

## Networked haptic cooperation using remote dynamic proxies

Zhi Li

Department of Mechanical Engineering  
University of Victoria  
Victoria, BC, Canada  
zhil@me.uvic.ca

Daniela Constantinescu

Department of Mechanical Engineering  
University of Victoria  
Victoria, BC, Canada  
Danielac@me.uvic.ca

### Abstract

*Networked haptic cooperation entails direct interaction among users as well as joint manipulation of virtual objects. To increase the realism of both types of interactions, this paper introduces remote dynamic proxies. Remote dynamic proxies are second order dynamic representations of users at the remote peer sites. They are generated according to dynamics laws and are controlled by the user whom they represent through a virtual coupler. Hence, they move in a physically intuitive manner and do not suffer from position discontinuities due to network packet transmission limitations. The remote dynamic proxies are integrated into a distributed control architecture for networked haptic cooperation. An experimental comparison of the new controller to two recently proposed controllers demonstrates smoother rendering of contact between users, as well as stable cooperation for larger network delays.*

### 1. Introduction

Networked haptic cooperation finds its applications in a variety of areas, including surgical training [9], telerehabilitation [15], and computer games. Multiple users haptically present in a virtual environment at the same time want: (1) to manipulate virtual objects together; and (2) to touch and feel each other. For example, an experienced surgeon and a remote resident may want to interact with a virtual organ simultaneously during surgical teletraining. A therapist may want to guide the hand of a remote patient during telerehabilitation.

Cooperative manipulation of virtual objects by multiple remote users has been the primary focus of existing haptics research. Both centralized (client-server) [6, 13] and distributed (peer-to-peer) [1, 3, 8, 7, 13, 14] controllers have been proposed. Distributed controllers have been shown to render higher contact stiffness [6] and maintain better

position coherency among copies of the shared virtual object [13] than centralized controllers.

Several distributed control architectures have been developed to combat the destabilizing effect of network delay on cooperative haptic interaction. In [14], three peer-to-peer haptic cooperation approaches have been compared. In those approaches, users manipulate their local copies of the shared virtual object and the controller coordinates among object copies at all peer sites. Stability and position coherency via virtual coupling [4], time domain passivity [12] and wave variables [10] coordination have been studied. The investigation in [14] has shown that: virtual coupling control is most sensitive to network delay; time domain passivity control may not be able to prevent distracting oscillations; wave variable control renders much smaller forces and results in much larger position errors between shared object copies than the other controllers. In [6], kinematic representations of the remote users have been provided at all peer sites. This has alleviated the effect of the communication delay on users' perception of their peers' interaction with the shared virtual object. Coherency among all copies of the shared object has been maintained via virtual couplers. Although not developed for this purpose specifically, the architecture in [6] can render both cooperative manipulations and direct user-to-user interaction.

Direct haptic interaction among multiple users has motivated the introduction of massless proxies with first order dynamics in [9]. Compared to the conventional, purely kinematic proxies [16, 11], the motion of the dynamic proxies can be better controlled during collisions with fixed virtual objects and among multiple proxies. The performance of proxies with first order dynamics during networked haptic cooperation has not been investigated. Proxies with second order dynamics have been proposed in [5]. For single user interaction with a slow virtual environment, the second order dynamic proxies have mitigated the effect of computational delay on the stability of the interaction and on user's perception of rigid contact.

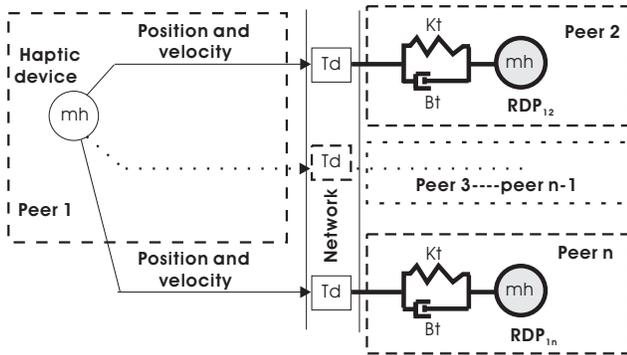
This paper introduces a distributed control architec-

ture for networked haptic cooperation that enables multiple users to manipulate virtual objects together as well as to interact with each other directly. The architecture uses the new concept of remote dynamic proxies to render contact between users. Remote dynamic proxies are second order dynamic representations of remote users at peer sites. As shown through experiments, they can be used to render physically intuitive direct interaction among users, as well as high contact stiffness in the presence of significant network delay.

The remote dynamic proxy concept is presented in Section 2 of this paper. The networked haptic cooperation architecture with remote dynamic proxies is developed for two peers in Section 3. Its performance is experimentally compared to the performance of the distributed control architectures with virtual coupling coordination from [6, 14] in Section 4. Conclusions and directions for future work end the paper in Section 5.

## 2. The remote dynamic proxy

To interact with each other directly, networked users need information about their remote peers. This section proposes to provide such information via remote dynamic proxies. As illustrated in Figure 1, a remote dynamic proxy is a proxy of a remote user in the local copy of the shared virtual environment of one of its peers. In particular, the remote dynamic proxy  $RDP_{12}$  represents Peer 1 in the virtual environment of Peer 2. Remote dynamic proxies have second order dynamics and are controlled by the user whom they represent via virtual coupling<sup>1</sup>. All remote dynamic proxies of a user have the same geometric features (for collision detection) and the same inertial and damping properties (for dynamic behavior) as those of the haptic device of the peer they embody.



**Figure 1. Remote dynamic proxies (RDPs).**

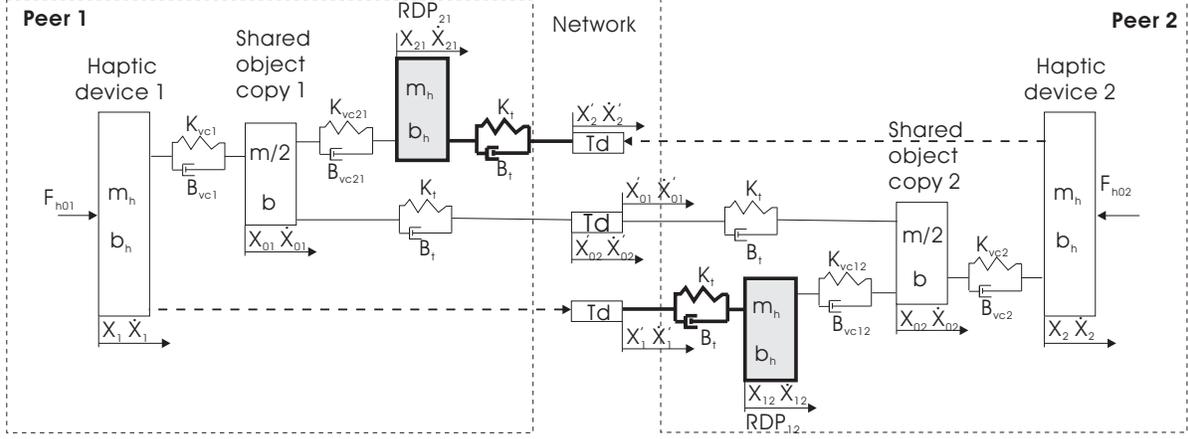
<sup>1</sup>In Figure 1, the virtual coupling is depicted as the bolded spring  $K_t$  and damper  $B_t$  between a remote dynamic proxy and the delayed position and velocity of the user whom it represents.

A key feature of remote dynamic proxies is that they render smooth user motion at peer sites regardless of the network packet transmission rate. During haptic interaction among remote users, each haptic device updates the position of the local peer at the frequency of the force control loop, typically of the order of the order of 1KHz. Updated positions of remote users are available at much lower rates that depend on the proficiency of the network packet transmission, network traffic, etc. This leads to discontinuities in the position of remote peers and, therefore, may destabilize the interaction. Because remote dynamic proxies are rendered according to dynamic laws, they smooth remote user position discontinuities. As validated through experiments in Section 4, smooth proxy motion mitigates the destabilizing effect of network delay on haptic cooperation.

## 3. Control architecture of networked haptic cooperation with remote dynamic proxies

This section introduces the distributed control architecture for networked haptic cooperation that incorporates the remote dynamic proxies discussed in Section 2. For simplicity of exposition and because a two-user experimental setup is employed in Section 4, the architecture is detailed for interaction between two peers herein. However, it can be directly extended to the interaction among multiple users.

The proposed distributed architecture is shown in Figure 2. In this architecture, the virtual environment of each peer comprises: (1) a copy of the virtual object which they cooperatively manipulate; and (2) a remote dynamic proxy of their peer,  $RDP_{12}$  and  $RDP_{21}$ . The network communication is represented via the network delay  $T_d$  between the two peers. The various stiffness and damping parameters characterize the various contact interactions:  $K_{vc1}$  and  $B_{vc1}$ , and  $K_{vc2}$  and  $B_{vc2}$  characterize the contact between the local user and their copy of the shared virtual object at Peer 1 and at Peer 2, respectively;  $K_{vc12}$  and  $B_{vc12}$ , and  $K_{vc21}$  and  $B_{vc21}$  characterize the contact between the remote virtual proxy and the local copy of the virtual object at Peer 1 and at Peer 2, respectively. Moreover,  $K_t$  and  $B_t$  represent the coordination gains between the two copies of the shared virtual object, as well as the gains of the control of the two users over their remote dynamic proxies. Note that the mass of the shared virtual object  $m$  is equally distributed between its two local copies, whereas the remote dynamic proxies have the same mass  $m_h$  as the haptic device which they represent. Kinematics notation is used as follows:  $x_1$  and  $x_2$ , and  $\dot{x}_1$  and  $\dot{x}_2$  are the positions and velocities of the two haptic devices;  $x_{12}$  and  $x_{21}$ , and  $\dot{x}_{12}$  and  $\dot{x}_{21}$  are the positions and velocities of the two remote dynamic proxies;  $x_{01}$  and  $x_{02}$ , and  $\dot{x}_{01}$  and  $\dot{x}_{02}$  are the positions and velocities of the local copies of the virtual cube; and an apostrophe notes a delayed signal available at its peer site, e.g.,  $x'_1$  is the



**Figure 2. Distributed control architecture with remote dynamic proxies.**

delayed position of haptic device 1 available at the Peer 2 site.

The dynamics of the networked haptic cooperation rendered via the distributed control architecture with remote dynamic proxies depicted in Figure 2 are:

- for the peer haptic devices:

$$F_{h01} - F_{vc1} = m_h \ddot{x}_1 + b_h \dot{x}_1 \quad (1)$$

$$F_{h02} - F_{vc2} = m_h \ddot{x}_2 + b_h \dot{x}_2 \quad (2)$$

- for the remote dynamic proxies:

$$F_{t12} - F_{vc12} = m_h \ddot{x}_{12} + b_h \dot{x}_{12} \quad (3)$$

$$F_{vc21} - F_{t21} = m_h \ddot{x}_{21} + b_h \dot{x}_{21} \quad (4)$$

- for the local copies of the virtual object:

$$F_{vc1} - F_{t1} - F_{vc21} = \frac{m}{2} \ddot{x}_{01} + b \dot{x}_{01} \quad (5)$$

$$F_{t2} - F_{vc2} + F_{12} = \frac{m}{2} \ddot{x}_{02} + b \dot{x}_{02} \quad (6)$$

where:

$$F_{vc1} = K_{vc1}(x_1 - x_{01}) + B_{vc1}(\dot{x}_1 - \dot{x}_{01}) \quad (7)$$

$$F_{vc2} = K_{vc2}(x_2 - x_{02}) + B_{vc2}(\dot{x}_2 - \dot{x}_{02}) \quad (8)$$

$$F_{t1} = K_t(x_{01} - x'_{02}) + B_t(\dot{x}_{01} - \dot{x}'_{02}) \quad (9)$$

$$F_{t2} = K_t(x'_{01} - x_{02}) + B_t(\dot{x}'_{01} - \dot{x}_{02}) \quad (10)$$

$$F_{vc12} = K_{vc12}(x_{12} - x_{02}) + B_{vc12}(\dot{x}_{12} - \dot{x}_{02}) \quad (11)$$

$$F_{vc21} = K_{vc21}(x_{01} - x_{21}) + B_{vc21}(\dot{x}_{01} - \dot{x}_{21}) \quad (12)$$

$$F_{t21} = K_t(x_{21} - x'_2) + B_t(\dot{x}_{21} - \dot{x}'_2) \quad (13)$$

$$F_{t12} = K_t(x'_1 - x_{02}) + B_t(\dot{x}'_1 - \dot{x}_{02}) \quad (14)$$

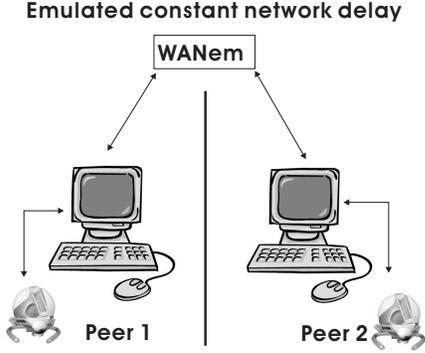
These dynamics are implemented on the networked haptic cooperation setup described in the following section, and used to investigate the performance of the distributed architecture with remote dynamic proxies via controlled experiments.

## 4. Experiments

### 4.1. Experimental setup

Figure 3 illustrates the experimental networked haptic cooperation system. The system uses two personal computers: one runs Window XP on an Intel Core 2 Duo CPU at 2.67GHz with 2 GB RAM; another one runs Window Vista on an Intel Core 2 Duo CPU at 1.67GHz with 3 GB RAM. A FALCON NOVINT haptic device is connected to each computer. The two devices provide 3DOF displacement sensing and force rendering and, thus, support point interaction in 3DOF virtual environments. The two computers are located in the same laboratory and can be screened visually from each other to prevent users to see each other's display. Copies of a shared virtual environment comprising a virtual cube in a rigid enclosure have been generated on each computer as C++ console applications. The two computers communicate over Internet using the UDP protocol and a Wide Area Network Emulator (WANem) [2] is used to simulate a constant network delay between them. The WANem runs on a separate personal computer. The position sensing and force rendering rate of the FALCON NOVINT haptic devices is 1KHz. The data transmission rate is 128Hz.

In the following sections, this networked haptic setup is used to contrast the controller with remote dynamic proxies introduced in Section 3 to the controllers with virtual



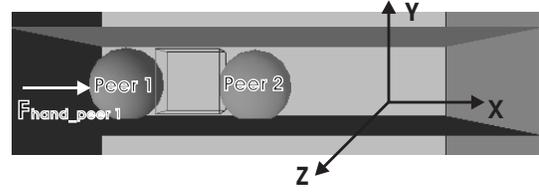
**Figure 3. The experimental networked haptic cooperation system.**

coupling coordination proposed in [14] and [6]. To make data collected during successive cooperative manipulations comparable, the experiments are controlled. In other words, the two human peers are replaced by forces applied to each FALCON NOVINT device through commands sent to motors via software. This effectively eliminates the inherent damping of the peers' hands from the interaction. Since the two haptic interfaces are impedance type devices, the controlled forces have no stabilizing effect compared to user-applied forces.

#### 4.2. Experiment I - cooperative manipulation

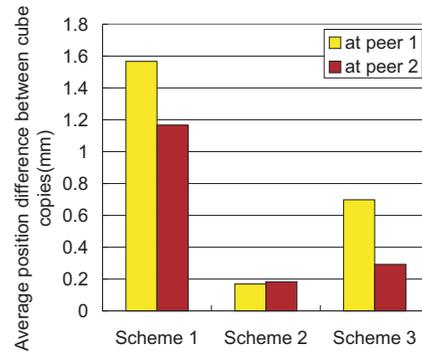
The first experiment investigates position coherency between the local copies of the rigid virtual cube in the rigid enclosure depicted in Figure 4. This figure is the snapshot of the screen of Peer 1 at the beginning of the experiment. The rigid enclosure is designed to ensure the same initial conditions among the successive experiments and to limit the interaction to the x-axis. The two users are initially at rest and in contact with the virtual cube. During the experiment, Peer 1 pushes the virtual cube and Peer 2 with a constant force  $F_{hand\_peer1} = 5N$ , whereas Peer 2 applies no force onto the shared cube. The various controller gains, chosen as in [14] and [6], are:  $K_{vc1} = K_{vc2} = K_{vc12} = K_{vc21} = 4000N/m$ ;  $B_{vc1} = B_{vc2} = B_{vc12} = B_{vc21} = 10Ns/m$ ;  $K_t = 2000N/m$ ;  $B_t = 300Ns/m$ . The mass of the virtual cube is  $m = 0.25kg$ , and the mass of the remote dynamic proxies is  $m_h = 0.01kg$ . Damping is incorporated neither in the remote dynamic proxies nor in the virtual cube, i.e.,  $b_h = b = 0Ns/m$ . A network delay  $T_d = 50ms$  is emulated via the WANem.

The three control architectures being contrasted via the experimental cooperative manipulation of the virtual cube in Figure 4 are: (1) the peer-to-peer architecture with virtual coupling coordination between cube copies investi-



**Figure 4. Snapshot of the beginning of Experiment I (cooperative manipulation) at Peer 1.**

gated in [14], hereafter called Scheme 1; (2) the distributed architecture with kinematic representation of the remote user in the local virtual environment and with virtual coupling coordination between cube copies employed in [6], hereafter called Scheme 2; and (3) the new peer-to-peer architecture with remote dynamic proxies introduced in Section 3, hereafter called Scheme 3. The experimental results are summarized in Figure 5.



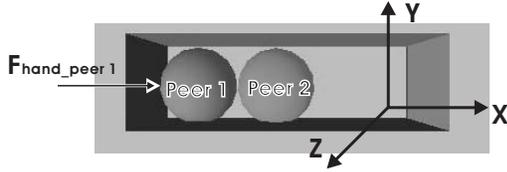
**Figure 5. Position coherency for haptic cooperation via: Scheme 1 [14]; Scheme 2 [6]; new Scheme 3 (remote dynamic proxies).**

Figure 5 illustrates that all three architectures maintain position coherency between the two copies of the virtual cube for the selected controller parameters. Scheme 3 (proposed in this paper) has higher performance than Scheme 1 [14]. Scheme 2 [6] has the best position coherency among the three architectures.

#### 4.3. Experiment II - user-to-user contact

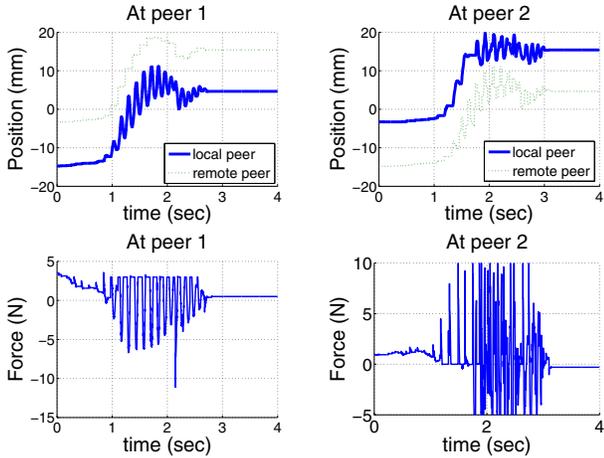
The second experiment demonstrates the rendering of direct user-to-user contact via Scheme 2 [6] and via the new Scheme 3 with remote dynamic proxies proposed in this work. Scheme 1 [14] is not included because it permits users to interact only via a shared virtual object.

Figure 6 is the snapshot of the screen of Peer 1 at the beginning of this second experiment. The two users are initially at rest and in contact with each other. During the experiment, Peer 1 pushes Peer 2 with constant force  $F_{hand\_peer1} = 3N$ , whereas Peer 2 applies no force onto Peer 1. The controller gains, chosen as in [6], are:  $K_{vc1} = K_{vc2} = K_{vc12} = K_{vc21} = 2000N/m$ ;  $B_{vc1} = B_{vc2} = B_{vc12} = B_{vc21} = 10Ns/m$ ;  $K_t = 2000N/m$ ;  $B_t = 10Ns/m$ . The mass of the virtual cube is  $m = 0.25kg$ , and the mass of the remote dynamic proxies is  $m_h = 0.01kg$ . Damping is incorporated neither in the remote dynamic proxies nor in the virtual cube, i.e.,  $b_h = b = 0Ns/m$ . A network delay  $T_d = 10ms$  is emulated via the WANem.

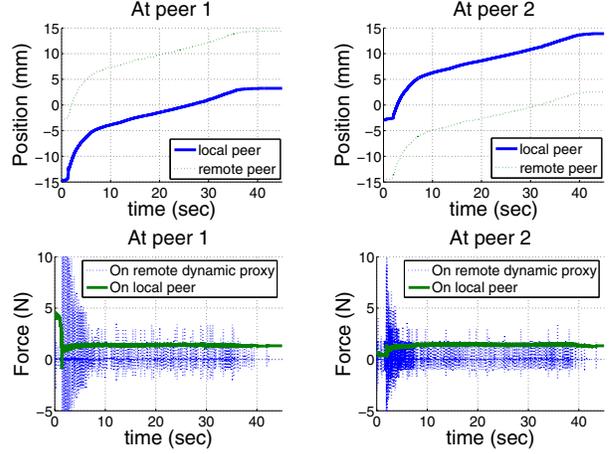


**Figure 6. Snapshot of the beginning of Experiment II (user-to-user contact) at Peer 1.**

The positions of the two peers and the forces they perceive are shown in Figure 7 for Scheme 2 [6], and in Figure 8 for Scheme 3 (the distributed architecture with remote dynamic proxies). Note in these figures that Scheme 2 renders intermittent contact, whereas Scheme 3 renders continuous and smooth contact. Hence, remote dynamic proxies improve users' perception of direct contact with peers.

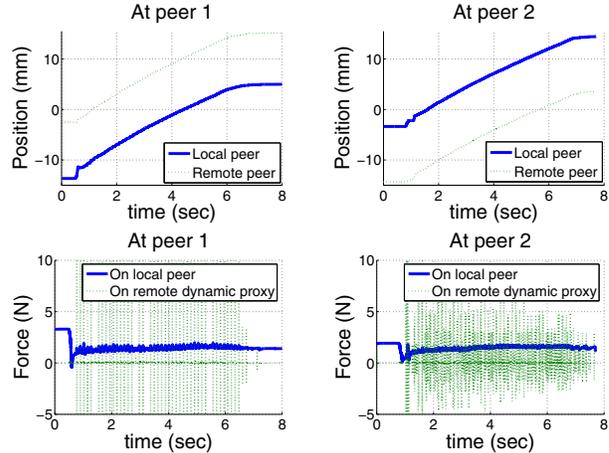


**Figure 7. User-to-user contact rendered via the distributed architecture in [6].**  $K_{vc1} = K_{vc2} = K_{vc12} = K_{vc21} = 2000N/m$ ;  $T_d = 10ms$ .

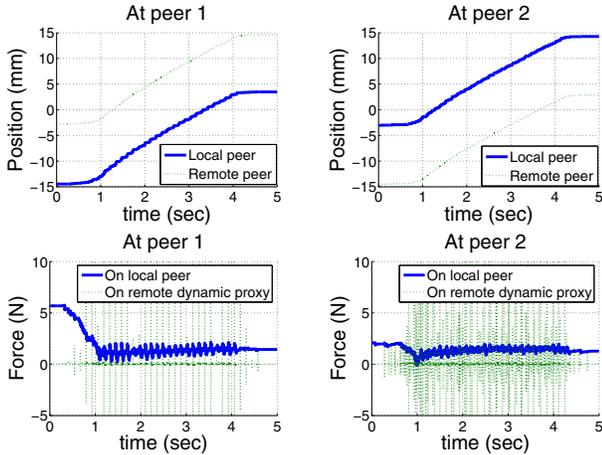


**Figure 8. User-to-user contact rendered using remote dynamic proxies.**  $K_{vc1} = K_{vc2} = K_{vc12} = K_{vc21} = 2000N/m$ ;  $T_d = 10ms$ .

To validate that remote dynamic proxies can be used to render stable user-to-user contact for larger network delay and higher contact stiffness, Experiment II is repeated for: (1) longer network delay  $T_d = 200ms$  (Figure 9); and (2) stiffer contact stiffness  $K_{vc1} = K_{vc2} = K_{vc12} = K_{vc21} = 4000N/m$  (Figure 10). For both these sets of parameters, the user-to-user interaction rendered via Scheme 2 [6] is unstable. In contrast, Figures 9 and 10 demonstrate that the two users perceive stable and smooth contact with their peer when the remote interaction is rendered via remote dynamic proxies.



**Figure 9. User-to-user contact rendered using remote dynamic proxies.**  $K_{vc1} = K_{vc2} = K_{vc12} = K_{vc21} = 2000N/m$ ;  $T_d = 200ms$ .



**Figure 10. User-to-user contact rendered using remote dynamic proxies.**  $K_{vc1} = K_{vc2} = K_{vc12} = K_{vc21} = 4000\text{N/m}$ ;  $T_d = 10\text{ms}$ .

## 5. Conclusion

This paper has introduced a distributed control architecture for networked haptic cooperation that enables multiple users to manipulate virtual objects together as well as to interact with each other directly. The proposed architecture uses remote dynamic proxies to render contact between users. Remote dynamic proxies are second order dynamic representations of users at their peer sites. They are controlled by the remote user whom they represent through virtual coupling. The distributed controller with remote dynamic proxies has been compared to existing distributed controllers via two experiments. The first, cooperative manipulation experiment has demonstrated that distributed control with remote dynamic proxies enforces position coherency between copies of the shared virtual object similarly to existing controllers. The second experiment has shown that distributed control with remote dynamic proxies enables users to perceive smooth direct user-to-user contact and higher contact stiffness under longer network delay.

Although it supports more realistic networked haptic cooperation than existing controllers, the proposed architecture renders the dynamics of the shared virtual object variable with the network delay. Users feel the object becoming heavier as the network delay increases. Future research will extend the current work in two directions: (1) the stability analysis for the purpose of developing design guidelines for the various feedback gains; (2) the use of remote dynamic proxies in conjunction with passive communications for the purpose of displaying the dynamics of the shared virtual object more faithfully to the networked users.

## References

- [1] P. Buttolo, R. Oboe, and B. Hannaford. Architectures for Shared Haptic Virtual Environments. *Computer & Graphics*, pages 1–10, 1997.
- [2] P. E. R. Centre. WANem 1.1 Wide Area Network Emulator User Guide. 1:1 – 15, 2007.
- [3] J. Cheong, S.-I. Niculescu, A. Annaswamy, and M. Srinivasan. Motion synchronization in virtual environments with shared haptics and large time delays. In *Eurohaptics Conf, 2005 & Symp on Haptic Interfaces for Virtual Environment & Teleoperator Syst, 2005. World Haptics 2005. First Joint*, volume 1, pages 277–282, 2005.
- [4] J. Colgate and M. Stanley. Issues in the Haptic Display of Tool Use. In *IEEE/RSJ Int Conf Intell Robot Syst*, pages 140–145, Pittsburgh, PA, 1995.
- [5] D. Constantinescu, S. Salcudean, and E. Croft. Local Models of Interaction for Realistic manipulation of Rigid Virtual Worlds. *Int J Robot Res*, 24(10):789–804, 2005.
- [6] M. Fotoohi, S. Sirouspour, and D. Capson. Stability and Performance Analysis of Centralized and Distributed Multi-rate Control Architectures for Multi-user Haptic Interaction. *Int J Robot Res*, 26(9):977C994, 2007.
- [7] M. Glencross, C. Jay, J. Feasel, L. Kohli, M. Whitton, and R. Hubbard. Effective Cooperative Haptic Interaction over the Internet. In *Virt Real Conf, 2007. VR '07. IEEE*, volume 1, pages 115–122, Manchester Univ., 2007.
- [8] J. Kim, H. Kim, B. K. Tay, M. Muniyandi, M. A. Srinivasan, J. Jordan, J. Mortensen, M. Oliveira, and M. Slater. Transatlantic Touch: A Study of Haptic Collaboration over Long Distance. *Presence*, 13(3):328–337, 2004.
- [9] P. Mitra and G. Niemeyer. Dynamic Proxy Objects in Haptic Simulations. In *IEEE Conf Robot Autom Mechatronics*, pages 1054–1059, 2004.
- [10] G. Niemeyer and J.-J. Slotine. Using wave variables for system analysis and robot control. In *IEEE Int Conf Robot Autom*, volume 2, pages 1619–1625, 1997.
- [11] D. Ruspini, K. Koralov, and O. Khatib. Haptic Interaction in Virtual Environments. In *IEEE/RSJ Int Conf Intell Robot Syst*, Genoble, France, 1997.
- [12] J.-H. Ryu and C. Preusche. Stable Bilateral Control of Teleoperators Under Time-varying Communication Delay: Time Domain Passivity Approach. In *IEEE Int Conf Robot Autom*, volume 1, pages 3508 – 3513, 2007.
- [13] G. Sankaranarayanan and B. Hannaford. Experimental comparison of internet haptic collaboration with time-delay compensation techniques. In *Symp Haptic Interf Virt Envir Teleop Syst*, volume 1, pages 259 – 266, 2008.
- [14] G. Sankaranarayanan and B. Hannaford. Experimental comparison of internet haptic collaboration with time-delay compensation techniques. In *IEEE Int Conf Robot Autom*, volume 1, pages 206–211, 2008.
- [15] H. Sugarman, E. Dayan, A. Weisel-Eichler, and J. Tiran. The Jerusalem Telerehabilitation System, a New, Low-Cost, Haptic Rehabilitation Approach. *CyberPsychology & Behavior*, 9(2):178–182, 2006.
- [16] C. Zilles and J. Salisbury. A Constraint-based God Object Method for Haptic Display. In *ASME Haptic Interf Virt Envir Teleop Syst*, pages 146–150, Chicago, IL, 1994.