Acceleration of Elastic Model's Motion Computation Based on Elastic Element Reduction

S. Miyazaki, J Hasegawa
Chukyo University
101 Tokodachi, Kaizu-cho, Toyota-city 470-0393, Japan

T. Yasuda, S. Yokoi
Nagoya University
Furo-cho, Chikusa-ku, Nagoya-city 464-8601, Japan

Abstract

This paper proposes a fast computation elastic model. It was constructed by a small number of elements. Elastic objects are constructed in various sizes of elements. Small elements are laid out on the surface and larger elements are confined to the center. A variety of element shapes and the way they fit into any voxel-based shape are presented here. It effectively reduces the number of elements, and saves computation time. This elastic element model is directed to the duties of real-time processing, performing consistent restoration, and being applicable to any shape of polyhedron elements.

Keywords: elastic object, deformable model, element reduction, acceleration, real time, computer graphics

1. Introduction

Physically-based modeling is an effective way to generate natural motion of deformable objects by defining a comparatively simple model that represents the local property of the object [1-3]. It has been developed in the field of computer animation for the purpose of generating real motion of deformation as an animation. Recently, its possibility is increasing to extend its real-time applications with high-resolution object models, thanks to great advances in computation performance. However, even with great advances in computation performance, if properties are limited to linear elasticity, high-resolution models are still hungry for computation power. It is necessary to construct objects with fewer elements.

Another function is to construct any shape of objects. Polygon-based models are popular ones in the field of real-time computer graphics. It can flexibly represent any shape with fewer elements. In general, the shape of elements should be equivalent in length in every direction for performing smooth deformation. It should be close to regular polyhedrons. Voxel-based models are another popular one which is often converted into polygon models for fast processing, if the object is non-deformable. However, it is suitable as deformable models, for it consists of the same size of cube. Conversions between polygon models and voxel ones have been studied for a long time and aren't complicated. Voxel models constructed from CT images are generally used in the field of medical imaging, one major field of study. Therefore, for now, the input shape is limited to a voxel model.

Constructing simple models suitable to its purpose is a necessity in real-time processing. For example, Promayon et al's model supposes elastic objects like a ball, in which an elastic surface like rubber surrounds air [4]. It consists of mass-and-spring lattice on the surface. In addition, the internal volume is constrained to be constant. As a result, this works to push every surface point to the outer sides. It really acts well with little computation. However, our interest is in the construction of general use and homogeneous elastic models rather than elastic models in which rubber-like models are arranged only on the surface. Then, in such models, elastic elements should be arranged in whole not only on the surface. Our study attempts the acceleration of computation in elements by employing variable sizes and reducing the number of elements.

2. Elastic Element

2.1 Overview

We also propose an original element model [5]. Many kinds of deformable models have been reported. Under the conditions of real-time processing, their foundations are classified into either mass-and-spring model or simplex element in Finite Element Method (FEM). Our model's criterion is different from those, but is derived from them. Restoration force is proportional to the displacements of vertices of the element from the reference position that gives its equilibrium shape. It is expanded from mass-and-spring models. In reference to displacement, its position is not given as a unit of length but as a unit of shape, this is common in FEM elements. These three models almost all similarly define the same force, when object deformation is small. However, our model does not have such serious problems like the other two models have, when object deformation is large.

2.2 Problems in Mass-and-Spring Models

Mass-and-spring model has an advantage that the implementation is easy, but the restoration force becomes more improper as the mass-and-spring lattice deforms. Especially, when the object is heavily compressed, almost all springs become parallel to each other. As a result, restoration force is depleted by competition among springs (Figure 1).

Promayon et al's model modifies this problem. They choose some neighboring vertices to define a base plane in elements and decide the force of some other neighboring vertices to keep location relative to the plane. This works well in their use of constructing elastic surfaces. Base planes should be laid along the object surface. However, applying the model in general to any shape element is not recommended. Deciding which vertices make the base plane is complicated.

2.3 Problems in FEM Elements

FEM element's advantage is that physical properties of real material are directly reflected to the model. It works well in such cases when its usage is limited to static behavior in stable layouts. However, in general use, problems occur during the object's rotation around itself. If the reference shape is stable at all times, as rigid rotation deviates from the reference, force error rapidly becomes as large as the rigid

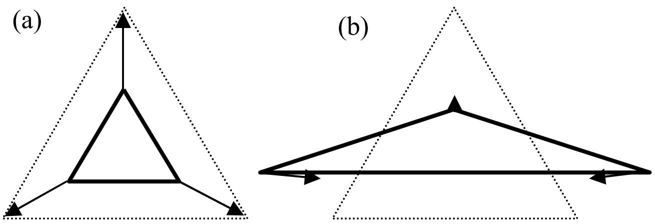

Figure 1 For mass-and-spring models, the bold lines represent the deformed state, the dotted lines represent the reference shape, and the arrows represent restoration force. In (b), you can see restoration force depleted in the vertical direction.

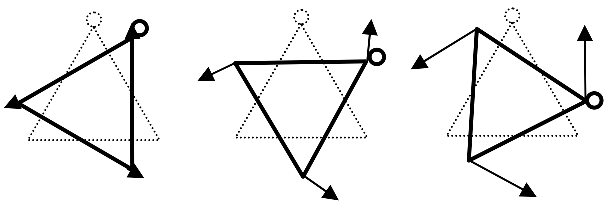

Figure 2 For simplex elements in FEM, unexpected force appears as large as its element rotation. Inadequacies have to be overcome in order to apply this to object rotating cases.

rotation (Figure 2). This theory exists only under small deformative objects. This problem will be solved by finding the ‘axis of principal stress’ and also by employing dynamic reference shapes which can chase deformed elements like ones being used in our model. Even if this theory is considered, our model is superior in shape flexibility as shown in 2.4.

2.4 Shape Flexibility

Our model permits any shape of elements. It is a great privilege constructing the proposed elastic model. In FEM models, elements must be subdivided into the base shapes of triangle or tetra. Mass-and-spring model’s problem, is how improper layout springs keep truss structure. The resultant force is often inadequate even if the layout is completed, because of competition among springs.

2.5 Stability

Elastic elements sometimes strongly deform by itself, especially when too strong a force acts upon it. It may cause an inconsistent state, in which an element turns itself inside out. Our model works to restore inconsistent states back to consistency, in any case. It is an unavoidable necessity for other models to detect inconsistent states and to cope with them. Although, this is not so easy, as computation time increases.

2.6 Implementation

This subsection easily explains our model. It decides elastic force only depending on a simple assumption. Elastic force is proportional to displacements of vertices; they are not ones in springs but ones from reference shapes. The reference shape’s location is decided in each element as is; both the resultant internal force and the moment of force of the internal stresses are equal to zero vectors.

$$\sum_i \boldsymbol{r}_i \times k\,(\boldsymbol{R}_i - \boldsymbol{r}_i) = k \sum_i \boldsymbol{r}_i \times \boldsymbol{R}_i = \vec{0} \qquad (1)$$

Here, vector $\boldsymbol{R}_i$ and $\boldsymbol{r}_i$ are relative positions of vertex i in the reference shape and the

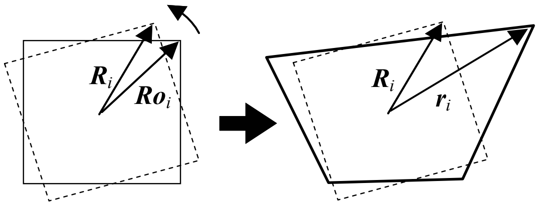

Figure 3 Position vectors have respect to the center of gravity, in the original reference shape, rotated reference shape, and in the deformed shape.

deformed shape with respect to the center of gravity, respectively (Figure 3). Force is proportional to displacements with proportionality constant k.
In the implementation, the initial values of { $\boldsymbol{R}_i$ } are registered as { $\boldsymbol{Ro}_i$ }, and { $\boldsymbol{R}_i$ } is defined as a result of the rotation of { $\boldsymbol{Ro}_i$ } around the center of gravity.

$$\sum_i \boldsymbol{r}_i \times M\,\boldsymbol{Ro}_i = \vec{0} \tag{2}$$

When deformed shape { $\boldsymbol{r}_i$ } and the initial reference shape { $\boldsymbol{Ro}_i$ } are given, rotation matrix M and the reference shape { $\boldsymbol{R}_i$ } are sequentially decided. It specifies a unique reference shape location. It means removing the rigid rotation component in deformation before deciding displacements. Ref [5] presents a fast computation method of Equation (2) for 3D models. This procedure has little complications, but as a result, its computation time is almost the same as mass-and-spring models.

It works equally as FEM elements when shapes are base ones, for exact compatibility, reference shapes should be distorted in Poison's ratio. More vertex shapes should be subdivided into base shapes, in every base shape, to evaluate distortions in more detail. Our model regards distortion as being constant in polygons. Preciseness is lost in deciding force, but it saves computation time.

3. Gradational Resolution Model

Considering that element reduction procedure is performed by replacing a few elements with one element that fills their domestic spaces. If this procedure can be applied only to elements whose deformations are comparatively small, this might be an effective element reduction. However, including procedures to search such elements spoils the acceleration of computation, because its computation time is

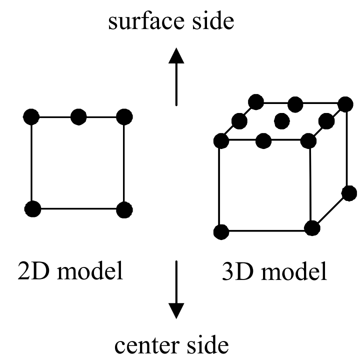

Figure 4 Elements in gradational resolution models.

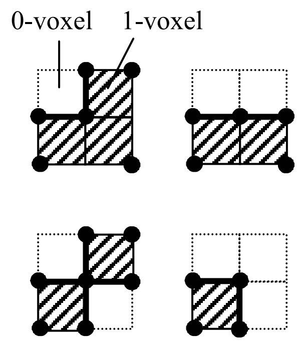

Figure 5 Variations in vertex layouts. Bold boundaries represent voxel object surfaces.

proportional to the number of elements. It is difficult to predict deformation tendencies beforehand, but it can be said that deformation is large around contact points with rigid objects. External force from rigid objects is far larger than internal force in elastic objects, and it works directly on the surface of elastic objects. Element's sharing vertices also constrain neighboring elements' deformation. This works less on the surface where fewer neighboring elements are joined.

Based on the above considerations, here we propose a model whose element sizes can be gradationally increased from the surface to the center. It effectively saves computation time and maintains voxel's original resolution on the surface.

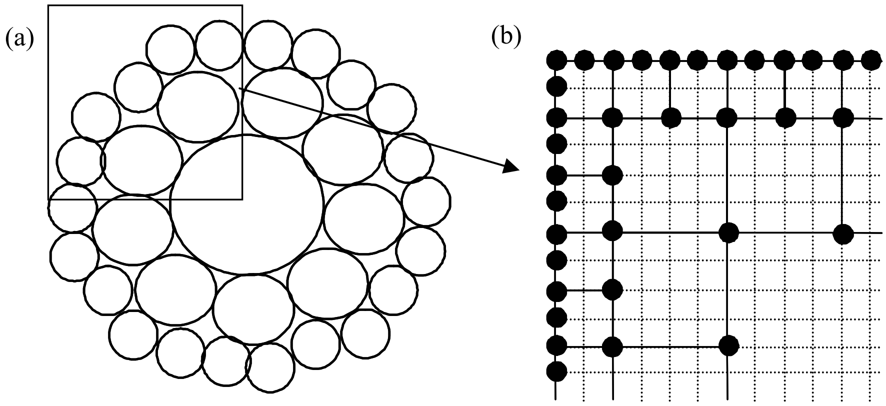

Figure 6 The image illustration (a), and the concrete image illustration (b) of a gradational resolution model. (b) is a magnified illustration of a corner of a cubic object.

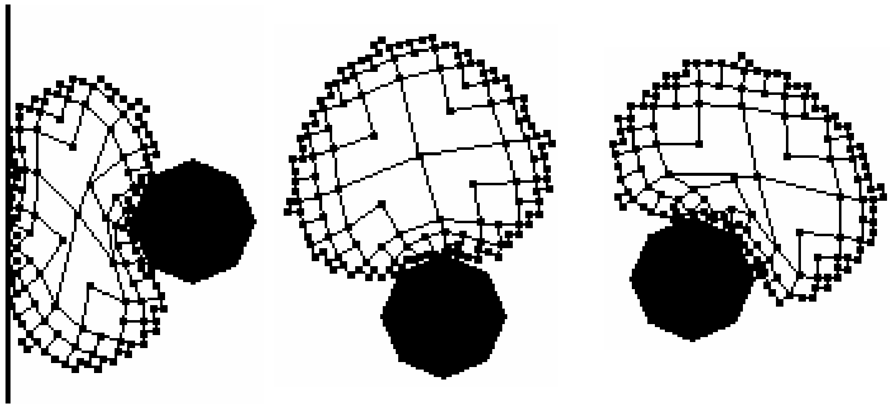

Figure 7 Implementation of manipulation environments. An elastic object can be manipulated by a rigid circle manipulator which is controlled by mouse.

Table 1 Real computation time and maximum element number available for real-time processing.

CPU	Pentium4 (1.9G)	AthlonXP (1900+)
computation time per vertex(μs)	7.06	4.22
max element number (k =10000)	68	114

3.1 Implementation

Vertex density in elements should be higher on the surface side than in the center, like one shown in Figure 4. Followings simply explain construction procedure with this element model in a two dimensional case. In the first step, voxel data is scanned in every neighboring 2x2-voxels independently to find surface 2x2-voxel ones, which includes boundaries of voxel object surface. Figure 5 shows how to layout vertices on them, that has variation in the number of 1-voxels. Then, all of the surface unit's 1-voxels are changed into 0-voxels. This procedure simultaneously prepares us for the next step. After this procedure, 2x2-voxel-units consist of either four 0-voxels or four 1-voxels. Then, procedure is repeated and 2x2 units become one voxel in the next step. These procedures resulted in such a model shown in Figure 6. Figure 7 is an example of a manipulation environment.

3.2 Performance of element reduction

The number of elements increases proportionally to the third power of object resolution. Computation's size is roughly proportional to the number of nodes. A n by n by n cubic voxel object has 8 n^3 nodes. They are reduced into 26 n^2 by the gradational resolution model, which has the same resolution as the surface. In three dimensional cases, a sphere-like object, which has 552 voxel elements before reduction, can fit mostly into a 10x10x10 cubic volume. It was reduced into 74 elements in the experiment. Table 1 shows real computation time running at PCs highest performance. It can process about 100 element 3D objects in real time at this current performance speed (k = 10000).

3.3 Advantage of discretization

The Euler method is a standard solution of motion computation, for it sequences dynamic motions, step by step. For fast computation, time steps should be as large as possible with consideration to the cycle of simple harmonic motion, performed by elastic elements. However, it delays internal force propagations in objects proportional to the number of medium elements. Our element reduction decreases this problem.

4. Conclusions

We proposed a fast computation elastic model constructed by a small number of elements. Elastic objects were constructed in various sizes of elements. Small elements were laid out on the surface and larger ones were confined to the center. A variety of element shapes and the way they fit into any voxel-base shape were presented here. It effectively reduced the number of elements, and saved computation time.

The elements in Figure 4 make too rapid a gradation for high resolution models. So, degrees in gradation should be controlled depending on applications.

Acknowledgments

We would like to thank Professor Teruo Fukumura of Chukyo University for his encouragement and assistance. Special thanks to Marvin Lenoar, a great English teacher and for English assistance on this project. This research was supported in part by the grant-in-aid for Private University High-Tech Research Center from the Ministry of Education, Japan.

References

[1] Terzopoulos D. Platt J. Barr A. Fleisher K. "Elastically Deformable Models", Computer Graphics, 21(4), pp.205-214, 1987.

[2] Terzopoulos D. Fleisher K. "Modeling Inelastic Deformation: Viscoelasiticity, Plasticity, Fracture", Computer Graphics, 22(4), pp.269-278, 1988.

[3] Norton A. Turk G. Bacon B. Gerth J. Sweeney P. "Animation of Fracture by Physical Modeling", The Visual Computer, 7, pp.210-219, 1991.

[4] Promayon E. Baconnier P. Puech C. "Physically-based Deformations Constrained in Displacements and Volume", In Computer Graphic Forum, 15(3), pp. 155-164, Eurographics96, 1996.

[5] Miyazaki S. Hasegawa J. Yasuda T. Yokoi S. "A Deformable Object Model for Virtual Manipulation Based on Maintaining Local Shapes", SCI2001, Orland, 2001, (URL : http://www.sccs.chukyo-u.ac.jp/~miyazaki/)