

Virtual Reality 2001 Conference
March 2001, Yokohama, Japon.

Realistic Haptic Rendering for Highly Deformable Virtual Objects

Cesar A. Mendoza, Christian Laugier

Inria^a Rhône-Alpes & Gravir^b
655 av. de l'Europe, Montbonnot, 38334 St Ismier Cedex, France
<http://www.inrialpes.fr/sharp>

10-12 november 2000

Previous works have presented solutions for stability problems arising from the difference between the sampling rate requirements for haptic devices (about 1 KHz) and the update rates of the physical objects being simulated (about 10 Hz). These methods work well when the objects are convex and non-deformable but when the object is deformable these methods might fail in obtaining realistic force feedback and exact graphical rendering. The reason of this is due to the concavities and unknown shapes that may appear in the deformable objects.

This paper proposes to make the haptic interaction with the local topology of the object and therefore taking into account the unknown changes and the concavities in the object shape. This local model will be updated at the simulation frequency rate. ...

Keywords : Deformable objects, haptic interaction, virtual reality, real-time. ...

Acknowledgements : This work was supported by the National Council of Science and Technology of Mexico (CONACYT) and the National Research Institute in Computer Science and Control of France (INRIA). Thanks to David Raulo, for his technical support in the development of the visual-haptic software platform.

...

^aInstitut National de Recherche en Informatique et en Automatique.

^bLab. Graphisme, Vision et Robotique.

Realistic Haptic Rendering for Highly Deformable Virtual Objects

C.A.Mendoza & C. Laugier

INRIA Rhone-Alpes *
GRAVIR †
ZIRST. 655 avenue de l'Europe
38330 Montbonnot Saint Martin
FRANCE

email : { Cesar.Mendoza-Serrano, Christian.Laugier } @inrialpes.fr

Keywords - Deformable objects, haptic interaction, virtual reality, real-time.

Abstract

Previous works have presented solutions for stability problems arising from the difference between the sampling rate requirements for haptic devices (about 1 KHz) and the update rates of the physical objects being simulated (about 10 Hz). These methods work well when the objects are convex and non-deformable but when the object is deformable these methods might fail in obtaining realistic force feedback and exact graphical rendering. The reason of this is due to the concavities and unknown shapes that may appear in the deformable objects.

This paper proposes to make the haptic interaction with the local topology of the object and therefore taking into account the unknown changes and the concavities in the object shape. This local model will be updated at the simulation frequency rate.

1 Introduction

In the field of virtual reality, most of the interactions between the human operator and the virtual world are done using visual and auditory information. The process of touching a virtual (computer-generated) object by using a mechanical device, known as haptic interaction, is becoming a common form of interaction in virtual environment simulations. Coupled with visual and auditory modalities, haptic sys-

tems can improve the perception of virtual worlds.

To allow a person to touch a virtual environment, haptic rendering systems simulate the forces generated by virtual objects. These systems use a force reflecting mechanical device to apply a force to a user. This force will give the illusion of physical contact with a real object.

The contact and reactive forces in the haptic systems have to be obtained at frequencies of the order of 1 KHz to give a realistic sensation of touch. However, in virtual environment simulations with deformable objects, graphic display frames are updated at rates of about 10 Hz.

The difference in the simulation rates can cause an oscillatory behavior in the haptic device. These non-desired oscillations may indicate that the interface is active and not passive. This is a direct consequence of the time delay and loss of information inherent in sampling. Non-passive haptic devices may become highly unstable and can inflict bodily harm on the operator.

Related works Colgate *et al.* [1] derived conditions of stability under which a haptic display would exhibit passive behaviour. Adams and Hannaford [2], gave a design algorithm to ensure stability of the haptic interface coupled to an arbitrary passive virtual environment, i.e. separating the haptic interface from the virtual environment simulation.

Ellis *et al.* [3] solved the difference rate problem by using a numerical approach. In this method they used prediction and correction to obtain a force feedback between the model updates. However, due to the unknown nature of human behaviour, prediction errors will keep a small error that will cause slight vibrations. D.d'Aulignac *et al.* [4] and W.Mark *et al.* [5]

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

†Lab.d'Informatique GRAPhique, Vision et Robotique

have proposed to make the haptic interaction with intermediate models, spheres or planes, instead of the complete physical model. These intermediate models are updated at the graphical rendering rate. They work well when they interact with non-deformable objects. When the object is deformable the methods will keep a good behavior depending on the actual shape of the object, i.e. its concavity or convexity.

Cavasoglu and Tendick [6] used a linearization of the non-linear deformable model and obtain a more simple model. They evaluate this simple linear model in the simulation loop obtaining errors due to the linearization for deformable objects with heterogeneous compliant surfaces.

In this paper, we present a solution for the difference between the frequency update rates between the haptic devices and virtual environments containing deformable objects. We based our approach in the local topology of the deformable object. The strategy used is to separate the haptic loop and graphical rendering loop, as proposed by [2]. We coupled the loops using the local topology of the deformable object; therefore, the haptic interaction will be done between the haptic device and the local topology of the model at a rate frequency of about 1KHz. This model will be updated by the parameters sent by the physical simulation of the virtual environment at a rate frequency of about 10 Hz.

2 Physical Simulation for the Deformable Virtual Object

In our case, the deformable objects in the virtual environment are modeled as meshes composed of triangular facets, which describe the surface of the object. This model is used for rendering the scene, and for the collision detection algorithm.

The physical model of the object is represented using a mass-spring network mapped onto the geometric mesh. The inertial properties of the object are implicitly defined by the distribution of the mass in all the vertices of the mesh. We will refer to the vertices with an attached mass as particles. The edges of the facets will be mapped onto linear or non-linear spring-damper connectors.

The purpose of the springs connectors is to give a viscous-elastic response to tensile forces, as modeled by the following linear equation:

$$\vec{F} = (-\lambda\Delta d - \mu\dot{d})\vec{k}$$

where λ is the rigidity factor of spring connector, μ is a damping factor, Δd is the relative variation of the

distance between the two connected particles, and \dot{d} is the relative speed between these two particles and $\vec{k} = \frac{\vec{b}-\vec{a}}{|\vec{b}-\vec{a}|}$ being a and b the connected particles.

Motions and deformations of the deformable objects are obtained by integrating at each time step the related dynamic equations. In the physical simulator, for highly deformable objects we make use of the classical Newton-Euler method as suggested by Boux de Casson *et al.* [12]. Then, each particle i is submitted to the following updating equations:

$$\begin{aligned} x_i^{t+\Delta t} &= x_i^t + \Delta t v_i^t \\ v_i^{t+\Delta t} &= v_i^t + \Delta t a_i^t \end{aligned}$$

where Δt is the time step used and x_i , v_i , and a_i are respectively the position, the speed and the acceleration of the particle i . The acceleration a_i of the particle is obtained from $F_i = ma$ where F_i is the total force applied on the particle, i.e. adding external forces, like collision, viscosity environment, etc. and internal forces, like tensile forces.

For simple and non-highly deformable objects we propose to use an implicit integration method, as suggested by D'Aulignac *et al.* [4]. Implicit integration is computationally expensive for real time applications but they are more stable than explicit integration methods.

3 The local topology model approach

To bridge the disparity between the physical model update frequency, and the haptic rendering frequency we propose to make the haptic interaction with an intermediate representation of the physical model. Instead of working with the complete model, the haptic loop will evaluate a more simple model. This intermediate representation will be updated at the same frequency as the physical model.

Previous works, Mark *et al.* [5] and Balankiuk *et al.* [8] have used spheres and planes as intermediate models for the haptic interaction. They work well when the interaction is done with a convex object, (see fig 1.a).

However, due to the deformable behavior of the virtual object we can also have objects with concave shapes. Thus, if the interaction happens to be on a concave section, several problems can arise. First, the graphical rendering will show the virtual probe inside the object. Second, because the local model was a plane or a sphere, bad force feedback is obtained (see fig 1.b).

Therefore, we propose a local model based on the

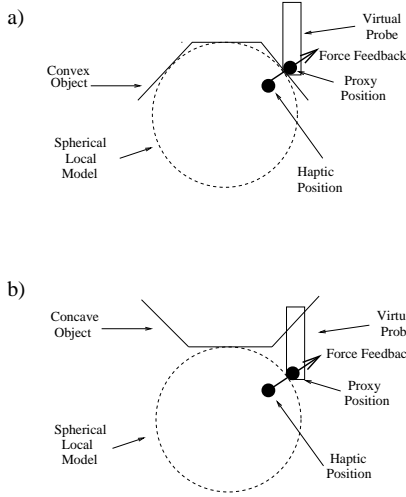


Figure 1: a) Local models as spheres or planes might give a correct force feedback for convex objects. b) For concave objects, the force feedback obtained might be incorrect.

topology of the virtual object. Instead of making the haptic interaction with all the facets that composes the object, we take a set of them. This is done by attaching to the virtual probe a long bar. Then, the simulator developed within the research group, Aladyn 3D-L, is used to identify the facet touched by the long bar, (see fig 2). Aladyn 3D-L based its collision detection algorithm making use of the OpenGL hardware and the approach proposed in [11]. This approach has the advantage of being very fast due to hardware acceleration.

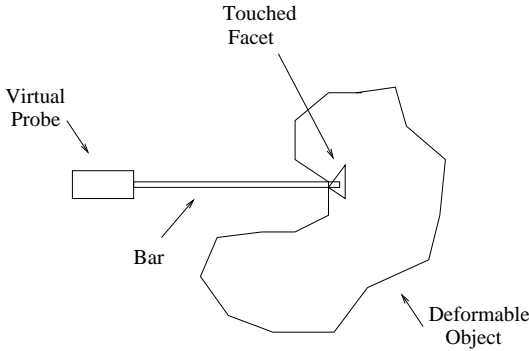


Figure 2: A long bar is attached to the virtual probe to identify the facet that will give the local model.

Only the neighbouring facets to the identified facet will compose the local topological model. In the haptic loop, we follow the constraint based rendering framework. The haptic device will interact with the plane

equations corresponding to each facet. These plane equations will be updated using the normal and the barycentre of each facet. The minimal distance from the haptic position and the planes is obtained and it is sent to the graphical rendering and simulator loop. If the distance is negative, a collision happens and a proper deformation is calculated.

Then we use the god-object [7] and the proxy [9] approach. If the distance is positive or zero the haptic position is in the free space and there will be no force feedback. The proxy can be moved directly to the finger position. If the distance is negative the haptic position is inside the obstacle. The proxy is then moved until it makes contact with the constraint surface in its linear path towards the haptic position. At this point, we continue to move the proxy while in contact with the surface, until the distance between the haptic point and the proxy is minimal. Generically, this problem can be written as :

$$\min_{\vec{x}} \quad distance = \|\vec{x} - \vec{x}_{haptic}\| \quad \text{such that} \\ \vec{n}_i(\vec{x} - \vec{p}_i) = 0$$

where $i = 1, 2, 3$, because for the 3 dof PHANToM device we have only three active planes, $\vec{x} = [x \ y \ z]^T$ is the *proxy position* and $\vec{x}_{haptic} = [x_h \ y_h \ z_h]^T$ is the position of the device. The expression $\vec{n}_i(\vec{x} - \vec{p}_i) = 0$ is the equation of the different constraint planes that forms the local surface. From the plane equation, \vec{n}_i is the normal to plane i and \vec{p}_i is any point in the plane i , in our case it is the barycenter point of the facet i . The equations of the planes expressed in this form works well for implementation purposes.

To implement this framework, the active planes are obtained by taking a line from the proxy position to the haptic position. If this line passes through the facet (within all of the edges), then that plane is considered as active.

For convex shapes, only one surface will be active at a time, but for concave shapes we can have up to three active planes.

Lagrange multipliers are used to obtain the location of the proxy in the surface of the local model as proposed by Zilles *et al.* [7] and Ruspini *et al.* [9]. The Lagrangian for the minimization problem becomes:

$$L = \frac{1}{2}(\vec{x} - \vec{x}_{haptic})^T(\vec{x} - \vec{x}_{haptic}) + \sum_{i=1}^3 \lambda_i [\vec{n}_i(\vec{x} - \vec{p}_i)]$$

Thus, the minimum distance is found by setting the derivatives of L to zero:

$$\frac{\partial L}{\partial \vec{x}} = 0 \\ \frac{\partial L}{\partial \lambda} = 0$$

We obtain the following equation system:

$$\begin{aligned} \vec{x} + N^T \lambda &= \vec{x}_{haptic} \\ N \vec{x} &= D \end{aligned}$$

where $N = [\vec{n}_1 \ \vec{n}_2 \ \vec{n}_3]^T$ is the set of normals to each constraint plane, $D = -[\vec{p}_1 \vec{n}_1 \ \vec{p}_2 \vec{n}_2 \ \vec{p}_3 \vec{n}_3]^T$ and $\lambda = [\lambda_1 \ \lambda_2 \ \lambda_3]$ are the Lagrange multipliers.

The proxy position \vec{x} can be obtained by solving the equation system, thus,

$$\vec{x} = \vec{x}_{haptic} - N^T \lambda$$

where $\lambda = [N^T]^{-1}(\vec{x}_{haptic} - N^{-1}D)$.

Once the proxy location is determined, we use Hooke's law as the impedance control technique to estimate the force that is going to be delivered to the user:

$$f = k(\vec{x} - \vec{x}_{haptic})$$

where k is proportional to the local stiffness.

4 Interaction between the deformable object and the virtual probe

Our approach uses three positions to describe the position of the probe: the *haptic position*, the *proxy position* and the *virtual position*. The *haptic position* represents the real position of the haptic device, the *proxy position* represents the position of the probe in the local model and the *virtual position* represents the position in the graphical rendering loop.

Collision Detection To detect the interaction between the objects in a scene, a collision algorithm must be used. It is well known that collision detection is computationally expensive, which is bad in a real-time context.

In this approach we do not detect the collision between the probe and the virtual object, instead, we attach to the probe, a long bar, invisible in the graphical rendering, as explained in the previous section, and we detect the collision between the bar and the virtual object. Using Aladyn 3D-L, we identify the facets touched, and from this facets we set the local model. The local model will send the distance from the closest facet to the *haptic position* of the probe. If the distance is negative there is a collision between the probe and the virtual object.

Processing physical interaction Once a collision is detected, an appropriate deformation for the virtual

object must be computed. Using the model proposed in [10] the force applied at a given point of an object where a collision has taken place is given by:

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

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

After the deformation, the *virtual position* of the probe is computed placing the probe on the deformed surface. This is done in the same way we place the *proxy* on the surface of the local model, i.e. using the approach of Zilles et al. [7] and Ruspini et al. [9].

5 Experimental Results

5.1 The Software Platform for the visual-haptic environment

The SHARP/GRAVIR team has developed a dynamic simulator called Aladyn 3D-L. It has been implemented using C++/STL/OPEN-GL and it provides force feedback through means of a haptic interface of type PHANToM. The dynamic simulator and the haptic interface are conceived as independent processes and they are connected via a local model. Figure 1 shows the basic structure of the visual-haptic platform.

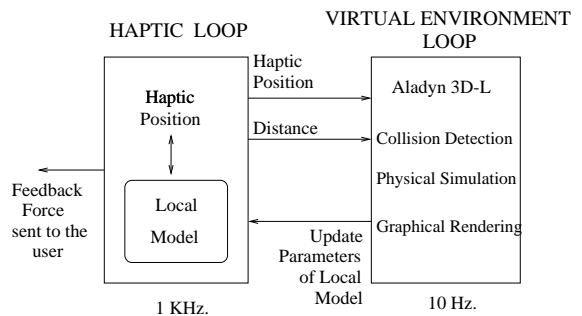


Figure 3: The Software Platform for the visual-haptic environment

The dynamic simulator carries out the physical simulation for the different deformable objects in the virtual environment. It receives the actual position of the virtual probe corresponding to the position of the end effector of the PHANToM arm, which is held by the user. It also checks whether there is a collision between the bar attached to the virtual probe and any

of the virtual objects in the scene. The simulator supplies the parameter to update the local model. This update is repeated at the same rate of the dynamic simulator (10 Hz).

The haptic loop provides the force feedback to the user based in the interaction with the local model, which is modeled as a rigid simple representation of the deformable object.

A virtual environment was implemented using the physical simulator Aladyn 3D-L. The environment consisted of a highly deformable object interacting with a virtual probe. The update simulation was about 10 Hz running in a SGI R10000 processor at 175 MHz. The haptic device employed was a desktop 3 dof PHANToM manipulator with an update rate of 1 KHz (see fig 4).

The system is stable, without oscillations and presents a realist haptic rendering.

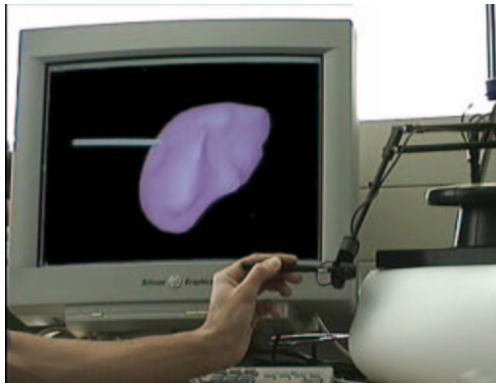


Figure 4: The Software Platform for the visual-haptic environment

6 Conclusion

In this paper we have presented a solution for the difference rate sampling between the haptic device and the physical simulation of deformable objects.

Deformable objects with concave and convex shapes provided reliable haptic and graphical rendering even when the simulation is far lower than the demanded haptic rendering frequency.

The topological local model presented a stable behaviour.

The use of a long bar to built the local model was a good way to establish the obstacle space and therefore the local model of the virtual object. In future work we will use a sphere instead of a bar to set the local model.

Acknowledgments This work was supported by the National Council of Science and Technology of Mexico (CONACYT) and the National Research Institute in Computer Science and Control of France (INRIA). Thanks to David Raulo, for his technical support in the development of the visual-haptic software platform.

References

- [1] E. Colgate, G. Schenkel, "Passivity of a Class of Sampled-Data Systems: Applications to Haptic Interfaces", *Proceedings of the American Control Conference*, pp. 3236-40, Baltimore, USA, (1994).
- [2] R. Adams, B. Hannaford, "Stable Haptic Interaction with Virtual Environments", *IEEE Transactions on Robotics and Automation*, vol. 15, no. 3, (June, 1999).
- [3] E. Ellis, N. Sarkar, M.A. Jenkins, "Numerical Methods for the Force Reflection of Contact", *ASME Transactions of Dynamic Systems Measurement and Control*, 119 (4), pp. 768-774.
- [4] D.d'Aulignac, R. Balaniuk, C. Laugier "A Haptic Interface for a Virtual Exam of the Human Thigh", *Proceedings of the International Conference on Robotics and Automation*, San Francisco, USA, (April 2000)
- [5] W.Mark, S.Randolph, M.Finch, J.Van Verth, R. Taylor, "Adding Force Feedback to Graphics Systems: Issues and Solutions ", *Computer Graphics Proc.*, Annual Conference Series, ACM SIGGRAPH, pp. 447-452, New Orleans, Louisiana, (August, 1996)
- [6] M.C. Cavusoglu, F. Tendick, "Multirate Simulation for High Fidelity Haptic Interaction with Deformable Objects in Virtual Environments", *IEEE International Conference on Robotics and Automation*, ICRA 2000, San Francisco, (April, 2000).
- [7] C.B. Zilles, J.K. Salisbury, "A Constraint-based God-object Method For Haptic Display", *ASME, Haptic Interfaces for Virtual Environment and Teleoperator Systems, Dynamics Systems and Control*, (Chicago, Illinois, Nov. 6-11), vol. 1, pp. 146-150, (1994)

- [8] R. Balaniuk, "Building a Haptic Interface based on a Buffer Model", *Proceedings of the 4th Phantom User Group Workshop*, Boston, USA, (October, 1999).
- [9] D. Ruspini, K. Kolarov, O. Khatib, "Haptic Interaction in Virtual Environments", *Proceedings of International Conference on Intelligent Robots and Systems*, IROS 97, Grenoble, FRANCE, (1997).
- [10] A. Deguet, A. Joukhadar and C. Laugier, "Models and Algorithms for the collision of rigid and deformable bodies", *Robotics: the algorithm perspective*, Proc. of the Workshop on the Algorithmic Foundations of Robotics, Houston, US. (March, 1998)
- [11] J.C. Lombardo, M.P. Cani and F. Neyret, "Real-time collision detection for virtual surgery", *Computer Animation*, Geneva Switzerland, (May, 1999)
- [12] F. Boux de Casson, C. Laugier, "Simulating 2D Tearing Phenomena for Interactive Medical Surgery Simulators" *Computer Animation 2000*, Pensilvania, USA, (May, 2000)