2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

High Fidelity Haptic Rendering of Stick-Slip Frictional Contact With Deformable Objects in Virtual Environments Using Multi-Rate Simulation

Paul Jacobs
Department of Electrical Engineering and
Computer Science
Case Western Reserve University
10900 Euclid Avenue, Cleveland, OH 44022,
USA
pxj18@cwru.edu

M. Cenk Cavusoglu
Department of Electrical Engineering and
Computer Science
Case Western Reserve University
10900 Euclid Avenue, Cleveland, OH 44022,
USA
cavusoglu@cwru.edu

***Abstract*— An increasingly common new modality in human-computer interaction is haptic interfacing, especially in the field of medical simulation. The order-of-magnitude difference in update rates between graphical deformable object simulations and haptic interfaces can be bridged using local low-order approximations. However, providing force feedback using local models complicates collision detection and response with the virtual tool, since the user interacts with lower-order proxies rather than the full simulated objects. A novel approach focusing on rolling contact with stick-slip friction is presented where all collision detection and response with the virtual tool is performed at the local level at the haptic time-scale, utilizing linearized low-order local models that approximate the behavior of the full model for the short time steps and small deformations involved.**

I. 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. The key obstacle to overcome in haptic interaction is the difference in update rates of the simulation, which typically is linked to the graphical update rate of between 10 and 60 Hz, and the update rate requirement of the haptic interface, which must be on the order of 1 kHz in order to be convincing to the operator. One common method of bridging this gulf is through multi-rate simulation, in which the virtual environment in its full complexity is simulated at the visual update rate, while a simpler simulation, often encompassing only parts of the environment local to the virtual instrument, is run in parallel at the haptic update rate and periodically re-synchronized with the full model [4]. The purpose of the work presented here is to explore multi-rate simulation techniques involving performing collision response using a stick-slip friction model in the haptic-rate simulation.

Several different types of deformable surface models have been suggested for use in haptic interaction. Lumped mass-spring-damper (MSD) models represent deformable objects with layers of masses connected by spring-damper pairs or damped springs [9]. MSD models are a common choice for surgical simulation for their ease and simplicity in modeling and for their low computational requirements. The main alternative to MSD systems are finite-element models (FEM), which provide a continuum model that is strongly physically based [10]. Other volumetric methods have been suggested to capture the realism of FEM techniques with lower computational requirements such as the method of finite spheres [9] and the "Long Element Model" [10]. Although these models have been found to be useful in haptic simulation, the discussion here is limited to the more widely used mass-spring-damper and FEM models.

A major difficulty in providing high-fidelity force feedback is the extremely demanding update rate requirements of haptic feedback. The human sense of touch is remarkably sensitive, and can distinguish between changes in force into the range of hundreds of Hz. It is generally accepted that the update rate of the haptic interface must be ten to twenty times higher than the highest frequency event that is to be simulated. Therefore, in order to render events in the 50 to 100 Hz range matching the capabilities of the PHANTOM and similar haptic interfaces, 1 kHz is widely considered the minimum update rate for realistic haptic feedback [6]. Real-time virtual environment simulations, on the other hand, are tailored towards visual update rates, and typically run between 10 - 60Hz, 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 rate has been the focus of the bulk of haptic research. As of the late 1990's, the typical practice was to low-pass filter the generated force to the bandwidth of the model update rate [13]. This was found to not be an adequate solution, as this effectively reduced 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, 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

1-4244-0602-1/07/$20.00 ©2007 IEEE.

and more computationally tractable model to provide only force feedback. Commonly, the visual update rate simulation models the entire virtual environment, while the haptic rate simulation models only an area local to the point of haptic interaction. Cavusoglu and Tendick used a local linearization of mass-spring-damper models, and showed that the spacial dependence of the state variables of each node of the linearized system is primarily on nearby nodes, and therefore a local model consisting of only a few layers surrounding the contact point can be used to provide satisfactory haptic interaction [4]. Multi-rate simulation techniques have also been developed that combine local model approaches and virtual coupling to handle both single and multi-point contact [2]. Control methods to avoid instability in multi-rate environments caused by time delay in communication channel have recently been examined [5]. Multi-resolution methods with coarser meshes used for the entire model and more detailed meshes for areas of local interest have also been used successfully in multi-rate simulation [1][14][8].

II. Multi-Rate Simulation Algorithm

The core of the approach to multi-rate simulation in this work 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. A 'local' simulation runs at the haptic server loop update rate, and simulates the behavior of a subset of the global model.

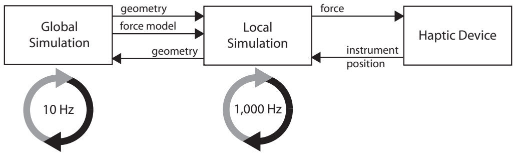

Fig. 1. **Haptic Simulation Block Diagram**

After each global update is generated at 100 ms intervals corresponding to the 10 Hz update rate, a local linear 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 local linear approximation model to provide force output to the user and then sends the state of the local model and haptic instrument back to the global model, which incorporates this information back into the global model, and then recomputes a new linear approximation for the next cycle.

III. Linearized Low-Order Approximations of Lumped Mass-Spring-Damper Models

At each global time-step, a local linearized approximation model of the deformable surface is constructed and sent to the local model simulation for simulating the dynamic behavior at the haptic update rate. In this work, mass-spring-damper networks are used as the global model. A similar treatment is possible using finite element models but not presented here.

The model used for deformable objects under consideration here 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

$$\frac{d}{dt}\left[\begin{array}{c} X \\ \dot{X} \end{array}\right] = \left[\begin{array}{c} \dot{X} \\ M^{-1} f(X, \dot{X}) \end{array}\right] \tag{1}$$

where X and $\dot{X}$ are respectively the state vectors containing the positions and velocities of each node, $f(X, \dot{X})$ is a function that maps the state vectors to a vector of forces on each node, and M^{-1} is an $3n \times 3n$ matrix of the form

$$M^{-1} = diag\left(\frac{1}{m_1}, \frac{1}{m_1}, \frac{1}{m_1}, \ldots \frac{1}{m_n}, \frac{1}{m_n}, \frac{1}{m_n}\right) \tag{2}$$

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:

$$\frac{d}{dt}\left[\begin{array}{c} X \\ \dot{X} \end{array}\right] \approx \left[\begin{array}{c} \dot{X} \\ M^{-1}\left(f_0 + F\left[\begin{array}{c} \Delta X \\ \Delta \dot{X} \end{array}\right]\right) \end{array}\right] \tag{3}$$

where

$$\begin{array}{rcl} f_0 & = & f(X_0, \dot{X}_0) \\ \Delta X & = & X - X_0 \\ \Delta \dot{X} & = & \dot{X} - \dot{X}_0 \\ F & = & \left[\begin{array}{cc} \frac{\partial f}{\partial X} & \frac{\partial f}{\partial \dot{X}} \end{array}\right]\Bigg|_{\substack{X=X_0 \\ \dot{X}=\dot{X}_0}} \end{array} \tag{4}$$

The linearization process does not reduce the order of the model, and so a reduction of the model to components spacially close to the virtual instrument contact point is needed to reduce the computational complexity so as to arrive at a model that can be simulated at the haptic update rate. The method used here to construct a low order approximation follows the method of Cavusoglu and Tendick that includes in the low order approximation the closest node on the surface of the mesh to the contact point and the immediately adjacent nodes, as well as all the springs and dampers connected to any of these nodes [4]. The springs and dampers connecting these nodes to the rest of the deformable model are replaced with springs and dampers with the same parameters connected to a stationary and immovable 'wall' with the justification that the high frequency response of the model depends mostly on contributions from the nodes that are immediate neighbors, and only secondarily on the rest of the mass-spring-damper network, and so for the high-frequency dependent haptic interaction, the remainder of the nodes can be considered stationary during each haptic time step.

IV. Collision Detection and Response in Multi-Rate Simulation

The method explored here incorporates a novel local collision detection algorithm in which 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 twofold. First, it allows for a more realistic high-frequency 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. In a networked environment in particular, the local side must continue to provide high fidelity interaction during unexpected delays. 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 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 is 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, but have several important limitations. Penalty force methods have difficulty taking friction into account, have trouble with very rigid objects, and are typically hard to generalize to multi-point contact between objects [11].

For this simulation, a hybrid approach is used for haptic instrument-deformable surface collisions that applies a geometric deformation to the deformable object directly, as in constraint-based methods, and then computes a 'constraint force' necessary to overcome the internal forces of the deformable surface model that oppose the deformation. This constraint force is applied to the nodes of the mesh, and is also used to compute a contact force that is applied to the haptic interface, much as a 'penalty' force would be.

A. Hybrid Collision Resolution in 3D MSD models

The hybrid collision resolution model uses a simplified stick-slip 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.

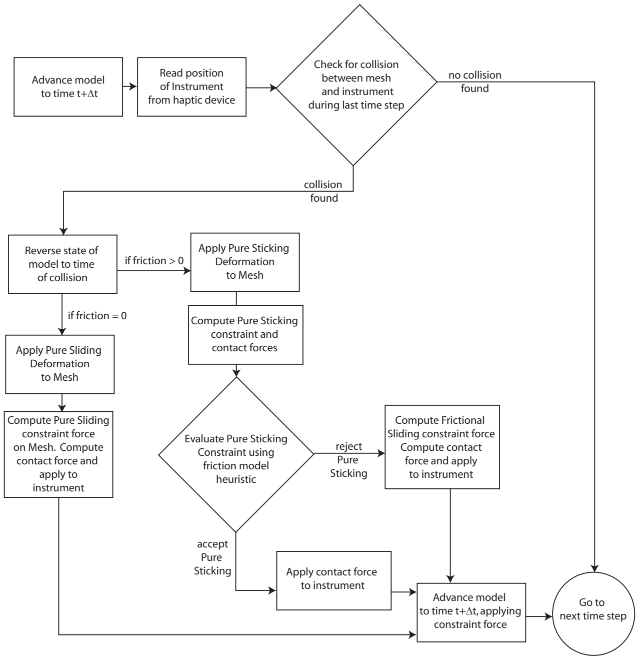
Advance model to time t+Δt
Read position of Instrument from haptic device
Check for collision between mesh and instrument during last time step
no collision found
collision found
Reverse state of model to time of collision
if friction > 0
Apply Pure Sticking Deformation to Mesh
if friction = 0
Compute Pure Sticking constraint and contact forces
Apply Pure Sliding Deformation to Mesh
Compute Pure Sliding constraint force on Mesh. Compute contact force and apply to instrument
Evaluate Pure Sticking Constraint using friction model heuristic
reject Pure Sticking
Compute Frictional Sliding constraint force Compute contact force and apply to instrument
accept Pure Sticking
Apply contact force to instrument
Advance model to time t+Δt, applying constraint force
Go to next time step

Fig. 2. **Three dimensional hybrid collision resolution model flowchart**

The hybrid collision response algorithm, illustrated in Figure 2, begins by advancing the global simulation to time $t + \Delta t$. The mesh is then checked for a collision between the instrument and a triangle of the mesh during the previous time step, and if one is found, the state of the simulation is reversed to the time of collision.

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. A constraint force is generated that will maintain the Pure Sliding behavior, as well as a corresponding contact force. The model is advanced to the next time step while applying the constraint force to the instrument, which will be haptically felt by the user. For models with friction, the mesh is first deformed by pushing the triangle away in the direction of instrument motion. A constraint force that will maintain the Pure Sticking behavior and a corresponding contact force are calculated, but before applying them, the forces are evaluated using a heuristic to determine if they are consistant with the stick-slip friction model. The contact force is broken down into forces normal (F_N) and tangential (F_T) to the mesh surface. The maximal frictional force consistent with the friction model is calculated as

$$F_{fr} = \mu F_N \tag{5}$$

If the force F_T is greater than F_{fr}, then the evaluation heuristic rejects the forces as inconsistent with the friction model, and therefore unrealistic to use to resolve the interaction, and the Frictional Sliding contact and constraint forces are then computed and applied instead. Otherwise, the Pure Sticking constraint is applied.

V. Determination of Constraint Forces

A. Sticking Constraint

The sticking mode of interaction assumes that the instrument makes contact with a triangle of the mesh at a contact point with baryocentric co-ordinate p and that at the end of the haptic time step, the instrument is still in contact with a point of the triangle with the same baryocentric co-ordinate relative to the new positions of the nodes of the same triangle. This constraint can be expressed relative to p as

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

Let the 9×9 matrix T_{11} be an orthogonal basis in baryocentric co-ordinates given by

$$T_{11} = \begin{bmatrix} \alpha I_{3\times3} & \beta I_{3\times3} & \gamma I_{3\times3} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \end{bmatrix} \tag{7}$$

where $I_{3\times3}$ is the 3×3 identity matrix, and the empty entries of T_{11} are generated using the Gram-Shmidt method. Define T to be an $3m\times3m$ transformation matrix from the canonical object frame into baryocentric co-ordinates relative to the contact triangle where m is the number of nodes in the model. We assume without loss of generality that the contact triangle has nodes x_1, x_2, and x_3. Then, T is given by

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

and

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

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

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

When computing the discrete equations of motion to advance the simulation to the beginning of the next time step, the equations of motion

$$\begin{aligned} \dot{x}_{new} &= \dot{x}_{old} + M^{-1} f(x,\dot{x})\Delta t \\ x_{new} &= x_{old} + \dot{x}_{new}\Delta t \end{aligned} \tag{11}$$

become

$$\begin{aligned} \dot{x}_{new} &= P(\dot{x}_{old} + M^{-1} f(x,\dot{x})\Delta t) \\ x_{new} &= x_{old} + \dot{x}_{new}\Delta t \end{aligned} \tag{12}$$

The force on the instrument due to the collision is assumed to be the sum of the force on each node of the triangle required to maintain the constraint. Define the $\tilde{P}$ operator to be

$$\tilde{P} \equiv T^T \begin{bmatrix} I_{3\times3} & 0 \\ 0 & 0 \end{bmatrix} T^{-T} \tag{13}$$

The force on the instrument is then given by

$$\begin{aligned} \begin{bmatrix} f_{c1} \\ f_{c2} \\ f_{c3} \\ \vdots \\ \vdots \end{bmatrix} &= \tilde{P} f(x,\dot{x}) \\ f_{instr} &= f_{c1} + f_{c2} + f_{c3} \end{aligned} \tag{14}$$

B. Pure Sliding Constraint

The sliding mode of interaction assumes that the instrument makes contact with a triangle of the mesh at a contact point with barycentric co-ordinate p, and at the end of the haptic time step, the contact point has moved only in 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

$$\begin{aligned} \dot{p}\cdot n &= 0 \\ \ddot{p}\cdot n &= 0 \end{aligned} \tag{15}$$

Define U_0 to be a 3×3 orthogonal matrix of the form

$$U_0 = \begin{bmatrix} & n^T & \\ \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots \end{bmatrix} \tag{16}$$

where the empty entries are generated using the Gram-Shmidt method. Define

$$U_{11} = \begin{bmatrix} U_0 & 0 & 0 \\ 0 & I_{3\times3} & 0 \\ 0 & 0 & I_{3\times3} \end{bmatrix} \tag{17}$$

Let U be an $3m \times 3m$ transformation matrix from the canonical object frame into baryocentric co-ordinates relative to the contact triangle, where m is the number of nodes in the model. We assume without loss of generality that the contact triangle has nodes x_1, x_2, and x_3. Then, U is given by

$$U = \begin{bmatrix} U_{11}T_{11} & 0 \\ 0 & I \end{bmatrix} \tag{18}$$

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

$$Q \equiv U^{-1} \begin{bmatrix} 0_{1\times1} & 0 \\ 0 & I \end{bmatrix} U \tag{19}$$

When computing the discrete equations of motion to advance the simulation to the beginning of the next time step, the equations of motion become

$$\begin{aligned} \dot{x}_{new} &= Q(\dot{x}_{old} + M^{-1} f(x, \dot{x}) \Delta t) \\ x_{new} &= x_{old} + \dot{x}_{new} \Delta t \end{aligned} \tag{20}$$

The force on the instrument due to the collision is assumed to be the sum of the force on each node of the triangle required to maintain the constraint. Define the $\tilde{Q}$ operator to be

$$\tilde{Q} \equiv U^T \begin{bmatrix} 1_{1\times1} & 0 \\ 0 & 0 \end{bmatrix} U^{-T} \tag{21}$$

The force on the instrument is then given by

$$\begin{aligned} \begin{bmatrix} f_{c1} \\ f_{c2} \\ f_{c3} \\ \vdots \\ \vdots \end{bmatrix} &= \tilde{Q} f(x, \dot{x}) \\ f_{instr} &= f_{c1} + f_{c2} + f_{c3} \end{aligned} \tag{22}$$

C. Frictional Sliding Contact Constraint

The frictional contact constraint is the same as the sticking contact constraint, except that the constraint forces are divided into forces normal, F_N, and tangential, F_T to the triangle surface. The forces tangential to the node surface are bounded by μF_N, where μ is the surface coefficient of friction, and then are applied as before. Since the sticking mesh deformation is used for frictional sliding, during frictional sliding, the instrument pushes the mesh surface, but the mesh can still slide around under the instrument tip, since the frictional force is insufficient to cause pure sticking. 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

$$\begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ \vdots \end{bmatrix} = f(x, \dot{x}) \qquad \begin{bmatrix} \tilde{f}_1 \\ \tilde{f}_2 \\ \tilde{f}_3 \\ \vdots \\ \vdots \end{bmatrix} = \tilde{P} f(x, \dot{x}) \tag{23}$$

using the definitions presented above in the Pure Sticking section. For each force f_i, let f_{n_i} and f_{t_i} be the components of f_i in the normal and tangential directions relative to the triangle normal n. Then the constraint force $\hat{f}_i$ are given by

$$\hat{f}_i = f_{n_i} + \mu \| f_{n_i} \| \frac{f_{t_i}}{\| f_{t_i} \|} \tag{24}$$

to be the node forces as computed for the sticking constraint. Then, the node forces for the frictional sliding case, $\mathcal{F}$, is given by

$$\mathcal{F} = \begin{bmatrix} f_1 - \hat{f}_1 \\ f_2 - \hat{f}_2 \\ f_3 - \hat{f}_3 \\ f_4 \\ f_5 \\ \vdots \\ \vdots \end{bmatrix} \tag{25}$$

The equations of motion for the frictional sliding case are then

$$\begin{aligned} \dot{x}_{new} &= Q(\dot{x}_{old} + M^{-1} \mathcal{F} \Delta t) \\ x_{new} &= x_{old} + \dot{x}_{new} \Delta t \end{aligned} \tag{26}$$

To determine the instrument force, first let f_{instr} be defined as

$$\begin{aligned} \begin{bmatrix} f_{c1} \\ f_{c2} \\ f_{c3} \\ \vdots \\ \vdots \end{bmatrix} &= \tilde{P} f(x, \dot{x}) \\ f_{instr} &= f_{c1} + f_{c2} + f_{c3} \end{aligned} \tag{27}$$

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

$$\hat{f}_{instr} = f_{instr_{n}} + \mu \| f_{instr_{n}} \| \frac{f_{instr_{t}}}{\| f_{instr_{t}} \|} \tag{28}$$

VI. Results

A simulation implementing these methods was successfully implemented in C++ using OpenGL as the graphics library. The global simulation was run on a 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 an 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. 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 full stick-slip friction algorithm.

A. Pure Slipping Constraint Results

The pure slipping case was tested by moving the instrument to one edge of the mesh and dragging it over the surface. Figure 3 shows the magnitude of the force on the instrument normal to the mesh surface. The magnitude of the tangential force response is 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.

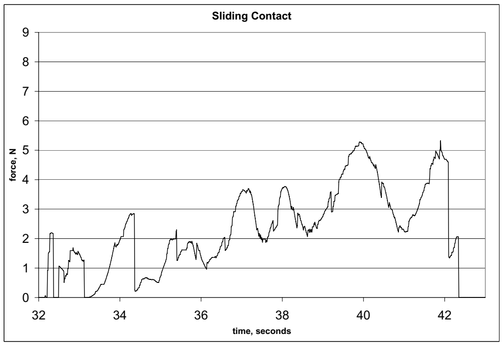

Fig. 3. **Pure Slipping dragging contact results.**

B. 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.

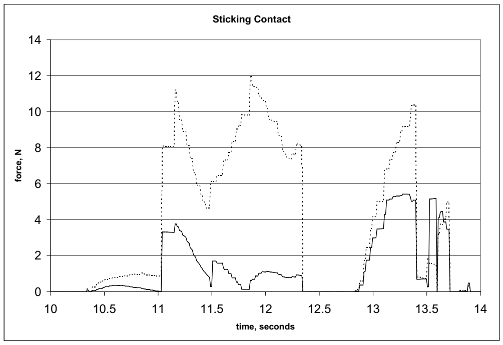

Fig. 4. **Pure Sticking contact results. Solid line is magnitude of normal force, dotted line is magnitude of tangential force.**

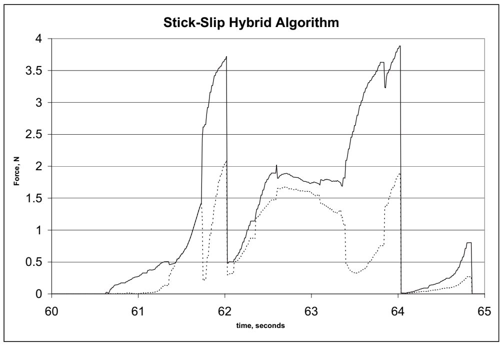

Fig. 5. **Hybrid stick-slip dragging results, co-efficient of friction 1.0. Solid line is magnitude of force normal, dotted line is magnitude of tangential force.**

C. Stick-Slip Frictional Sliding Friction Results

In this section, the full stick-slip 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 co-efficient μ of friction μ of 1.0, with results as shown in Figure 5. 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, instead the algorithm transitions and the tangential force decreases.

VII. Discussion

A. Extensions to Literature

The new linearization method for Mass-Spring-Damper models is an improvement over existing methods of building local approximations. In addition, the stick-slip collision response offers a new direction in contact resolution in haptic environments.

The 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.

B. Limitations

Several limitations qualify the results of this study. First, haptic virtual environments with deformable surfaces remain rare, and detailed results from an alternative implementation from another research group with which to compare these results were not available. This is compounded by the fact that MSD models are not physically based, meaning that there is no straightforward method of determining MSD mesh parameters from physical substances, like human tissue, except for approximate methods, such as [7], [12]. 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.

C. Conclusion

A novel method of generating linear approximations of MSD models was designed, implemented, and successfully tested. A haptic update rate contact-resolution algorithm was also test that provided a satisfactory haptic experience.

D. Future Work

The limitations found in 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 [3].

Acknowledgments

This work was supported in part by National Science Foundation under grants CISE IIS-0222743, EIA-0329811, and CNS-0423253, and US DoC under grant TOP-39-60-04003.

References

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

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

[3] M. C. Çavuşoğlu, T. Goktekin, and F. Tendick. GiPSi: A framework for open source/open architecture software development for organ level surgical simulation. *Submitted to the IEEE Transactions on Information Technology in Biomedicine*, 2005. (Under Review).

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

[5] C. Cho, M. Kim, C.S. Hwang, J. Lee, and J.B. Song. Stable haptic display of slowly updated virtual environment with multirate wave transform. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2465–2470, April 2005.

[6] S. Choi and H.Z. Tan. Discrimination of virtual haptic textures rendered with different update rates. In *Proceedings of the First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 114–119, March 2004.

[7] A. Van Gelder. Approximate simulation of elastic membranes by triangulated. spring meshes. *Journal of Graphics Tools*, 3(2):21–42, September 1998.

[8] D. James and D. K. Pai. Multiresolution green's function methods for interactive simulation with force feedback. In *ACM Transactions on Graphics*, pages 47–82, 2003.

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

[10] J. Kim, S. De, and M.A. Srinivasan. A fast method to simulate virtual deformable objects with force feedback. In *7th International Conference on Control, Automation, Robotics, and Vision (ICARCV 2002)*, pages 413–418, 2002.

[11] C. Lennerz, E. Schoemer, and T. Warken. A framework for collision detection and response. In *Proceedings of the 11th European Simulations Symposium*, pages 309–314, 1999.

[12] C. Paloc, F. Bello, R. I. Kitney, and A. Darzi. Online multiresolution volumetric mass spring model for real time soft tissue deformation. In *Lecture Notes in Computer Science*, volume 2489, pages 219–226. Springer, 2002.

[13] F. Tendick, M. Downes, T. Goktekin, M. C. Çavuşoğlu, D. Feygin, X. Wu, R. Eyal, M. Hegarty, and L. W. Way. A virtual environment testbed for training laparoscopic surgical skills. *Presence*, 9(3):236–255, June 2000.

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