Summary of *Cutting in deformable objects*

Thesis by Han-Wen Nienhys, Universiteit Utrecht, 2003
Summary by Christopher Sewell

Chapter 1

This chapter just gives a brief overview of the thesis. That's what I'm about to do here...

Chapter 2

This chapter gives a pretty complicated overview of continuum mechanics, the Finite Element Method, unconstrained optimization algorithms, and time integration algorithms. It's a good time. I don't really understand a whole lot, but here are a few seemingly useful things I learned.

Continuum Mechanics. A deformation may be described by a function $p(z,t)$ which maps points in a body (a subset β of $\Re^3$) to new positions at a given time: $p : \beta \times \Re \rightarrow \Re^3$. The derivative of this function with respect to the reference position z is the deformation gradient F, $F = \frac{\partial p}{\partial z}$. The volume change due to a deformation $p(z,t)$ is $\det(F)$. The (right Cauchy-)Green deformation tensor C is defined $C = F^T \cdot F$. For an infinitesimally long fiber from z to z+h, its length after deformation is $\sqrt{h \cdot C \cdot h}$. The displacement of a point may be described by a function $u(z,t) = p(z,t) - z$, and its derivative with respect to the reference position is G, $G = \frac{\partial u}{\partial z}$. The material strain tensor E is defined by $E = \frac{1}{2}(C - I)$, and its linear geometry approximation is ε, $\varepsilon = \frac{1}{2}(G + G^T)$. The stress tensor S is $S = 2\mu\varepsilon + \lambda I(\text{trace}(\varepsilon))$. The approximation of the energy density function W is $\tilde{W} = \mu\text{trace}(\varepsilon * \varepsilon) + \frac{1}{2}\lambda(\text{trace}(\varepsilon))^2$. The parameters λ and μ are the Lamé parameters, which can be derived from Young's Modulus, $E = \mu\frac{3\lambda+2\mu}{\lambda+\mu}$ and the Poisson ratio $\upsilon = \frac{\lambda}{2(\lambda+\mu)}$. Young's Modulus is the resistance of a material to stretching; the Poisson ratio is the ratio between transverse contraction and longitudinal stretching. Given a traction t, the dilation in the direction of the force is $\frac{t}{E}$, and the contraction in the perpendicular direction is $\frac{\upsilon t}{E}$.

Finite Element Method. The description given here of the FEM was difficult, for me at least, to understand. So instead I'll try to illustrate it with a couple of examples from my numerical analysis textbook (with some extra steps shown for clarity). The specific FEM used in this thesis is more understandable than the general description of the method given in Chapter 2, and is given in Chapter 3. The FEM attempts to find a approximation to the true solution $y(t)$ with a function $u(t)$ that is a finite collection of basis functions (usually piecewise polynomials):

$$y(t) \approx u(t) = \sum_{i=1}^{n} x_i \phi_i(t)$$

The coefficients x_i are determined according to some criteria. One such criteria is the Collocation Method, in which the differential equation is satisfied exactly at a certain number of points. As an example, let's try to solve the boundary value ordinary differential equation

$$y'' = 6t, \qquad 0 \leq t \leq 1$$

with boundary conditions y(0)=0 and y(1)=1, with one interior mesh point (t=0.5), using the trial function

$$u(t) = x_0 + x_1 t + x_2 t^2$$

Substituting in the boundary conditions,

$$x_0 = 0 \qquad and \qquad x_1 + x_2 = 1$$

At the interior mesh point t=0.5, we set the trial function equal to the true solution:

$$y''(0.5) = u''(0.5) = 2x_2 = f(0.5) = 6(0.5)$$

From this system of three equations, we get

$$x_0 = 0, \qquad x_1 = -0.5, \qquad x_2 = 1.5$$

which gives the approximate solution

$$u(t) = -0.5t + 1.5t^2$$

Of course, in a real problem there would be many more than one interior mesh points, leading to a much larger system of such linear equations. The analytical solution is $y(t) = t^3$, which is equal to the approximate solution at the boundary points and at the mesh point

$$y(0.5) = u(0.5) = 0.125$$

Another criteria is the Galerkin Method, which specifies that the residual is orthogonal to the space spanned by the basis functions. Using the "hat" basis functions:

$$\phi_1(t) = \begin{cases} -2t + 1 & 0 \leq t \leq 0.5 \\ 0 & \text{else} \end{cases}$$

$$\phi_2(t) = \begin{cases} 2t & 0 \le t \le 0.5 \\ -2t+2 & 0.5 < t \le 1.0 \\ 0 & \text{else} \end{cases}$$

$$\phi_3(t) = \begin{cases} 2t-1 & 0.5 \le t \le 1.0 \\ 0 & \text{else} \end{cases}$$

we want an approximate solution

$$y(t) \approx u(t) = x_1\phi_1(t) + x_2\phi_2(t) + x_3\phi_3(t)$$

Since $y(0) = 0$, $x_1 = 0$, and since $y(1) = 1$, $x_3 = 1$. According to the criteria, the residual $u''(t) - f(t)$ must be orthogonal to each basis function. To get x_2, then, we set the inner product of the residual and ϕ_2 equal to zero. The inner product on a function space is the integral of the product of the functions, so

$$\int_0^1 (u''(t) - 6t)\phi_2(t)dt = 0$$

$$\int_0^1 u''(t)\phi_2(t)dt - 6\int_0^1 t\phi_2(t)dt = 0$$

$$u'(t)\phi_2(t)|_0^1 - \int_0^1 u'(t)\phi_2'(t)dt - 6(\int_0^{1/2} t(2t) + \int_{1/2}^1 t(-2t+2)) = 0$$

$$0 - \sum_{i=1}^3 x_i \int_0^1 \phi_i'(t)\phi_2'(t)dt - 6(\frac{1}{12} + \frac{1}{6}) = 0$$

$$-(x_1(-1/h) + x_2(2/h) + x_3(-1/h)) - \frac{3}{2} = 0$$

With h, the spacing between mesh points, equal to $\frac{1}{2}$, we get

$$-2x_1 + 4x_2 - 2x_3 = -\frac{3}{2}$$

Again, in a real problem there would be many more than one interior mesh points, leading to a system of such linear equations. Using x_1 and x_3, previously determined by the boundary conditions, we get $x_2 = \frac{1}{8}$, yielding the approximate solution function

$$u(t) = \frac{1}{8}\phi_2(t) + \phi_3(t)$$

At t=0.5, this also gives the correct value of 0.125.

Linear Conjugate Gradient Method. The potential energy function in a linear elastic FEM problem may be of the form

$$\Pi(x) = \frac{1}{2}x^T Kx - b^T x$$

where K is a given symmetric positive definite matrix and b is a given vector. We want to find an x that minimizes Π. The basic idea is to calculate the next approximation x_{k+1} based on the current approximation x_k, moving a certain amount α_k in a chosen direction d_k:

$$x_{k+1} = x_k + \alpha_k d_k$$

Given an initial guess x_0, the pseudocode for the conjugate gradient method is given:

```
$k \leftarrow 0$
$r_0 \leftarrow b - Kx_0$
while $\|r_k\|$ too large
        if $k = 0$ $\beta_k \leftarrow 0$
        else $\beta_k \leftarrow \frac{\|r_k\|^2}{\|r_{k-1}\|^2}$
        $d_k \leftarrow r_k + \beta_k d_{k-1}$
        $\alpha_k \leftarrow \frac{\|r_k\|^2}{d_k^T(Kd_k)}$
        $x_{k+1} \leftarrow x_k + \alpha_k d_k$
        $r_{k+1} \leftarrow r_k - \alpha_k K d_k$
        $k \leftarrow k + 1$
```

Non-linear conjugate gradients and time integration are also discussed.

Mechanics of soft tissue. Elastine, found in skin, artery walls, and lung tissue, is a rubbery material that is almost perfectly elastic – its loading and unloading cycles are almost equal. Collagen is a load bearing material found in tendons, bone, skin, and blood vessels, and can deform differently in different directions (aelotropic or anisotropic). Hysteresis is the phenomenon in which a tissue offers more resistance when first stretched than during a following unload. Creep is the process in which a tissue continues to distend further after an initial elastic response when stressed with a constant load. After a number of loads and unloads, the differences between the cycles disappear, and the tissue is said to be preconditioned.

Chapter 3

This chapter describes the specific method used to simulate cutting and deformations in a mesh. It begins with a brief description of previous work on FEM deformable meshes, including work by Bro-Nielsen, James and Pai, Cotin and Picinbono, Zhuang and Canny, Szekely, and Ganovelli.

Finite Element Model. As described under Continuum Mechanics in Chapter 2, the displacement of a point may be described by a function $u(z, t)$, and its derivative with respect to the reference position z is G,

$$G = \frac{\partial u}{\partial z}$$

The linearized material strain tensor is ε,

$$\varepsilon = \frac{1}{2}(G^T + G)$$

With linear elasticity, the first Piola-Kirchoff stress tensor T is equal to the stress-tensor S,

$$T = S = 2\mu\varepsilon + \lambda \text{trace}(\varepsilon)\text{I}$$

The Lamé parameters μ and λ can be solved from Young's Modulus and the Poisson Ratio, as I described under Chapter 2 Continuum Mechanics. Given the four points of a linear tetrahedron τ with reference positions z_1 ... z_4, a tensor Z transforms them from a unit tetrahedron to τ:

$$Z = (z_1 - z_4) \bigotimes e_1 + (z_2 - z_4) \bigotimes e_2 + (z_3 - z_4) \bigotimes e_3$$

The elastic force for each of the first three vertices $j = 1$ ... 3 of a tetrahedron τ are

$$f^{el}_{j,\tau} = -v(\tau)(T \cdot (Z^{-1})^T) \cdot e_j$$

with $v(\tau)$ the volume of the tetrahedron. The total elastic force for the tetrahedron should be zero, so the elastic force of the fourth vertex is defined by

$$f^{el}_{4,\tau} = -\sum_{j=1}^{3} f_{j,\tau}$$

In a static problem, the elastic forces should balance the external forces (body forces and surface tractions) at each node, so, for a given node i,

$$\sum_{\tau} f^{el}_{i,\tau}(u) + f^{ex}_i = 0$$

This leads to the linear system

$$Ku = f$$

The primitive stiffness matrix K is $3m$ by $3m$, where m is the total number of nodes (and the three is due to equations for the x, y, and z coordinates), and both the displacement vector u and the force vector f are of size $3m$. Some number N_{fix} of the nodes may be fixed and thus have a constant displacement, reducing the dimensions of the system to $3m - 3N_{fix}$. They solve this system using a matrix-free conjugate gradient algorithm by computing Kd for a given displacement d without explicitly computing K, since elastic forces in a node are only dependent on displacements of its neighbors.

Cutting. The path swept by a scalpel is approximated with a triangulation, $\triangle_1, \ldots, \triangle_k$. Each sweep triangle, ordered in time, is intersected with the edges of the mesh, marking the node of each intersected edge closest to the intersection point. Any face of a tetrahedron for which all three nodes are marked is "selected". To obtain a smoother cut, an internal node on the cut surface is moved to its projection on the plane of the cut surface. A boundary node is moved to the closest such projection among all boundary faces incident with it (if the projection lies in the face). However, if all nodes of a tetrahedron are selected and moved, flat elements and other degeneracies can occur. They iteratively collect a list of all degeneracies, heuristically remove them (usually by collapsing short edges by joining the two vertices of the edge), and repeat as long as the number of degeneracies is decreased. At the end, remaining ones are cut out, which can lead to spurious cracks in the incision. Tetrahedron flatness is defined as the minimum height divided by the maximum edge length.

Results and Discussion. They tested their deformation methods with a cantilever beam fixed at one end, having an analytical solution for the deflection of the loose end of $y = \frac{4F}{Ew}(\frac{l}{h})^3$, where E is Young's Modulus, F is the load, and l,w, and h are the beam dimensions. Errors reduced from 29.3% with 425 nodes to 3.3% with 8181 nodes. Their cutting algorithm successfully produced cuts without increasing mesh size, but their degeneracy removal methods sometimes result in topology changes, spurious cracks, or disconnected cuts, and is, in their words, unsatisfying in its heuristic nature. Also, since it must remember intersections from previous sweep triangles to produce connected incisions, and account for each change in the mesh when converting back to the reference positions, they admit it is slow and error prone.

Chapter 4

This chapter attempts to compare the convergence rates of a dynamic relaxation algorithm (specifically, SS22 time-integration with lumped masses and lumped damping) and a static relaxation algorithm (specifically, nonlinear Conjugate Gradient). Theoretically, it is shown that they have the same convergence rates given optimally chosen parameters for the dynamic algorithm. They are compared experimentally on a cantilever beam with various hyperelastic compressible material modes: St. Venant-Kirchoff, Linear, Neo-Hookean, and Veronda-Westmann. Dynamic relaxation has the advantage of iteration results with physical meanings, while static relaxation has the advantages of greater stability and no need for manual selection of parameters. Both are faster with smaller and higher-quality meshes. Neither method is really quite fast enough for real-time.

Chapter 5

This chapter discusses improving the quality of the mesh when cutting by using Delaunay criteria. A high-quality mesh has no angles close to 0 or π, since these can lead to high condition numbers. A Delaunay Triangulation is defined as a triangulation in which the circumcircle of each triangle does not contain any other points in the set. An edge is "legal" if the circumcircle of neither of its incident triangles contains the opposite node of the other incident triangle (empty-circle criterion). If an edge is "illegal", it can be flipped by removing that edge and replacing it with an edge between the two opposite nodes. By flipping all illegal edges until none remain, a Delaunay Triangulation of the set can be obtained. A Delaunay Triangulation maximizes the minimum angle across all triangles, which is equivalent to most other definitions of mesh quality.

Cuts in 2D. When the scalpel enters the mesh, an "active" node is attached to it and moves along with it. If the entry is sufficiently close to an existing boundary node, this node becomes the active node; otherwise, a new node is created. This produces a dent (rather than a cut) in the mesh at first (to avoid short edges), but once the scalpel is sufficiently far from the entry point, two new nodes are created on both sides of the entry point and connected to the active node, and the triangle defined by these three points is removed. As the scalpel (and thus also the active node) continue to move, the (imaginary) edge between the original entry point and the active node may fail the empty-circle criterion. If so, two new nodes are created where the opposite nodes intersect the scalpel path and are connected to these nodes. (There are some useful figures illustrating this on pages 90 and 91.) Again to avoid arbitrarily short edges, when the scalpel is close to exiting the boundary of the mesh, a new node is created at its current location, and the edge between this node and the boundary is dissected and then collapsed.

Cuts on 3D Surfaces. Given a triangulated 3-D surface, the scalpel position is represented by a line segment and intersected with the mesh. If there is one point of intersection, the active node is moved there. If the 2-D projection of the triangles incident to an active edge violates the empty-circle criterion, a flip is performed, although this is not always possible in 3-D without changing the topology (in which case the flip is not performed). When removing a node, edges are flipped in incident "ears" (two adjacent triangles that have the node as a vertex with the sum of the four total angles opposite the node less than π) until the node is a vertex of only three triangles. Then the node is removed and more flips are performed if necessary. Large movements of the scalpel are subdivided so that the active node does not move across the opposite edge of an incident triangle, causing a topological change. If the scalpel enters through the interior of the mesh, a new active node is created, and once it has moved sufficiently far from the entry point, a new node is placed at the entry point. The edge between these two nodes is split to form an incision. To prevent multiple incisions from interacting with each other, each triangle is allowed to be incident to at most one active node. If the scalpel line segment intersect multiple triangles, the incision branches into multiple incisions.

Results and Discussion. They implemented their 2-D and 3-D surface cutting algorithms and tested them with simple test cases – small, static meshes, with no haptics. In 2-D, it did improve mesh quality (maximum angle reduced from 60^o to 30^o) without increasing mesh size, although the incisions produced did not trace the actual scalpel movement as closely as the subdivision method (which replaces each sliced triangle with three triangles) with which it was compared (see figure on page 99). The effects of incision branching were not tested. They say that their 3-D method would not easily generalize to tetrahedral volume meshes because the Delaunay criteria would allow nearly co-planar "sliver" tetrahedra, which make for bad mesh quality. Also, face-flipping (the 3-D equivalent of edge flipping) is often not topologically possible.

Chapter 6

This chapter presents a new approach to modelling needle insertion in 2-D meshes, which is simpler than cutting. They claim their method performs comparably to previous work with 2-D needle insertion using linear elastic models, but that it also allows for nonlinear models and would more easily generalize to 3-D needle insertion (although they apparently did not actually implement the 3-D case). The mesh is refined close to the needle using edge bisection, and an iterative relaxation algorithm is used. I didn't really read this chapter very carefully, but if we ever decide we need good needle insertion in our simulation, we can look back at it.

Chapter 7

This chapter discusses some of the implementation details of their mesh representations. They use an abstract simplex data structure, maintaining an ordered sequence of vertices. Again I didn't read much in this chapter.