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

A Multi-rate Control Approach to Haptic Interaction in Multi-user Virtual Environments

M. Fotoohi, S. Sirouspour* and D. Capson

Department of Electrical and Computer Engineering, McMaster University
Hamilton, Ontario L8S 4K1, Canada
(*) Email: sirouspour@ece.mcmaster.ca

Abstract— **High-fidelity haptic interaction in multi-user environments over general Ethernet-based Local Area Networks (LAN) and Metropolitan Area Networks (MAN) can be challenging but has promising applications. Under typical network traffic conditions, the 1kHz real-time control rate suggested in the literature for stable haptic simulation is well above that achievable by conventional network protocols such as the UDP and TCP/IP. To overcome this limitation, a decentralized multi-rate control approach is proposed in which local force-feedback loops are executed at higher rates than data packet transmission between the user workstations. Mathematical models for stability and performance analysis of such multi-rate haptic control systems are presented. Analytical and experimental results demonstrate improved performance and stability for the distributed control architecture when compared with a centralized controller.**

Index Terms— **Haptic Interfce Control, Cooperative Haptics, Collaborative Haptics, Multi-user Haptics, Collaborative Virtual Environments, Multi-rate Control Systems.**

I. Introduction

Haptic interaction in shared virtual environments (VEs) is an emerging area of research with promising applications [1]. Examples include training of surgical tasks, haptic-assisted rehabilitation, teaching writing skills to children, sports training, as well as network-based multi-user games. In network-based multi-user haptics, users can interact across communication links such as Ethernet-based LANs, MANs or more broadly Wide Area Networks (WANs). Such configurations remove physical barriers and permit users to interact over long distances. The problem, however, is that network data communication is generally nondeterministic and suffers from delay, jitter, packet loss, and limited packet transmission rate. These can all adversely affect the performance and stability of cooperative haptics and pose a formidable control design challenge.

The focus of the present study is on cooperative haptic simulation over LANs and high-speed MANs. Such fast networks can connect users within a single building, over several buildings across a university campus, or even over a geographical area as large as a city. They can provide small and large companies, universities, hospitals, and similar large organizations with a fast and reliable means of communication. In Table I, the results of a network communication experiment between two workstations running the VxWorks real-time operating system with the UDP protocol over a mixed 100Mbps/1Gbps LAN are presented. These results suggest that for the scope of this work the data packet loss is negligible and the network communication delay between the user workstations is a fraction of the network sampling time. However, the packet transmission rate is well below the 1kHz required for stable simulation of stiff contacts.

Ave. round-trip delay	2.44 ms
RMS jitter	0.49 ms
Packet loss	2 packets out of 2.7 million packets
Achievable packet rate	128 Hz

TABLE I
CHARACTERISTICS OF A TYPICAL LAN UNDER NORMAL TRAFFIC DURING 36 MINUTES OF PACKET TRANSMISSION.

In this paper, the effect of a limited packet transmission rate and a relatively small communication delay on the stability and performance of network-based multi-user haptic interaction is investigated. Two multi-rate control architectures for haptic rendering in cooperative VEs are examined. The first is a centralized framework in which the user workstations send measurements of the haptic device positions to a server computer that performs all the calculations. In the second approach, a distributed architecture, each workstation runs a local copy of the VE and simultaneously communicates with other computers to coordinate the user actions. Mathematical models are developed to compare the stability and performance characteristics of these multi-rate control architectures.

While a large body of research in modeling and control of haptic interfaces has been dedicated to single-user applications, (e.g. see [2], [3], [4], [5]), the problem of cooperative haptics has received relatively little attention in the past. In [6], Yoshikawa and Ueda proposed a general structure for force-feedback to the users without addressing performance and stability degradation due to the above-noted network limitations. In [7], the authors investigated three different implementations for shared haptic environments depending on how the virtual environment is manipulated by the users. The effect of communication time delay on the performance of shared haptic virtual environments has been experimentally studied in [8], [9]. In [10] and [11], model-based controllers and wave variable-based techniques have been proposed for delay compensation in multi-user haptic rendering.

The rest of this paper is organized as follows. The cooperative haptic control architectures will be introduced in Section II. Mathematical modeling of multi-rate cooperative

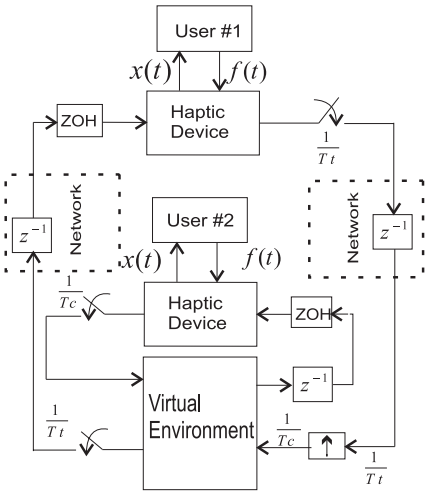

Fig. 1. Centralized control architecture.

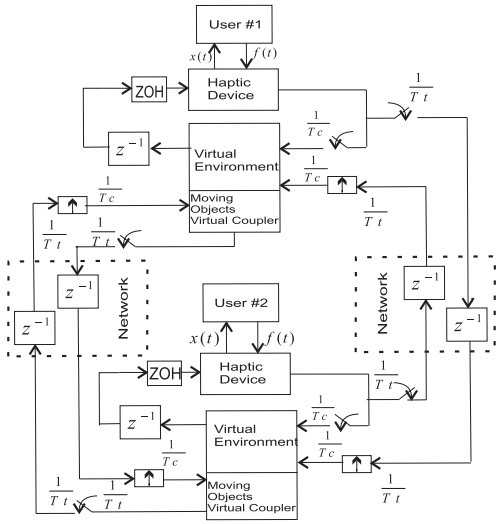

Fig. 2. Distributed control architecture.

haptic control will be discussed in Section III. Results of stability and performance analysis will be presented in Sections IV and V, respectively. Experimental results are given in Section VI, followed by conclusion remarks in Section VII.

II. Control Architectures for Cooperative Haptics

In multi-user virtual environments, the users are often connected through a network of computers that enable cooperative manipulation of shared objects. For such applications, two control architectures namely a centralized and a distributed are introduced and compared here. Although the given arguments are based on a dual-user setting, they can be easily generalized to multi-user configurations.

A. Centralized cooperative haptics

Most of prior relevant research in cooperative haptics is based on centralized control. In such architecture, as illustrated in Fig. 1 for a dual-user configuration, a central server that is host to the VE simulator collects and processes the information acquired by all user workstations and returns interaction forces along with objects and other user states to individual users. For simple applications, all components of the VE simulation can run at a high update rate. However, network delay and limited packet transmission rate can reduce the maximum achievable stiffness for the users across the network in this configuration.

B. Distributed cooperative haptics

A distributed control architecture such as the one shown in Fig. 2 can mitigate network-related stability and performance degradation of a centralized architecture at the expense of running a local copy of the VE simulation at all user workstations. To create the perception of a shared environment, each user receives the states, e.g. positions and velocities of all other users for force-feedback calculation. Discrepancy among corresponding shared dynamic objects is avoided by establishing spring-damper type virtual couplers between each pair. As in the case of the centralized architecture, any data communication over the network is subject to a low packet transmission rate and a small latency.

III. Modeling of Multi-rate Cooperative Haptics

The advantage of the distributed architecture over the centralized one is rather evident in the case of contact with static objects in shared environments. In the distributed framework, local high-rate delay-free feedback loops allow for rendering of rigid contacts whereas the network low transmission rate and delay restrict the maximum achievable stiffness in the centralized approach.

Models of single-axis/dual-user virtual-coupler-based haptic interaction under the centralized and distributed frameworks are shown in Figs. 3 and 4, respectively. In these figures, m_h^1 and m_h^2 are the combined masses of the pairs of user/haptic devices, respectively, $m_o = m_{o1} = m_{o2}$ is the mass of virtual object, k's and b's are the stiffness and damping of corresponding virtual couplers, x and $\bar{x}$ are local and network transmitted positions, and f_h^1 and f_h^2 are users' exogenous force inputs. The additional virtual couplers between the virtual objects in the distributed controller, represented by (k_{o1}, b_{o1}) and (k_{o2}, b_{o2}) in Fig. 4 are intended to prevent position drift between the two copies of the shared object. The delay in the communication channel is assumed to be one network sample time in each direction. Note that these multi-input/multi-output (MIMO) systems involve multi-rate discrete and continuous-time states due to the discrete-time nature of control synthesis, presence of the network element and zero-order-hold (ZOH) circuits, as well as the continuous-time dynamics of the haptic devices.

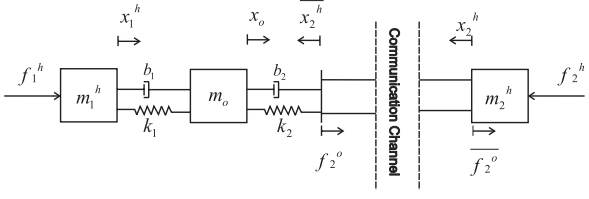

Fig. 3. Virtual-coupler-based centralized cooperative haptics.

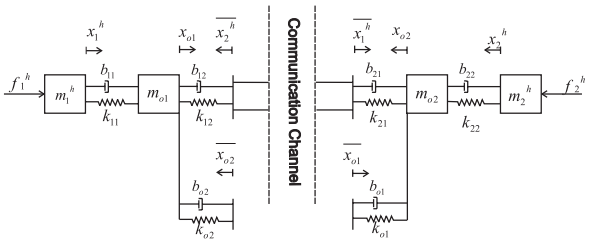

Fig. 4. Virtual-coupler-based distributed cooperative haptics.

In this paper, the state-space modeling approach for multi-rate discrete-time systems proposed in [12] is utilized for the analysis of the cooperative haptic control architectures. In the interest of space, only a general formulation of the problem is discussed and the details are omitted. For each architecture in Figs. 1 and 2 using the models in Figs. 3 and 4, the open-loop continuous-time model of the system, including the dynamics of the users, haptic interfaces and the virtual object can be written as (see Fig. 5)

$$\dot{x}(t) = Ax(t) + Bw(t), \quad y(t) = Cx(t) \tag{1}$$

where $x(t) \in R^{n_0}$ is the vector of system states, i.e. positions and velocities, $w(t) = (u_t(t), u_c(t))^T$ is the vector of inputs and $y(t) = (y_t(t), y_c(t))^T$ is the vector of measurements, and subscripts t and c refer to the network and control sampling rates. It is assumed that the network and control sample times are multiples of a base sample time t_{lcm}, i.e. $T_i = N_i t_{lcm}$, and the least common multiple of N_is is denoted by N_{lcm}. For the discrete-time realization of the system, an expanded state vector is defined as

$$x_D[k] = \begin{pmatrix} x((k-1)T_{lcm} + t_{lcm}) \\ \vdots \\ \vdots \\ x((k-1)T_{lcm} + (N_{lcm}-1)t_{lcm}) \\ x(kT_{lcm}) \end{pmatrix} \tag{2}$$

with $T_{lcm} = N_{lcm} t_{lcm}$. The expanded discrete output vector $y_D[k] = \left(y_{D_t}[k],\ y_{D_c}[k]\right)^T$ where

$$y_{D_i}[k] = \begin{pmatrix} y_i(kT_{lcm}) \\ y_i(kT_{lcm} + T_i) \\ \vdots \\ \vdots \\ y_i(kT_{lcm} + (N_i-1)T_i) \end{pmatrix} \quad i = t, c \tag{3}$$

The expanded discrete input vector u_D can be defined similarly. Using the above definitions, it can be shown that [12]

$$\begin{aligned} x_D[k+1] &= A_D x_D[k] + B_D u_D[k] \\ y_D[k] &= C_D \left\{U_1 x_D[k+1] + U_2 x_D[k]\right\} \end{aligned} \tag{4}$$

where the expressions for A_D, B_D, and C_D are derived in [12] and U_1, U_2 are block diagonal matrices given by

$$\begin{aligned} U_1 &= diag(I_{n_0}, I_{n_0}, \cdots, I_{n_0}, 0) \\ U_2 &= diag(0, 0, \cdots, 0, I_{n_0}) \end{aligned} \tag{5}$$

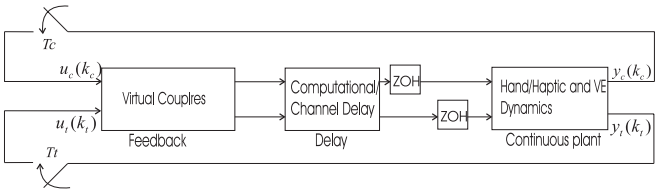

Fig. 5. Block diagram representation of the multi-rate feedback control system.

By replacing $x_D[k+1]$ in second line of (4) from first line of (4), one may obtain a standard state-space representation as follows

$$\begin{aligned} x_D[k+1] &= A_D x_D[k] + B_D u_D[k] \\ y_D[k] &= \hat{C}_D x_D[k] + \hat{D}_D u_D[k] \end{aligned} \tag{6}$$

where $\hat{C}_D = C_D U_1 A_D + C_D U_2$ and $\hat{D}_D = C_D U_1 B_D$. The one-sample delay elements associated with computation and data transmission can be incorporated into the discrete-time model by augmenting the state vector with the delayed input signals. The derivation of delayed state transition matrices, $\tilde{A}$, $\tilde{B}$, $\tilde{C}$, and $\tilde{D}$, is straightforward and will not be presented here.

Once the open-loop discrete-time difference equations are obtained, the closed-loop dynamics can be formed using the feedback law $u_D = F_D * y_D$, where F_D is the feedback gain matrix whose elements are constant and consist of the stiffness and damping parameters of all virtual couplers present in the system. In the interest of space, the actual form of F_D will not be presented here. Finally, the closed-loop space transition matrix A_D^c can be computed as

$$A_D^c = \widetilde{A}_D + \widetilde{B}_D F_D (I - \widetilde{D}_D F_D)^{-1} \widetilde{C}_D \tag{7}$$

The closed-loop system is stable if and only if all eigenvalues of this matrix lie inside the unit circle.

IV. Stability Analysis

In this section, the stability margins of the centralized and distributed control architectures with respect to changes in the stiffness parameters of the virtual couplers are compared. The values of constant parameters in all scenarios are $m_1^h = m_2^h = 0.1$kg, $m_o = m_{o1} = m_{o2} = 0.4$kg, $T_t = 1/128$s, $T_c = 1/1024$s and b=10Ns/m for all couplers in both configurations.

Fig. 6(a) illustrates the stability region of the centralized architecture, when all system parameter except k_1 and k_2 in Fig. 3 are fixed. The marginal values of these parameters determine the maximum stiffness that can be presented to each user. Clearly, the network low transmission rate and delay have contributed to a significant reduction in the margin of stability with respect to k_2 for the user across the network.

The stability analysis was carried out for the distributed control architecture as well. In Fig. 6(b), the stable region with respect to the parameters $k_{11} = k_{22}$ and $k_{12} = k_{21}$ in Fig. 4 is plotted. For this case, $k_{o1} = k_{o2} = 1000$N/m are constant. Note that the stability region is noticeably enlarged when compared to that of the centralized architecture in

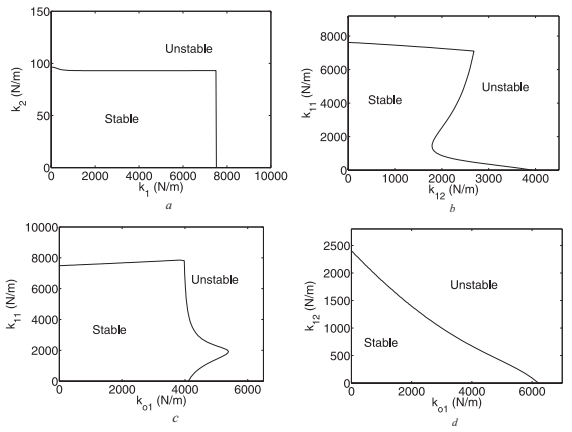

Fig. 6. The region of stability for: (a) the centralized control architecture; (b) the distributed control architecture when $k_{o1} = k_{o2} = 1000$N/m are fixed; (c) the distributed control architecture when $k_{12} = k_{21} = 300$N/m are fixed; (d) the distributed control architecture when $k_{11} = k_{22} = 2000$N/m are fixed.

Fig. 6(a). To study the effect of the coordinating virtual couplers between the objects, $k_{12} = k_{21} = 300$N/m were held constant while $k_{11} = k_{22}$ and $k_{o1} = k_{o2}$ where changed. The results are given in Fig. 6(c). Finally, Fig. 6(d) demonstrates the stable region for the case in which $k_{11} = k_{22} = 2000$N/m are constant while $k_{o1} = k_{o2}$ and $k_{12} = k_{21}$ are changed.

V. Performance Analysis

In this section using the models in Fig. 7, the performance of the centralized and distributed architectures for manipulating a virtual object in free motion, and in contact with a rigid wall is analyzed. In rigid contact, the stiffness of the wall k_w is relatively large whereas in free motion k_w and b_w are set to zero. In the succeeding analysis for the centralized controller, the users on the server and remote workstations are denoted as local and remote user, respectively. The decentralized controller is symmetric with respect to the two users and therefore, only one user is considered in its analysis. To compare the controllers, the perceived impedances by the users in each architecture are compared with that of a mass discretized with a sample rate of T_t.

First, it is assumed that one user is manipulating the virtual object in free motion while the second user input force is set to zero. The perceived impedance of the object is defined as the ratio of input force f_i^h to the output velocity v_i^h in Fig. 7. The following parameters are used in this case: $T_c = 1/1024$s, $T_t = 1/128$s, $m_1^h = m_2^h = 0.1$kg, $m_o = .4$kg, $k_1 = 1000$N/m, $k_2 = 85$N/m, $b_1 = 15$N.s/m, $b_2 = 3$N.s/m, $b_{21} = 2$N.s/m, $k_{21} = 400$N/m, $k_{o2} = k_{o1} = 300$N/m, $b_{o2} = b_{o1} = 10$N.s/m, $k_{11} = k_{22} = 3000$N/m, $b_{22} = b_{11} = 20$N.s/m.

The frequency responses are shown in Fig. 8(a)-(b) from which it is clear that the local user in the centralized architecture observes the closest impedance to that of the virtual mass, compared with the remote and distributed users. This should not be surprising since the network element is absent from the local user feedback control loop. It should be

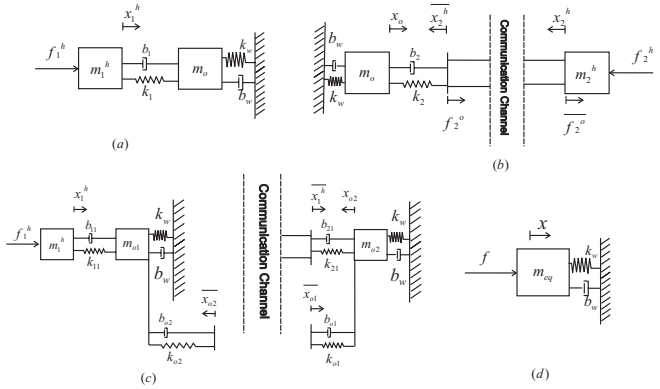

Fig. 7. Model of haptic interaction with $k_w = 30$kN/m and $b_w = 20$Ns/m in hard contact and $b_w = k_w = 0$ in free motion: (a) local user in centralized controller; (b) remote user in centralized controller; (c) distributed controller; (d) the ideal system.

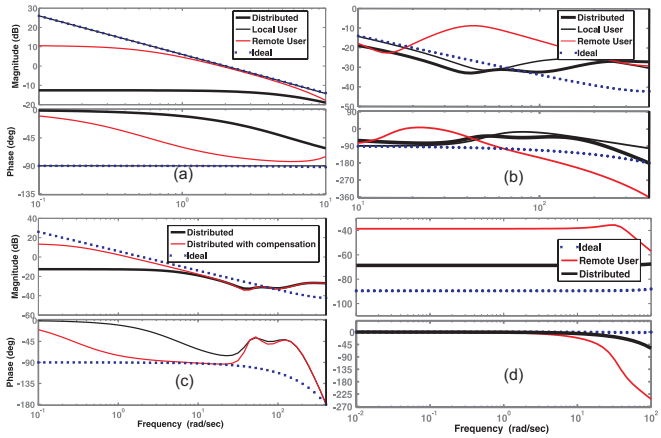

Fig. 8. User perceived admittances: (a) free-motion at low frequency; (b) free-motion at medium-to-high frequencies; (c) distributed controller after compensation; (d) contact with rigid wall.

noted, however, that at high frequencies the effect of spring-damper coupler becomes dominant and the response deviates from that of a mass. At medium-to-high frequencies, the remote user in the centralized controller observes the farthest deviation from the ideal response. At low frequencies the virtual object presents a dominant viscous behavior for the distributed controller and the local user in the centralized controller. The amount of this damping can be analytically calculated as a function of system parameters using the discrete-time multi-rate models introduced in Section III. The result for the case of $N = 2$ in the centralized controller is given below:

$$b_{cent} = \frac{k_2}{2/T_c - b_2/m_o} \tag{8}$$

The expressions of the damping term for the distributed controller for $N = 2$, as well as those in the case of $N > 2$ are rather long and will not be presented here. For the centralized controller, it is evident from (8) that amount of viscous damping increases as the control rate decreases. A similar behavior is observed in the case of the distributed controller. To compensate the undesirable effect of this viscous friction on the user's perception of the object,

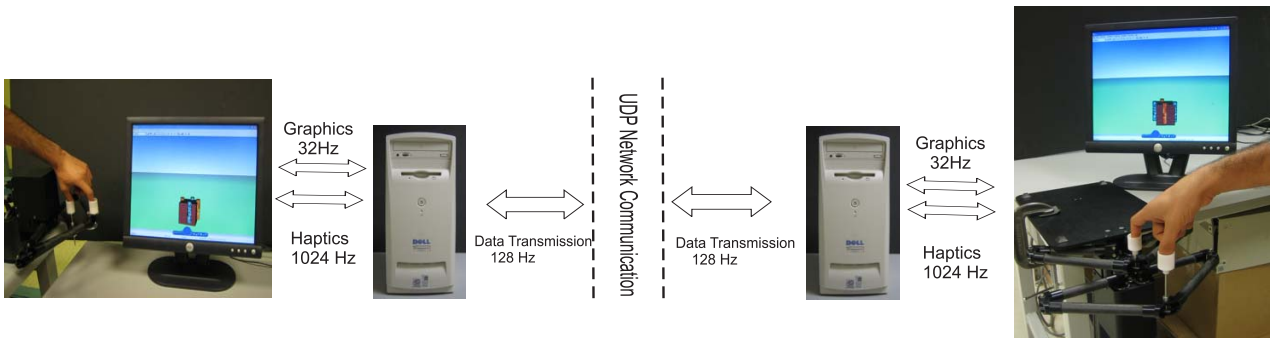

Fig. 9. The experimental setup.

it is proposed that a negative damping be added to the object dynamics based on the analytical results. For instance, Fig. 8(c) shows an improvement in the low-frequency response after introducing the compensator $b_o = -3$N.s./m in the distributed architecture. The damping values can be computed off-line for every dynamic single-body object in the virtual environment using the results of this paper.

As shown in Section IV, the distributed controller has significantly higher marginal values for the stiffness couplers compared to those of the centralized controller. In rigid contacts the stiffness observed by the user can be approximated by that of the virtual coupler. Fig. 8(d) displays the the ratio of the hand position x_i^h to user's input force f_i^h when the user is pushing the object against a rigid wall. The controller parameters are the same as those in free motion and the wall stiffness and damping are $k_w = 30$kN/m, and $b_w = 20$N.s/m, respectively. At low frequencies the magnitude of response is close to $1/k_2$, $1/k_{11}$ and $1/k_w$ for the remote, distributed and ideal system, respectively.

VI. Experimental Results

The experimental setup shown in Fig. 9 consists of two modified dual-pantograph mechanisms from Quanser Inc. that enable dual-user/dual-finger haptic interaction. The users can grasp and cooperatively manipulate a virtual box in a plane by moving it in x and y directions and rotating it around the z axis, both in free motion and in contact with rigid walls. The fixed-step Euler integration routine is used to simulate the dynamics of the virtual box at an update rate of 1 kHz. The user workstations are linked through a LAN and communicate using the UDP protocol at a packet transmission rate of 128 Hz. The Matlab Virtual Reality toolbox has been employed for the graphics at an update rate of 32 Hz. The control code has been implemented using the Matlab Real-time Workshop toolbox and runs under Tornado/VxWorks real-time operating system.

The control parameters for the centralized controller were set to: $k_{N1} = k_{N2} = 800$N/m,$b_{N1} = b_{N2} = 3$Ns/m, $k_{T1} = k_{T2} = 1200$N/m, $b_{T1} = b_{T2} = 10$Ns/m, $k_{o1x} = k_{o1y} = k_{o2x} = k_{o2y} = 400$N/m, $b_{o1x} = b_{o1y} = b_{o2x} = b_{o2y} = 8$Ns/m, $k_{o1R} = k_{o2R} = 10$Nm/rad, $b_{o1R} = b_{o2R} = 0.1$N.s/rad where N and T denote the normal and tangential directions at the finger/box contact points, respectively. The indices x, y and R denote the x, y and rotation coordinates of the box. In the distributed architecture, the control parameters were chosen as $k_{N11} = k_{N22} = k_{N12} = k_{N21} = 800$N/m and $k_{T11} = k_{T22} = k_{T12} = k_{T21} = 1200$N/m with the damping values being the same as those in the centralized architecture. Furthermore, for the moving object virtual couplers, $k_{o1x} = k_{o1y} = k_{o2x} = k_{o2y} = 400$N/m, $k_{o1R} = k_{o2R} = 10$N.m/rad, $b_{o1x} = b_{o1y} = b_{o2x} = b_{o2y} = 8$Ns/m, $b_{o1R} = b_{o2R} = 0.1$ N.m.s/rad.

Fig. 10 displays the profiles of the box-finger normal interaction force for one of the fingers as well as the virtual box position. While the users can cooperatively grasp and move the virtual box under the distributed controller, such operation is almost impossible with the centralized controller due to its instability. This is evident from the force and position profiles in Fig. 10 where the user has to exert significant damping in order to barely grasp the object and move it in the plane, as well as from the ripples in the box x-y trajectory. In contrast, the distributed architecture can provide a smooth and stable rigid interaction with the box under similar circumstances.

Experiments were conducted in order to investigate the effect of network low packet rate and delay on the users' perceived impedances of the virtual object. To apply consistent forces in different experiments, the user forces are emulated through the control signals. One of the pantograph mechanisms is moved along a sinusoidal path with an amplitude of 0.05m and a frequency of 2 rad/sec in the y direction using a proportional-derivative controller, while a constant force was applied to the second pantograph along the the same direction. With this arrangement, the box was grasped and moved along the y direction by the two pantographs. The control parameters are the same as those in the previous case. Ideally, the sum of the forces applied on the virtual object must be equal to the inertial force required for moving the box along the sinusoidal path.

In Fig. 11(a), the local user force profile in the centralized architecture is compared with that of the user in the distributed architecture. When uncompensated, the user in the distributed system has to apply a noticeably larger force in order to generate the same motion. This is consistent with the analytical result that had predicted an extra viscous damping in the system response due to network low packet rate and delay. As can be seen in Fig. 11(a), introducing a negative damping in the object dynamics can significantly improve the

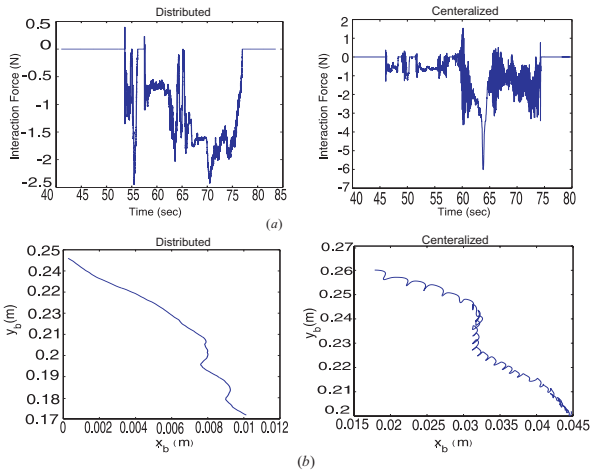

Fig. 10. Comparison of distributed and centralized architectures in experiment: (a) finger-box interaction force; (b) box trajectory in the x-y plane.

response. The values of the damping were 0.88Ns/m for the linear axes of motion, and 0.025N.s.m/rad for the rotational motion, all chosen based on the results of analysis. To calculate the damping value for the rotational compensator, a similar model to that in Fig 4 is used with the exception that m_o is replaced with the rotational inertia of the box and the coupler forces are multiplied by an average torque arm length, to calculate the applied torque. In the experiments, the users observed a noticeable improvement in the system response after the inclusion of the damping compensators.

To compare the control architectures in rigid contact, the virtual box is pushed against a stiff wall with parameters $k_{Nw} = 4000$N/m, $k_{Tw} = 7000$ N/m, $b_{Nw} = 30$Ns/m, $b_{Tw} = 50$Ns/m, using an emulated user force of $f_y = 1.5\sin(3t)$N). Instability in the centralized controller is avoided by setting the remote user parameters to $k_N = 170N/m$, $b_N = 3$Ns/m, $k_T = 270$N/m, $b_T = 5$Ns/m. The resulting haptic device displacements are plotted in Fig 11(b). The user in the distributed controller and the local user in the centralized controller perceive a stiffer contact, as is evident from their smaller penetration in the virtual wall.

VII. Conclusions

Networked collaborative haptic environments present new challenges to the designers of haptic-enabled virtual reality systems. These are mainly due to network constraints such as limited packet transmission rate, latency, data loss and jitter. Motivated by LAN and MAN-based multi-user haptic applications, the effect of data transmission rate and delay on the stability of such systems was examined. Using mathematical descriptions for multi-rate MIMO discrete-time systems, the stability and performance characteristics of a centralized and a distributed control architecture were compared. Analytical results as well as experiments conducted with a dual-user/dual-finger haptic platform demonstrated that the distributed controller possesses noticeably lager stability margins and improved fidelity.

In future, the effect of larger network delays, jitter, and packet loss which are the main characteristics of the Internet

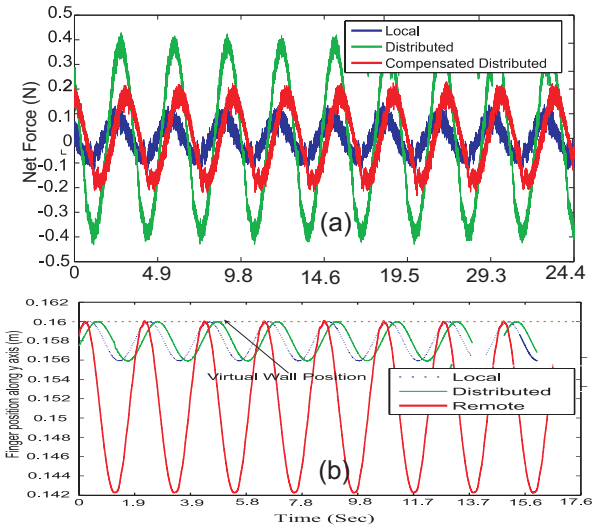

Fig. 11. (a) The net force exerted on the object for free motion along a sinusoidal path; (b) finger position when the user pushes the object against a stiff wall.

will be studied and solutions for stable cooperative haptic interaction under such conditions will be sought.

Acknowledgments

The authors would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) and Ontario Centres of Excellence (OCE) for supporting this research.

References

[1] R. Waters and J. W. Barrus, "The rise of shared virtual environments," in *Int. Symposium on Hapanic Interfaces for Virtual Environment and Teleoperator Systems*, pp. 20–25, March 1997.

[2] J. Colgate and G. Schenkel, "Passivity of a class of sampled-data systems: Application to haptic interfaces," *J. Robot. Syst.*, vol. 14, no. 1, pp. 37–47, 1997.

[3] M. Sirouspour, S. Dimaio, S. Salcudean, P. Abolmaesumi, and C. Jones, "Haptic interface control-design issues and experiments with a planar device," in *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 789–794, April 2000.

[4] R. Adams and B. Hannaford, "Control law design for haptic interfaces to virtual reality," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 1, pp. 3–13, 2002.

[5] V. Hayward and et al., "Haptic devices and interfaces," *Sensor Review*, vol. 24, no. 1, pp. 16–29, 2004.

[6] T. Yoshikawa and H. Ueda, "Construction of virtual world using dynamics modules and interaction modules," in *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 2358–2364, April 1996.

[7] P. Buttolo, R. Oboe, and B. Hannaford, "Architectures for shared haptic virtual environments," *Computers and Graphics*, vol. 21, pp. 421–429, 1997.

[8] K. Hikichi, H. Morino, I. A. K. Sezaki, and Y. Yasuda, "The evaluation of delay jitter for haptics collaboration over the Internet," in *IEEE Global Telecommunications Conference*, pp. 17–21, Nov 2002.

[9] M. Alhalabi, S. Horiguchi, and S. Kunifuji, " An experimental study on the effects of network delay in cooperative shared haptic virtual environment," *Computers and Graphics*, vol. 27, pp. 205–213(9), 2003.

[10] H. Arioui, A. Kheddar, and S. Mammar, "Architectures for shared haptic virtual environments," *Journal of Intelligent and Robotic Systems*, vol. 37, pp. 193–207, 2003.

[11] C. Carignan and P. Olsson, "Cooperative control of virtual objects over the Internet using force-reflecting master arms," in *Proc. IEEE Int. Conf. Robot. Automat.*, pp. 1221–1226, April 2004.

[12] M. Araki and K. Yamamoto, "Multivariable multirate sampled-data systems: State-space description, transfer characteristics, and Nyquist criterion," *IEEE Transactions on Automatic Control*, vol. 31, pp. 145–154, 1986.