The Visual Computer

Blending surface generation using a fast and accurate analytical solution of a fourth-order PDE with three shape control parameters

Lihua You, Jian J. Zhang, Peter Comninos

National Centre for Computer Animation, Bournemouth Media School, Bournemouth University, Dorset BH12 5BB, United Kingdom
E-mail: jzhang@bournemouth.ac.uk

Published online: 27 April 2004

In this paper, we propose to use a fourth-order partial differential equation (PDE) to solve a class of surface-blending problems. This equation has three vector-valued shape control parameters. It incorporates all the previously published forms of fourth-order PDEs for surface blending and can generate a larger class of blending surfaces than existing equations. To apply the proposed PDE to the solution of various blending problems, we have developed a fast and accurate resolution method. Our method modifies Navier's solution for the elastic bending deformation of thin plates by making it satisfy the boundary conditions exactly. A comparison between our method, the closed-form solution method, and other existing analytical methods indicates that the developed method is able to generate blending surfaces almost as quickly and accurately as the closed-form solution method, far more efficiently and accurately than the numerical methods and other existing analytical methods. Having investigated the effects that the vector-valued shape parameters and the force function of the proposed equation have on the blending surface, we have found that they have a significant influence on its shape. They provide flexible user handles that surface designers can use to adjust the blending surface to acquire the desired shape. The developed method was employed in the investigation of surface-blending problems where the primary surfaces were expressed in parametric, implicit, and explicit forms.

Key words: Surface blending – Fourth-order partial differential equations – Fast and accurate PDE solution – Vector-valued shape parameters – Force function

Correspondence to: J.J. Zhang

1 Introduction

Surface modeling is a popular subject that includes a number of topics such as surface generation [2], surface blending [18] including hole filling [9], surface reconstruction [26], surface manipulation [10] and deformation [19]. In this paper, we present a new method for surface blending using a vector-valued partial differential equation (PDE).

Blending surfaces are used to join two or more primary surfaces smoothly and are extensively used in computer graphics, computer animation and computer-aided design. Their function is to satisfy aesthetic, functional or certain manufacturing requirements.

Many methods have been developed to solve a variety of surface-blending problems. Rossignac and Requicha identified four major categories of blends: blending surfaces generated by strong functional constraints, aesthetic blends, fairings, and rounds and fillets [15]. More recently, Vida et al. classified methods of constructing parametric blends as rolling-ball-based blends, spine-based blends, trimline-based blends, blends based on polyhedral methods and other methods including a cyclide solution, PDE-based blends and Fourier-based blends. Vida et al. felt that the PDE-based blending method was a powerful means of surface generation [18].

The PDE-based approach was initially proposed by Bloor and Wilson [1] and has since been enhanced and applied widely. This method regards a blending surface as the solutions to a set of PDEs. In their early work, Bloor et al. [1–4] proposed second- and fourth-order PDEs with one parameter and discussed the closed-form solutions of a few simple surface-blending problems. A second-order PDE cannot guarantee both position continuity and tangential continuity between the blending surface and the primary surfaces that it joins. This limitation, however, can be overcome with the fourth- or higher order PDEs. In their subsequent work, Bloor et al. concentrated on a fourth-order biharmonic-like PDE, which takes the form

$$\left(\frac{\partial^2}{\partial u^2}+\boldsymbol{a}^2\frac{\partial^2}{\partial v^2}\right)^2 \boldsymbol{X}(u,v)=0, \tag{1}$$

where $\boldsymbol{a}$ is a vector-valued parameter and $\boldsymbol{X}(u,v) = [x(u,v)y(u,v)z(u,v)]^T$ denotes a vector-valued function defined over some finite region of the (u,v) parameter space.

The Visual Computer (2004) 20:199–214
Digital Object Identifier (DOI) 10.1007/s00371-004-0241-7

Because of the difficulty of achieving a closed-form solution for the majority of practical problems, numerical resolution approaches were considered. These methods include the finite element method (FEM) [7], the finite difference method (FDM) [6, 8, 20, 21], and the collocation method [3]. Recently, Li and Li and Chang developed a boundary penalty finite element method for surface blending [11–13]. Equation 1 was also combined with the equation of motion in dynamics and solved with FDM for direct manipulation and interactive sculpting of PDE surfaces [10]. These numerical methods are effective in solving the PDE, unfortunately with a high computational cost. In interactive computer graphics and computer-aided design, as well as in other industrial applications that require high computational efficiency, such numerical methods are not applicable. To tackle this problem, Bloor and Wilson developed the Fourier series method and the pseudo-spectral method [2, 4]. When a small number of Fourier series terms are used, these methods are efficient. Due to this advantage, they were later used to develop an interactive computer graphics interface [16, 17] and have been applied as a major method for their case studies [5, 14]. However, as demonstrated in this paper, these two methods are not very accurate. Therefore, developing a fast and accurate method to generate PDE surfaces remains an unsolved problem.

The parameter $\boldsymbol{a}$ in PDE 1 has a strong influence on the shape of the surface generated. As there is only one parameter in Eq. 1, the flexibility it provides is limited. To create blending surfaces that satisfy a wider range of aesthetic and functional requirements, the introduction of more parameters in this equation becomes necessary. In our previous work, we proposed a more general fourth-order PDE for vase design [25] and discussed the suitability and efficiency of free-form surface generation using PDEs of different orders [24]. In addition, we have proposed a single-patch surface and applied it in the shape design of tools for metal forming [23].

As shown in this paper, the equation we propose can generate a set of blending surfaces that is a superset of the set of surfaces generated by the PDEs found in the literature. To make our equation applicable to interactive surface modelling, we will develop an efficient and accurate solving method for the proposed PDE. We will also examine the effect of the parameters and the force function of the proposed PDE on the shape of the generated blending surfaces. Finally, we will present a number of examples that demonstrate the application of the proposed PDE and its resolution method.

2 The partial differential equation

In this section we propose to use more parameters in the PDE that act as user handles, thus allowing the surface designer to achieve the desired shape for the blending surface. The proposed fourth-order PDE has three vector-valued shape control parameters and a force function and takes the form

$$\left(\boldsymbol{b}\frac{\partial^4}{\partial u^4}+\boldsymbol{c}\frac{\partial^4}{\partial u^2\partial v^2}+\boldsymbol{d}\frac{\partial^4}{\partial v^4}\right)\boldsymbol{X}(u,v)=\boldsymbol{f}(u,v), \quad (2)$$

where $\boldsymbol{b}=[b_x b_y b_z]^T$, $\boldsymbol{c}=[c_x c_y c_z]^T$ and $\boldsymbol{d}=[d_x d_y d_z]^T$ are vector-valued shape parameters and $\boldsymbol{f}(u,v)=[f_x(u,v),\, f_y(u,v),\, f_z(u,v)]^T$ is a vector-valued force function.

In Eqs. 1 and 2, we have employed a vector operator that generates a new vector by taking the product of the corresponding elements of two vectors to be its elements. That is to say, if $\boldsymbol{e}=[e_1 e_2 e_3]^T$ and $\boldsymbol{g}=[g_1 g_2 g_3]^T$ are two vectors, we can obtain a new vector $\boldsymbol{h}=[h_1 h_2 h_3]^T$ with its elements being $h_i=e_i g_i (i=1,2,3)$.

Equation 2 is a more general form of Eq. 1 and includes all forms of the existing fourth-order PDEs used for surface generation. Equation 2 can be reduced to Eq. 1 by setting $b_i=1$, $c_i=2a_i^2$, $d_i=a_i^4$, (for $i=x,y,z$), and $\boldsymbol{f}(u,v)=[000]^T$.

The positional and tangential continuity conditions on the boundaries of the blending surface can be expressed in the following form:

$$\begin{aligned}\boldsymbol{X}(0,v) &= \boldsymbol{g}_1(v)\\ \boldsymbol{X}_u(0,v) &= \boldsymbol{g}_2(v)\\ \boldsymbol{X}(1,v) &= \boldsymbol{g}_3(v)\\ \boldsymbol{X}_u(1,v) &= \boldsymbol{g}_4(v),\end{aligned} \quad (3)$$

where $\boldsymbol{g}_i$ (for $i=1,2,3,4$) are known functions on the boundaries and $\boldsymbol{x}_u=\frac{\partial \boldsymbol{x}}{\partial u}$.

For some simple surface blending problems, the closed-form solutions of Eq. 2 under the boundary conditions of Eq. 3 are obtainable. However,

such solutions do not usually exist for the majority of complex blending problems. Thus numerical methods such as FEM and FDM are often used to find a solution. These numerical methods, however, are very time consuming and hence unsuitable for applications where computational efficiency is of paramount importance, such as in real-time graphics applications. Here, we will develop a new resolution method that is computationally efficient and yields accurate results.

The functions in the boundary conditions (Eq. 3) can be divided into two groups under which PDE 2 may or may not have a closed-form solution, i.e.,

$$\begin{aligned}
\boldsymbol{X}(0,v) &= \sum_{j=1}^{J} \boldsymbol{a}_{j1}\bar{\boldsymbol{g}}_j(\xi v) + \sum_{k=1}^{K} \boldsymbol{b}_{k1}\hat{\boldsymbol{g}}_k(v) \\
\boldsymbol{X}_u(0,v) &= \sum_{j=1}^{J} \boldsymbol{a}_{j2}\bar{\boldsymbol{g}}_j(\xi v) + \sum_{k=1}^{K} \boldsymbol{b}_{k2}\hat{\boldsymbol{g}}_k(v) \\
\boldsymbol{X}(1,v) &= \sum_{j=1}^{J} \boldsymbol{a}_{j3}\bar{\boldsymbol{g}}_j(\xi v) + \sum_{k=1}^{K} \boldsymbol{b}_{k3}\hat{\boldsymbol{g}}_k(v) \\
\boldsymbol{X}_u(1,v) &= \sum_{j=1}^{J} \boldsymbol{a}_{j4}\bar{\boldsymbol{g}}_j(\xi v) + \sum_{k=1}^{K} \boldsymbol{b}_{k4}\hat{\boldsymbol{g}}_k(v),
\end{aligned} \tag{4}$$

where $\sum_{j=1}^{J} \boldsymbol{a}_{ji}\bar{\boldsymbol{g}}_j(\xi v)(i = 1, 2, 3, 4)$ is one group under which PDE 2 may have a closed-form solution and $\sum_{k=1}^{K} \boldsymbol{b}_{ki}\hat{\boldsymbol{g}}_k(v)(i = 1, 2, 3, 4)$ is the other group subject to which PDE 2 does not have a closed-form solution.

Generally, when the particular solution corresponding to the force function is not taken into account, the closed-form solution of PDE 2 subject to the first group of functions in the boundary conditions (Eq. 4) can be written as the following unified form:

$$\bar{\boldsymbol{X}}(u,v) = \sum_{j=1}^{J} \boldsymbol{G}_j(u)\bar{\boldsymbol{g}}_j(\xi v). \tag{5}$$

Depending on the differential properties of function $\bar{\boldsymbol{g}}_j(\xi v)$, the unknown function $\boldsymbol{G}_j(u)$ has the following three mathematical expressions (the detailed derivations are given in the appendix).

When $\frac{d^2\bar{\boldsymbol{g}}_j(\xi v)}{dv^2} = \frac{d^4\bar{\boldsymbol{g}}_j(\xi v)}{dv^4} = 0$ and without considering the force function, the unknown function $\boldsymbol{G}_j(u)$ can be written as

$$\boldsymbol{G}_j(u) = \sum_{k=1}^{4} \boldsymbol{c}_{jk}u^{k-1} \quad (j = 1, 2, \cdots, J), \tag{6}$$

where $c_{jk}(j = 1, 2, \cdots, J; k = 1, 2, 3, 4)$ are the unknown constants and can be determined by substituting Eq. 6 into Eq. 5, then substituting Eq. 5 into the boundary conditions (Eq. 4).

When $\frac{d^2\bar{\boldsymbol{g}}_j(\xi v)}{dv^2} = -\xi^2\bar{\boldsymbol{g}}_j(\xi v)$ and $\frac{d^4\bar{\boldsymbol{g}}_j(\xi v)}{dv^4} = \xi^4\bar{\boldsymbol{g}}_j(\xi v)$, and $4\boldsymbol{bd} \le \boldsymbol{c}^2$, the unknown function $\boldsymbol{G}_j(u)$ can be written as

$$\begin{aligned}
\boldsymbol{G}_j(u) &= \boldsymbol{c}_{j1}e^{r_{j1}u} + \boldsymbol{c}_{j2}e^{r_{j2}u} + \boldsymbol{c}_{j3}e^{r_{j3}u} + \boldsymbol{c}_{j4}e^{r_{j4}u} \\
&\quad \text{for } 4\boldsymbol{bd} < \boldsymbol{c}^2 \\
\boldsymbol{G}_j(u) &= (\boldsymbol{c}_{j1} + \boldsymbol{c}_{j2}u)e^{r_{j1}u} + (\boldsymbol{c}_{j3} + \boldsymbol{c}_{j4}u)e^{r_{j2}u} \\
&\quad \text{for } 4\boldsymbol{bd} = \boldsymbol{c}^2,
\end{aligned} \tag{7}$$

where the unknown constants $c_{jk}(j = 1, 2, \cdots, J; k = 1, 2, 3, 4)$ can be determined in the same way as above.

When $\frac{d^2\bar{\boldsymbol{g}}_j(\xi v)}{dv^2} = \xi^2\bar{\boldsymbol{g}}_j(\xi v)$ and $\frac{d^4\bar{\boldsymbol{g}}_j(\xi v)}{dv^4} = \xi^4\bar{\boldsymbol{g}}_j(\xi v)$, and $4\boldsymbol{bd} \le \boldsymbol{c}^2$, the unknown function $\boldsymbol{G}_j(u)$ can be written as

$$\begin{aligned}
\boldsymbol{G}_j(u) &= \boldsymbol{c}_{j1}\cos r_{j1}u + \boldsymbol{c}_{j2}\sin r_{j1}u \\
&\quad + \boldsymbol{c}_{j3}\cos r_{j2}u + \boldsymbol{c}_{j4}\sin r_{j2}u \ \text{ for } 4\boldsymbol{bd} < \boldsymbol{c}^2, \\
\boldsymbol{G}_j(u) &= (\boldsymbol{c}_{j1} + \boldsymbol{c}_{j2}u)\cos r_{j1}u \\
&\quad + (\boldsymbol{c}_{j3} + \boldsymbol{c}_{j4}u)\sin r_{j1}u \ \text{ for } 4\boldsymbol{bd} = \boldsymbol{c}^2.
\end{aligned} \tag{8}$$

For all other cases except for those given above, Eq. 2 has no closed-form solution. The basic functions for these cases and the particular solution corresponding to the force function are classified into $\sum_{k=1}^{K} \boldsymbol{b}_{ki}\hat{\boldsymbol{g}}_k(v)$ $(i = 1, 2, 3, 4)$, and their approximate solutions are being sought. Here we propose a new approximate solution – the pseudo-Navier solution.

PDE 2 is very similar to the governing equation for the elastic bending deformation of a thin plate under lateral loading. However, the boundary conditions are different. For the elastic bending deformation of thin plates subject to simply supported boundary conditions, Navier's solution is very popular. This solution uses a double-sine series that can satisfy the simply supported boundary conditions of the plates. In surface blending, however, the boundary conditions of PDE 2 cannot be met by the double-sine series. Therefore,

we propose a modified Navier solution as follows:

$$\hat{\boldsymbol{X}}(u,v)=\sum_{k=1}^{K}(\boldsymbol{c}_{k0}+\boldsymbol{c}_{k1}u+\boldsymbol{c}_{k2}u^2+\boldsymbol{c}_{k3}u^3)\hat{\boldsymbol{g}}_k(v)+u(u-1)^2\sum_{m=1}^{M}\sum_{n=1}^{N}\boldsymbol{A}_{mn}\sin mu\sin nv. \tag{9}$$

Substituting Eq. 9 into the boundary conditions (Eq. 4) and comparing the coefficients of functions $\hat{\boldsymbol{g}}_k(v)$, we obtain four linear algebraic equations for each $\hat{\boldsymbol{g}}_k(v)$. Solving these equations, we determine the unknown constants $\boldsymbol{c}_{ki}(i=0,1,2,3)$ as follows:

$$\begin{aligned}&\boldsymbol{c}_{k0}=\boldsymbol{b}_{k1}\\&\boldsymbol{c}_{k1}=\boldsymbol{b}_{k2}\\&\boldsymbol{c}_{k2}=-3\boldsymbol{b}_{k1}-2\boldsymbol{b}_{k2}+3\boldsymbol{b}_{k3}-\boldsymbol{b}_{k4}\\&\boldsymbol{c}_{k3}=2\boldsymbol{b}_{k1}+\boldsymbol{b}_{k2}-2\boldsymbol{b}_{k3}+\boldsymbol{b}_{k4}\quad(k=1,2,\cdots,K).\end{aligned} \tag{10}$$

Then, we replace the unknown constants $\boldsymbol{c}_{ki}(i=0,1,2,3)$ of Eq. 9 with Eq. 10 and substitute Eq. 9 into PDE 2. Since Eq. 9 is not a closed-form solution of PDE 2 subject to functions $\sum_{k=1}^{K}\boldsymbol{b}_{ki}\hat{\boldsymbol{g}}_k(v)$ $(i=1,2,3,4)$, PDE 2 cannot be met exactly and a residual function emerges. After some mathematical manipulation, this residual function can be obtained as follows (notice it takes account of the force function):

$$\boldsymbol{R}(u,v)=\sum_{m=1}^{M}\sum_{n=1}^{N}\boldsymbol{B}(u,m,n)\boldsymbol{A}_{mn}\sin nv+\sum_{k=1}^{K}\boldsymbol{D}_k(u,v)-\boldsymbol{f}(u,v), \tag{11}$$

where

$$\begin{aligned}\boldsymbol{D}_k(u,v)=\;&\boldsymbol{d}(\boldsymbol{b}_{k1}+\boldsymbol{b}_{k2}u)\\&+\left[2\boldsymbol{c}\frac{\partial^2\boldsymbol{g}_k(v)}{\partial v^2}+\boldsymbol{d}u^2\frac{\partial^4\boldsymbol{g}_k(v)}{\partial v^4}\right]\\&\times(-3\boldsymbol{b}_{k1}-2\boldsymbol{b}_{k2}+3\boldsymbol{b}_{k3}-\boldsymbol{b}_{k4})\\&+\left[6\boldsymbol{c}u\frac{\partial^2\boldsymbol{g}_k(v)}{\partial v^2}+\boldsymbol{d}u^3\frac{\partial^4\boldsymbol{g}_k(v)}{\partial v^4}\right]\\&\times(2\boldsymbol{b}_{k1}+\boldsymbol{b}_{k2}-2\boldsymbol{b}_{k3}+\boldsymbol{b}_{k4})\end{aligned} \tag{12}$$

and

$$\begin{aligned}\boldsymbol{B}(u,m,n)=\;&\{\boldsymbol{b}[-12(3u-2)+u(u-1)^2m^2]m^2\\&+\boldsymbol{c}[-2(3u-2)+u(u-1)^2m^2]n^2\\&+\boldsymbol{d}u(u-1)^2n^4\}\sin nu\\&+\{\boldsymbol{b}[24-4(u-1)(3u-1)m^2]m\\&-2\boldsymbol{c}(3u-1)(u-1)mn^2\}\cos mu.\end{aligned} \tag{13}$$

Within the resolution region, uniformly choosing L collocation points and substituting the values of the u and v parametric variables at these collocation points into Eq. 11, we obtain the L residual values shown below:

$$\boldsymbol{R}(u_l,v_l)=\sum_{m=1}^{M}\sum_{n=1}^{N}\boldsymbol{B}(u_l,m,n)\boldsymbol{A}_{mn}\sin nv_l+\sum_{k=1}^{K}\boldsymbol{D}_k(u_l,v_l)-\boldsymbol{f}(u_l,v_l). \tag{14}$$

The unknown constants in the above equation are $\boldsymbol{A}_{mn}(m=1,2,\cdots M;\,n=1,2,\cdots,N)$. They can be determined by minimising these residual values. Following the treatment given in [22], we finally reach the following linear algebraic equations for x, y and z:

$$\boldsymbol{B}^T\boldsymbol{B}\boldsymbol{A}=\boldsymbol{B}^T\boldsymbol{D}. \tag{15}$$

Solving Eq. 15 for the unknown constants $\boldsymbol{A}_{mn}(m=1,2,\cdots M;\,n=1,2,\cdots,N)$ and substituting these back into Eq. 9, the solution to PDE 2 under functions $\sum_{k=1}^{K}\boldsymbol{b}_{ki}\hat{\boldsymbol{g}}_k(v)$ $(i=1,2,3,4)$ can be obtained. Superimposing the closed-form solution (Eq. 5) upon the pseudo-Navier solution (Eq. 9), the general solution of PDE 2 subject to boundary conditions (Eq. 3) is obtained:

$$\boldsymbol{X}(u,v)=\bar{\boldsymbol{X}}(u,v)+\hat{\boldsymbol{X}}(u,v). \tag{16}$$

For a given blending problem, boundary conditions (Eq. 3) are known. Therefore, we can use the general solution Eq. 16 to generate the blending surface.

3 Accuracy and efficiency

The solution we have developed satisfies the boundary conditions exactly, and the surface errors are minimised. Therefore, it should be accurate. In ad-

dition, since the solution is analytical in nature, it should be computationally efficient. In this section, we compare this solution to the closed-form solution and other existing analytical solutions and demonstrate its accuracy and efficiency.

To compare the method we have developed with the closed-form resolution method, we construct a blending problem whose closed-form solution of PDE 2 exists. Our task is to blend a circular cylinder with a nonclosed surface. The boundary conditions for this blending problem are:

$$\begin{array}{lll} u=0 & x=a_0+a_1v+a_2e^v & \dfrac{\partial x}{\partial u}=0 \\ & y=a_3+a_4v & \dfrac{\partial y}{\partial u}=0 \\ & z=h_0 & \dfrac{\partial z}{\partial u}=h_1 \\ u=1 & x=-r\cos v & \dfrac{\partial x}{\partial u}=0 \\ & y=r\sin v & \dfrac{\partial y}{\partial u}=0 \\ & z=0 & \dfrac{\partial z}{\partial u}=h_2, \end{array} \tag{17}$$

where $a_i (i = 0, 1, 2, 3, 4)$ and h_0 are the geometric parameters defining the upper boundary curve, r is the radius of the bottom boundary circle, and h_1 and h_2 are the parameters controlling the first derivatives on the boundary curves (trimlines).

According to Eq. 4, all functions in Eq. 17 should be classified as the first group functions under which the closed-form solution of PDE 2 exists. However, to investigate the accuracy and efficiency of our method, we here regard e^v as a function of the second group under which PDE 2 is solved with the pseudo-Navier method. Therefore, the general solution of PDE 2 can be written as

$$\begin{aligned} x &= \sum_{i=0}^{3}(c_{x1i}+c_{x2i}v)u^i + G_{x3}(u)\cos v + \sum_{i=0}^{3} c_{x4i}u^i e^v \\ &\quad + u(u-1)^2\sum_{m=1}^{M}\sum_{n=1}^{N} A_{xmn}\sin mu \sin nv \\ y &= \sum_{i=0}^{3}(c_{y1i}+c_{y2i}v)u^i + G_{y3}(u)\sin v \\ z &= \sum_{i=0}^{3} c_{z1i}u^i, \end{aligned} \tag{18}$$

where $G_{x3}(u)$ and $G_{y3}(u)$ take the form of Eq. 7 and the unknown constants in Eq. 18 are determined by substituting them into the boundary conditions (Eq. 17).

When using the pseudo-Navier method, the terms of the double-sine series are set to $M = N = 3$ and the number of collocation points is taken to be 49. Taking the shape parameters to be $b_i = 1$, $c_i = 2a_i^2$, $d_i = a_i^4$, $a_i = 3 \quad (i = x, y, z)$ and the geometric parameters in Eq. 17 to be $a_0 = -0.5$, $a_1 = 0.158$, $a_2 = 1.02 \times 10^{-4}$, $a_3 = 1.5$, $a_4 = -0.477$, $r = 1$ and $h_0 = -h_1 = -h_2 = 2$, we obtain the blending surface shown in Fig. 1a.

PDE 2 subject to function e^v in the boundary conditions (Eq. 17) is solved with the closed-form resolution method. Therefore, the x component in Eq. 18 is changed to

$$x = \sum_{i=0}^{3}(c_{x1i}+c_{x2i}v)u^i + G_{x3}(u)\cos v + G_{x4}(u)e^v, \tag{19}$$

where $G_{x4}(u)$ takes the form of Eq. 8.

With the same shape parameters and geometric parameters, we have generated the blending surface shown in Fig. 1b using the closed-form solution. Comparing Figs. 1a and b, we notice that there are no discernible differences between them. This indicates that the method we have developed is very close to the closed-form resolution.

When using the Fourier series method [2], functions v and e^v in boundary conditions (Eq. 17) must be expanded into the following Fourier series:

$$\begin{aligned} v &= \pi - \sum_{n=1}^{\infty}\frac{2}{n}\sin nv \\ e^v &= \frac{e^{2\pi}-1}{\pi}\Big[0.5+\sum_{n=1}^{\infty}\frac{1}{1+n^2}(\cos nv - n\sin nv)\Big]. \end{aligned} \tag{20}$$

Then, the blending surface can be created with the following Fourier series solution of PDE 1 whose unknown constants are determined by substituting the solution into the boundary conditions (Eq. 17) and making use of Eq. 20:

$$\begin{aligned} \boldsymbol{X}(u, v) &= \boldsymbol{A}_0(u) \\ &\quad + \sum_{n=1}^{N}[\boldsymbol{A}_n(u)\cos nv + \boldsymbol{B}_n(u)\sin nv], \end{aligned} \tag{21}$$

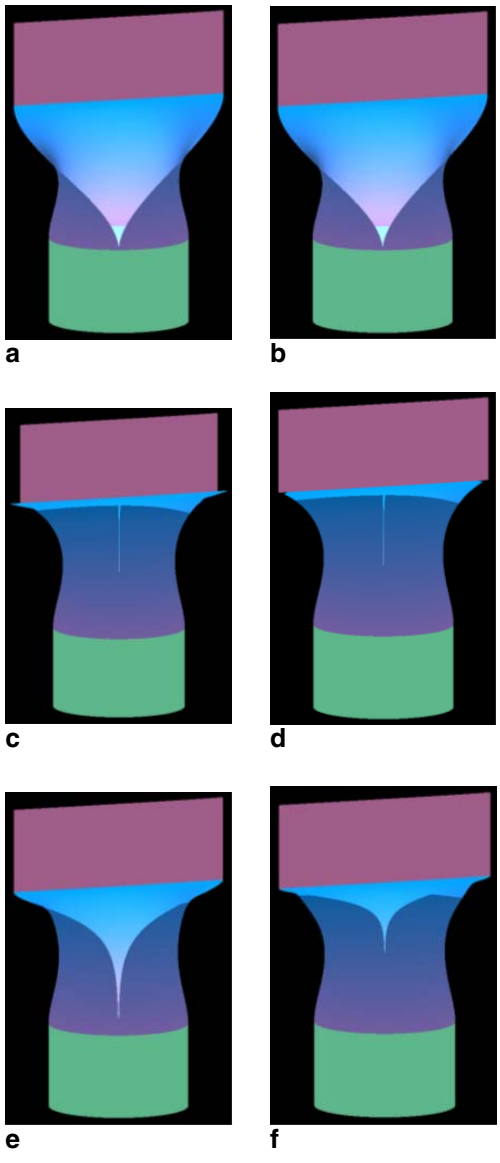


Fig. 1. Surface blending with different methods

where the concrete forms of the unknown functions $\boldsymbol{A}_0(u)$, $\boldsymbol{A}_n(u)$ and $\boldsymbol{B}_n(u)$ can be found in [4].

The vector-valued parameter in Eq. 1 was taken to be $a_i = 3$ $(i = x, y, z)$. For this vector-valued parameter, PDE 1 and PDE 2 with $b_i = 1, c_i = 2a_i^2, d_i = a_i^4$, $a_i = 3$ $(i = x, y, z)$ are identical. Under the same boundary conditions, they should produce identical blending surfaces. However, when using $N = 20$ for the number of terms of the Fourier series (Eq. 21), a different blending surface was generated as shown in Fig. 1c. Even when we increased the number of series terms to $N = 100$, the generated blending surface did not improve noticeably (Fig. 1d). It is thus obvious that since the boundary conditions were not met by the solution of Eq. 21, large errors occur at the boundary curve and on the entire surface.

To meet the boundary conditions exactly, Bloor and Wilson introduced a remainder function which can be written as [4]:

$$\boldsymbol{R}(u, v) = \boldsymbol{r}_1(v)e^{\omega u} + \boldsymbol{r}_2(v)ue^{\omega u} + \boldsymbol{r}_3(v)e^{-\omega u} + \boldsymbol{r}_4(v)ue^{-\omega u}. \tag{22}$$

This remainder function is then added to the Fourier series Eq. 21, which leads to the pseudo-spectral solution shown below:

$$\tilde{\boldsymbol{X}}(u, v) = \boldsymbol{X}(u, v) + \boldsymbol{R}(u, v). \tag{23}$$

The unknown functions $\boldsymbol{r}_i(v)$ $(i = 1, 2, 3, 4)$ in Eq. 22 were determined by making Eq. 23 satisfy the boundary conditions (Eq. 17) exactly.

Using the same shape and geometric parameters, when taking the number of terms of the Fourier series to be $N = 2$, the blending surface shown in Fig. 1e was generated. Unfortunately, this surface also shows discrepancy from the accurate shape shown in Fig. 1b. Increasing the number of terms of the Fourier series to $N = 5$, the blending surface shown in Fig. 1f was generated, which is a worse result when compared with the accurate surface of Fig. 1b. It indicates that increasing the number of terms of the Fourier series cannot improve the blending surface generated.

Although the introduction of the remainder function allows the boundary conditions to be satisfied exactly, PDE 2 has been violated, which leads to the visible surface error. Therefore, both the Fourier series method and the pseudo-spectral method are less accurate than the proposed method in solving PDE 1 and in blending surface generation.

The above observation can also be justified by a quantitative comparison. For this we introduce the following error function, which measures the relative average error between two surfaces:

$$E = \frac{1}{I_u \times J_v} \sum_{i=1}^{I_u} \sum_{j=1}^{J_v} \Big\{ \Big\{ [x(u_i, v_j) - \tilde{x}(u_i, v_j)]^2 + [y(u_i, v_j) - \tilde{y}(u_i, v_j)]^2 + [z(u_i, v_j) - \tilde{z}(u_i, v_j)]^2 \Big\}^{\frac{1}{2}} / \Big\{ [x(u_i, v_j)]^2 + [y(u_i, v_j)]^2 + [z(u_i, v_j)]^2 \Big\}^{\frac{1}{2}} \Big\}, \tag{24}$$

where (u_i, v_j) stands for the point used to calculate the relative average error of two surfaces and $I_u \times J_v$ is the number of points from which the error is determined.

Within the resolution region, $I_u = 100$ by $J_v = 100$ points are set, i.e., 10 000 uniformly distributed points within the blending surface are used to determine the relative average error of two surfaces given by Eq. 24 above. Compared to the closed-form solution, the error of the proposed pseudo-Navier solution is 6.637916×10^{-4}, while that of the pseudo-spectral method for $N = 2$ is 5.845711×10^{-2}, and for $N = 5$ it is 6.294766×10^{-2}; the error of the Fourier series method for $N = 20$ is 6.565263×10^{-2} and for $N = 100$ it is 6.513351×10^{-2}. Therefore, our proposed solution produces a much smaller error. In contrast, both the pseudo-spectral method and the Fourier series method bring in larger errors among which the error caused by the Fourier series method is the largest.

In addition to accuracy, the method we have developed is also very efficient. When using the closed-form resolution method, the time taken to determine all the unknown constants is less than 10^{-6} of a second on an 800-MHz PC. When using our pseudo-Navier method, the time taken to determine all the unknown constants in the given example ($M = N = 3$ and 49 collocation points) is also less than 10^{-6} of a second. This indicates that our method can create blending surfaces almost as quickly as the closed-form resolution method. For the Fourier series method, the time used to determine all the unknown constants is 0.032 s for $N = 20$ and 0.047 s for $N = 100$. For the pseudo-spectral method, the time used to determine all the unknown constants and the unknown functions $\boldsymbol{r}_i(v)(i = 1, 2, 3, 4)$ at 100 uniformly distributed points is 0.016 s for both $N = 2$ and $N = 5$. These data indicate that the proposed method is far more efficient than the pseudo-spectral method and the Fourier series method. Since numerical methods cannot achieve the same efficiency and precision as the closed- form solution, our method is also more accurate and efficient than the numerical methods.

4 Effects of shape parameters and force function on the shape of blending surface

In this section, we will study the effect of the shape parameters and force function of the proposed fourth-order PDE on the shape of the generated blending surface.

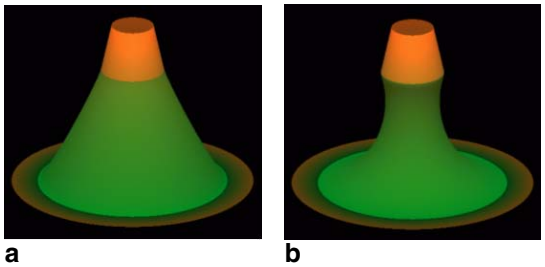
4.1 Shape parameters

The effect of the shape parameters on the surface shape will be demonstrated by blending the frustum of a circular cone with a plane containing a circular trimline. The boundary conditions for this blending surface can be written as

$$\begin{aligned}
u = 0 \quad & x = r_0 \cos v \quad && \frac{\partial x}{\partial u} = r_0' \cos v \\
& y = r_0 \sin v && \frac{\partial y}{\partial u} = r_0' \sin v \\
& z = 0 && \frac{\partial z}{\partial u} = 0 \\
u = 1 \quad & x = r_1 \cos v && \frac{\partial x}{\partial u} = r_1' \cos v \\
& y = r_1 \sin v && \frac{\partial y}{\partial u} = r_1' \sin v \\
& z = h_1 && \frac{\partial z}{\partial u} = h_1',
\end{aligned} \tag{25}$$

where r_0 and r_1 are the radii of the circular trimlines on the plane and at the bottom of the frustum of the cone, respectively, h_1 is the distance of the trimline of the conical frustum from the plane, and r_0', r_1' and h_1' are the parameters controlling the first derivatives on the trimlines.

The geometric parameters in this example are taken to be $r_0 = 1.8$, $r_0' = -0.5$, $r_1 = 0.6$, $r_1' = -0.2$, $h_1 = 2$ and $h_1' = 0.8$. Initially, we set the shape parameters of PDE 2 to be $b_x = b_y = d_x = d_y = 1$ and $c_x = c_y = 2$ and produce the blending surface depicted in Fig. 2a. Then, we change the shape parameter values to $b_x = b_y = 0.001$, $c_x = c_y = 0.1$ and generate the blending surface shown in Fig. 2b.



a b

Fig. 2. Effect of shape parameters on the shape of blending surfaces

It is evident that, using the same geometric parameters, different shape parameter values of PDE 2 produce blending surfaces with significantly different shapes. The proposed fourth-order PDE is able not only to reproduce all the blending surfaces generated by the equations proposed in [1] but also to generate a much richer class of blending surfaces, indicating that the fourth-order PDE with three vector-valued shape control parameters is more powerful than the other existing fourth-order PDEs.

4.2 Force function

To investigate the effect of the force function, we here consider the blending between a cylinder and a plane containing a specified circle subjected to given force functions. The blending will be performed in the parametric range of $0 \le u \le 1$ and $0 \le v \le 2\pi$. The boundary conditions for this blending task are given as

$$
\begin{array}{lll}
u = 0 & x = r_0 \cos v & \dfrac{\partial x}{\partial u} = 0 \\
 & y = r_0 \sin v & \dfrac{\partial y}{\partial u} = 0 \\
 & z = h & \dfrac{\partial z}{\partial u} = h' \\
u = 1 & x = r_1 \cos v & \dfrac{\partial x}{\partial u} = r_1' \cos v \\
 & y = r_1 \sin v & \dfrac{\partial y}{\partial u} = r_1' \sin v \\
 & z = 0 & \dfrac{\partial z}{\partial u} = 0,
\end{array}
\tag{26}
$$

where r_1 and r_0 are the radii of the circular trimlines on the plane and at the bottom of the cylinder, respectively, h represents the distance of the trimline of the cylinder from the plane, and r_1' and h' are the parameters controlling the first derivatives on the trimlines.

The solution of homogeneous Eq. 2 under boundary conditions (Eq. 26) can be obtained using our method. Here we only discuss the particular solution of Eq. 2 under some given force functions.

First, we set the force function on the right-hand side of Eq. 2 to

$$
\begin{aligned}
f_x &= p_0 \cos v \\
f_y &= p_0 \sin v \\
f_z &= 0,
\end{aligned}
\tag{27}
$$

where p_0 is a force constant.

The geometric parameters in this example are taken to be $r_0 = r_1' = 1$, $r_1 = 2$, $h = 2.5$ and $h' = -1$. The shape control parameters of Eq. 2 are taken to be $b_x = d_x = b_y = d_y = 1$ and $c_x = c_y = 2.5$. When no force function is applied (i.e., $p_0 = 0$), the generated blending surface is shown in Fig. 3a. When we set the force constant to $p_0 = 100$, the blending surface is subjected to a distributed tensile force. Therefore, the blending surface is pulled outwards, resulting in a convex shape as shown in Fig. 3b. Similarly, when we use $p_0 = -500$, a pushing force is acted on the surface. Consequently, the blending surface is pushed inwards, as seen in Fig. 3c.

Next, we discuss how different force functions influence the shape of blending surface. To do so, we change the force function in Eq. 27 to

$$
\begin{aligned}
f_x &= p_0(1 + \eta u^2) \cos v \\
f_y &= p_0(1 + \eta u^2) \sin v \\
f_z &= 0,
\end{aligned}
\tag{28}
$$

where η is a force parameter.

When we set $p_0 = 10$ and $\eta = 120$, a gradually increased distributed tensile force is applied as we move from the top of the blending surface toward the bottom. The force function pulls the bottom of the blending surface more strongly outwards, resulting in the shape shown in Fig. 3d. The shape of the blending surface depicted in Fig. 3e is arrived at by setting the force parameter $\eta = -100$. This parameter value causes a distributed pushing force to increase gradually as we move from the top of the blending surface toward the bottom. Figures 3d and e

show that the larger the forces which are applied at the bottom part of the blending surface, the larger the deformations.
The effect of the force function on the shape of the blending surface is very intuitive, hence allowing the surface designer to obtain the desired shape for a blending surface.

5 Applications of blending surfaces

All primary surfaces in practical use can be expressed in parametric, implicit or explicit mathematical representations. The objective of surface blending is to create the blending surfaces efficiently and effectively for these representations. In this section we give a number of examples to demonstrate the application of the proposed method. Where a primary surface is not initially represented parametrically, we will convert it into a parametric representation before applying blending operations.

Parametric primary surfaces

When faced with the problem of generating a blending surface between two primary surfaces whose mathematical representation is given in a parametric form, the first step is to determine the trimlines on the primary surfaces. This can frequently be done relatively easily by selecting values for a parametric variable in the primary surface equations. The tangential continuity conditions on the trimlines can also be obtained by finding the first partial derivatives of the surfaces on the trimlines. After the boundary conditions have been determined, the next step is to find the general solution of Eq. 2 under these boundary conditions. Below we give some examples to illustrate the application of the solution of Eq. 2 in such parametric blends.

Blending a cylinder with a plane containing a circular trimline

This example is similar to the surface-blending problem examined in the previous section where we used a force function. Here, however, no force function is used and the trimline on the cylinder is taken to be the space curve given below. The boundary conditions for this surface-blending problem are given by

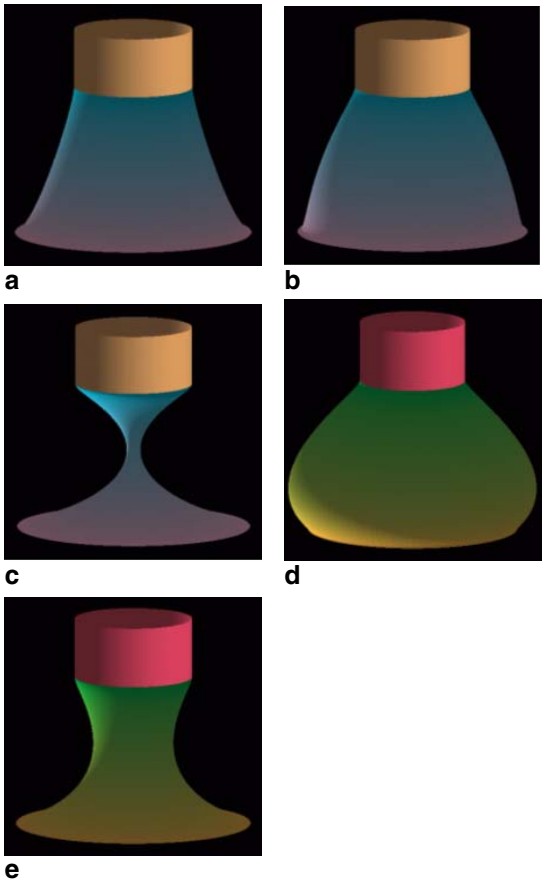


Fig. 3. Effect of force function on the shape of blending surface

$$
\begin{array}{lll}
u=0 & x=r\cos v & \dfrac{\partial x}{\partial u}=0 \\
 & y=r\sin v & \dfrac{\partial y}{\partial u}=0 \\
 & z=h_0+h_1\sin\dfrac{v}{2} & \dfrac{\partial z}{\partial u}=-\left(h_0'+h_1'\sin\dfrac{v}{2}\right) \\
u=1 & x=x_0+R\cos v & \dfrac{\partial x}{\partial u}=R'\cos v \\
 & y=y_0+R\sin v & \dfrac{\partial y}{\partial u}=R'\sin v \\
 & z=0 & \dfrac{\partial z}{\partial u}=0,
\end{array}
\tag{29}
$$

where r and R are the radii of the cylinder and the circle on the plane, x_0, y_0 are the co-ordinates of the centre of the circle on the plane, h_0 is the distance

of the lowest point, h_1 is the distance of the highest point of the trimline on the cylinder from the plane, and h'_0, h'_1, R' are the parameters controlling the first derivatives on these trimlines.
Setting $r = 0.8$, $R = 1.5$, $R' = 2$, $h_0 = h'_0 = 0.5$, $h_1 = h'_1 = 1$, $x_0 = y_0 = 0$ and shape control parameters $b_x = b_y = b_z = 0.5$, $c_x = c_y = c_z = 2$ and $d_x = d_y = d_z = 1$ in the solution generated by our method, we obtain the blending surface shown in Fig. 4a.

Blending an elliptic hyperboloid with an inclined plane

In this example we blend one sheet of an elliptic hyperboloid with a specified circle on an inclined plane. After some algebraic manipulation, the boundary conditions of the blending surface can be written as

$$\begin{array}{lll}
u=0 & x = a\sinh 1.2\cos v & \dfrac{\partial x}{\partial u} = a'\cosh 1.2\cos v \\
 & y = b\sinh 1.2\sin v & \dfrac{\partial y}{\partial u} = b'\cosh 1.2\sin v \\
 & z = h_0 + h_1\cosh 1.2 & \dfrac{\partial z}{\partial u} = h'_1\sinh 1.2 \\
u=1 & x = R\cos\alpha\cos v & \dfrac{\partial x}{\partial u} = R'\cos\alpha\cos v \\
 & y = R\sin v & \dfrac{\partial y}{\partial u} = R'\sin v \\
 & z = R\sin\alpha\cos v & \dfrac{\partial z}{\partial u} = R'\sin\alpha\cos v,
\end{array} \tag{30}$$

where a, b, and h_1 are the parameters defining the shape and size of the elliptic hyperboloid, h_0 defines its position along the height direction, R is the radius of the circle on the inclined plane, α is the angle of inclination of this plane with respect to the horizontal plane (where the angle is measured in a counterclockwise positive fashion) and a', b', h'_1 and R' are the parameters controlling the first derivatives on the trimlines.
Setting $R = R' = 1.5$, $a = 0.7$, $b = 0.4$, $h_0 = 0.3$, $h_1 = h'_1 = 0.8$, $a' = -0.7$, $b' = -0.4$, $\alpha = -30°$ and shape control parameters $b_x = b_y = b_z = d_x = d_y = d_z = 1$ and $c_x = c_y = c_z = 2$, we get the blending surface depicted in Fig. 4b.

Blending a primary surface whose trimline consists of sine and cosine functions with a plane at a prescribed ellipse

For this blending task, given that the solving region is over $\Omega : \{0 \le u \le 1;\ 0 \le v \le 2\pi\}$, the primary surface consisting of sine and cosine functions can be described by the following parametric equations:

$$\begin{aligned}
x &= (1+\xi u^2)(R_0\cos v + R_1\cos kv) \\
y &= (1+\xi u^2)(R_0\sin v + R_1\sin kv) \\
z &= h_0 + h_1 u
\end{aligned} \tag{31}$$

and the plane is assumed to be generated by the parametric equations

$$\begin{aligned}
x &= au\cos v \\
y &= bu\sin v \\
z &= 0.
\end{aligned} \tag{32}$$

Taking the trimline to be at $u = u_0$ on the primary surface (Eq. 31) and at $u = u_1$ on the plane (Eq. 32), the boundary conditions on these trimlines can be written as

$$\begin{array}{lll}
u=0 & x = (1+\xi u_0^2)(R_0\cos v + R_1\cos kv) & \\
 & \dfrac{\partial x}{\partial u} = 2\xi' u_0(R_0\cos v + R_1\cos kv) & \\
 & y = (1+\xi u_0^2)(R_0\sin v + R_1\sin kv) & \\
 & \dfrac{\partial y}{\partial u} = 2\xi' u_0(R_0\sin v + R_1\sin kv) & \\
 & z = h_0 + h_1 u_0 & \dfrac{\partial z}{\partial u} = h'_1 \\
u=1 & x = au_1\cos v & \dfrac{\partial x}{\partial u} = a'\cos v \\
 & y = bu_1\sin v & \dfrac{\partial y}{\partial u} = b'\sin v \\
 & z = 0 & \dfrac{\partial z}{\partial u} = 0.
\end{array} \tag{33}$$

Using the developed method, the blending surface is generated as shown in Fig. 4c. The relative geometric parameters used in this blending operation were taken to be $u_0 = R_1 = 0.1$, $u_1 = 1$, $R_0 = 0.9$, $h_1 = h'_1 = 1.5$, $h_0 = \xi = \xi' = 0.5$, $a = a' = 1.6$ and $b = b' = 1.2$, and shape control parameters are the same as above.

Blending two intersecting cylinders

Unlike the closed-form resolution method which can only tackle a small number of blending problems, our method is more powerful and can deal with more complex blending problems. To demonstrate this, we use as an example the blending of two intersecting cylinders. Although it is achievable with some traditional methods, such as the rolling-ball method,

this problem has proven tricky for the PDE-based approach. The expensive FE and FD methods were the only means of doing this in the past. The boundary conditions for this blending problem are given as follows:

$$
\begin{aligned}
u=0 \quad & x = s\cos v && \frac{\partial x}{\partial u} = 0 \\
& y = s\sin v && \frac{\partial y}{\partial u} = 0 \\
& z = \sqrt{(r+k_1)^2 - s^2\cos^2 v} \\
& \frac{\partial z}{\partial u} = \frac{r+k_1}{\sqrt{(r+k_1)^2 - s^2\cos^2 v}} \\
u=1 \quad & x = (s+l_1)\cos v && \frac{\partial x}{\partial u} = t\cos v \\
& y = (s+l_1)\sin v && \frac{\partial y}{\partial u} = t\sin v \\
& z = \sqrt{r^2 - (s+l_1)^2\cos^2 v} \\
& \frac{\partial z}{\partial u} = -\frac{t\cos^2 v}{\sqrt{r^2 - (s+l_1)^2\cos^2 v}},
\end{aligned}
\tag{34}
$$

where s, r, k_1 and l_1 are the geometric parameters defining the two boundary curves, and t is the parameter controlling the first derivatives on the boundary curve.

According to boundary conditions (Eq. 34), PDE 2 has a closed-form solution for the x and y components, but it does not have a closed-form solution for the z component. Our method is very powerful in solving this class of problems. The general solution for this blending problem can be expressed as

$$
\begin{aligned}
x &= G_{x1}(u)\cos v \\
y &= G_{y1}(u)\sin v \\
z &= \sum_{i=0}^{3}\Bigg[c_{z1i}\sqrt{(r+k_1)^2 - s^2\cos^2 v} \\
&\quad + \frac{c_{z2i}}{\sqrt{(r+k_1)^2 - s^2\cos^2 v}} \\
&\quad + c_{z3i}\sqrt{r^2 - (s+l_1)^2\cos^2 v} \\
&\quad + \frac{c_{z4i}\cos^2 v}{\sqrt{r^2 - (s+l_1)^2\cos^2 v}}\Bigg]u^i \\
&\quad + u(u-1)^2\sum_{m=1}^{M}\sum_{n=1}^{N} A_{zmn}\sin mu\sin nv.
\end{aligned}
\tag{35}
$$

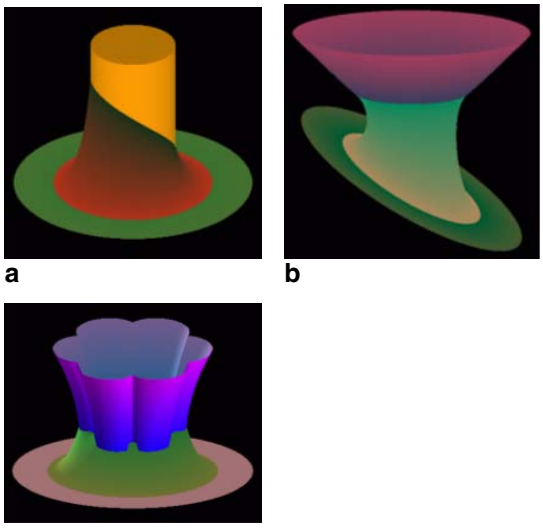


Fig. 4. Blending between two parametric surfaces

Using our method and the geometric parameters $s = 0.7, r = 1.2, k_1 = 0.5, l_1 = 0.3$ and $t = 0.5$ and shape control parameters $b_x = b_y = b_z = c_x = c_y = c_z = d_x = d_y = d_z = 1$, we determine all the unknown constants in Eq. 35. The blending surface generated from the general solution (Eq. 35) is shown in Fig. 5.

Implicit primary surfaces

If the primary surfaces are given in implicit forms, usually the first step is to convert them into a parametric form. Then the treatment outlined above can be applied to find the closed-form solution of Eq. 2 for the given surface-blending problem.

Blending a sphere with an ellipsoid

Let us examine the blending surface between a sphere and an ellipsoid. The implicit equations for their primary surfaces are given by

$$x^2 + y^2 + (z-H)^2 = R^2 \tag{36}$$

and

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1. \tag{37}$$

The following are the equivalent boundary conditions in a parametric form:

$$
\begin{aligned}
u=0 \quad & x=\frac{\sqrt{2}}{2}a\cos v & \frac{\partial x}{\partial u}&=\frac{\sqrt{2}}{2}a'\cos v\\
& y=\frac{\sqrt{2}}{2}b\sin v & \frac{\partial y}{\partial u}&=\frac{\sqrt{2}}{2}b'\sin v\\
& z=\frac{\sqrt{2}}{2}c & \frac{\partial z}{\partial u}&=\frac{\sqrt{2}}{2}c'\\
u=1 \quad & x=\frac{\sqrt{3}}{2}R\cos v & \frac{\partial x}{\partial u}&=\frac{1}{2}R'\cos v\\
& y=\frac{\sqrt{3}}{2}R\sin v & \frac{\partial y}{\partial u}&=\frac{1}{2}R'\sin v\\
& z=H-\frac{1}{2}R & \frac{\partial z}{\partial u}&=\frac{\sqrt{3}}{2}R',
\end{aligned}
\tag{38}
$$

where H defines the position of the trimline on the sphere along the height direction, R is the radius of the sphere, a, b and c are the geometric parameters defining the ellipsoid, and a', b', c' and R' are the parameters controlling the first derivatives on the trimlines.

The blending surface under boundary conditions (Eq. 38) for $a=1.4$, $b=0.9$, $c=c'=0.6$, $R=R'=0.7$, $H=2$, $a'=-1.4$, $b'=-0.9$ and shape control parameters $b_x=b_y=b_z=d_x=d_y=d_z=1$ and $c_x=c_y=c_z=2$ is depicted in Fig. 6.

Explicit primary surfaces

Similarly, some explicitly formulated primary surfaces can also be treated. A proper substitution of the variables in the explicit form should first be made. Then the explicit equations are transformed into a parametric form.

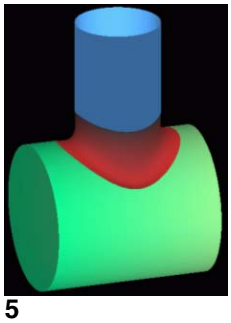
Blending an arbitrary ruled surface with a plane at a specified straight line

Here, the ruled surface is given by

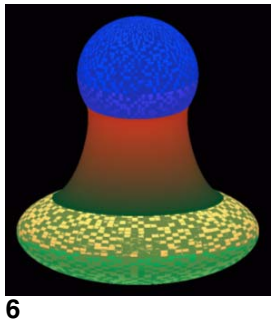
$$y=4x-16.8x^2+25.6x^3-12.8x^4. \tag{39}$$

First, we introduce a new variable v, which has the following relationship with x:

$$x=\sin v. \tag{40}$$



5



6

Fig. 5. Blending surface between two intersecting cylinders

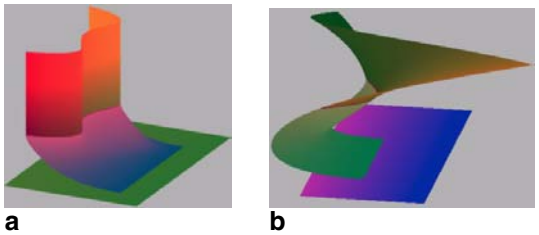
Fig. 6. Blending between two implicit surfaces

Now, the boundary conditions of the blending surface can be expressed as

$$
\begin{aligned}
u=0 \quad & x=p_0 v & \frac{\partial x}{\partial u}&=0\\
& y=s_0 & \frac{\partial y}{\partial u}&=-s_0'\\
& z=0 & \frac{\partial z}{\partial u}&=0\\
u=1 \quad & x=\sin v & \frac{\partial x}{\partial u}&=0\\
& y=-13.2+23.2\sin v & \frac{\partial y}{\partial u}&=0\\
& \quad +14.8\cos 2v-6.4\sin 3v & &\\
& \quad -1.6\cos 4v & &\\
& z=h_0 & \frac{\partial z}{\partial u}&=h_0',
\end{aligned}
\tag{41}
$$

where p_0 and s_0 define the position of the trimline on the plane, h_0 defines the position of the trimline on the ruled surface from the plane, and s_0' and h_0' are the parameters controlling the first derivatives on the trimlines.

Setting $p_0=\frac{2}{\pi}$, $s_0=0.7$, $s_0'=1.4$, $h_0=0.3$, $h_0'=0.45$ and shape control parameters $b_x=b_y=b_z=d_x=d_y=d_z=1$ and $c_x=c_y=c_z=2$, we obtain the blending surface in Fig. 7a.



a b

Fig. 7. Blending between two explicit surfaces

Blending a plane containing a specified straight line and an arbitrary surface

Here the surface is defined by the following explicit equation:

$$y = -0.2z^4 - 2xz^3 + 8x^2z^2 - 5x^3z + 0.3x^4. \qquad (42)$$

To obtain the boundary conditions in parametric form, we replace its variables using the following substitutions:

$$\begin{aligned} x &= \sin v \\ z &= u, \end{aligned} \qquad (43)$$

which leads to the following boundary conditions:

$$\begin{array}{lll} u = 0 & x = p_0 v & \dfrac{\partial x}{\partial u} = 0 \\ & y = s_0 & \dfrac{\partial y}{\partial u} = -s_0' \\ & z = 0 & \dfrac{\partial z}{\partial u} = 0 \\ u = 1 & x = \sin v & \dfrac{\partial x}{\partial u} = 0 \\ & \begin{aligned} y = 1.52658 & \\ -2.682 \sin v & \\ +3.18 \cos 2v & \\ +0.75 \sin 3v & \\ +0.0375 \cos 4v & \end{aligned} & \begin{aligned} \frac{\partial y}{\partial u} &= 4.6272 \\ & -5.91 \sin v \\ & -4.8 \cos 2v \\ & +1.25 \sin 3v \end{aligned} \\ & z = 0.6 & \dfrac{\partial z}{\partial u} = 1, \end{array} \qquad (44)$$

where p_0 and s_0 are as above and s_0' is the parameter controlling the first derivative on the trimline of the plane.

The blending surface created using $p_0 = \frac{2}{\pi}$, $s_0 = 0.1$, $s_0' = 2.55$ and the same shape control parameters as above is depicted in Fig. 7b.

6 Conclusions

A more general form of the fourth-order PDE with three vector-valued shape parameters was proposed for solving various surface-blending problems. All forms of existing fourth-order PDEs with one vector-valued parameter are in fact special cases of the proposed equation. Therefore, the proposed fourth-order PDE can generate a superset of all blending surfaces capable of being generated by the previously published equations.

We have developed an accurate and efficient method for determining the analytical solutions of the proposed PDE under a set of arbitrary boundary conditions. This has made the PDE-based approach much more usable with more realistic problems. A comparison with the closed-form resolution method and other existing analytical methods indicates that our method is able to generate blending surfaces almost as quickly and accurately as the closed-form resolution method and more accurately and efficiently than the Fourier series method, the pseudo-spectral method and other numerical methods. In addition, our method is able to cope with various complex surface-blending problems.

The effect that the shape parameters and the force function have on the shape of the blending surface has been investigated. We have shown that these shape parameters and the force function act as powerful shape control tools. With appropriate use of them, designers can create blending surfaces of a desired shape. Finally, a number of surface-blending problems were examined. These surface-blending problems included the blending of some different primary surfaces whose mathematical representations were expressed in parametric, implicit or explicit forms.

In this paper, we have only discussed the applications of the PDE-based geometric modeling method in surface blending. In fact, this method can also be applied to the generation of various free-form surfaces. In Fig. 8, we give three surface generation examples. The first demonstrates the capacity of local deformation which can maintain position continuity (Fig. 8a) or tangent continuity (Fig. 8b). The second example shows a model made up of three-sided,

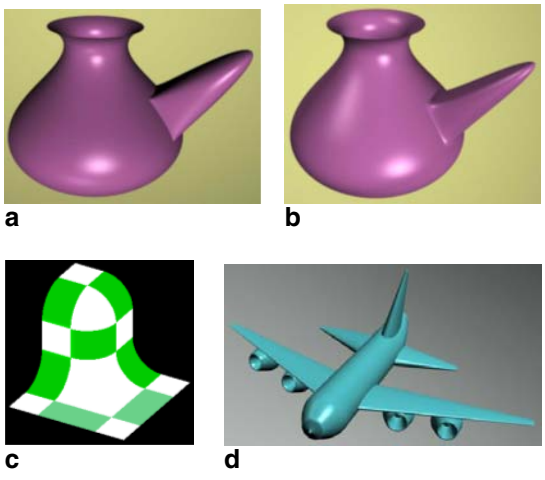


Fig. 8. Some examples of surface generation

four-sided and five-sided surface patches (Fig. 8c). Since any complex surface can be decomposed into a net of simple surface patches, this example demonstrates its ability to generate various complex surfaces. The third example is an aeroplane produced by dividing it into a number of patches and assembling them together (Fig. 8d). Our next step is to implement a user-friendly graphics interface, so that surface blending, generation and manipulation operations can be performed more easily in a graphical environment.

Appendix

When $\frac{d^2\bar{\boldsymbol{g}}_j(\xi v)}{dv^2} = \frac{d^4\bar{\boldsymbol{g}}_j(\xi v)}{dv^4} = 0$ and without considering the force function, Eq. 2 becomes

$$\frac{d^4\boldsymbol{G}_j(u)}{du^4} = 0 \quad (j = 1, 2, \cdots, J) \tag{A1}$$

and whose solution can be written as

$$\boldsymbol{G}_j(u) = \sum_{k=1}^{4} \boldsymbol{c}_{jk} u^{k-1} \quad (j = 1, 2, \cdots, J), \tag{A2}$$

where $c_{jk}(j = 1, 2, \cdots, J; k = 1, 2, 3, 4)$ are the unknown constants.

When $\frac{d^2\bar{\boldsymbol{g}}_j(\xi v)}{dv^2} = -\xi^2\bar{\boldsymbol{g}}_j(\xi v)$ and $\frac{d^4\bar{\boldsymbol{g}}_j(\xi v)}{dv^4} = \xi^4\bar{\boldsymbol{g}}_j(\xi v)$, Eq. 2 without the force function becomes

$$\left(\boldsymbol{b}\frac{d^4}{du^4} - \boldsymbol{c}\xi^2\frac{d^2}{du^2} + \boldsymbol{d}\xi^4\right)\boldsymbol{G}_j(u) = 0 \quad (j = 1, 2, \cdots, J) \tag{A3}$$

whose characteristic equation is

$$\boldsymbol{b}r_j^4 - \boldsymbol{c}\xi^2 r_j^2 + \boldsymbol{d}\xi^4 = 0 \quad (j = 1, 2, \cdots, J). \tag{A4}$$

Solving the above equation, we obtain its roots as follows:

$$r_{j1,j2,j3,j4} = \pm\xi\sqrt{\frac{\boldsymbol{c}}{2\boldsymbol{b}}\left(1 \pm \sqrt{1 - \frac{4\boldsymbol{bd}}{\boldsymbol{c}^2}}\right)}. \tag{A5}$$

When $4\boldsymbol{bd} < \boldsymbol{c}^2$, $1 - \frac{4\boldsymbol{bd}}{\boldsymbol{c}^2} > 0$ and the four roots of the characteristic equation are real and can be represented with the above equation. When $4\boldsymbol{bd} = \boldsymbol{c}^2$, $1 - \frac{4\boldsymbol{bd}}{\boldsymbol{c}^2} = 0$, only two distinctive real roots exist, which are

$$r_{j1,j2} = \pm\xi\sqrt{\frac{\boldsymbol{c}}{2\boldsymbol{b}}}. \tag{A6}$$

When $4\boldsymbol{bd} > \boldsymbol{c}^2$, $\sqrt{1 - \frac{4\boldsymbol{bd}}{\boldsymbol{c}^2}} = \sqrt{-\left(\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2} - 1\right)} = \sqrt{i^2\left(\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2} - 1\right)} = i\sqrt{\left(\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2} - 1\right)}$. Since i is an imaginary number, the square root of $\frac{\boldsymbol{c}}{2\boldsymbol{b}}\left(1 \pm i\sqrt{\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2} - 1}\right)$ does not exist. Therefore, the characteristic equation for this case has no closed-form solution. In practice, it does not cause a real problem, as one can always avoid such a combination of the shape parameters. However, if we want to tackle this case, we can use the pseudo-Navier approximate solution. Therefore, for the first two cases, i.e., $4\boldsymbol{bd} < \boldsymbol{c}^2$ and $4\boldsymbol{bd} = \boldsymbol{c}^2$, the unknown function $\boldsymbol{G}_j(u)$, which does not include the imaginary number, can be written as

$$\begin{aligned}
&\boldsymbol{G}_j(u) = \boldsymbol{c}_{j1}e^{r_{j1}u} + \boldsymbol{c}_{j2}e^{r_{j2}u} + \boldsymbol{c}_{j3}e^{r_{j3}u} + \boldsymbol{c}_{j4}e^{r_{j4}u} \quad \text{for } 4\boldsymbol{bd} < \boldsymbol{c}^2,\\
&\boldsymbol{G}_j(u) = (\boldsymbol{c}_{j1} + \boldsymbol{c}_{j2}u)e^{r_{j1}u} + (\boldsymbol{c}_{j3} + \boldsymbol{c}_{j4}u)e^{r_{j2}u} \quad \text{for } 4\boldsymbol{bd} = \boldsymbol{c}^2,
\end{aligned} \tag{A7}$$

where the unknown constants $c_{jk}(j = 1, 2, \cdots, J; k = 1, 2, 3, 4)$ can be determined as above.

When $\frac{d^2\bar{\boldsymbol{g}}_j(\xi v)}{dv^2} = \xi^2\bar{\boldsymbol{g}}_j(\xi v)$ and $\frac{d^4\bar{\boldsymbol{g}}_j(\xi v)}{dv^4} = \xi^4\bar{\boldsymbol{g}}_j(\xi v)$,

Eq. 2 without the force function becomes

$$\left(\boldsymbol{b}\frac{d^4}{du^4}+\boldsymbol{c}\xi^2\frac{d^2}{du^2}+\boldsymbol{d}\xi^4\right)\boldsymbol{G}_j(u)=0 \quad (j=1,2,\cdots,J) \tag{A8}$$

whose characteristic equation is

$$\boldsymbol{b}r_j^4+\boldsymbol{c}\xi^2 r_j^2+\boldsymbol{d}\xi^4=0 \quad (j=1,2,\cdots,J). \tag{A9}$$

Solving the above equation, we obtain its roots as follows:

$$r_{j1,j2}=\pm i\bar{r}_{j1}, \quad r_{j3,j4}=\pm i\bar{r}_{j2}, \tag{A10}$$

where $\bar{r}_{j1}=\xi\sqrt{\frac{\boldsymbol{c}}{2\boldsymbol{b}}\left(1+\sqrt{1-\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2}}\right)}$ and $\bar{r}_{j2}=\xi\sqrt{\frac{\boldsymbol{c}}{2\boldsymbol{b}}\left(1-\sqrt{1-\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2}}\right)}$.

As above, when $4\boldsymbol{bd}<\boldsymbol{c}^2$, $1-\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2}>0$, the four roots of the characteristic equation are imaginary and can be represented with the above equation. When $4\boldsymbol{bd}=\boldsymbol{c}^2$, $1-\frac{4\boldsymbol{bd}}{\boldsymbol{c}^2}=0$, only two distinctive roots exist. These two imaginary roots are

$$r_{j1,j2}=\pm i\hat{r}_{j1}, \tag{A11}$$

where $\hat{r}_{j1}=\xi\sqrt{\frac{\boldsymbol{c}}{2\boldsymbol{b}}}$.

When $4\boldsymbol{bd}>\boldsymbol{c}^2$, the characteristic equation for this case has no closed-form solution and the pseudo-Navier approximate solution can be used. Therefore, the unknown function $\boldsymbol{G}_j(u)$ only has the following two forms:

$$\begin{aligned}\boldsymbol{G}_j(u)&=\boldsymbol{c}_{j1}\cos\bar{r}_{j1}u+\boldsymbol{c}_{j2}\sin\bar{r}_{j1}u+\boldsymbol{c}_{j3}\cos\bar{r}_{j2}u\\&\quad+\boldsymbol{c}_{j4}\sin\bar{r}_{j2}u \text{for } 4\boldsymbol{bd}<\boldsymbol{c}^2\\ \boldsymbol{G}_j(u)&=(\boldsymbol{c}_{j1}+\boldsymbol{c}_{j2}u)\cos\hat{r}_{j1}u\\&\quad+(\boldsymbol{c}_{j3}+\boldsymbol{c}_{j4}u)\sin\hat{r}_{j1}u \text{for } 4\boldsymbol{bd}=\boldsymbol{c}^2.\end{aligned} \tag{A12}$$

Removing the overbar " ¯ " and hat " ˆ " in the above equation, Eq. (A12) becomes equation (8).

For the other cases except for those given above, Eq. 2 cannot be transformed from a partial different equation to an ordinary differential equation. Therefore, it has no closed-form solution. These cases and the solution corresponding to the right-hand side term of PDE 2, i.e., the force function, are classified into $\sum_{k=1}^{K}\boldsymbol{b}_{ki}\hat{\boldsymbol{g}}_k(v)(i=1,2,3,4)$.

References

1. Bloor MIG, Wilson MJ (1989) Generating blend surfaces using partial differential equations. Comput Aided Des 21(3):165–171
2. Bloor MIG, Wilson MJ (1990a) Using partial differential equations to generate free-form surfaces. Comput Aided Des 22(4):202–212
3. Bloor MIG, Wilson MJ (1990b) Representing PDE surfaces in terms of B-splines. Comput Aided Des 22(6):324–331
4. Bloor MIG, Wilson MJ (1996) Spectral approximations to PDE surfaces. Comput Aided Des 28(2):145–152
5. Bloor MIG, Wilson MJ (2000) Method for efficient shape parametrization of fluid membranes and vesicles. Phys Rev E 61(4):4218–4229
6. Bloor MIG, Wilson MJ, Hagen H (1995) The smoothing properties of variational schemes for surface design. Comput Aided Geom Des 12(4):381–394
7. Brown JM, Bloor MIG, Susan M, Wilson MJ (1990) Generation and modification of non-uniform B-spline surface approximations to PDE surfaces using the finite element method. In: Ravani B (ed) Advances in design automation, vol 1. Computer Aided and Computational Design, ASME Press, pp 265–272
8. Cheng SY, Bloor MIG, Saia A, Wilson MJ (1990) Blending between quadric surfaces using partial differential equations. In: Ravani B (ed) Advances in design automation, vol 1. Computer and Computational Design, ASME Press, pp 257–263
9. Davis J, Marschner SR, Garr M, Levoy M (2002) Filling holes in complex surfaces using volumetric diffusion. In: Proceedings of the 1st international symposium on 3D data processing, visualization, transmission, Padova, Italy 19–21 June 2002, pp 428–438
10. Du H, Qin H (2000) Direct manipulation and interactive sculpting of PDE surfaces. In: Proceedings of EUROGRAPHICS 2000, Computer Graphics Forum, Interlaken, Switzerland, 20–25 August 2000, 19(3):61–270
11. Li ZC (1998) Boundary penalty finite element methods for blending surfaces: I. Basic theory. J Comput Math 16:457–480
12. Li ZC (1999) Boundary penalty finite element methods for blending surfaces: II. Biharmonic equations. J Comput Appl Math 110:55–176
13. Li ZC, Chang CS (1999) Boundary penalty finite element methods for blending surfaces: III. Superconvergence and stability and examples. J Comput Appl Math 110:241–270
14. Mimis AP, Bloor MIG, Wilson MJ (2001) Shape parameterization and optimization of a two-stroke engine. J Propulsion Power 17(3):492–498
15. Rossignac JR, Requicha AAG (1984) Constant-radius blending in solid modeling. Comput Mech Eng 3:65–73
16. Ugail H, Bloor MIG, Wilson MJ (1999a) Techniques for interactive design using the PDE method. ACM Trans Graph 18(2):195–212
17. Ugail H, Bloor MIG, Wilson MJ (1999b) Manipulation of PDE surfaces using an interactively defined parameterisation. Comput Graph 23:525–534
18. Vida J, Martin RR, Varady T (1994) A survey of blending methods that use parametric surfaces. Comput Aided Des 26(5):341–365

19. Whitaker RT, Breen DE (1998) Level-set models for the deformation of solid objects. In: Proceedings of the conference on implicit surfaces, Seatle, 15–16 June 1998, pp 19–35
20. You LH, Zhang JJ (2001) Finite difference surface representation considering effect of boundary curvature. In: Proceedings of the 5th international conference on information visualisation, London, 25–27 July 2001. IEEE Press, New York, pp 404–409
21. You LH, Zhang JJ, Comninos P (1999) Cloth deformation modelling using a plate bending model. In: Proceedings of the 7th international conference in Central Europe on computer graphics, visualisation and interactive digital media, Plzen, Czech Republic, 8–12 February 1999, pp 485–491
22. You LH, Zhang JJ, Comninos P (2000) A volumetric deformable muscle model for computer animation using weighted residual method. Comput Meth Appl Mech Eng 190:853–863
23. You LH, Hu JH, Shi YH, Zhang JJ (2003) Single-patch surfaces for tool shape design and finite element analysis of hot metal forming. J Mater Process Technol (in press)
24. Zhang JJ, You LH (2001) Surface representation using second, fourth and mixed order partial differential equations. In: Proceedings of the international conference on shape modelling and applications, Genoa, Italy, 7–11 May 2001. IEEE Press, New York, pp 250–256
25. Zhang JJ, You LH (2002) PDE based surface representation – vase design. Comput Graph 26(1):89–98
26. Zhao H-K, Osher S, Fedkiw R (2001) Fast surface reconstruction using the level set method. In: Proceedings of the IEEE workshop on variational and level set methods in computer vision (VLSM), Vancouver, Canada, 13 July 2001, pp 194–201

DR. L. H. YOU is currently a research fellow at the National Centre for Computer Animation, Bournemoth Media School, Bournemouth University, UK. His research interests are in computer graphics, computer animation and geometric modelling.