

*Full Length Research Paper*

# Application of molecular modeling with mass-spring systems for computer simulation and animation

A. Sait Sekercioglu and A. Duysak\*

Computer Engineering Department, Dumlupinar University, Kutahya, Turkey

Accepted 3 August, 2009

**This paper demonstrates the use of potential functions of molecular modeling in mass-spring systems. In mass-spring systems, force acting on each mass point is determined by the spring length between linked mass-points. In molecular modeling, forces are calculated using bond length (spring length), bond angle and stretch-bond interactions. Molecular modeling therefore provides more accurate force function. Molecular modeling, however, works on mesh free structures, particles. In the developed model, potential calculations of particles are adapted to triangulated models as in mass-spring systems. The potentials of mass-points are then calculated using the connectivity of triangle mesh based on potential functions of molecular mechanics. Simulation results show that proposed method produces realistic simulations.**

**Key words:** Molecular modeling, potential functions, mass-spring systems, soft tissue modeling and simulation.

## INTRODUCTION

Steadily improving imaginary graphic applications became indispensable part of nowadays technology in which Computer Graphics' model deformation has significant role. Model deformation implies simulating physical behaviors of objects, which has an elastic and plastic structure. Modeling and simulation deformable objects have broad range of application areas such as character animation, simulating natural phenomenon and surgery simulators. There are two distinct approaches used in deformation simulations: non-physically based and physically based. Non-physically based approaches, e.g. free form, have their limitation in realistically representing their real world counterparts. The deformation characteristic of the object in these methods is not taken into account. Physically based methods include physical properties of the object and well known equations like Newton's force law.

There have been numerous physically based techniques developed in computer graphics (Nealen et al., 2005). Finite element method (FEM) and mass-spring systems (MSS) are the two popular physically-based methods. FEM models the mechanical properties (stress-strain relationship) and requires solution of partial different-

tial equations. FEM, therefore, is computationally expensive and is not always suitable for interactive applications. Researches have developed techniques to improve accuracy (Wu et al., 2001) and speed (DeBunne et al., 2001) of the FEM. A less accurate but faster method is MSS, in which a discrete model consists of mass-point connected by springs, is simulated using Hooke's Law (Baraff and Witkin, 1999). There are number of works reported on improving accuracy (Duysak and Zhang, 2003; Lloyd et al., 2007) and speed (Kang et al., 2000) of MSS. Another physically based method is known as particle systems.

Particle systems, which were used in early works for modeling fuzzy objects such as clouds, were adapted to be used in physically based animation of solids and liquids (Tonnesen, 1991). Particle systems work based on potential energies of each particle, which is the sum of the pair-wise potential energies between particles. Particle systems are very famous for molecular modeling in physics. Molecular modeling, also known as molecular mechanics, is a technique to calculate energy and geometry of molecules (Rapaport, 2004). In this technique, a force field, which describes functional form and parameter sets of potential energy of a system of particles (generally atoms), is used. A basic form of force field includes forces between bonded atoms and between non-bonded atoms.

\*Corresponding author. E-mail: [aduysak@mail.dumlupinar.edu.tr](mailto:aduysak@mail.dumlupinar.edu.tr)

A good example of use of molecular modeling in deformation simulation is reported in (Pithioux et al., 2005). They used Lennard-Jones potential formulation to calculate forces between atoms. Their work is a classical application of particle systems and can not be used with mass-spring systems. We developed a new scheme which combines particle systems and mass-spring system. We applied the method of molecular modeling to objects which is represented by triangles. The triangles have vertices connected by springs as in mass-spring systems. Each mass-point is treated as an atom and spring is bond between two atoms. Then, bond stretching and angle bending formulas of molecular modeling are used to calculate internal forces.

## MATERIALS AND METHODS

### Mass-spring systems

Mass-spring systems have been widely used in computer graphics applications because of their simplicity, speed and their acceptable accuracy. In mass-spring simulations the geometry of a deformable object is represented by a 3D mesh consisting of  $n$  nodes, which are interconnected by  $m$  links. Each node in the mesh represents a virtual mass and is called a mass-point. Displacements of these mass-points describe the deformation of the object. It is assumed that the springs are weightless. These springs define the distance relationships between mass-points because the deformation characteristic is embedded into the 3D mesh by spring parameters; namely stiffness and damping. A simple representation of such a system is shown in Figure 1.

In mass-spring systems, the mass-point forces consist of forces from the environment and forces generated by the springs themselves. The total force exerted on each mass-

$$f_{total} = f_k + f_d(k) + f_{ext} \quad (1)$$

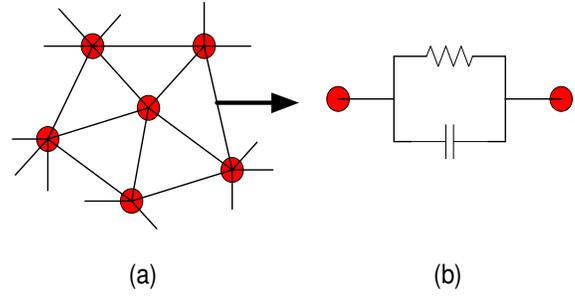
where  $f_{ext}$  represents external forces from environment and  $f_k$  and  $f_d$  are the spring forces due to spring stiffness and damping. Since the external force is assumed to be known (e.g. applied from user), the internal force generated between mass-points (atoms) is given as (Mesit et al., 2007),

$$f_{int} = f_k + f_d = \sum [k(r - r_0) - d(\|v_a - v_b\|)] \frac{x_b - x_a}{\|x_b - x_a\|} \quad (2)$$

Where;  $x_a$  and  $x_b$  are position vectors of mass-points,  $v_a$  and  $v_b$  are velocity vectors of the mass-points,  $r_0$  is the spring rest length (distance between mass-points) and  $k$  and  $d$  are the spring and viscosity (or damping) constants respectively. Equation 2 indicates that mass-spring systems use a linear function based on the distance of the mass-points. Angle information of bonded (linked) mass-points is not included in this function.

### Potential energies from molecular modeling for triangles

In molecular modeling, the potential of each atom is calculated



**Figure 1.** (a) a mesh structure consists of mass-points (atoms) and springs (bonds) is used to model deformable object and (b) ideal spring whose stiffness and damping are represented by a resistor and capacitor respectively

based on atomic interaction between atoms that are linked by covalent bonds and between non-bonded (non-covalent). Covalent bonds in molecular modeling are similar to springs in mass-spring systems. Similarly, atoms in molecular modeling correspond to mass-points in mass-spring systems. Molecular modeling or particle systems are designed based on mesh free dynamics (Tonnesen, 1991). Mass-spring systems however work on triangulated models. Most of the modeling and simulation application use triangulated models.

In this work, instead of using mesh free representation, we assume that deformable object consists of particles (atoms) that are connected by springs (bonds) forming triangles. Internal forces in this model occur when a particle (an atom) and its neighbors interacted to each other. There are two preservations in the proposed model: the first one is bond distance preservation between two bonded particles and the second one is preservation of angle formed by three atoms. Potential energy functions of atoms that are bonded by triangle mesh are given in following sections.

### Potential energy due to bond stretching

The bond representation of a triangular model is given in Figure 2a. In this model the covalent links are springs and vertices of the triangle are the atoms. The potential energy for bond stretching and compressing can be defined (Rapaport, 2004; Burket and Allinger, 1982) based on Hooke's Law as,

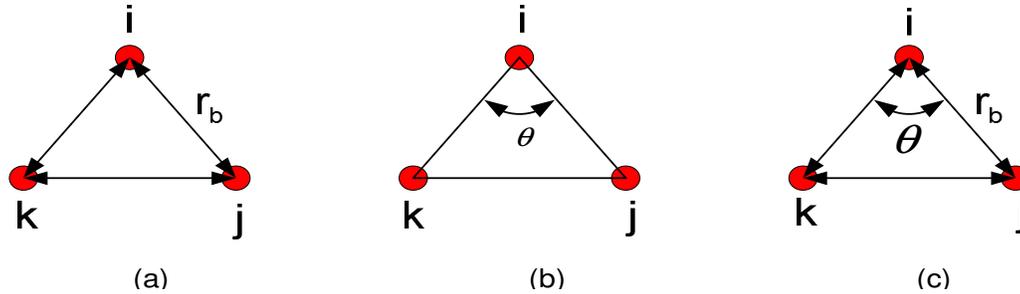
$$\Phi_b = \sum_{1,2 \text{ pairs}} \frac{1}{2} k_b (r_b - r_0)^2 \quad (3)$$

Where  $k_b$  is the bond force constant,  $r_b$  is the bond current distance and  $r_0$  is bond equilibrium length. Any deviation from equilibrium length will change the potential energy.

### Potential energy due to bond angle bending

Angle bending energy is also included in molecular dynamics. The angle is defined by three atoms forming the triangle as in Figure 2b. The angle potential is given as,

$$\Phi_\theta = \sum_{bond} \frac{1}{2} k_\theta (\cos \theta - \cos \theta_0)^2 \quad (4)$$



**Figure 2.** Triangle element: building block of deformable object. (a) Vertices are considered as atoms while the links between atoms are bonds with bond length  $r_b$ , (b) The bond angle,  $\theta$ , is formed by three atoms of a triangle and (c) bond stretch-bend interaction.

Where  $k_\theta$  is the angular constant,  $\cos \theta$  is the bond current angle and  $\cos \theta_0$  is the equilibrium bond angle. Any changes on equilibrium angle will change the bond angle potential.

**Potential energy due to stretch-bend interactions**

A cross term potential function is also employed in molecular mechanics to handle situations where changes in bond angle forces the two bonds forming the angle to stretch to alleviate the strain. The cross term potential function given in equation 5 therefore includes bond stretching and bending terms.

$$\Phi_{b,\theta} = \sum_{triangle} (\cos \theta - \cos \theta_0) [(r - r^0)_a + (r - r^0)_b] \quad (5)$$

Where  $a$  and  $b$  represent bonds to a common atom. Cross-term effect is illustrated in Figure 2c.

**Total internal forces and system dynamics**

The potential energy of a particle (an atom or a mass-point)  $\Phi$  is obtained summing of potential energies between bonded atoms.

$$\Phi = \Phi_b + \Phi_\theta + \Phi_{b,\theta} \quad (6)$$

The force  $f$  acting on individual atom (mass-point) is then calculated using negative gradient of the potential energy:

$$f = -\nabla \Phi \quad (7)$$

There are three potential energies defined (equations 3, 4, 5) for the proposed model. The internal forces for each potential energy are derived. Details of the derivation of these force functions are given in Appendix. Force due to bond length changes (bond stretch) is:

$$f_b = -\nabla_b \Phi_b = -k_b (r_b - r_0) \vec{r}_b \\ = -k_b (r_{ij} - r_{ij}^0) \vec{r}_{ij} + k_b (r_{ik} - r_{ik}^0) \vec{r}_{ik} + k_b (r_{jk} - r_{jk}^0) \vec{r}_{jk} \quad (8)$$

Where  $i, j$  and  $k$  are indices of atoms and vectors  $\vec{r}_{ij}, \vec{r}_{ik}$  and  $\vec{r}_{jk}$  show force directions for bonds between atoms. Bond current distances are  $r_{ij}, r_{ik}$  and  $r_{jk}$ . Initial bond distances are represented by  $r_{ij}^0, r_{ik}^0$  and  $r_{jk}^0$ . Bond angle bending force can be obtained as:

$$f_\theta = -\nabla_\theta \Phi_\theta = -k_\theta [\cos \theta - \cos \theta_0] \frac{r_{jk}}{r_{ik} r_{ij}} \vec{r}_{jk} \quad (9)$$

The force function caused by stretch- bend interaction is derived as:

$$f_{b,\theta} = -\nabla_{b,\theta} \Phi_{b,\theta} \\ = -\frac{r_{jk}}{r_{ij} r_{ik}} [(r_{ij} - r_{ij}^0) + (r_{ik} - r_{ik}^0)] \vec{r}_{jk} + (\cos \theta - \cos \theta_0) \vec{r}_{ij} + (\cos \theta - \cos \theta_0) \vec{r}_{ik} \quad (10)$$

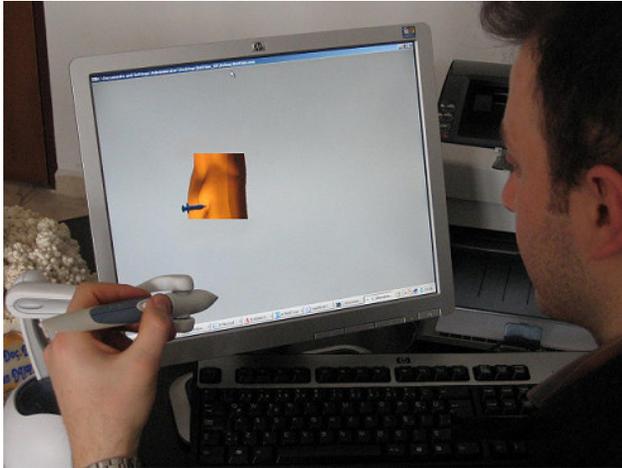
Total internal force of the proposed method is the sum of the forces due to bond stretching, angle bending and stretch-bond interactions,

$$f_{int} = f_b + f_\theta + f_{b,\theta} \quad (11)$$

If we compare the internal force function of proposed method (equation 11) with that of mass-spring systems (equation 1), proposed method provides a more physically realistic force function representing characteristics of deformable objects.

**RESULTS AND DISCUSSIONS**

Simulation algorithm needs to handle collisions which can occur between the particles of the deformable object or with the simulation environment. We used an algorithm described in (Provot, 1997) to handle any possible collision. Once the internal forces are known (equation 11) one of the many well known integration method can be



**Figure 3.** Syringe insertion is simulated using Phantom Omni haptic device

used, e.g. Euler explicit integration. In this work, the Verlet Integration Method is used to compute positions and velocities of points in deformable model. In the Verlet algorithm,  $x(t)$  defines the position,  $v(t)$  is the velocity and  $a(t) = \frac{F(t)}{m}$  is the acceleration at time  $t$  of a point with mass  $m$ . Iteration time step is defined with  $\Delta h$ . The Verlet integration method is given as (Teschner et al., 2004),

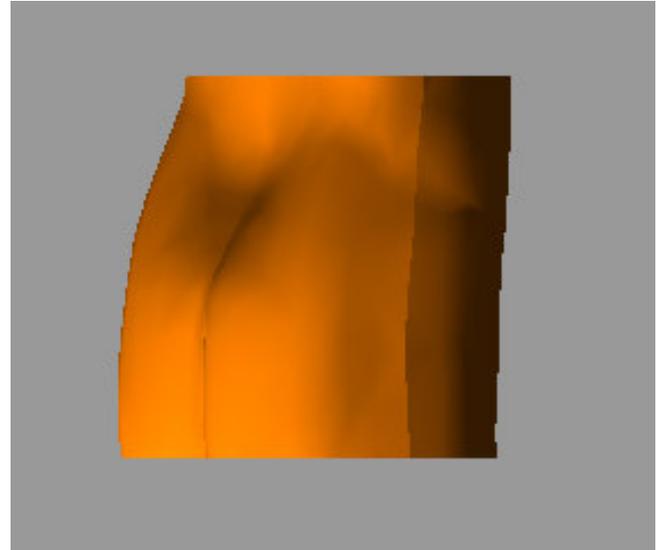
$$\begin{aligned}
 x(t + \Delta h) &= 2x(t) - x(t - \Delta h) + \Delta h^2 a(t) + O(\Delta h^4) \\
 v(t) &= \frac{x(t + \Delta h) - x(t - \Delta h)}{2\Delta h} + O(\Delta h^2)
 \end{aligned}
 \tag{12}$$

In the first equation, the calculation of positions is independent of velocities but requires knowledge about positions at time  $t - \Delta h$ . Computational cost of the Verlet Integration method is similar to Euler’s method. We created our simulation environment using C++, OpenGL and Haptic Device APIs as a software part with a Phantom Omni haptic device as hardware as shown in Figure 3. ]

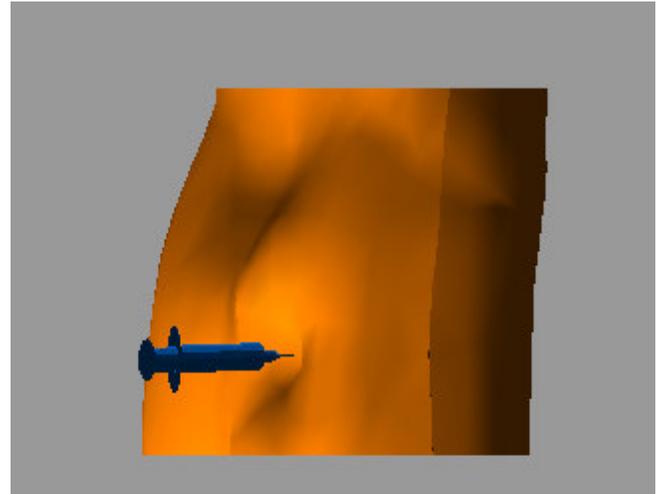
We first simulated syringe insertion to a human model. The model consists of 1886 atoms (vertices) and 868 bonds (springs). We were able perform interactive real-time simulations. Figure 4 shows a syringe deforming the model.

The second simulation is performed by pulling a moon model by a hook, Figure 5a. The moon model consists of 3198 atoms and 9504 bonds. Real-time performance is also achieved for this simulation.

A third simulation is also provided to show multiple interactions, Figure 5b. A simple net is hit by a ball and a tool from different sides while it is pulled down from one of its atom. As can be seen from the Figure, three different in-



a

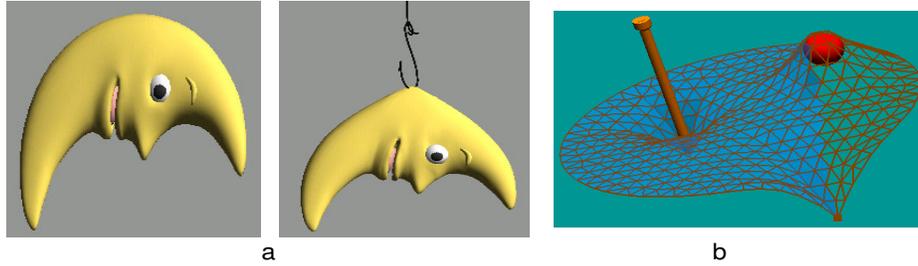


b

**Figure 4.** Syringe insertion simulation; a) before the simulation and b) after deformation is performed.

interactions are handled successfully by the simulation algorithm.

In this work, we incorporate mass-spring systems and molecular dynamics. Force functions are derived from potential functions of molecular modeling. Developed algorithm works on triangular meshes unlike classical particular systems. Previous works only used Lennard-Jones potential formula for calculating internal forces. In this work we use bond stretching, angle bending and stretch-bend interaction formulas for internal force generation. We successfully applied our algorithm to various deformable object simulations. The proposed algorithm works in real-time as mass-spring systems but provides physically more accurate results because of the accurate internal force generation based on molecular modeling.



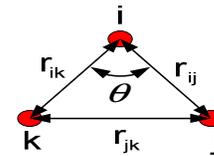
**Figure 5.** (a) A moon model is pulled by a hook. (b) A simple net is faced with several interactions, a tool and a ball.

**REFERENCES**

Baraff D, Witkin A (1999). Physically based modeling course notes, Course 36, Siggraph'99.  
 Burket U, Allinger NL (1982). Molecular Mechanics, Am. Chem. Soc.  
 Debunne G, Desbrun M, Cani MP, Barr A (2001). Dynamic real-time deformations using space and time adaptive sampling. Computer Graphics proceedings Aug 2001, proceeding of Siggraph'01.  
 Duysak A, Zhang JJ (2003). Identification of simulation parameters using neural networks. 6th International conference on Computer Graphics and Artificial Intelligence pp. 189-199.  
 Kang YM, Choi JH, Cho HG (2000). Fast and stable animation of cloth with an approximated implicit method. Proceedings of the Computer Graphics International (CGI'00) pp. 247-255.  
 Lloyd BA, Szekely G, Harders M (2007). Identification of spring parameters for deformable object simulation. IEEE Trans. Visualization Comput. Graphics, 13(5): 1081-1094.  
 Mesit J, Guha R, Chaudhry S (2007). 3D Soft Body Simulation Using Mass-Spring Systems with Internal Pressure Force and Simplified Implicit Integration. J. Comput. 2(8).  
 Nealen A, Müller M, Keiser R, Boxermann E, Carlson M (2005). Physically Based Deformable Models in Computer Graphics Proceedings of Eurographics pp. 71-94.  
 Pithioux M, Lopez O, Meier U, Monserrat C, Juan MC, Lcaniz M (2005). ParSys: a new particle system for the introduction of on-line physical behaviour to three-dimensional synthetic objects. Comput. Graphics pp. 135-144.  
 Provat X (1997). Collision and self collision handling in cloth model dedicated to design garments. Graphic Interface pp.177-89.  
 Rapaport DC (2004). "The Art of Molecular Dynamics Simulation", 2nd Edition, Cambridge University Press.  
 Teschner M, Heidelberger B, Muller M, Gross M (2004). A versatile and robust model for geometrically complex deformable solids. Computer Graphics International. Proceedings pp. 312-319.  
 Tonnesen D (1991). Modeling liquids and solids using thermal particles. In Graphics Interface pp. 255-262.  
 Wu X, Downes MS, Goktekin T, Tendick F (2001). Adaptive nonlinear finite Elements For deformable body simulation using dynamic progressive meshes. Eurographics, 20: 349-358.

**APPENDIX**

The building block of triangulated models is a simple triangle as shown  
 Equation 4 includes a  $\cos \theta$  term and finding force function therefore requires derivation of this  $\cos \theta$  function. Below equation represents the relationships between bond lengths and  $\cos \theta$ ,



$$r_{jk}^2 = r_{ij}^2 + r_{ik}^2 - 2r_{ij}r_{ik} \cos \theta$$

$$\cos \theta \text{ term is then } \cos \theta = \frac{r_{ij}^2 + r_{ik}^2 - r_{jk}^2}{2r_{ij}r_{ik}}$$

Taking gradient of equation 4 with respect to  $r_{jk}$  is then,

$$f_b - k_\theta [\cos \theta - \cos \theta_0] (\cos \theta)'$$

Where;

$$(\cos \theta)' = -\frac{r_{jk}}{r_{ik}r_{ij}} \vec{r}_{jk}$$

Equation 5 is given based on triangular notation as

$$\Phi_{b, \theta} = \sum_{\text{triangle}} (\cos \theta - \cos \theta_0) [(r_{ij} - r_{ij}^0) + (r_{ik} - r_{ik}^0)]$$

Negative gradient of this function with respect to  $r_{ij}$ ,

$r_{ik}$  and  $\cos \theta$  is

$$f_{b, \theta} = -\left[ \frac{\partial \Phi}{\partial \theta} \vec{r}_{jk} + \frac{\partial \Phi}{\partial r_{ij}} \vec{r}_{ij} + \frac{\partial \Phi}{\partial r_{ik}} \vec{r}_{ik} \right]$$

$$f_{b, \theta} = -[(\cos \theta)' [(r_{ij} - r_{ij}^0) + (r_{ik} - r_{ik}^0)] + [(r_{ij})' + (r_{ik})'] \cos \theta] \vec{r}_{jk} - (\cos \theta - \cos \theta_0) \vec{r}_{ij} + (\cos \theta - \cos \theta_0) \vec{r}_{ik}$$

Since derivative of  $\cos \theta$  is already known, the above equation can be rearranged to obtain the force function given by equation 10.