This is the authors version of an article published as :
H. Jung, S. Cotin, C. Duriez, J. Allard, D. Y. Lee, "High Fidelity Haptic Rendering for Deformable Objects Undergoing Topology Changes," in *EuroHaptics 2010*, LNCS 6191, pp. 262–268, 2010.

# High Fidelity Haptic Rendering for Deformable Objects Undergoing Topology Changes

Hoeryong Jung[1], Stephane Cotin[2], Christian Duriez[2], Jeremie Allard[2] and Doo Yong Lee[1]

[1] Department of Mechanical Engineering, KAIST, 335 Gwahangno, Yuseong-gu Daejeon, 305-701, Korea
[2] INRIA Lille-North Europe, 59650 Villeneuve d'Ascq, France

**Abstract.** The relevance of haptic feedback for minimally invasive surgery has been demonstrated at numerous counts. However, the proposed methods often prove inadequate to handle correct contact computation during the complex interactions or topological changes that can be found in surgical interventions. In this paper, we introduce an approach that allows for accurate computation of contact forces even in the presence of topological changes due to the simulation of soft tissue cutting. We illustrate this approach with a simulation of cataract surgery, a typical example of microsurgery.

**Keywords**: haptics, biomechanics, simulation, surgery, soft tissue, cutting.

## 1 Introduction

High fidelity haptic rendering of deformable objects undergoing topology changes is a fundamental requirement in surgery simulation, in particular for the simulation of dissection or other processes involved in the separation of soft tissues. However, topological changes introduce several challenges, related to re-meshing of the domain, real-time simulation of the deformation, contact computation, and haptic rendering. Several approaches have been proposed for re-meshing the domain while maintaining a relatively good mesh quality [1]. Considering the deformation aspect, handling topological changes essentially requires the update of the stiffness matrix (assuming the deformation is modeled using a finite element approach). As most recent real-time deformation methods are based on non-linear models (geometrical and/or material) [2][3], which require an update of the stiffness matrix at every time step, the handling of topological changes adds little overhead to the process. The main difficulty comes from haptic rendering, in particular in applications where multiple complex contacts occur. Examples of such applications are common in surgery simulation. In this paper we consider the particular case of cataract surgery, an example of microsurgery where haptics can play a key role during training. Approaches for providing haptic feedback in the case of soft tissue interactions often rely on a simple contact point rather than an actual area of contact, and rarely account for friction. While such simplifications can produce plausible results in certain contexts, it can lead to incorrect results in some applications, such as surgery, where multiple anatomical structures can be in contact simultaneously.

This paper introduces a new method for providing high-fidelity force feedback during soft tissue manipulation involving topological changes. The proposed approach can be seen as an extension of previous works on contact modeling [4] or haptics [5]. These works rely on the computation of a compliance matrix and resolution of a linear complementary problem (LCP). Although more computationally expensive than other approaches, such a method guarantees that all interpenetrations are solved at the end of each time step in the simulation while providing stable and efficient haptic feedback. The following sections present our contribution. Section 2 introduces the surgical context and background material regarding our simulation method. Section 3 describes the main contribution of this paper, i.e. an efficient numerical approach for updating a compliance matrix even during topological changes. Finally, section 4 presents some results in the context of cataract surgery simulation and section 5 discusses possible future directions.

## 2 Simulation Overview

### 2.1 Cataract Surgery

Cataract surgery, along with glaucoma surgery and vitro-retinal surgery, is one of the more frequent and difficult procedures of ophthalmic surgery. A cataract is an opacification in the natural lens of the eye. It represents an important cause of visual impairment and, if not treated, can lead to blindness. Cataract surgery has made important advances over the past twenty years, but remains technically challenging. The objective of our simulation is to reproduce with great accuracy the three main steps of cataract surgery: capsulorhexis, phacoemulsification and implantation of an intraocular lens. In this paper we focus on the phacoemulsification, which consists in using a microscopic surgical device which tip vibrates at an ultrasonic frequency (40,000 Hz) to locally fragment the natural lens of the eye. This progressively leads to the separation of the lens into several small parts, which can then be sucked out of the eye through a small incision.

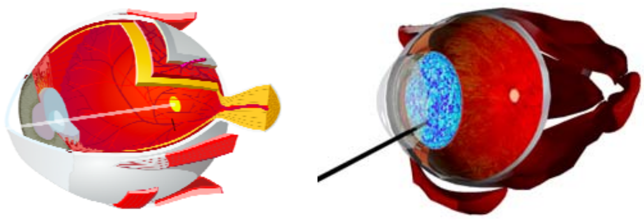


**Fig.1**. *Left*: detailed anatomy of the eye, with the retina (in red), the cornea, and the lens. *Right*: the different anatomical models used in our simulation, and an illustration of the phacoemulsification device interacting with the tetrahedral mesh of the lens (the particular rendering of the lens is only for the purpose of showing the size of the tetrahedral elements).

### 2.2 Deformable model

The dynamics of the eye lens is modeled using the FEM based differential equation:

$$M\dot{v} = F(u, v) + R \tag{1}$$

Where v and u are velocity and displacement vector of the nodes. M is the mass matrix, R the vector of contact forces and F(u,v) is the sum of external and internal forces. In our case, we model the eye lens as a linear elastic material, which undergo large displacements. To model this behavior we rely on a co-rotational finite element method [3]. At each time step, we compute a linearization of the finite element model: $\mathrm{F}(\mathrm{u}+d\mathrm{u},\mathrm{v}+d\mathrm{v})=\mathrm{F}(\mathrm{u},\mathrm{v})-\mathrm{K}d\mathrm{u}-\mathrm{B}d\mathrm{v}$, where K=K(u), the tangent matrix is non constant. As we will see in the following section, we can take advantage of this particular model to optimize the computation of the compliance matrix required in the contact solver and haptic rendering loop.

### 2.3 Contact model

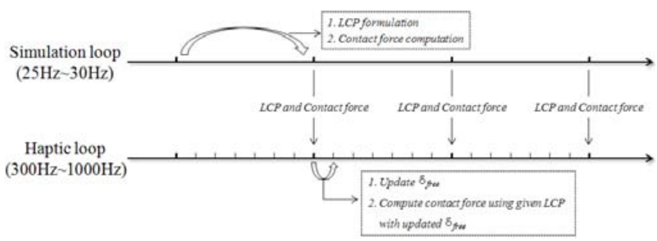
To solve for contacts occurring between the microsurgical instruments and the different anatomical structures of the eye (such as the lens, lens capsule and cornea) we rely on the method introduced in [4]. This method computes the forces required to solve all interpenetrations within a single time step, thus guaranteeing a contact-free configuration at the beginning of the next time step. The core computation of the algorithm involves solving a linear complementary problem (LCP).

$$\delta=\mathrm{HCH}^{\mathrm{T}} f+\delta^{\mathrm{free}} \qquad \text{with} \qquad 0 \le \delta \perp f \ge 0 \tag{2}$$

Where $\mathrm{C}=\left(\frac{1}{h^2}\mathrm{M}+\frac{1}{h}\mathrm{B}+\mathrm{K}\right)^{-1}$ is called the compliance matrix which needs to be recomputed and every time step. Obviously the computation of C can quickly become prohibitive when the number of nodes in the mesh increases. To solve this issue, it is possible to take advantage of the co-rotational formulation, where non-linearities are mainly related to rotations. The computation of C is only performed for the rest configuration and updated at each time step based on a local estimation of the rigid motion at each node (see [4] for more details).

### 2.4 Haptic rendering

For the haptic rendering, which requires high rate (300Hz~1000Hz) force computation, we employ multi-rate haptic rendering technique [5], separating the haptic loop from the simulation loop. At every time step of the simulation loop, we build the LCP using the given compliance matrix, and compute the contact force from the LCP. And the LCP and the contact force are transmitted to the haptic loop so that those can be used to compute inter-sampled force between time steps of the simulation loop. In the haptic loop, the interpenetration depth $\delta^{free}$ is updated using new position of haptic device, and the inter-sampled contact force is computed using




the given LCP with updated $\delta_{free}$ at every time step of haptic loop.

**Fig. 2.** Multi-rate haptic rendering technique. The inter-sampled force is computed using the LCP given by the simulation loop with updated $\delta_{free}$ at every time step of haptic loop.

# 3 Topology Modifications

One of the basic tasks in the simulation of phacoemulsification procedure is locally fragmenting the lens of the eye with the microscopic surgical device. We simulate the action of the device on the eye lens by consecutively removing tetrahedra intersected by the tip of the device.

### 3.1 System matrix modification

When the tetrahedron is removed from the current mesh of the model, the local system matrix of the tetrahedron should be subtracted from the current system matrix:

$$\left(\frac{1}{h^2}\mathbf{M}+\frac{1}{h}\mathbf{B}+\mathbf{K}\right)_{i+1} = \left(\frac{1}{h^2}\mathbf{M}+\frac{1}{h}\mathbf{B}+\mathbf{K}\right)_{i} - \mathbf{G}_e\left(\frac{1}{h^2}\mathbf{m}_e+\frac{1}{h}\mathbf{b}_e+\mathbf{k}_e\right)\mathbf{G}_e^T \quad (3)$$

Where, $\left(\frac{1}{h^2}\mathbf{m}_e+\frac{1}{h}\mathbf{b}_e+\mathbf{k}_e\right)$ is the local system matrix of the removed tetrahedron, and $\mathbf{G}_e$, $\mathbf{G}_e^T$ are the globalization matrices which map the rows and the columns of the local system matrix to the global system matrix.

### 3.2 Compliance matrix update algorithm

As stated in section 2.3, the contact model relies on a pre-computation of the compliance matrix on the rest shape configuration. But the removal of tetrahedron makes this matrix no more valid as all values are affected. Since an element removal alters only a few entries of the original system matrix, low rank inverse update algorithms, such as Sherman-Morrison-Woodbury formula [7], can be a reasonable solution for real-time computation [6]. The modified compliance matrix can be computed from the current compliance matrix as follows.

$$\mathrm{C}_{i+1} = \mathrm{C}_i - \mathrm{C}_i\mathrm{G}_e\left(\mathrm{S}_e^{-1}+\mathrm{G}_e^T\mathrm{C}_i\mathrm{G}_e\right)^{-1}\mathrm{G}_e^T\mathrm{C}_i = \mathrm{C}_i - \mathrm{P}_i\mathrm{Q}_i^{-1}\mathrm{P}^T \quad (4)$$

Where $\mathbf{S}_e = -\left(\frac{1}{h^2}\mathbf{m}_e+\frac{1}{h}\mathbf{b}_e+\mathbf{k}_e\right)$, $\mathbf{P}_i = \mathbf{C}_i\mathbf{G}_e$ and $\mathbf{Q}_i = \mathbf{S}_e^{-1}+\mathbf{G}_e^T\mathbf{C}_i\mathbf{G}_e^T$. The matrix $\mathbf{P}_i$ and $\mathbf{G}_e^T\mathbf{C}_i\mathbf{G}_e^T$ are simply constructed by extracting corresponding rows and columns of the matrix $\mathbf{C}_i$. While the above formula is computationally less expensive than re-computing the inverse of the modified system matrix from scratch, the matrix multiplication $\mathbf{P}_i\mathbf{Q}_i^{-1}\mathbf{P}^T$ has a complexity of $O(n^2)$, where $n$ is the dimension of the compliance matrix, which is still too demanding to be performed in real-time.

Building the LCP used to compute the contact force in the haptic loop, only involves the part of the compliance matrix related to the nodes in contact. Consequently, we do not have to compute all entries of the compliance matrix, and it leads us to update only the necessary part of the compliance matrix rather than complete update. Once we obtain the matrix $\mathbf{Q}_i^{-1}$, one entry of the compliance matrix $\mathbf{C}_{i+1}$ can be computed with ($m^2$+$m$) operations, where $m$ is the dimension of $\mathbf{Q}_i$. Even though $m$ is a very small number (12 in case of one tetrahedron removal), we should minimize the number of operations because ($n$ x $m$) entries of $\mathbf{C}_{i+1}$ should be computed in the next modification to construct the matrix $\mathbf{P}_{i+1}$. We diagonalize the matrix $\mathbf{Q}_i$ using $\mathbf{Q}_i = \mathbf{W}_i \mathbf{\Lambda}_i \mathbf{W}_i^T$ ( $\mathbf{\Lambda}_i$ is a diagonal matrix). It reduces the number operations to *2m*.

$$\begin{aligned}\mathbf{C}_{i+1} &= \mathbf{C}_i - \mathbf{P}_i \left(\mathbf{W}_i \mathbf{\Lambda}_i \mathbf{W}_i^T\right)^{-1} \mathbf{P}_i^T \\ &= \mathbf{C}_i - \mathbf{L}_i \mathbf{\Lambda}_i^{-1} \mathbf{L}_i^T\end{aligned} \tag{5}$$

Where $\mathbf{L}_i = \mathbf{P}_i \mathbf{W}_i$. Whenever tetrahedra are removed during the simulated phaco-emulsification procedure, we compute the matrices $\mathbf{L}_i$ and $\mathbf{\Lambda}_i^{-1}$, and save them as follows.

$$\mathbf{C}_k = \mathbf{C}_0 - \mathbf{L}_0 \mathbf{\Lambda}_0^{-1} \mathbf{L}_0^T - \mathbf{L}_1 \mathbf{\Lambda}_1^{-1} \mathbf{L}_1^T - \cdots - \mathbf{L}_{k\text{-}1} \mathbf{\Lambda}_{k\text{-}1}^{-1} \mathbf{L}_{k\text{-}1}^T \tag{6}$$

Where $\mathbf{C}_0$ is the initial compliance matrix and *k* is the number of modification. One entry of the modified compliance matrix whose row and column index is *i*, *j*, is computed quickly as follows.

$$\mathbf{C}_k(i,j) = \mathbf{C}_0(i,j) - \sum_{q=0}^{k\text{-}1} \left( \sum_{l=1}^{m} \mathbf{L}_q(i,l) \mathbf{L}_q(j,l) \mathbf{\Lambda}_i^{-1}(l,l) \right) \tag{7}$$

## 4 Results

We have evaluated our method in the framework of the cataract surgery simulation. The experiments were performed on a 2.33GHz Xeon processor with 3GB of memory. Fig. 3 shows a simulation of the phacoemulsification step of the cataract surgery, where the eye lens is fragmented by a microsurgical device. The eye lens model consists of 4862 tetrahedra and 1113 nodes.

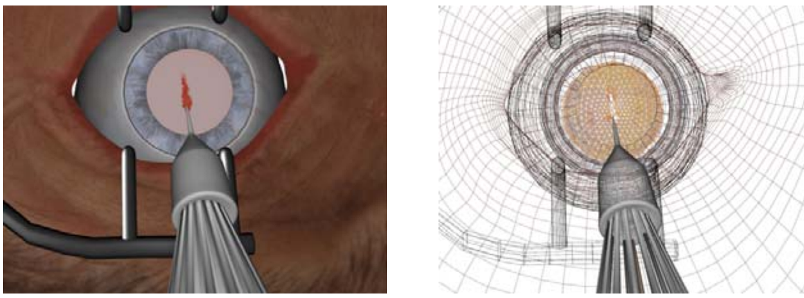


**Fig. 3.** Cataract surgery simulation: the eye lens is progressively fragmented using a phacoemulsification device while realistic visual and haptic feedbacks are provided.

We measured the computation time spent to update the compliance matrix, and compared it with the previous method which completely updates the compliance matrix. As shown in Fig. 4, the proposed algorithm achieves better performance than the previous method when the number of removed elements is small, however, the computation time linearly increases as the number of removed elements increase. The growth in the computation time is caused by construction of the matrix $\mathbf{P}_k = \mathbf{C}_k \mathbf{G}_k$. The computation time required to construct $\mathbf{P}_k$ is linearly dependent on the number of modification as shown in equation (8). However when the number of removed elements per time step remains small, we obtain fast update.

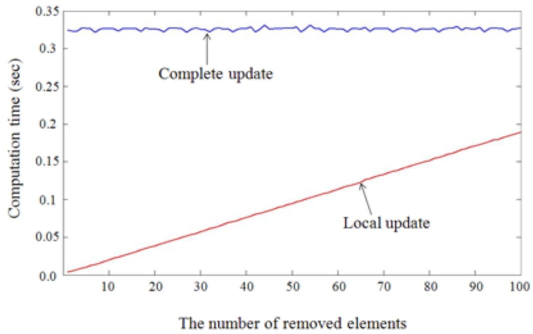



**Fig. 4.** Computation time necessary to update the compliance matrix as a function of the number of removed elements

## 5 Conclusion

In this paper we introduce a numerical technique for accurately computing contact forces during soft tissue manipulation, even in the presence of topological changes. The method is based on a local update of the compliance matrix, compatible with geometrically non-linear elastic models. Our approach allows for accurate contact force computation and realistic haptic rendering. Future work will be directed toward more precise cutting simulation involving element subdivision. While element removal only induces the modification to the system matrix, subdivision induces an extension of the matrix dimension, making it more difficult to update the compliance matrix.

## Acknowledgement

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. R01-2007-000-11353-0) and the Brain Korea 21 Project in 2009.

## References

1. Sifakis, E., Der, K. G. and Fedkiw, R.: Arbitrary Cutting of Deformable Tetrahedralized Objects. In: Proceeding of Eurographics, pp. 73--80. ACM, San Diego (2007)
2. Comas, O., Taylor, Z., Allard, J., Ourselin, S., Cotin, S. and Passenger, J.: Efficient nonlinear FEM for soft tissue modelling and its GPU implementation within the open source framework SOFA. In: Proceedings of ISBMS, pp. 28--39. Springer, London (2008)
3. Felippa, C.A.: A systematic approach to the element independent corotational dynamics of finite elements: Technical Report CU-CAS-00-03, Center for Aerospace Structures (2000)
4. Saupin, G., Duriez, C., Cotin, S. and Grisoni, L.: Efficient Contact Modeling using Compliance Warping. In: Computer Graphics International Conference, 2008
5. Saupin, G, Duriez, C. and Cotin, S.: Contact Model for Haptic Medical Simulations. In: Proceedings of ISBMS, pp. 157--165. Springer, London (2008)
6. Lee, B., Popescu, D. C., Joshi, B., and Ourselin, S.: Efficient Topology Modification and Deformation for Finite Element Models Using Condensation. In: Medicine Meets Virtual Reality 14, pp. 299--304. IOS Press, Los Angeles (2006)
7. Sherman, J. and Morrison, W. J.: Adjustment of an Inverse Matrix Corresponding to Changes in the Elements of a Given Column or a Given Row of the Original Matrix. Annals of Mathematical Statistics. 20, 621 (1949)