Proceedings of the 2000 IEEE
International Conference on Robotics & Automation
San Francisco, CA • April 2000

A Haptic Interface for a Virtual Exam of the Human Thigh

Diego d'AULIGNAC, Remis BALANIUK* and Christian LAUGIER
SHARP Project
GRAVIR/INRIA Rhône Alpes - France
{Diego.D_Aulignac, Remis.Balaniuk, Christian.Laugier}@inrialpes.fr

Abstract

This paper proposes a method for interfacing a force-feedback device of type PHANToM to a spring-damper model of the human thigh. The model was defined from experimental data and it is simulated using implicit integration. The main difficulty in this is that while the PHANToM needs to receive the force values at a rate of 1KHz, the physical model runs at a maximum speed of 100Hz. Supplying forces at this frequency leads to unrealistic vibration in the force feedback. The novelty of our approach is the use of a local model supplying reliable force values at a high frequency. The purpose of this work is to contribute for the implementation of an echographic simulator with force-feedback.

1 Introduction

Echography, in general terms, exploits the information which an acoustic signal provides when it is reflected off a structure to determine the position and shape of the latter. A common exam is the echography of the thigh to detect a thrombosis in the vein. A healthy vein will compress under the influence of an external force while a vein affected by thrombosis will only partially or even not at all compress, depending on the stage in the evolution of the illness. Depending on the pressure the practitioner applies with the echographic probe on the thigh, he will get an image from which the current state of the vein can be deduced, and hence a possible thrombosis diagnosed.

Unfortunately, the diagnosis requires a considerable amount of experience, which currently is gathered on real patients under the supervision of an expert. This makes learning slow and certain pathologies are rarely encountered. The goal of our work is to lay the groundwork for the development of an echographic simulator with force feedback. In the final system, the trainee will be looking at artificially generated echographic images, and interacting with a computer simulated dynamic thigh model through a haptic interface.

Henry [6] examined how a set of echographic images could be used to construct a more general model that would take into account the orientation and pressure exerted with the probe on the thigh. Thus we can generate an image that is dependant on the deformation. However, this does not provide us with a sense of the force needed to provoke this deformation. This notion of force is of paramount importance in the simulation of an echographic exam to be of use for training purposes. The model used to simulate the dynamics is based on a spring-damper network as described in Section 3. The parameters chosen for this model are derived from measurements on a real thigh as published in [4].

However, this model will only update its state at a maximum frequency of approximately 100Hz. Our aim is to provide force feedback through means of a haptic interface of type PHANToM[1]. These interfaces require a very high update rate for the force and typically this frequency is around 1KHz. However, the physical model that simulates the behavior of the thigh is not able to provide force values at such rates.

Previously, [9] already suggested decoupling the haptic servo loop from the main application, while [1] proposed a multi-resolution approach where only areas of haptic interest are updated at a high frequency.

In this paper we present a local model for deformable objects to bridge the gap between the simulation and haptic frequency.

2 Overview of the system

Figure 1 shows how the physical model (see Section 3) and the haptic interface (PHANToM servo

*also assistant professor at the Universidade Católica de Brasilia - Brasil

[1] see http://www.sensable.com

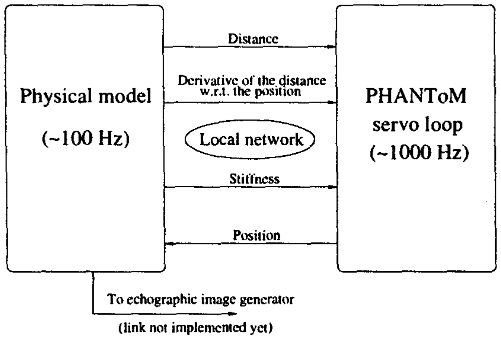
Physical model
(~100 Hz)
Distance
Derivative of the distance w.r.t. the position
Local network
Stiffness
Position
PHANToM
servo loop
(~1000 Hz)
To echographic image generator
(link not implemented yet)

Figure 1: System overview.

loop) are connected via the local network by socket connections. The haptic interface is conceived as an external module of the system simulating the physical model. Based on a simplified local model (see Section 4) the force values are estimated inside the servo loop at the haptic rate.

The physical model of the thigh does not supply force values to the haptic interface. It receives the actual position of the virtual echographic probe, corresponding to the position of the end effector of the PHANToM arm, which is held by the user. It computes and returns the minimum distance from the probe to the thigh as well as the partial derivatives of this distance with respect to the probe position. It passes also a linear approximation of the local stiffness (i.e. rate of change of the force with respect to a change of position in the direction normal to the surface of the thigh). Finally it updates its own dynamic state. These steps are repeated in a loop at a rate which directly depends on the physical model computational delay (~100Hz).

The local model is updated using the values passed to it by the physical model. This update is repeated at the same rate of the physical model update. However, the servo loop rate does not depend on the local model update rate and the local model will always approximatively emulate the physical model.

The link to the echographic image generator is currently being implemented.

3 Description of the physical model

Based on the experimental data and the computational requirements, a two layer lumped element model is chosen. The two layer model is composed of a surface mesh of masses, linear springs and dampers, and a set of nonlinear springs orthogonal to the surface to model volumetric effects by giving normal support to the surface mesh (see Figure 2). The deformation-force relation of the linear and nonlinear springs are of the form

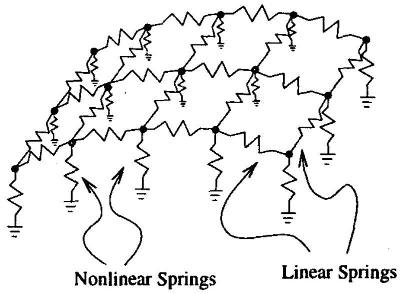

Figure 2: Two layer model of the thigh.

$$f(x) = ax$$

$$f(x) = \frac{x}{ax+b}$$

where x is the difference in the length of the springs with respect to their original, resting length. The parameters of the springs have actually been fitted to match the real deformation measured on a human thigh [4].

Implicit Integration Since the forces given by the above equations can be quite large, this will lead to stiff differential equations. Using simple explicit integration methods may lead to instability and divergence. This issue has been discussed by [3] in the context of cloth simulation. They propose an *implicit* integration method, also known as *backward* Euler method. In its general form it has the following formulation.

$$\Delta Y \left[\frac{1}{\Delta t} I - \lambda (\nabla f)|_{Y=Y_0} \right] = f(Y_0) \tag{1}$$

where Y is the state vector expressing the positions and velocities of all particles. Thus ΔY represents the change in state and $f(Y_0)$ the derivative at the current state. Further, I is the identity matrix, ∇f the Jacobian matrix derived from the spring configuration of the object, and λ is a constant where $\lambda \in [0, 1]$. The resulting square matrix is composed of n^2 elements, where n is the size of the state vector. However, since particles not fully interconnected, this matrix is sparse. Thus we use the conjugate-gradient method [10] to solve Equation 1.

Collision Detection Further, to detect the interaction between objects in a scene, a collision detection algorithm must be used. It is well known that collision detection is a computationally expensive task and a potential bottleneck in every application that aims to achieve real-time performance. For this reason it is essential to optimize the routines to their maximal extent. [7] proposes an algorithm that obtains, in linear time, the points of contact on two deformable concave objects, as well as the direction of the contact and the volume of inter-penetration.

Alternatively, an approach for detecting the collision between deformable and rigid objects of rectangular shape has been proposed by [8]. It makes use of the OpenGL hardware to detect the polygons within a bounding box. If this bounding is superimposed onto the rigid, rectangular object, the polygons in interaction will be detected. This approach has the advantage of being very fast due to hardware acceleration and is used in this application.

Processing physical interactions Once a collision has been detected an appropriate response must be computed. There exist several collision response models but for deformable objects the penalty method seems to be the most appropriate (see [5] for more details). Using this approach the force applied at a given point of an object where a collision has taken place is given by

$$\vec{F_c} = \begin{cases} (-\lambda v - \mu \dot{v} v)\vec{k} & \text{if } v < 0 \\ \vec{0} & \text{otherwise} \end{cases} \qquad (2)$$

where λ is the rigidity factor of the collision, μ is a damping factor (which represents the dissipation of energy), v the volume of inter-penetration, and $\vec{k}$ the contact direction.

To avoid interpenetration, the rigidity constant has to be quite large, which leads to stiff differential equations. This represents no problem for the implicit integration scheme. However, if an equal and opposite force vector is passed to the force-feedback device at the global simulation rate of 100 Hz this leads to unacceptable trembling. To resolve this problems we propose to use a local approximation at a higher frequency, as explained in the next section.

4 Haptic interface using a buffer model

The disparity between the physical model update frequency and the haptic rendering frequency can compromise the haptic interface. The haptic device needs to display forces at rates around 1KHz to maintain a realistic sense of touch. Simple solutions as forces interpolation cause delays in the haptic rendering, and presenting force feedback in the presence of even small delays has been shown to create operator-induced instabilities [11].

To bridge this disparity we use a "buffer model" between the physical model simulator and the haptic device. This approach was first proposed in [2]. The haptic interface is conceived as an external module of the physical simulator, able to estimate the contact forces to be displayed using its own simplified physical model. This simple model locally emulates the "real" physical model and can estimate contact forces at the haptic rate. The buffer model is a generic numerical model, defined by a set of parameters, continuously adapted in order to fit the values informed by the physical simulator. The contact forces estimation is based on a very simple physical simulation, with a point interacting with a non deformable constraint surface defined by the buffer model.

This constraint surface represents the nearest portion of the thigh with respect to the probe. The position and the shape of this surface will be defined by the model parameters. The constraint surface is defined in the haptic device configuration space. In an ideal framework, the haptic device would offer force and torque feedback, and the proxy would move in a 6-D configuration space. In this case, the proxy motions can be expressed in full probe motions (6 dof : translation and rotation). The PHANToM haptic device we use has no torque feedback, and the proxy is a point in a 3-D configuration space. The proxy motions are expressed only by translations of the probe in the virtual environment (3 dof). The probe does not rotate.

Inside the haptic interface we use a constraint-based framework. The haptic probe location inside the configuration space is defined by two values: its effective position and a virtual location, subject to the constraining obstacles. This virtual location, called the "god-object" in [13] and "virtual proxy" in [12], is the place the described object would be if the obstacles were infinitely stiff.

The haptic interface will be defined by two processes: the model manager and the haptic loop.

The model manager interacts with the physical model, passing the position and obtaining the minimum distance between the haptic probe and its closest contact point on the virtual thigh. Using this distance and its derivatives, the manager will adapt the constraint surface to locally fit these values. The physical model must also transmit the local stiffness at the contact point.

The model manager uses the obtained distance (D) and its partial derivatives (D') to update the buffer model.

To explain how the buffer model is updated let us consider the distance D as being defined by an unknown mapping $f : \Re^n \rightarrow \Re$:

$$D = f(x_i)\ (i = 1, ..., n)$$

$$D' = \left[\frac{\partial f}{\partial x_1} ... \frac{\partial f}{\partial x_n}\right]$$

The buffer model can be defined as a function ϕ : $\Re^{r+n} \rightarrow \Re$, which can be written as :

$$\phi(x) \equiv \phi(u, x)$$

for certain $u = [u_1 ... u_r]^T \in \Re^r$. We will refer to $u_1, ..., u_r$ as the *parameters* and to $x_1, ..., x_n$ as the *variables* . The variables correspond to the configuration space dimensions.

We define also the partial derivatives for ϕ :

$$\phi' = \left[\frac{\partial \phi}{\partial x_1} ... \frac{\partial \phi}{\partial x_n}\right]$$

The fitting algorithm to adapt the buffer model with respect to f is based on the following first order approximations :

$$\Delta\phi \approx \sum_{i=1}^{r} \Delta u_i . \frac{\partial \phi}{\partial u_i}$$

$$\Delta\phi'_i \approx \sum_{j=1}^{r} \Delta u_j . \frac{\partial \phi'_i}{\partial u_j}\ (i = 1, ..., n)$$

We define $\Delta u = [\Delta u_1 ... \Delta u_r]$ as the vector of corrections to be applied to the model parameters,

$$\frac{d\phi}{du} = \left[\frac{d\phi}{du_1} ... \frac{d\phi}{du_n}\right]^T$$

as the vector of the partial derivatives of ϕ with respect to the model parameters,

$$\Delta\phi' = [\Delta\phi'_1 ... \Delta\phi'_n]^T$$

as the vector of the differences on the values of ϕ' obtained by the application of the corrections Δu to the model parameters, $\Delta\phi$ as the difference on ϕ we obtain applying Δu to the model parameters and :

$$\frac{d\phi'}{du} = \begin{bmatrix} \frac{\partial \phi'_1}{\partial u_1} & ... & \frac{\partial \phi'_n}{\partial u_1} \\ ... & ... & ... \\ \frac{\partial \phi'_1}{\partial u_r} & ... & \frac{\partial \phi'_n}{\partial u_r} \end{bmatrix}$$

as the matrix of partial derivatives of ϕ' with respect to u.

Writing in a matrix form :

$$\begin{bmatrix} \Delta\phi \\ \Delta\phi' \end{bmatrix} = \begin{bmatrix} \frac{d\phi}{du} \\ \frac{d\phi'}{du} \end{bmatrix} du = \mathbf{J}\ \mathbf{du}$$

We consider the problem of fitting ϕ to locally emulate f as the problem of finding the du corrections to be applied to the model parameters u so that the model estimation errors ($\Delta\phi, \Delta\phi'$) are minimized. The model estimation errors are assumed to be :

$$\Delta\phi = D - \phi(u, x)$$

$$\Delta\phi' = D' - \phi'$$

We use the pseudo inverse of $\mathbf{J}$ ($\mathbf{J}^\dagger = (\mathbf{J}^\mathbf{T}\mathbf{J})^{-\mathbf{1}}\mathbf{J}^\mathbf{T}$) to estimate the parameters corrections (du) :

$$du = \mathbf{J}^\dagger \begin{bmatrix} \Delta\phi \\ \Delta\phi' \end{bmatrix}$$

Using iteratively this approach the model manager will continously keep $\phi(u, x) \approx f(x)$ nearby the user effector position.

The haptic loop follows the constraint based rendering framework. The contact forces are estimated using the constraint surface, the proxy position, the user motion and the informed stiffness.

To implement this framework, at each time step of the haptic loop we obtain the haptic device configuration (the "probe position") and we change the proxy position to reduce its distance to the probe position subject to the constraint surface.

Generically, this problem can be written as :

$$\begin{array}{ll} minimize & D(x) = \frac{1}{2}(x - p)^T(x - p)\ s.t. \\ & \phi(x) \geq 0. \end{array}$$

where x is the proxy position and p the probe position.

Before moving the proxy we must to test if it will collide with the constraint surface. The collision detection between the proxy and the surface can be simplified when the ϕ function defines a large surface.

Using ϕ we simply compute the distance between the probe position and the constraint surface. If the distance is positive or zero then the probe is in the free space and there will not be collision; in this case the proxy can be moved directly towards the probe position. If the distance is negative then the probe is inside the obstacle. The proxy is moved until it reaches a contact with the constraint surface in its linear path towards the probe position. At this point it is no longer possible to move directly to the probe position, but it may be still possible to reduce the distance between the proxy and probe positions, keeping the surface contact. The minimization problem becomes:

$$\begin{aligned} \textit{minimize} \quad & D(x) = \tfrac{1}{2}(x-p)^T(x-p) \ s.t. \\ & \phi(x) = 0. \end{aligned}$$

Zilles and Salisbury [13] and Ruspini *et al.* [12] solve this problem using constraining planes and Lagrange multipliers. We use the same approach introducing a first order approximation for ϕ, corresponding to a constraint plane tangent to the constraint surface and going through the proxy position ($x = xp$). See [2] for more details in this approach. In the simulation of the echographic exam the buffer model was implemented using a spherical model

$$\phi(u,x) = \sqrt{(x_1-u_1)^2 + (x_2-u_2)^2 + (x_3-u_3)^2} - u_4$$

where (u_1, u_2, u_3) determine the sphere center and u_4 the sphere radius. The vector $x = [x_1 \ x_2 \ x_3]$ represents the probe Cartesian coordinates, and $u = [u_1 \ u_2 \ u_3 \ u_4]$ denotes the parameters to be identified by the model manager.

Force estimation in the constraint based framework becomes easy after the proxy location is determined. Simple impedance control techniques can be used to estimate the forces, using the given local stiffness.

The general form of the forces estimation can be given by the Hooke's law:

$$f = k(x-p)$$

where k is proportional to the local stiffness, x the proxy position and p the probe position.

In the constraint based framework, a number of additional effects can be easily implemented. Friction can be simulated by controlling the proxy motion. Texture can be simulated controlling the constraint plane [12]. Force shading is used to make a regular polygonal surface to be perceived as a continuous smooth surface. All these effects have also been implemented in our approach.

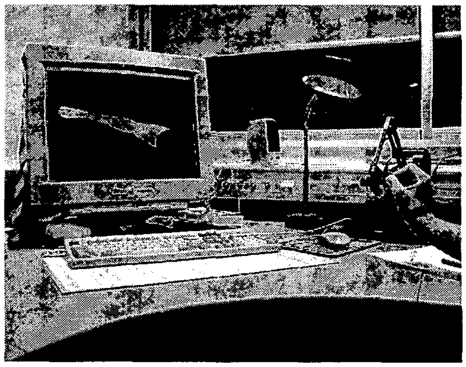

Figure 3: Using the system.

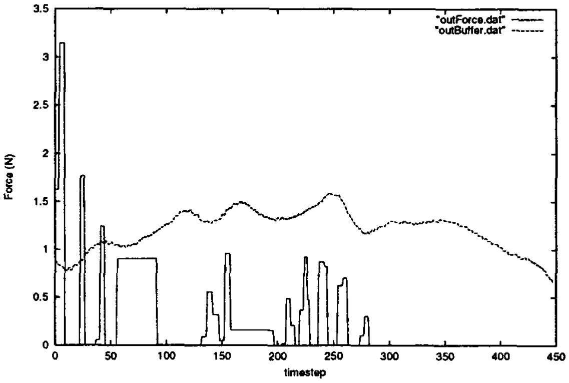

Figure 4: Evolution of the force over a time interval using the haptic buffer model (dashed), and without (solid).

5 Results

Figure 3 shows a user manipulating the PHANToM to interact with the system. The typical update frequency of our physical model on a SGI R10000 processor at 175 MHz is approximately 100Hz.

Figure 4 shows the evolution of the magnitude of the force during a manipulation of the virtual thigh. When using the buffer model we obtain the dashed curve. However, for a similar interaction without the buffer model we obtain the solid curve. This curve presents serious discontinuities; when forces return to a zero value the user has lost contact with the surface of the thigh due to the trembling. This trembling is due to the large variation of the forces.

6 Conclusions

In this paper we have presented a physical simulation system providing graphic and haptic interfaces for a virtual exam of the human thigh. A spring damper model of the human thigh was defined from experimental data. This model is simulated in real-time using an implicit integration method. The haptic interface was conceived using the buffer model approach. Using this approach the haptic rendering rate is less dependant on the computational delay of the physical model. This approach can provide reliable haptic rendering even when the simulation frequency is far lower than the demanded haptic rendering frequency.

The purpose of this work is to contribute for the development of an echographic simulator with force-feedback.

Acknowledgments Thanks to the GMCAO project at the TIMC lab in Grenoble, and in particular to Jocelyne Troccaz for her help. This work was partially supported by an INRIA grant within the framework of the AISIM cooperative research action and the France-Berkeley Fund.

Related web sites For more information on the project visit our project home-page: http://www.inrialpes.fr/sharp/modelisation/index-en.html

References

[1] O. Astley and V. Hayward. Multirate haptic simulation achieved by coupling finite element meshes through norton equivalents. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 1, pages 989–994, Leuven (BE), May 1998.

[2] R. Balaniuk. Using fast local modeling to buffer haptic data. In *Proc. of the PHANToM Users Group Workshop*, Boston, MA (US), October 1999.

[3] D. Baraff and A. Witkin. Large steps in cloth simulation. In *Computer Graphics (Proc. SIGGRAPH)*, pages 43–54, 1998.

[4] D. d'Aulignac, C. Laugier, and M. C. Cavusoglu. Towards a realistic echographic simulator with force feedback. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, volume 2, pages 727–732, Kyongju (KR), October 1999.

[5] A. Deguet, A. Joukhadar, and C. Laugier. Models and algorithms for the collision of rigid and deformable bodies. In P. K. Agarwal, L. E. Kavraki, and M. T. Mason, editors, *Robotics: the algorithmic perspective*, pages 327–338. A K Peters, 1998. Proc. of the Workshop on the Algorithmic Foundations of Robotics. Houston, TX (US). March 1998.

[6] D. Henry. *Outils pour la modélisation de structure et la simulation d'examens échographiques.* PhD thesis, Univ. Joseph Fourier, Grenoble (FR), 1997. (in french).

[7] A. Joukhadar, A. Wabbi, and C. Laugier. Fast contact localisation between deformable polyhedra in motion. In *Proc. of the IEEE Computer Animation Conf.*, pages 126–135, Geneva (CH), June 1996.

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

[9] W.R. Mark, S.C. Randolph, M. Finch, J.M. Van Verth, and R.M. Taylor. Adding force feedback to graphical systems: Issues and solutions. In *Computer Graphics* Proceedings, Annual Conference Series, Proc. SIGGRAPH '96. ACM SIGGRAPH, 1996.

[10] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C.* Cambridge Univ. Press, 2 edition, 1992.

[11] P. Richard, G. Birbent, P. Coiffet, G. Burdea, D. Gomez, and N. Langrana. Effect of frame rate and force feedback on virtual object manipulation. In *Presence*, 1996.

[12] D. C. Ruspini, K. Kolarov, and O. Khatib. Haptic interaction in virtual environments. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, pages 128–133, Grenoble (FR), September 1997.

[13] C.B. Zilles and J. K. Salisbury. A constraint-based god-object method for haptic display. In *Proceedings of ASME Haptic Interfaces for Virtual Environment and Teleoperator Systems*, volume 1, pages 149–150, Chicago, IL (US), 1994.