

# Real-time and Physically Realistic Simulation of Global Deformation<sup>\*</sup>

Yan Zhuang<sup>†</sup>

John Canny<sup>‡</sup>

Computer Science Department, University of California, Berkeley, CA 94720-1776

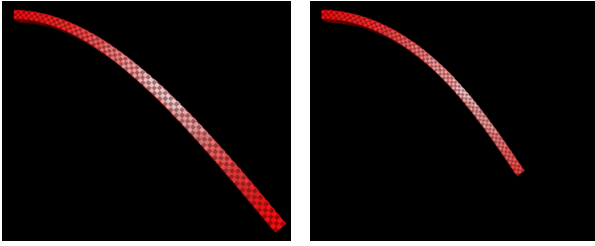
**Global Deformation Using Nonlinear FEM:** We apply the displacement based finite element methods (FEM) to the simulation of large motions and global deformations of deformable objects. A linear strain model leads to unacceptable distortion (figure 1). To avoid this problem, we apply a *quadratic strain* instead (figure 1). Essentially this requires solving the following *nonlinear* system of differential equations

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{R}(\mathbf{u}) = \mathbf{F} \quad (1)$$

where  $\mathbf{u}$  is the nodal displacements;  $\dot{\mathbf{u}}$  and  $\ddot{\mathbf{u}}$ , the respective velocities and accelerations;  $\mathbf{F}$ , the external force;  $\mathbf{M}$ , the mass matrix;  $\mathbf{D}$ , the damping matrix; and  $\mathbf{R}(\mathbf{u})$ , the *nonlinear* internal force due to deformation.

For a soft material such as live tissue, the material stiffness is small. This makes explicit time integration scheme appropriate because we can take large time steps. In particular We apply explicit Newmark scheme to equation (1).

We diagonalize the mass matrix  $\mathbf{M}$  and the damping matrix  $\mathbf{D}$ . This leads to a decoupled system of nonlinear equations, which requires no matrix inversions to solve.



**Figure 1:** The left end of the beam is fixed. The left image shows its *distorted* deformation under gravity, using linear strain. The right image shows the *undistorted* deformation, under the same gravitational force, using quadratic strain.

**Collision Integration Scheme:** Simulating deformable object collisions using a penalty method [3] requires tiny time steps to produce visually satisfactory animations. A general impulse collision [1] is considered more efficient and accurate but still requires more computation than collision-free dynamics.

For deformable object collisions, the collision time can be assumed finite (unlike the instantaneous collision of rigid bodies). By recognizing this, we propose an efficient way to handle collision.

Consider the collision between a deformable body with a stationary rigid body (figure 2). Assume at time  $t_n$ , the node  $p$ , with velocity  $\hat{\mathbf{v}}(p)_n$ , is colliding with a rigid surface of outward normal  $\hat{\mathbf{n}}$ . Then the non-penetration constraint at node  $p$  can be enforced by setting the normal component of  $\hat{\mathbf{v}}(p)_{n+1}$  to zero as following:

$$\hat{\mathbf{v}}(p)_{n+1} = \hat{\mathbf{v}}(p)_n + (\hat{\mathbf{v}}(p)_n \cdot \hat{\mathbf{n}})\hat{\mathbf{n}} \quad (2)$$

If we choose  $\Delta t_{n+1} = \Delta t_n$  for the Newmark scheme, we have

$$\hat{\mathbf{u}}_{n+2} \cdot \hat{\mathbf{n}} = \hat{\mathbf{u}}_n \cdot \hat{\mathbf{n}} \quad (3)$$

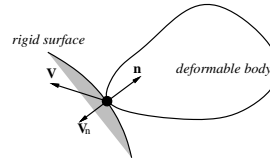
<sup>\*</sup>Supported by a Multi-Disciplinary Research Initiative grant for 3D Visualization, sponsored by BMDO with support from ONR.

<sup>†</sup>yzhuang@cs.berkeley.edu

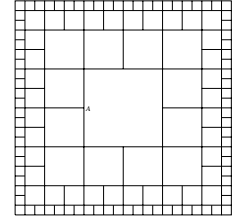
<sup>‡</sup>jfc@cs.berkeley.edu

This shows that the non-penetration constraint is exactly enforced after two time steps, *without* solving a constrained problem.

For a decoupled system, this collision handling scheme can be easily generalized to multiple point collision constraint. And unlike the coupled system, no explicit impulse calculation is necessary for frictionless collisions. When friction has to be considered, the equivalent impulse can be easily computed without matrix inversion. This collision-handling integration scheme can be easily generalized to collisions between a deformable body and a moving rigid body, and to collisions between deformable bodies. It is worth noting that a deformable object's resting on a surface can be handled exactly the same way.



**Figure 2:** A flexible body collides with a rigid body.



**Figure 3:** A 2D example of graded mesh.

**Graded Mesh:** While 2D FEM has great success in achieving real time performance in computer graphics applications, the computational cost is much higher for 3D applications, mainly due to the increase in the number of elements in the mesh. In a roughly uniform 2D finite element mesh, the number of elements is about  $O(n^2)$ , where  $n$  is the average number of elements in each principle direction. However a similar 3D mesh would have  $O(n^3)$  elements, which leads to a much larger system of equations.

Nicolson [2] shows that the cutoff spatial frequency of an object in response to external loads decreases faster than  $1/d$  in terms of the distance  $d$  away from the surface. Therefore if we use a 3D graded mesh similar to the 2D example in figure 3, we will lose little accuracy with respect to static forces on the surface, while reducing the complexity of the problem from  $O(n^3)$  to  $O(n^2)$ . A similar grading property also applies to tetrahedral mesh.

To maintain the geometric compatibility at the element interface, we simply set the displacement of edge constrained nodes, such as node  $A$  in figure 3, to the midpoint of the corresponding edge. Similar constraint apply to the face constrained nodes in a 3D mesh.

On a 400MHz Pentium II PC, a uniform mesh of 1331 elements needs about 0.11 seconds per time step. The graded mesh with the same accuracy needs only 0.06 seconds per step.

## References

- [1] David Baraff and Andrew Witkin. Dynamic simulation of non-penetrating flexible bodies. In *Computer Graphics: Proceedings of SIGGRAPH*, pages 303–308. ACM, 1992.
- [2] Edward John Nicolson. *Tactile Sensing and Control of a Planar Manipulator*. PhD thesis, EECS, University of California, Berkeley, 1987.
- [3] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. Elastically deformable models. *Computer Graphics*, 21, July 1987.