Tissue Cutting using Finite Elements and Force Feedback

C. Mendoza and C. Laugier

INRIA** Rhône-Alpes & GRAVIR, ZIRST 655 av. de l'Europe,
38330 Montbonnot Saint Martin, France
{Cesar.Mendoza-Serrano, Christian.Laugier}@inrialpes.fr
http://www.inrialpes.fr/sharp

Abstract. This paper presents a methodology to simulate 3D cuts in deformable objects. It uses an *explicit* finite element approach to simulate deformations in real-time. We use a non-linear strain tensor formulation (Green tensor) to allow large displacements. Haptics is used to allow touching sensation of the cutting procedure.

1 Introduction

Cutting is an essential component in medical simulators. Simulating cutting of biological tissues requires real-time interactions, accurate geometric models of anatomical structures and realistic dynamic models. Previous works address cutting by *removing* [1] from the simulation the elements that collide with the cutting tool or by *subdividing* [2][3] the colliding elements. Removing elements destroys the material from the virtual organ. In some cases, this is not realistic since the mass of the organ is not preserved. To increase realism, the number of simulated elements is incremented. This might cause a slow down in the simulation. On the other hand, subdivision is more realistic, but the number of simulated elements increases and therefore the simulation is slowed down as well.

In our previous works [7], we have started a new approach: *separating the elements* instead of removing or dividing them. The approach does not increment the number of elements during the simulations and preserves the mass of the organ. We implemented it in a 2D mass-spring model. Later, Nienhuys et. al. [4] has used the same idea to approach 3D cutting.

This paper is organized as follows: first we describe the physical deformable model used in section 2. Next, we present the cutting algorithm in section 3. Section 4 presents some of our results. Finally, we conclude and present future research in section 5.

** Inst. Nat. de Rech. en Informatique et en Automatique.

2 Physical Model

We used an explicit formulation of finite element methods [1][5][8] to simulate the dynamics of the biological tissue. The model is similar to the mass-spring deformable model, since each node of the tetrahedral mesh uses Newtonian laws of motion. However, the finite element method has a stronger physical and mathematical foundation and therefore, we have chosen it for our simulations. Finite element methods (FEM) partition the object into sub-elements on which the physical equations are expressed. Instead of merging all these equations in a large matrix system, an *explicit* FEM solves each element independently. The idea is to write the elastic energy of a tetrahedron as a function of the displacement of its 4 vertices. It uses the balance equation of each element to obtain the force at each node in function of the displacement of neighbor nodes. Then, instead of obtaining the equilibrium position by solving a large matrix system, we only integrate the force at each node to obtain the new position for the node. We use a non-linear Green strain tensor, ϵ, allowing *large displacements*. The Green strain is expressed by a 3 x 3 matrix. Its (i, j) coefficient is:

$$\epsilon_{ij} = (\frac{\partial \boldsymbol{x}}{\partial u_i}\frac{\partial \boldsymbol{x}}{\partial u_j} - \delta_{ij}) \tag{1}$$

where the Kronecker delta is $\delta_{ij} = 1$ if $i = j$ or zero otherwise. We assume that our material is isotropic and consider linear elasticity to link stress and strain, as follows:

$$\sigma_{ij}^{(\epsilon)} = \sum_{k=1}^{3} \lambda \epsilon_{kk} \delta_{ij} + 2\mu\epsilon_{ij}. \tag{2}$$

The material's rigidity is determined by the value of μ, and the resistance to changes in volume (dilation) is controlled by λ. The total internal force that a tetrahedron exerts on a node is [5]:

$$\mathbf{f}_{[i]}^{el} = -\frac{vol}{2}\sum_{j=1}^{4}\mathbf{p}_{[j]}\sum_{k=1}^{3}\sum_{l=1}^{3}\beta_{jl}\beta_{ik}\sigma_{kl} \tag{3}$$

where vol is the volume of the tetrahedron, $\mathbf{p}$ the position of the nodes of the tetrahedron in the world coordinates and β, the inverse barycentric matrix that links the world positions to the material coordinates. The total internal force acting on the node is obtained by summing the forces exerted by all elements that are attached to the node. Finally, we use a modified-Euler scheme to integrate the dynamics of each node.

3 3D volumetric cutting algorithm

Once a collision detection has been detected between the cutting tool and the object, we follow the next steps to carry out cutting phenomena: (1) a geometric and physical criteria, (2) select and separate tetrahedrons, (3) local remeshing and (4) force feedback.

3.1 Geometric and physical criteria

Geometric criteria: We first determine if the user displacements on the surface of the object corresponds to a *cutting attempt* or not.

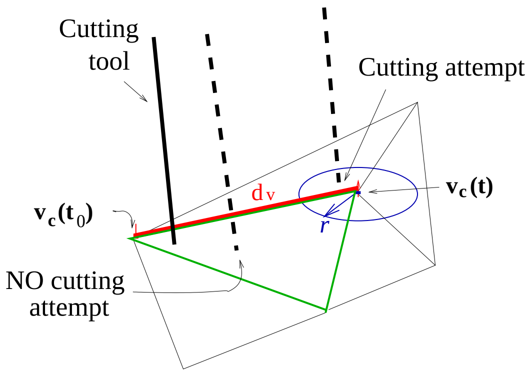

Fig. 1. Determining cutting attempts.

Let $C_p(t)$ be the colliding point at time t between the virtual tool and a facet on the surface of the object. Let $v_c(t)$ be the closest vertex to $C_p(t)$ and t_0 the moment of the first contact. Define a neighborhood $\uplus$ around the vertex $v_c(t)$. We consider a *cut attempt* if:

1. $v_c(t) \neq v_c(t_0)$
2. $Cp(t) \in \uplus$

The first condition states that the closest vertex to the colliding points at times t and t_0 must be different. The second condition avoids degenerated cuts due to small movements (e.g. a very small displacement of the tool in the middle of the facet may satisfy the first condition). To cut, the user is forced to execute larger displacements by constraining the tool to lie on the neighborhood $\uplus$, see figure 1. For simplicity, we have chosen the neighborhood to be a circular region with radius r. The value of r determines the size of $\uplus$. Since the facets of a mesh are, in general, of different sizes and forms, the value of r must be computed as a function of the size of the current colliding facet. Thus,

$$r = \alpha d_v \tag{4}$$

where d_v is the distance between $v_c(t)$ and $v_c(t_0)$, see figure 1. This distance changes depending on the colliding facet. The parameter α determines the size of the neighborhood.

Physical criteria A cut attempt is not enough to break apart the object. Some physical aspects, that take into account the physical interaction between the object and the cutting tool, have to considered. To do that, we consider the internal behavior of the object when it is subjected to external loads produced by the tool. According to fracture mechanics, an object may be broken due to two different type of failures: (a) *Tensile failure:* This corresponds to loading *normal* to the failure surface. If the failure is produced by pushing rather than pulling then we can have a *compressive failure.* (b) *Shear failure:* This corresponds to loading *tangential* to the failure surface. We analyze the forces that cause these failures.

First, note that during the contact, the internal forces equilibrate the external load produced by the tool. The internal force distribution can be represented by an equivalent set of resultants, F, and moments, M, see figure 2. From classical

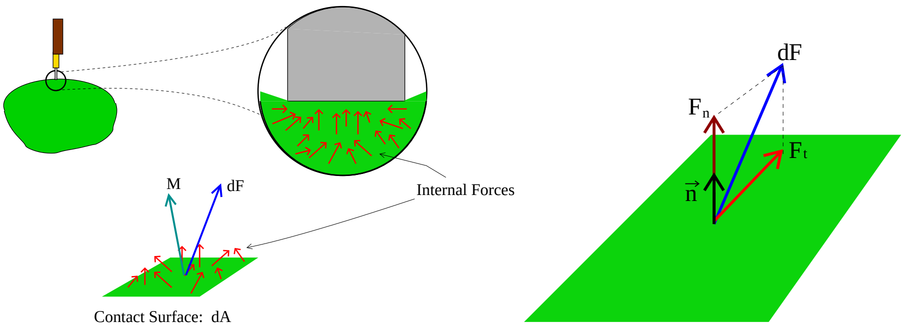

Fig. 2. Internal forces during contact between cutting tool and object

mechanics [9], the traction, Tr, provides a measure of the direction and intensity of the loading at a given point and it is defined as:

$$Tr = \lim_{dA \to 0} \frac{dF}{dA}. \quad (5)$$

Decomposing the force into normal and tangential components, see figure 2, and introducing a sharpness factor, κ, for the cutting tool, we have a *cutting traction* vector:

$$Tc = \frac{1}{\kappa} \left[(\lim_{dA \to 0} \frac{|F_n|}{dA}) \boldsymbol{n}_1 + (\lim_{dA \to 0} \frac{|F_t|}{dA}) \boldsymbol{n}_2 \right] \quad (6)$$

where n_1 is the normal to the plane and n_2 is the tangent to that same plane,(i.e. a normal in another perpendicular plane). Most of the measurable parameters available in the literature are given using the *fracture toughness*, K_I, of the material which is the critical stress intensity required to produce a failure in a material. Therefore, we put the *cutting traction vector*, T_c, in terms of the stress:

$$T_c = \frac{1}{\kappa}(\sigma \boldsymbol{n}_1 + \tau \boldsymbol{n}_2). \quad (7)$$

where σ is the *normal* stress and τ the *shear stress.* In the 3D case, T_c takes the following form:

$$\begin{bmatrix} t_{e_x} \\ t_{e_y} \\ t_{e_z} \end{bmatrix} = \frac{1}{\kappa} \begin{bmatrix} \sigma_{e_x x} & \tau_{e_x y} & \tau_{e_x z} \\ \tau_{e_y x} & \sigma_{e_y y} & \tau_{e_y z} \\ \tau_{e_z x} & \tau_{e_z y} & \sigma_{e_z z} \end{bmatrix} \begin{bmatrix} n_{e_x} \; n_{e_y} \; n_{e_z} \end{bmatrix}. \tag{8}$$

where $\boldsymbol{n}_i$ is the normal of each plane of the infinitesimal cube. For simplicity, take Γ as the set of normal and shearing stresses. The object is broken when the maximum stress takes a value greater than the material toughness, K_I. From classical mechanics, the maximum shearing stress is computed using the eigenvalues, σ_1,σ_2 and σ_3 of Γ.

$$\tau_{max} = \frac{1}{2} max\{|\sigma_1 - \sigma_2|, |\sigma_1 - \sigma_3|, |\sigma_2 - \sigma_3|\}. \tag{9}$$

and the maximum normal stress, σ_{max} is the greatest eigenvalue of Γ. Finally, we define our *cutting stress*, σ_c as:

$$\sigma_c = \frac{1}{\zeta} \min\,(\sigma_{max}, \tau_{max}). \tag{10}$$

where $\zeta \in [0.1\ 1]$ is a parameter representing the *damage* in the cutting area. Finally, a cut is produced if a *cutting attempt* has occurred and if

$$\sigma_c \geq K_I \tag{11}$$

where K_I is the material toughness of the object.

3.2 Select and separate tetrahedrons

To select the tetrahedrons which need to be separated to broken the object we consider a *cutting line* on the surface of the object. This cutting line is given by the set of vertices selected as follows: it starts at $v_0 = v_c(t_0)$, the closest vertex to the previous colliding point at t_0, it continues to $v_1 = v_c(t)$, the closest vertex to the current colliding point, such that $v_c(t_0) \neq v_c(t)$. The ending vertex, v_2 of the cutting line is the one that best fits the profile of the cut. We consider that the *cut attempt* is executed in the direction, $\boldsymbol{s}_1$, from $C_p(t_0)$ to $C_p(t)$ and that the cut is as straight as possible. We define $\boldsymbol{s}_i$ as the vectors from the possible projected vertices to $C_p(t)$; v_2 will be chosen as the vertex v_i whose vector $\boldsymbol{s}_i$ is minimum with respect to $\boldsymbol{s}_1$:

$$v_2 = v_i \quad \text{such that} \quad \min\,\{\angle(\boldsymbol{s}_1, \boldsymbol{s}_i)\}. \tag{12}$$

Let $\boldsymbol{s}_1$ be the vector from v_0 to v_1 and $\boldsymbol{n}$ the normal to the facet as shown in figure 3. Define P as the plane spanned by $\boldsymbol{n}$ and $\boldsymbol{s}_1$. Set T as the set of tetrahedrons, e^T, sharing the vertex v_1. Then, from e^T, we separate the tetrahedrons, that are in one side of the plane from those that belongs to the other side of the

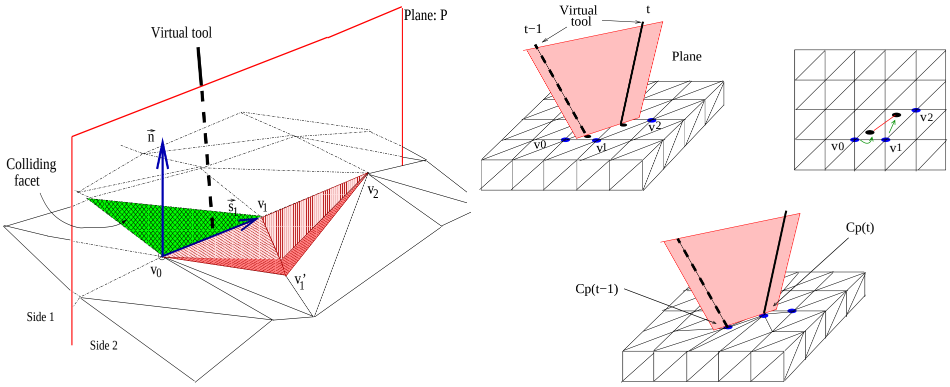

Fig. 3. (left)A set of tetrahedra is put in one side of the plane (Side 1) and another set is put in the other (Side 2). The plane is spanned by vectors $\boldsymbol{s}_1$ and $\boldsymbol{n}$. (right) Repositioning the vertices of the tetrahedrons to fit the cut profile

plane, by only splitting v_1. When a tetrahedron, e^T, is divided by the plane, P, the tetrahedron will lie in the side where its furthest vertex lies.

Separating tetrahedrons may create singularities or zero area joints (e.g. tetrahedrons connected only by one vertex). Our data structure, based in a *abstract simplicial complex* K, let us identify these singularities efficiently.

3.3 Local remeshing

To reflect the cut, we reposition the vertices of the cutting line by translating v_0 and v_1 to $C_p(t-1)$ and $C_p(t)$ respectively. The vertex v_2 is not moved until the next cutting step, when it will renamed as v_0, see figure 3. We update the β matrix and the volume of the involved tetrahedrons to keep the physical validity of the model.

3.4 Force Feedback

Haptic interaction was included to increase realism. We solve the different rate frequency problem between the physical (about 20 Hz) and the haptic simulations (about 1 KHz) by separating the haptic and the simulation loop and linking them by a *buffer model* as we have done in [6]. The buffer model computes the haptic forces.

We also model the sensation felt by the user when the material is broken. When a cut is executed in the physical simulation, the haptic rendering *switches* from the buffer model to a model that simulates a haptic cut. We use the curve of the force obtained during fracture proposed by [10]. From it, we propose to compute the haptic force as:

$$F_{haptic}(I) = F_c e^{-\xi I} \tag{13}$$

where F_c is the force of the buffer model at the moment of switching, I is an iteration counter in the haptic loop and ξ is a parameter indicating the lost of energy during the cut. The value of ξ is calculated empirically. The value of I is incremented at each haptic interaction and set it to zero when a cut task has finished.

4 Results

In figure 4 we show a physical simulation of an object representing a human knee graft ligament. It is composed of 100 tetrahedrons simulated using explicit finite elements and non-linear Green formalism ($\lambda = 140000$, $\mu = 11000$, $\psi = 10$, $\phi = 80$, sharpness $\kappa = 0.0004$, damage $\zeta = 1$). A PHANToM device is coupled to the simulation to render the sensation of touching and cutting the object. We use a 800 MHz. processor and we obtain 30 Hz. for the physical and graphical simulation. The haptic rendering reaches the 1000 Hz and present a stable behavior. Large deformations presented self-collisions.

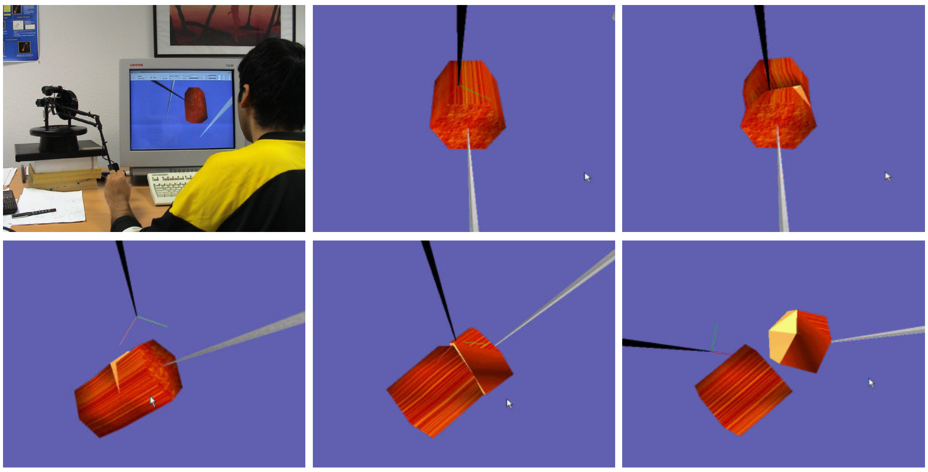

Fig. 4. Cutting a human ligament using a force feedback device

The haptic sensation of the cut has been largely influenced by the visual rendering.

5 Conclusion

We have proposed a 3D cutting algorithm for biological tissue using explicit finite elements and a non-linear strain tensor (Green). The algorithm uses a physical criteria based in a stress approach and the interpretation of the user movements to break the virtual object. The data structure allows efficient detection of zero area links between the tetrahedrons. We have integrated a stable haptic approach to give the user a touching sensation of the cut.

The implementation of our approach is a proof-of-concept of this technique. This work has opened a new road of research: object self-collisions caused during a cut.

6 Acknowledgements

This work has been supported by the National Council of Science and Technology of Mexico (CONACYT) and by the National Research Institute in Computer Science and Control (INRIA), France.

References

1. S. Cotin, H. Delingette and N. Ayache, A Hybrid Elastic Model allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation, The Visual Computer 2000, vol 16, no 8, pages 437–452.
2. Bielser, D., Maiwald, V. A. and Gross, M. H., Interactive Cuts through 3-Dimentional Soft Tissue, EUROGRAPHICS'99, vol 18, pages 31-38, 1999.
3. Mor, A. and Kanade, T., Modifying Soft Tissue Models : Progressive Cutting with Minimal New Element Creation, MICCAI, Medical Image Computing and Computer Assisted Intervention, oct. USA, 2000.
4. Nienhuys, H. and Vanderstappen, F., Combining finite element deformation with cutting for surgery simulations, EUROGRAPHICS, 2000.
5. James O'Brien and Jessica Hodgins , Graphical Models and Animation of Brittle Fracture , SIGGRAPH Conference Proc. 1999.
6. Mendoza, C. and Laugier, C., Realistic Haptic Rendering for Highly Deformable Virtual Objects, IEEE Virtual Reality (VR), Yokohama Japan, march 2001.
7. Mendoza, C., Laugier, C. and Boux-de-Casson, F., Virtual Reality Cutting Phenomena using Force Feedback for Surgery Simulations, Interactive Medical Image Visualization and Analisys, MICCAI, Holland,2001.
8. Debunne, G., Desbrun, M. and Cani, M. and Barr, A. , Dynamic Real-Time Deformations using Space and Time Adaptive Sampling, Computer Graphics, SIGGRAPH, USA ,2001.
9. Fung, Y.C., A First Course in Continuum Mechanics., Prentice-Hall, Englewood Cliffs, N.J., 1969
10. Mahvash, M. and Hayward, V. , Haptic Rendering of Cutting: A Fracture Mechanics Approach, Haptics-e, http://www.haptics-e.org, vol. 2, no. 3, nov 2001.