Haptic Subdivision: an Approach to Defining Level-of-detail in Haptic Rendering

Jian Zhang, Shahram Payandeh and John Dill
Computer Graphics and Robotics Laboratories,
School of Engineering Science
Simon Fraser University
Burnaby, BC V5A 1S6, Canada
{jzhanga|shahram|dill}@cs.sfu.ca

Abstract

Soft objects are often desired in applications such as virtual surgery training. Soft object simulations are computationally intensive because object deformation involves numerically solving a large number of differential equations. However, realistic force feedback requires deformation be computed fast and graphic feedback requires deformation be highly detailed. In this paper, we propose an approach that balances these requirements by subdividing the area of interest on a relatively coarse mesh model. Thus we keep the number of nodes of the model under control so that the simulation can be run at a sufficiently high rate for force feedback. The model we use is based on a mass-spring model. When a portion of the surface is subdivided, new values of mass and spring constants are determined such that computed force feedback offers the user the same reaction force as before subdivision.

1. Introduction

In recent years, haptic feedback began to play a role in the development of virtual environment simulations. Devices such as the PHANToM [23] can provide the user with a sense of touch to some degree by sending reaction forces to the user's hands, providing a new way for the user to interact with objects in a virtual environment. Of interest are so-called "soft" objects that deform under applied forces, such as organs simulated as parts of medical training systems or surgical simulation systems.

Two common physics-based models used for representing deformable objects are mass-spring models and the finite element method (FEM). In general the mathematics involved in mass-spring models is relatively simple and easy to implement. For example a compression wave model on a mass-spring system to achieve bulging and curving motions of worms and snakes was proposed by Miller [1]. A 3D mass-spring lattice to simulate contraction of human arm muscles is given in [2]. An effort to model the surface force characteristics of the human thigh, a two layer mass-spring model with linear and nonlinear springs is proposed in [7][8] to fit the force curve obtained from probing a real human thigh. A spring-damper model to represent the vertebrae in an epidural injection simulator is used in [4]. Most mass-spring models are tailored for the specific application under investigation, thus the mass-spring models used in [1], [2] and [7] are substantially different. In [1], a chain of masses connected by springs would suffice. This is not the case in [2] where volumetric deformation is highly desired; therefore a 3D lattice of masses and springs is needed. If highly detailed deformation features are desirable, a 3D lattice of masses and springs would introduce too much computation. In this paper we adopt a mass-spring model similar to that proposed in [7], mainly focused on modeling object surfaces, but which takes into account internal material property of an object.

The finite element method (FEM) is based on continuum mechanics and numerical analysis. It offers more accuracy than the mass-spring models in physically based simulation. [18] and [6] modeled facial tissue to predict the results of craniofacial surgeries. There are a hepatic surgery simulation system proposed by [20] and an eye surgery simulation system by [21]. However, the need to manipulate large matrices and solve large numbers of differential equations imposes performance difficulties to apply FEM to real time interactive haptic feedback applications. Some work has been done on FEM models to make a trade-off between accuracy and update rate. For example, [14] used "finite spheres" as an approximation to FEM in a local area; [15] and [16] interpolated forces between calculated forces from a low update rate deformation model to feed the demand of high update rate haptic simulation. The other approach to speed up run-time simulation is to isolate some procedures of the real-time computation, precompute them and later combine precomputed results with states of a simplified run-time model. A boundary element method to compute system response of a linear model offline was proposed by [17]. Precomputation of "elementary deformation" is also an important part of [20] for hepatic surgery simulation system. [19] used implicit integration

to achieve stability with large time steps in their cloth simulation. However they did not target at real-time simulation.

Surface subdivision is a useful tool in computer graphics to create smooth surfaces out of polygon meshes. The two main categories of subdivision schemes are interpolatory schemes and approximating schemes. Typical approximating schemes include Doo-Sabin [23], Catmull-Clark [24] and Loop [3]. Typical interpolatory schemes include "modified butterfly" [26] and Kobbelt [27]. Recently [12] proposed a new method to interactively position subdivision control vertices by applying forces to a finite element model composed of these vertices. Subdivision schemes introduce new nodes into the local area of the mesh. If the local area is also deformed, more nodes indicate that more detailed deformation features can be expressed than before subdivision.

In this paper we propose the concept of haptic subdivision with levels of detail as a methodology to balance update rate and computational complexity. When the user interacts with the object (a mass-spring model), the local area of the physical model is refined, while other parts of the model remain coarse. Deformation and force will then be computed using this refined model with mixed levels of details. The main difference of our approach from others' work is that refinement is performed directly on the underlying physical model, rather than on its graphic representation.

Some parameters of the physical model need to be determined after the refinement, including masses and spring constants. In general it is difficult to obtain an analytical expression of the spring constants after subdivision in terms of those before subdivision [28], therefore we propose a guideline on how to determine these parameters. Experiments based on this guideline have revealed the relationship between parameters.

The rest of this paper is divided into the following sections: Section 2 is a brief description of the mass-spring model used in this paper. Section 3 provides the details of surface subdivision, including the determination of parameters of physical model. Section 4 is a brief discussion of our result and in section 5 we envision some possible directions of future work.

2. Mass-spring model

In this paper we use a mass-spring model for surface modeling. Although easy to implement, the mass-spring model is capable of simulating a wide range of non-rigid behavior found in nature. In our model, the simulated surface is divided into small triangles, where at each vertex a mass point (a node) is defined. A linear spring is mounted along each triangle edge. These springs are called "mesh springs" because they model the surface of the object. The mesh spring constant is denoted K_m.

When a soft elastic object deforms, the interior of the object also contributes to both the shape of deformation and the force feedback to the user. To reflect this fact in our model, each node is also connected by a spring to its initial position, thus "home springs" denoted K_h. We add simple damping to the mesh by applying a force proportional to the velocity of each mass point, but in the opposite direction.
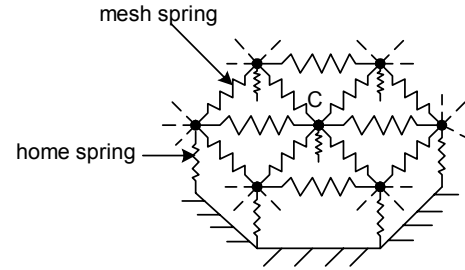

Figure 1: part of a mass-spring model

Figure 1 shows part of the modeled surface. The mass-spring model can be described as a system of differential equations. $\mathbf{x}$, $\mathbf{v}$ and $\mathbf{a}$ are position, velocity and acceleration vectors; f_i is the total force vector on a node and m_i is the node's mass value.

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{a} \end{bmatrix}, \textbf{ where } \mathbf{a} = \begin{bmatrix} f_1/m_1 \\ f_2/m_2 \\ . \\ . \\ f_n/m_n \end{bmatrix}. \quad (1)$$

f_i can be written as

$$f_i = -K_d v_i + K_h \cdot l_i + K_m \cdot \sum_j \frac{l_{ij}}{|l_{ij}|}\left(l_{ij} - r_{ij}\right) + f_i^e \quad (2)$$

where l_i is the displacement vector from the node's current position to its initial position; l_{ij} is the vector pointing from the i th node to its j th directly connected neighbor node; r_{ij} is the rest length of the mesh spring connecting the i th node and its j th neighbor node; f_i^e is the external force applied on the i th node. In equation (2) we see the force exerted on the node in question consists of two major components: internal forces and external forces. Internal forces include damping effect, home spring force, and mesh spring forces.

3. Haptic subdivision

The portion of the object being deformed by user interaction is an area of high interest. It thus seems clear

that such areas should be represented by a finer mesh than the rest of the object.

Thus for computational efficiency, we would like to work with a relatively coarse mesh, subdivided to a finer mesh in the area of interest. Since we provide both visual and haptic rendering, both meshes need be subdivided. Doing this for the geometric mesh is straightforward. However, it is more complex to subdivide the physical (haptic) mesh since the mass spring parameters must also be "subdivided", i.e. recalculated.

In this section we first briefly review geometric subdivision, then describe the experimental study we did to find suitable values for physical parameters during haptic subdivision.

3.1. Geometric Subdivision

Figure 2 (a) shows a part of a coarse triangular mesh and (b) shows the same part after subdivision. All nodes in (b) are new nodes. New nodes can be divided into two categories: one includes the nodes corresponding to the original nodes, such as node A', shown as large dots in (b); the other includes the nodes corresponding to the original edges, such as node E, shown as small dots. For convenience, we call the two kinds of nodes "type 1" and "type 2" respectively in the rest of this paper. The thick lines in (b) are new edges connecting a type 1 node to a type 2 node. These lines outline the shape of the original mesh. The thin lines are new edges connecting two type 2 nodes. Dashed line segments represent the edges connecting to the other nodes on the mesh. Every triangle on the original mesh is split into four smaller ones (in this example), and these new triangles form the new mesh. By introducing new nodes and new triangles properly into the new mesh, a more accurate representation of the smooth surface is obtained.

Figure 2 shows subdivision of a triangular mesh. Subdivision schemes dedicated to triangular meshes include the Loop [3] and the Modified Butterfly [26] schemes. There are also schemes for other kinds of polygons, such as the Catmull-Clark [24] and the Kobbelt [27] schemes for quadrilaterals. One of the main properties shared by all schemes is locality, i.e., the position of a new node should only depend on its corresponding neighborhood on the original mesh. For example, in Figure 2 (b), the position of node E only depends on nodes A, B, C and D on the original mesh. Locality guarantees that moving a node on the original mesh only affects the shape of subdivided mesh locally.

The mesh in Figure 2 (b) was obtained by using the Loop scheme on the mesh in Figure 2 (a). The Loop scheme generates type 1 and type 2 nodes. If the original mesh has V nodes and E edges, the number of nodes on the new mesh will be V+E. The locality property of the Loop scheme is embodied by the fact that all new node positions are defined as a weighted sum of the positions

of the neighbor nodes on the original mesh. Figure 3 shows the patterns used to define positions for type 1 and type 2 nodes.

(a)

(b)
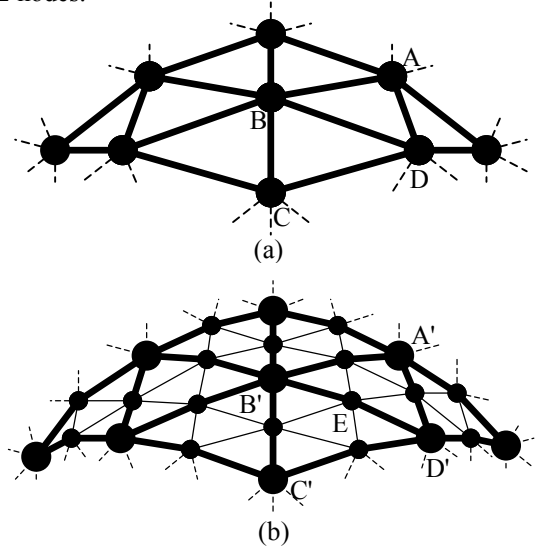
Figure 2: subdivision eg.. In (a), large dots are the original nodes. In (b), large dots are the new nodes corresponding to the original nodes. Smaller dots are new nodes corresponding to the original edges.


(a): Pattern for type 2 nodes   (b): Pattern for type 1 nodes
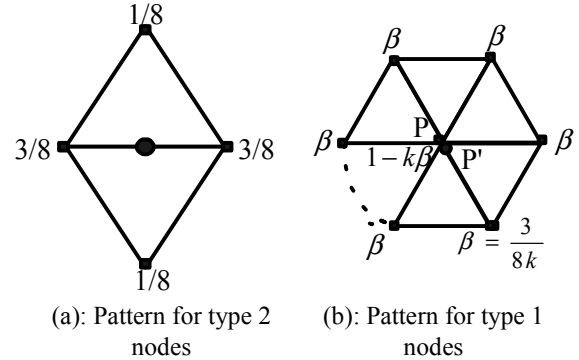
Figure 3: patterns used in the Loop scheme for interior nodes. K is the number of neighbor nodes directly connected to the original node P. β is the weighting factor for the neighbor nodes. Both in (a) and (b) the weighting factors sum to 1.

In Figure 3 small filled rectangles are nodes on the original mesh; small filled circles are the new nodes introduced by subdivision. The patterns in Figure 3 are parts of the interior of the original mesh. Boundary cases are not considered for now. There are nodes surrounding the patterns, but because of locality, those nodes do not contribute to the positions of the new nodes in Figure 3. The fractions beside the original nodes are weighting factors applied to the coordinates of the original nodes to obtain those of the new node.

If a new node is at the boundary of the new mesh, other rules have to be applied (Figure 4). Again we see the property of locality: boundary nodes are weighted

sums of nodes in that local area. The weighting factors again sum to 1.

$$\frac{1}{2} \qquad\qquad \frac{1}{2} \qquad\qquad \frac{1}{8} \qquad \frac{3}{4} \qquad \frac{1}{8}$$

(a)   (b)
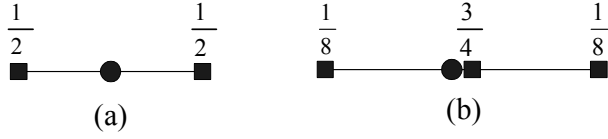
Figure 4: patterns used in the Loop scheme for boundary nodes; (a) is type 2; (b) is type 1.

In general, gaps can occur between subdivided areas and non-subdivided areas. This is corrected by introducing new polygons to fill the gaps.

3.2. Determination of physical parameters

In this study the user is assumed to interact with a contacted node on the mesh. Our method will apply equally well if the user interacts with a finite area. To facilitate the discussion about haptic subdivision, we define the "first ring" around a contact node (e.g. node C in Figure 1) to be the area containing all the nodes that are connected directly to that node, as well as the node itself. The "second ring" is defined as the area containing all the nodes that have a direct connection to the first ring, including the first ring. Third ring", "fourth ring" etc. are defined similarly. We name subdivision in the first ring "one-ring subdivision". "Two-ring subdivision" and so forth are defined similarly. As the number of rings involved in subdivision increases, the number of nodes on the new mesh and computational cost increases exponentially. Therefore we focus on one-ring and two-ring subdivision.

3.2.1. Determining mass parameter

After subdivision, masses for new nodes need to be determined. Since we want the user to feel that he is interacting with the same object, the total mass of the subdivided area should be the same as before. If this area has N nodes before subdivision and N' nodes after subdivision and each node has a mass value of m before subdivision, then the mass value m' for each new node in this area is

$$m' = m\frac{N}{N'} = mN_r \tag{3}$$

This new mass value is used in deforming the new mesh.

3.2.2. Guideline to determine spring constants

Recall we denote home spring constants as K_h and mesh spring constants as K_m. Before further subdivision, we need to determine the new home spring constant K'_h and new mesh spring constant K'_m. If we keep the spring constants unchanged after subdivision, this area will be stiffer than its counterpart on the original mesh since there

are more springs preventing the surface from deforming. Therefore we need to find values for K'_h and K'_m to keep the overall deformation and contact force consistent before and after subdivision.

In this paper we propose a method as a guideline for finding values for K'_h and K'_m. This method is based on the hypothesis that the mesh should have the same overall deformation and contact force before and after subdivision: if force F is needed to displace a node of the original mesh surface to a distance x, then with K'_h and K'_m defined on subdivided area around that contact node, the same force F should displace the type 1 node corresponding to the original contact node to the same distance x. This idea can be expressed as

$$F'(K'_h, K'_m, x) = F(K_h, K_m, x) \tag{4}$$

Our original mesh is a regular equilateral triangular mesh. If one-ring subdivision is applied on the mesh shown around node C in Figure 5 (a), the new mesh will be like that illustrated in Figure 5 (b). New edges are shown in thin lines in Figure 5 (b), among which are edges added to prevent cracks (shown in gray). Two-ring subdivision is carried out similarly.

(a)   (b)
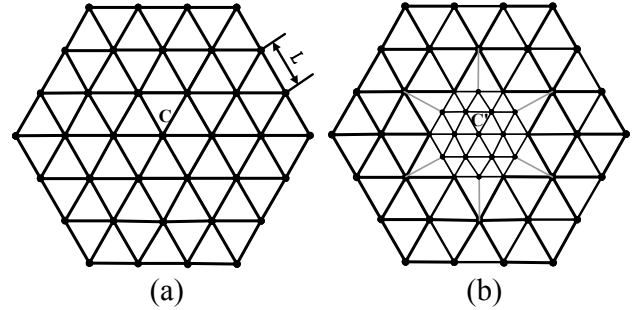
Figure 5. One-ring subdivision of a coarse mesh

force magnitude (N)

Kh=Km=0.06

x/L
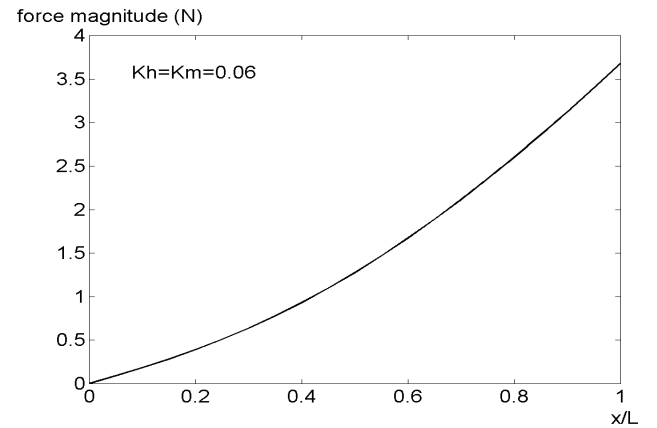
Figure 6. Force magnitude vs. displacement of node C. x: displacement; L: initial element size.

To find values of K'_h and K'_m in equation (4), K_h, K_m and x must be measured. Therefore we displace the

central node C on the coarse mesh to various distances, and calculate the total reaction force at C, namely F_c. We assume in our analysis that displacement of any of nodes on our mesh to be no more than the element size L of the original mesh. For this experiment, L is initialized to 30mm and K_h and K_m to 0.06 N/mm. Figure 6 is the resulting plot of F_c as a function of the normalized displacement (x/L) before the mesh is subdivided.

3.2.3. An initial approach for determining spring constants

One simple approach to finding K_h' and K_m' is to make the force for a given displacement the same after subdivision as before. Thus we must reduce the home spring constant when the number of nodes increases after subdivision. Let us denote the area to subdivide as A; if there are N and N' nodes in A before and after subdivision, the relation between K_h and K_h' can be written as:

$$K_h' = K_h \frac{N}{N'} = K_h N_r \qquad (5)$$

Now we need to find appropriate values for K_m'. At each displacement, after subdivision we adjust K_m' until the force matches the magnitude on Figure 6 corresponding to that displacement (upper plot in Figure 7). In this figure we also recorded the value of K_m' when K_h' remains unchanged after subdivision (lower plot in Figure 7). Plots in Figure 7 are obtained using one-ring subdivision.
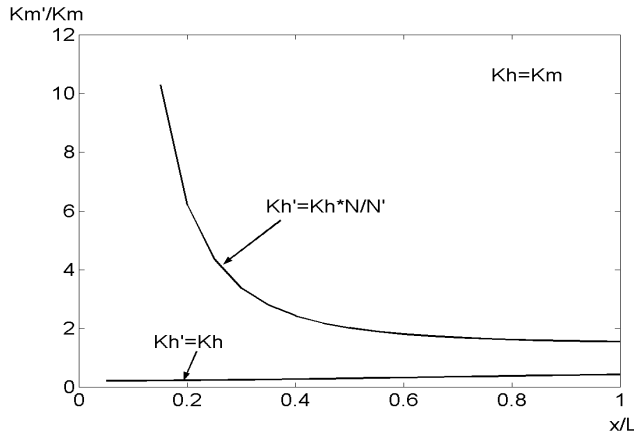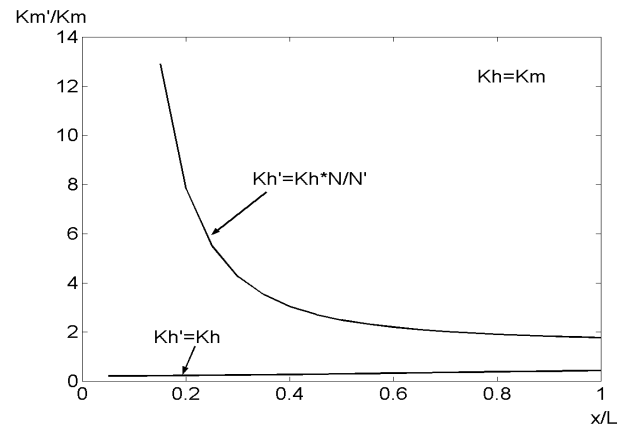

Figure 7. K_m' / K_m vs. displacement of node C (one-ring subdivision). x: displacement; L: initial element size.

In Figure 7, as displacement approaches zero, the upper plot asymptotically approaches infinity. This is due to the dominance of home springs at very small displacements, where even a large change in mesh spring constant cannot compensate for the change of home

spring constant. Because of this numerical instability equation (5) is not a desirable solution to finding K_h' and K_m'.

The lower plot shows an approximately linear relationship. This relationship does not cause instability to the mesh after subdivision. It also shows that K_m' is smaller than K_m, which is reasonable, i.e. to reach the same stiffness, a denser mesh requires smaller spring constants. However, since the subdivided area has more home springs with the same spring constant than before subdivision, this part of the object will be stiffer. We want the user to feel the same force before and after subdivision, therefore this relationship also has its drawbacks.

Figure 8. K_m' / K_m vs. displacement of node C (two-ring subdivision). x: displacement; L: initial element size.

The counterparts of the plots in Figure 7 with two-ring subdivision are drawn in Figure 8 and show similar properties to those in Figure 7. Thus the number of rings subdivided does not affect the general form of the plots.

Because of the instability and poor physical property preservation ability that the plots in Figure 7 and Figure 8 indicate, we need a better approach for determining spring constants.

### 3.2.4. An improved approach for determining spring constants

We have two unknowns, K_h' and K_m', but equation (4) only offers one constraint. Another constraint is needed. One method is to fix the ratio of K_m' / K_h'. Given this ratio, we adjust both K_m' and K_h' until the force magnitude at a displacement after subdivision matches the corresponding point on Figure 6.

Figure 9 shows plots of K_h' / K_h with $K_m' / K_h' = 4$, 1 and 0.25. The plots of K_m' / K_m with the same ratios of K_m' / K_h' are shown in Figure 10. Figure 9 and Figure 10

assume one-ring subdivision; Figure 11 and Figure 12 are two-ring counterparts.
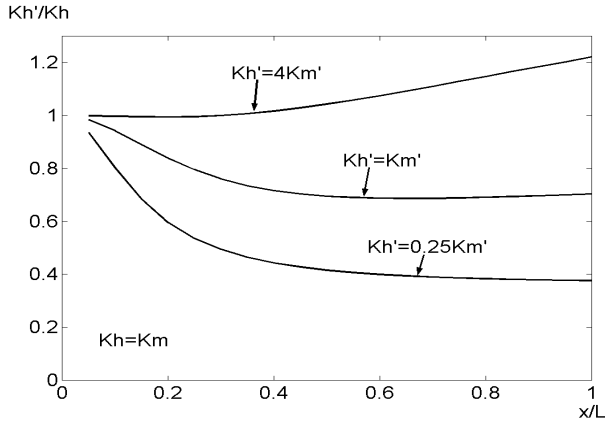
Figure 9. K'_h / K_h **vs. Displacement of node C (one-ring subdivision). x: displacement; L: initial element size.**
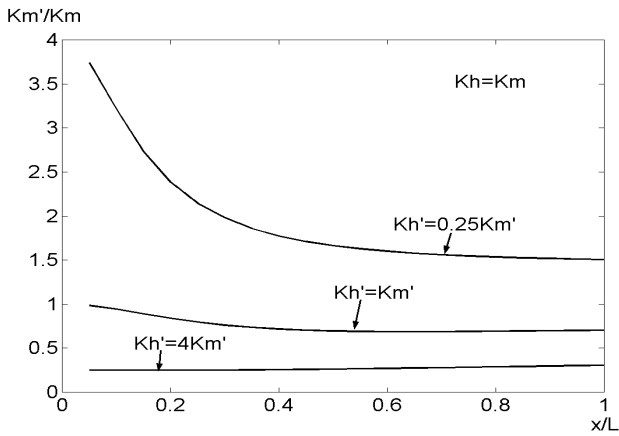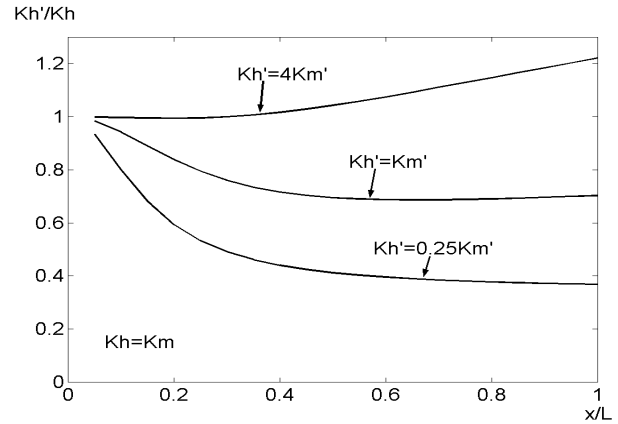

Figure 10. K'_m / K_m **vs. Displacement of node C (one-ring subdivision). x: displacement; L: initial element size.**

computation power is needed. One should choose the number of rings best suitable to the application.

Figure 11. K'_h / K_h **vs. Displacement of node C (two-ring subdivision). x: displacement; L: initial element size.**

Figure 12. K'_m / K_m **vs. Displacement of node C (two-ring subdivision). x: displacement; L: initial element size.**
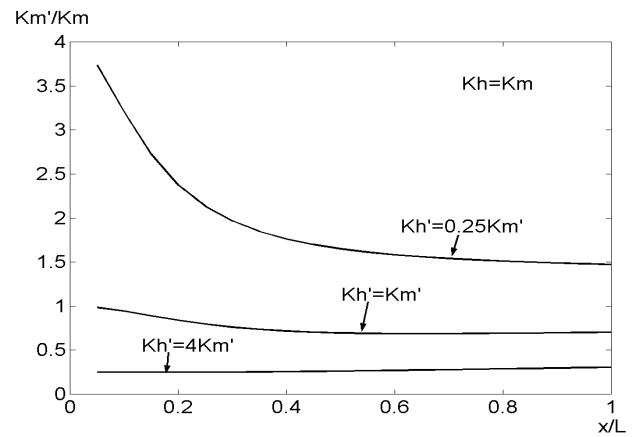
In Figure 9 and 11, all the plots start from a ratio of K'_h / K_h close to 1 when the displacement is small. This implies that at small displacements home spring constant should have the same value as that before subdivision. When K'_h has a ratio of 4 over K'_m, the plot shows the least variation. This is also true in Figure 12 (actually they are the same plot drawn in different ordinates). Although the $K'_h = 0.25K'_m$ plot shows the largest variation in value, it is still within the value range where the mesh stays stable. This fact proves that keeping the K'_m / K'_h ratio constant is a more stable method than that proposed by equation (5). Comparing the plots in Figure 7 and Figure 9, we see this stability comes from the fact that K'_h is allowed to change as displacement of the contact node varies.

Comparing Figure 11 and Figure 12 with Figure 9 and Figure 10 we can conclude that the number of rings subdivided does not change the characteristics of the plots. The more rings involved in subdivision, the more

The number of rings does not change the nature of the shape of the plots in Figure 7 through Figure 12. However, one may wonder if the ratio of K_m / K_h plays a role in shaping the plots. In another experiment we set K_m / K_h to be 10 and 0.1, respectively, which are significantly different from the ratio 1.0 used to obtain the above plots, and then collected data that corresponds to Figure 6 through Figure 12 with these ratios. The data show the similar tendencies to those in Figure 6 through Figure 12. Therefore the initial ratio of K_m / K_h does not affect the basic relationship between K'_m and K_m, or that between K'_h and K_h.

Since different ratios of K'_m / K'_h can be used to meet the requirement stated by equation (4), as illustrated in from Figure 9 to Figure 12, which ratio to choose? Let's first consider what impact K'_m / K'_h can have on the shape of deformation of the mesh. If K'_m is extremely large and

K'_h approaches zero, the mesh would act like a rigid shell. This is the case of an infinite large ratio of K'_m / K'_h . On the other hand, if K'_h is extremely large and K'_m approaches zero, whenever the user is displacing a node, no surrounding nodes would be displaced because they are "fixed" in their initial position by the extremely stiff home springs. Therefore the ratio between mesh spring constant and home spring constant determines the shape of deformation. To keep the shape of deformation consistent before and after deformation, it is thus important to keep the ratio of mesh spring constant over home spring constant unchanged. In the experiment we carried in obtaining Figure 9 to Figure 12, the ratio of K_m / K_h is 1 before subdivision, therefore after subdivision the plots corresponding to $K'_h = K'_m$ in Figure 9 and Figure 11 should be used for new values of K'_m and K'_h, although these plots are not the plots with least variation in values.

The plots of $K'_h = K'_m$ in Figure 9 and Figure 11 show a function of displacement of the contacted node. This indicates haptic subdivision requires variable spring constants dependent on contact node displacements. To apply variable spring constants on the subdivided mesh, a look-up table can be developed which contains values of K'_h and K'_m for discrete displacements. The spring constant values at current displacement of contact node can then be obtained by interpolating between entries of the look-up table.

3.3. Summary of haptic subdivision

In short haptic subdivision involves three steps:
1) Subdivide the haptic mesh, keep a constant ratio between mesh spring constant and home spring constant, obtain a value curve like the middle one in Figure 11 and save the values in a table. This is the preprocessing stage.
2) At run time, subdivide the haptic mesh around the contacted node with a standard geometric subdivision scheme and construct a new mesh.
3) Apply new spring constants to the new mesh and deform.

Figure 13 shows haptic subdivision on a triangulated hemisphere. Figure 13 (a) is the original mesh when deformed; (b) is the mesh deformed with one-ring subdivision; (c) is the mesh deformed with two-ring subdivision. Note the more accurate and realistic appearance of (c) over (a).

4. Conclusions and discussions

In this paper we propose a novel approach which combines physically based modeling with surface subdivision techniques to achieve deformation simulation with levels of detail. Although much research has been done on geometrically subdividing a surface in computer graphics literature, how to determine the physical parameters of the mesh has not been addressed yet. In our study we designed experiments to find good values for spring constants with equation (4) as a guideline. The procedures of haptic subdivision as a whole are outlined at the end of section 3. These procedures can be integrated into existing interactive deformable object simulation to achieve better realism.

There are many aspects worth further improvement in our work. First, our mass-spring model may be extended to be multi-layer with nonlinear springs to simulate complex deformation such as that of human tissues. Secondly, haptic subdivision on a hybrid finite-element-mass-spring model may offer more accurate simulation with levels of detail. Thirdly, point contact in our implementation can be extended to area contact to accommodate a broader range of applications.
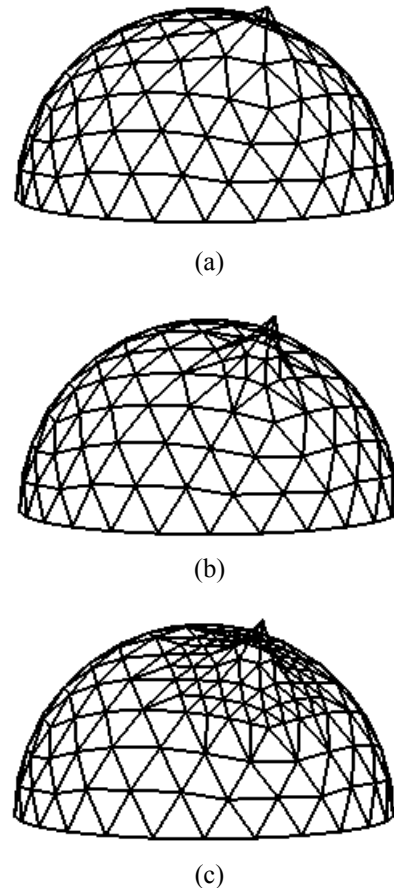

(a)

(b)

(c)

Figure 13: Haptic subdivision applied on a hemisphere. (a): the deformed original mesh. (b): one-ring subdivision applied around the contact point. (c) two-ring subdivision applied around the contact point. Note the improved appearance of (c) over (a).

References

[1] G. S. P. Miller, "The Motion Dynamics of Snakes and Worms", *Computer Graphics*, Volume 22, Number 4, 1988, pp 169-178.

[2] J. E. Chadwick, "Layered Construction for Deformable Animated Characters", *Computer Graphics*, Volume 23, Number 3, 1989, pp 243-252.

[3] C. Loop, "Smooth Subdivision Surfaces Based on Triangles", *Master's thesis*, University of Utah, Department of Mathematics, 1987.

[4] T. Dang, M. A. Srinivasan, and T. M. Annaswamy, "Development and Evaluation of an Epidural Injection Simulator with Force Feedback for Medical Training", *Proceedings of Medicine Meets Virtual Reality*, 2001, pp. 97-102.

[5] L. L. H. Holton, "Force Models for Needle Insertion Created from Measured Needle Puncture Data", *Proceedings of Medicine Meets Virtual Reality*, 2001, pp. 180-186.

[6] E. Keeve, P. Pfeifle, B. Girod, and S. Girod, "Anatomy-based Facial Tissue Modeling Using the Finite Element Method", *Proceedings of the Conference on Visualization*, 1996, pp 21-28.

[7] F. Tendick, M. Hegarty and L. W. Way, "A Virtual Environment Testbed for Training Laparoscopic Surgical Skills", *Presence*, Vol. 9, No. 3, June 2000, pp. 236-255.

[8] D. d'Aulignac, C. Laugier and M. C. Cavusoglu, "Towards a realistic echographic simulator with force feedback", *Proceedings of the 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 727-732.

[9] D. d'Aulignac, R. Balaniuk and C. Laugier, "A Haptic Interface for a Virtual Exam of the Huamn Thigh", *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, San Francisco, April 2000, pp. 2452-2457.

[10] I. Brouwer, J. Ustin, L. Bentley, A. Sherman, N. Dhruv, and F. Tendick, "Measuring In Vivo Animal Soft Tissue Properties for Haptic Modeling in Surgical Simulation", *Proceedings of Medicine Meets Virtual Reality* 2001, IOS Press, Amsterdam, Jan. 2001, pp. 69-74.

[11] M. C. Cavusoglu and F. Tendick, "Multirate Simulation for High Fidelity Haptic Interaction with Deformable Objects in Virtual Environments", *Proceedings of 2000 IEEE International Conference on Robotics & Automation*, San Francisco, April 2000, pp. 2458-2465.

[12] C. Mandal, H. Qin and B. C. Vemuri, "A Novel FEM-Based Dynamic Framework For Subdivision Surfaces", *Computer Aided* Design (Special issue on solid modeling), 32(8&9), 2000, pp. 479-497.

[13] T. W. Sederberg and S. R Parry, "Free-Form Deformation of Solid Geometric Models", *Proceedings of the 13th Annual Conference on Computer Graphics*, Dallas, TX USA, August 18 - 22, 1986, pp. 151 - 160.

[14] S. De, J. Kim and M. A. Srinivasan, "A Meshless Numerical Technique for Physically Based Real Time Medical Simulations", *Proceedings of Medicine Meets Virtual Reality*, 2001, IOS Press, Amsterdam, Jan. 2001, pp. 113-118.

[15] M. C. Cavusoglu and F. Tendick, "Multirate Simulation for High Fidelity Haptic Interaction with Deformable Objects in Virtual Environments", *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, April 2000, pp. 2458-2465.

[16] Y. Zhuang and J. Canny, "Haptic Interaction with Global Deformations", *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, pp. 2428-2433.

[17] D. L. James and D. K. Pai, "ArtDefo: Accurate Real Time Deformable Objects", *Proceedings of the SIGGRAPH 1999 Annual Conference on Computer graphics*, Los Angeles, CA USA, August 8 - 13, 1999, pp. 65-72.

[18] R. M. Koch, M. H. Gross, F. R. Carls, D. F. von Büren, G. Frankhauser, and Y. I. H. Parish, "Simulating Facial Surgery Using Finite Element Models", *Proceeding of the 23rd Annual Conference on Computer Graphics*, New Orleans, LA USA, August 4-9, 1996, pp 421-428.

[19] D. Braraff and A. Witkin, "Large Steps in Cloth Simulation", *Proceedings of the 25th Annual Conference on Computer Graphics*, Orlando, FL USA, July 19-24, 1998, pp. 43-54.

[20] S. Cotin, H. Delingette and N. Ayache, "Real Time Elastic Deformations of Soft Tissues for Surgery Simulation", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, No. 1, January-March, 1999, pp. 62-73.

[21] M. A. Sagar, D. Bullivant, G. D. Mallinson, P. J. Hunter and I. W. Hunter, "A Virtual Environment and Model of the Eye for Surgical Simulation", *Proceedings of the 21st Annual Conference on Computer Graphics*, Orlando, FL USA, July 24-29, 1994, pp 205-212.

[22] "The PHANToM Hardware Interface", http://www.sensable.com/haptics/products.html

[23] D. Doo and M. Sabin, "Analysis of the Behavior of Recursive Division Surfaces near Extraordinary Points", *Computer Aided Design,* Volume 10, Number 6 (1978), pp 356-360.

[24] E. Catmull and J. Clark*,* "Recursively Generated B-Spline Surfaces on Arbitrary Topological Meshes", *Computer Aided Design*, Volume 10, Number 6, 1978, pp 350-355.

[25] N. Dyn, D. Levin, and J. A. Gregory, "A Butterfly Subdivision Scheme for Surface Interpolation with Tension Control", *ACM Transaction on Graphics*, Volume 9, Number 2, April 1990, pp 160-169.

[26] D. Zorin, P. Schröder and Wim Sweldens, "Interpolating Subdivision for Meshes with Arbitrary Topology", *Proceedings of the 23rd Annual Conference on Computer Graphics*, New Orleans, LA USA, August 4-9, pp189-192.

[27] L. Kobbelt, "Interpolatory Subdivision on Open Quadrilateral Nets with Arbitrary Topology", *Proceedings of Eurographics 96*, Computer Graphics Forum, 1996, pp 409-420.