

# Multirate Analysis of Haptic Interaction Stability with Deformable Objects

Federico Barbagli<sup>1</sup> Domenico Prattichizzo<sup>2</sup> Kenneth Salisbury<sup>3</sup>

<sup>1</sup> PERCRO, Scuola Studi Superiori S. Anna, Italy. fed@sssups.it

<sup>2</sup> DII, Università di Siena, Italy. prattichizzo@ing.unisi.it

<sup>3</sup> Robotics Lab and CATSS Lab, Stanford University, U.S.A. jks@robotics.stanford.edu

## Abstract

This paper describes a new technique for allowing multiple users to haptically interact with a set of deformable slowly-simulated objects in a stable manner. Stability has been approached in the past by various researchers using passivity theory in order to avoid having to model the human operator closing the haptic loop. None of these solutions however can work well without the use of high update rates and thus break down in the case of haptic interaction with slowly simulated virtual environments such as the ones featuring highly precise deformable objects. This is particularly true for the case of surgical simulation with force feedback, where precision is a key issue and where complexity can reach high levels. The techniques presented in this paper are based on the concepts of local model for haptic interaction adapted to deformable objects. Such approach allows multiple users to stably interact with a same object while feeling the influence of other users on the same object. Experimental results employing a PHANTOM haptic interface are proposed for a simple example.

## 1 Introduction

Haptic interfaces (HI) allow physical interaction with virtual environments through the sense of touch. Combined haptic and visual interfaces have been around for decades in a variety of forms enabling simple remote manipulation. Yet, despite an enormous increase in research activity in the last few years, the science of haptics (perception and manipulation with our hands) is still a technology in its infancy. State of the art consists of three degrees-of-freedom

(DOF) interfaces, i.e. devices with three motors that are able to actuate three linear forces at the end effector. The most well known and commercially successful interfaces of this kind are the PHANTOM, produced by SensAble technologies, and the Delta, produced by ForceDimension. Interfaces with higher number of DOFs exist too.

With the term *haptic rendering* we refer to that branch of haptics research that deals with calculating the right interaction force between a virtual representation of the user and a virtual object. In order for the simulation to be realistic such forces should be stable. Early efforts on haptic rendering dealt with simple objects, such as the so called *virtual wall* [1, 2, 3]. This has been followed by research on complex virtual environments (VE) featuring rigid objects [4, 5, 6, 7]. Many current research efforts center on deformable objects to be employed in VR applications for medical training [8, 9].

Two main families of haptic devices have been defined in the past: admittance and impedance [10]. In the following we will focus our attention only on the latter, i.e. to mechanical devices configured to render a commanded force while provided with a measurement of position and/or velocity. This is motivated by the larger availability of impedance devices due to their more limited cost and higher ease of use. Thus from now on, with the term *haptic rendering* we will refer to haptic rendering for impedance displays.

Designing haptic rendering algorithms is a non trivial task. Most of the difficulties are due to the fact that users always penetrate inside virtual objects. While this usually goes unnoticed, because of the limitations of the human position system, various problems can arise. Stability issues may arise in the form of

vibration or unreasonable peaks of force feedback. Unrealistic effects such as transparency of thin objects may also arise <sup>1</sup>.

Both of these aspects have been studied in the past in the case of interaction with rigid objects. Stability for haptic interfaces is a complex control problem since it features a human operator inside the control loop. In order not to model such element passivity theory has been used [2] showing that the maximum stiffness that can be simulated with haptic interfaces in a stable way is proportional to the servo rates used and to the damping in the interface.

Haptic rendering techniques have been developed for the case of single point of contact interaction. The *god-object* [4] and the *virtual proxy* [5] are typically used for single point of contact haptic interaction with rigid objects.

The goal of this paper is to analyze some problems that arise in haptic rendering techniques with complex deformable objects and to present new interaction techniques for such case. It is important to note that the proposed solution is independent from the particular implementation chosen for simulating deformable objects. The reminder of this paper is organized as follows: Section 2 analyzes possible instabilities that arise in the case of interaction with deformable objects; Section 3 presents a possible solution for such instabilities; Section 4 briefly discusses future developments.

## 2 Haptic rendering with deformable objects

Both stability and haptic rendering techniques described above tend to break down in the case of virtual environments employing deformable objects. In the following we will analyze why this happens and propose a possible solution.

### 2.1 Haptic rendering issues

As far as haptic rendering techniques are concerned, some attempts have been made by Rusini and Khatib [5] in order to extend the proxy algorithm to non-rigid objects. What is proposed by the authors is, however, applied to "soft rigid objects", i.e. objects whose shape does not change in time but whose stiffness

<sup>1</sup>(see [4] for a more thorough description of potential problems)

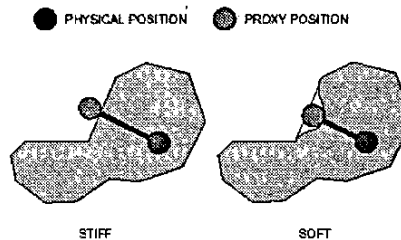


Figure 1: Computing the proxy position using the static image of the object (picture taken from [5])

is limited, as opposed to slowly simulated deformable objects, i.e. objects that assume a new shape when forces interact with them. In the former case the proxy is still computed using the rigid polygonal representation of the object. The force feedback, however, is computed using a point that lays between haptic interface and proxy positions (see Fig. 1). In the latter case the proxy cannot be computed using a static representation of the object, since such representation is constantly changing. Using the undeformed object to compute the proxy will lead to wrong interaction forces. More specifically *users interacting with the same object will not be able to feel the effect of other users on the same object.*

The haptic rendering techniques used for deformable objects should thus be based on the current representation of the object.

### 2.2 Stability issues

Most high precision techniques for deformable objects tend to be computationally expensive. As a consequence it is normally very hard to reach high control rates (> 1KHz) and thus to simulate a large band of impedances [2].

Various solutions, all based on decoupling haptic and simulation loops, have been proposed [11, 6, 12]. The basic idea behind decoupling haptic loop from simulation loop is to use a simple implicit function that approximates, to a good extent, a small part of the object being touched. More specifically such *intermediate representation* or *local model* represents the part of the object which is closest to the current position of the haptic interface (Fig. 2).

Past implementations of local models, typically god-object or proxy, have been imple-

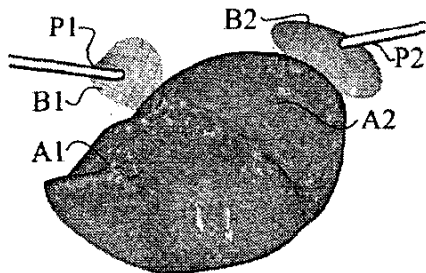


Figure 2: Local representation of object surface

mented using different techniques. Adachi et al. in [11] used a plane, always tangent to the virtual object surface, as an *intermediate representation*. The same has been done by Mark et al. in [6], even though some adjustments have been made to the previous results. Balaniuk in [12] used a spherical *local model*. These solutions work well in the case of rigid objects, but have not been used in the case of deformable ones and such extension does not appear to be trivial.

Computing a local model for rigid objects (or for “soft rigid objects”) can be seen as an “open loop” problem. Given a new probe position inside the VE, a new local model can be computed solely based on geometric considerations. The same does not apply to deformable objects. The local model position depends on the state of the object’s surface. This state, on the other hand, depends on the interaction force between user and virtual object, i.e. on the local model position. Hence a closed loop is created. Such closed loop can become unstable, as discussed in the following, thus driving both the VE and the haptic interface in a vibrating state that completely destroys any sense of realism.

### 2.3 Mathematical description of the problem

The closed loop local model can be rigorously described. Consider the simple case of a one-dimensional deformable object, i.e. a spring, used in conjunction with different types of local models. We will analyze this in a multirate system framework.

Typically simulations comprising both deformable objects and haptic interfaces feature two separate loops. The haptic loop is a process

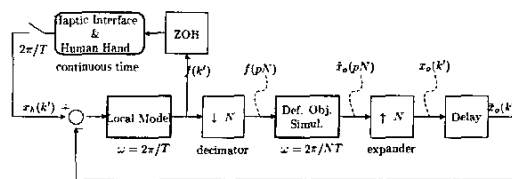


Figure 3: All the blocks that are typically present in a deformable object simulation with haptic feedback. Note that two separate closed loops exist.

that reads the new position of the haptic interface while it’s being moved by the human operator, computes the new interaction force with the VE and writes such force to the haptic interface. Such process runs at high servo rates (e.g. 1KHz), as previously described, due to stability issues. The simulation loop is a process that computes how the interaction force between haptic interface and VE influences the deformable object surface. Such process is usually slow due to the complexity of the simulated environment (e.g. 20Hz). For simplicity such loops are considered synchronized. The simulation engine receives data from the haptic interface every  $NT$  secs and responds to such data every  $NT$  secs while the haptic loop runs every  $T$  secs. The overall structure of such loops is depicted in Fig. 3 where the  $N$ -fold decimator  $f(pN)$ , being  $p$  an integer number, of the high rate signal  $f(k')$  and the  $N$ -fold expander have been used to model the multirate system consisting of two synchronized dynamics, the VE simulation engine running at low rate  $\frac{2\pi}{T}$  (20Hz) and the local model which is  $N$  times faster  $\frac{2\pi}{NT}$  (1KHz).

Since we are considering a single dimensional case, the position of the local model, and of the proxy used on it, are coincident with the free end of the object (see Fig. 4), and are thus computed 20 times per second. The local model is based on a simple spring, i.e. there is always a purely elastic element between proxy and HI positions. Thus given a new HI position  $x_h$  every  $T$  secs, the corresponding interaction force to the user is given by  $K_h(x_h - x_o)$ , where  $K_h$  denotes the local model stiffness. Such force is then sampled, every  $NT$  secs, by the simulation block, which returns a new deformable object surface position  $x_o$ , and thus a new local model position, after  $NT$  secs. We assume the deformable

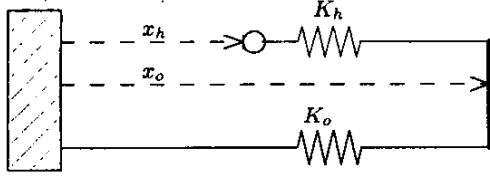


Figure 4: The mechanical model of the haptic interface interacting with a deformable object.

object to have local stiffness  $K_o$ . It is important to note that the deformable object surface position used to compute the interaction force that is fed to the HI is constant for periods of  $NT$  seconds, i.e. the local model position  $\bar{x}_o$  at a given step  $k$  is given by

$$\begin{aligned} \bar{x}_o(k') &= x_o(k'T - T_d(k')) \\ T_d(k') &= (k' - \lfloor k'/N \rfloor N)T \end{aligned} \quad (1)$$

where  $\lfloor a \rfloor$  denotes the integer part function. This type of behavior is often referred to as a *sawtooth* delay, and has been studied for instance for internet-based teleoperation [13, 14].

While the time domain formulation of the problem makes it easier to analyze the system, the study of the system stability is better approached by employing Z-transforms. Let  $L(z)$  be the discrete transfer function representing the Local Model force algorithm, and  $D(z)$  the discrete transfer function representing the deformable object surface algorithm. The multiple rates in the system ( $L(z)$  refers to a period of  $T$  seconds while  $D(z)$  to a period of  $NT$  seconds) and the sawtooth delay make the stability analysis not obvious.

It can be shown that the sawtooth delay is equivalent to the combination of a pure  $NT$  delay before the expander in Fig. 3 and a discrete-time zero-order hold whose impulse response is

$$H(z) = 1 + z^{-1} + z^{-2} + \dots + z^{-(N-1)} \quad (2)$$

after the expander, i.e. with sampling period  $T$  as reported in Fig. 5.

As a consequence of this and of the assumption that the two loops are synchronized the stability analysis for the simulation loop becomes simpler. In this case  $L(z) = K_h$  and it is an easy matter to verify that the input-output relationship between the slow-rate signals  $u(k)$  and  $y(k)$  in Fig. 5 is given by  $y(k) = -K_h \delta(k)$ .

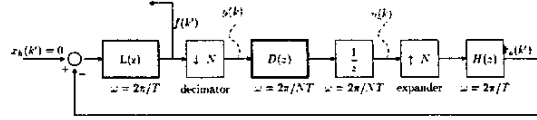


Figure 5: Simulation loop with generic discrete time transfer functions.

Therefore, it has been shown that for a linear elastic model of the deformable object, the virtual environment simulation loop analysis reduces to study the simple control loop at the lower rate  $\omega = \frac{2\pi}{NT}$

$$G(z) = \frac{1}{1 + \frac{K_o}{K_h} z}$$

whose asymptotic stability is guaranteed if and only if

$$K_h < K_o \quad (3)$$

Extensions of multi-rate analysis to more involved local models and VE dynamics other than continuous order hold systems are discussed in [15]. It is the authors' opinion that the multi-rate framework is a theoretical setting well suited to analyze and synthesize local models for deformable object haptic interactions.

### 3 A local model for deformable objects

In order for the local model proposed in [12] to meet the conditions in section 2.3 the following algorithm is proposed:

**Algorithm:**

**Step 1** the local model is calculated inside the slow simulation loop using the current surface configuration of the deformable object, as if it was static, as well as the current haptic interface position;

**Step 2**  $K_h$  is chosen to satisfy stability condition 3 and assigned to the non-rigid local model stiffness;

**Step 3** the force that the local model returns to deformable object simulation, for it to compute its next configuration, is evaluated as a function of the  $N$  forces returned by the local model to the haptic interface inside a  $NT$  period.

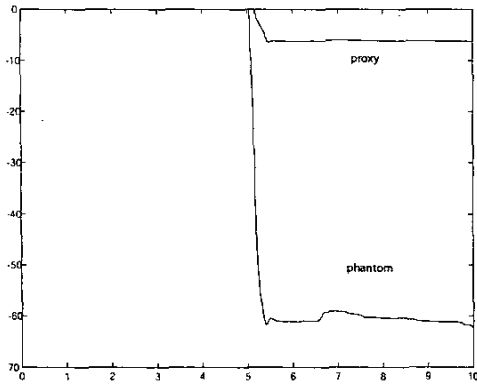


Figure 6: Linear spring response to a step input ( $K_o = 0.9$  and  $K_h = 0.1$ )

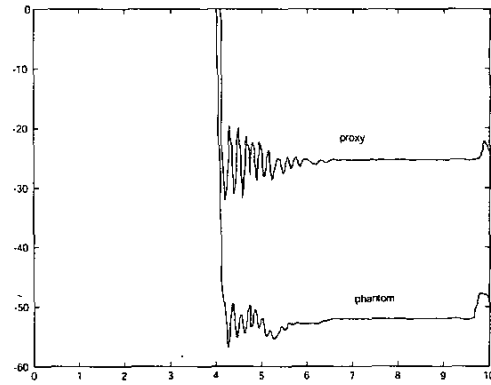


Figure 7: Linear spring response to a step input ( $K_o = 0.2$  and  $K_h = 0.19$ )

Such solution has been tested for the simple case of a one-dimensional spring using the Desktop PHANTOM haptic interface ([www.sensable.com](http://www.sensable.com)). The spring are assumed to be linear. In the case of a non-linear spring,  $K_h$  will be simply adapted to the current value of the spring stiffness  $K_o$ . Experimental results are shown in Fig. 6-8 for different values of  $K_h$  and  $K_o$ . Phantom position  $x_h(kT)$  and proxy position  $\bar{x}_o(kT)$  are plotted in millimeters while the stiffness coefficients are expressed in Newton over millimeters. In each of such experiments the user penetrates the object almost instantaneously and then holds the position, which corresponds to a step input on  $x_h$ .

Fig. 6 shows that for  $K_o = 0.9$  and  $K_h = 0.1$ , the system exhibits a stable behavior while for  $K_o = 0.2$  and  $K_h = 0.19$  in (Fig. 7) stability is critical and is definitively lost in Fig. 8 where condition (3) is not met.

As it appears clear the ratio between  $K_h$  and  $K_o$  not only determines the stability, or lack of, for the deformable object but also the settling time for the surface of the object (proxy position). It is important to note that long settling time lead to noticeable oscillations (see Fig. 7), which is an effect that might corrupt, in part, the overall sense of realism. Thus it should be  $K_o \gg K_h$  whenever possible.

On the other hand using a local model stiffness, such that  $K_o \gg K_h$  is met, means that the steady state step response for the system in Fig. 5 is given by

$$\bar{x}_o = \frac{K_h}{K_h + K_o} x_h$$

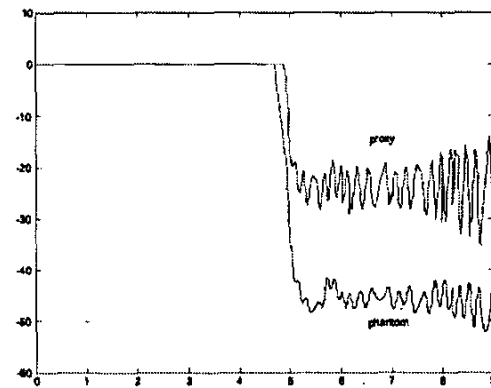


Figure 8: Linear spring response to a step input ( $K_o = 0.2$  and  $K_h = 0.22$ )

which means that the surface settles to a position that is only a fraction of the penetration of the haptic interface inside the object. While this might be fine for objects that are not very compliant, in the case of soft objects the haptic interface must penetrate the object to an extent that is considerably larger than the actual deformation for the object. For instance, if  $K_o = 9K_h$  the settling time is very short (see Fig. 6) but the object surface moves only 1cm for a physical movement of the haptic interface equal to 10 centimeters. Work is in progress to synthesize local models able to cope with this intrinsic tradeoff between performance (settling time) and realistic haptic interaction with deformable objects.

## 4 Conclusions

This paper describes new techniques allowing users to haptically interact with a set of deformable slowly-simulated objects. In particular the paper analyzes possible causes of instability that are specific to the case of deformable objects. The techniques proposed in this paper are based on the concepts of local model for haptic interaction previously introduced by various researchers, which are adapted for the particular case of deformable objects. Experimental results featuring a simple one-dimensional scenario are reported. Future work will involve using a variable  $K_h$  in order to further stabilize the haptic interaction as well as to simulate high frequency interaction forces on the user.

## References

- [1] M. Minsky, M. Ouh-young, O. Steele, F. Brooks, and J. Behensky, "Feeling and seeing: issues in force display," *Computer Graphics*, vol. 24(2), pp. 235-243, 1990.
- [2] J. Colgate and J. Brown, "Factors affecting the z-width of a haptic display," in *Proceedings IEEE Int. Conf. Robotics and Automation*, Los Alamitos, CA, 1994, pp. 3205-3210.
- [3] B. Gillespie, M. Cutkosky, and S. User-Speci, "Rendering of the virtual wall," in *Proceedings of the ASME International Mechanical Engineering Conference and Exposition*, Atlanta, GA, November 1996, ASME, vol. 58, pp. 397-406.
- [4] C. Zilles and J. Salisbury, "A constraint-based god-object method for haptic display," in *Proc. IEE/RSJ International Conference on Intelligent Robots and Systems, Human Robot Interaction, and Cooperative Robots*, 1995, vol. 3, pp. 146-151.
- [5] Diego C. Ruspini, Krasimir Kolarov, and Oussama Khatib, "The haptic display of complex graphical environments," in *Computer Graphics (SIGGRAPH 97 Conference Proceedings)*. ACM SIGGRAPH, 1997, pp. 345-352.
- [6] W. Mark, S. Randolph, M. Finch, J. Van Verth, and R. Taylor, "Adding force feedback to graphics systems: Issues and solutions," in *Computer Graphics: Proceedings of SIGGRAPH'96*, August 1996, pp. 447-452.
- [7] Colgate J.E. Miller B.E. and Freeman R.A., "Guaranteed stability of haptic systems with nonlinear virtual environments," *IEEE Transactions on Robotics and Automation*, vol. 16, pp. 712-719, 2000.
- [8] Stephane Cotin, Herve Delingette, and Nicholas Ayache, "Real-time elastic deformations of soft tissues for surgery simulation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 1, pp. 62-73, 1999.
- [9] R. Balaniuk and I. Costa, "Lem - an approach for real time physically based soft tissue simulation," in *Proceedings of the IEEE International Conference on Robotics and Automation - ICRA 2001.*, San Francisco, CA, 2001.
- [10] T. Yoshikawa, Y. Yokokohji, T. Matsumoto, and X. Zheng, "Display of feel for the manipulation of dynamic virtual objects," *Trans. ASME, Journal of Dynamic Systems, Measurement, and Control*, vol. 117, no. 4, pp. 554-558, 1995.
- [11] Y. Adachi, T. Kumano, and K. Ogino, "Intermediate representation for stiff virtual objects," in *IEEE Virtual Reality Annual Intl. Symposium*, Research Triangle Park, N. Carolina, March 1995, pp. 203-210.
- [12] R. Balaniuk, "Using fast local modeling to buffer haptic data," in *Proceedings of Fourth PHANTOM Users Group Workshop-PUG99*, 1999.
- [13] Kevin Brady and Tzyh-Jong Tarn, "Internet-based teleoperation," in *Proc. of the 2001 IEEE Int. Conf. Robotics and Automation*, Seoul, Korea, May 2001.
- [14] G. Niemeyer and J.J. Slotine, "Toward force-reflecting teleoperation over the internet," in *IEEE International Conference on Robotics and Automation (ICRA)*, Leuven, Belgium, 1998.
- [15] D. Prattichizzo F. Barbagli and Kenneth Salisbury, "Local dynamic models for haptic interactions with deformable objects: a multi-rate approach," Tech. Rep.