Anisotropic Elasticity and Force Extrapolation to Improve Realism of Surgery Simulation

Guillaume Picinbono Jean-Christophe Lombardo Hervé Delingette Nicholas Ayache

Epidaure Project
I.N.R.I.A.
06902 Sophia-Antipolis Cedex, BP 93, France
http://www-sop.inria.fr/epidaure/ Nicholas.Ayache@sophia.inria.fr

Abstract

In this paper, we describe the latest developments of the minimally invasive hepatic surgery simulator prototype developed at INRIA. The goal of this simulator is to provide a realistic training test-bed for performing laparoscopic procedures. Therefore, its main functionality is to simulate the deformation and cutting of tri-dimensional anatomical models with the help of two virtual laparoscopic surgical instruments. Throughout this paper, we present the general features of the simulator including the implementation of different bio-mechanical models based on linear elasticity and finite element theory and the integration of two force-feedback devices in the simulation platform. More precisely, we describe two new important developments that improve the overall realism of the simulator. First, we can create bio-mechanical models that include the notion of anisotropic deformation. Indeed, we have generalized the linear elastic behavior of anatomical models to "transversally isotropic" materials, i.e. materials having one privileged direction of deformation. The second improvement is related to the problem of haptic rendering. Currently, we are able to achieve a simulation frequency of 25Hz (visual real-time) with anatomical models of complex geometry and behavior. But to achieve a good haptic feedback requires a frequency update of applied forces typically above 300Hz (haptic real-time). Thus, we propose a force extrapolation algorithm in order to reach haptic real-time.

1 Introduction

A major and recent evolution in abdominal surgery has been the development of laparoscopic surgery. In this type of surgery, abdominal operations such as hepatic resection are accomplished through small incisions. A video camera and special surgical tools are introduced into the abdomen, allowing the surgeon to perform a less traumatizing operation. The drawback of this technique is essentially for the surgeon who needs to learn and adapt himself to this new type of surgery. In this context, surgical simulation systems could be a great help in the training process.

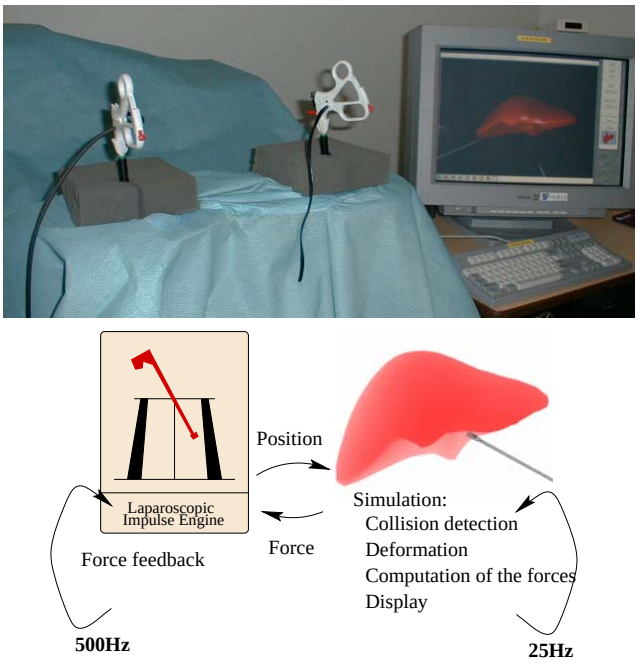


Figure 1: Surgery simulator prototype developed at INRIA

Several research groups work on surgery simulation with different methods and applications. We can cite applications to knee arthroscopic surgery [16], gynaecologic laparoscopy [24], or abdominal trauma surgery

[5].

There are several key problems in the development of a surgical simulator [2, 19]. To simulate a surgical gesture, we must be able to model the organ (both geometrically an physically), to detect and treat collisions between the organ and the virtual tools, to visualize implied deformations and to compute the reaction forces to apply to the tools.

To study these problems, INRIA gathered six teams in a joint action AISIM [1], this led to several results. To simulate the deformation of the organ, AISIM's members followed several approaches, the spring/mass models [6], the finite element method [10, 14] described in this paper, and the hierarchical multi-resolution finite volume model [12]. To evaluate the validity of the simplifications made on real-time physical models, a non-linear incompressible model with an external shell [25] fitted with bio-mechanical results was developed. A new method was designed to detect collisions between a deformable object (the organ) and a rigid tool passing though a fixed point: the idea of this LCN (Lombardo Cani Neyret [18]) method is to replace the complex algorithms usually involved in collision detection by an intensive use of the graphics hardware of modern workstations. By making an analogy between the rendering and the collision detection process and using OpenGL, we are able to detect collisions between the organ and the tool in about $0.1ms$ on an Onyx2. Finally, the rendering processes, both visual and haptic were addressed. This led to a new technique to map undistorted textures on complex geometries [21]. Our solution to the haptic rendering issue [22] is described later in this paper.

This article focuses on the developement of a laparoscopic surgery simulator prototype (figure 1) based on linear elasticity and finite element theory and permiting the use of force feedback. Here we give a general description of the two key components of the simulator: we first describe a generalization of "Tensor/Mass" models [14] which allows us to simulate real-time deformations and cutting of anisotropic materials. Then we deal with the problem of haptic rendering: while up to date physically based deformable models are able to run at sufficient frequencies to provide a visual real time (about 25Hz), this is not the case for haptic rendering. It is commonly admitted that to give good haptic sensations, the forces must be refreshed at more than 300Hz for soft objects. We propose to extrapolate the forces produced by the simulation at a visual rate to feed the force feedback device at the appropriate frequency (typically 500Hz).

2 Deformable models

2.1 Linear elasticity

The physical behavior of soft tissue may be considered as linear elastic if its displacement and deformation remain small [15, 20] (less than 10% of the mesh size). To describe a linear elastic model, we first need to define a reference volumetric anatomical model $\mathcal{M}_{\text{initial}}$ corresponding to its rest position. Under external constraints, for instance a surgical instrument, the anatomical model $\mathcal{M}_{\text{initial}}$ is deformed. We represent the deformation of a volumetric model from its rest shape with a *displacement vector* $\mathbf{U}(x,y,z)$ for $(x,y,z) \in \mathcal{M}_{\text{initial}}$ and we write $\mathcal{M}_{\text{deformed}} = \mathcal{M}_{\text{initial}} + \mathbf{U}(x,y,z)$. The displacement vector $\mathbf{U}(x,y,z)$ has three components:

$$\mathbf{U}(x,y,z) = \begin{cases} u(x,y,z) \\ v(x,y,z) \\ w(x,y,z) \end{cases} .$$

With this displacement vector, we define the linearized *Green-St Venant strain tensor* (3×3 symmetric matrix) E by:

$$E = \frac{1}{2}\left(\nabla\mathbf{U} + \nabla\mathbf{U}^t\right). \tag{1}$$

From the principal invariants of E:

$$l_1 = tr\, E \qquad l_2 = tr\, E^2, \tag{2}$$

we can express the linear elastic energy W_{Elastic}, for homogeneous isotropic materials, by the following formula (see [11]):

$$W_{\text{Elastic}} = \frac{\lambda}{2}\,(trE)^2 + \mu\, trE^2, \tag{3}$$

where λ and μ are the *Lamé coefficients* characterizing the stiffness of the material.

Equation 3, known as *Hooke's law*, shows that the elastic energy of a deformable object is a quadratic function of the displacement vector.

2.2 Anisotropic elasticity

Isotropic behavior is overly restrictive when modeling human tissue. In fact a lot of anatomical structures such as muscles, tendons, ligaments, blood vessels, are strongly anisotropic. That is why we are particularly interested in simulating materials having a different behavior in a given direction, which are called **transversally isotropic** materials.

A.J.M. Spencer gives a very detailed theoretical description of **transversally isotropic** materials in [23]. In the literature we can find applications for modeling human organs such as the knee ligaments [26] and the eye [17].

We propose to apply the transversal isotropy to linear elasticity in the framework of real-time deformable models. For such materials, the elastic energy of equation 3 must be modified in order to account for the anisotropy. But first, we explain in more detail the meaning of the strain tensor E. We consider an elementary cube in $\mathcal{M}_{\text{initial}}$ and then look at its shape after applying the displacement $\mathbf{U}(x,y,z)$. The local cube deformation is characterized by six *components of strain* corresponding to the relative elongations ($\epsilon_x = \partial u/\partial x$, ϵ_y, and ϵ_z) in the three cube directions and the relative changes of angles ($\gamma_{xy} = \partial u/\partial y + \partial v/\partial x$, γ_{xz}, and γ_{yz}) between the cube faces.

The isotropic elastic energy of equation 3 can be written as:

$$\begin{aligned} W_{\text{Elastic}} = & \quad \frac{\lambda}{2}\left(\epsilon_x+\epsilon_y+\epsilon_z\right)^2+\mu\left(\epsilon_x^2+\epsilon_y^2+\epsilon_z^2\right) \\ + & \quad \frac{\mu}{2}\left(\gamma_{xy}^2+\gamma_{xz}^2+\gamma_{yz}^2\right). \end{aligned} \tag{4}$$

This energy is isotropic since the same weight is given to each direction of stretch and shear. For transversally isotropic materials, it is necessary to define two sets of Lamé constants:

- (λ^L, μ^L): **Longitudinal** Lamé constants in a given direction having unitary vector $\mathbf{a_0}$;
- (λ^T, μ^T): **Transverse** Lamé constants in the plane transversal to $\mathbf{a_0}$;
- $\Delta\lambda = \lambda^L - \lambda^T$ and $\Delta\mu = \mu^L - \mu^T$.

For instance, if the z axis is the direction of anisotropy $\mathbf{a_0} = (0 \quad 0 \quad 1)$, then we need to add to the isotropic energy of equation 4, the anisotropic contribution ΔW_{Aniso} defined as:

$$\begin{aligned} \Delta W_{\text{Aniso}} = & \quad \Delta\lambda\left(\epsilon_x+\epsilon_y+\epsilon_z\right)\epsilon_z+\left(\frac{\Delta\lambda}{2}+\Delta\mu\right){\epsilon_z}^2 \\ & +2\Delta\mu\left({\epsilon_z}^2+\frac{1}{4}{\gamma_{xz}}^2+\frac{1}{4}{\gamma_{yz}}^2\right). \end{aligned} \tag{5}$$

In the general case, this anisotropic elastic energy can be simply written with the introduction of two new invariants l_4 and l_5 linked to the strain tensor E and direction $\mathbf{a_0}$:

$$l_4 = {\mathbf{a_0}}^t\, E\, \mathbf{a_0} \qquad l_5 = {\mathbf{a_0}}^t\, E^2\, \mathbf{a_0}. \tag{6}$$

Then the anisotropic energy can be written as:

$$\Delta W_{\text{Aniso}} = l_4 l_1 \Delta\lambda + 2 l_5 \Delta\mu - l_4^2\left(\frac{\Delta\lambda}{2}+\Delta\mu\right). \tag{7}$$

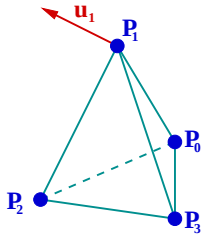
Finally, the total elastic energy of a transversally isotropic material is:

$$W_{\text{Trans-iso}} = W_{\text{Elastic}} + \Delta W_{\text{Aniso}}. \tag{8}$$

2.3 Finite element formulation

Our deformable models are based on a finite element model consisting of a conformal tetrahedral mesh. At each point $\mathbf{M}(x,y,z)$ inside tetrahedron $\mathbf{T}_i$, the displacement vector is expressed as a function of the displacements $\mathbf{U}_k$ of the vertices $\mathbf{P}_k$:

$$\mathbf{U}(x,y,z) = \textstyle\sum_{k=0}^{3} \mathbf{U}_k \Lambda_k(x,y,z)$$



where Λ_k are the barycentric coordinates of $\mathbf{M}$ in the tetrahedron.

We can then write the isotropic linear elastic energy of $\mathbf{T_i}$ as a function of its vertex displacements:

$$W_{\text{Elastic}} = \sum_{k=0}^{3}\sum_{l=0}^{3} \mathbf{U}_k^t [K_{kl}^{\mathbf{T}_i}] \mathbf{U}_l, \tag{9}$$

where $[K_{kl}^{\mathbf{T}_i}] = \lambda\alpha_k\alpha_l^t + \mu\alpha_l\alpha_k^t + \mu(\alpha_k^t\alpha_l)Id_3$ is the tetrahedron contribution to the stiffness tensor of the edge $(\mathbf{P}_k, \mathbf{P}_l)$ (or of the vertex $\mathbf{P}_k$ if $k = l$) and $\{\alpha_{\mathbf{k}} = (-1)^k/6V(\mathbf{T}_i)(\mathbf{P}_{k+1} \wedge \mathbf{P}_{k+2} + \mathbf{P}_{k+2} \wedge \mathbf{P}_{k+3} + \mathbf{P}_{k+3} \wedge \mathbf{P}_{k+1}), k = 0 - -3\}$ are the shape vectors of the tetrahedron. Likewise, we obtain a similar equation for the anisotropic part of the elastic energy:

$$W_{\text{Aniso}} = \sum_{k=0}^{3}\sum_{l=0}^{3} \mathbf{U}_k^t [A_{kl}^{\mathbf{T}_i}] \mathbf{U}_l, \tag{10}$$

$$\begin{aligned} A_{kl}^{\mathbf{T}_i} = & \Delta\lambda(\mathbf{a_0}\mathbf{a_0}^t)(\alpha_k\alpha_l^t) - \left(\frac{\Delta\lambda}{2}+\Delta\mu\right)(\mathbf{a_0}\mathbf{a_0}^t)(\alpha_k\alpha_l^t)(\mathbf{a_0}\mathbf{a_0}^t) \\ & + \frac{\Delta\mu}{2}\left[(\mathbf{a_0}\mathbf{a_0}^t)(\alpha_l\alpha_k^t) + (\alpha_l\alpha_k^t)(\mathbf{a_0}\mathbf{a_0}^t) + (\alpha_k^t\alpha_l)(\mathbf{a_0}\mathbf{a_0}^t) \right. \\ & \left. + (\mathbf{a_0}\mathbf{a_0}^t) : (\alpha_k\alpha_l^t) Id_3\right]. \end{aligned}$$

To obtain the force $\mathbf{F}_k^{\mathbf{T}_i}$ applied to the vertex $\mathbf{P}_k$ produced by tetrahedron $\mathbf{T}_i$, we derive the elastic energy

with respect to the vertex displacement $\mathbf{U}_k$:

$$\begin{aligned}\mathbf{F}_k^{\mathbf{T}_i} &= 2\sum_{l=0}^{3}[K_{kl}^{\mathbf{T}_i} + A_{kl}^{\mathbf{T}_i}]\mathbf{U}_l \\ &= [G_{kk}^{\mathbf{T}_i}]\mathbf{U}_k + \sum_{l=0;l\neq k}^{3}[G_{kl}^{\mathbf{T}_i}]\mathbf{U}_l. \qquad (11)\end{aligned}$$

Construction of the tensor:
We obtain the global force applied on vertex a $\mathbf{P}_k$ by adding the contributions of all the tetrahedra sharing this vertex:

$$\mathbf{F}_k = [G_{kk}]\mathbf{U}_k + \sum_{\mathbf{P}_l \in N(\mathbf{P}_k)}[G_{kl}]\mathbf{U}_l, \qquad (12)$$

$$G_{kk} = \sum_{\mathbf{T}_i \in N(\mathbf{P}_k)} G_{kk}^{\mathbf{T}_i} \qquad G_{kl} = \sum_{\mathbf{T}_i \in N(\mathbf{P}_k,\mathbf{P}_l)} G_{kl}^{\mathbf{T}_i}.$$

Thus, $\mathbf{F}_k$ are linear functions of the displacement vectors of each node $\mathbf{P}_k$.

2.4 Pre-computed model [10]

The most common method to solve the elasticity problem formulated in equation 12 is to build the global linear system $[K]\ \mathbf{U} = \mathbf{F}$ which gives the displacement of all the nodes as a function of boundary conditions and external forces. Unfortunately, such linear systems cannot be solve in real-time for complex objects. The main idea is to pre-compute unitary deformations of each node and to decompose, during the simulation, any deformation into a sum of unitary deformations. This method allows real-time simulation for complex objects like anatomical structures but cannot model any change of mesh topology. This drawback led us to develop a new model presented below.

2.5 Tensor/Mass model [14]

Given a tetrahedral mesh of a solid —in our case an anatomical structure— we build a data structure incorporating the notion of vertices, edges, and tetrahedra. For each vertex, we store its neighboring tetrahedra, its current position $\mathbf{P}_k$, its rest position $\mathbf{P}_k^0$, and tensor $[G_{kk}]$. For each edge, we store its two vertices as well as the tensor $[G_{kl}]$. Finally for each tetrahedron, we store its four vertices and its six edges as well as the Lamé coefficients $\lambda_k^L, \mu_k^L, \lambda_k^T, \mu_k^T$, the direction of anisotropy a_0, and the four shape vectors $\alpha_{\mathbf{k}}$.

2.5.1 Numerical integration

We use a Newtonian differential equation:

$$m_i\frac{d^2\mathbf{P}_i}{dt^2} = \gamma_i\frac{d\mathbf{P}_i}{dt} + \mathbf{F}_i \qquad (13)$$

as the equation governing the motion of our linear elastic model. This equation is related to the differential equation found in continuum mechanics [3]:

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{C}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{R}. \qquad (14)$$

Following finite elements theory, the mass $\mathbf{M}$ and damping $\mathbf{C}$ matrices are sparse matrices that are related to the stored physical properties of each tetrahedron. In our case, we consider that $\mathbf{M}$ and $\mathbf{C}$ are diagonal matrices, i.e., that mass and damping effects are concentrated at vertices. This simplification called *mass-lumping* decouples the motion of all nodes and therefore allows us to write equation 14 as the set of independent differential equations (13) for each vertex.

Furthermore, we choose an *explicit integration scheme* where the elastic force is estimated at time t in order to compute the vertex position at time $t+1$:

$$\left(\frac{m_i}{\Delta t^2} - \frac{\gamma_i}{2\Delta t}\right)\mathbf{P}_i^{t+1} = \mathbf{F}_i + \frac{2m_i}{\Delta t^2}\mathbf{P}_i^t - \left(\frac{m_i}{\Delta t^2} + \frac{\gamma_i}{2\Delta t}\right)\mathbf{P}_i^{t-1}.$$

2.5.2 Simulation of cutting

One of the basic tasks in surgery simulation consists in cutting soft tissue. With the dynamic linear elastic model, this task can be achieved efficiently. We simulate the action of an electric scalpel on soft tissue by successively removing tetrahedra at places where the instrument is in contact with the anatomical model.

When a collision between the instrument and a tetrahedron is detected, local deformation tensors associated with the tetrahedron are subtracted from the current deformation tensors at the tetrahedron edges and vertices. Since the tensors are only updated is locally, this is performed in a very efficient manner. For instance, when removing the tetrahedron T_i, 10 update operations are performed:

$$[K_{jj}] = [K_{jj}] - [K_{jj}^{T_i}] \qquad [K_{jk}] = [K_{jk}] - [K_{jk}^{T_i}].$$

Finally, we update the list of displayed triangles located on the mesh surface. By locally updating tensors, the tissue has exactly the same properties as if we had removed the corresponding tetrahedron at its rest position. Because of the volumetric continuity of finite element modeling, the tissue deformation is realistic during the cutting.

2.6 Results

We present two results showing the main advantages of the tensor/mass model. The first one is a little experience on anisotropic behavior. We take two identical cylinders: one is isotropic (figure 2a and 2c) and the other is 5 times stiffer in the z direction (figure 2b and 2d). First, we pinch the cylinders by applying forces perpendicular to the anisotropic direction. We can see that the range of the deformation is the same for the two models. Furthermore the deformation is more realistic in the second case because it is smoothed along the anisotropic direction. On the second set of cylinders (figure 2c and 2d) we can see the deformation resulting from the application of the same force in the z direction on the top face of each cylinder (the bottom face is fixed). As predicted by the theory, the model is much stiffer in the anisotropic direction, and the stretch is much smaller.

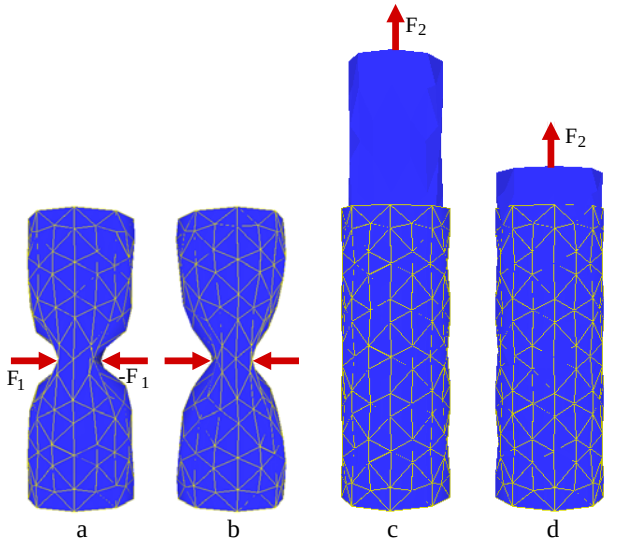


Figure 2: Comparison between isotropic (a and c) and anisotropic (b and d) cylinders

The second result (figure 3) shows a simulation of laparoscopic liver surgery during which the user pulls the right part of the deformable models with one tool while cutting it with another.

2.7 Hybrid model [14]

A combination of the two previous deformable models have been proposed in [14]: a hybrid model combines a large pre-computed model with a smaller Tensor/Mass model allowing for topology changes.

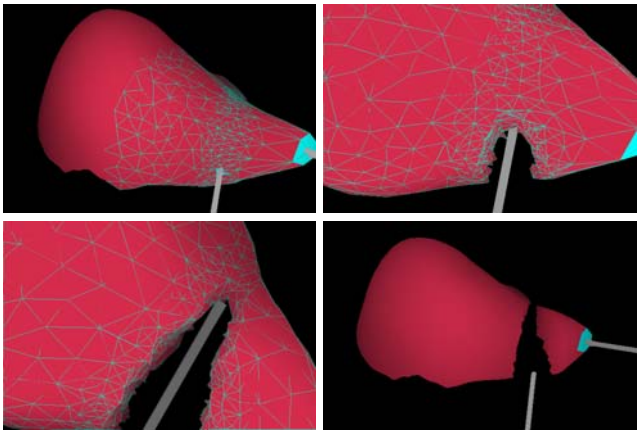


Figure 3: Simulation of hepatectomy

3 Force feedback

Haptic rendering requires a high update rate (ranging from 300Hz for soft objects to 10kHz for rigid contact) to give the user the sensation of continuity.
To achieve a satisfactory haptic feedback, two approaches have been proposed:

- computing forces empirically [4], for example by using a force proportional to the penetration depth of the tool in the object. In this case the force is not computed from a physical deformation, but uniquely from geometric constraints.
- using a simplified physical model. The simplification can be performed in two ways, either by decreasing the mesh size [8] or by performing as much pre-computation as possible [10].

We propose another solution based on human characteristics. Indeed, it has been shown [7] that if haptic rendering is very precise (we can feel force variations above 300 Hz), human gesture can be sampled at a much lower frequency (from 1Hz for the answer to an unexpected signal to 10Hz for a reflex action). Thus, all applied forces must be updated at a high rate, but, because it is related to the user's hand gesture, their evolution is quite slow. The idea is to estimate forces between two time steps of the deformable model simulation thanks to extrapolation. A more detailed presentation of this study can be found in [22].

3.1 Force extrapolation

Our aim is to generate forces at a rate of 500Hz from forces computed by the deformable object simulation at a rate of about 30Hz. The simulation loop gives us a

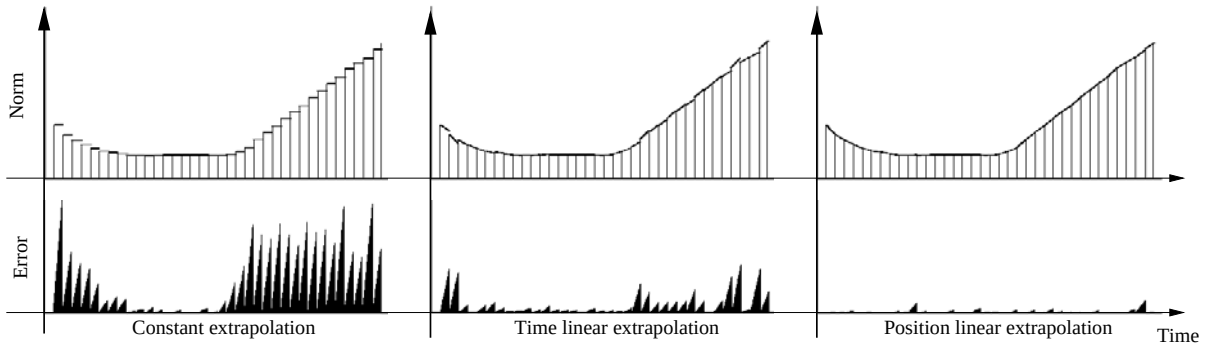


Figure 4: Evaluating the different extrapolation methods

discrete series of parameters $(t_n, \mathbf{P}_n, \mathbf{F}_n)$ representing the force $\mathbf{F}_n$ applied to the tool in position $\mathbf{P}_n$ at time t_n. Good quality force feedback can be reached by a force update at about 500Hz. So, we must choose an extrapolation function $\mathbf{F}(t)$ providing an estimation of the force to apply to the tool at time t $(t_n \leq t < t_{n+1})$ according to already known data $(t_i, \mathbf{P}_i, \mathbf{F}_i),\ i = 0..n$.

The hardware of the *Laparoscopic Impulse Engine* assumes a **constant extrapolation** between two received forces. This has several advantages. First, it does not require any additional computation. Second, as the applied force results from the deformation computation, such an extrapolation scheme ensures that only valid forces are applied without the risk of damaging the device. The main problem with this method is the discontinuity of the applied force which gives the sensation of touching a rough surface as soon as the update rate becomes too low (under about 300Hz).

Another way to estimate the current value of a signal changing over time is to extrapolate it over time. As our deformable model sketches a linear elastic behavior, we only consider **linear extrapolation over time**. This method gives better results than the previously described one. The force discontinuities are less noticeable. But we must face a new problem, as the applied forces are not the ones that the simulation of the deformable model computes, they can be arbitrarily large. These force amplitude peaks occur especially when the time step increases.

The force changes are mainly due to the tool movement. In addition, it is possible to query at a very high rate the position of the tool during the extrapolation. These observations lead us to develop a force estimator based on the tool position: **linear extrapolation over position**. We project the current tool position $\mathbf{P}$ onto the line defined by the two previous tool position $\mathbf{P}_{n-1}$ and $\mathbf{P}_n$ to obtain $\mathbf{P}$' (figure 5). We can then consider the norm ratio for extrapolation:

$$\mathbf{F}^p(t) = \mathbf{F}_n + \frac{\|\mathbf{P}' - \mathbf{P}_n\|}{\|\mathbf{P}_n - \mathbf{P}_{n-1}\|}(\mathbf{F}_n - \mathbf{F}_{n-1}) \qquad t_n \leq t < t_{n+1}$$

We notice that the error induced by the tool position projection is null when $\mathbf{P}_{n-1}$, $\mathbf{P}_n$, and $\mathbf{P}$ are aligned, in other words when the tool trajectory is a line.

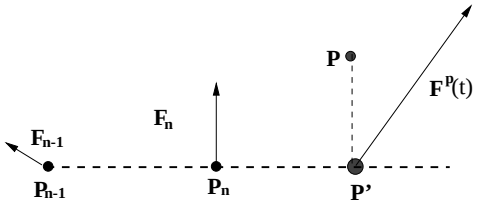


Figure 5: Tool position projection for extrapolation over position

All of these three extrapolation methods were implemented in our surgery simulator. In order to compare and to evaluate them, several experiments were performed. The results are presented in the next section.

3.2 Results

To evaluate and compare the three extrapolation methods, the time, the tool position, and the force computed by the simulation of the deformable model were recorded during several surgery simulation sessions. We have interpolated the force *a posteriori* to have a reference for the computation of the errors. We present results on the norm of the force. The top line of figure 4 shows the original data set with the impulses and the extrapolated one as a line. The norm of the difference between the extrapolated and the interpolated forces, which is taken as a measure of the error, is plotted on the bottom line.

We note that the linear extrapolation over position gives very interesting results (very few discontinuities and no singular forces). We tried the same type of experiment with different simulation frequencies and with different tool movements. The position linear extrapolation always gave the best results, which is confirmed by the sensation perceived during simulation.

4 Conclusion

The surgery simulator prototype presented in this paper allows us to simulate several surgical gestures with good visual and haptic realism. Our future work will focus on modeling more complex bio-mechanical behavior, with the introduction of an external shell representing the *capsule de Glisson*, and the development of non-linear elastic models. An other improvement will be the modeling of the hepatic vessels which constitute one of the main problems during liver surgery.

References

[1] "Action Incitative SIMulation de chirurgie", http://www.inria.fr/epidaure/AISIM.

[2] N. Ayache, S. Cotin, H. Delingette, J.-M. Clement, J. Marescaux, and M. Nord. "Simulation of endoscopic surgery", *Journal of Minimally Invasive Therapy and Allied Technologies (MITAT)*, 7(2):71–77, July 1998.

[3] K.-L. Bathe. *Finite Element Procedures in Engineering Analysis*, Prentice-Hall, 1982.

[4] C. Bosdogan, C. Ho, M. A. Srinivasan, S. D. Small, and S. L. Dawson. "Force Interaction in Laparoscopic Simulation: Haptic Rendering Soft Tissues", *Proceedings of the Medicine Meets Virtual Reality (MMVR'6) Conference*, pages 28–31, January 1998.

[5] M. Bro-Nielsen, D. Helfrick, B. Glass, X. Zeng and H. Connacher. "VR Simulation of Abdominal Trauma Surgery", *Proceedings of the Medicine Meets Virtual Reality (MMVR'98) Conference*, pages 117–123, January 1998.

[6] F. Boux de Casson, C. Laugier "Modeling the Dynamics of a Human Liver for a Minimally Invasive Surgery Simulator", *MICCAI'99*, September 1999, Cambridge UK.

[7] T.L. Brooks. "Telerobotic response requirements", *Proceedings of the IEEE International Conference on systems, Man and Cybernetics*, pages 113–120, 1990. Los Angeles.

[8] G. Burdea, G. Patounakis, V. Popescu, and R. E. Weiss. "Virtual Reality Training for the Diagnosis of Prostate Cancer", In *IEEE International Symposium on Virtual Reality and Applications (VRAIS'98), Atlanta, Georgia*, pages 190–197, March 1998.

[9] S. Cotin, H. Delingette, and N. Ayache. "Efficient linear elastic models of soft tissues for real-time surgery simulation", Technical Report RR-3510, INRIA, 1998.

[10] S. Cotin, H. Delingette, and N. Ayache. "Real-time elastic deformations of soft tissues for surgery simulation", *IEEE Transactions On Visualization and Computer Graphics*, 5(1):62–73, January-March 1999.

[11] P. G. Ciarlet. *Mathematical elasticity Vol. 1: Three-dimensional elasticity*, ISBN 0-444-70259-8, Amsterdam, 1987.

[12] G. Debunne, M. Desbrun, A. Barr, M.-P. Cani. "Interactive multiresolution animation of deformable models", *10th Eurographics Workshop on Computer Animation and Simulation (CAS'99)*, September 1999.

[13] H. Delingette. "Towards realistic soft tissue modeling in medical simulation", *Proceedings of the IEEE : Special Issue on Surgery Simulation*, pages 512–523, March 1998.

[14] H. Delingette, S.Cotin, and N.Ayache "A Hybrid Elastic Model allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation", *Computer Animation*, Geneva Switzerland, May 26–28 1999.

[15] Y. C. Fung. *Biomechanics - Mechanical Properties of Living Tissues*, Springer-Verlag, second edition, 1993.

[16] S. Gibson, C. Fyock, E. Grimson, T. Kanade, R. Kikinis, H. Lauer, N. McKenzie, A. Mor, S. Nakajima, H. Ohkami, R. Osborne, J. Samosky, and A. Sawada. "Volumetric object modeling for surgical simulation", *Medical Image Analysis*, 2(2), pp 121–132, 1998.

[17] M. Kaiss and P. Le Tallec. "La modélisation numérique du contact œil-trépan", *Revue Européenne des éléments Finis*, 5(3):375–408, 1996.

[18] J.-C. Lombardo, M.-P. Cani, and F. Neyret. "Real-time collision detection for virtual surgery", *Computer Animation*, Geneva Switzerland, May 26–28 1999.

[19] J. Marescaux, J-M. Clément, V. Tassetti, C. Koehl, S. Cotin, Y. Russier, D. Mutter, H. Delingette, N. Ayache. "Virtual Reality Applied to Hepatic Surgery Simulation : The Next Revolution", *Annals of Surgery*, 228(5), pp 627–634, 1998.

[20] W. Maurel, Y. Wu, and N. M. T. D. Thalmann. *Biomechanical Models for Soft Tissue Simulation*, ESPRIT Basic Research Series. Springer-Verlag, 1998.

[21] F. Neyret and M.-P. Cani. "Pattern-based texturing revisited", *SIGGRAPH'99*, August 1999, Los Angeles CA.

[22] G. Picinbono and J.-C. Lombardo. "Extrapolation: a Solution for Force Feedback?", *International Scientific Workshop on Virtual Reality and Prototyping*, June 3–4, 1999, Laval, France.

[23] A. J. Spencer. *Continuum Theory of Fiber-Reinforced Composites*. Springer-Verlag, New York, 1984.

[24] G. Székely, Ch. Brechbühler, R. Hutter, A. Rhomberg and P. Schmid. "Modelling of Soft Tissue Deformation for Laparoscopic Surgery Simulation", *proc. MICCAI 98*, Cambridge MA, USA, October 11-13, 1998.

[25] M. Vidrascu. "Simulation of the deformations of the liver by domain decomposition", *mini-symposim Decomposition de domaines de USNCCM99*, August 1999, Boulder.

[26] Jeffrey A. Weiss, Bradley N. Maker, and Sanjay Govindjee. "Finite element implementation of incompressible, transversely isotropic hyper-elasticity", *Computer Methods in Applied Mechanics and Engineering*, 135:107–128, 1996. (U.C. Berkely Report UCB/SEMM-95/07).