Vertex-preserving Cutting of Elastic Objects

Megumi Nakao\textsuperscript{*}
Nara Institute of Science and Technology

Kotaro Minato
Nara Institute of Science and Technology

Naoto Kume
Kyoto University

Shin-ichiro Mori
University of Fukui

Shinji Tomita
Kyoto University

\textbf{Abstract}

This paper proposes vertex-preserving cutting methods on finite element models for interactive soft tissue simulation. Unlike existing methods, we aim to shape variety of incisions using only initial vertices of tetrahedral meshes. Neither tetrahedral decomposition nor vertex creation is used. The number of vertices is preserved. This avoids increase of computation cost as well as allows fast update of physical status of finite element models. To preserve 3D shape and sharp feature of initial meshes through on-the-fly mesh modification, constraints are introduced to the topological update scheme. In our model, the size of stiffness matrix is constant. Our framework efficiently simulates several varieties of smooth incisions with sufficient quality for surgical simulation, and also achieves interactive performance in complex meshes with thousands of elements.

Keywords: Soft Tissue Cutting, Finite Element Modeling, Interactive Simulation

\section{1 Introduction}

Virtual cutting of deformable, elastic objects is one of key elements for interactive application like virtual surgery. The simulation of cutting has two main aspects: geometric, topological aspect regarding how to define 3D shape of incision, and physical aspect concerned with deformation and reaction force. Tetrahedral meshes are popular representation to model volumetric 3D geometry of objects. Removal of tetrahedral elements is a simple approach to model incision and new physical status after cutting elastic objects\cite{1}\cite{2}. Since only removal of elements have problems of yielding visual artifacts and of decreasing total volume of the models, topological adaptation is required. One approach for topological update is subdivision of tetrahedral meshes\cite{1}\cite{2}, which defines incision using newly created small elements. Splitting mesh on the boundary of elements\cite{4} is another approach. Combined methods have been well explored\cite{5}. However, so far, most algorithms for topological change are complex and increase their computation cost in progress of cutting manipulation. This drawback is mostly derived from procedural new vertex creation which depends on initial mesh topology.

This paper presents new vertex-preserving cutting methods of finite element models for interactive surgical simulation. The key concept is to model incision geometrically and physically while preserving the number of vertices. Neither tetrahedral decomposition nor vertex creation is used. This avoids increase of computation time and allows fast update of physical status of the model. To preserve 3D shape and sharp feature of initial meshes through on-the-fly mesh modification, constraints are introduced to the topological update scheme. Our algorithms efficiently simulate several varieties of smooth incisions, and achieved interactive performance in complex meshes with thousands of elements.

\textsuperscript{*}e-mail: meg@is.naist.jp

\section{2 Vertex-Preserving Cutting}

\subsection{2.1 Overview}

We model the blade of the scalpel as a line segment $PQ$, and the elastic object as a tetrahedral mesh $(V, E)$, with vertices $V$ and tetrahedral elements $E$. Figure 1 briefly explains our basic concept using 2D outline of the process. When a blade path is given as the dotted line shown in (a), the vertices of the intersected elements are projected onto the path. Two parallel lines (surfaces in the 3D virtual space) composed by relocated vertices are used to shape the cut surfaces. Some vertices are constrained to preserve 3D shape and sharp feature of initial meshes.

After the vertices have been relocated, removing elements that intersect the blade path makes it possible to model the tiny space between cut surfaces and simulate the physical behavior of the incision. In the finite element modeling, the stiffness matrix must be updated through relocation of vertices and the elimination of elements. Note that with methods that use element decomposition to represent the cutting process, the size of the stiffness matrix increases. However, in this model the size is constant. We utilize this feature for solving linear equation in finite element formulation, and present fast deformation algorithms for time-varying stiffness matrix.

\subsection{2.2 Cut Surface Generation}

This section introduces the proposed topological update algorithms for vertex-preserving cutting. Figure 2 illustrates the basic process for generating a new geometry of incision: cut surfaces by relocation of existing interior vertices of three tetrahedral elements. Firstly, we define the blade at time $t$ by $P_tQ_t$. The movement of the blade from time $t-1$ to $t$ forms square-shaped surface trajectory $S_t$. This surface is called sweep surface as described in [5]. When the sweep surface intersects an edge of a tetrahedral element, the element is thought to be cut, and cut surfaces are configured through the relocation of vertices. In order to handle progressive cut, at least one vertex of the cut element should be updated onto the blade $P_tQ_t$. When a tetrahedral element is cut, the vertices of the element are projected orthogonally onto the sweep surface $S_t$. 

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{fig1.png}
\caption{Modeling of incision through vertex relocation and element removal for vertex-preserving finite element modeling. The progressive cutting is supported by update of vertices around the blade. Some vertices are constrained to preserve 3D shape and sharp feature of initial meshes.}
\end{figure}
This update for intersected tetrahedra make possible to define cut surfaces.

\[ \Delta v_i = \{ n_t \cdot (v_i - p_t) \} n_t \]  

where \( n_t \) is the normal vector of the sweep surface \( S_t \).

Vertices that are updated outside of sweep surfaces \( S_0, S_1, \ldots, S_t \) generate visual artifact that the shape of incision does not depend on the movement of the blade. Therefore, such vertices are again relocated by projecting them onto the boundary of given sweep surfaces as shown in Figure 2(d). The boundary of sweep surfaces are composed by several line segments \((p_k, p_{k-1}) (k = 1, \ldots, t)\) and the blade position \((p_t, q_t)\) at time \( t \). We obtain the minimum distance between the vertex \( v' \) and one of boundary line segments, and relocate the vertex to the nearest boundary of sweep surfaces. For example, in case of projecting the vertex onto the blade \((p_t, q_t)\), additional update \( \Delta v'_i \) is defined as the following equation:

\[ \Delta v'_i = \frac{((q_t - p_t) \cdot (v'_i - p_t))}{|q_t - p_t|^2} (q_t - p_t) \]  

The surface vertex must be updated to a new position that approximately forms initial surface around it. Therefore, our model constrains relocation of the surface vertex on its tangent plane. Intersection between the sweep surface \( S_t \) and the tangent plane \( S_i \) of the vertex \( v_i \) defines an intersection line. The surface vertex \( v_i \) is updated to the intersection line using its perpendicular.

3 RESULTS

Figure 3 demonstrates interactive cutting and deformation examples using a plate model that represents 200mm × 200mm × 5mm soft tissue or skin. The bottom edge of this model is fixed. Our demonstration system allows users to cut into the model at any position and to deform it interactively. Deformation of incision by different external force and curved incision can be also simulated. Relocation of vertices and additional update onto the boundary of the sweep surfaces form smooth cut surfaces which along with the blade trajectory. Cutting two organ models were simulated using texture-based volume rendering as shown in Figure 4. Cutting procedure is interactively simulated with sufficient visual quality for surgery simulation.

4 CONCLUSION

This paper proposed vertex-preserving cutting on finite element models for interactive surgical simulation. The key concept was to model variety of incisions while preserving the number of vertices. Methods for topology modification by vertex relocation with constraint and fast finite element computation were presented. Our algorithms efficiently simulate several varieties of smooth incisions, and achieved interactive performance in complex meshes with thousands of elements. We continue to improve the cutting framework and to develop interactive preoperative rehearsal system.

REFERENCES