The International Journal of
Robotics Research
29(14) 1778–1792

DOI: 10.1177/0278364910378540
ijr.sagepub.com
SAGE

High Fidelity Haptic Rendering of Frictional Contact with Deformable Objects in Virtual Environments using Multi-rate Simulation

Paul Jacobs, Michael J Fu and M Cenk Çavuşoğlu

Abstract

Haptics is an increasingly common modality in human-computer interfacing. The focus of this paper is the problem arising from the difference between the high sampling rate requirements of haptic interfaces and the significantly lower update rates for physical models simulated in virtual environments. This is a critical problem, especially for applications involving haptic manipulation of deformable objects simulated in virtual environments, such as in surgical simulation. In this paper, a multi-rate simulation approach was developed to address this problem. The proposed method employs linear low-order approximations to model the inter-sample behavior of the high-order non-linear deformable object models. The basic method is also extended to achieve high-fidelity rendering of haptic manipulations involving sliding-type frictional contact. The proposed approach uses a local geometric model in addition to the local dynamic model, and performs collision detection and response as part of the high update rate haptic loop. Experimental results that validate the proposed methods are also presented.

Keywords

Medical robots and systems, human-centered and life-like robotics, haptics and haptic interfaces, virtual reality, multi-rate simulation

1. Introduction

The value of haptic interaction in surgical simulation applications has led to a great deal of research interest into the challenges involved in providing haptic force-feedback in virtual environment simulations with deformable surfaces. To achieve this, a key obstacle to overcome is the difference in update rate requirements for deformable object models and haptic interfaces. Simulation of deformable object models are typically linked to the graphical update rate of 10–60 Hz because of computational limitations. Haptic interfaces, on the other hand, need to operate at update rates within an order-of-magnitude of 1 kHz in order to be convincing to the operator. One common method of bridging this gulf is through multi-rate simulation. In multi-rate simulation, the virtual environment is simulated in its full complexity at the visual update rate, while a simpler simulation is run in parallel at the haptic update rate and periodically re-synchronized with the full model.

In this paper, a multi-rate simulation method that uses a local linear low-order approximation to model the inter-sample behavior of the non-linear full-order deformable object model is proposed to address this problem (Section 2). The proposed method is justified by model reduction techniques from system theory and the approach is applied to non-linear physical models. This method is also extended to achieve high-fidelity rendering of sliding-type frictional contact by employing a local geometric model in addition to the local dynamic model (Section 3). The local geometric model is used for performing collision detection and response as part of the high update rate haptic loop, with a novel constraint-based hybrid collision response method (Section 4). Experimental results that validate the proposed methods are also presented (Section 5), followed by a discussion of the results and related stability implications (Section 6).

1.1. Related Work in the Literature

Several different types of deformable object models have been suggested for use in haptic interaction. Lumped mass-spring-damper (MSD) models, such as those presented by Terzopoulos et al. (1987), represent deformable objects with layers of masses connected by spring-damper pairs or damped springs. MSD models are a common choice for surgical simulation due to their ease of implementation and

Case Western Reserve University, Cleveland, OH, USA

Corresponding author:
M Cenk Çavuşoğlu
Case Western Reserve University, Cleveland, OH, USA
Email:cavusoglu@case.edu

low computational requirements. A common alternative to MSD systems are finite-element models (FEM), which provide a continuum model that is strongly physically based and were used by Bro-Nielsen (1998), Cotin et al. (1999), and Wu and Tendick (2004). Other volumetric methods have been suggested to capture the realism of FEM techniques with lower computational requirements, such as the boundary-element method (James and Pai 1999; Renaud and Feng 2003), the method of finite spheres (Kim et al. 2002) and the "Long Element Model" (Sundaraj et al. 2002). Although these models have been found to be useful in haptic simulation, the discussion here is limited to the more widely used MSD and FEM models.

The human sense of touch is remarkably sensitive, and can distinguish between changes in force at frequencies up to hundreds of Hertz. It is generally accepted that the update rate of haptic interfaces must be ten to twenty times higher than the highest frequency event that is to be simulated. Therefore, in order to render events at a 50–100 Hz rate to match the capabilities of the PHANTOM or similar haptic interface, 1 kHz is widely considered the minimum update rate for realistic haptic feedback (Choi and Tan 2004). Real-time virtual environment simulations, on the other hand, are tailored towards visual update rates between 10–60 Hz and it is infeasible to significantly increase the update rate of the physical simulation due to computational limitations. Overcoming this orders-of-magnitude difference in update rates is an important problem in haptics research. The typical practice is to use the same force between the model updates, or to low-pass filter the generated force to the bandwidth of the model update rate. However, this is not an adequate solution as it effectively reduces the haptic update rate to the visual update rate.

Several methods have been proposed to solve this problem through multi-rate simulation. Multi-rate simulation techniques aim to divide a virtual environment simulation into two parallel simulations, one running at the visual update rate and the other at the haptic update rate. The visual update rate simulation models the virtual environment in its full complexity and provides visual feedback to the user, while the haptic update rate simulation uses a simpler and more computationally tractable model to provide only force feedback. Astley and Hayward (1998) took a multi-scale, multi-rate FEM approach to this problem. In their method, a coarse linear finite element mesh modeled the behavior of the overall object and a finer finite element mesh running at a higher update rate was used locally where interactions occurred. Their work was based on decoupling the coarse mesh and the fine mesh by using Norton equivalents as interfaces. This is only applicable to linear finite element cases and the update rates reported were still significantly below the 1 kHz rate necessary for haptics. In an earlier stage of this current study, Çavuşoğlu and Tendick (2000) used a local linearization of MSD models in a multi-rate scheme and showed that the state variables for dominant modes of the linearized system were primarily dependent on nearby nodes. Therefore, a local model consisting of only a few layers surrounding the contact point can be used to provide haptic interaction with improved fidelity and stability.

Multi-rate haptic simulation algorithms using local approximations around the point of contact have previously been studied for rigid objects. Simple geometric shapes, such as planar or spherical approximations were investigated by d'Aulignac et al. (2000). Also, multi-resolution methods with coarser meshes used for the entire model and more detailed meshes for areas of local interest were implemented in multi-rate simulation by James and Pai (2003) and Zhang et al. (2002).

More recently, Barbagli et al. (2005) have developed multi-rate simulation techniques that combine local model approaches and virtual coupling to handle both single- and multi-point contact. Also, Cho et al. (2005) have examined control methods to avoid instability in multi-rate environments caused by time delay in communication channels.

High fidelity haptic rendering of high bandwidth/high frequency interaction with deformable surfaces, such as frictional sliding over a rough surface, is an open problem in haptics in which there have been some advancement. A general real-time drift-free model of friction, which did not include the effect of normal forces on friction, was proposed by Armstrong and Hayward (2000). In addition, Mahvash and Hayward (2002, 2004, 2005) demonstrated a passive multi-rate simulation for haptic interaction with a deformable body, which included friction models. Their algorithm achieved a haptic update rate of 2 kHz and a global update rate of 100 Hz using pre-computed tool contact behavior for a specific material type to reduce computation loads. Mahvash (2006) extended the use of pre-computed forces to simulate tissue separation in real-time by interpolating unknown forces from pre-computed ones. Campion and Hayward (2008) also proposed a haptic texturing algorithm which renders frictional sliding forces based on the height of the rendered texture.

The current work uses a multi-rate simulation approach as a solution for real-time simulation of deformable objects. This multi-rate approach is explained in detail in the next section.

2. Multi-rate Simulation

The core of multi-rate simulation approaches is to divide the necessary computational tasks into those that must be performed at the servo-loop update rate of the haptic interface and those that can be performed at the same rate as the overall simulation. The algorithm is divided into two basic blocks as shown in Figure 1. The "global" simulation incorporates the entire virtual environment and runs at the visual update rate in the order of magnitude of 10 Hz.

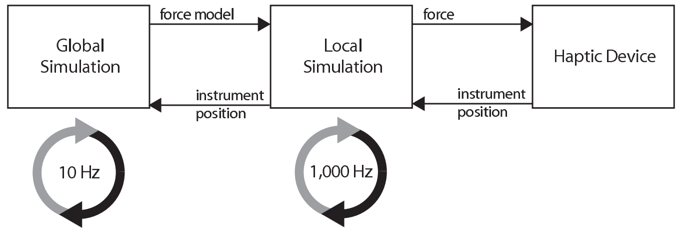

Fig. 1. Block diagram of the basic multi-rate simulation algorithm using a local dynamic model.

A "local" simulation runs at the haptic device update rate, and simulates the behavior of a subset of the global model. In the proposed approach, the global deformations of the deformable object are modeled by the full-order non-linear model, and simulated at the visual update rate, while the local deformations are modeled with the reduced-order models and simulated at the haptic update rate.

After each update of the global model, a low-order approximation model is generated and passed to a second simulation, running either in a separate process or thread in single-computer operation, or running on a second computer in networked operation. This second simulation uses the low-order approximation model to provide force output to the user and then sends the state of the haptic instrument back to the global model, which then re-computes a new low-order approximation for the next cycle.

In the following subsections, the method for constructing low-order approximate dynamic models will be developed and analyzed. In this work, lumped MSD networks are used as the underlying global dynamic model for the deformable objects, because of their simplicity and popularity. A similar treatment is possible using FEMs, but is not presented here.

2.1. Multi-rate Simulation using Linearized Approximations

In order to analyze the construction of the low-order approximate dynamical models that can be used in multi-rate simulation, we start with the paradigm given in Figure 2. Linearization is a basic step. The linearized model gives the tangential behavior of the full model. As we want to capture the behavior in between the model updates, the deformation will be small, and therefore, the linear approximation will remain valid.

The model under consideration here used for deformable objects is a network of n masses connected by damped springs which is being deformed by a virtual instrument at a single contact point on the outer surface of the mesh. The outer surface of the mesh is composed of an array of triangular polygons which are constructed using the mass nodes as vertices. The behavior of the system is governed by a non-linear differential equation of the form

$$\frac{d}{dt}\begin{bmatrix} X \\ V \end{bmatrix} = \begin{bmatrix} V \\ M^{-1}f(X,V) \end{bmatrix}, \qquad (1)$$

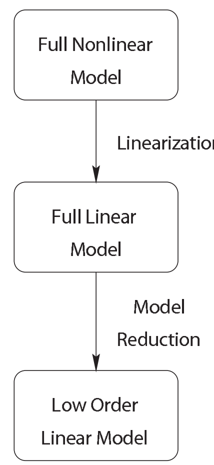

Fig. 2. Construction of the low-order model.

where X and V are respectively the state vectors containing the positions and velocities of each node, $f(X,V)$ is a function that maps the state vectors to a vector of forces on each node, and M^{-1} is a $3p \times 3p$ matrix (p is the dimension of the global model, 3 dimensions for each node) of the form

$$M^{-1} = \mathrm{diag}\left(\frac{1}{\mathrm{m}_1}, \frac{1}{\mathrm{m}_1}, \frac{1}{\mathrm{m}_1}, \ldots \frac{1}{\mathrm{m}_\mathrm{p}}, \frac{1}{\mathrm{m}_\mathrm{p}}, \frac{1}{\mathrm{m}_\mathrm{p}}\right), \qquad (2)$$

where m_i is the mass of the node with index i.

The computational requirements of this relatively simple deformable model prevent it from being simulated at the haptic update rate. Therefore, at each global time step, a linearized discrete model is constructed by taking the tangent behavior of the system

$$\frac{d}{dt}\begin{bmatrix} X \\ V \end{bmatrix} \approx \begin{bmatrix} V \\ M^{-1}\left(f_0 + F\begin{bmatrix} x \\ v \end{bmatrix}\right) \end{bmatrix},$$

where

$$\begin{aligned} f_0 &= f(X_0, V_0) \\ x &= X - X_0 \\ v &= V - V_0 \\ F &= \left[\frac{\partial f}{\partial X} \;\; \frac{\partial f}{\partial V}\right]\Bigg|_{\substack{X=X_0 \\ V=V_0}}, \end{aligned} \qquad (3)$$

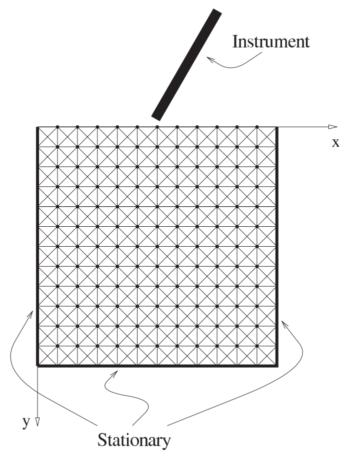

Fig. 3. Two dimensional lumped element mesh.

and X_0, V_0 are the nodal position and velocity vectors at the global current time step.

At this point it is important to note that the linearized system will have the same order as the full model. Therefore, the improvement by just using a linear model is limited and it will still be difficult, if not impossible, to simulate in real time. Therefore, model reduction is the critical step of the approach as it is the means of getting a temporally local haptic model which can be simulated in real time.

2.2. *Order Reduction for Constructing a Low-order Linear Approximation*

In order to evaluate the effectiveness of model reduction, consider a two dimensional critically-damped 12 × 12 lumped element mesh being indented by an instrument (Figure 3). Each node of the mesh has a lumped mass, which is connected to the neighboring nodes (diagonal as well as lateral neighbors) with a spring and dampers. Three edges of the mesh are constrained to be stationary. Linearization of this system gives a two-input two-output 524th order linear dynamical system.

With the input node set at coordinate (50, 0), which is the top middle node, we performed a balanced model reduction on this model (Zhou et al. 1996). The resulting error (due to two degrees of freedom, two states, and five adjacent nodes) of the 20th-order approximation of the input–output response was less than 1% of the full-order model (Table 2, damping ratio 1.0). This is a significant reduction in computational complexity while virtually maintaining the accuracy of the model. The frequency responses of the original and reduced-order systems are shown in Figure 4. The responses of the two systems are essentially indistinguishable except in normal-tangential interactions, where the response magnitudes in both conditions are very small (less than −200 dB).

The results observed above are actually true for a large range of material parameter values. The accuracy of the model reduction with respect to changes in the material properties was examined by varying the stiffness and damping parameters for a 2D mesh with 12 nodes and reduced the full model to a 20th-order model. Stiffness was found to have no effect on the accuracy of the reduced model. Damping, on the other hand, affected the accuracy of the reduced model, as seen in Table 2. However, the error was less than 13.7% for damping ratios higher than 0.1. The damping ratio for each node was defined as $b/\sqrt{2mk}$, where m was 0.001 kg and k represented stiffness.

Deformable materials encountered in most applications, such as surgical simulators, typically have lumped element models with second-order damping ratios higher than 0.1, so this does not pose a major limitation to the proposed method. For instance, Armentano et al. (2007) identified the damping ratio for MSD models of the carotid artery's arterial walls to be 0.7–0.9, while Wakeling and Nigg (2001) reported that the skin covering the quadriceps had a damping ratio in the range of 0.14–0.73.

Table 1. Model Reduction Times and Accuracy of Reduced-order Models. The Reduced 2D Models were 20th Order (Due to Two Degrees of Freedom, Two States, and Five Adjacent Nodes) and the Reduced 3D Models were 102nd Order (Due to Three Degrees of Freedom, Two States, and 17 Adjacent Nodes)

2D Nodes	Full Model Order	Time (s)	% Error
5 × 5	76	0.07	3.89×10^{-6}
10 × 10	356	2.93	3.96×10^{-3}
15 × 15	836	47.14	3.83×10^{-2}
20 × 20	1,516	293.15	2.01×10^{-1}
25 × 25	2,396	1,172.1	3.07×10^{-1}
3D Nodes	**Full Model Order**	**Time (s)**	**% Error**
4 × 4 × 4	210	0.36	1.08×10^{-13}
5 × 5 × 5	474	7.85	4.84×10^{-13}
6 × 6 × 6	894	57.21	1.51×10^{-12}
7 × 7 × 7	1,506	274.98	2.84×10^{-12}
8 × 8 × 8	2,346	1,539.3	5.73×10^{-12}

Table 2. Model Reduction Accuracy for Damping Ratio Variations for a 2D 12 × 12 Mesh Reduced to 20th Order (Due to Two Degrees of Freedom, Two States, and Five Adjacent Nodes)

Damping Ratio	% Error
0.001	51.1
0.01	38.4
0.1	13.7
0.2	4.24
0.6	1.46×10^{-1}
0.7	6.41×10^{-2}
1.0	1.95×10^{-2}
1.5	2.91×10^{-3}
2.0	9.70×10^{-4}

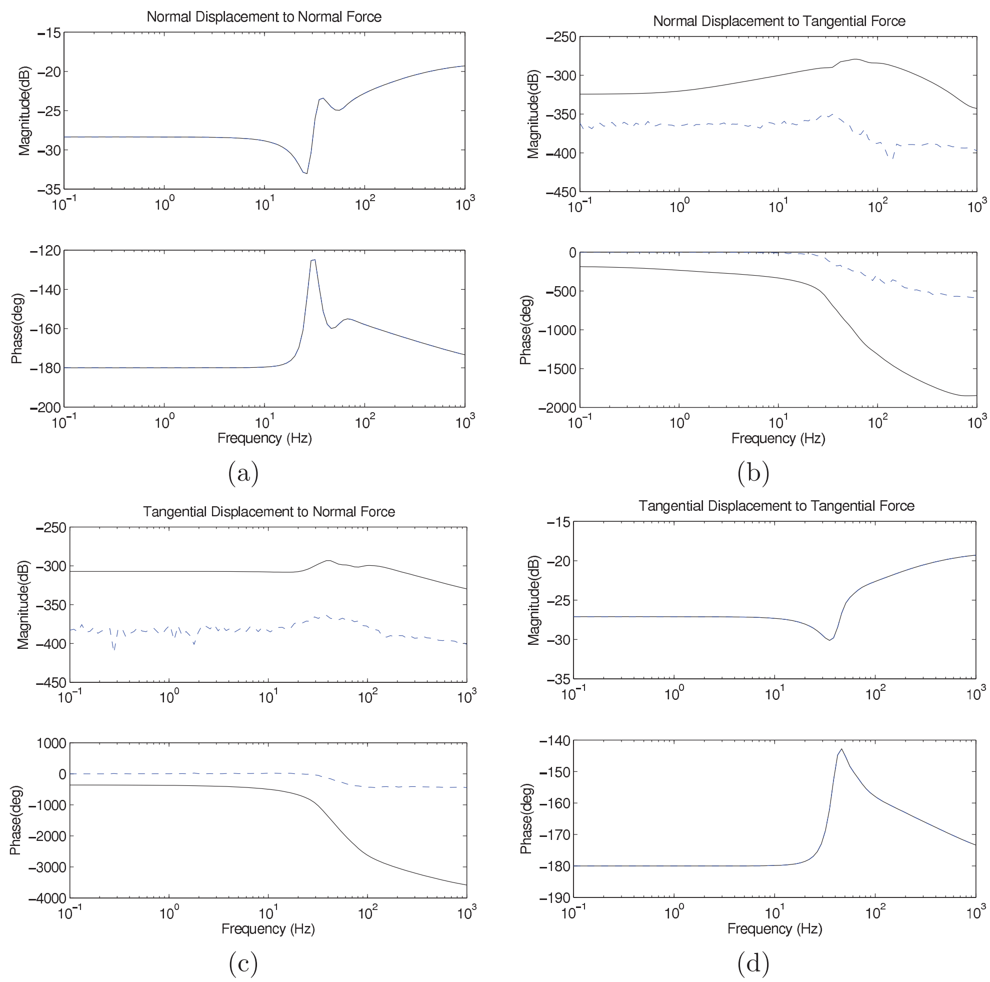

Fig. 4. Frequency responses of the original and reduced-order systems. The solid line is the reduced-order model, the dashed line is the full-order model. (a) Normal displacement to normal force. (b) Normal displacement to tangential force. (c) Tangential displacement to normal force. (d) Tangential displacement to tangential force.

A similar trend is also valid when the mesh density is increased. In Table 1, the error of the reduced models were analyzed as a function of the mesh density in 2D and 3D while maintaining the physical dimensions of the object.

As noted above, balanced model reduction also requires costly calculations. Balanced optimal Hankel model reduction (`sysbal.m` in Matlab) was used and has a complexity of $O(N^3)$, where N is the order of the full-order model. As an illustration of the typical computing times on a modern machine, the computation times for the balanced model reduction of different mesh sizes are reported in Table 1. These computation times were measured on a 2.4 GHz Intel CoreDuo CPU with 3 GB of RAM running Matlab 2009a on the Windows Vista operating system.

The original states of the system before order reduction are the positions and velocities of the lumped masses at the vertices of the mesh. To visualize the spatial properties of the reduced model, the states of the new model are shown in Figure 5. The figure shows the magnitude of the components of the reduced-order model states with respect to the location on the mesh.

The states of the new low-order model show that it is a local approximation. This result is actually expected, because stress decays inversely proportional to the square of the distance from the load in a semi-infinite linear elastic body under a point load (Timoshenko and Goodier 1951).

2.3. *Real-time Algorithm: Construction of a Local Low-order Linear Model*

It is important to note that balanced model reduction also requires costly calculations, which prevents the use of this algorithm as a part of the on-line computation (computation times are investigated in Section 5). However, the analysis in the previous section reveals that the approximation given by the balanced model reduction algorithm in a homogeneous medium is a local model, i.e. the force response depends mostly on the states spatially close to the interaction location. Therefore, a natural way to construct a low-order approximation with significantly less computation is to construct a local linear model directly

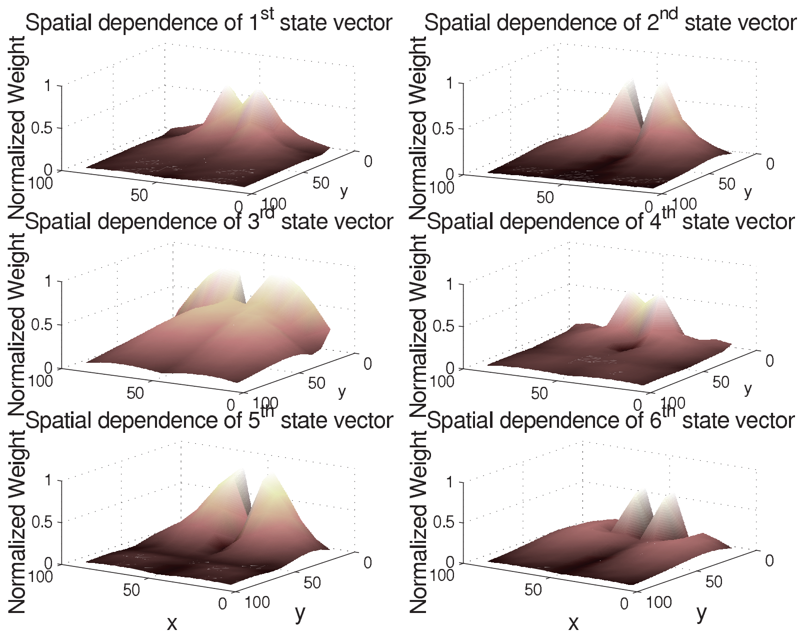

Fig. 5. Spatial dependence of the states of the reduced-order model.

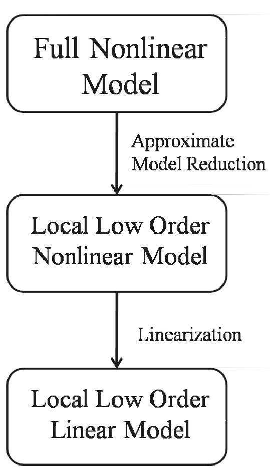

Fig. 6. Real-time construction of the low-order model.

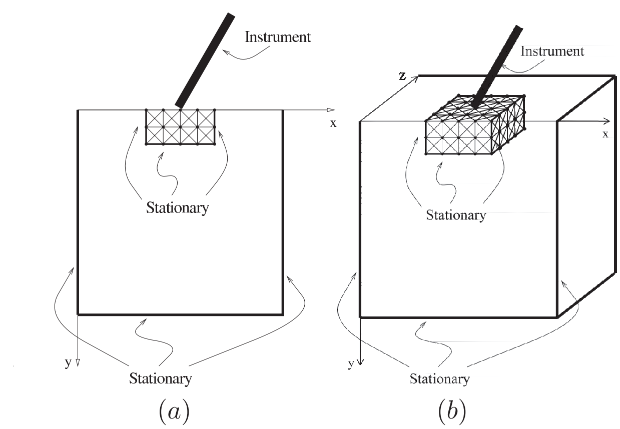

Fig. 7. The (a) 2D 4 × 2 and (b) 3D 4 × 4 × 2 local low-order approximations.

from the full-order model (Figure 6). The 2D and 3D local linear approximations we will demonstrate in this paper are shown in Figure 7. Both are critically damped and model the local behavior of the 4 × 2 2D mesh and 4 × 4 × 2 3D mesh near the instrument contact location. The 2D mesh was used for the frequency response analysis while the 3D mesh was used for the computer simulation experiments.

The frequency response of the local linear approximation, along with the frequency responses of the full linear model and a reduced-order system with the same number of states as the local linear approximation calculated by balanced order reduction, are shown in Figure 8. The local model approximates the behavior in the high frequency range, whereas its DC gain is significantly off. However, it is important to note that the local model is used only to estimate the inter-sample behavior of the full model, and therefore only needs to be close to the full model in the frequency range of around 10–1000 Hz, which is the case here.

These results show that the local linear approximation is a suboptimal approximation, as expected. However, it can be constructed on the fly with minimal computation and give sufficiently accurate behavior in the frequency range of interest. This computation complexity is constant, $O(1)$, and does not depend on the size or order of the underlying full model. For example, the local low-order linear model in the 3D case, shown in Figure 7(b), was implemented in C and constructed in 5.3 ms on a 2.8 GHz Intel Pentium D(TM) processor.

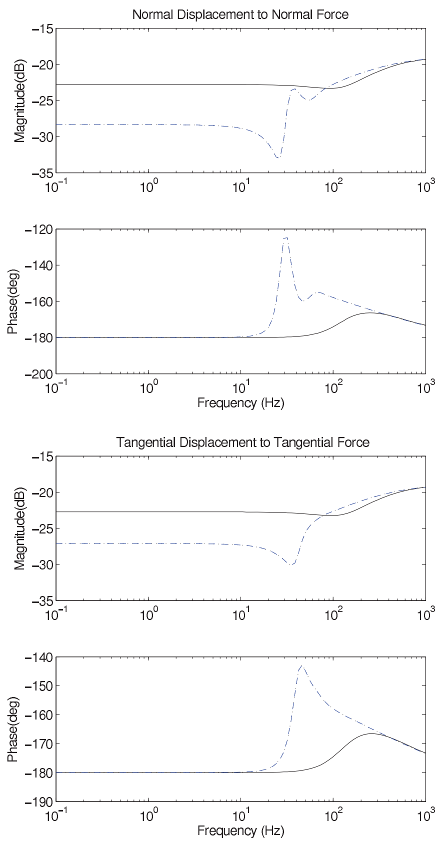

Fig. 8. Frequency responses of the local linear approximation (solid line), full linear model (dashed line) and reduced-order model (dotted line). Note that the dashed and dotted lines overlap. Interactions between the normal and tangential directions are not plotted because the associated magnitudes are very small (less than −200 dB).

Although the basic multi-rate simulation framework enables real-time simulations, it has several drawbacks which can be improved upon. In the next section, the basic framework is extended by taking into account the geometric properties of the local area around the manipulator tip.

3. Multi-rate Simulation with Local Model Geometric Feedback

Typically, multi-rate haptic simulations follow the outline of Figure 1; the flow of information from the local model to the global model is very simple and consists only of position and velocity information from the haptic instrument. Either contact-resolving collision response is done only on the global model, or collision response is performed in parallel on both global and local simulations. Usually, any deformations made to the local model are discarded every time a new local model is constructed.

These types of algorithms have several drawbacks. First, the rate of input from the haptic instrument to the virtual environment is restricted to the global simulation update rate. At first glance this may not seem to be an important constraint since voluntary human hand motion in tasks such as handwriting rarely exceed 6 Hz. In fact a study of eye surgeons pins the figure even lower to 2 Hz (Riviere et al. 1997). However, in the case of sliding frictional contact, an instrument skipping across a rough surface would be restricted from changing mode of contact faster than the visual update rate. Second, it introduces latency between the haptic input and changes in the global model that may result in a haptically apparent discontinuity for the human operator. Also, if the post-haptic-interaction local model is discarded at the end of the haptic time step and replaced with a new local model, which is computed with only the new position of the haptic instrument, the new model may have a large discontinuity in force output. It is possible to interpolate between force values generated at the end of one global time-step with the initial forces generated by the next local model. However, if the discontinuity is large and the global time step is long, then the force output may be perceived as being “muddy” or delayed.

The alternative method presented here is built on the hypothesis that the aforementioned drawbacks can be avoided by including model geometry in the feedback from the local to the global model, as shown in Figure 9. Therefore, any changes to the local model that have been made at the haptic update rate are incorporated back into the global model. This method is motivated by the assumption that haptic fidelity can be improved by performing collision detection and resolution at the haptic rate on a model local to the point of haptic interaction.

The challenging case of sliding frictional contact is used as a testbed for this algorithm. This approach has several difficult requirements: the local model must accurately simulate the behavior of the global model (at least for the short time intervals and small deformations involved), and the collision detection response algorithm needs to operate effectively within the highly demanding 1 ms time-step.

This approach differs significantly from previous work in that the flow of information back from the local to the global model includes the local mesh geometry in addition to haptic instrument position. In addition to this algorithm, the following section’s collision detection and response rules were implemented in order to simulate the effects of coulomb friction at the manipulator tip.

4. Collision Detection and Response in Multi-rate Simulation

The collision detection and response method explored here incorporates a novel local collision detection algorithm. Specifically, the local simulation performs collision detection and response “in-between” frames of the global model. The rationale behind performing collision detection and response at the local level is two-fold. Firstly, it allows for a more realistic high-frequency

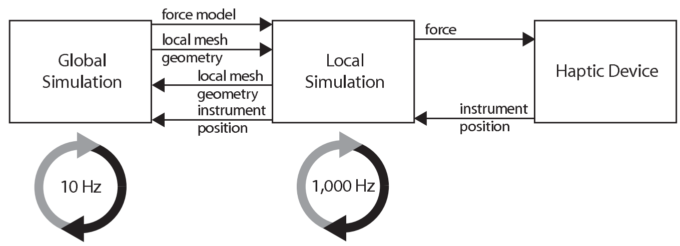

Fig. 9. Multi-rate simulation using a local geometric model in addition to the local dynamic model.

interaction. In particular, intermittent contact, like dragging an instrument across a rough surface, may be negatively affected by latency and the low update rate of the global model. Specifically, in a networked environment, the computer running the local simulation must continue to provide high fidelity interaction during unexpected communication delays with another computer running the global simulation. Maintaining geometric information sufficient to perform collision response can alleviate these issues. Secondly, for simple models, the response of the local model may be superior to the computed response of the global model, as the local model has more detailed information about the trajectory of the instrument during the global model time period.

Two common paradigms are used for collision response algorithms: constraint-based and penalty-based collision response. Constraint-based methods, sometimes referred to as geometrically-based methods, resolve the collision by enforcing constraints that prevent the model from entering an invalid state. For example, in a simple simulation of a ball rolling across a surface, a constraint-based method might apply the constraint that the ball's position in the normal direction must remain above the level of the surface. Constraint-based methods have the disadvantage that they do not directly calculate contact forces, which are necessary for haptic feedback. Penalty-force-based methods operate by connecting virtual springs to interpenetrating objects that pull them apart. Penalty force methods require little computation and compute a contact force suitable for haptic interaction.

For this work, deformable-surface collisions with a haptic instrument were performed using what we call hybrid collision resolution, as described below.

4.1. *Hybrid Collision Resolution in 3D Mass-spring-damper Models*

The hybrid collision resolution algorithm is illustrated in Figure 10. First, a geometric deformation was applied directly to the deformable object (consistent with the constraint-based method). Next, we calculate the "constraint forces" necessary to overcome the internal forces that oppose the deformation of the deformable-surface model. These constraint forces are applied to the nodes of the mesh, and are also used to compute the contact forces applied

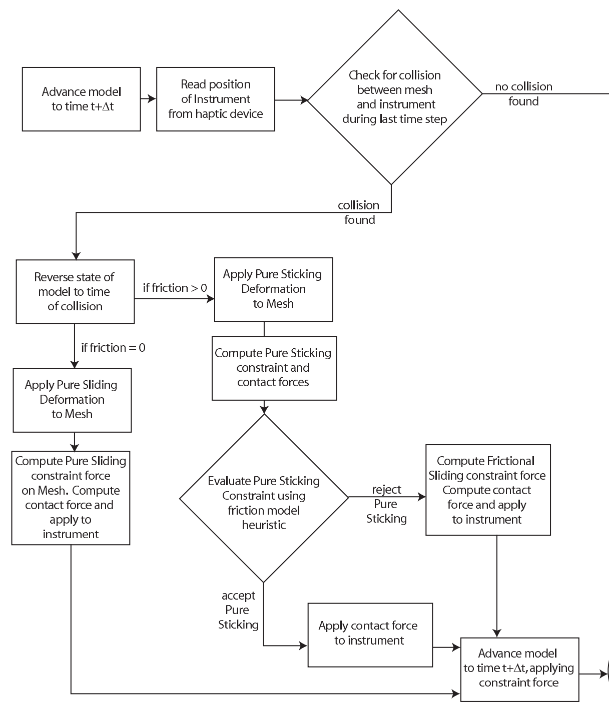

Fig. 10. 3D hybrid collision resolution algorithm flowchart.

to the haptic interface (consistent with the "penalty" force method).

The hybrid collision resolution model uses a Coulomb sliding-friction model. In this model, there are three possible modes of interaction between the haptic instrument and the deformable surface model:

1. Pure sticking – the tip of the instrument drags the contact point of the mesh along with it.
2. Pure sliding – the tip of the instrument pushes down on the mesh, but exerts no force tangential to the mesh surface (the contact is frictionless).
3. Frictional sliding – the tip of the instrument applies both a normal and a tangential force on the surface, but the initial contact point does not remain underneath the instrument tip throughout the contact.

The hybrid collision response algorithm begins by advancing the local simulation to time $t + \Delta t$. The mesh is then checked for a collision between the haptic instrument and a triangle of the mesh during the previous time step. If one is found, the state of the simulation is reversed to the time of collision, $t_c = t + \Delta t_c$, which is identified using binary subdivision of the time step Δt.

In the case of models without friction, the mesh is first deformed by pushing the impacted triangle away from the instrument in the triangle normal direction. The constraint force that will maintain the pure sliding behavior as well as the corresponding contact force are generated (as described in Section 4.2). The model is then advanced by $\Delta t' = \Delta t - \Delta t_c$ to the next time step, $t + \Delta t$, while applying the constraint force to the instrument, which will be haptically perceived by the user. For models with friction, the mesh is first deformed by pushing the triangle away in the direction of instrument motion. The constraint force that will maintain the pure sticking behavior and the corresponding contact force are calculated (Section 4.2), but before applying them, the forces are evaluated using a heuristic to determine if they are consistent with the sliding friction model. The contact force is broken down into components normal (f_n) and tangential (f_t) to the mesh surface. The maximal frictional force consistent with the friction model is calculated as

$$F_{fr} = \mu \|f_n\|. \tag{4}$$

If the magnitude of the tangential force, $\|f_t\|$, is greater than F_{fr}, then the evaluation heuristic rejects the forces as inconsistent with the friction model, and therefore unrealistic for resolving the interaction. In this case, the frictional sliding contact and constraint forces are then computed (Section 4.2) and applied instead. Otherwise, the pure sticking constraint is applied.

4.2. Determination of Constraint Forces

During collision resolution, after the deformation of the mesh, a constraint process is used to determine the contact force. Separate constraints are used in the pure sticking, pure sliding, and frictional sliding cases. In order to identify particular points on the surface of a deformable triangle, baryocentric coordinates are used. Baryocentric coordinates are a concept borrowed from FEM techniques, where a given point P on a triangle is identified by the areas of the sub-triangles formed by extending lines from the node vertices to the interior point (Figure 11). Normalized baryocentric coordinates are used such that the position of an interior point P on the triangle with vertices x_1, x_2, and x_3 can be expressed as

$$p = \alpha x_1 + \beta x_2 + \gamma x_3 \tag{5}$$

where $\alpha = \frac{A_1}{A}$, $\beta = \frac{A_2}{A}$, $\gamma = \frac{A_3}{A}$, A is the total area of the triangle, and (A_1, A_2, A_3) are the areas of the corresponding sub-triangles (Figure 11). As $A = A_1 + A_2 + A_3$, the sum of the baryocentric coordinates of a point is unity, i.e., $\alpha + \beta + \gamma = 1$.

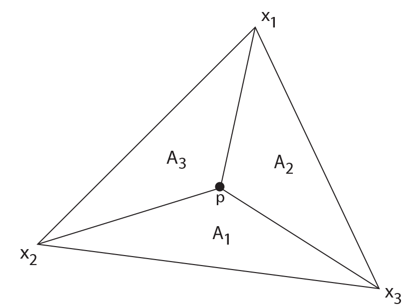

Fig. 11. Baryocentric coordinates.

4.2.1. Sticking Constraint The sticking mode of interaction assumes that the instrument makes contact with a triangle of the mesh at a contact point p, with baryocentric coordinates $p = (\alpha, \beta, \gamma)$, and stays there. Specifically, at the end of the haptic time step, even though the positions of the triangle nodes have changed, the instrument is still in contact with the baryocentric point p. This constraint can be expressed relative to p as

$$\dot{p} = \ddot{p} = 0. \tag{6}$$

The constraint (6) can be enforced by transforming the state vectors from the canonical object coordinates to baryocentric coordinates and canceling the corresponding components of the nodal velocities, as follows.

Let the 9×9 matrix T_{11} be an orthogonal transformation matrix in the form

$$\mathrm{T}_{11} = \begin{bmatrix} \alpha I_{3\times3} & \beta I_{3\times3} & \gamma I_{3\times3} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \end{bmatrix}, \tag{7}$$

where $I_{3\times3}$ is the 3×3 identity matrix, and the empty entries of T_{11} are generated using the Gram–Schmidt method. T is defined to be a $3m \times 3m$ transformation matrix from the canonical object frame into baryocentric coordinates relative to the contact triangle, where m is the number of nodes in the model. If we assume without loss of generality that the contact triangle has nodes x_1, x_2, and x_3, then T is given by

$$\mathrm{T} = \begin{bmatrix} \mathrm{T}_{11} & 0 \\ 0 & I \end{bmatrix} \tag{8}$$

and

$$\mathrm{T} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} p \\ \vdots \\ \vdots \end{bmatrix}. \tag{9}$$

In order to enforce the constraint $\dot{p} = \ddot{p} = 0$, the P operator is defined to be

$$P \equiv \mathrm{T}^{-1} \begin{bmatrix} 0_{3\times3} & 0 \\ 0 & I \end{bmatrix} \mathrm{T}, \tag{10}$$

such that taking the product Pv will cancel out the components of v and cause $\dot{p} \neq 0$ when the simulation is advanced. Then, the discrete equations of motion for advancing the simulation to the beginning of the next time step

$$v(t+\Delta t) = v(t_c) + M^{-1}F\begin{bmatrix} x(t_c) \\ v(t_c) \end{bmatrix}\Delta t'$$
$$x(t+\Delta t) = x(t_c) + v(t_c)\,\Delta t', \tag{11}$$

where

$$v(t_c = t+\Delta t_c) = v(t) + M^{-1}F\begin{bmatrix} x(t) \\ v(t) \end{bmatrix}\Delta t_c$$
$$x(t_c = t+\Delta t_c) = x(t) + v(t+\Delta t_c)\,\Delta t_c, \tag{12}$$

become

$$v(t+\Delta t) = P\left(v(t_c) + M^{-1}F\begin{bmatrix} x(t_c) \\ v(t_c) \end{bmatrix}\Delta t'\right)$$
$$x(t+\Delta t) = x(t_c) + v(t_c)\,\Delta t'. \tag{13}$$

The force on the instrument due to the collision, f_{instr}, is the sum of the forces on each node of the triangle required to maintain the constraint. The operator $\tilde{P}$ is defined to be

$$\tilde{P} \equiv T^{\mathrm{T}}\begin{bmatrix} I_{3\times 3} & 0 \\ 0 & 0 \end{bmatrix}T^{-\mathrm{T}}, \tag{14}$$

noting the reciprocal nature of the nodal velocities and the nodal forces. The force on the instrument is then given by

$$\begin{bmatrix} f_{c1} \\ f_{c2} \\ f_{c3} \\ \vdots \\ \vdots \end{bmatrix} = \tilde{P}f(x,v),$$
$$f_{\text{instr}} = f_{c1} + f_{c2} + f_{c3}. \tag{15}$$

4.2.2. Pure Sliding Constraint The sliding mode of interaction assumes that the instrument makes contact with baryocentric point p on a triangle of the mesh and, at the end of the haptic time step, moves to point on a plane tangential to the triangle. This ensures that the mesh surface can move tangentially to the instrument, but cannot inter-penetrate it at the end of the time step. This is a weaker constraint than the sticking case. The constraint can be expressed relative to p and the triangle normal n as

$$n^{\mathrm{T}}\dot{p} = 0,$$
$$n^{\mathrm{T}}\ddot{p} = 0. \tag{16}$$

Similar to the pure sticking case, the constraint (16) can be enforced by transforming the state vectors from the canonical object coordinates to baryocentric coordinates and canceling the corresponding normal components of the nodal velocities, as follows.

Let U_0 to be a 3×3 orthogonal transformation matrix of the form

$$U_0 = \begin{bmatrix} & n^{\mathrm{T}} & \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \end{bmatrix}, \tag{17}$$

where the empty entries are generated using the Gram–Schmidt method. U_{11} is defined as

$$U_{11} = \begin{bmatrix} U_0 & 0 & 0 \\ 0 & I_{3\times 3} & 0 \\ 0 & 0 & I_{3\times 3} \end{bmatrix}. \tag{18}$$

Let U be a $3m \times 3m$ transformation matrix (where m is the number of dimensions for the local approximation) from the canonical object frame into baryocentric coordinates relative to the contact triangle, assuming again without loss of generality, that the contact triangle has nodes x_1, x_2, and x_3. Then, U is given by

$$U = \begin{bmatrix} U_{11} & 0 \\ 0 & I \end{bmatrix}T = \begin{bmatrix} U_{11}T_{11} & 0 \\ 0 & I \end{bmatrix} \tag{19}$$

where T_{11} is as defined above. In order to enforce the constraint $\dot{p}\cdot n = \ddot{p}\cdot n = 0$, the Q operator is defined to be

$$Q \equiv U^{-1}\begin{bmatrix} 0_{1\times 1} & 0 \\ 0 & I \end{bmatrix}U, \tag{20}$$

such that taking the product Qv cancels out the components of v and causes $n^{\mathrm{T}}\dot{p} \neq 0$ when the simulation is advanced. Then, the discrete equations of motion (11) to advance the simulation to the beginning of the next time step become

$$v_{\text{new}} = Q(v_{\text{old}} + M^{-1}f(x,v)\,\Delta t')$$
$$x_{\text{new}} = x_{\text{old}} + v_{\text{new}}\Delta t'. \tag{21}$$

The force on the instrument due to the collision is the sum of the forces on each node of the triangle required to maintain the constraint. If the $\tilde{Q}$ operator is defined as

$$\tilde{Q} \equiv U^{\mathrm{T}}\begin{bmatrix} 1_{1\times 1} & 0 \\ 0 & 0 \end{bmatrix}U^{-\mathrm{T}}, \tag{22}$$

again, noting the reciprocal nature of the nodal velocities and the nodal forces, then the force on the instrument is given by

$$\begin{bmatrix} f_{c1} \\ f_{c2} \\ f_{c3} \\ \vdots \\ \vdots \end{bmatrix} = \tilde{Q}f(x,v),$$
$$f_{\text{instr}} = f_{c1} + f_{c2} + f_{c3}. \tag{23}$$

4.2.3. Frictional Sliding Contact Constraint The frictional contact constraint is very similar to the sticking contact constraint, except that, during frictional contact, the mesh can slide around under the instrument tip. This is due to the fact that the frictional force is insufficient to cause pure sticking. The constraint forces are calculated as in Section 4.2.1, but then divided into the components normal, f_n, and tangential, f_t, to the triangle surface. The forces tangential to the surface are bounded by $\mu\|f_n\|$, where μ is the surface coefficient of friction, which are applied as before. Once again, we assume without loss of generality that the contact triangle has nodes x_1, x_2, and x_3. The force on the nodes is then given by

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ \vdots \end{bmatrix} = f(x, v), \qquad \begin{bmatrix} \tilde{f}_1 \\ \tilde{f}_2 \\ \tilde{f}_3 \\ \vdots \\ \vdots \end{bmatrix} = \tilde{P}f(x, v) \tag{24}$$

using the definitions presented above for the pure sticking case (Section 4.2.1). For each force $\tilde{f}_i$, let $\tilde{f}_{i_n}$ and $\tilde{f}_{i_t}$ be the components of f_i in the normal and tangential directions relative to the triangle normal n such that

$$\tilde{f}_{i_n} = nn^T\tilde{f}_i \tag{25}$$

$$\tilde{f}_{i_t} = \tilde{f}_i - \tilde{f}_{i_n}. \tag{26}$$

The constraint force $\hat{f}_i$ are given by

$$\hat{f}_i = \tilde{f}_{i_n} + \mu\|\tilde{f}_{i_n}\|\frac{\tilde{f}_{i_t}}{\|\tilde{f}_{i_t}\|} \tag{27}$$

to be the node forces as computed for the sticking constraint, but limited in the tangential direction by the dynamic friction value. The resulting nodal forces for the frictional sliding case, $\mathcal{F}$, is given by

$$\mathcal{F} = \begin{bmatrix} f_1 - \hat{f}_1 \\ f_2 - \hat{f}_2 \\ f_3 - \hat{f}_3 \\ f_4 \\ f_5 \\ \vdots \\ \vdots \end{bmatrix}. \tag{28}$$

The equations of motion for the frictional sliding case are then given by

$$v_{new} = Q(v_{old} + M^{-1}\mathcal{F}\Delta t'),$$
$$x_{new} = x_{old} + v_{new}\Delta t'. \tag{29}$$

In order to determine the instrument force, first let f_{instr} be defined as

$$\begin{bmatrix} f_{c1} \\ f_{c2} \\ f_{c3} \\ \vdots \\ \vdots \end{bmatrix} = \tilde{P}f(x, v)$$
$$f_{instr} = f_{c1} + f_{c2} + f_{c3} \tag{30}$$

as in the pure sticking case. Let f_{instr_n} and f_{instr_t} be the normal and tangential components of that force relative to the triangle normal. Then, the instrument force for the frictional sliding case is

$$\hat{f}_{instr} = f_{instr_n} + \mu\|f_{instr_n}\|\frac{f_{instr_t}}{\|f_{instr_t}\|}. \tag{31}$$

A single time update of the local low-order linear model using the algorithm described above was computed in less than 1 ms on a PC with a 2.4 GHz Intel Pentium 4(TM) processor. Therefore, a 1 kHz haptic update rate was achieved by using the proposed algorithm.

5. Results

The first subsection presents results of the validation of the basic multi-rate algorithm on a 3D 12 × 12 × 12 global mesh approximated to a 4 × 4 × 4 local mesh. The next two subsections examine the performance claims of pure sliding and pure sticking modes of the algorithm, and the final section presents results for the full sliding frictional contact algorithm.

5.1. Comparison of Local and Global Model Simulations

A simulation implementing the proposed multi-rate simulation methods was successfully implemented in C++ using OpenGL as the graphics library. The simulation dynamics were computed using the current algorithm and the graphic rendering was implemented using techniques consistent with Çavuşoğlu et al. (2006). The global simulation was run on an IBM PC running Windows XP, and the haptic simulation was run on an IBM PC running the QNX real-time operating system. The global simulation PC used dual Pentium Xeon(TM) processers running at 2.8 GHz. The local simulation PC used a Pentium 4(TM) processor running at 2.4 GHz. A PHANTOM(TM) version 1.5 manipulator was used as the haptic interface. The two computers were connected by a 100 Mbs ethernet network running a custom network protocol implemented on top of UDP. The computers were connected through a dedicated ethernet switch to eliminate the possibility of interference from other traffic on the network.

Two sets of tests were conducted to validate the basic multi-rate simulation scheme. The tests presented in

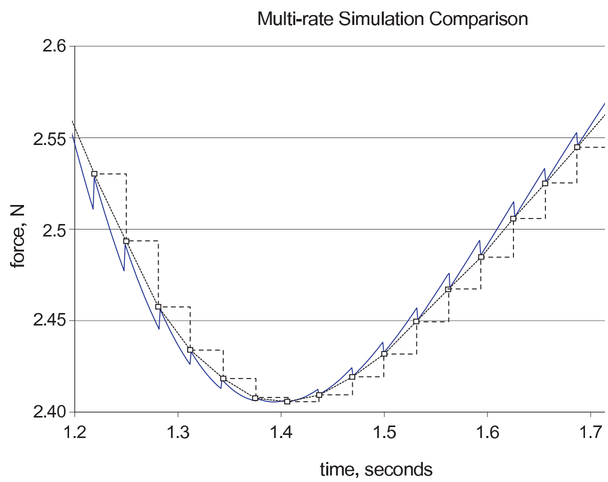

Fig. 12. Multi-rate simulation results. The dotted, smooth line is global model running at 1 kHz. The dashed, jagged line is global model running at 30 Hz. The solid, jagged line is local model running at 1 kHz.

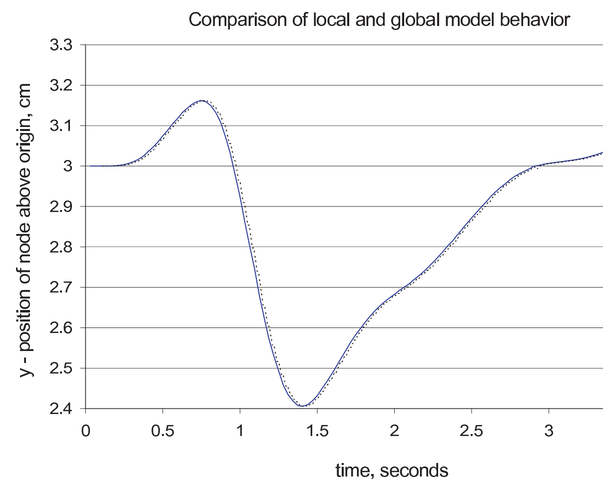

Fig. 13. Multi-rate simulation results, *y*-position of test node versus time. The solid line is global model running at 30 Hz, the dashed line is local model running at 1 kHz.

this subsection were performed using instrument motion specified programmatically, rather than using the haptic device for input (as it was done in the evaluations reported in the subsequent subsections), so that the input could be kept constant between test runs in order to establish the difference in performance between the global and local model simulations.

In the first set of tests, the performance of the multi-rate simulation scheme was compared with a purely global model-based simulation. Three test cases were constructed, each using the same pre-determined instrument motion, in which a single node was pulled 2 cm above the surface of the mesh, and then released. The force exerted on the instrument as the surfaces deforms and presses against it was then computed.

The first test case uses the global model simulation only, updated at 30 Hz as specified in the multi-rate algorithm, but does not run the local model simulation. The second test case also uses the global model simulation only, but updates it at 1 kHz. This is not possible in real time, but this test case provides a base line for comparison of the behavior of the simulation. Finally, the last test case uses the multi-rate simulation technique described above, in which the global model runs at 30 Hz, and the local model running at 1 kHz is used to generate output “between” global updates. The force output during the same part of the trial run is presented in Figure 12.

The global model running at 1 kHz provides the most accurate output; the force follows a smooth sinusoidal curve consistent with the MSD model. The global model running at 30 Hz exhibits the same approximate behavior, but the force output shows a stair-step pattern due to the lower update rate. This method results in a large discontinuity after every global update. The multi-rate simulation follows the tangent behavior of the system between each global update. While there is a discontinuity as each global update is received and a new tangent behavior is calculated, the magnitude of the jump is much lower than the height of the stair-step discontinuity from the global model-only test case.

The second set of tests was performed to determine the effect of local model reduction on the simulation. Since in the multi-rate simulation, part of the surface local to the instrument tip is being simulated using the local model simulation while the rest is being simulated using the global model simulation, tests were performed to establish if this division resulted in inaccuracies in the model behavior. In these tests, the instrument was held immobile over a section of the global model, so that a patch of the surface under the instrument tip was simulated using the local model. The surface was then deformed programmatically by pulling the node under the instrument tip upwards and then releasing it, so that a wave-like motion spread through the model. The instrument was held at a sufficient distance to ensure that it never touched the surface during the trial. The test was then repeated with the instrument moved away from the surface, so that no local model was used, and the entire simulation was performed using the global model. The position of the deformed node along the *y*-axis was then compared between the two tests, as presented in Figure 13.

The behavior of the multi-rate simulation was found to show a high degree of agreement with the global model only simulation, with primarily a delay of one global time step differentiating the two.

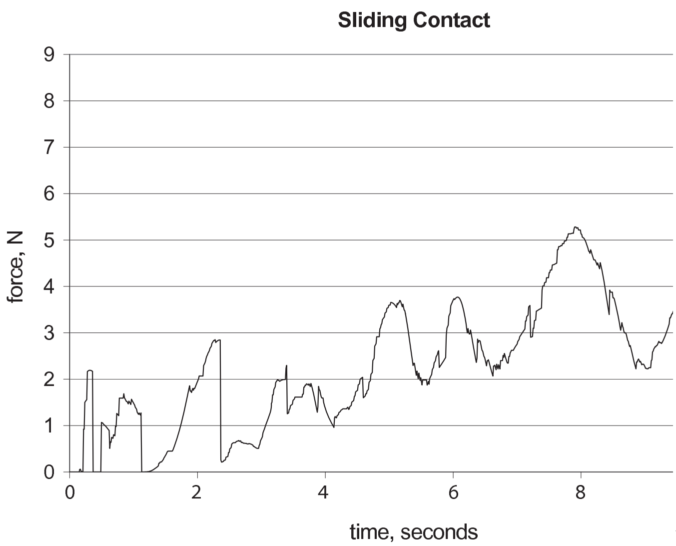

Fig. 14. Pure slipping dragging contact results.

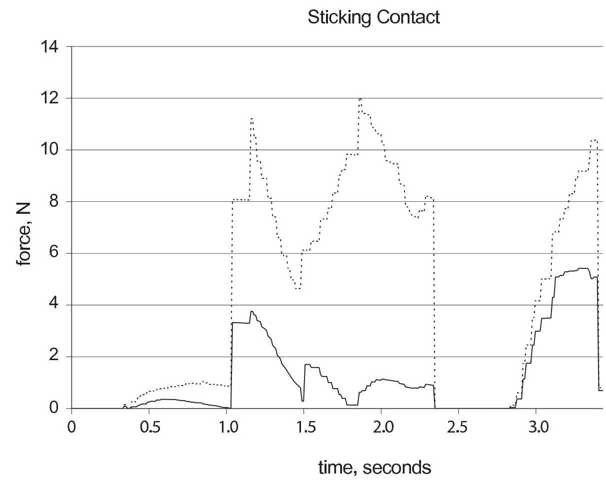

Fig. 15. Pure sticking contact results. The solid line is magnitude of normal force, the dotted line is magnitude of tangential force.

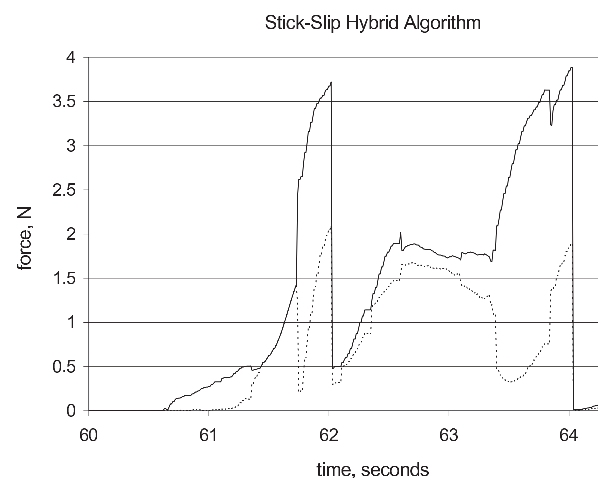

Fig. 16. Hybrid sliding frictional contact dragging results, coefficient of friction 1.0. The solid line is magnitude of force normal, and the dotted line is magnitude of tangential force.

5.2. Pure Slipping Constraint Results

The pure slipping case was tested by manually moving the instrument to one edge of the mesh and dragging it over the surface. Figure 14 shows the magnitude of the force on the instrument normal to the mesh surface. The magnitude of the tangential force response was zero throughout the experiment. When the instrument is pushed into the surface, contact forces are generated in the normal direction but not in the tangential direction, consistent with a zero-friction model.

5.3. Pure Sticking Constraint Results

The dragging contact experiment was also repeated to test pure sticking contact to simulate a "sticky" high co-efficient of friction surface. A tangential as well as a normal force was observed, with a tangential force often larger than the normal, as the motion of the instrument is mostly in the tangential direction.

5.4. Sliding Frictional Contact Results

In this section, the full sliding frictional contact algorithm was tested. The algorithm was allowed to switch between pure sticking and frictional sliding, and the dragging contact experiment presented in the pure sticking constraint results above was repeated. The test was performed with a coefficient of friction (μ) of 1.0, with results as shown in Figure 16. The tangential force is bounded by the normal force, since $f_t \leq \mu f_n$. Therefore, when the force generated by the sticking constraint rises above the normal force, the algorithm switches from pure sticking to frictional sliding, leading to a decrease in force. A typical transition event can be seen at $t = 61.8$. Before the transition, the mesh is being dragged under the instrument tip by the sticking constraint. As the mesh is deformed, the force required to maintain the constraint increases, leading to an increase in both the normal and tangential force. In the pure sticking results presented above, the tangential force increases above μ times the normal force, since the dragging motion is mostly in the tangential direction. However, in the frictional sliding case, the algorithm transitions and the tangential force decreases.

6. Discussion

In this paper, a multi-rate simulation method to handle the difference between the sampling rate requirements of the haptic interfaces and physical models during

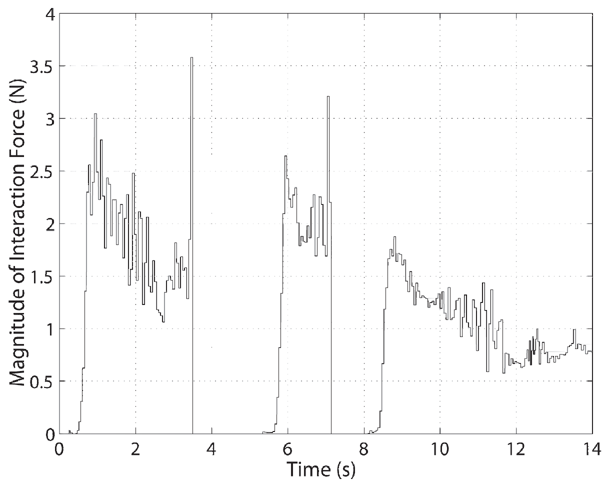

Fig. 17. Oscillations observed when the local linear approximation was not used.

haptic interaction with simulated deformable objects, was presented. The proposed method used model reduction and linear approximation to model the inter-sample behavior of the non-linear full-order model. Based on an order reduction analysis, a simple local linear approximation scheme, which can be computed in real-time, was developed. The method was then extended to achieve high-fidelity rendering of sliding frictional contact by using a local geometric model, in addition to the local dynamic model, and performing collision detection and response as part of the high update rate haptic loop. Experimental results that validated the proposed methods were also presented.

The proposed linearization method was found to produce high quality force output at the haptic update rate. In addition, it was established experimentally that the linearization could be constructed and simulated both within the time constraints of the global and haptic update rates and also with a high degree of accuracy.

Model reduction analysis revealed that the accuracy of the reduced-order models decreased as the mesh density increased and as the damping of the nodes decreased. This was not a surprise since more nodes and less damping would lead to higher-order behavior which the reduced-order models would have difficulty preserving.

MSD models are not physically based, meaning that there is no straightforward method of determining MSD mesh parameters from physical substances, like human tissue. Therefore, evaluation of the algorithm remains largely qualitative. Also, the mesh used in the MSD model was relatively coarse, leading to discontinuities in force when traveling over triangle boundaries.

It is important to note that the local linear approximation used here is not the only choice. For example, it is possible to construct a local model which better approximates the low frequency behavior by extending the outer layer of spring and dampers of Figure 7 to the outer edges of the body and suitably scaling their parameters to reflect the change in equivalent stiffness/damping as the mesh depth changes. These different local approximations and constraints for the local model were explored in detail in Çavuşoğlu (2000).

The major limit of this method is that the local states must be the dominant ones. This could be violated if the material was inhomogeneous, for example, if the deep tissue were significantly more compliant than at the surface so that most of the deformation occurred in states far from the interaction. In this case, the method by Astley and Hayward (1998) would be useful, if the model was linear. Other effects that could violate the dominance of local modes include significant geometric non-linearities or discontinuities in the tissue that produced large local stresses away from the instrument contact. However, the locality of the dominant modes can always be checked by performing off-line model reduction, as shown in Section 2.2.

6.1. *Stability Implications*

Stability of haptic interaction with virtual environments is important in haptic interface design. One of the critical determinants of the stability during interaction is the simulation update rate, where an increase in the update rate of the model improves stability (Minsky et al. 1990; Colgate 1993). In the proposed method, having the low-order linear model running at a faster update improves the stability of the haptic interaction as the virtual environment model runs at 1 kHz instead of 10 Hz. This effect can also be observed in the implementation of our method described above. However, stability of our method is difficult to prove analytically because the resulting system is a multi-rate non-linear sampled-data system. In our simulations, when the local linear approximation was not used, the haptic interface tended to have oscillatory behavior when the operator loosens his/her grip (Figure 17). This oscillatory behavior was not present with the local linear approximation even when the operator completely released the instrument.

6.2. *Future Work*

The limitations found in the testing of the implementation of this project suggest several fruitful avenues for future work. The coarse granularity of the mesh was found to cause discontinuities and loss of contact as the instrument rolls over triangle boundaries. Multi-resolution methods that dynamically re-triangulate the mesh are one possibility to overcome these issues. Work has already begun to adapt methods from this study to the GiPSi simulation framework, an open source simulation framework for surgical simulation with haptic feedback (Çavuşoğlu et al. 2006).

Acknowledgements

This work was supported by the National Science Foundation, USA (grant numbers CISE IIS-0222743, CISE EIA-0329811, CISE CNS-0423253, and CISE IIS-0805495) and by the US Department of Commerce (grant number TOP-39-60-04003).

References

Armentano, R. L., Barra, J. G., Pessana, F. M., Craiem, D. O., Graf, S., Santana, D. B. and Sanchez, R. A. (2007). Smart smooth muscle spring dampers. *IEEE Engineering in Medicine and Biology Magazine*, 26: 62–70.

Armstrong, B. and Hayward, V. (2000). A new computational model of friction applied to haptic rendering. *Experimental Robotics VI* (*Lecture Notes in Computer Science*, Vol. 250). New York, Springer, pp. 404–412.

Astley, O. R. and Hayward, V. (1998). Multirate haptic simulation achieved by coupling finite element meshes through Norton equivalents. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'98)*, pp. 989–994.

Barbagli, F., Prattichizzo, D., and Salisbury, K. (2005). A multirate approach to haptic interaction with deformable objects, single and multipoint contacts. *The International Journal of Robotics Research*, 24(9): 703–715.

Bro-Nielsen, M. (1998). Finite element modeling in surgery simulation. *Proceedings of the IEEE*, 86(3): 490–503.

Campion, G. and Hayward, V. (2008). On the synthesis of haptic textures. *IEEE Transactions on Robotics*, 24(3): 527–536.

Çavuşoğlu, M. C. (2000). Multirate simulation for high fidelity haptic interaction with deformable objects in virtual environments. *Technical report, Electronics Research Laboratory Memo M00/5*, University of California at Berkeley.

Çavuşoğlu, M. C., Goktekin, T. and Tendick, F. (2006). GiPSi: A framework for open source/open architecture software development for organ level surgical simulation. *IEEE Transactions on Information Technology in Biomedicine*, 10(2): 312–321.

Çavuşoğlu, M. C. and Tendick, F. (2000). Multirate simulation for high fidelity haptic interaction with deformable objects in virtual environments. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2000)*, pp. 2458–2465.

Cho, C., Kim, M., Hwang, C., Lee, J. and Song, J. (2005). Stable haptic display of slowly updated virtual environment with multirate wave transform. *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 2465–2470.

Choi, S. and Tan, H. (2004). Discrimination of virtual haptic textures rendered with different update rates. *Proceedings of the First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pp. 114–119.

Colgate, J. E. (1993). Robust impedance shaping telemanipulation. *IEEE Transactions on Robotics and Automation*, 9(4): 374–384.

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

d'Aulignac, D., Balaniuk, R. and Laugier, C. (2000). A haptic interface for a virtual exam of the human thigh. *IEEE International Conference on Robotics and Automation*, pp. 2452–2457.

James, D. and Pai, D. K. (2003). Multiresolution Green's function methods for interactive simulation with force feedback. *ACM Transactions on Graphics*, 22(1): 47–82.

James, D. L. and Pai, D. K. (1999). ARTDEFO: Accurate real time deformable objects. *Proceedings of SIGGRAPH 99: 26th International Conference on Computer Graphics and Interactive Techniques*, pp. 65–72.

Kim, J., De, S. and Srinivasan, M. (2002). Computationally efficient techniques for real time surgical simulation with force feedback. *Proceedings of the 10th Symposium on Haptic Interfaces For Virtual Environment and Teleoperator Systems (HAPTICS'02)*, pp. 51–57.

Mahvash, M. (2006). Novel approach for modeling separation forces between deformable bodies. *IEEE Transactions on Information Technology in Biomedicine*, 10(3): 618–626.

Mahvash, M. and Hayward, V. (2002). Haptic rendering of tool contact. *Proceedings of Eurohaptics*, pp. 110–115.

Mahvash, M. and Hayward, V. (2004). High fidelity haptic synthesis of contact with deformable bodies. *IEEE Computer Graphics and Applications*, 24: 48–55.

Mahvash, M. and Hayward, V. (2005). High fidelity passive force-reflecting virtual environments. *IEEE Transactions on Robotics*, 21(1): 38–46.

Minsky, M., Ouh-young, M., Steele, O., F. P. Brooks, J. and Behensky, M. (1990). Feeling and seeing: Issues in force display. *Proceedings of the 1990 Symposium on Interactive 3D Graphics* New York, ACM, pp. 235–243.

Renaud, C. and Feng, Z.-Q. (2003). BEM and FEM analysis of Signorini contact problems with friction. *Computational Mechanics*, 31: 390–399.

Riviere, C., Rader, R. and Khosla, P. (1997). Characteristics of hand motion of eye surgeons. *Proceedings of the 19th Annual Conference of the IEEE Engineering in Medicine and Biology Society*.

Sundaraj, K., Mendoza, C. and Laugier, C. (2002). A fast method to simulate virtual deformable objects with force feedback. *7th International Conference on Control, Automation, Robotics, and Vision (ICARCV 2002)*, pp. 413–418.

Terzopoulos, D., Platt, J., Barr, A. and Fleischer, K. (1987). Elastically deformable models. *Proceedings of SIGGRAPH 87: 14th Annual Conference on Computer Graphics*, pp. 205–214.

Timoshenko, S. and Goodier, J. N. (1951). *Theory of Elasticity*, 2nd edn. New York, McGraw-Hill. pp. 214–215.

Wakeling, J. M. and Nigg, B. M. (2001). Soft-tissue vibrations in the quadriceps measured with skin mounted transducers. *Journal of Biomechanics*, 34: 549–543.

Wu, X. and Tendick, F. (2004). Multigrid integration for interactive deformable body simulation. *Proceedings of the International Symposium on Medical Simulation (ISMS 2004)*, (*Lecture Notes in Computer Science*, Vol. 3078), Berlin, Springer, pp. 92–104.

Zhang, J., Payandeh, S. and Dill, J. (2002). Haptic subdivision: an approach to defining level-of-detail in haptic rendering. *Proceedings of the 10th Symposium on Haptic Interfaces For Virtual Environment and Teleoperator Systems (HAPTICS'02)*, pp. 201–208.

Zhou, K., Doyle, J. C. and Glover, K. (1996). *Robust and Optimal Control*. Prentice-Hall, NJ. pp. 155–172.