous surface blending is most widely applied since it can meet the requirements of the majority of applications and is easy to create. For tangent continuous surface blending, an important issue is how to achieve both effective shape control and at the same time exact satisfaction of tangent continuity, since both requirements are extremely important as discussed below.

Exact satisfaction of blending boundary constraints can guarantee a smooth transition between primary and blending surfaces. Such a smooth transition can lead to smooth movements and avoid undesirable impact. Here, we take a car as an example to demonstrate this. The car moves along a straight path consisting of two segments $C_1(u)$ and $C_2(u)$. If the transition between $C_1(u)$ and $C_2(u)$ is C^1 continuous, we have $\partial C_1(u)/\partial u = \partial C_2(u)/\partial u$. It indicates that when the car moves from $C_1(u)$ to $C_2(u)$, the velocity of the car will not change which guarantees a smooth movement and avoids the impact caused by the abrupt velocity change. In contrast, if the transition between $C_1(u)$ and $C_2(u)$ is not C^1 continuous, $\partial C_1(u)/\partial u \neq \partial C_2(u)/\partial u$, when the car moves from $C_1(u)$ to $C_2(u)$, the velocity of the car is suddenly changed which causes an undesirable impact and an unsmooth movement.

Shape control is required in many situations. Desirable shapes of blending surfaces can lead to a better performance such as a more uniform stress distribution in engineering applications or a more pleasing appearance for aesthetic purpose.

In existing work, shape manipulation of blending surfaces is usually achieved by adjusting the magnitude of blending boundary tangents. In order to ensure the continuity of both the direction and magnitude of blending boundary tangents, new shape control handles have to be introduced which is not an easy task. In this paper, we will address this issue with a new blending method. It constructs C^1 continuous blending surfaces with a controllable shape by sweeping a three-dimensional curve called a generator along two three-dimensional trajectories called the trimlines. Positional and tangential continuities at the trimlines will be satisfied

exactly. The shape of the generator is controlled by a vector-valued fourth order ODE.

The rolling-ball blending is very popular in connecting two separate primary surfaces smoothly. With this blending method, shape control is achieved by changing the radius of the rolling ball. However it can fail when two boundary curves (trimlines) of a blending surface are specified. Our proposed method can solve this problem effectively by simply adjusting the values of shape control parameters involved in the vector-valued ODE.

The partial differential equation (PDE) based surface blending is especially suitable for the smooth connection between two separate primary surfaces. However, such a blending method has two weaknesses: 1). it is difficult to blend more than two separate primary surfaces, 2). for complicated surface blending problems, analytical solutions of PDEs do not exist. Numerical solutions or approximate solutions of PDEs are the only choices which result in more computational cost. In contrast, ODE swept surface-based blending can blend multiple (more than two) primary surfaces and is very efficient in generating required blending surfaces.

In what follows, we firstly review the related work in Section 2. Then, we investigate the mathematical model and an analytical solution of curve-based surface blending in Section 3 and give other analytical solutions in Appendix A. Next, we implement and validate our proposed analytical solutions in Section 4, and investigate shape control of blending surfaces in Section 5. Finally, we use an example to demonstrate the application of our proposed approach in blending more than two primary surfaces in Section 6, and conclude the work given in this paper in Section 7.

2 Related work

There are many publications on surface blending [1]. In this section, we only briefly review some popular approaches and those which are closely related to the work given in this paper.

Rolling-ball blending is the most popular method. It was proposed by Rossignac and Requicha [2]. Rolling-ball blending can be used to blend both implicit and parametric surfaces. Depending on whether the radius of the rolling ball changes or not, rolling-ball blending can be divided into constant-radius [2-6] and variable-radius [7-9] blends. Recently, Whited and Rossignac introduced the concept of relative blending and a set theoretic formulation for variable-radius blending [10].

Blending surfaces constructed with the rolling ball methods are circular. Some other blending methods can create noncircular blending surfaces such as branching blends with Pythagorean normal surfaces [11], vertex blending using S-patches [12, 13], N-sided hole filling [14-20], and the following partial differential equation-based blends.

Surface blending using partial differential equations is an effective approach especially in dealing with various blending problems between two primary surfaces. With such an approach, a blending surface can be constructed from the solution to a vector-valued PDE which satisfies the positional, tangential or higher order continuities at the

trimlines. PDE-based surface blending was pioneered in [21]. Since analytical solutions to PDEs are difficult to obtain, numerical and approximate analytical solutions were investigated. Li [22, 23] and Li and Chang [24] proposed boundary penalty finite element methods of surface blending. Bloor and Wilson developed a perturbation method to generate blending surfaces [25]. They also investigated a pseudo-spectral method for the construction of regular 4-sided patches [26]. You et al. presented approximate analytical methods [27, 28]. In addition to partial differential equation-based surface blending, solid modelling using partial differential equations has also been investigated in [29, 30].

PDE-based surface blending methods involve two parametric variables. They are very difficult to solve and have low efficiency. In contrast, ODE-based approaches involve one parametric variable only. They are very easy to solve and have high efficiency. Due to this reason, ODE-based approaches have been applied in shape manipulation [31] and skin deformation determination [32, 33].

The work given in this paper will introduce ODE-based approaches into surface blending to develop a swept surface and use the surface to achieve surface blending with shape control and exact satisfaction of C^1 continuity.

3 Mathematical model and solution

For parametric representation of C^1 continuous blending surfaces, tangential continuity is defined by first partial derivatives of primary surfaces at the trimlines. Therefore, blending boundary constraints for surface blending with positional and tangential continuities can be written as,

$$
\begin{aligned}
u = 0 \qquad & \mathbf{S}(u,v) = \mathbf{C}_0(v) \qquad \frac{\partial \mathbf{S}(u,v)}{\partial u} = \overline{\mathbf{C}}_0(v) \\
u = 1 \qquad & \mathbf{S}(u,v) = \mathbf{C}_1(v) \qquad \frac{\partial \mathbf{S}(u,v)}{\partial u} = \overline{\mathbf{C}}_1(v)
\end{aligned}
\tag{1}
$$

where the vector-valued function $\mathbf{S}(u,v) = \begin{bmatrix} S_x(u,v) & S_y(u,v) & S_z(u,v) \end{bmatrix}^T$ is the mathematical representation of a blending surface, u and v are parametric variables, and $\mathbf{C}_0(v) = \begin{bmatrix} C_{0x}(v) & C_{0y}(v) & C_{0z}(v) \end{bmatrix}^T$ and $\mathbf{C}_1(v) = \begin{bmatrix} C_{1x}(v) & C_{1y}(v) & C_{1z}(v) \end{bmatrix}^T$ are vector-valued positional functions, and $\overline{\mathbf{C}}_0(v) = \begin{bmatrix} \overline{C}_{0x}(v) & \overline{C}_{0y}(v) & \overline{C}_{0z}(v) \end{bmatrix}^T$ and $\overline{\mathbf{C}}_1(v) = \begin{bmatrix} \overline{C}_{1x}(v) & \overline{C}_{1y}(v) & \overline{C}_{1z}(v) \end{bmatrix}^T$ are first partial derivatives of primary surfaces at the trimlines.

The functions $\mathbf{C}_0(v)$, $\overline{\mathbf{C}}_0(v)$, $\mathbf{C}_1(v)$ and $\overline{\mathbf{C}}_1(v)$ in blending boundary constraints (1) can be determined below.

If the trimline $\mathbf{C}_i(v)$ ($i = 0$ or 1) is an isoparametric line of a primary surface $\mathbf{P}_i(u,v)$ where u and v are two parametric variables, $\mathbf{C}_i(v)$ can be easily determined by taking the parametric variable u to be the parametric value u_m of the isoparametric line, i.e., $\mathbf{C}_i(v) = \mathbf{P}_i(u_m,v)$. Similarly, the first partial derivative of the primary surface at the isoparametric line can be formulated as $\overline{\mathbf{C}}_i(v) = \partial \mathbf{P}_i(u_m,v)/\partial u$ ($i = 0$ or 1).

If the trimline is not an isoparametric line of a primary surface $\mathbf{P}_i(s,t)$ where s and t are two parametric variables,

the trimline is determined by $\mathbf{C}_i(v) = \mathbf{P}_i(s(v),t(v)) = \mathbf{P}_i(v)$ and the tangent $\overline{\mathbf{C}}_i(v) = \mathbf{T}_i(v)$ on the trimline is determined by $\mathbf{T}_i(v) = -[\partial\mathbf{P}_i(s(v),t(v))/\partial s](dt/dv) + [\partial\mathbf{P}_i(s(v),t(v))/\partial t](ds/dv)$ where v is a curve parameter [34].

In order to manipulate the shape of blending surfaces, the parameters which affect the shape but have no influence on the blending boundary constraints should be introduced into the mathematical representation of the blending surfaces. A vector-valued ODE provides an effective way of introducing such parameters due to the following reason. The geometric representation of the solution to a vector-valued ODE is a three-dimensional curve and a blending surface can be created by sweeping the curve (generator) along two trimlines and exactly satisfying the boundary tangent at the trimlines at the same time. The parameters involved in the ODE have an influence on the shape of the curve and can be used as shape control handles of the blending surface. In this paper, blending surfaces created with this approach are called ODE blending surfaces.

Since fourth order ODEs involve four unknown constants which can be used to satisfy the four positional and tangential functions in Eq. (1), we propose to use the following vector-valued equation for surface blending with positional and tangential continuities.

$$\alpha\frac{d^4\mathbf{G}(u)}{du^4} + \beta\frac{d^2\mathbf{G}(u)}{du^2} + \gamma\mathbf{G}(u) = 0 \tag{2}$$

where $\mathbf{G}(u) = [G_x(u)\ \ G_y(u)\ \ G_z(u)]^T$ is a vector-valued function defining a generator used to create swept surfaces, and α, β and γ are called shape control parameters used as shape control handles for the generator.

The vector-valued ODE (2) can be changed into an algebraic equation by taking each component of the vector-valued function of the generator to be

$$G_\lambda(u) = e^{ru} \atop (\lambda = x, y, z) \tag{3}$$

Substituting Eq. (3) and the second and fourth derivatives of $\mathbf{G}(u)$ with respect to the parametric variable u into Eq. (2) and deleting e^{ru}, the following algebra equation is reached,

$$\alpha r^4 + \beta r^2 + \gamma = 0 \tag{4}$$

Depending on different combinations of shape control parameters, Eq. (4) has different solutions. Here we only give the solution for $\beta^2 = 4\alpha\gamma$ and $\alpha/\beta < 0$. All other solutions are given in Appendix A.

For $\beta^2 = 4\alpha\gamma$ and $\alpha/\beta < 0$, solving the nonlinear algebra equation (4), we obtain the following roots,

$$r_{1,2,3,4} = \pm q_1 \tag{5}$$

where

$$q_1 = \sqrt{-\beta/(2\alpha)} \tag{6}$$

With the roots given in Eq. (5), the solution to Eq. (2) can be written as,

$$\mathbf{G}(u) = \mathbf{d}_1 e^{q_1 u} + \mathbf{d}_2 u e^{q_1 u} + \mathbf{d}_3 e^{-q_1 u} + \mathbf{d}_4 u e^{-q_1 u} \tag{7}$$

where $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{d}_3$ and $\mathbf{d}_4$ are vector-valued unknown constants.

In order to determine the unknown constants in Eq. (7), we substitute it into Eq. (1), conduct the sweeping operation by

solving for the four vector-valued unknown constants $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{d}_3$ and $\mathbf{d}_4$, and obtain,

$$\begin{aligned}
\mathbf{d}_1 &= [a_{22}\mathbf{f}_1(v) - a_{12}\mathbf{f}_2(v)]/a_0 \\
\mathbf{d}_2 &= [a_{11}\mathbf{f}_2(v) - a_{21}\mathbf{f}_1(v)]/a_0 \\
\mathbf{d}_3 &= \mathbf{C}_0(v) - [a_{22}\mathbf{f}_1(v) - a_{12}\mathbf{f}_2(v)]/a_0 \\
\mathbf{d}_4 &= \overline{\mathbf{C}}_0(v) + q_1\mathbf{C}_0(v) - 2q_1[a_{22}\mathbf{f}_1(v) - a_{12}\mathbf{f}_2(v)]/a_0 \\
&\quad - [a_{11}\mathbf{f}_2(v) - a_{21}\mathbf{f}_1(v)]/a_0
\end{aligned} \tag{8}$$

where,

$$a_0 = a_{11}a_{22} - a_{12}a_{21} \tag{9}$$

and,

$$\begin{aligned}
a_{11} &= e^{q_1} - (1 + 2q_1)e^{-q_1} \\
a_{12} &= e^{q_1} - e^{-q_1} \\
a_{21} &= q_1 e^{q_1} - q_1 e^{-q_1} + 2q_1^2 e^{-q_1} \\
a_{22} &= (1 + q_1)e^{q_1} - (1 - q_1)e^{-q_1} \\
\mathbf{f}_1(v) &= -(1 + q_1)e^{-q_1}\mathbf{C}_0(v) - e^{-q_1}\overline{\mathbf{C}}_0(v) + \mathbf{C}_1(v) \\
\mathbf{f}_2(v) &= [q_1 e^{-q_1} - (1 - q_1)q_1 e^{-q_1}]\mathbf{C}_0(v) - (1 - q_1)e^{-q_1}\overline{\mathbf{C}}_0(v) + \overline{\mathbf{C}}_1(v)
\end{aligned} \tag{10}$$

Substituting Eq. (8) back into Eq. (7), the mathematical equation of swept surfaces for $\beta^2 = 4\alpha\gamma$ and $\alpha/\beta < 0$ can be written as,

$$\begin{aligned}
\mathbf{S}(u,v) &= \{e^{-q_1 u} + q_1 u e^{-q_1 u} - (1 + q_1)e^{-q_1}g_1(u) + [q_1 e^{-q_1} - (1 - q_1) \\
&\quad q_1 e^{-q_1}]g_2(u)\}\mathbf{C}_0(v) + [ue^{-q_1 u} - e^{-q_1}g_1(u) - (1 - q_1) \\
&\quad e^{-q_1}g_2(u)]\overline{\mathbf{C}}_0(v) + g_1(u)\mathbf{C}_1(v) + g_2(u)\overline{\mathbf{C}}_1(v)
\end{aligned} \tag{11}$$

where,

$$\begin{aligned}
g_1(u) &= \left\{q_1\left[e^{q_1} + (2q_1 - 1)e^{-q_1}\right]u(e^{-q_1 u} - e^{q_1 u}) + \left[(1 + q_1)e^{q_1} \right.\right. \\
&\quad \left.\left. - (1 - q_1)e^{-q_1}\right]\left[e^{q_1 u} - (1 + 2q_1 u)e^{-q_1 u}\right]\right\}/A_0 \\
g_2(u) &= \left\{\left[e^{q_1} - (1 + 2q_1)e^{-q_1}\right]u(e^{q_1 u} - e^{-q_1 u}) + 2q_1\left(e^{q_1} - \right.\right. \\
&\quad \left.\left. e^{-q_1}\right)ue^{-q_1 u} + \left(e^{q_1} - e^{-q_1}\right)\left(e^{-q_1 u} - e^{q_1 u}\right)\right\}/A_0
\end{aligned} \tag{12}$$

and,

$$\begin{aligned}
A_0 &= \left[e^{q_1} - (1 + 2q_1)e^{-q_1}\right]\left[(1 + q_1)e^{q_1} - (1 - q_1)e^{-q_1}\right] \\
&\quad - \left(e^{q_1} - e^{-q_1}\right)\left[q_1 e^{q_1} - q_1 e^{-q_1} + 2q_1^2 e^{-q_1}\right]
\end{aligned} \tag{13}$$

With the same treatment, we derive other analytical solutions (A4), (A9) and (A15) of Eq. (2) in Appendix A. Using these obtained solutions (11), (A4), (A9) and (A15), we can tackle various surface blending problems with effective shape control and exact satisfaction of positional and tangential continuities at trimlines. We will demonstrate this with some examples given in the following sections.

Since the boundary constraints $\mathbf{C}_0(v)$, $\overline{\mathbf{C}}_0(v)$, $\mathbf{C}_1(v)$ and $\overline{\mathbf{C}}_1(v)$ are explicitly incorporated in the mathematical expressions (11), (A4), (A9) and (A15) of the blending surfaces, the main task of surface blending is to determine the boundary curves $\mathbf{C}_0(v)$ and $\mathbf{C}_1(v)$ and first partial derivatives $\overline{\mathbf{C}}_0(v)$ and $\overline{\mathbf{C}}_1(v)$ at the boundary curves which can be readily obtained from the primary surfaces as discussed at the beginning of this section. Therefore, our proposed surface blending method is easy to use.

Once the blending boundary constraints at the trimlines are known, blending surfaces are analytically determined from

one of Eqs. (11), (A4), (A9) and (A15), leading to high computational efficiency. Even boundary constraints are represented at the discrete points at the trimlines, our proposed method can also construct blending surfaces quickly because of the explicit mathematical expressions of blending surfaces.

4 Implementation and validation

In this section, we implement our proposed method and validate it with various surface blending tasks.

The first task is to create a blending surface between an open surface and a closed conic surface. This example is used to validate Eq. (11) in surface blending. Denoting $\mathbf{S} = \mathbf{S}(u,v) = \begin{bmatrix} S_x & S_y & S_y \end{bmatrix}^T$, the boundary constraints for this blending task can be written as,

$$
\begin{aligned}
u = 0 \quad & S_x = 0.1\sinh(17v + 0.1) + 1.1297\sin v & \frac{\partial S_x}{\partial u} = -0.027\sin v \\
& S_y = 0.3\cosh 0.3v + 1.1297\cos v & \frac{\partial S_y}{\partial u} = -0.027\cos v \\
& S_z = -1.5 + e^{0.3} & \frac{\partial S_z}{\partial u} = -0.5e^{0.3} \\
u = 1 \quad & S_x = 0.8\sin v & \frac{\partial S_x}{\partial u} = 0.2\sin v \\
& S_y = 0.8\cos v & \frac{\partial S_y}{\partial u} = 0.2\cos v \\
& S_z = -1.9 & \frac{\partial S_z}{\partial u} = -3.6
\end{aligned}
\tag{14}
$$

Substituting the vector-valued functions $\mathbf{C}_0(v)$, $\overline{\mathbf{C}}_0(v)$, $\mathbf{C}_1(v)$ and $\overline{\mathbf{C}}_1(v)$ determined by Eq. (14) into Eq. (11), and setting shape control parameters to: $\alpha = \gamma = 1$, and $\beta = -2$, the blending surface is obtained and depicted in Fig. 1 where Fig. 1(b) and Fig. 1(c) are from different views of the blending surface in Fig. 1(a). The images given in Figure 1 clearly show that the upper and bottom primary surfaces are smoothly connected together through the in-between blending surface which demonstrates successful applications of Eq. (11) in surface blending.

The second blending task is to generate a blend between a circular torus and an elliptic hyperboloid. This blending task is used to validate Eq. (A4) in surface blending. The boundary constraints for this blending task are
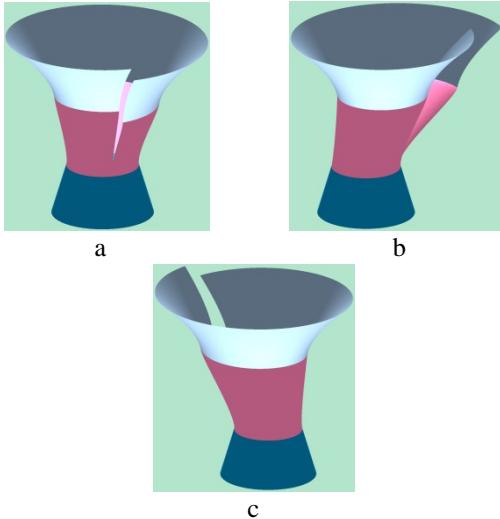

Fig. 1 Blending surface constructed with Eq. (11)

$$
\begin{aligned}
u = 0 \quad & S_x = (1.7 + 0.45\cos 0.85)\cos v & \frac{\partial S_x}{\partial u} = 2.45\sin 0.85\cos v \\
& S_y = (1.7 + 0.45\cos 0.85)\sin v & \frac{\partial S_y}{\partial u} = 2.45\sin 0.85\sin v \\
& S_z = 0.45\sin 0.85 & \frac{\partial S_z}{\partial u} = -2.45\cos 0.85 \\
u = 1 \quad & S_x = 1.5\cosh 0\cos v & \frac{\partial S_x}{\partial u} = (\sinh 0 + \cosh 0)\cos v \\
& S_y = \cosh 0\sin v & \frac{\partial S_y}{\partial u} = 0.7(\sinh 0 + \cosh 0)\sin v \\
& S_z = -1.5\sinh 0 & \frac{\partial S_z}{\partial u} = -1.5\cosh 0
\end{aligned}
\tag{15}
$$

For this blending task, we set the shape control parameters to: $\alpha = \gamma = 1$ and $\beta = 2$. The obtained blending surface was depicted in Fig. 2 where Fig. 2(a) is from the front view and Fig. 2(b) is from the side view.
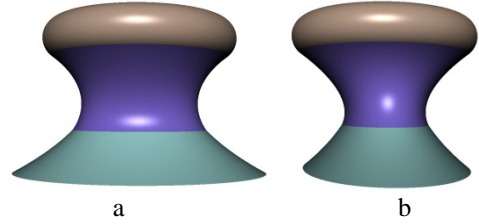

Fig. 2 Blending surface constructed with Eq. (A4)

The third example is to construct a blending surface between two conic frustums: one has some wrinkles on its surface and the other has a circular cross section. This example is used to validate Eq. (A9) in surface blending. The boundary constraints for this blending task are,

$$
\begin{aligned}
u = 0 \quad & S_x = 1.013[0.909\cos v + 0.051\cos(12v)] \\
& \frac{\partial S_x}{\partial u} = 0.26[0.909\cos v + 0.051\cos(12v)] \\
& S_y = 1.013[0.909\sin v + 0.051\sin(12v)] \\
& \frac{\partial S_y}{\partial u} = 0.26[0.909\sin v + 0.051\sin(12v)] \\
& S_z = 0.95 & \frac{\partial S_z}{\partial u} = -2.1 \\
u = 1 \quad & S_x = 1.3\cos v & \frac{\partial S_x}{\partial u} = -0.5\cos v \\
& S_y = 1.3\sin v & \frac{\partial S_y}{\partial u} = -0.5\sin v \\
& S_z = -1.2 & \frac{\partial S_z}{\partial u} = -2
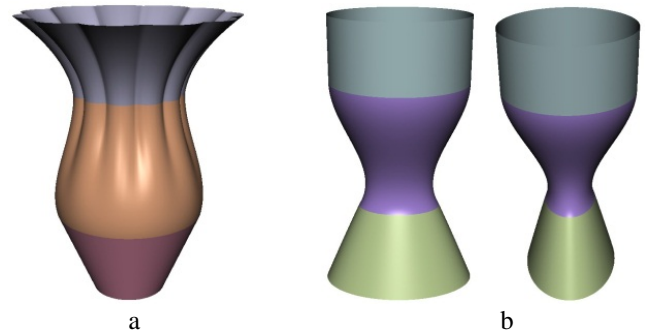\end{aligned}
\tag{16}
$$

Fig. 3 Blending surface constructed with a). Eq. (A9), and b). Eq. (A15)

Setting the shape control parameters to: $\alpha = \gamma = 1$ and $\beta = 3$, the obtained blending surface was depicted in Fig. 3a.

The fourth example is to produce a blending surface between a circular cylinder and an elliptic cylinder of two

sheets. This example is used to validate Eq. (A15) in surface blending. The boundary constraints for this blending task can be written as,

$$
\begin{array}{llll}
u = 0 & S_x = 0.9\cos v & & \dfrac{\partial S_x}{\partial u} = 0 \\[2mm]
& S_y = 0.9\,\sin v & & \dfrac{\partial S_y}{\partial u} = 0 \\[2mm]
& S_z = 0.4 & & \dfrac{\partial S_z}{\partial u} = -1.8 \\[2mm]
u = 1 & S_x = 0.6\sinh 1.4\cos v & & \dfrac{\partial S_x}{\partial u} = 0.6\cosh 1.4\cos v \\[2mm]
& S_y = 0.4\sinh 1.4\sin v & & \dfrac{\partial S_y}{\partial u} = 0.4\cosh 1.4\sin v \\[2mm]
& S_z = -1.5\cosh 1.4 & & \dfrac{\partial S_z}{\partial u} = -1.35\sinh 1.4
\end{array} \tag{17}
$$

With Eqs. (A15) and (17), and setting the shape control parameters to: $\alpha = \beta = \gamma = 1$, the generated blending surface is given in Fig. 3b where the left one is from the front view and the right one is from the side view.

The last example is to blend two intersecting cylinders. It is used to demonstrate the application of our proposed approach in complicated surface blending tasks. The boundary constraints for this blending task are

$$
\begin{array}{ll}
u = 0 & S_x = \xi_1 \cos v \qquad\qquad \dfrac{\partial S_x}{\partial u} = 0 \\[2mm]
& S_y = \xi_1 \sin v \qquad\qquad \dfrac{\partial S_y}{\partial u} = 0 \\[2mm]
& S_z = \sqrt{(\xi_2 + \xi_3)^2 - \xi_1^2 \cos^2 v} \\[2mm]
& \dfrac{\partial S_z}{\partial u} = \dfrac{\xi_2 + \xi_3}{\sqrt{(\xi_2 + \xi_3)^2 - \xi_1^2 \cos^2 v}} \\[2mm]
u = 1 & S_x = (\xi_1 + \xi_4)\cos v \qquad \dfrac{\partial S_x}{\partial u} = \cos v \\[2mm]
& S_y = (\xi_1 + \xi_4)\sin v \qquad \dfrac{\partial S_y}{\partial u} = \sin v \\[2mm]
& S_z = \sqrt{\xi_2^2 - (\xi_1 + \xi_4)^2 \cos^2 v} \\[2mm]
& \dfrac{\partial S_z}{\partial u} = -\dfrac{\zeta \cos^2 v}{\sqrt{\xi_2^2 - (\xi_1 + \xi_4)^2 \cos^2 v}}
\end{array} \tag{18}
$$

where ξ_1, ξ_2, ξ_3 and ξ_4 are the geometric parameters defining the two boundary curves.
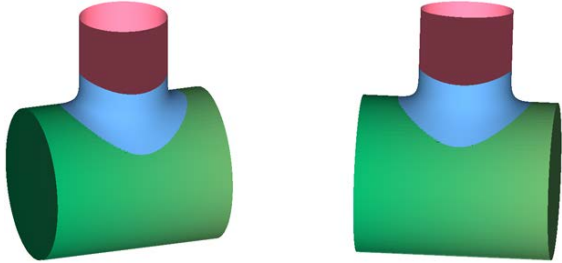

Fig. 4 Blending surface between two intersecting cylinders

Using Eqs. (11) and (18), and setting the shape control parameters to: $\alpha = \gamma = 1$ and $\beta = -2$, the generated blending surface is shown in Fig. 4. This example indicates that our proposed approach solves the complex surface blending problem successfully.

5 Shape control of blending surfaces

The biggest advantage of our proposed blending approach is that it not only controls the shape of the blending surfaces effectively but also satisfies the same blending boundary constraints exactly. We will use an example below to demonstrate this. This example is to blend a petal-like surface to the frustum of an elliptic cone. The boundary constraints for this example are,

$$
\begin{array}{ll}
u = 0 & S_x = 1.015(0.9\cos v + 0.1\cos 7v) \\[2mm]
& \dfrac{\partial S_x}{\partial u} = 0.02(0.9\cos v + 0.1\cos 7v) \\[2mm]
& S_y = 1.015(0.9\sin v + 0.1\sin 7v) \\[2mm]
& \dfrac{\partial S_y}{\partial u} = 0.02(0.9\sin v + 0.1\sin 7v) \\[2mm]
& S_z = 1.65 \qquad\qquad \dfrac{\partial S_z}{\partial u} = -0.1 \\[2mm]
u = 1 & S_x = 1.6\cos v \qquad\qquad \dfrac{\partial S_x}{\partial u} = 1.2\cos v \\[2mm]
& S_y = 1.2\sin v \qquad\qquad \dfrac{\partial S_y}{\partial u} = 1.2\sin v \\[2mm]
& S_z = 0 \qquad\qquad\qquad \dfrac{\partial S_z}{\partial u} = -1.8
\end{array} \tag{19}
$$

Inserting Eq. (19) into Eq. (A15), when we set the shape control parameters to: $\alpha = \gamma = 1$, and $\beta = 0.5$, the obtained blending surface was indicated in Fig. 5(a). If we changed these shape control parameters to: $\alpha = 3.7$, $\beta = 2.5$ and $\gamma = 1$, the blending surface given in Fig. 5(b) was produced. The blending surface shown in Fig. 5(c) was created by setting the shape control parameters to: $\alpha = 3.7$, $\beta = 2$ and $\gamma = 18$, and that presented in Fig. 5(d) was achieved by setting the shape control parameters to: $\alpha = 2$, $\beta = 0.5$, and $\gamma = 1$.
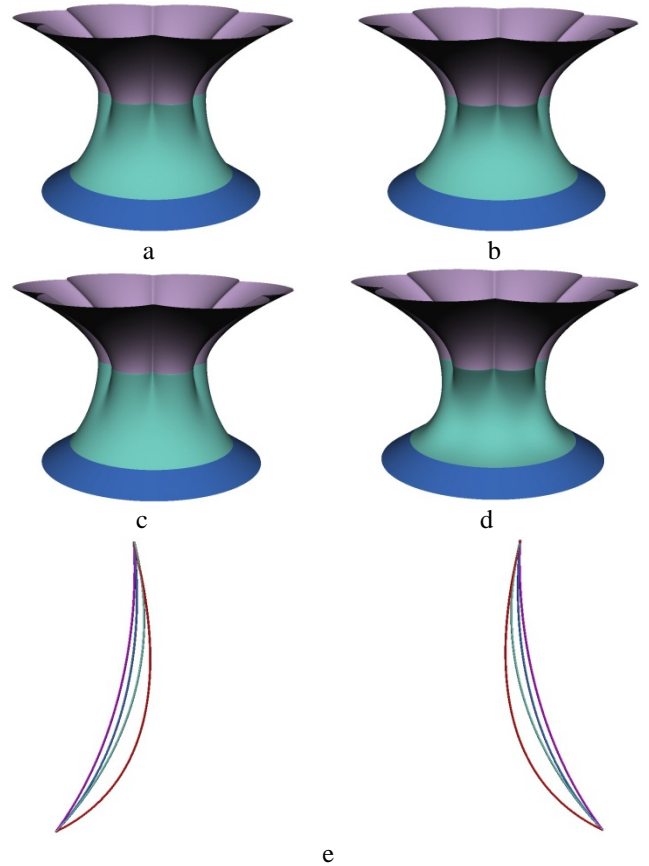

Fig. 5 Blending surfaces created with different shape control parameters (a) $\alpha = \gamma = 1$, $\beta = 0.5$, (b) $\alpha = 3.7$, $\beta = 2.5$, $\gamma = 1$, (c) $\alpha = 3.7$, $\beta = 2$, $\gamma = 18$, (d) $\alpha = 2$, $\beta = 0.5$, $\gamma = 1$, (e) profile curves of Fig. 4(a)-4(d)

In order to demonstrate the shape changes of the blending surface depicted in Fig. 5(a)-5(d) more clearly, we gave the profile curves of the blending surface in Fig. 5(e) where the profile curves in blue, green, purple and red are from Fig. 5(a), Fig. 5(b), Fig. 5(c) and Fig. 5(d), respectively.

The images in Fig. 5(a)-5(e) indicate that shape control parameters are very effective in controlling the shape of blending surfaces. By manipulating these shape control parameters, the shape of blending surfaces is controlled effectively while the continuities at the trimlines between the blending surface and primary surfaces keep unchanged due to the same blending boundary constraints.

Now, we investigate the relationships between shape control parameters and the shape of blending surfaces. In order to exclude the influence of blending boundary tangents on blending surfaces, we keep the blending boundary constraints (19) unchanged.

First, we fix the second and third shape control parameters $\beta = \gamma = 1$, take the first shape control parameter $\alpha = 1$, and obtain the red profile curve of the blending surface shown in Fig. 6(a). Next, we change the first shape control parameter to $\alpha = 2$, the green profile curve of the blending surface is generated. Further raising the shape control parameter to $\alpha = 2.5$, the blue profile curve of the blending surface is created. These profile curves indicate that raising the first shape control parameter, the whole blending surface becomes uniformly more concave.
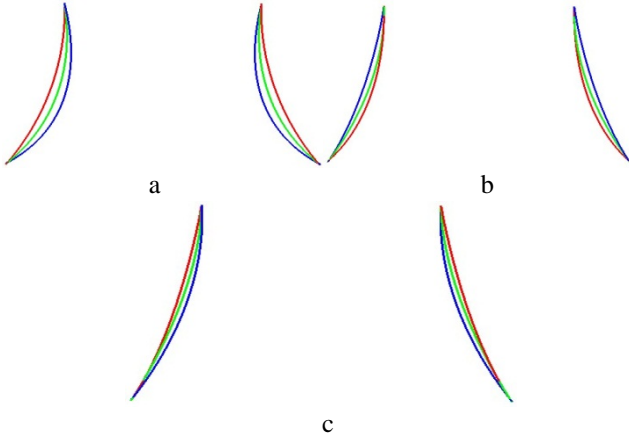

Fig. 6 Relationships between shape control parameters and the shape of blending surfaces (a) $\beta = \gamma = 1$, $\alpha = 1$ (red profile), $\alpha = 2$ (green profile), $\alpha = 2.5$ (blue profile), (b) $\alpha = \gamma = 1$, $\beta = 0.1$ (red profile), $\beta = 1.5$ (green profile), $\beta = 3$ (blue profile), (c) $\alpha = \beta = 1$, $\gamma = 0.1$ (red profile), $\gamma = 0.3$ (green profile), $\gamma = 0.5$ (blue profile)

Next, we fix the first and third shape control parameters $\alpha = \gamma = 1$. When the second shape control parameter is set to $\beta = 0.1$, the red profile curve indicated in Fig. 6(b) is produced. Increasing the parameter to $\beta = 1.5$ and $\beta = 3$, respectively, the green and blue profile curves of the blending surface are obtained. These profile curves demonstrate that increasing the second shape control parameter, the blending surface becomes less concave. When the increase of the parameter is small ($\beta = 1.5$), the influence mainly occurs in the lower part of the blending

surface. However, when the parameter is large ($\beta = 3$), the influence becomes more uniform.

Finally, we fix the first and second shape control parameters $\alpha = \beta = 1$, and change the third shape control parameter only. When the third shape control parameter is set to $\gamma = 0.1$, the achieved red profile curve of the blending surface is depicted in Fig. 6(c). Changing the third shape control parameter to 0.3 and 0.5, respectively, the green and blue profile curves are created and shown in the same figure. These profile curves suggest that increasing the third shape control parameter, the blending surface becomes more concave. When the increase is not big ($\gamma = 0.3$), the influence of the parameter on the top part of the blending surface is more significant. However, when the increase becomes larger ($\gamma = 0.5$), the influence changes to more uniform.

Apart from its advantage of effective shape manipulation and exact satisfaction of C^1 continuous boundary constraints, our method can also create blending surfaces quickly. On a laptop with a 1.66 GHz CPU and using 100×100 uniformly distributed surface points, our approach took 0.063 second to generate the blending surface in Fig. 5(a), suggesting that our proposed approach can create blending surfaces fast.

6 Blending among more than two primary surfaces

Our proposed approach is not only powerful in blending two primary surfaces, but also effective in constructing blending surfaces connecting more than two primary surfaces, which is usually a difficult task for existing surface blending approaches such as PDE-based surface blending. We demonstrate the effectiveness of our approach with an example below.

In this example, we blend three planes shown in Fig. 7(a). The blending operation is divided into two steps. In the first step, we construct blending surfaces between the top and front planes, between the top and side planes, and between the front and side planes. In the second step, we create a blending surface between the three constructed blending surfaces.

The boundary constraints for the surface blending between the top and front planes can be written in the following form,

$$
\begin{array}{llll}
u=0 & S_x(u,v)=l_x v & & \dfrac{\partial S_x(u,v)}{\partial u}=0 \\[2mm]
 & S_y(u,v)=y_0+r & & \dfrac{\partial S_y(u,v)}{\partial u}=-0.5l_y \\[2mm]
 & S_z(u,v)=z_0+l_z+r & & \dfrac{\partial S_z(u,v)}{\partial u}=0 \\[2mm]
u=1 & S_x(u,v)=l_x v & & \dfrac{\partial S_x(u,v)}{\partial u}=0 \\[2mm]
 & S_y(u,v)=y_0 & & \dfrac{\partial S_y(u,v)}{\partial u}=0 \\[2mm]
 & S_z(u,v)=z_0+l_z & & \dfrac{\partial S_z(u,v)}{\partial u}=-0.5l_z
\end{array}
\tag{20}
$$

The boundary constraints used to construct the blending surface between the top and side planes can be written as ,

$$
\begin{array}{lll}
u = 0 & S_x(u,v) = l_x v & \dfrac{\partial S_x(u,v)}{\partial u} = 0.5 l_x \\[2mm]
& S_y(u,v) = y_0 + r + l_y v & \dfrac{\partial S_y(u,v)}{\partial u} = 0.0 \\[2mm]
& S_z(u,v) = z_0 + l_z + r & \dfrac{\partial S_z(u,v)}{\partial u} = 0 \\[2mm]
u = 1 & S_x(u,v) = l_x + \mathrm{r} & \dfrac{\partial S_x(u,v)}{\partial u} = 0 \\[2mm]
& S_y(u,v) = y_0 + r + l_y v & \dfrac{\partial S_y(u,v)}{\partial u} = 0 \\[2mm]
& S_z(u,v) = z_0 + l_z & \dfrac{\partial S_z(u,v)}{\partial u} = -0.5 l_z
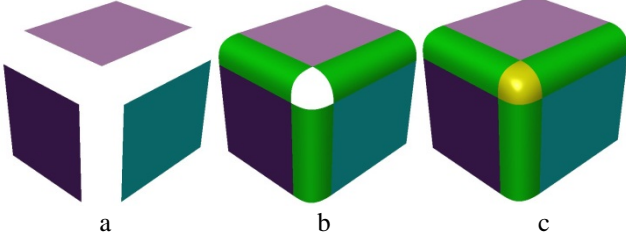\end{array}
\tag{21}
$$

Fig. 7 Blending among more than two separate primary surfaces

Similarly, the boundary constraints employed to construct the blending surface between the front and side planes take the form of,

$$
\begin{array}{lll}
u = 0 & S_x(u,v) = l_x & \dfrac{\partial S_x(u,v)}{\partial u} = 0.5 l_x \\[2mm]
& S_y(u,v) = y_0 & \dfrac{\partial S_y(u,v)}{\partial u} = 0.0 \\[2mm]
& S_z(u,v) = z_0 + l_z v & \dfrac{\partial S_z(u,v)}{\partial u} = 0 \\[2mm]
u = 1 & S_x(u,v) = l_x + \mathrm{r} & \dfrac{\partial S_x(u,v)}{\partial u} = 0 \\[2mm]
& S_y(u,v) = y_0 + r & \dfrac{\partial S_y(u,v)}{\partial u} = 0.5 l_y \\[2mm]
& S_z(u,v) = z_0 + l_z v & \dfrac{\partial S_z(u,v)}{\partial u} = 0
\end{array}
\tag{22}
$$

In the above equation, y_0, z_0, l_x, l_y, l_z and r are geometric parameters used to define the top, front and side planes.

Taking the geometric parameters in the above equations to be: $y_0 = l_x = l_z = 1$, $z_0 = 0.5$, $l_y = 1.2$ and $r = 0.25$, and setting the shape control parameters to: $\alpha = \gamma = 1$, and $\beta = 2$, we constructed the blending surfaces between these planes using Eq. (A4) and depicted these blending surfaces in Fig. 7(b).

Finally, we generate the blending surface between the three constructed blending surfaces shown in Fig. 7(b). This blending surface is a 3-sided patch. The boundary constraints for this 3-sided patch were determined below.

The top boundary at $u = 0$ for this blending problem is a point which is determined by setting $u = 0$ and $v = 1$ in Eq. (A4) of the blending surface between the top and front planes, i. e., $\mathbf{S}^{TF}(0,1)$ or by setting $u = 0$ and $v = 0$ in Eq. (A4) of the blending surface between the top and side planes, i. e., $\mathbf{S}^{TS}(0,0)$ where the superscripts TF and TS indicate the blending surface between the top and front planes and between the top and side planes, respectively. Although the top boundary of the 3-sided patch is a point, the boundary tangent changes continuously from that

determined by $\mathbf{S}^{TF}(u,v)$ to the one determined by $\mathbf{S}^{TS}(u,v)$, i. e., from $\mathbf{T}^0 = \partial \mathbf{S}^{TF}(0,1)/\partial u$ to $\mathbf{T}^1 = \partial \mathbf{S}^{TS}(0,0)/\partial u$. The boundary tangent $\mathbf{T}^t(v)$ at any position v can be obtained by interpolating $\mathbf{T}^0$ and $\mathbf{T}^1$ which is $\mathbf{T}^t(v) = \mathbf{T}^0 + (\mathbf{T}^1 - \mathbf{T}^0) l(v)$ where the superscript t indicates the top boundary and $l(v)$ is the normalized arc length of the bottom boundary curve.

The normalized arc length of the bottom boundary curve can be determined as follows. First, we calculate the total length of the bottom boundary curve with the equation $L = \int_0^1 \sqrt{[dS_x^{FS}(u,1)/du]^2 + [dS_y^{FS}(u,1)/du]^2 + [dS_z^{FS}(u,1)/du]^2}\, du$ where $\mathbf{S}^{FS}(u,1)$ is the vector-valued representation of the bottom boundary curve and the superscript FS indicates the blending surface between the front and side planes. Then, we calculate the length of the bottom boundary curve from $u = 0$ to any position u which is $L(u) = \int_0^u \sqrt{[dS_x^{FS}(u,1)/du]^2 + [dS_y^{FS}(u,1)/du]^2 + [dS_z^{FS}(u,1)/du]^2}\, du$. The normalized arc length $l(v)$ of the bottom boundary curve is determined by $l(v) = L(v)/L$ where the parametric variable u in $L(u)$ has been replaced by the parametric variable v.

The bottom boundary is a curve represented by $\mathbf{C}^b(u) = \mathbf{S}^{FS}(u,1)$ where the superscript b indicates the bottom boundary of the 3-sided blending surface. The boundary tangent at the bottom boundary is determined by $\mathbf{T}^b(u) = \partial \mathbf{S}^{FS}(u,1)/\partial v$.

When constructing the 3-sided blending surface, the parametric direction u for the bottom boundary is changed into the parametric direction v. Accordingly, the boundary curve and boundary tangent for the bottom boundary become $\mathbf{C}^b(v)$ and $\mathbf{T}^b(v)$.

With the above treatment, the boundary constraints for the 3-sided blending surface are represented by

$$
\begin{array}{llll}
u = 0 & \mathbf{S}(u,v) = \mathbf{S}^{TF}(0,1) & \dfrac{\partial \mathbf{S}(u,v)}{\partial u} = \mathbf{T}^t(v) \\[3mm]
u = 1 & \mathbf{S}(u,v) = \mathbf{C}^b(v) & \dfrac{\partial \mathbf{S}(u,v)}{\partial u} = \mathbf{T}^b(v)
\end{array}
\tag{23}
$$

Still using the same shape control parameters, we obtained the 3-sided blending surface and depicted it in Fig. 7(c). This example demonstrates that our proposed approach can blend more than two primary surfaces easily and effectively.

7 Conclusions

Based on swept surfaces controlled by a vector-valued fourth order ODE, we have developed a new surface blending method to achieve effective shape control of blending surfaces and exact satisfaction of both positional and tangential continuity constraints. With this method, the generator used to construct blending surfaces is created with the analytical solutions to the ODE and its shape is manipulated by the shape control parameters involved in the ODE. Blending surfaces are generated by sweeping the generator along two three-dimensional trajectories and making it exactly satisfy the positional and tangential constraints at the trimlines. A number of examples were

presented to demonstrate the applications of our proposed approach in surface blending.

Due to the analytical nature of our approach, it can generate various blending surfaces very efficiently. Since blending boundary constraints are explicitly included in the mathematical expressions of the blending surfaces, our proposed approach is easy to use. By making blending surfaces exactly meet the constraints of boundary curves and first partial derivatives at the boundary curves, C^1 continuous surface blending is achieved. The shape of the blending surfaces constructed with our proposed approach is controlled effectively by simply manipulating the shape control parameters while maintaining the same continuities at the trimlines. Apart from its capacity in creating a blending surface between two primary surfaces, it is also effective in blending more than two primary surfaces together.

References

1. Vida, J., Martin, R.R., Varady, T.: A survey of blending methods that use parametric surfaces. Computer-Aided Design 26(5), 341-365 (1994)
2. Rossignac, J.R., Requicha, A.A.G.: Constant-radius blending in solid modeling. Computers in Mechanical Engineering 3(1), 65-73 (1984)
3. Choi, B.K., Ju, S.Y.: Constant-radius blending in surface modeling. Computer-Aided Design 21(4), 213-220 (1989)
4. Barnhill, R.E., Farin, G.E., Chen, Q.: Constant-radius blending of parametric surfaces. Computing Supple. 8, 1-20, (1993)
5. Farouki, R.A.M., Sverrisson, R.: Approximation of rolling-ball blends for free-form parametric surfaces. Computer-Aided Design 28(11), 871-878 (1996)
6. Kós, G., Martin, R.R,, Várady, T.: Methods to recover constant radius rolling ball blends in reverse engineering. Computer Aided Geometric Design 17, 127-160 (2000)
7. Chuang, J.-H., Lin, C.-H., Hwang, W.-C.: Variable-radius blending of parametric surfaces. The Visual Computer 11, 513-525 (1995)
8. Chuang, J.H., Hwang, W.C.: Variable-radius blending by constrained spine generation. The Visual Computer 13, 316-329 (1997)
9. Lukács, G. Differential geometry of G^1 variable radius rolling ball blend surfaces. Computer Aided Geometric Design 15, 585-613 (1998)
10. Whited, B., Rossignac, J.: Relative blending. Computer-Aided Design 41, 456-462 (2009)
11. Krasauskas, R.: Branching blend of natural quadrics based on surfaces with rational offsets. Computer Aided Geometric Design 25, 332-341 (2008)
12. Zhou, P., Qian, W.-H.: A vertex-first parametric algorithm for polyhedron blending. Computer-Aided Design 41, 812-824 (2009)
13. Zhou, P., Qian, W.-H.: Polyhedral vertex blending with setbacks using rational S-patches. Computer Aided Geometric Design 27, 233-244 (2010)
14. Schichtel, M.: G^2 blend surfaces and filling of N-sided holes. IEEE Computer Graphics and Applications 13(9), 68-73 (1993)
15. Hsu, K.L., Tsay, D.M.: Corner blending of free-form N-sided holes. IEEE Computer Graphics and Applications 18(1), 72-78 (1998)
16. Piegl, L.A., Tiller, W.: Filling n-sided regions with NURBS patches. The Visual Computer 15(2), 77-89 (1999)
17. Li, G.Q., Li, H.: Blending parametric patches with subdivision surfaces. Journal of Computer Science and Technology 17(4), 498-506 (2002)
18. Hwang, W.C., Chuang, J.H.: N-sided hole filling and vertex blending using subdivision surfaces. Journal of Information Science and Engineering 19, 857-879 (2003)
19. Yang, Y.-J., Yong, J.-H., Zhang, H., Paul, J.-C., Sun, J.-G.: A rational extension of Piegl's method for filling n-sided holes. Computer-Aided Design 38(11), 1166-1178 (2006)
20. Shi, K.-L., Yong, J.-H., Sun, J.-G.: Filling n-sided regions with G^1 triangular Coons B-spline patches. The Visual Computer 26, 791-800 (2010)
21. Bloor, M.I.G., Wilson, M.J.: Generating blend surfaces using partial differential equations. Computer-Aided Design 21(3), 165-171 (1989)
22. Li, Z.C.: Boundary penalty finite element methods for blending surfaces, I. Basic theory. Journal of Computational Mathematics 16, 457-480 (1998)
23. Li, Z.C.: Boundary penalty finite element methods for blending surfaces, II. Biharmonic equations. Journal of Computational and Applied Mathematics 110, 155-176 (1999)
24. Li, Z.C.: Chang C-S. Boundary penalty finite element methods for blending surfaces, III, Superconvergence and stability and examples. Journal of Computational and Applied Mathematics 110, 241-270 (1999)
25. Bloor, M.I.G., Wilson, M.J., Mulligan, S.J.: Generating blend surfaces using a perturbation method. Mathematical and Computer Modeling 31(1), 1-13 (2000)
26. Bloor, M.I.G., Wilson, M.J.: An analytic pseudo-spectral method to generate a regular 4-sided PDE surface patch. Computer Aided Geometric Design 22, 203-219 (2005)
27. You, L.H., Zhang, J.J., Comninos, P.: Blending surface generation using a fast and accurate analytical solution of a fourth order PDE with three shape control parameters. The Visual Computer 20, 199-214 (2004)
28. You, L.H., Comninos, P., Zhang, J.J.: PDE blending surfaces with C^2 continuity. Computers & Graphics 28(6), 895-906 (2004)
29. Bloor, M.I.G., Wilson, M.J.: Functionality in solids obtained from partial differential equations. Computing 8, 21-42 (1993)

30. You, L.H., Chang, J., Yang, X.S., Zhang, J.J.: Solid modeling based on sixth order partial differential equations. Computer-Aided Design 43(6), 720-729 (2011)

31. You, L.H., Yang, X.S., You, X.Y., Jin, X., Zhang, J.J.: Shape manipulation using physically based wire deformations. Computer Animation and Virtual Worlds 21, 297-309 (2010)

32. You, L.H., Yang, X.S., Zhang, J.J.: Dynamic skin deformation with characteristic curves. Computer Animation and Virtual Worlds 19(3-4), 433-444 (2008)

33. Chaudhry, E., You, L.H., Jin, X., Yang, X.S., Zhang, J.J.: Shape modeling for animated characters using ordinary differential equations. Computers & Graphics 37, 638-644 (2013)

34. Koparkar, P.: Parametric blending using fanout surfaces. In Proceedings of Symposium on Solid Modeling Foundations and CAD/CAM Applications. Austin Texas, USA, 5-7 June, pp. 317-327. ACM Press, (1991)

Appendix A: Other analytical solutions of Eq. (2)

For $\beta^2 = 4\alpha\gamma$ and $\alpha/\beta > 0$, solving the nonlinear algebra equation (4), the following roots are found,

$$\eta_{1,2,3,4} = \pm iq_2 \tag{A1}$$

where i is an imaginary unit and,

$$q_2 = \sqrt{\beta/(2\alpha)} \tag{A2}$$

With the roots given in Eq. (A1), the analytical solution to Eq. (2) becomes,

$$G(u) = d_1 \cos q_2 u + d_2 \sin q_2 u + d_3 u \cos q_2 u + d_4 u \sin q_2 u \tag{A3}$$

where d_1, d_2, d_3 and d_4 are vector-valued unknown constants.

In order to determine the unknown constants in Eq. (A3), we perform the same sweeping operation by substituting it into Eq. (1), and solving for the four unknown constants d_1, d_2, d_3 and d_4. Then, we substitute the unknown constants back into Eq. (A3), and obtain,

$$
\begin{aligned}
S(u,v) = \big(& \cos q_2 u + A_2 \sin q_2 u - q_2 A_2 u \cos q_2 u - uctg q_2 \sin q_2 u \\
& + A_2 A_5 u \sin q_2 u \big) C_0(v) + \big(A_3 \sin q_2 u + u \cos q_2 u - q_2 A_3 u \cos q_2 u \\
& - uctg q_2 \sin q_2 u + A_3 A_5 u \sin q_2 u \big) \overline{C}_0(v) + \big(-A_4 \sin q_2 u + q_2 A_4 \\
& u \cos q_2 u + u \sin q_2 u / \sin q_2 - A_4 A_5 u \sin q_2 u \big) C_1(v) + \big(\sin q_2 u - q_2 \\
& u \cos q_2 u + A_5 u \sin q_2 u \big) \overline{C}_1(v) \big/ A_1
\end{aligned} \tag{A4}
$$

where,

$$
\begin{aligned}
A_1 &= q_2/\sin q_2 - \sin q_2 \\
A_2 &= \big[q_2 \sin q_2 + ctg q_2 (\sin q_2 + q_2 \cos q_2) \big] \big/ A_1 \\
A_3 &= \big[-(\cos q_2 - q_2 \sin q_2) + ctg q_2 (\sin q_2 + q_2 \cos q_2) \big] \big/ A_1 \\
A_4 &= (\sin q_2 + q_2 \cos q_2) \big/ (A_1 \sin q_2) \\
A_5 &= (q_2 \cos q_2 - \sin q_2) / \sin q_2
\end{aligned} \tag{A5}
$$

For $\beta^2 > 4\alpha\gamma$, solving the nonlinear algebra equation (4) gives the following roots,

$$
\begin{aligned}
\eta_{1,2} &= \pm iq_3 \\
r_{3,4} &= \pm iq_4
\end{aligned} \tag{A6}
$$

where i is an imaginary unit and,

$$
\begin{aligned}
q_3 &= \sqrt{\beta(1 - \sqrt{1 - 4\alpha\gamma/\beta^2})/(2\alpha)} \\
q_4 &= \sqrt{\beta(1 + \sqrt{1 - 4\alpha\gamma/\beta^2})/(2\alpha)}
\end{aligned} \tag{A7}
$$

With the roots given in Eq. (A7), the analytical solution to Eq. (2) takes the form of,

$$G(u) = d_1 \cos q_3 u + d_2 \sin q_3 u + d_3 \cos q_4 u + d_4 \sin q_4 u \tag{A8}$$

where d_1, d_2, d_3 and d_4 are vector-valued unknown constants.

Same as above, we substitute Eq. (A8) into Eq. (1), carry out the sweeping operation, and determine the four unknown constants d_1, d_2, d_3 and d_4. After substituting these unknown constants back to Eq. (A8), the mathematical equation of blending surfaces becomes,

$$
\begin{aligned}
S(u,v) = & \big[-g_3(u) \cos q_4 / A_6 - g_4(u) q_4 \sin q_4 / A_6 + \cos q_4 u \big] C_0(v) \\
& \big[-g_3(u) \sin q_4 / q_4 / A_6 + g_4(u) \cos q_4 / A_6 + \sin q_4 u / q_4 \big] \\
& \overline{C}_0(v) + g_3(u) C_1(v) \big/ A_6 - g_4(u) \overline{C}_1(v) / A_6
\end{aligned} \tag{A9}
$$

where,

$$
\begin{aligned}
A_6 = & q_3 (\cos q_3 - \cos q_4)^2 - (\sin q_3 - q_3 \sin q_4 / q_4) \\
& (q_4 \sin q_4 - q_3 \sin q_3)
\end{aligned} \tag{A10}
$$

and,

$$
\begin{aligned}
g_3(u) = & q_3 (\cos q_3 - \cos q_4)(\cos q_3 u - \cos q_4 u) + (q_4 \sin q_4 \\
& - q_3 \sin q_3)(q_3 \sin q_4 u / q_4 - \sin q_3 u) \\
g_4(u) = & (\cos q_3 - \cos q_4)(q_3 \sin q_4 u / q_4 - \sin q_3 u) + (\sin q_3 \\
& - q_3 \sin q_4 / q_4)(\cos q_3 u - \cos q_4 u)
\end{aligned} \tag{A11}
$$

For $\beta^2 < 4\alpha\gamma$, solving the nonlinear algebra equation (4) generates the four roots below,

$$\eta_{1,2,3,4} = \pm q_5 \pm iq_6 \tag{A12}$$

where i is an imaginary unit and,

$$
\begin{aligned}
q_5 &= \sqrt[4]{\gamma/\alpha} \sqrt{0.5 - \beta/(4\sqrt{\alpha\gamma})} \\
q_6 &= \sqrt[4]{\gamma/\alpha} \sqrt{0.5 + \beta/(4\sqrt{\alpha\gamma})}
\end{aligned} \tag{A13}
$$

With the roots given in Eq. (A13), the solution to Eq. (2) is found to be,

$$
\begin{aligned}
G(u) = & d_1 e^{q_5 u} \cos q_6 u + d_2 e^{q_5 u} \sin q_6 u + d_3 e^{-q_5 u} \cos q_6 u \\
& + d_4 e^{-q_5 u} \sin q_6 u
\end{aligned} \tag{A14}
$$

where d_1, d_2, d_3 and d_4 are vector-valued unknown constants.

Substituting Eq. (A14) into Eq. (1) and doing the sweeping operation, the 4 unknown constants are determined and blending surfaces satisfying Eqs. (1) and (2) are found to be,

$$
\begin{aligned}
S(u,v) = & \Big[-A_9 g_5(u) \big/ q_6 - (A_7 q_5 / q_6 - A_8) g_6(u) + e^{-q_5 u} \cos q_6 u \\
& + q_5 e^{-q_5 u} \sin q_6 u / q_6 \Big] C_0(v) + \Big[-A_7 g_6(u) \big/ q_6 - e^{-q_5} \\
& \sin q_6 g_5(u) / q_6 + e^{-q_5 u} \sin q_6 u / q_6 \Big] \overline{C}_0(v) + g_5(u) C_1(v) \\
& + g_6(u) \overline{C}_1(v)
\end{aligned} \tag{A15}
$$

where,

$$
\begin{aligned}
A_7 &= q_6 e^{-q_5} \cos q_6 - q_5 e^{-q_5} \sin q_6 \\
A_8 &= q_5 e^{-q_5} \cos q_6 + q_6 e^{-q_5} \sin q_6 \\
A_9 &= q_5 e^{-q_5} \sin q_6 + q_6 e^{-q_5} \cos q_6 \\
g_5(u) &= \Big\{ A_{12} \Big[\big(e^{-q_5 u} - e^{q_5 u} \big) \sin q_6 u \Big] + A_{13} \Big[\big(e^{q_5 u} - e^{-q_5 u} \big) \\
& \quad \cos q_6 u - 2 q_5 e^{-q_5 u} \sin q_6 u / q_6 \Big] \Big\} \Big/ A_{14} \\
g_6(u) &= \Big\{ A_{10} \Big[\big(e^{q_5 u} - e^{-q_5 u} \big) \sin q_6 u \Big] + A_{11} \Big[\big(e^{-q_5 u} - e^{q_5 u} \big) \\
& \quad \cos q_6 u + 2 q_5 e^{-q_5 u} \sin q_6 u / q_6 \Big] \Big\} \Big/ A_{14}
\end{aligned} \tag{A16}
$$

and,

$$A_{10} = \left(e^{q_5} - e^{-q_5}\right)\cos q_6 - 2q_5 e^{-q_5}\sin q_6 / q_6$$

$$A_{11} = \left(e^{q_5} - e^{-q_5}\right)\sin q_6$$

$$A_{12} = \left(q_5\cos q_6 - q_6\sin q_6\right)e^{q_5} + A_8 - 2q_5 A_7 / q_6 \qquad (A17)$$

$$A_{13} = \left(q_5\sin q_6 + q_6\cos q_6\right)e^{q_5} - A_9$$

$$A_{14} = A_{10}A_{13} - A_{11}A_{12}$$

Lihua You is currently an associate professor at the National Centre for Computer Animation, Bournemouth University, UK. He received his M.Sc. degree and Ph.D. degree from Chongqing University, China and another Ph.D. degree from Bournemouth University, UK. His current research interests are in computer graphics, computer animation, and geometric modeling.

Hassan Ugail is the director of the Centre for Visual Computing at Bradford. He has a first class BSc. Honours degree in Mathematics from King's College London and a Ph.D. in the field of Geometric Design from the School of Mathematics at University of Leeds. Prof. Ugail's research interests include geometric modeling, computer animation and functional design. Prof. Ugail has a number of patents on novel techniques relating to geometry modeling, animation and 3D data exchange. He is a reviewer for various international journals, conferences and grant awarding bodies. His recent innovations have led to the formation of a university spin-out company Tangentix Ltd, with investments from venture capitalists. He is also the recipient of the University of Bradford Vice Chancellor's Award for Knowledge Transfer which he received for his outstanding contribution to research and knowledge transfer activities.

Dr. Baoping Tang received his B.Sc. degree from Chongqing Technology University, China, in 1993, and received his MSc. degree and Ph.D. degree from Chongqing University, China, in 1996 and 2003, respectively. He is a professor at Chongqing University, China. His research interests include software engineering, virtual instrument and wireless sensor network.

Xiaogang Jin is a professor of the State Key Lab of CAD & CG, Zhejiang University. He received his B.Sc. degree in computer science in 1989, M.Sc. and Ph.D. degrees in applied mathematics in 1992 and 1995, all from Zhejiang University. His current research interests include implicit surface computing, special effects simulation, mesh fusion, texture synthesis, crowd animation, cloth animation and facial animation.

X. Y. You was awarded a B.Sc. degree in computer science by the University of Warwick, UK. He is interested in the research into computer graphics etc.

Jian J. Zhang is currently Professor of Computer Graphics at the National Centre for Computer Animation, Bournemouth University, UK and leads the Computer Animation Research Centre. He is also a cofounder of the UK's Centre for Digital Entertainment, funded by the Engineering and Physical Sciences Research Council. His research focuses on a number of topics relating to 3D virtual human modelling, animation and simulation, including geometric modelling, rigging and skinning, motion synthesis, deformation and physics-based simulation.