Tangible Cooperation in Shared Virtual Environment

Sylvia Irawati [1]　Jong-Phil Kim [2]　Jinwook Kim [3]　Heedong Ko [4]

Abstract

Recent advanced technologies enable multiple users to share the virtual environment and work together as they are collocated. Additional sensory information such as haptic could improve the cooperation. In this paper, we propose a server-client architecture with multi-rate haptic control to support a tangible cooperation. Using our approach, the system is able to maintain a consistent simulation state across multiple users as well as to provide a high-fidelity stable haptic interaction. To verify our approach, we have developed an experimental application and tested the cooperation among multiple users. The results confirm that our system is able to provide coherency among clients as well as haptic transparency.

Keywords: haptic, tangible, multi-user virtual environment, virtual reality, collaboration, cooperation

This work was supported by Korea Institute of Science and Technology (KIST) through the Tangible Web Project.

[1] main-author : Ph.D student at University of Science and Technology, Research Student at Korea Institute of Science and Technology;
e-mail: sylvi@imrc.kist.re.kr

[2] co-author : Imaging Media Research Center, Korea Institute of Science and Technology;　e-mail: lowtar74@imrc.kist.re.kr
[3] co-author : Imaging Media Research Center, Korea Institute of Science and Technology; e-mail: jwkim@imrc.kist.re.kr
[4] corresponding author : Imaging Media Research Center, Korea Institute of Science and Technology; e-mail: ko@imrc.kist.re.kr

1. Introducttion

Nowadays, people are living in the advanced interface technologies that enable the users to interact with a computer-simulated environment, such as in Virtual Reality. Coupling with the network technologies, multiple users can share the virtual world and perform a cooperative task together. There are a number of popular multi-user virtual environment projects, such as SecondLife[13], OpenSimulator[12], Croquet[10], etc. which enable users to share the experiences in the virtual world. However, most of them support only desktop interaction metaphors, such as mouse and keyboard inputs. There are only a few examples of multi-user online applications that successfully use a haptic interface for collaboration.

In the collaborative application, a transparent haptic interaction could increase task performances as well as sense of co-present between remote participants. However, providing a high-fidelity haptic interaction in multi-user environments over general Local Area Network (LAN) could be a challenging task. Unlike the visual and auditory displays which have a directional information flow, haptic has bidirectional energy flow. Thus, the stability and the responsiveness are very important factors for haptic rendering. Meanwhile, the network data communication usually suffers from delay, jitter, packet loss, and limited packet transmission rate that could affect the consistency of simulation states across multiple participants and also the stability and fidelity of haptic interaction.

Here, we propose a server-client architecture with multi-rate haptic rendering control to support a scalable tangible cooperation. The system decouples the simulation, visual rendering, and haptic rendering into asynchronous components that run at different sites with different rates. Using this approach, the system is able to maintain consistent states across multiple users as well as to provide a high-fidelity stable haptic interaction.

The rest of the paper is organized as follows: in the next section, we review some related works; we then present the overview of our proposed system followed by the implementation of our system and our testbed application and result analysis; and finally we draw some conclusion and provide an outlook on our future work.

2. Related Work

There have been various attempts to overcome the adverse effects of the network problems for shared haptic environments which could be classified into two main approaches: peer-to-peer architecture and server-client architecture.

In the peer-to-peer architecture, peers directly apply a user' s input to their own simulation. Thus, it can avoid additional input latency which could enable a stable haptic interaction. Fotoohi et al. [1] proposed a multi-rate control for distributed architecture which highly increases stability region of haptic interaction. Glencross et al. [2] proposed an optimistic simulation and roll-back technique for maintaining synchrony between simulations. Sankaranarayanan and Hannaford [8] proposed a virtual coupling scheme for maintaining position coherency between multiple copies of a virtual object in each peer. However, in the peer-to-peer architecture with a large number of objects and peers, it is hard to ensure consistent simulation states at each peer in the absence of a ground truth to correct the inconsistent states.

One of the most straightforward approaches to increase the consistency is a server-client architecture, where the server holds the ground truth and sends the state to all clients. However, the clients may suffer from a round-trip delay which can highly reduce the haptic fidelity and make haptic interaction unstable. Kim et al. [4] proposed a way to maintain the haptic stability by adding large amounts of damping but it highly reduces the transparent haptic sense. Lee and Kim [6] proposed a transport scheme for haptic collaborations by reducing the transmission rate of haptic events using a network-adaptive aggregated packetization and the priority-based filtering. Here, we propose a server-client architecture with multi-rate haptic rendering control to support a provide a high-fidelity stable haptic cooperation in the shared virtual environment.

3. System Overview

Multi-user virtual environment enables multiple users who are connected through computer network to cooperatively interact with the shared virtual environment. Using the tangible interfaces, multiple users can simultaneously collaborate to manipulate the shared object.

Our system is based on a server-client architecture where the server manages the consistent simulation states among multiple users and the client provides an interface for the user to interact with the shared virtual environment. It decouples the simulation, visual rendering, and haptic rendering into asynchronous components that run at different sites with different rates. The main different part from the conventional server-client architecture is the haptic rendering component on the client side that applies a multi-rate control approach in order to provide a high-fidelity stable haptic interaction.

As shown in figure 1, the system is composed of two main parts: Server and Client. The server, in which the simulation engine is located, is responsible for simulating the virtual environment' s behavior and synchronizing the object states across multiple clients by distributing the updates to all clients. The server uses a database to store the simulation data persistently.

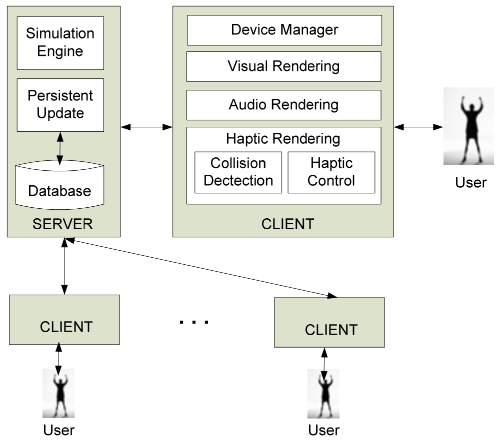

Figure 1 System Overview

Multiple clients connect to the server to share the interactive environment. Each client might have different composition of components depending on the needs. The client is mainly responsible for receiving the user input and rendering the multi sensory outputs (visual, auditory, and haptic). Each rendering component requires a different update rate. Visual rendering typically runs at 60Hz to provide a stable visual display. On the other hand, a haptic rendering system requires a higher update rate (>1kHz) in order to provide a high-fidelity stable haptic feedback.

The simulation engine on the server runs the physics simulation. It does the collision detection for all objects in the virtual environment. Meanwhile, the haptic rendering component on the client is responsible for collision detection between haptic interaction point (HIP) and the virtual objects. It computes the interaction forces between the HIP and the virtual objects populating the environment. Using our approach, the system can save the computation time to do the collision detection and to calculate the force feedback. Moreover, processing the force feedback calculation in the client side can reduce the adverse affect caused by the network problems.

In addition, a haptic control algorithm is applied to our haptic rendering system to improve the stability and fidelity of haptic interaction. Haptic control is responsible for sending the response force to the haptic device by applying some control algorithm to minimize the error between ideal and applicable forces maintaining passivity. Our system uses an Energy Bounding Algorithm (EBA) as proposed by Kim et al. [5] for the haptic control component to enable a high frequency update rate.

4. Implementation

The implementation of our system is based on VARU framework [3], as shown in Figure 2. Our visual rendering is mainly based on the OpenSceneGraph [11]. The simulation engine used in the server and client sides is based on Virtual Physics library [14]. A persistent database (MySQL database), is used in the server, to store the simulation data persistently. For managing the peripheral devices, we use VRPN library [7] which provides an interface between the applications and the interaction devices, such as joystick, force feedback devices, etc.

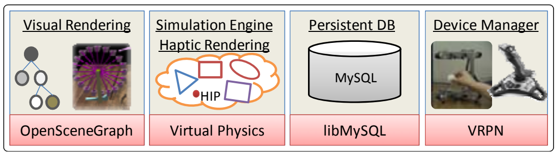

Figure 2 System Implementation

Using our system, the application developer could easily configure virtual reality application to work with specific scenario. All components of the system are configurable through XML description, so it can easily adapt to different application scenario. Moreover, the system is designed to be extensible in which a new component could be integrated to the system by implementing a new module that can work together with the existing components.

5. Experiments

We have developed a tangible cooperative application that involves multiple users who cooperate in the Placing task. Figure 3 shows the snapshot of the application. At the beginning, the blocks are located outside the container. The users have to locate the blocks into the container (in the middle, following the color clues) by collaborating with the other users.

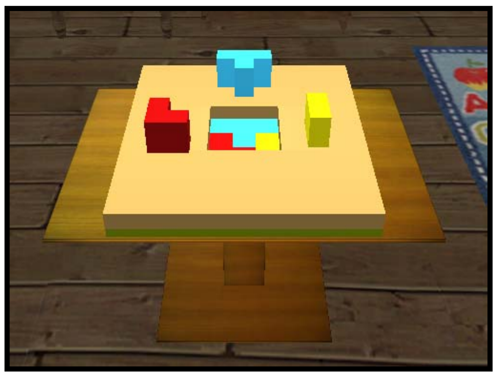

Figure 3 Experimental application

5.1 Experimental Setup

In this experiment, two users work together to complete the task. One user is located in the CAVE-like environment and interacting with the system using a human-scale haptic device, SPIDAR [9] (client_1). The other user is located at the desktop environment and interact using a desktop SPIDAR device (client_2).

Using SPIDAR the users are able to push and drag the objects but not to pull them. Therefore, cooperation between the users is required to locate the blocks into the container more easily.

In order to evaluate the quality of haptic interaction, we perform the virtual wall simulation to measure the stability and fidelity. We test two different experimental conditions: (1) general server-client architecture without multi-rate control and (2) our proposed approach with simulated network jitter and latency. For each experimental trial, we measure the X directional position and force of the end-effector of the haptic device. To verify our system in terms of the coherency among multiple clients, we also recorded the object position in each client and compare the results.

5.2 Result and Discussion

Figure 4-5 shows the results of haptic stability for condition 1 and 2. Without a multi-rate control the haptic fidelity is highly reduced due to a large computational delay on the server side. When stiffness 100N/m is applied, the haptic feedback is stable (figure 4a). However, when the stiffness value is increased to 200N/m, the position of the end-effector (blue line) relative to the virtual wall (red dot line) oscillates which means the haptic feedback is unstable. Thus, the user can not recognize the stiff contact with the virtual object.

Using our proposed approach, a server-client with a multi-rate control, the stability and fidelity of haptic feedback can be increased. Figure 5a shows the position of the end effector relative to the virtual wall position. Compared to figure 4a, we can see that the penetration depth of the end-effector to the virtual wall is smaller and the displayed stiffness is much higher (figure 5b). It means that the system can provide a correct perception of solid contact.

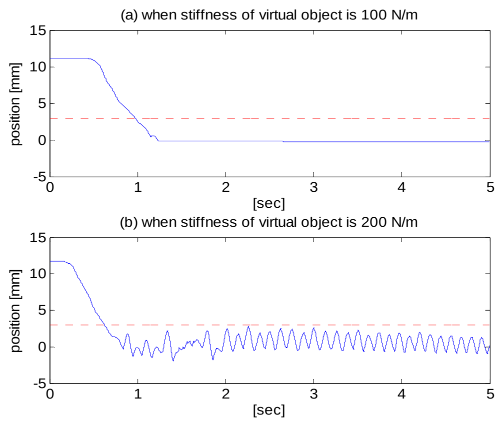

Figure 4 Haptic stability in the shared VE without multi-rate control. Network condition: latency=0ms

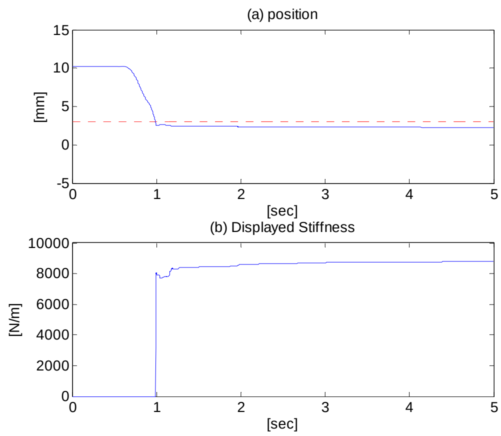

Figure 5 Haptic stability and fidelity in the shared VE with multi-rate control. Network condition: average jitter=150ms

In order to confirm the coherency of multiple clients, we record the object position in the server and client sides. Figure 6 shows the trajectory of object A (position X) during the simulation time (6-10 seconds) which is affected by the user inputs. The result shows that the clients receive the update from the server after several milliseconds depending on the network jitter which cause the clients have inconsistent simulation state temporarily. However, these inconsistencies will be updated soon after the updates from server arrive.

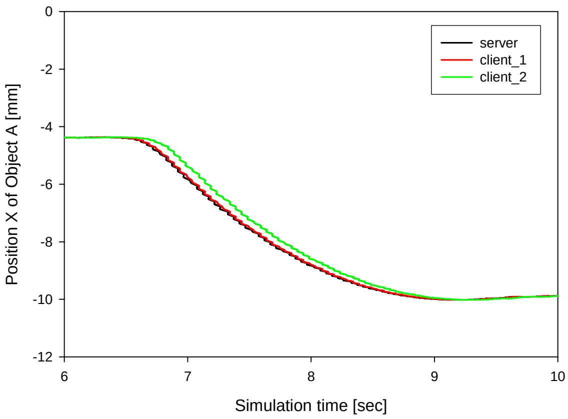

Figure 6 Trajectory of object A (position x) in the server and multiple client sides. Network condition: average jitter=150ms

6. Conclusion

Multi user virtual environment enables multiple users to share the virtual experience and do the interaction tasks cooperatively as they are collocated. We propose a server-client architecture integrated with a multi rate control to provide consistent simulation states among clients as well as to provide transparent haptic interactions. The preliminary results show that our system is able to provide coherency among clients as well as haptic transparency. Further experiments are required to evaluate the quality of the shared haptic environment and to explore more cooperation tasks among multiple clients with different interfaces to see the effect of the haptic in the collaboration or cooperation.

References

[1] M. Fotoohi, S. Sirouspour, and D. Capson, "Stability and performance analysis of centralized and distributed multi-rate control architectures for multi-user haptic interaction", The International Journal of Robotics Research, vol. 26, no. 9, pp. 977 994, 2007.

[2] M. Glencross, C. Jay, J. Feasel, L, Kohli, M. Whitton, R. Hubbold, "Effective Cooperative Haptic Interaction over the Internet", Proc. of IEEE Virtual Reality, pages 115-122, March 2007.

[3] S. Irawati, S. Ahn, J. Kim, and H. Ko, "VARU framework: Enabling rapid prototyping of VR, AR and ubiquitous applications", Proc. of IEEE Virtual Reality Conference, pp 201-208, 2008.

[4] K. 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: Teleoperators and Virtual Environments. Volume 13. Number 3 pp 328-337. 2004

[5] J. P. Kim, C. Seo, J. Ryu, "A Multirate Energy Bounding Algorithm for High Fidelity Stable Haptic Interaction Control", Proc. of International Joint Conference SICE-ICASE, pp. 215-220, 2006

[6] S. Lee and J. Kim. "Haptic Event Prioritization and Network Adaptation Scheme for Collaborative Virtual Environments", Proc. of IEEE Global Telecommunication Conference, pp 2151-2155, 2007.

[7] M. R. Taylor II, T. C. Hudson, A. Seeger, H. Weber, J. Juliano, and A. T. Helser, "VRPN: A Device-Independent, Network-Transparent VR Peripheral System", Proc. of ACM International Symposium on Virtual Reality Software and Technology, 2001

[8] G. Sankaranarayanan, and B. Hannaford, "Virtual Coupling Schemes for Position Coherency in Networked Haptic Environments", Proc. BioRob, pp. 853- 858, 2006.

[9] M. Sato, "Development of String-based Force Display: SPIDAR", Proc. of International Conference on Virtual System and Multimedia, 2002.

[10] Croquet, http://www.opencroquet.org

[11] Openscenegraph. http://openscenegraph.org.

[12] OpenSimulator, http://opensimulator.org

[13] SecondLife, http://secondlife.com/

[14] Virtual Physics Library.
http://virtualphysics.imrc.kist.re.kr.