

Manipulation of Elastically Deformable Surfaces through Maya Plug-in

L. H. You, Javier Romero Rodriguez, Jian J. Zhang
National Centre for Computer Animation
Bournemouth University
United Kingdom

{lyou@bournemouth.ac.uk, javier@javier3d.com, jzhang@bournemouth.ac.uk}

Abstract

In this paper, we develop a mathematical model from the theory of plate bending in elasticity which relates physical properties of a surface to its elastic deformation. We present the finite difference solution of the mathematical model and implement it using the Maya API and the MEL scripting language. We examine the effects of material properties and other factors on surface shapes and demonstrate applications of the proposed approach in controlling the shape of elastically deformable surfaces.

Keywords---Elastically deformable surfaces, mathematical model, finite difference solution, Maya implementation, shape manipulation, computer animation

1. Introduction

Deforming objects to create natural and realistic shapes and movements is an interesting topic which has been investigated since the beginning of computer animation. By now, quite a number of methods have been developed for this aim.

Common efforts are to use Bézier, B-spline and NURBS to build geometrical models, and manipulate surface shapes through control points, knots and weights [1]. With these pure geometric methods, the realism of objects depends on the visual judgment of modellers. Therefore, different modellers will create somewhat different appearances of the same objects.

In order to attack this issue, physically based modeling and animation was introduced. Terzopoulos and his co-workers applied elastic theory to develop dynamic differential equations for flexible materials such as rubber, cloth and paper [2,3]. Later on, Terzopoulos and Fleischer further incorporated viscoelasticity, plasticity and fracture [4]. By introducing mass distributions, internal deformation energies, and other

physical quantities into NURBS, Terzopoulos and Qin proposed a dynamic NURBS [5]. With integration of dynamic NURBS into swung surfaces, Qin and Terzopoulos presented a dynamic NURBS swung surface method which was used to cope with surfaces with symmetries and topological variability [6]. They also extended triangular B-splines to triangular NURBS, formulated the mathematical model of dynamic triangular NURBS from Lagrangian mechanics, and employed the finite element method to solve the mathematical model and generate surfaces defined over arbitrary, nonrectangular domains [7]. Mandal et al. developed a new approach based on physics and geometric subdivision and used it to manipulate the smooth limit surface dynamically [8]. Léon and Verson, and Guillet and Léon applied a bar network mechanics method to deform free-form surfaces [9,10]. Celniker and Gossard minimized the energy functional under user controlled geometric constraints and loads, proposed a curve and surface finite element method, and applied it in free-form surface generation [11]. Güdükbay and Özgüç gave a physically based modeling algorithm consisting of a primal formulation and a hybrid formulation derived from the theory of pure elasticity and employed it to animate deformable objects [12]. Using forces as a main sculpting tool and minimizing the energy functional of a surface, Vassilev presented an efficient means to manipulate deformable B-spline surfaces [13].

Surfaces and solid volume can also be described by the solution to a partial differential equation subjected to suitably defined boundary conditions. This idea was first introduced for surface blending [14] and free-form surface generation [15] by Bloor and Wilson. In order to solve the fourth order partial differential equation efficiently, various analytical methods were proposed. The closed form solutions of partial differential equations for some simple blending surfaces were dealt with in [14] and those for generation of solid volume were given in [16]. Fourier series method was discussed in [15]. By using a remainder function to exactly satisfy boundary conditions, an approximate analytical method

was presented to perform partial differential equation (PDE) based surface modelling [17]. After that, a perturbation method was proposed to create blending surfaces [18]. In addition to the work in development of various analytical resolution methods, PDE based modeling approaches have been applied in various industries. For instance, Sevant et al. used partial differential equation method to parameterize a flying wing and obtained its design with maximized lift [19]. Mimis et al. introduced PDE based surface modeling into design of a two-stroke engine and optimized its scavenging properties based on computational fluid dynamics calculations [20].

PDE based modeling has many new features such as the capacity in using a single patch to define a complicated surface, more powerful and flexible manipulation in shape control of surfaces, and potential in merging functionality. Therefore, it has been paid a greater amount of attention in recent years. Ugail investigated how the spine of a PDE surface can be generated and applied in shape parameterization of the PDE surface [21]. He and his colleagues also examined interactive design using sixth order partial differential equations [22]. By combining partial differential equations with equation of motion, Du and Qin obtained some new approaches of dynamic surface and solid modeling [23,24,25]. You and Zhang studied different surface functions and achieved some efficient and accurate methods for surface blending [26,27] and generation [28].

By now, the partial differential equations used in surface and solid modeling are not directly related to material properties. The main objective of this paper is to address this issue, derive a mathematical model to reflect material properties and implement it into the animation software Maya for manipulation of elastically deformable surfaces.

2. Mathematical model

An arbitrary surface in a three-dimensional space has x , y and z components. Each of these components is a function of parametric variables u and v which are called position functions. The geometric representation of each position function changing with two parametric variables u and v is a three-dimensional surface. Using the theory of plate bending in elasticity, we can derive a mathematical model describing the relationship between the position functions and parametric variables u and v , and relate material properties of a surface to its deformation. In the following, we take x component as an example and introduce the derivation of the mathematical model.

As indicated in Figure 1, an infinitesimal element of $du \times dv$ is taken out from a surface describing the relation between x position function and parametric variables u and v . An arbitrary external force P is applied on the infinitesimal element. This force can be a concentrated force, a line distributed force or an area

distributed force. Considering all these cases, we can write this force in the form of $P(u, v)$. Under the action of this force, the surface will deform and the caused internal forces in the surface can be described with those acting on the four boundaries of the infinitesimal element. These internal forces are: shear forces Q_u and Q_v , bending moments M_u and M_v , and the twisting moment $M_{uv} = M_{vu}$.

Taking the equilibrium of moments around u and v axes, we obtain

$$\begin{aligned} Q_u &= \frac{\partial M_u}{\partial u} + \frac{\partial M_{vu}}{\partial v} \\ Q_v &= \frac{\partial M_v}{\partial v} + \frac{\partial M_{uv}}{\partial u} \end{aligned} \quad (1)$$

The condition of force equilibrium along x -axis gives the following equation

$$\frac{\partial Q_u}{\partial u} + \frac{\partial Q_v}{\partial v} + P(u, v) = 0 \quad (2)$$

Substituting equation (1) into (2), one obtains the equilibrium equation below

$$\frac{\partial^2 M_u}{\partial u^2} + 2 \frac{\partial^2 M_{uv}}{\partial u \partial v} + \frac{\partial^2 M_v}{\partial v^2} + P(u, v) = 0 \quad (3)$$

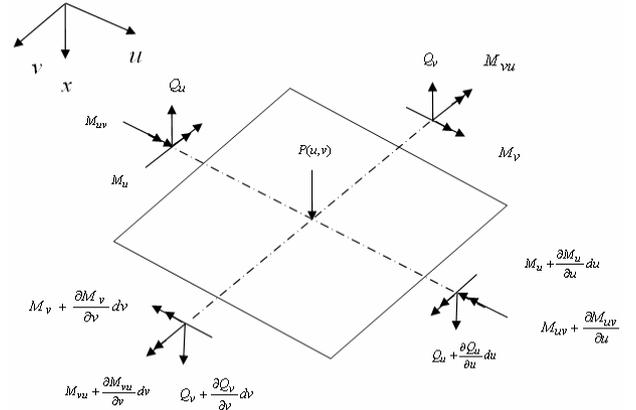


Figure 1 The forces and moments acting on an infinitesimal element of a surface

All surfaces in the natural world exhibit certain material or physical properties. These properties can be described with elasticity, isotropy, orthotropy and so on. Here, we consider elastic and orthotropic cases. Similar to those given in the textbooks of elasticity or plates, the relations between the bending moments, twisting moment and position function x have the forms of

$$\begin{aligned} M_u &= -\frac{E_u t^3}{12(1 - \mu_u \mu_v)} \left(\frac{\partial^2 x}{\partial u^2} + \mu_v \frac{\partial^2 x}{\partial v^2} \right) \\ M_v &= -\frac{E_v t^3}{12(1 - \mu_u \mu_v)} \left(\frac{\partial^2 x}{\partial v^2} + \mu_u \frac{\partial^2 x}{\partial u^2} \right) \\ M_{uv} &= -\frac{G t^3}{6} \frac{\partial^2 x}{\partial u \partial v} \end{aligned} \quad (4)$$

where t is the thickness, E_u and E_v are Young's moduli, μ_u and μ_v are Poisson's ratios, and G is the shear modulus of a surface.

Substituting (4) into (3) and combining the same terms, we obtain the equation for the x component as below

$$\frac{E_u t^3}{12(1-\mu_u \mu_v)} \frac{\partial^4 x}{\partial u^4} + \left[\mu_v \frac{E_u t^3}{6(1-\mu_u \mu_v)} + \frac{G t^3}{3} \right] \frac{\partial^4 x}{\partial u^2 \partial v^2} + \frac{E_v t^3}{12(1-\mu_u \mu_v)} \frac{\partial^4 x}{\partial v^4} = P(u, v) \quad (5)$$

After some mathematical operations, Eq. (5) can be further transformed into the following equation

$$E_u \frac{\partial^4 x}{\partial u^4} + [2\mu_v E_u + 4(1-\mu_u \mu_v)G] \frac{\partial^4 x}{\partial u^2 \partial v^2} + E_v \frac{\partial^4 x}{\partial v^4} = \frac{12(1-\mu_u \mu_v)}{t^3} P(u, v) \quad (6)$$

Through the above equation, the material properties of a surface are related to the position function of the surface.

For the y and z components, similar equations to (6) can be derived. In the interest of space, their concrete forms are not provided here.

Given two non-coincident 3D curves, we can create a surface which passes these two curves. The slope of the surface can be defined with the tangents of the surface at these two curves. Therefore, the boundary conditions for the surface can be regarded as a combination of boundary curves and tangents. Still taking x component as an example, the boundary conditions required in solving Eq. (6) are represented by the following equations

$$\begin{aligned} u=0 & \quad x=b_0(v) & \quad \frac{\partial x}{\partial u} &= b_1(v) \\ u=1 & \quad x=b_2(v) & \quad \frac{\partial x}{\partial u} &= b_3(v) \\ v=0 & \quad x=b_4(u) & \quad \frac{\partial x}{\partial v} &= b_5(u) \\ v=1 & \quad x=b_6(u) & \quad \frac{\partial x}{\partial v} &= b_7(u) \end{aligned} \quad (7)$$

where $b_0(v)$, $b_2(v)$, $b_4(u)$ and $b_6(u)$ are the boundary curves, and $b_1(v)$, $b_3(v)$, $b_5(u)$ and $b_7(u)$ are the boundary tangents.

Equations (6) and (7) give a mathematical model which defines an elastically deformable surface through its material properties, geometric parameters, boundary constraints and external forces.

3. Resolution

The mathematical model represented by Eqs. (6) and (7) is very difficult to solve analytically. In order to present a general resolution method, we here consider the finite difference method. It is to transform a

mathematical model into a set of linear algebraic equations. Using the mesh given in Figure 2 and central difference approximation, the first and fourth partial derivatives can be transformed into

$$\begin{aligned} \left(\frac{\partial x}{\partial u} \right)_0 &= \frac{x_1 - x_3}{2h} \\ \left(\frac{\partial x}{\partial v} \right)_0 &= \frac{x_2 - x_4}{2h} \\ \left(\frac{\partial^4 x}{\partial u^4} \right)_0 &= \frac{1}{h^4} [6x_0 - 4(x_1 + x_3) + x_9 + x_{11}] \\ \left(\frac{\partial^4 x}{\partial u^2 \partial v^2} \right)_0 &= \frac{1}{h^4} [4x_0 - 2(x_1 + x_2 + x_3 + x_4) \\ &\quad + x_5 + x_6 + x_7 + x_8] \\ \left(\frac{\partial^4 x}{\partial v^4} \right)_0 &= \frac{1}{h^4} [6x_0 - 4(x_2 + x_4) + x_{10} + x_{12}] \end{aligned} \quad (8)$$

where h is the interval in the parametric directions u and v .

Substituting Eq. (8) into Eq. (6), a linear algebraic equation below for each interior node was obtained

$$\begin{aligned} & 2\{(3+4\mu_v)E_u + [3E_v + 8(1-\mu_u \mu_v)G]\}x_0 - 4 \\ & [(1+\mu_v)E_u + 2(1-\mu_u \mu_v)G](x_1 + x_3) - 4 \\ & [\mu_v E_u + E_v + 2(1-\mu_u \mu_v)G](x_2 + x_4) + 2 \\ & [\mu_v E_u + 2(1-\mu_u \mu_v)G](x_5 + x_6 + x_7 + x_8) \\ & + E_u(x_9 + x_{11}) + E_v(x_{10} + x_{12}) \\ & = \frac{12(1-\mu_u \mu_v)}{t^3} h^4 P_{x0} \end{aligned} \quad (9)$$

where P_{x0} is the force component acting at the typical node 0 in x direction.

Introducing Eq. (8) into Eq. (7), the boundary conditions become

$$\begin{aligned} x_0(u=0) &= b_0(v_0) \\ x_1(u=0) - x_3(u=0) &= 2hb_1(v_0) \\ x_0(u=1) &= b_2(v_0) \\ x_1(u=1) - x_3(u=1) &= 2hb_3(v_0) \\ x_0(v=0) &= b_4(u_0) \\ x_2(v=0) - x_4(v=0) &= 2hb_5(u_0) \\ x_0(v=1) &= b_6(u_0) \\ x_2(v=1) - x_4(v=1) &= 2hb_7(u_0) \end{aligned} \quad (10)$$

where $u=0$, $u=1$, $v=0$ and $v=1$ indicate that the typical node is at the boundary curves defined by these quantities respectively, and u_0 and v_0 are parametric values of the typical node at the boundary curves.

Putting Eqs. (9) and (10) together and writing the resulting linear algebraic equations in the form of matrix, we reach

$$\mathbf{KX} = \mathbf{F} \quad (11)$$

where \mathbf{K} is a square matrix consisting of the coefficients before the unknown constants determined by Eqs. (9) and (10), \mathbf{X} and \mathbf{F} are column vectors consisting of the

unknown position functions at all nodes and the known constants, respectively.

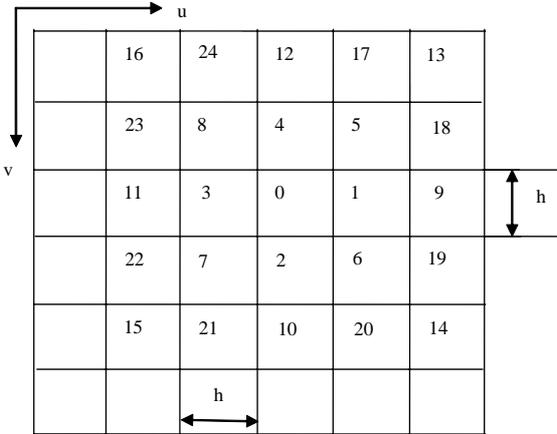


Figure 2 Mesh of finite difference approximation

Eq. (11) is a sparse matrix. We use the resolution method for large, sparse matrix and the corresponding standard C++ program to solve the linear algebraic equations defined by Eq. (11). The developed software module is called ElastSolver.

It can be seen that the time spent to obtain the solution depends on the number of the nodes. When the node number is not very large, the proposed finite difference method can reach real-time performance.

4. Maya implementation

In order to take full advantage of the powerful functions of the animation software Maya, we implement the above mathematical model into Maya 6.0/7.0 package through the API using C++. Some scripts are developed as well using MEL (Maya Embedded Language).

When a manipulation task of elastically deformable surfaces starts, we firstly specify a deformation region which may be global or local as well as a force acting region, then input physical properties and deformation constraints, obtain the geometric information of objects from Maya, pass them into ElastSolver to calculate deformations of the objects, and output the deformed information of the objects into Maya to generate the new shapes of the objects. The general structure of Maya implementation is shown in Figure 3.

Maya is used as a central information gathering tool. The correct definition of the data flow system between the software, the mesh, the ElastSolver engine and users is the key to success of our implementation. In Maya, data are transferred between a network of nodes in the Dependency graph. DataFromMaya node is responsible for passing the geometric information of un-deformed objects including mesh information and the values of parametric variables u and v and position functions x , y and z at all nodes onto ElasticNode.

Through a user friendly interface UserInput, a deformation region and a force acting region are determined. Then, the direction and size of the forces $P_i(u, v)$ ($i = x, y, z$) are specified. Finally, material properties which are Young's moduli E_u, E_v , Poisson's ratios μ_u, μ_v , shear modulus G , surface thickness h , and boundary tangent constraints $b_1(v), b_3(v), b_5(u)$ and $b_7(u)$ are inputted. All the information is transferred to the ElasticNode.

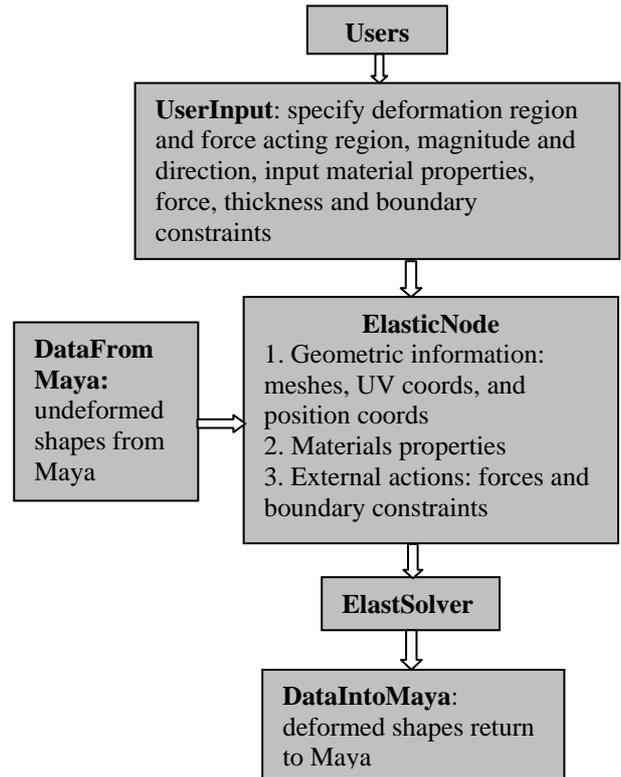


Figure 3 General structure of Maya implementation

Once the ElasticNode has received the data form both Maya through DataFromMaya and the user, it processes them and transforms them into those required by the mathematical model and sends them to ElastSolver. With the above finite difference method, ElastSolver solves the set of linear algebraic equations determined by the input information and geometric data, and obtains the deformed shape of the objects. Then, it gives the deformation information back to Maya via DataIntoMaya and Maya generates the deformed objects.

5. Numerical Applications

With the Maya implementation developed above, in this section, we examine material properties and other factors which affect the shape of surfaces and present some examples to demonstrate the applications of the proposed method in manipulation of elastic surfaces and objects.

In order to investigate how different material properties, surface thickness, boundary tangents and force functions are related to the shape of surfaces, the deformations of a planar surface will be discussed below. For an arbitrary surface such as a curved surface, the method given here still applies. For this case, the final deformed position will be the sum of the initial position and the deformation of the surface.

The basic data of material properties, boundary tangent and force functions are taken to be: Young's moduli $E_u = E_v = 1$, Poisson's ratios $\mu_u = \mu_v = 0.3$ shear modulus $G = 1$, surface thickness $t = 1$, boundary tangents $b_1 = b_3 = b_5 = b_7 = 0$, and force functions $P_x = P_y = 0$ and $P_z = 50$. The surface shape caused by these parameters is depicted in Figure 4a.

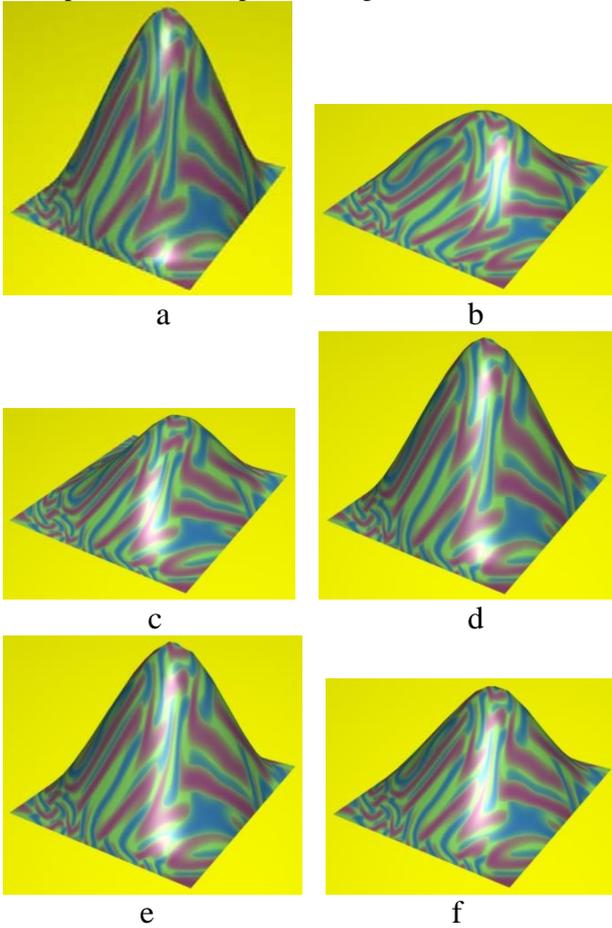


Figure 4 The effects of material properties

Firstly, the effects of material properties were considered. To this end, all other parameters are kept unchanged, and only one material property was altered each time. By changing E_u to 5, the surface in Figure 4b was obtained. Due to the increase of Young's modulus, the stiffness of the surface against deformations rises, leading to a significant reduction of surface deformation. Varying E_v to 5 generates the image in Figure 4c. Although the values of two Young's moduli are identical, they act in different directions which results in the visible difference of surface shapes. Poisson's ratios

can also make an obvious impact on surface shapes. Raising Poisson's ratio μ_u from 0.3 to 0.8, the surface shape was changed to that in Figure 4d. It is quite different from that in Figure 4a. However, when two Poisson's ratios take a same value, the difference between two surfaces is very small as shown in Figure 4d and Figure 4e where μ_v was also increased from 0.3 to 0.8. It indicates when implementing Poisson's ratios into a user's handle, only one of them need to be taken into account. The influence of shear modulus G on the surface shape is similar to that of Young's moduli. A variation of shear modulus from 1 to 3 reduces the surface deformation and brings in the image indicated in Figure 4f.

Next, we study how surface thickness changes the shape of surfaces. It is easy to understand that thicker the surface thickness, larger the deformation resistance. Setting the value of surface thickness t to 1.5, the surface deformation was noticeably lowered as demonstrated in Figure 5.

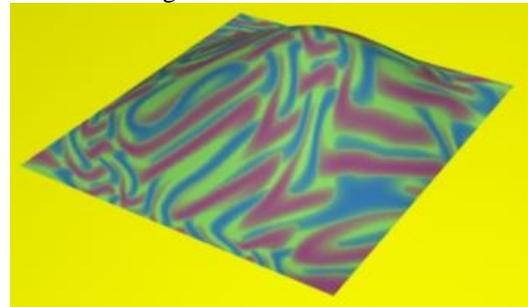


Figure 5 The effects of surface thickness

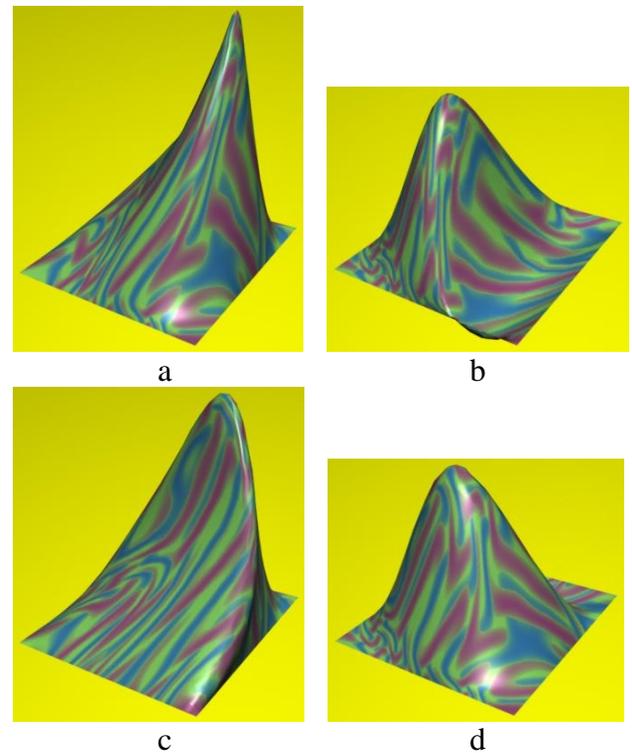


Figure 6 The effects of boundary tangents

Thirdly, how boundary tangents affect the surface shape is tackled. Taking boundary tangents $b_1 = 5$, $b_3 = 2.5$, $b_5 = 4$ and $b_7 = 2$, respectively, the surface shapes were changed into those given in Figure 6a, 6b, 6c and 6d. Clearly, different values of boundary tangents bring in different deformations. The largest deformation occurs in the region near the boundary whose boundary tangent is changed. Then, the deformation is weakened gradually when moving towards the inner area of the surface.

Finally, the influence of force functions is examined. By setting the force component P_z to -30, the image in Figure 7a was produced. Since the direction of the force is reversed, the deformation in the opposite direction is induced. Keeping $P_z = 50$ and introducing $P_x = 50$ and $P_y = -80$, respectively, the surface shapes in Figure 7b and Figure 7c were generated. These two force components result in the variation of both size and direction of the deformation, strongly supporting the argument that the force function can make a profound difference to the surface deformation.

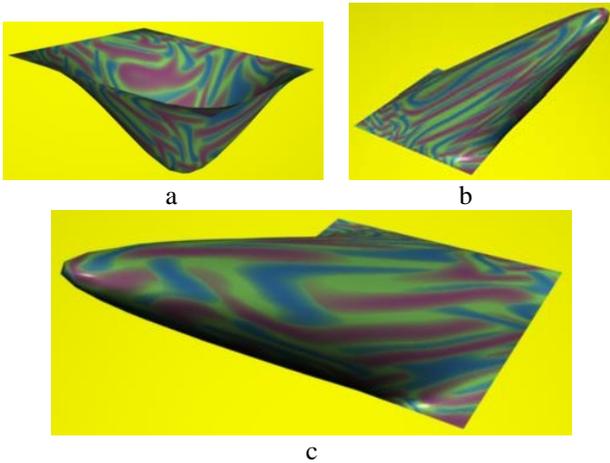


Figure 7 The effects of force functions

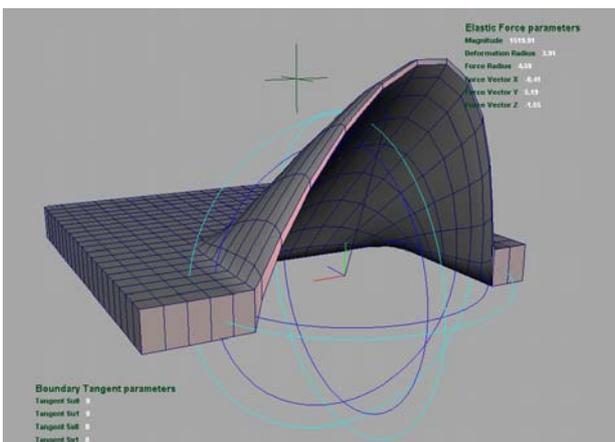


Figure 8 Pull of a ductile block

With the above Maya plug-in, all the factors affecting surface deformations are the user handles, with

which the user can easily control surface shapes and obtain a desired surface.

Apart from shape manipulation of elastically deformable surfaces, the developed approach can be used to control deformations of objects which consist of a number of surface patches. Applying the Maya implementation presented above, we carried out the shape manipulation of objects shown in Figures 8, 9 and 10. In Figure 8, a ductile block is pulled upward. In Figure 9, the block is subjected to a twist action. In Figure 10, a flexible thick sheet is deformed by two spheres with different radii.

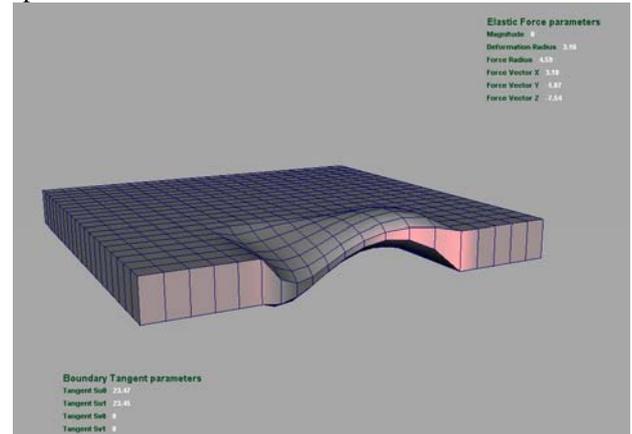


Figure 9 Twist of a ductile block

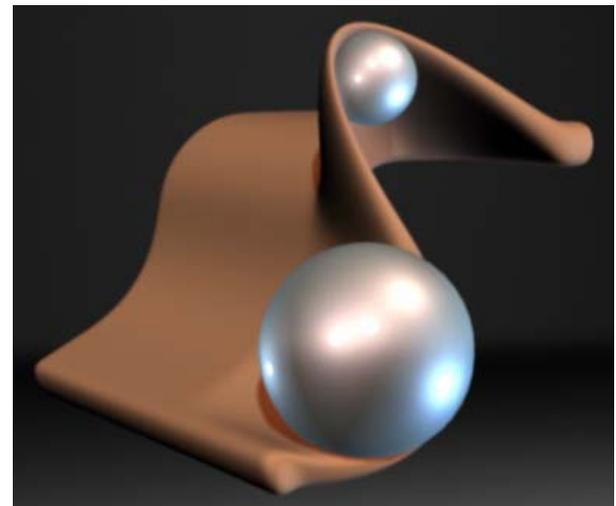


Figure 10 A flexible thick sheet under action of two spheres

6. Conclusions

In order to relate material properties to deformations of surfaces, in this paper, we used the theory of plate bending in elasticity to develop the mathematical model of physically based elastic deformations. With the obtained model, Young's moduli, shear modulus, Poisson's ratios, surface thickness, externally applied force and boundary constraints are linked together and

jointly contribute to physically based deformations of surfaces. We also presented the finite difference method to solve the mathematical model and implemented it into Maya. With the developed Maya implementation, we discussed the effects of material properties, surface thickness, force function and boundary tangent constraints on surface shapes, and applied our approach in manipulation of some complicated deformations of objects.

Acknowledgements

This project is in part supported by a grant from the British Arts and Humanities Research Council. The authors are grateful for the donation of the Maya licenses from Autodesk.

References

- [1] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*. Academic Press (4th Edition), 1997.
- [2] D.Terzopoulos, J. Platt, A. Barr and K. Fleischer. Elastically deformable models. *Computer Graphics*, 1987, 21(4):205-214.
- [3] D.Terzopoulos and K. Fleischer. Deformable models. *The Visual Computer*, 1988, 4:306-331.
- [4] D.Terzopoulos and K. Fleischer. Modeling inelastic deformation: viscoelasticity, plasticity, fracture. *Computer Graphics*, 1988, 22(4):269-278.
- [5] D. Terzopoulos and H. Qin. Dynamic NURBS with geometric constraints for interactive sculpting. *ACM Transactions on Graphics*, 1994, 13(2):103-136.
- [6] H. Qin and D. Terzopoulos. Dynamic NURBS swung surfaces for physical-based shape design. *Computer-Aided Design*, 1995, 27(2):111-127.
- [7] H. Qin and D. Terzopoulos. Triangular NURBS and their dynamic generations. *Computer Aided Geometric Design*, 1997, 14:325-347.
- [8] C. Mandal, H. Qin and B.C. Vemuri. Dynamic modeling of butterfly subdivision surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 2000, 6(3):265-287.
- [9] J.C. Léon and P. Veron. Semiglobal deformation and correction of free-form surfaces using a mechanical alternative. *The Visual Computer*, 1997, 13:109-126.
- [10] S. Guillet and J.C. Léon. Parametrically deformed free-form surfaces as part of a variational model. *Computer-Aided Design*, 1998, 30(8):621-630.
- [11] G. Celniker and D. Gossard. Deformable curve and surface finite-elements for free-form shape design. *Computer Graphics*, 1991, 25(4):257-266.
- [12] U. Güdükbay and B. Özgüç. Animation of deformable models. *Computer-Aided Design*, 1994, 26(12): 868-875.
- [13] T.I. Vassilev. Interactive sculpting with deformable nonuniform B-splines. *Computer Graphics Forum*, 1997, 16:191-199.
- [14] M.I.G. Bloor and M.J. Wilson. Generating blend surfaces using partial differential equations. *Computer-Aided Design*, 1989, 21(3):165-171.
- [15] M.I.G. Bloor and M.J. Wilson. Using partial differential equations to generate free-form surfaces. *Computer-Aided Design*, 1990, 22(4):202-212.
- [16] M.I.G. Bloor and M.J. Wilson. Functionality in solids obtained from partial differential equations. *Computing (Suppl.)*, 1993, (8): 21-42.
- [17] M.I.G. Bloor and M.J. Wilson. Spectral approximations to PDE surfaces. *Computer-Aided Design*, 1996, 28(2):145-152.
- [18] M.I.G. Bloor, M.J. Wilson and S.J. Mulligan. Generating blend surfaces using a perturbation method. *Mathematical and Computer Modelling*, 2000, 31(1):1-13.
- [19] N.E. Sevant, M.I.G. Bloor and M.J. Wilson. Aerodynamic design of a flying wing using response surface methodology. *Journal of Aircraft*, 2000, 37(4):562-569.
- [20] A.P. Mimis, M.I.G. Bloor and M.J. Wilson. Shape parameterization and optimization of a two-stroke engine. *Journal of Propulsion and Power*, 2001, 17(3):92-498.
- [21] H. Ugail. Spine based shape parameterisation for PDE surfaces. *Computing*, 2004, 72:195-206.
- [22] S. Kubiesa, H. Ugail and M.J. Wilson. Interactive design using higher order PDEs. *The Visual Computer*, 2004, 20:682-693.
- [23] H. Du and H. Qin. Direct manipulation and interactive sculpting of PDE surfaces. *Computer Graphics Forum (Eurographics 2000)*, 2000, 19(3):261-270.
- [24] H. Du and H. Qin. A shape design system using volumetric implicit PDEs. *Computer-Aided Design*, 2004, 36:1101-1116.
- [25] H. Du and H. Qin. Dynamic PDE-based surface design using geometric and physical constraints. *Graphical Models*, 2005, 67:43-71.
- [26] L.H. You, P. Comninou and J.J. Zhang. PDE blending surfaces with C^2 continuity. *Computers & Graphics*, 2004, 28:895-906.
- [27] L.H. You, J.J. Zhang and P. Comninou. Blending surface generation using a fast and accurate analytical solution of a fourth order PDE with three shape control parameters. *The Visual Computer*, 2004, 20:199-214.
- [28] J.J. Zhang and L.H. You. Fast surface modelling using a 6th order PDE. *Computer Graphics Forum*, 2004, 23(3):311-320.