A mass-spring model for real time deformable solids

2 authors:

Tzvetomir Vassilev
University of Ruse Angel Kanchev
42 PUBLICATIONS 374 CITATIONS

Bernhard Spanlang
Virtual Bodyworks SL
71 PUBLICATIONS 3,447 CITATIONS

A MASS-SPRING MODEL FOR REAL TIME DEFORMABLE SOLIDS

Tzvetomir Vassilev, Bernhard Spanlang

Abstract
This paper presents a mass-spring model for real-time simulation of volume preserving deformable solids. A new type of springs that show collective behaviour was developed called "support springs", which model the "matter" inside the object and make it preserve its volume without the need of explicit volume computations during the simulation as it is done in conventional methods. Comparing the volume during simulation with the initial volume of the deformable solid demonstrates the accuracy of our approach. Experiments on different geometry show the low computational complexity, which is linear to the number of triangles of the deformable solid. Interactions between supportive clothing and a deformable female upper torso are shown in a simulation at the end of the paper.

1.Introduction

For years physical modelling and animation of deformable objects has been a problem of interest in the computer graphics society. Some of the first steps were initiated by Terzopoulos et al. [14, 15]. Their team described elastically and plastically deformable models and used the finite element method and energy minimisation techniques borrowed from mechanical engineering.

Volume preservation is of interest in geometric modelling [2, 7, 8, 12, 13], virtual reality surgery simulations [3-5, 10] and entertainment industry [1, 9]. Geometric constraints are applied to free form deformations (FFD) as described in [13] to allow constant volume of deformed objects. Rappoport et al. [12] applied an iterative Lagrange multiplier method, called Uzawa based volume preservation, to constrain deformations to preserve the volume of an object modified by FFDs. Aubert and Bechmann [2] use a similar approach to [12], claiming to be more flexible by introducing an independent deformation function. They compute the exact volume of the triangular surface in a similar way as proposed in [8]. To allow handling of curved surface solids Hirota et al. [7] use a multi level of detail approach employing FFDs. Chadwick [3] applied FFDs to change the appearance of muscles and fatty tissue by changing the FFD control points according to a multi layered character skeleton. Hookean springs are added to automatically simulate stretch and squash of muscles and tissue. Volume preservation is not attempted here. Chen and Zeltzer [4] implement a finite element method (FEM) to create a complete biomechanical model of muscle action for cartoon character animation. This approach is very accurate but due to its computational complexity very slow. Promayon et al. [10] approximate surfaces of volumes by mass-points linked to their neighbours. Volume preservation is achieved by constraining the model to its volume calculated using an approach similar to [8]. Nedel and Thalmann [9] described a mass-spring system for modelling real-time muscle deformation. They presented the muscle shape as a surface based model fitted to the boundary of medical image data. In order to control the muscle volume during deformation a new type of springs was introduced called “angular springs”. The muscle deforms under the impact of external forces, but only when they are applied on a preliminary defined line called "action line", which represents the direction of the forces produced by the muscle on the bones. Aubel and Thalmann [1] extend [9] by introducing a new multi-layer model similar to [3]. "Action lines" are defined in a more general way using poly-lines.

The main objective of our work was to develop a real-time mass-spring model for simulating volume-preservation deformable objects. A mass-spring system was chosen because of its simplicity and low computational complexity. A new kind of springs was introduced called "support springs", which model the "matter" inside the object and make it preserve its volume.

2.Mass-spring system

A general mass-spring system consists of n mass points, each of them being linked to its neighbours by massless springs of natural length greater than zero. Let $\mathbf{p}_i(t)$, $\mathbf{v}_i(t)$, $\mathbf{a}_i(t)$, where i=1,…, n, be respectively the positions, velocities, and accelerations of the mass points at time t. The system is governed by the basic Newton's law $\mathbf{f}_i = m\,\mathbf{a}_i$, where m is the mass of each point and $\mathbf{f}_i$ is the sum of all forces applied at point $\mathbf{p}_i$. The force $\mathbf{f}_i$ can be divided in two categories. The **internal forces** are due to the tensions of the springs. The overall internal force applied at the point $\mathbf{p}_i$ is a result of the stiffness of all springs linking this point to its neighbours. The **external forces** can differ in nature depending on what type of simulation we wish to model. The most frequent ones are gravity and viscous damping.

The formulations make it possible to compute the force $\mathbf{f}_i\,(t)$ applied on point $\mathbf{p}_i$ at any time t. The fundamental equations of Newtonian dynamics can be integrated over time by a simple Euler method.

3.Support springs with explicit volume computation

The main feature of the traditional mass-spring system is that it consists of individual springs, i.e. the response of each spring depends only on its own elongation and not on the elongation of other springs. The idea of this work was to create an ensemble of springs where the response of each member depends on the state of the whole team. In particular the algorithm generates a response that will preserve the volume of a deformable solid object.

Let B be a body, whose surface is triangulated, as the ellipsoid shown in Figure 1. If all the vertices are connected to the body centre, then its volume can be computed as the sum of all tetrahedron volumes:

$$V = \frac{1}{3}\sum_{i=1}^{ntr} S_i h_i = \frac{1}{3}\sum_{i=1}^{ntr} S_i l_{ki}\cos\alpha_i = \frac{1}{6}\sum_{i=1}^{ntr} \mathbf{l}_{ki} \bullet (\mathbf{p}_{i1}\mathbf{p}_{i2} \wedge \mathbf{p}_{i1}\mathbf{p}_{i3})\,, \qquad (1)$$

where ntr is the number of all triangles, S_i is the area of the i-th triangle, h_i is the height of i-th tetrahedron, l_{ki} is the length one of its edges connecting the surface vertex to the body centre, α_i is the angle between the edge and h_i and $\mathbf{p}_{i1}$, $\mathbf{p}_{i2}$, $\mathbf{p}_{i3}$ are the i-th triangle vertices.

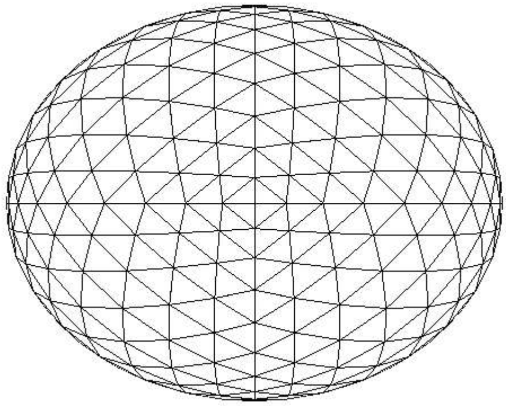

Figure 1. Ellipsoid with a triangulated surface

The deformable volume preservation body is constructed as follows. All surface vertices are connected to each other with regular springs, described in the previous chapter. These springs model the elastic membrane of the body and keep the triangles' surface approximately constant. In addition each vertex is connected to the centre with a spring. We call the collection of these springs support springs. They model the "filling" of the 3-D object. Let us make the following definitions. If l is the natural length of the i-th support spring and $l_i(t)$ is the length of the same spring at time t, and S(t) and $V(t)$ are the body surface and volume at time t, then the following lengths are defined:

$$\Delta l_{tot}(t) = (V(t) - V(0)) / S(t), \quad \Delta l_i(t) = l_i(t) - l_i^0 \tag{2}$$

The force acting on the surface vertex $\mathbf{p}_i$ at time t due to the i-th support spring is computed as

$$\mathbf{f}_i(t) = -K(\Delta l_{tot}(t) + C\Delta l_i(t))\,\mathbf{u}_i\,, \tag{3}$$

where K is the stiffness of the springs, $\mathbf{u}_i = (\mathbf{p}_i - \mathbf{c})/l_i$ is a unit vector, $\mathbf{c}$ is the body centre, and C is a coefficient in (0, 1), which can be varied. As evident from Equation 4, the response of a support spring is a result from two different behaviours. The first one is the collective reaction, i.e. each spring opposes to the change in the volume, trying to preserve it constant. The second addend gives the individual behaviour of the spring. The coefficient C controls the proportion between the collective and individual behaviour. It can be varied, depending on the type of simulation. The bigger its value is, the stiffer the object, which requires larger forces to deform it.

4.Support springs with implicit volume computation

The above-described approach has two major drawbacks. Firstly it is computationally expensive for objects with a large number of faces. According to equation 1 for each triangle one vector product and one scalar product need to be computed and the length of each support spring is needed according to equation 3. Secondly, our tests showed that the volume preservation accuracy and the simulation quality depend very much on the spring stiffness K in equation 3. In order to get pleasing results, this coefficient must be in a very narrow interval, which unfortunately depends on the magnitude of applied forces. Otherwise the volume preservation accuracy gets low and for some values of K the simulation even becomes unstable.

This is why an approximation of equation 1 has been derived. Let S_{AV} be the average triangle area. Dividing the two sides of equation 1 by S_{AV} and rearranging the sum on all support springs (tetrahedron edges) instead of all triangles one can get

$$V / S_{AV} = \frac{1}{3}\sum_{i=1}^{ntr} S_i / S_{AV} \cos\alpha_i l_{ki} = \frac{1}{3}\sum_{i=1}^{ns} l_i \sum_{j=1}^{ni} S_j / S_{AV} \cos\alpha_j \tag{4}$$

The sum $\sum_{j=1}^{ni} S_j / S_{AV} \cos\alpha_j$ indicates that the i-th support spring participates in the volume calculation of n_i tetrahedrons. There is freedom in selecting one of the three edges for each tetrahedron, one so can make sure that each edge is met at least once, i.e. $n_i>0$ for all $i=1, \ldots, ns$. If we omit the cosines and substitute $c_i = \sum_{j=1}^{ni} S_j / S_{AV}$ we derive the following equations

$$l_{tot}(t) = \frac{1}{3}\sum_{i=1}^{ns} c_i l_i(t), \qquad \Delta l_{tot}(t) = l_{tot}(t) - l_{tot}(0) \tag{5}$$

We consider that the coefficients c_i do not change significantly over time and can be regarded as constants. So the only thing that needs to be computed during simulation is the length of each support spring. The technique tries to preserve the value $l_{tot}(t)$ constant, which is the weighted total length of all support springs.

The force acting on the surface vertex $\mathbf{p}_i$ at time t due to the i-th support spring is computed as

$$\mathbf{f}_i(t) = -\, c_i K(\Delta l_{tot}(t) + C\Delta l_i(t))\, \mathbf{u}_i\ . \tag{6}$$

According to equation 6 the reaction force does not depend only on the changes of spring lengths but also on the coefficient c_i. This reflects the difference in the triangles face areas.

5.Volume preservation test

The method of support springs with implicit volume computation makes two approximations, that is why the following volume preservation tests were performed on a deformable ellipsoid and sphere. Forces were applied on top and bottom areas of the body. The external and internal forces reached their equilibrium and then the forces were released and the solid restored its original state.

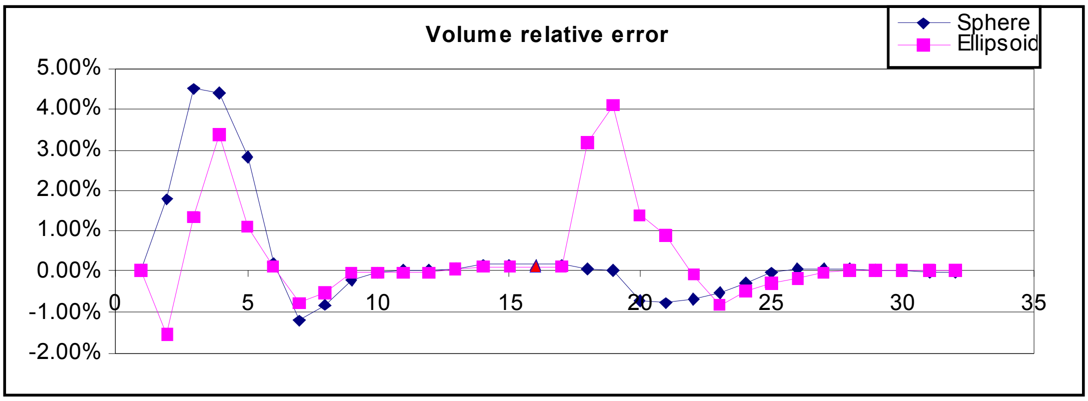

Figure 2. Results from the volume preservation test.

The results are shown in Figure 2. The relative error $e(t)=(V(t)-V(0))/V(0)$ was computed for 32 evenly distributed time points. The first 16 of them were during the deformation due to external forces. The simulation reached its equilibrium state in point 16 (marked with a triangle) and then the forces were removed. The highest errors were measured for the first several points after applying the external forces and the first several points after removing them. A logical explanation for this is that at those stages the speed of deformation is too high and the model cannot completely "catch up" with the fast changes.

6.Results

The algorithms were implemented on a Pentium III PC, 700 MHz, 128 MB, using TGS Open Inventor for rendering the images. Figure 3 shows two examples of a deformed ellipsoid and a hemisphere pressed against a table. The left image was generated for a tessellated sphere, scaled in the direction of the Y-axis, with 578 vertices and 1152 triangles. The entire simulation, including deforming to the equilibrium and then restoring the original shape, took 3.85 seconds, which required the generation of 180 frames. This means that the simulation ran at a speed of 46 frames per second, which is in real time.

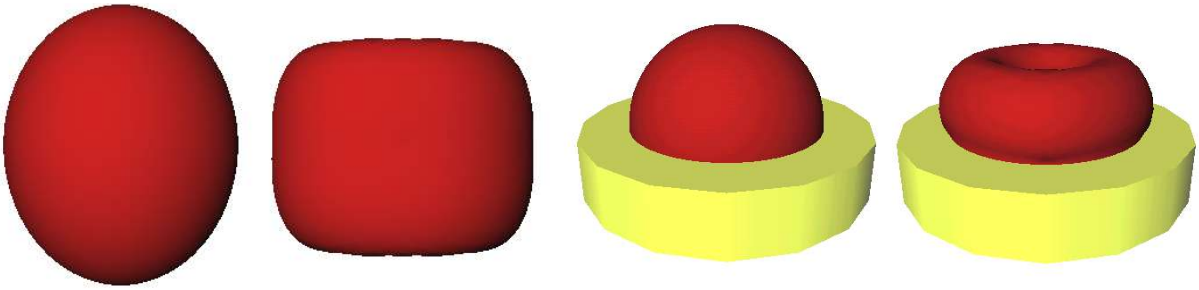

Figure 3. Deformed ellipsoid (left) and hemisphere on a table (right)

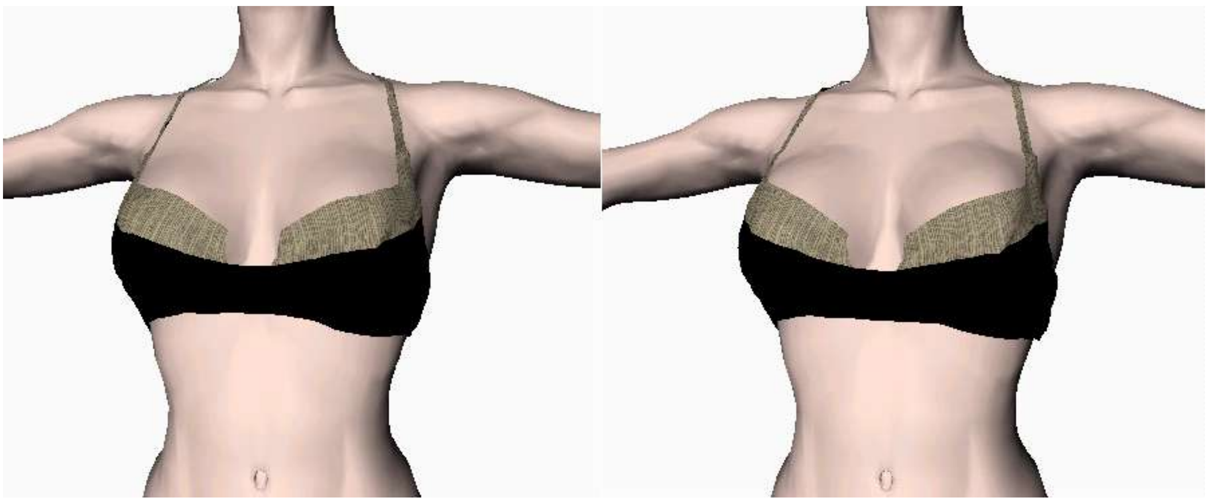

Figure 4. Bra on a virtual body; left: no deformation, right: breast deformation

The method for volume preserving deformation was implemented in a system for dressing virtual people [16] for simulating deformable human body parts. In the first version of the system the human body was considered as a rigid body and it was not deformed during the simulation. In order to enhance the realism in tight clothing as underwear and swimming suits, now the user can define deformable body parts as breast and belly, which deform during simulation under the forces of touching cloth. Figure 4 shows an example of dressing a virtual female body with a bra.

In order to check the algorithm complexity, we measured its speed for different numbers of triangles on the sphere. Results are given in Table 1. The times were measured using the profiling feature of Microsoft Visual C++ 6 and they are total times for computing the forces, integration of equations and rendering the image. As shown in the table the simulation still runs in real time for as many as 6728 triangles.

Number of Triangles	Time for 100 frames (s)	Time per frame (ms)	Frames per second
1152	2.13	21.30	46.9
2592	3.30	33.03	30.2
4608	4.99	49.94	20.0
6728	6.68	66.81	14.9

Table 1. Times for deforming a sphere with a different number of triangles

7.Conclusions

Mass-spring models for simulating deformable volume preservation objects have been defined. The techniques are applicable for objects presented with their triangulated surfaces. They use the so-called "support springs", which act as an ensemble and their response depends on the change in the object's volume as well as on each individual spring. The first method is based on explicit volume computations. Its volume preservation properties are better, but it is more computationally expensive and is not always stable for real-life problems. The second method assumes some approximations and in fact preserves the weighted total length of the support springs. As a result it is faster and more robust. The technique has a very good speed and it runs in real time (14 frames per second) for objects with as many as 7000 triangles.

8.References

[1] AUBEL, A., D. Thalmann. Realistic Deformation of Human Body Shapes. Computer Animation and Simulation, Interlaken 2000.

[2] AUBERT, F., D. Bechmann. Volume-preserving space deformation. Computers&Graphics 1997, 21(5), 625-639.

[3] CHADWICK, J. E., D. R. Haumann, R. E. Parent. Layered construction for deformable animated characters. Proceedings of SIGGRAPH 1989.

[4] CHEN, D. T., D. Zeltzer. Pump it up: Computer animation of a biomechanically based model of muscle using the finite element method. Proceedings of SIGGRAPH 1992.

[5] DEBUNNE, G., M. Desbrun, A. H. Barr, M.-P. Cani. Interactive multiresolution animation of deformable models, Eurographics Workshop on Computer Animation and Simulation 1999.

[6] DESBRUN, M., M. Meyer, P. Schröder, A. H. Barr. Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow. Proceedings of SIGGRAPH. 1999.

[7] HIROTA, G., R. Maheshwari, M. C. Lin. Fast volume-preserving free form deformation using multi-level optimization. ACM Symposium on Solid Modeling and Applications, ACM Press New York 1999.

[8] LIEN, S., J. Kajiya. A Symbolic Method for Calculating the Integral Properties of Arbitrary Nonconvex Polyhedra. IEEE CG&A 1984, 4(10), pp. 35-41.

[9] NEDEL L., D. Thalmann. Real time muscle deformation using mass-spring systems. CG Int. 1998, 156-165.

[10] PROMAYON, E., P. Baconnier, C. Puech. Physically-based model for simulating the human trunk respiration movements. Proceedings of Eurographics 1996.

[11] PROVOT, X. Deformation constraints in a mass-spring model to describe rigid cloth behaviour. Proceedings of Graphics Interface 1995, pp. 141-155.

[12] RAPPOPORT, A., A. Sheffer, M. Bercovier. Volume-Preserving Free-Form Solids. Solid Modeling, Salt Lake City 1995.

[13] SEDERBERG, T. W., S. R. Parry . Free-form deformation of solid geometric models. Proc. of SIGGRAPH 1986.

[14] TERZOPOULOS, D., J. Platt, A. Barr, K. Fleischer. Elastically Deformable Models. Computer Graphics (Proc. SIGGRAPH 1987), 21 (4), pp. 205-214.

[15] TERZOPOULOS, D., K. Fleischer. Deformable Models. Visual Computer 1988, 4, pp. 306-331.

[16] VASSILEV, T., Spanlang, B., Chrysanthou, Y. Fast Cloth Animation on Walking Avatars, Eurographics 2001.