Physically Based Modeling for Interactive Simulation and Games
Scribe Notes for the lecture on February 7th
Spring 2011
FunShing Sin

Finite Element Method (FEM) and Mass Spring System

Mass Spring System is easy to understand and implement. However, it has many disadvantages. For example, it has no material properties, meaning that it is very difficult to model different materials. Also, it does not converge to the true solution as the spring spacing goes to zero.

In contract, Finite Element Method (FEM) is a more accurate way to solve Partial Differential Equations (PDEs). In FEM, objects are discretized into many small elements (e.g. tetrahedra). Then the continuum PDEs is converted to a discretized system of Ordinary Differential Equations (ODEs), which can be time-stepped numerically.

Mesh Representation

One way to represent an object in computer graphics is to decompose the object into many small tetrahedra. Figure 1 shows two examples where the bunny and the dragon are discretized into volumetric meshes. A good quality volumetric mesh is very important to underlying simulations because ill-shaped tetrahdra are bad for stiffness matrix conditioning [5] and will cause numerical instabilities. Creating volumetric meshes with good dihedral angles still remains a difficult problem and is an active research area in computational geometry.

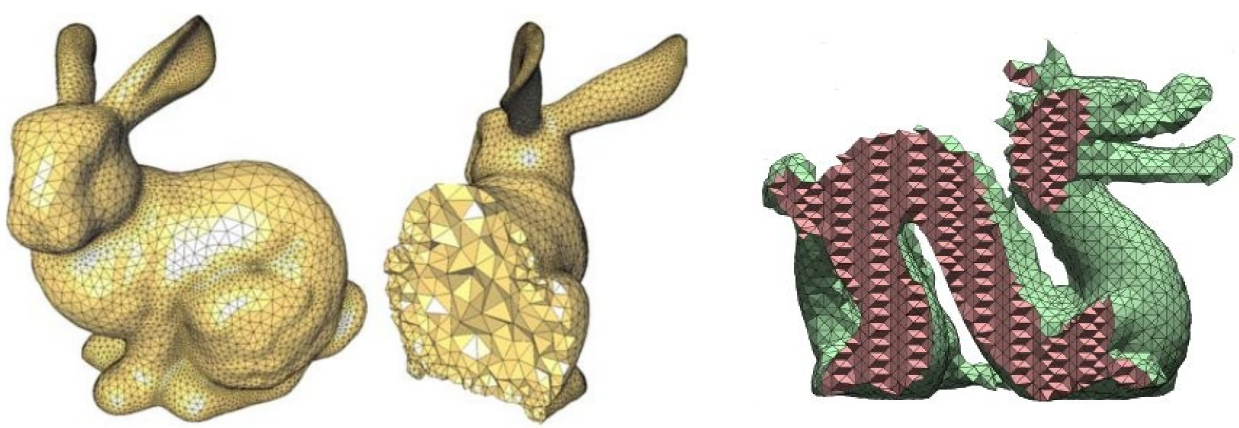

Figure 1: The volumetric meshes for bunny [1] and dragon [2]

Embedded Surface Mesh

Given a high resolution surface mesh, a very fine volumetric mesh is required to capture the surface details. However, such fine volumetric mesh may contain a lot of tetrahedra, and performing calculations on a large number of tetrahedra can be very time consuming. To alleivate this, we can embed the surface mesh in a coarse volumetric mesh (see Figure 2). We use the surface mesh for rendering as well as for collision dynamics, but all of the elasticity and plasticiy computations come from the coarse volumetric mesh. We can then intepolate deformations from the volumetric mesh to the surface mesh using barycentric coordinates.

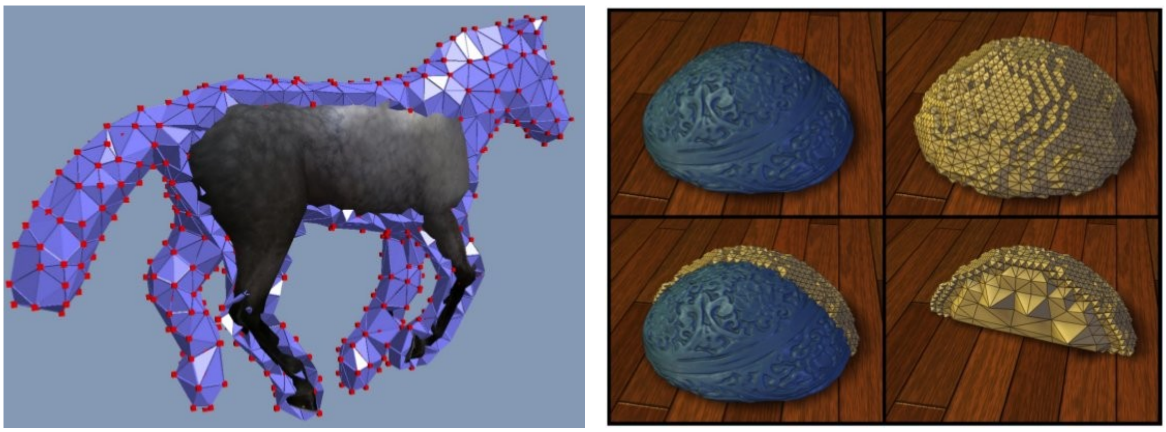

Figure 2: (Left) A detailed surface is embedded in a coarse volumetric mesh [3], (Right) A high reolution surface mesh (blue) embedded into a low resolution mesh (gold) [6]

Young's Modulus and Poisson Ratio

The Young's Modulus E and the Poisson Ratio ν are two important parameters in deformable object simulations. Let us consider a beam with cross section area S and initial length l_0 as shown in Figure 3.

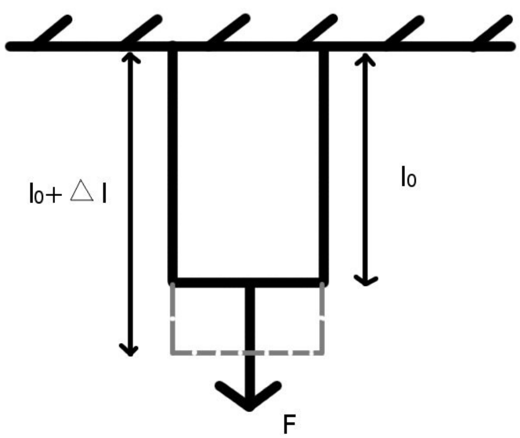

Figure 3: A beam is fixed at one end and is pulled at the other end by an extern force F

When an extern force F is applied in the direction of the beam perpendicular to the cross section S, the beam with original length l expands by Δl . These quantities are related via Hooke's law as

$$\frac{F}{S}=E\frac{\Delta l}{l_0}$$

The equation states that the elongation of the beam increases as the force per area increases. If the force per area (i.e. lhs) is fixed, then the elongation is small when the Young's modulus E is large. On the other hand, if the Young's modulus is small then the elongation is large. Therefore, the Young's modulus E describes the *stiffness* of the beam.

The behavior of the beam can also be described by the following equation:

$$\frac{\Delta V}{V_0}=(1-2\nu)\frac{\Delta l}{l_0}$$

where V_0 is the initial volume of the beam, and ΔV is the change of the volume. If we keep $\frac{\Delta l}{l_0}$ fixed, then when ν equals 0.5, the change of volume would be 0, which means the volume is conserved. Therefore, the Poisson ratio ν describes the compressibility of the beam.

Displacement Vector

The displacement (or deformation) of the mesh vertices can be concatenate into one long vector. Let say we discretize the beam into six triangles and deform the beam to the right:

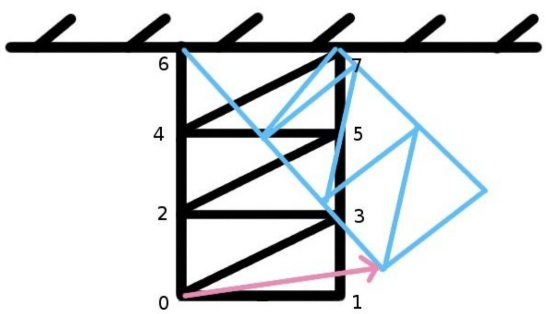

Figure 4: A beam (in black color) is deformed to the right (in blue color).
The red arrow shows the displacement vector $[u_{0,x}\, u_{0,y}\, u_{0,z}]$ of the vertex 0.

The displacement vector can be written as

$$\vec{u}=\begin{bmatrix} u_{0,x} \\ u_{0,y} \\ u_{0,z} \\ \vdots \\ u_{7,z} \end{bmatrix}$$

where the first sub-index represents the n-th vertex, and the second sub-index represents the x, y, or z component of the displacement at n-th vertex (e.g. $u_{0,x}$ is the x-component of the displacement of vertex 0) . Note that the size of the concatenated displacement vector $\vec{u}$ equals to the number of vertices in the mesh multiplies with 3 (i.e. DOF of $\vec{u}$ = # vertices * 3).

Equations of Motion

The governing equation (derived from the principle of virtual work of Lagrangian mechanics) that describes the dynamics of the system can be written as:

$$M\ddot{u}+D\dot{u}+f_{int}(u)=f_{ext}(u)$$

Here, $u\in\mathbb{R}^{3n}$ is the concatenated displacement of the vertices, n is the number of vertices, $M\in\mathbb{R}^{3n,3n}$ is the mass matrix, $D\in\mathbb{R}^{3n,3n}$ is the damping matrix, $f_{int}(u)\in\mathbb{R}^{3n}$ is the internal deformation force, and $f_{ext}(u)\in\mathbb{R}^{3n}$ is the external force (e.g. gravity).

The internal deformation force $f_{int}(u)$ can be written as the following using Taylor series:

$$f_{int}(0+u)\approx f_{int}(0)+\frac{\partial f_{int}}{\partial u}(u)+O(\|u\|^2)$$

where the derivative $\frac{\partial f_{int}}{\partial u}$ is evaluated at the rest/initial configuration. The derivative is also known as the *Stiffness Matrix K*, which denotes the Jacobian matrix of the internal forces.

In our beam example, the Stiffness Matrix K has the following layout:

$$K=\begin{bmatrix}\frac{\partial f_{int}^{1}}{\partial u_{0,x}} & \frac{\partial f_{int}^{1}}{\partial u_{0,y}} & \cdots & \frac{\partial f_{int}^{1}}{\partial u_{7,z}}\\ \vdots & \vdots & \ddots & \vdots\\ \frac{\partial f_{int}^{24}}{\partial u_{0,x}} & \frac{\partial f_{int}^{24}}{\partial u_{0,y}} & \cdots & \frac{\partial f_{int}^{24}}{\partial u_{7,z}}\end{bmatrix}$$

where f_{int}^{m} is the m-th component of the f_{int} vector.

At the rest/initial configuration, the displacement vector is $\vec{0}$ and so the internal force $f_{int}(0)$ is also $\vec{0}$.

Therefore, the internal force can be written as:

$$f_{int}(u)\approx\frac{\partial f_{int}}{\partial u}(u)=Ku$$

And our governing equation becomes:

$$M\ddot{u}+D\dot{u}+Ku=f_{ext}(u)$$

Key Idea of "Interactive Virtual Materials"

The key idea of the paper "Interactive Virtual Materials" is to compute elastic forces at a tetrahedron using the following steps:

1) Rotate the deformed tet to an unrotated frame.
2) Compute the force by multiplying the rotated displacement with the stiffness matrix.
3) Rotate the force back to the deformed frame.

To determine the rotation of a tetrahedron, the simplest way is to use Gram-Schmidt method. For more accurate results, *Polar Decomposition* should be used:

$$A=RS$$

The polar decomposition decomposes the deformation gradient A into a rotation matrix R and a scaling matrix S. Notice that S is a 3x3 symmetric matrix, which has 3 orthogonal eigenvectors (see Appendix C2 of [4]). So it can be decomposed into $S=Q\Lambda Q^{-1}$ where Q is the eigenvectors and Λ is the eigenvalues. Therefore, S can be think of a matrix that scales a vector along the directions of the eigen vectors.

References

[1] Alliez, P., Cohen-Steiner, D., Yvinec, M., And Desbrun, M. 2005. Variational Tetrahedral Meshing. *ACM Transcations on Graphics 24,* 3, 617-625. Special issue on Proceedings of SIGGRAPH 2005.

[2] Labelle, F., And Shewchuk, J. R. 2007. Isosurface Stuffing: Fast Tetrahedral Meshes With Good Dihedral Angles. *ACM Transactions on Graphics 26,* 3 (Aug.), 57.

[3] Muller, M., Stam, J., James, D., Thurey, N. 2008. Real Time Physics: Class Notes. *ACM SIGGRAPH 2008 class notes*, pp. 1-90.

[4] Shewchuk, J. R. 1994. An Introduction to the Conjugate Gradient Method Without the Agonizing Pain. Technical Report CMU-CS-TR-94-125. Carnegie Mellon University.

[5] Shewchuk, J. R. 2002. What is a Good Linear Element? Interpolation, Conditioning and Quality Measures, *11 th Int. Meshing Rountable.* pp. 115-126

[6] Wojtan, C., And Turk, G. 2008. Fast Viscoelastic Behavior with Thin Features. *ACM Transactions on Graphics 27,* 3, 47.