Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers and Structures 81 (2003) 2601–2609
Computers & Structures
www.elsevier.com/locate/compstruc

Efficient shape parametrisation for automatic design optimisation using a partial differential equation formulation

H. Ugail [a,*], M.J. Wilson [b]

[a] Department of Electronic Imaging and Media Communications, School of Informatics, University of Bradford, Bradford BD7 1DP, UK
[b] Department of Applied Mathematics, University of Leeds, Leeds LS2 9JT, UK

Received 7 November 2002; accepted 8 July 2003

Abstract

This paper presents a methodology for efficient shape parametrisation for automatic design optimisation using a partial differential equation (PDE) formulation. It is shown how the choice of an elliptic PDE enables one to define and parametrise geometries corresponding to complex shapes. By using the PDE formulation it is shown how the shape definition and parametrisation can be based on a boundary value approach by which complex shapes can be created and parametrised based on the shape information at the boundaries or the character lines defining the shape. Furthermore, this approach to shape definition allows complex shapes to be parametrised intuitively using a very small set of design parameters.

Thus, it is shown that the PDE based approach to shape parametrisation when combined with a standard method for numerical optimisation is capable of setting up automatic design optimisation problems allowing practical design optimisation to be more feasible.

Keywords: PDEs; Parametric design; Shape parametrisation; Automatic optimisation

1. Introduction

One of the requirements of any practical design optimisation system is the ability to parametrise the shape of the objects. In parametric design the basic approach is to develop a generic description of an object or a class of objects in which the shape is controlled by the values of a set of design variables or parameters. A new design, created for a particular application, is obtained from this generic template by selecting particular values for the design parameters so that the item has particular properties suited to that application.

It has been noted by several authors in the shape optimisation literature that the most important aspect of shape optimisation is the choice of the design variables to be used and how the shape is parametrised in terms of these design variables [9,13,15,27,28]. Choosing too many variables will considerably complicate the design problem with severe implications on the computational time required, and having too few variables may result in only a limited range of design alternatives being obtained [9,13]. It is therefore a basic requirement that a wide range of shapes, that can be defined by a relatively small number of parameters, are accessible to the method of optimisation used. Thus, from the point of view of automatic design optimisation, that is, the automatic process of systematically finding the most preferable specific shape from a given set of designs, it is desirable that the shape in question be completely defined in terms of a small parameter set. Furthermore, if the chosen method for shape parametrisation is to be successfully used as part of a practical design tool, it is also necessary that each of the parameters have

* Corresponding author.
E-mail addresses: h.ugail@bradford.ac.uk (H. Ugail), mike@maths.leeds.ac.uk (M.J. Wilson).

0045-7949/$ - see front matter
doi:10.1016/S0045-7949(03)00321-3

an intuitively predictable effect on the shape of the surface.

Various methods of defining and parametrising shape for use in design optimisation can be found in the literature. For example, an early method for shape parametrisation is the nodal coordinate approach that uses the coordinates of the nodes of the discrete finite-element model as design variables [9]. In practical design optimisation situations this results in a large number of design variables leading to great inefficiency in design optimisation. Several methods to overcome these initial drawbacks for shape parametrisation have been put forward. These include the mesh parametrisation approach [7], the use of solid modelling [14] and the natural design variable method [21]. Although these methods are relatively easy to implement for realistic design optimisation problems because of the very large number of design variables associated with them, they often lead to high cost and thus very often make optimisation to be prohibitive.

Another popular approach for defining and parametrising shape for use in design optimisation is the so-called spline approach where the shape is represented by means of a series of polynomial functions. Two common methods of spline based surface representation are Bezier and B-splines [20]. Typically, the surface to be represented are broken into a mesh of mainly rectangular curvilinear regions. A surface patch is then defined over each region whose shape is determined by a set of control points. The shape parameters in this formulation are the coordinates of each control point [4]. It can thus be seen that in order to represent the shape of a practical object it would require a number of surface patches often involving large number of shape parameters. Furthermore, for design optimisation involving complex geometry the spline based approach makes it difficult to maintain smooth transitions between adjacent surface patches.

The aim of this paper is to discuss a technique for efficient shape parametrisation for automatic design optimisation based on solutions to suitably chosen partial differential equations known as the PDE method [1]. The method adopts a boundary-value approach in which the shape of the object in question is decomposed into a series of surface patches bounded by 'character-lines', where the number of patches are being kept as low as possible. Using boundary data appropriately defined along the character-lines the PDE method produces smooth surfaces between them. Furthermore, the shape of the surface is efficiently described in terms of a small number of shape parameters, thus enabling efficient automatic optimisation to be carried out.

The outline of the paper is as follows. The paper discusses how the PDE method can be used to create a parametric surface. As mentioned above the shape parametrisation is vital for automatic optimisation, and therefore a methodology that allows PDE surfaces to be efficiently parametrised is discussed. Examples of automatic design optimisation is also discussed later in the paper. In particular, two examples from two different settings are discussed to demonstrate the capability of the methodology presented. The first example considers the optimal design of an object of minimum weight enclosing a fixed volume by imposing constraints on the load bearing characteristics of the object. The second example discusses the prediction of stable shapes of biological vesicles occurring within biological organisms subject to a given set of physical conditions in which such vesicles are to be found.

2. PDE method and geometry definition

The PDE method was initially introduced into the area of computer-aided geometric design as a method for blend generation [1]. The problem of blend generation is essentially that of being able to generate a smooth surface that acts as a bridging transition between neighbouring primary surfaces. Thus, the PDE method produces a parametric surface $\underline{X}(u,v)$, defined as a function of two parameters u and v on a finite domain $\Omega \subset R^2$, by specifying boundary data around the edge region of $\partial\Omega$. Typically the boundary data are specified in the form of $\underline{X}(u,v)$ and a number of its derivatives on $\partial\Omega$. Moreover, this approach regards the coordinates of point u and v as a mapping from that point in Ω to a point in the physical space. To satisfy these requirements the surface $\underline{X}(u,v)$ is regarded as a solution of a PDE of the form

$$D^m_{u,v}\underline{X}(u,v) = \underline{F}(u,v), \tag{1}$$

where $D^m_{u,v}\underline{X}(u,v)$ is a partial differential operator of order m in the independent variables u and v, while $\underline{F}(u,v)$ is vector valued function of u and v. Since boundary-value problems are considered here, it is natural to choose $D^m_{u,v}\underline{X}(u,v)$ to be elliptic.

Various elliptic PDEs could be used, although for the work described here the PDE chosen is based on the biharmonic equation $\nabla^4 = 0$ namely,

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

Here the boundary conditions on the function $\underline{X}(u,v)$ and its normal derivatives $\frac{\partial \underline{X}}{\partial n}$ are imposed at the edges of the surface patch.

With this formulations one can see that the elliptic partial differential operator in Eq. (2) represents a smoothing process in which the value of the function at any point on the surface is, in some sense, a weighted average of the surrounding values. In this way a surface is obtained as a smooth transition between the chosen

set of boundary conditions. The parameter a is a special design parameter which controls the relative smoothing of the surface in the u and v directions [2].

General discussions of the PDE method related to computer-aided geometric design has been discussed before and more details can be found in [1,2,26]. It has also been shown how surfaces satisfying a given set of functional requirements can be created by a suitable choice of the boundary conditions and appropriate values for the various design parameters associated with the method [8,23,24].

2.1. Solution of the PDE

There exist many methods to determine the solution of Eq. (2). In some cases, where the boundary conditions can be expressed as relatively simple analytic functions of u and v, it is possible to find a closed form solution. However, for a general set of boundary conditions a numerical method often need to be employed.

Restricting to periodic boundary conditions here a closed form analytic solution of Eq. (2) based on spectral approximation is discussed. Choosing the parametric region to be $0 \leqslant u \leqslant 1$ and $0 \leqslant v \leqslant 2\pi$ the periodic boundary conditions can be expressed as:

$$\underline{X}(0,v) = \underline{P}_0(v), \tag{3}$$

$$\underline{X}(1,v) = \underline{P}_1(v), \tag{4}$$

$$\underline{X}_u(0,v) = \underline{d}_0(v), \tag{5}$$

$$\underline{X}_u(1,v) = \underline{d}_1(v). \tag{6}$$

Note that the boundary conditions $\underline{P}_0(v)$ and $\underline{P}_1(v)$ define the edges of the surface patch at $u = 0$ and $u = 1$ respectively. Using the method of separation of variables, the analytic solution of Eq. (2) can be written as

$$\underline{X}(u,v) = \underline{A}_0(u) + \sum_{n=1}^{\infty}[\underline{A}_n(u)\cos(nv) + \underline{B}_n(u)\sin(nv)], \tag{7}$$

where

$$\underline{A}_0 = \underline{a}_{00} + \underline{a}_{01}u + \underline{a}_{02}u^2 + \underline{a}_{03}u^3, \tag{8}$$

$$\underline{A}_n = \underline{a}_{n1}e^{anu} + \underline{a}_{n2}e^{anu} + \underline{a}_{n3}e^{-anu} + \underline{a}_{n4}e^{-anu}, \tag{9}$$

$$\underline{B}_n = \underline{b}_{n1}e^{anu} + \underline{b}_{n2}e^{anu} + \underline{b}_{n3}e^{-anu} + \underline{b}_{n4}e^{-anu}, \tag{10}$$

where $\underline{a}_{n1}$, $\underline{a}_{n2}$, $\underline{a}_{n3}$, $\underline{a}_{n4}$, $\underline{b}_{n1}$, $\underline{b}_{n2}$, $\underline{b}_{n3}$ and $\underline{b}_{n4}$ are vector-valued constants, whose values are determined by the imposed boundary conditions at $u = 0$ and 1.

For a given set of boundary conditions, in order to define the various constants in the solution, it is necessary to Fourier analyse the boundary conditions and identify the various Fourier coefficients. When the boundary conditions can be expressed exactly in terms of a finite Fourier series, the solution given in Eq. (7) will also be finite. However, this is often not possible, in which case the solution will be the infinite series given Eq. (7).

An efficient technique for finding an approximation to $\underline{X}(u,v)$ is described in [3] based on the sum of the first few Fourier modes and a 'remainder term', i.e.,

$$\underline{X}(u,v) = \underline{A}_0(u) + \sum_{n=1}^{N}[\underline{A}_n(u)\cos(nv) + \underline{B}_n(u)\sin(nv)] + \underline{R}(u,v), \tag{11}$$

where $N \leqslant 6$ and $\underline{R}(u,v)$ is a remainder function defined as,

$$\underline{R}(u,v) = \underline{r}_1(v)e^{wu} + \underline{r}_2(v)e^{wu} + \underline{r}_3(v)e^{-wu} + \underline{r}_4(v)e^{-wu}, \tag{12}$$

where $\underline{r}_1$, $\underline{r}_2$, $\underline{r}_3$, $\underline{r}_4$ and w are obtained by considering the difference between the original boundary conditions and the boundary conditions satisfied by the function

$$\underline{F}(u,v) = \underline{A}_0(u) + \sum_{n=1}^{N}[\underline{A}_n(u)\cos(nv) + \underline{B}_n(u)\sin(nv)]. \tag{13}$$

This solution technique is considerably faster than looking for a very accurate solution to Eq. (2) using numerical methods such as finite element or finite difference. An important point to note here is that although the solution is approximate, this new solution technique guarantees that the chosen boundary conditions are exactly satisfied since the remainder function $\underline{R}(u,v)$ is calculated by means of the difference between the original boundary conditions and the boundary conditions satisfied by the function $\underline{F}(u,v)$.

It is noteworthy that the biharmonic form of equations similar to Eq. (2) are well known for their applicability in plate and shell analysis. Moreover, similar analytic solutions techniques as described above such as Navier's or Levy's methods are also used to solve the related biharmonic equations analytically. More details on this, for example, can be found in [25].

2.2. Interactive definition of PDE geometry

For the purpose of creating an intuitive design interface which can be used with ease by a designer, the boundary conditions can be defined by means of curves in 3-space. For example, Fig. 1 shows a typical set of boundary curves and the corresponding PDE surface with the value of a in Eq. (2) taken to be 1.1000. Note that the curves marked $p1$ and $p2$ correspond to the boundary conditions on the function $\underline{X}(u,v)$, where

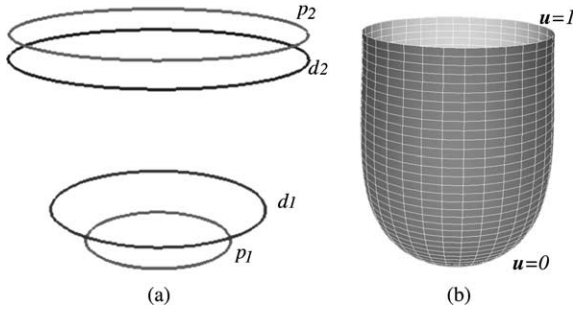

Fig. 1. A typical PDE surface: (a) the boundary curves and (b) the corresponding PDE surface patch.

$p1(v) = \underline{X}(0, v)$ and $p2(v) = \underline{X}(1, v)$. A vector field corresponding to the difference between the points on the curves marked $p1$ and $p2$ and those marked $d1$ and $d2$ respectively, corresponds to the conditions on the function $\frac{\partial \underline{X}}{\partial n}$ such that:

$$\frac{\partial \underline{X}}{\partial n} = [\underline{p}(v) - \underline{d}(v)]s, \tag{14}$$

where s is a scalar. The conditions defined by $p1$, $p2$ and $d1$, $d2$ are known as the 'positional boundary conditions' and 'derivative boundary conditions' respectively [23,24]. Note that the surface patch will not necessarily pass through the curves which define the derivative boundary conditions.

2.3. Parametrisation of the PDE geometry model

The shape parametrisation used here is based on the parametrised boundary curves that define the shape of the surface. Essentially, this parametrisation is defined in such a way that linear transformations, such as translation, rotation and dilation, of the boundary curves can be carried out interactively by a designer. It can be seen that such a parametrisation allows a designer sitting at a workstation to create and modify the geometry in an intuitive manner.

For the work presented here, the parametrisation on the boundary curves is denoted using the notation c_{kp_i} $(k = 1, 2, \ldots)$, $(i = x, y, z)$. Here c indicates the type of curve, with the letter p denoting the position curves and the letter d denoting the derivative curves. The index k ranges from 1 to 2 corresponding to the $u = 0$ and $u = 1$ boundary edges (respectively) of the surface. The letter p denotes the type of parameter: T for a translation, R for a rotation and D for a dilation. Finally the letter i denotes the coordinate directions relevant to a particular type of parameter. Adjustments to the values of these parameters along with the value of a in Eq. (2) can be used to create and manipulate complex geometries in an intuitive fashion.

Table 1
Values for the design parameters for the boundary $d2$ of the surfaces shown in Figs. 1(b) and 2(b)

Parameter	Fig. 1	Fig. 2
d_{2T_x}	0.000	0.000
d_{2T_y}	0.850	0.750
d_{2T_z}	0.000	0.000
d_{2D_x}	0.700	0.500
d_{2D_y}	0.700	0.500
d_{2D_z}	0.000	0.000
d_{2R_x}	0.000	0.000
d_{2R_y}	0.000	0.000
d_{2R_z}	0.000	0.000
a	1.100	1.100

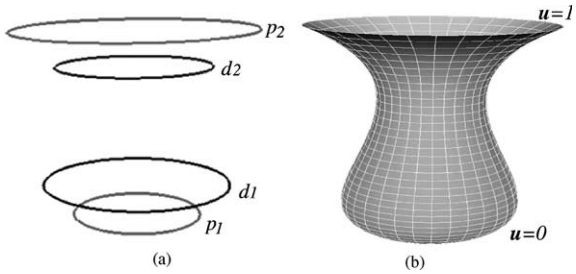

Fig. 2. The effect on the shape of the surface by changing the design parameters corresponding to the boundary $d2$ for the surface shown in (a) the boundary curves and (b) the corresponding PDE surface patch.

The effect of the chosen parameters on the surface shape is easy to appreciate. For example, Table 1 shows the values of the chosen parameters for the boundary curve $d2$ for the surface shown in Fig. 1. In order to show the effect of the design parameters one can now choose a different set of values for the parameters for the boundary $d2$. Table 1 shows new values chosen for the parameters and the resulting surface is shown in Fig. 2. As can be seen, these new parameter values have produced a dilation followed by a translation downwards of the boundary curve $d2$.

Often to create complex shapes, more than one surface patch needs to be joined together along common boundaries thereby forming a composite surface [22,23]. For example, Fig. 3 shows the shape of a biological vesicle created using two surface patches with a common boundary. The extension of the parametric model discussed above to cater for such composite bodies comprising more than one PDE patch is a straightforward exercise.

3. Automatic design optimisation

In this section, the PDE based shape parametrisation system discussed above is used to describe a methodo-

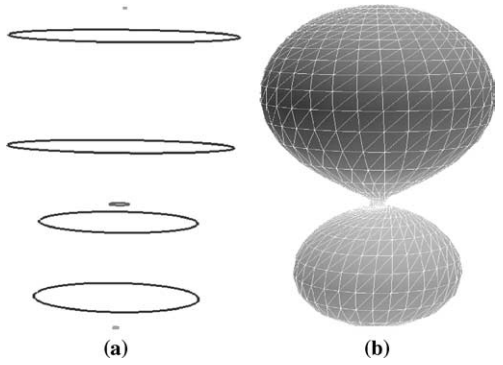

Fig. 3. Shape of a biological vesicle created using two PDE surface patches with a common boundary: (a) the boundary curves and (b) the corresponding PDE surface patches.

logy for the practical automatic design optimisation. In particular, two examples of automatic design optimisation are discussed, i.e. the design optimisation of a plastic container by way of minimising its mass subject to a given level of strength and a fixed volume, and the prediction of stable shapes of a class of biological vesicles so as to predict the structural arrangement of the molecules forming a biological membrane. To do this the initial shapes are created using PDE surfaces which are then parametrised using the surface parametrisation methods discussed in Section 2.3. The parameters are varied automatically by means of a suitable optimisation routine, where the value of the design merit function is computed for each parameter set.

A typical design optimisation problem consists of maximising or minimising an objective function without violating a set of constraints. There exist a wide variety of methods for numerical optimisation [11]. The choice of a particular method is problem specific and involves considerations such as the computational cost of evaluating the function to be optimised, and also the behaviour of the function within the design space.

For the purpose of demonstrating the techniques discussed in the paper, here the optimisation is performed by solving a constrained optimisation problem formulated by means of the objective function f, the design parameters associated with the geometry of the shape and the constraints imposed. e.g. strength and volume. This is carried out using an augmented Lagrange multiplier method [11] along with the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method [17]. Here BFGS is chosen for the sake of illustration, other methods such as sequential quadratic programming (SQP) can equally be used.

Since the BFGS is a gradient based optimisation scheme, during the optimisation one would need to perform a design sensitivity analysis so that the necessary gradients are fed to the minimisation algorithm. Here it has been chosen to evaluate the design sensitivity numerically by means of a finite difference scheme. In particular, the gradients of the objective function f are approximated by forward difference.

3.1. Numerical example illustrating the automatic design optimisation of a thin-walled structure

Consider the shape shown in Fig. 4 which describes the shape of a container suitable for packaging food product such as yoghurt. The boundary conditions for creating this shape is similar to those shown in Fig. 1 except for the derivative condition $d1$ which is as described below. This is the initial shape used here to demonstrate the above mentioned techniques for automatic shape optimisation. Note that, although this particularly simple shape is only chosen for the purposes of demonstration, a great many yoghurt containers are made with this simple shape. However, the techniques are equally applicable to more complicated designs.

An interesting feature which is introduced into the initial design of the container is the ridges found at the base of the container. These ridges are created by means of the corresponding derivative curve, $d1$, which in this case is defined by means of a cubic B-spline [20] and takes the form:

$$\underline{d}_1(v) = \sum_i \underline{c}_i B_i(v), \tag{15}$$

where B_i is a cubic B-spline, and $\underline{c}_i$ are the control points.

The control points, $\underline{c}_i$, of the spline are chosen so that the curve $d1$ has the shape shown in Fig. 5. The number of control points chosen determines the number of ridges on the container. In order to define the amplitude of the ridges a translation of the control points, normal to the curve, within a confined xy planar region is carried out. Thus, the amount of translation of the control

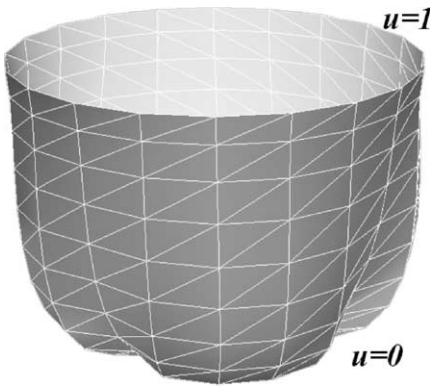

Fig. 4. An example shape of a container used for automatic optimisation.

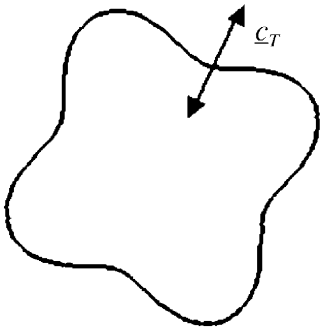

Fig. 5. Cubic B-spline curve corresponding derivative condition enabling to create the ridges at the base of the container.

points, normal to the curve determines the prominence of the ridges. The translation of the control points, introduces an extra shape parameter which is referred as $\underline{c}_T$. Fig. 5 shows the B-spline curve illustrating the parameter $\underline{c}_T$.

For the optimisation, the shape shown in Fig. 4 is taken to be the initial geometry of the container. The parametrisation on the container is chosen so that automatic variations of the parameters produce possible realistic shapes. As mentioned earlier the goal of this optimisation is to design a container possessing the minimum amount of material subject to a required strength. In particular, here the problem of stacking such containers on top of each other is considered. Assuming the container to be considered is at the bottom of the stack, such a container will experience a stress (due to the weight of the rest of the containers in the stack) and hence becomes slightly deformed. It is the excessive shear stress which can cause most damage to the material under consideration. Thus, a measure for the required strength of the container can be computed by calculating the maximum shear stress within the loaded container by means of performing a thin shell finite element analysis as described below.

In order to calculate the stresses occurring within the walls of the container when it is loaded, the elastic material model for a thin shell structures is assumed [16]. Thus, the relation between the stresses in the container to the displacement of the material can be described as

$$\underline{\sigma} = \underline{D}\underline{B}\delta, \tag{16}$$

where $\underline{\sigma}$ is a vector of stress; $\underline{\delta}$, the vector of displacements; $\underline{D}$, a matrix dependent on the properties of the material and $\underline{B}$, a differential operator.

The geometry of the container is divided into subregions to form the shell elements which in this case are three noded triangular elements. Thus, given the PDE geometry corresponding to the shape of a container, this geometry is then discretised, by means of the two-dimensional (u, v) parameter space, to obtain a valid finite element mesh. In particular, the task of node numbering is relatively easy to deal with using the discretised (u, v) parameter space. To create the appropriate shell elements a thickness is generated by means of calculating normals to the surface points defining the finite element mesh. Then using Eq. (16) which relates the stresses in the structure to the displacements, a set of differential equations are formed determining the displacement of the structure for some given boundary condition displacements. These equations are then solved using the finite element method [12].

Assuming the container is composed of polystyrene with the material properties as in Table 2 the strength of the container is characterised by means calculating the maximum shear stress occurring within the container. A vertical force of 15 $\mathrm{N\,m^{-1}}$ (equivalent to the weight of about 30 yoghurt containers) is applied around the rim of the container at $u = 1$. It is also assumed that the base $u = 0$ of the container is fixed and the thickness of the polystyrene sheet used to produce the container is 0.8 mm.

With this formulation the design objective here is set to be the minimisation of the mass of the container subject to a given maximum shear stress. Hence the process of optimisation requires the calculation of the maximum shear stress that occurs in the solution for every design to be analysed.

Using the principal components of shear stresses occurring within the structure, the maximum shear stress $\sigma^p_{\max}$ occurring on any plane through a point p is calculated. Thus, the measure for the strength of the container is set to be the maximum shear stress occurring in the whole structure, i.e.

$$f = \underbrace{\max}_{(\text{all points})} \{\sigma^p_{\max}\}. \tag{17}$$

The objective here is to find the *internal* shape of the container with fixed edges. Therefore, the changes in the parameters introduced on the derivative boundary curves corresponding to the shape of the container are only varied, i.e. both the positional boundary curves are kept fixed. In particular, to obtain a favourable range of shapes, the translation in y direction and dilations in the xy plane of these two curves within defined limits are considered.

Table 2
Material properties used for the finite element analysis

Material property	Value
Young's modulus (E)	2.4×10^4 $\mathrm{kN\,m^{-2}}$
Poisson's ratio	0.33
Yield stress	2.5×10^3 $\mathrm{kN\,m^{-2}}$

Table 3
Parameter values for the shape of the container used in the optimisation

Parameter	Minimum	Maximum	Initial	Optimal
d_{1T_y}	−0.400	−0.001	−0.400	−0.134
d_{1D_x}	0.100	0.800	0.450	0.300
d_{1D_y}	0.100	0.800	0.450	0.298
d_{2T_y}	0.001	0.400	0.400	0.401
d_{2D_x}	0.100	0.800	0.450	0.370
d_{2D_y}	0.100	0.800	0.450	0.378
a	1.000	7.000	1.000	1.075
c_T	−0.300	0.300	0.200	0.101

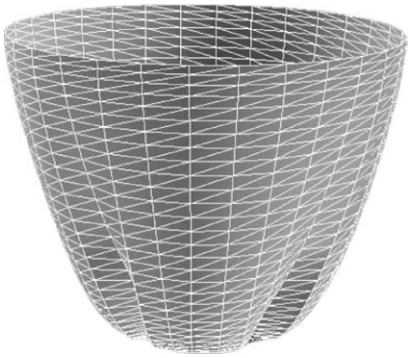

Fig. 6. Optimal design for strength of the container.

With the above formulation the design parameters and their initial values for the optimisation of the container are shown in Table 3. Note that the table also shows the chosen range for each parameter. The range specified for each design parameter (by means of choosing a maximum and a minimum) allows the parameters to be varied within the specified ranges enabling alternative shapes to be created within the design space automatically. These ranges are chosen to ensure that the geometry of sensible shapes are fed into the optimisation routine. The required strength of the container is specified so as the level of stress occurring within the loaded structure is always less than 30% of the yield stress. This is chosen to ensure that the designs which are prone to buckling are always rejected during the course of optimisation. The design space is further restricted by choosing a volume constraint for the container. For this particular example a fixed volume for the container of 150 ml was assumed.

Once the geometry is parametrised, the design parameters and their ranges along with the value for the required volume of the container are automatically varied by the BFGS optimisation routine. This routine searches the design space in order to find the design with lowest possible value of the chosen merit function. The optimisation took a little over 5 h to complete on a PC with a 1.2 GHz processor and 1 GB RAM. The values of the parameters obtained for the optimal design is shown in Table 3 and the optimal shape is shown in Fig. 6. The resulting optimal shape had a relative reduction in mass of 23.8%.

3.2. Numerical example illustrating the prediction of stable structures of vesicles occurring in biological organisms

This example discusses how the method of shape parametrisation based on the PDE formulation can be used to predict the stable structures of vesicles commonly found in biological organisms. The vesicles considered here are essentially lipid molecules typically consisting of a polar hydrophilic head and a hydrophobic tail made of hydrocarbon chains [10]. Such amiphiphilic molecules when placed in an aqueous solution can spontaneously aggregate to form encapsulating bags [5]. Despite the relatively simple structure of their walls, these vesicles can adopt a surprisingly wide variety of different shapes and even topologies. It is noteworthy that various shapes and topologies are adopted by these vesicles during the aggregation so as to reduce the surface energy of the membrane [18].

The aim here is to predict the stable shapes of the vesicles by means of automatic optimisation subject to a given set of physical condition with the merit function being the surface energy of the membrane. Various models for predicting the surface energy of a membrane can be found in the literature. For the work described here a widely accepted model for predicting the surface energy of a membrane due to Canham [6] that is based upon the surface curvature (SC) of the membrane is used. The SC model is based on the fact that the local energy density of the membrane is proportional to the sum of the squares of the principal curvatures and a quantity known as the spontaneous curvature to reflect the possible axisymmetric configuration of the membrane. It is noteworthy that particular shapes are adopted by a given vesicle in order to locally minimise the energy functional subject constraints of constant area and volume. Hence, for a given surface shape S, the surface energy E of a vesicle is given by a surface integral of the form

$$E(S) = \int (C_1 + C_2 - C_0)^2 \, \mathrm{d}A + \int (C_1 C_2) \, \mathrm{d}A, \tag{18}$$

where C_1 and C_2 are the principal curvatures, C_0 is the spontaneous curvature and $\mathrm{d}A$ is an element of the surface. As described in Section 2, the surface is given in closed form allowing the computation of principal curvatures and the Jacobian relating the area element $\mathrm{d}A$ in the (u, v) parameter space.

Note that although Eq. (18) is somewhat similar to the integral minimised by a minimal surface, the two problems are somewhat different. For example, in this case the optimisation is carried out for a closed surface and subject to the constraints of fixed volume and area.

The approach to predicting vesicle shapes adopted is similar to that described in the previous example where a parametric representation of the surface corresponding to the shape of vesicle is created. The geometry of the vesicle shape is represented using two PDE surface patches joined together with a common boundary as shown in Fig. 3. Once again the design parameters are identified at the boundary curves with their starting values and ranges directly fed to the BFGS routine. The optimisation is carried out subject to the constraints of constant surface area and enclosed volume. Due to the scale invariance of the SC model the vesicle shapes depend on two dimensionless parameters known as the reduced volume v and the reduced spontaneous curvature c_0 given by

$$v = V \Big/ \left(\frac{4\pi}{3} R_0^3 \right), \tag{19}$$

and

$$c_0 = C_0 R_0, \tag{20}$$

where $R_0 = 2\sqrt{\pi A}$, where V and A are the volume and surface area of the vesicle respectively.

With these settings the optimisation is started at some initially chosen point in the parameter space. The routine allows to detect the local minimum of the surface energy for a given value of the reduced spontaneous curvature as a function of the reduced volume. To achieve this a starting value of reduced spontaneous curvature c_0 and a starting set of values for the design parameters are chosen. This allows to find an initial stationary state for a given value of the reduced volume v. The optimisation is repeated for a new value of v using the previously found stationary state as a starting point for the new optimisation. Thus, for a given value of c_0 starting from $v = 0.5$ the optimisation was repeated with 0.1 increments until $v = 1.0$ has been reached.

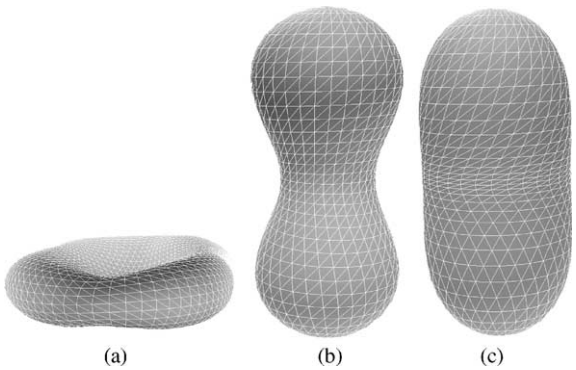

Fig. 7. Sample shapes of vesicles obtained during optimisation for $c_0 = 0$: (a) $v = 0.58$, (b) $v = 0.79$, (c) $v = 0.85$.

Figs. 7 and 8 show the results of sample vesicle shapes of different volumes obtained for $c_0 = 0$ and $c_0 = 3.0$ respectively. The optimisation took about 2 h to reach at each stationary state using the computing hardware mentioned earlier. These results were validated, by comparing the energy verses reduced phase diagrams obtained during the optimisation, against those reported in [19] and were found to be in close agreement.

4. Conclusions

The purpose of this paper was to present a methodology for automatic design optimisation based on a PDE formulation enabling efficient shape definition and shape parametrisation. It has been shown how the choice an elliptic PDE enables to create surfaces corresponding to complex shapes. Such shapes can be created using the boundaries or the character lines describing them. The solution of the chosen PDE enables to represent the geometry model whose shape can be con-

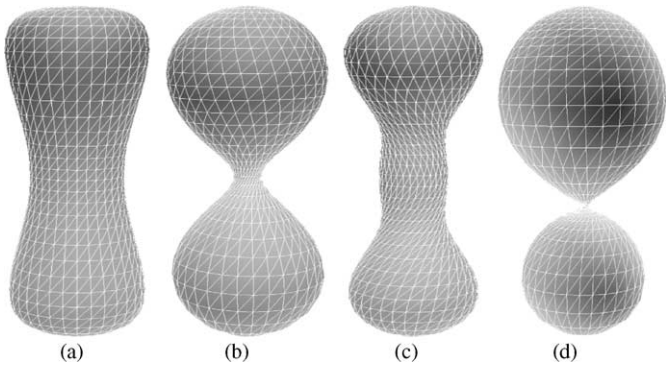

Fig. 8. Sample shapes of vesicles obtained during optimisation for $c_0 = 3.0$: (a) $v = 0.81$, (b) $v = 0.69$, (c) $v = 0.70$, (d) $v = 0.76$.

trolled by the chosen boundary conditions. Since the parametrisation of the geometry model is based on the boundary curves defining the shape, complex shapes can be created in an intuitive fashion.

This paper discusses two examples to demonstrate how the methodology can be used for automatic design optimisation. The first example demonstrates the automatic design of a thin-walled structure where the objective was to design a plastic container that uses the minimum amount of material subject to a given strength and volume. The second example discusses the predictions of stable shapes of biological vesicles by means minimising the surface energy associated with the vesicle membrane. These examples clearly demonstrate that the proposed approach to shape parametrisation when combined with a standard method for numerical optimisation is capable of setting up automatic design optimisation problems allowing practical design optimisation to be more feasible.

For the work presented here, to take advantage of approximate analytic solution to the chosen PDE, only shapes defined using periodic boundary conditions have been used. Thus one would argue that the range of shapes accessible to the optimisation routine to be somewhat limited. This is however not a drawback of the methodology since presently there exist efficient numerical schemes to solve elliptic PDEs. However, more work needs to be done in order to improve the shape definition and parametrisation so that freedom of local manipulation of shapes may be introduced so as to improve the scope of the method as a tool which can be widely used for automatic design optimisation.

Acknowledgements

The authors wish to acknowledge the support of EPSRC grant GR/M73125 under which some of this work was completed.

References

[1] Bloor MIG, Wilson MJ. Generating blend surfaces using partial differential equations. Computer-Aided Des 1989; 21:165–71.
[2] Bloor MIG, Wilson MJ. Using partial differential equations to generate freeform surfaces. Computer-Aided Des 1990;22:202–12.
[3] Bloor MIG, Wilson MJ. Spectral approximations to PDE surfaces. Computer-Aided Des 1996;28:145–52.
[4] Braibant V, Fleury C. Shape optimal design using B-splines. Comput Meth Appl Mech Eng 1984;44:247–67.
[5] Bouligand Y. Remarks on the geometry of micelles, bilayers and cell membranes. Liquid Cryst 1999;26(4): 501–15.
[6] Canham PB. The minimum energy of bending as a possible explanation of the biconcave shape of the human red blood cell. J Theor Biol 1970;26:61–81.
[7] Chang KH, Choi KK. A geometry based parametrisation method for shape design of elastic solids. Mech Struct Mach 1992;20:215–52.
[8] Dekanski CW, Bloor MIG, Wilson MJ. The representation of marine propeller blades using the PDE method. J Ship Res 1995;38(2):108–16.
[9] Francavilla A, Ramakrishanan CV, Zienkiewicz OCZ. Optimisation of shape to minimise stress concentration. J Strain Anal 1975;10:63–70.
[10] Gennis RB. Biomembranes: molecular structure and function. New York: Springer-Verlag; 1989.
[11] Greig DM. Optimisation. London: Longman; 1980.
[12] Hinton E, Owen DRJ. Finite element software for plates and shells. Swansea: Pineridge Press; 1984.
[13] Imam MH. Three-dimensional shape optimisation. Int J Numer Meth Eng 1982;18:661–73.
[14] Kodiyalam S, Kumar V, Finigan PM. Constructive solid geometry approach to three-dimensional shape optimisation. AIAA J 1992;30:1408–15.
[15] Mücke R. Remarks on the applicability of structural optimization methods in the practical engineering design process. Des Optim: Int J Product Process Improvement 1999;1(2):137–53.
[16] Niordson FI. Shell theory. Amsterdam: Elsevier Science Publishers; 1985.
[17] Press WH, Teukolsky SA, Vetterling WT, Flannery BP. Numerical recipes in C. UK: Cambridge University Press; 1992.
[18] Seifert U. Configurations of fluid membranes. Adv Phys 1997;46:13–137.
[19] Seifert U, Berndl K, Lipowsky R. Shape transformations of vesicles: phase diagrams for spontaneous-curvature and bilayer-coupling models. Phys Rev A 1991;44:1182–202.
[20] Schumaker LL. Spline functions: basic theory. New York: John Wiley and Sons; 1981.
[21] Tortorelli DA. A geometric representation scheme suitable for three-dimensional shape optimisation. Mech Struct Mach 1993;21:95–121.
[22] Ugail H, Bloor MIG, Wilson MJ. On interactive design using the PDE method. In: Dæhlen M, Lyche T, Schumaker LL, editors. Mathematical methods for curves and surfaces II. Nashville, TN: Vanderbilt University Press; 1998.
[23] Ugail H, Bloor MIG, Wilson MJ. Techniques for interactive design using the PDE method. ACM Trans Graph 1999;18(2):195–212.
[24] Ugail H, Bloor MIG, Wilson MJ. Manipulations of PDE surfaces using an interactively defined parametrisation. Comput Graph 1999;24(3):525–34.
[25] Ventsel E, Krauthammer T. Thin plates and shells. New York: Marcel Dekker; 2001.
[26] Vida J, Martin RR, Varady T. A survey of blending methods that use parametric surfaces. Computer-Aided Des 1994;26:341–65.
[27] Yao TM, Choi KK. 3-D shape optimal design and automatic finite element regridding. Int J Numer Meth Eng 1989;28:369–84.
[28] Yoo YM, Haug EJ, Choi KK. Shape optimal design of an engine connecting rod. ASME J Mech Transmission Automat Des 1984;106:415–9.