

Research Article

Providing QoS for Networked Peers in Distributed Haptic Virtual Environments

Alan Marshall, Kian Meng Yap, and Wai Yu

School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast BT7 1NN, Northern Ireland, UK

Correspondence should be addressed to Kian Meng Yap, m.yap@qub.ac.uk

Received 29 February 2008; Accepted 17 June 2008

Recommended by Mohammed Ghanbari

Haptic information originates from a different human sense (touch), therefore the quality of service (QoS) required to support haptic traffic is significantly different from that used to support conventional real-time traffic such as voice or video. Each type of network impairment has different (and severe) impacts on the user's haptic experience. There has been no specific provision of QoS parameters for haptic interaction. Previous research into distributed haptic virtual environments (DHVEs) have concentrated on synchronization of positions (haptic device or virtual objects), and are based on client-server architectures. We present a new peer-to-peer DHVE architecture that further extends this to enable force interactions between two users whereby force data are sent to the remote peer in addition to positional information. The work presented involves both simulation and practical experimentation where multimodal data is transmitted over a QoS-enabled IP network. Both forms of experiment produce consistent results which show that the use of specific QoS classes for haptic traffic will reduce network delay and jitter, leading to improvements in users' haptic experiences with these types of applications.

Copyright © 2008 Alan Marshall et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

There has been recent interest in the transmission of multimodal information over the Internet [1], and in particular the transmission of haptic information [2, 3] (haptic is sense of touch and force feedback and comes from the Greek word "haptikos" to *grasp, touch*. Telehaptics concerns remote haptic operations over network connections). The introduction of the haptic sense of touch (i.e., reflected force) refers to the perceptual kinaesthesia sensing of events such as heat, pressure, force, or vibration. This paper involves research into how new types of distributed applications which involve haptic devices, in addition to visual and aural information, can be carried over the Internet. Specifically, it considers an emerging class of applications that enable users to interact haptically with virtual environments.

By definition, a virtual environment (VE) is a space that provides users with the illusion of acting in a real world. However in addition to audio and visual information, the provision of haptic feedback (the sense of touch) can profoundly improve the way we interact with virtual

environments. Systems that support interfaces between a haptic device and a virtual environment are called haptic virtual environments (HVEs). HVE uses include military and space exploration; the sense of touch will also enable blind people to interact with each other within a virtual environment. The HVE modalities include graphics (and possibly video), sound, and force. Recent research [2, 3] has shown that to have a satisfying experience in interacting with an HVE, the graphics and haptic update rates need to be maintained at around 30 Hz and 1 KHz, respectively. In distributed HVEs (DHVE) for remote collaborations, the haptic device is separated from the virtual environment and remotely affects and manipulates it. In DHVEs, one or multiple users may interact with the virtual environment, and possibly with other users with haptic devices. Users may take turns in manipulating a virtual object as in collaborative environments or may simultaneously modify the same object as in cooperative environments [4].

Today most haptic applications are standalone systems. However, it is apparent that the ability to provide distributed haptic applications across a universally accessible medium

such as the Internet will dramatically increase their profile to a much wider range of users. Typically, different types of data are exchanged between hosts in DHVE systems (e.g., graphics, audio, positional information, and reflected force). However in order to produce useful performance, haptic applications require feedback within small and guaranteed timescales to achieve stable haptic interactions. It is clear that the best effort service offered by current IP networks is insufficient to meet the needs of the distributed haptic applications. In order to interact successfully with haptic devices, haptic applications require stringent quality of service (QoS) from the network. Impairments such as time delay, packet jitter, and packet loss each have different (and severe) impacts on remote haptic collaborations. This creates significant challenges but also opens up enormous potential for new applications and new network architectures. Therefore, the effective transmission of haptic data in DHVEs is a new research area which presents a number of challenges to the underlying network. Methods to impart some level of prioritised service into the next generation Internet have resulted in the development of new network architectures that provide different quality of service (QoS) levels for different types of traffic. The most prominent QoS architectures and protocols that are now recommended by the IETF include: RSVP, DiffServ, and MPLS [5–7]. However these have been designed to support the transmission of real-time services such as voice and video. The provision of high (or specific) QoS for multisensory communication and effective human computer interaction has not been addressed to date.

Because it originates from a different human sense (touch), the QoS required to support haptic traffic is significantly different from that used to support conventional real-time traffic such as voice or video. To date there has been little or no attempts to quantify or qualify the QoS requirements of DHVEs. Each network impairment affects the sense of force feedback in a particular way. For example, considerable network delay may make the user feel that the virtual object is heavier. Subsequently, the user tries to push the virtual object with larger force. Delay also desynchronizes the different copies of the virtual environment. Jitter makes the user feel that the object's mass is variable, and can also make the system unstable. Packet loss can reduce the power of the force felt by the user. Previous work [8, 9] suggests that the bandwidth of haptic feel is between 500 Hz and 1 KHz, and that users can tolerate end-to-end delays of approximately 30 milliseconds without much degradation to their perception of force. However subsequent trials [2] have established that they are much more sensitive to network jitter; after 3 milliseconds all the users noticed significant degradation of the force impression, generally in the form of instability in the DHVE or oscillations at the surfaces of virtual objects. The effect of packet jitter can be reduced in real-time voice and video applications through the use of a playout or “jitter buffer”; this approach can also be used for haptic traffic, however in this case it can also significantly increase the delay experienced by DHVE application; there is subsequently a need to define the optimum length of the jitter buffer without affecting the quality of the perceived touch interactions.

Techniques to reduce this delay in the network will therefore benefit the overall quality of interaction with the DHVE. Recent studies by the authors have shown that the haptic experience deteriorates as network-induced packet delay and packet jitter increases beyond 115 milliseconds and 11 milliseconds, respectively [10]. It is recognized that the performance of multimedia traffic can be improved by using QoS architectures that reduce these network impairments [11], and it is therefore expected that the performance of DHVE-based applications can also be enhanced by applying QoS mechanisms. The work presented in this paper presents an investigation into providing specific network QoS (e.g., Diffserv [5]) for haptic traffic.

2. RELATED RESEARCH

A number of systems have been developed specifically for collaboration in virtual environments, including DIVE, CALVIN, and COVEN [12]. Eraslan [13] and Yu et al. [14] investigate the behaviour of a DIVE application in best effort and differentiated services networks with different queuing disciplines. In this work, a DVE application called virtual environment supporting multiuser interaction over IPv6 (VESIR-6) network is deployed. Some experiments have been conducted on network quality issues such as packet loss and delay. The outcome is that IPv6 offers high-quality network infrastructure possibility for real-time DVEs. Allison [15] considers the effects of varying amounts of simulated constant delay on the performance of a simple collaborative haptic task. The task was performed with haptic feedback alone or combined with visual feedback. Subjects were required to pull a virtual linear spring as rapidly as possible, while maintaining a target simulated spring force between their end effectors and that of their collaborators. In their experiment, they incorporate the TiDeC [16] in order to reduce the effect of network delay. TiDeC is a proprietary time-delay compensation system based on prediction of human movement. This does reduce the effect of constant delay, however it neither considers packet loss nor works well with large levels of network jitter. When delay increased, it resulted in a decrease in performance, either in deviation from target spring force and in increased time to complete the task. Performance of TiDeC have been studied and compared with other time compensation techniques, that is, dead reckoning, some results are published in [17]. Traylor [18] describes their recent work with UDP over Ethernet as a communication channel between a remote computer and a custom embedded controller built for a fingerscale 3 DOF force-feedback haptic interface (the 3-DOF ministick). Jay [19] describes an experiment to model the effect of latency across two connected peers sharing a collaborative environment. Although QoS is not considered, their experiment showed that consequences of latency on human interaction can be complex and can vary according to both modality and movement type. The participants in the experiment were clearly able to perceive the effects of delay, and rated the difficulty of the task and the disruption of feedback to be consistently higher with every increment in the level of latency above 50 milliseconds.

Some researchers have attempted to characterize the network parameters required for medical applications that use haptics. In [20] it is reported that a good user experience using a haptic autohandshake requires: 128 kbps bandwidth, packet loss <10%, delay <20 milliseconds and jitter <1 millisecond. Conversely, in order to achieve a good user perception of remote stereo viewing requires: 40 Mbps bandwidth, packet loss <0.01%, delay <100 milliseconds and is not sensitive to jitter. Jeffay [1] and Hudson [21] investigate the problem of supporting continuous data generated by distributed virtual environment (DVEs) applications. They use a nanoManipulator as a haptic device which integrates 3D graphics and force feedback to give a virtual environment interface to scanned probe microscope (SPM). Their experiment described considers the effect of delay and delay-jitter on the haptic force display. Instead of presenting a solid, sharp-edged, stable surface, delayed force feedback results in soft, mushy surfaces, making the use of haptics ineffective or unstable. Their experiments were conducted in a router for three types of flow control: (i) first in, first out (FIFO), (ii) random early detection (RED), and (iii) class-based threshold (CBT). The best QoS was achieved using the CBT flow control with a packet drop-rate of 1.3%, average latency 28.4 milliseconds and an average TCP throughput of 790 kbps.

Nishino et al. [22] propose a new distributed virtual reality architecture to realize a practical system on a dedicated long-haul international network. Some preliminary experiments using the Korea-Japan high-speed research network to validate the proposed method are also mentioned. Their applications handle two tasks, one is a lifting task, and the other is handshaking. The first task can achieve an acceptable rate of completion with up to 32 milliseconds delay, packet loss up to 53%, and jitter up to 60 milliseconds. The second can achieve a reasonable performance with delay up to 13 milliseconds, packet loss up to 40%, and jitter up to 25 milliseconds. Their experimental tasks are mainly based on client-server approaches and the graphical update has to be performed in server side which could result in scalability problems. Cheong [23] uses motion synchronization control with a peer-to-peer shared virtual environment. This type of control can be effective when the round trip delay is less than 300 milliseconds. Lee [24] proposes an intramedia synchronization scheme which adjusts the play out of haptic media according to network delay. Their peer-to-peer architecture describes an adaptive control to reduce the transmission rate by using a buffer and transmission rate control based on number of haptic updates in the buffer.

None of the client-server or peer-to-peer architectures mentioned consider applying QoS to improve their performance. While some of the preceding works have investigated the effects of network impairments on specific haptic applications, over specific communications links, to date there have been no attempts to characterize the levels required of the Internet's QoS mechanisms in order that it can provide service for a complete class of haptic applications, that is, DHVEs. In [25], we presented an investigation into how DHVE traffic could be supported over a QoS-enabled

network. Here we extend this work to include a consideration of how specific network architectures can be deployed to provide the DHVE, and subsequently the QoS required by these architectures. We also consider how these architectures can be used to support a larger number of traffic flows from these types of applications. Our study has been conducted with both experimental and simulation models in order to study the network QoS characteristics required for haptic media in networks carrying multimodal traffic. The contributions of the work presented in this paper are: (i) a new peer-to-peer DHVE application has been developed in order to generate haptic traffic [10], (ii) from analysis of the traffic, a custom OPNET PDF model [3] has been developed and used in the simulations in order to allow us to examine large-scale haptic traffic, (iii) examination of the behaviour of haptic traffic in multimodal systems when carried over an IP network with and without QoS, (iv) an empirical investigation into the network parameters required for haptic traffic transmission over a QoS-enabled IP network, and subsequently (v) we provide recommendations to improve the transmission of haptic traffic by using Class-based weight fair queue (CBWFQ) and an implementation of DiffServ's code point (DSCP) QoS mechanism. The major research objective is therefore to reduce haptic traffic delay and jitter in distributed multisensory environments. The challenge is to apply QoS to this type of traffic and ensure its effective transmission in real time. Finally, we conclude by stating our findings and future work.

3. DISTRIBUTED HAPTIC VIRTUAL ENVIRONMENT ARCHITECTURES

DHVEs support interfaces between multiple haptic devices and multiple virtual environments regardless of geographical constraints. The force feedback device used in this paper is the PHANToM desktop [26] from SensAble Technologies Inc. It is used to manipulate moving virtual objects and to provide the user with feedback from the virtual environment. The PHANToM desktop has an arm workspace of 16 cm × 12 cm × 7 cm and can provide force up to 3.3 N in 3 axis directions; the force computation is based on the spring-damper model [26]. Contact with virtual objects is simulated by computing the force that resists the haptic device's haptic interface point (HIP) from penetrating the virtual object's surface. This approach uses a proxy that transforms the HIP and is referred to as the surface contact point (SCP). The PHANToM desktop has maximum stiffness of (3×1020 N/m) to allow realistic simulation of contact with walls and hard objects. It can generate 1000 packets/s of position and force data during haptic collaboration actions.

DHVEs may have two modes of operation. In collaborative mode users take turns in manipulating the virtual objects while in co-operative mode they can simultaneously modify them [4]. A specific DHVE may operate with just one or both modes. Most collaborative (or co-operative) virtual environments adopt one of two commonly available network distribution architectures: client-server or peer-to-peer. Each architecture has its own specific advantages and shortcomings. Client-server architectures

provide consistency and synchronization among the clients because simulation activities are processed in a centralized server. Also, the required computing power of each client is lower than that required for peer-to-peer systems. The main disadvantage of the client-server approach is that the local view of the environment is only updated after a round-trip to the server, which may impart a significant delay. The client-server architecture also has a scalability problem as the number of clients increase so the load on the server can increase exponentially. We use a peer-to-peer architecture as a DHVE system throughout our studies [10]. Peer-to-peer systems offer the benefits of scalability and decentralized control, however, there are significant challenges associated with synchronizing not only the virtual environments across networked peers, but also the transmitted forces [10].

3.1. Network parameters for DHVE traffic

Network QoS performance is generally described using four basic parameters: (i) delay: the difference between the time when the packet has been sent and the time when it is received. (ii) Jitter: the statistical variance of delay measured as the average time between two successively received IP packets. (iii) Packet loss: expressed as a percentage of the number of packets not received, to the number of packets sent. (iv) Throughput: the number of packets that can be transmitted in a fixed amount of time. Each of these imparts specific effects into a user's experience in a DHVE. Delay makes the user's device go through a virtual object before it is felt. This is because the position of the remote virtual objects is delayed making the user push the virtual object with greater force. This degrades the users' perception of "effective collaboration." Delay also desynchronizes the different copies of the virtual environment. Jitter makes the user feel that the virtual object's mass is variable, and can make the system unstable (e.g., it can produce oscillations on the surfaces of objects). Packet loss can reduce the amount of force felt by the user and changes the apparent weight of objects. The HIP is the representation of the haptic device cursor in the virtual environment, and packet loss can also result in loss of contact between the HIP and the virtual object. Packet loss can also cause abrupt force feedback. A minimum throughput is required for successful transmission of haptic traffic in distributed haptic application. Out-of-sequence packets cause abrupt movements (backwards or forwards) in DHVE applications.

Real time transmission with low latency over long distance is the main challenge for networked haptic applications. The aim of network level QoS is to provide stable bandwidth, controlled jitter (i.e., consistent latency) in addition to improved packet loss. The QoS parameter values for haptic traffic are different from traditional real-time (e.g., VOIP) Internet applications; for example, network latency >50 milliseconds can lead to instability in tele-haptic interaction. The network characteristics considered for the DHVE flows are the bandwidth of the connection, the packet delay, packet jitter, and packet loss. Table 1 shows the DHVE haptic traffic network parameters versus other

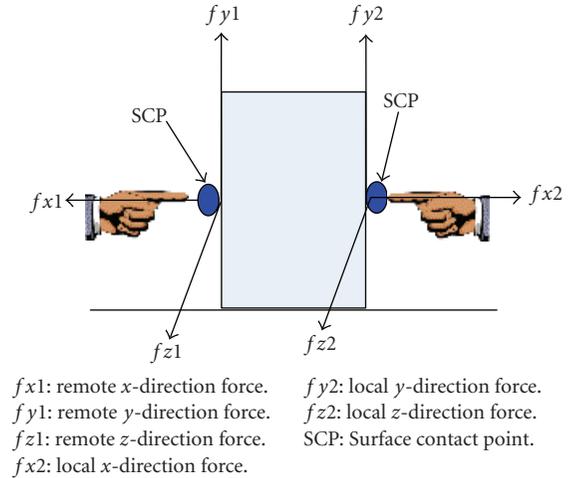


FIGURE 1: Force feedback in collaborative action.

types of network service. It is clear that haptic media is more sensitive to delay and jitter than other traffic types.

4. SYNCHRONIZATION OF POSITION AND FORCE INTERACTION IN HAPTIC COLLABORATIONS

Positional synchronization is a major challenge in distributed shared virtual environments [29]. This becomes even more challenging in peer-to-peer architectures, and any techniques that can reduce the delay and jitter between peers can be expected to improve synchronization and hence the overall system performance. In our peer-to-peer architecture, each peer has their own copy of the virtual environment database. Position synchronization and force collaboration are both implemented using this network architecture. Position synchronization is achieved by transmitting the difference in position, which is calculated from current and previous positions. The difference in position of the local peer is transmitted to the remote peer who adds this difference to its local position in order to achieve position synchronization. When two forces push a virtual object at the same time, their vector sum will decide in which direction the virtual object will move. As shown in Figure 1, the reaction force is computed in proportion to the remote force, the depth of the PHANTOM cursor inside the virtual object, and velocity between the cursor and the virtual object. During collaboration, if local and remote forces are applied to the opposite faces of the cube, they cancel each other. In contrast, if local and remote forces are applied to same face of the cube, the resultant force is the summation of the local force and remote force.

4.1. Position synchronization and force interaction algorithm

Figure 2 shows a time event diagram which illustrates how our algorithm achieves position synchronization and force computation. The next position of the virtual object is the summation of the local position and remote position

TABLE 1: Different traffic types parameters [2, 3, 27, 28].

Traffic	Characteristics	QoS requirements
Haptic	Transmission rate of 1000 packet/sec	Delay < ~50 ms
	Constant packet rate	Throughput ~500 kbps–1 Mbps
	Sensitive to jitter and delay	Jitter < ~2 ms Packet loss < ~10%
Voice	Alternating talk spurts	Delay < ~150 ms
	Silence interval	Throughput ~22 kbps–200 kbps
	Talk-spurts produce constant packet	Jitter < ~30 ms Packet loss < ~1%
Video	Highly bursty traffic	Delay < ~400 ms
	Long-range dependencies	Throughput ~2.5 Mbps–5 Mbps
		Jitter < ~30 ms Packet loss < ~1%
Data	Poisson type	Zero or near-zero packet loss
	Long-range dependencies	Delay may be important

displacement. When the local PHANToM touches a virtual object, the movement of the local virtual object follows the velocity generated by the local force without adding the remote box position displacement. When the local PHANToM is not touching the virtual object, the total movement of the local virtual object is equal to the summation of local and remote position displacement. In terms of force manipulation, when the PHANToM touches the local object, the total force is the addition of both local and remote forces. In contrast, when the local PHANToM does not touch the local object, the total force at the local site is equal to the remote force only. When there are two forces applied to a single virtual object the resultant force is the vector summation of these forces. Therefore, the movement of the virtual object follows that of the resulting force. The differences in position, force, and time at the local peer are sent to the remote peer and vice versa.

Tests have been conducted to evaluate the performance of this synchronization algorithm. Figure 3 provides some results that can represent the accuracy of the algorithm. The X-position discrepancy of a moving virtual box is obtained by capturing (in real time) the X-position of the virtual cube across two networked peers, computer A and computer B. Figure 3(a) shows the x-position trajectory of the virtual cube. Figure 3(b) shows that there is less than 5mm discrepancy in the X-position of each peer, which is very low. In addition, the coefficient of correlations [30], which shows the covariance of the X-position between computer A and B, is 0.99760. This is to demonstrate if any linear relationship of the two X-position values at computer A and B can be obtained. The correlation is 1 in the case of an increasing linear relationship, -1 in the case of a decreasing linear relationship, and 0 in the case of an independent correlation. A correlation coefficient of 1 indicates a perfect match between the X-position values. Other values between all these cases indicate the degree of linear dependence between the X-position values. Thus, the high value shows that the X-position trajectories of the two networked peers

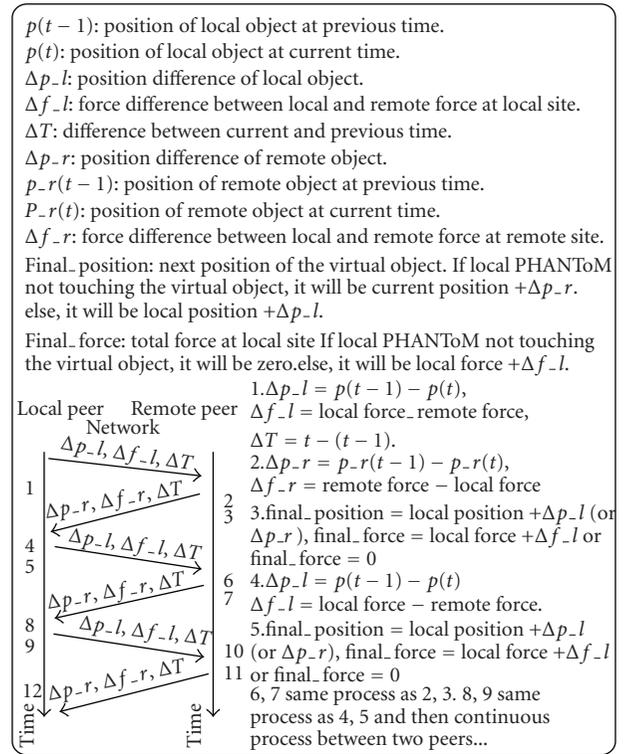


FIGURE 2: Position, force, and time events on local and remote peers.

are very closer to each other and hence the algorithm is working well.

5. EXPERIMENTAL AND SIMULATION ARCHITECTURES

Figure 4 shows the approach taken. Haptic traffic from a DHVE application was first captured in an experimental test bed, and the subsequent traffic patterns analyzed.

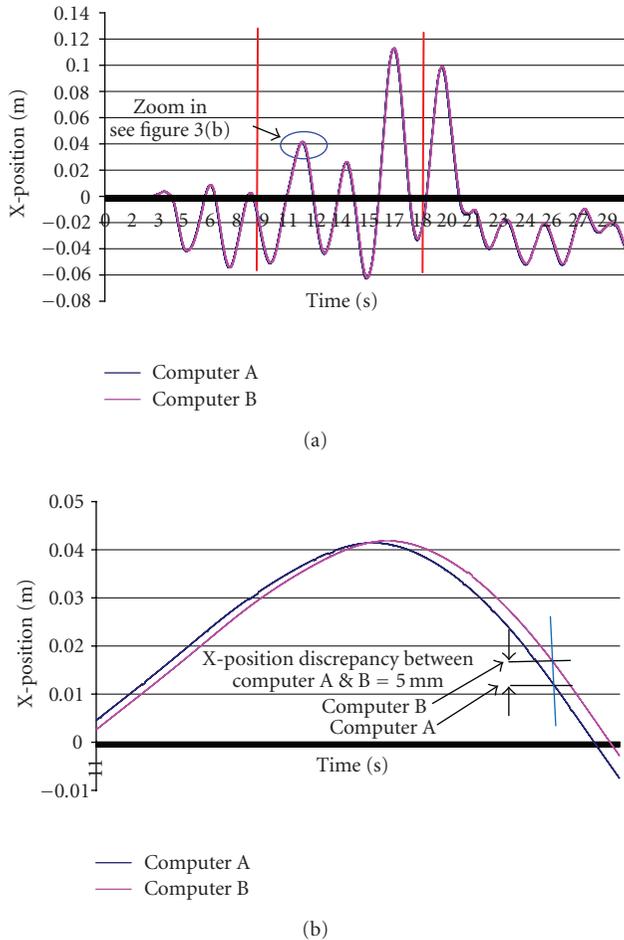


FIGURE 3: (a) X-position discrepancy of the two networked peers A & B with no network impairments. (b) X-position discrepancy of the two network peers A & B, zoom in section of (a).

A custom PDF model was then created for use in the network simulation tool OPNET [3]. A simulation model of DHVE applications running over a network was then developed. The OPNET network model is similar to the experiment test bed. The PDF model is used to generate haptic application traffic to run in the simulated DiffServ network. Subsequently, the effect of running haptic traffic over a DiffServ IP network is obtained. This approach is used to overcome some of the limitations of a test bed. Using this, we are able to simulate a larger scale DHVE environment without the restriction of physical resources. However, the limitation of the simulation model is that we cannot simulate the user's haptic perception which is something that can only be studied in a real-world environment. The experimental architecture applies a QoS mechanism (in the form of Class-based WFQ) which is able to reduce the delay of the haptic traffic and so improve the user's haptic experiences.

5.1. Experimental architecture

The objective of the experimental system is to enable us to generate haptic traffic and study the network QoS

characteristics for haptic traffic transmission over a QoS-enabled IP network. We use Matlab Simulink, Real-time Workshop v6.1, Virtual Reality Toolbox v4.0.1, and the proSENSE toolbox from HandshakeVR [16] to develop our experimental system. This experimental architecture is scalable and the current research mainly focuses on 2 users which will be extended to multiusers in the forthcoming work.

5.1.1. Design of experimental system

We have developed an experimental platform based on a peer-to-peer network architecture in order to study network QoS characteristic for haptic traffic. A difference is that in our operation, we transmit the haptic interface point (HIP) position, the virtual objects' positions, timestamp, and the force vectors between the networked peers. In Figure 5, the PCs are running with the VR environment and haptic rendering (e.g., workstation #1 and workstation #2). In this case, the force feedback device used is the PHANToM omni [26] from SensAble Technologies Inc. This is used to manipulate moving virtual objects and to provide the user with force feedback from the virtual environment when the HIP touches a virtual object. The PHANToM Omni generates 1000 packets/s of position and force data during haptic collaboration actions. The workstation connects to the PHANToM Omni through a FireWire interface. Haptic traffic flows between workstations 1 & 2 over the Ethernet network connection. The complete DHVE network architecture which is based on this basic architecture is described in the next section.

5.1.2. Experimental system overview

In Figure 6, four computers are connected through a bottleneck Ethernet link. The gigabit link is running on limited bandwidth of 10Mbps through the two Cisco routers A and B. The experimental hardware is comprised of haptic devices, host and target system hardware, background traffic generator hardware, and network devices. In the test bed, the host and target system is executed in the same PC (i.e., PC 1 and PC 2). A network monitoring tool called "IP Traffic" [31] is used as the traffic generator software as well as being used as the traffic capturing tool.

In operation, PCs 1 and 2 are running DHVE Matlab applications, and PCs 3 and 4 function as background traffic generators for the bottleneck link. The mean background traffic setting is throughput 10137 kbps, packet size 1460 bytes, and with UDP protocol. The buffer sizes (transmitter and receiver) in each interface of the switches, router A and B are zero. The haptic traffic is given various CBWFQ weights in contrast with a constant background traffic weight. Figure 7 shows the Matlab haptic environment which consists of a virtual environment workspace comprising one moving cube, one static cube and two ball spheres which represent local and remote PHANToM cursors (HIPs). The size of the virtual cubes is 4 cm \times 4 cm \times 4 cm. The workspace boundary is 7 cm on each side. The cubes are modeled to simulate the mass, damping, form, position, velocity, and

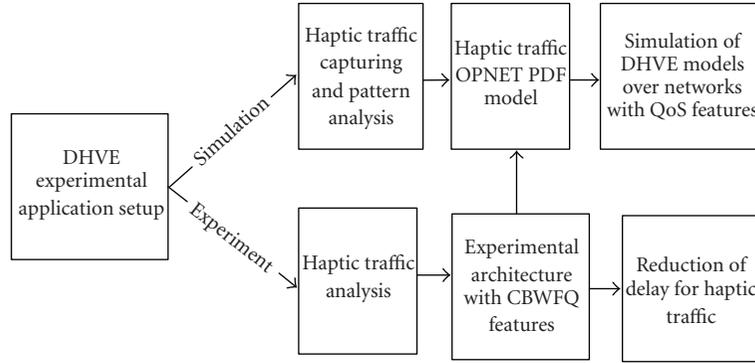


FIGURE 4: Experiment and simulation approaches.

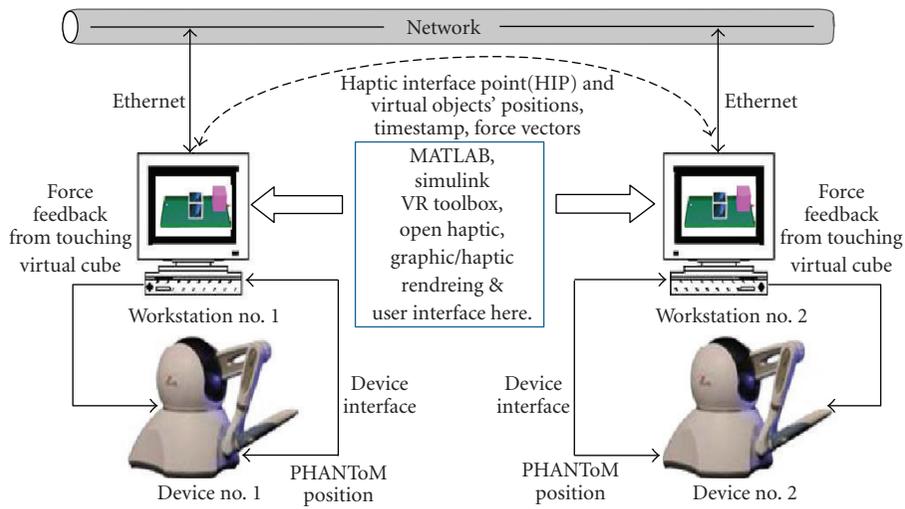


FIGURE 5: Design of distributed haptic peer-to-peer application.

acceleration of the dynamic virtual objects. Their physical properties are: mass = 5 kg, stiffness = 300 N/m, and damping factor = 7, respectively.

In Figure 8, subjects are able to feel the two virtual cubes but not the work platform; each peer has the same virtual environment as shown in Figure 7. The blue cube in the middle of Figure 7 is movable and whereas the pink cube on the right hand side of the subject is static. Subjects are able to push the moving blue cube by using two PHANToM devices and feel the momentum, force, and velocity of the virtual cube. In addition, they are able to perform collaborative and co-operative tasks on one or both cubes. When running, users at PC 1 and PC 2 push the 3D cubes in the virtual environment, (see Figure 8) force is generated at a PHANToM whenever its HIP touches a virtual cube. This force data together with the HIP and virtual objects' positions are transmitted from PC1 to PC2 and vice versa.

Previous works on distributed haptic environments have concentrated on synchronization of positions (haptic device or virtual objects) [32]. The peer-to-peer architecture presented here further extends this to enable the force interaction between two users. Thus, the force data is sent

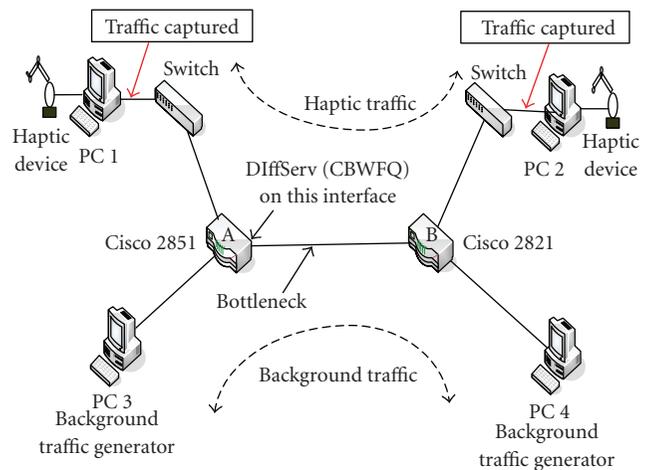


FIGURE 6: Experimental model of distributed peer-to-peer architecture.

over to remote peer in addition to the position information. From this, the traffic flows were found to require 736 kbps

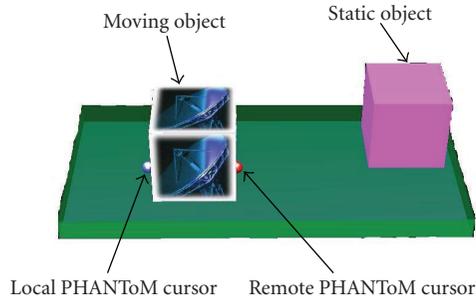


FIGURE 7: Snapshot of the implemented collaborative haptic virtual environment.

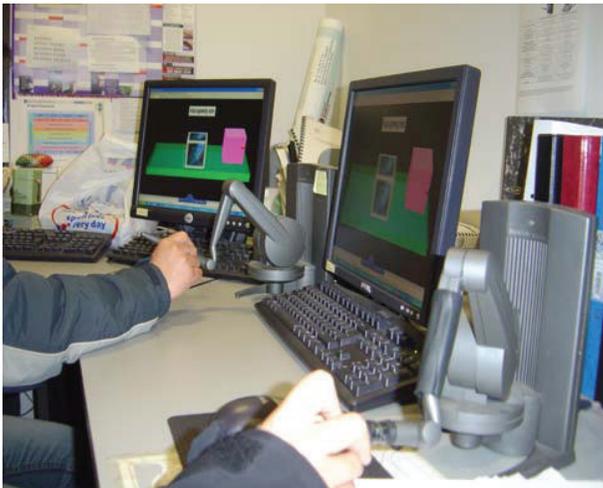


FIGURE 8: Two users are collaborating a moving cube in the DHVE.

for haptic traffic in each direction. From analysis of the distributions of the traffic delay and bandwidth, subsequent PDF models for use OPNET were developed and used to simulate haptic traffic along with other multimedia traffic sources, in the network simulator.

5.1.3. Haptic traffic queue configurations

In the experimental test bed, DHVE traffic is classified and prioritized in the routers using Class-based weight fair queuing (CBWFQ) from Cisco systems [33]. CBWFQ is a congestion management mechanism that is offered by Cisco for its router platforms and is typical of the QoS mechanisms found in today's routers. It is not available in the OPNET [34] modeling environment, however, it is based on proportionally-fair fluid-flow packet scheduling techniques similar to weighted fair queuing (WFQ), and both CBWFQ and WFQ provide similar functionalities for traffic queuing and bandwidth management. CBWFQ extends WFQ by allowing users to define the classes used in WFQ. The classes can be determined by protocol, access control lists (ACLs), IP precedence, or input interface. Each class can be allocated different bandwidth guarantees in terms of its scheduler queue weight. This approach allows greater control of the haptic traffic when it is received together with

other traffic. Figure 9 shows the processing applied to haptic traffic packets at the ingress to a number of interfaces in a router; traffic is forwarded to a specific interface according to its type, where it is subsequently classified and scheduled. The priority queue is served as long as it is nonempty; the CBWFQ queues are then served in proportion to their weights. When CBWFQ queues have consumed any reserved bandwidth or become empty, the best effort queue is then served.

Figure 9 shows the queue setup of haptic and background traffic for the output port (egress port) of Cisco router A in the experimental test bed shown in Figure 6. The haptic traffic class is set with weights of 0, 1, 5, 10, 15, or 30. A Similar setup is applied to the egress interface of Router A in the simulation model shown in Figure 10. Voice and video applications are treated in different classes during WFQ classifications. The background traffic class is set to best effort. In order to improve the haptic traffic transmission under background traffic load, the CBWFQ weight of the haptic traffic class was varied. The haptic class weight was not set higher than 30 because after that the delay was found to be almost zero. This is because the CBWFQ guarantees enough bandwidth (736 kbps in our application) for haptic traffic. The percentage background traffic is calculated with the ratio of 10 Mbps. For example, 10% background traffic will generate 1 Mbps from router A to router B.

5.2. DHVE simulation model

Current network simulators are designed to model existing traffic types such as voice, video, and data traffic, and as such there are no models to represent haptic traffic. As there was no generalized distribution model that is able to represent haptic traffic in OPNET, a custom probability density function (PDF) model was created. Details of this model are presented in [3]. In order to customize a simulation haptic network model, empirical haptic traffic is captured from the test bed, analyzed and then the OPNET PDF model is created. This is then applied as traffic in the network simulation. Figure 10 shows eighteen PCs connected with two switches and routers. The two routers A and B are connected across a 10 Mbps link in order to study the effect of WFQ on haptic traffic. The link creates a bottleneck between the routers; background traffic builds up traffic congestion at router A and thus permits the implementation of WFQ at the egress interface of router A. The other network links are 100 Mbps. The haptic domains 1, 2 are configured to run a custom application task that simulates a DHVE application by using the custom OPNET PDF model. In addition, there are PCs running video, audio, FTP, Email, HTTP, and database applications as multimedia traffic flows. In this case, video and audio have been set with streaming traffic. The system is running with weight fair queuing (WFQ) enabled in the output interface of router A as shown. WFQ dynamically classifies network traffic into individual flows and assign each flow a fair share of the total bandwidth. Unlike priority queuing, each flow is serviced in according to their weight. The weight assigned to haptic traffic is then increased in steps. Additionally, a low-latency queue (LLQ)

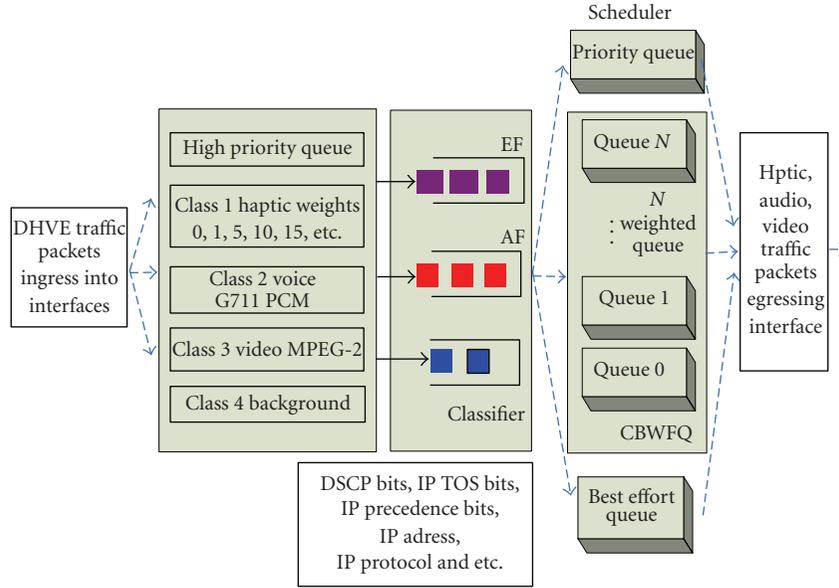


FIGURE 9: Diffserv treatments for haptic, audio, video and background traffic packets.

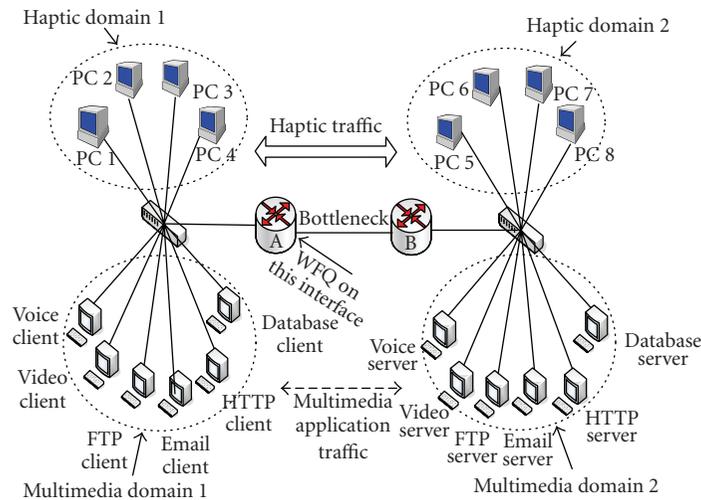


FIGURE 10: Distributed haptic virtual network simulation model with audio, video, ftp, http, and database applications.

provides a priority queue function which is equivalent to Diffserv’s “Expedited forwarding” (EF) queue.

6. EXPERIMENTAL AND SIMULATION RESULTS

6.1. Experiment results

As shown in Table 2, the haptic traffic effective throughput is reduced sharply at 97%–99% background traffic load because of traffic congestion starving the bandwidth available for these flows. The effective traffic throughput is also reduced significantly at 95%-96% background load. The effective traffic throughput is maintained at 1000 packets/s at up to 90% background traffic load. In summary, the packet effective throughput from each DHVE machine drops significantly above 90% background load. From the physical

experiment, it was observed that at these levels, the user will feel vibration in the PHANToM and also large abrupt force feedback. At this point, the haptic system becomes unstable and the PHANToM is not able to hold stable at position because it keeps vibrating. This highlights that there is a minimum bandwidth required for a DHVE application.

Figure 11 shows the experiment results when haptic traffic is allocated CBWFQ bandwidth weights of 1, 5, 10, and 30. The result shows that when the haptic traffic is given higher bandwidth, the packet transit delay is reduced. In Figure 11, haptic traffic end-to-end delay increases to 200 milliseconds whenever background traffic increases and the haptic traffic is under best-effort treatment. This means that the router A in Figure 6 has not been set with any QoS mechanisms. When CBWFQ is employed, the delay of haptic traffic is reduced from 200 milliseconds (best effort),

TABLE 2: Haptic traffic with background load.

Background loading (%)	Packet rate (packet/s)
60	981.957045
70	991.862847
80	999.131579
90	987.389474
95	793.389474
96	509.647368
97	77.894737
98	37.368421
99	60

↑
User feels abrupt force feedback, haptic system becomes unstable

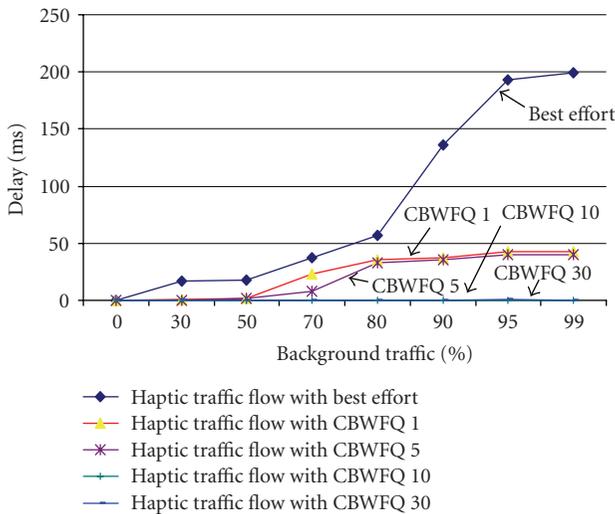


FIGURE 11: Haptic traffic end-to-end delay versus background load with different WFQ weights at router A.

to less than 1 millisecond (CBWFQ weight = 30) under a background load of 95% load. Setting the CBWFQ haptic weight = 1 with a guaranteed bandwidth of 1 Mbps results in a significant improvement over best effort, and setting CBWFQ haptic weights of 10 and 30 can definitely reduce the delay further as shown.

6.2. Simulation results

This section investigates the haptic traffic characteristic with WFQ-enabled on output interface of router A in Figure 10. Figure 12 shows end-to-end delay of individual haptic, voice and video traffic flows with 45% background traffic loading on the bottleneck link (10 Mbps). The combined flows increase the bottleneck link utilization up to 98%. Figure 12 shows that with a best-effort only service, haptic traffic incurs nearly 730 milliseconds of end-to-end delay which is totally unacceptable for a haptic operation. The simulation results shown in Figure 13 are the end-to-end delay of the haptic, voice, and video traffic flows with different WFQ weights. The results are obtained by varying the weights

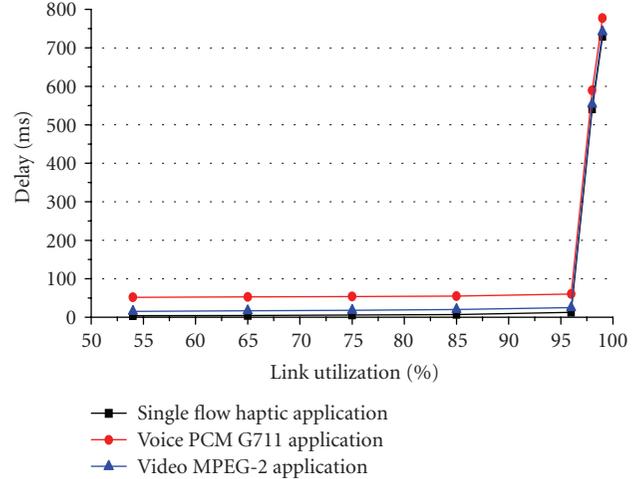


FIGURE 12: End-to-end traffic delay versus link utilization.

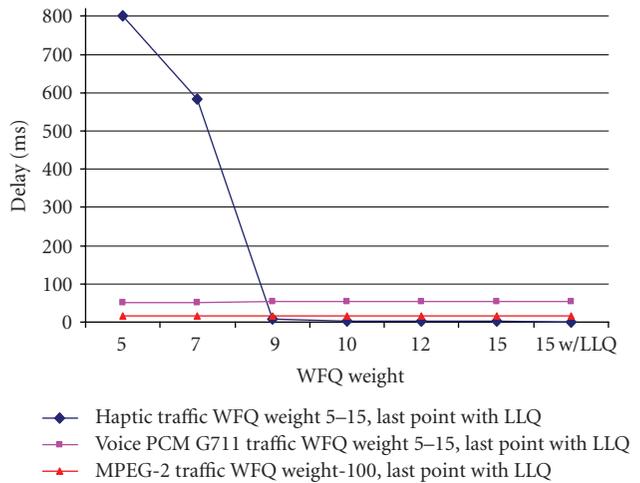


FIGURE 13: End-to-end delay of haptic, voice and video traffic with different WFQ weights.

for all other traffic (except haptic traffic) flowing through the output interface of router A. The audio and video traffic flows have been set to achieve end-to-end delays of below 100 milliseconds (which is reasonable for audio and video streaming applications). The WFQ weight ranges from best effort (WFQ = 0) to WFQ weight = 15. Initially, the best effort IP network caused end-to-end delay of 800 milliseconds in the haptic traffic; however, this delay is improved by introducing prioritised service class for haptic traffic. It can be observed that the end-to-end delay of the haptic traffic has decreased from 800 milliseconds (WFQ weight = 2) to 1.14 milliseconds (WFQ weight = 15). The delay is further reduced to 0.7 milliseconds with the low-latency queue (LLQ) enabled on the interface. This result shows that the introduction of WFQ improves the QoS provided to the haptic traffic.

Figure 14 shows the throughput of haptic traffic at the Ethernet layer; it highlights the reduction in throughput when WFQ weight < 9. The result is obtained by setting

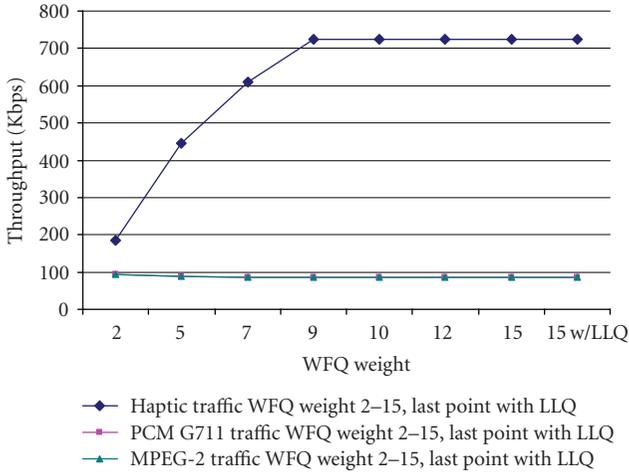


FIGURE 14: Haptic, voice, and video traffic throughput with different WFQ weights.

different WFQ weights for the haptic, voice, and video traffic. The haptic packets are 64 bytes, which become 92 bytes at Ethernet layer. Therefore, the total throughput at the Ethernet layer is $92 \times 1000 \times 8 = 736$ Kbps. This is closely matched to the values in Figure 14.

Figure 15 shows the relationship between the rate of the haptic traffic received whenever the rate of the bottleneck link is reduced from 10 Mbps to E1(2.048 Mbps) and/or T1(1.544 Mbps). This is shown for multiple haptic network flows (up to 10 flows). As shown, the rate of received haptic traffic is reduced sharply when there are three flows or more, because of traffic congestion starving the bandwidth available for these flows. It is important for a remote haptic receiver to receive around 1000 packets/s in order for the haptic application to maintain a constant network throughput of 850 Kbps. This in turn helps maintain the local haptic feedback control loop and so eliminate instability. The T1 link exhibits poorer performance than the E1 link simply because an E1 link has higher bandwidth capacity than a T1 link. In summary, the packet rate received from each DHVE machine drops significantly when there are two haptic flows. From the physical experiment, it was observed that at these levels, the user would feel vibration in the PHANToM and also large abrupt force feedback.

6.3. Discussion of simulation and experiment results

Sections 6.1 and 6.2 have presented the experimental and simulation results, respectively. While both approaches have been shown to yield comparable results, there are some discrepancies. The end-to-end delay of the simulated haptic traffic decreases to about 2 milliseconds when the WFQ weight is 9. The test bed result shows that the end-to-end delay drops to 1 millisecond when CBWFQ weight = 10. This is because CBWFQ can allocate a minimum amount of bandwidth in which haptic traffic has exclusive use. Thus, the simulation model is comparable with the experiment test bed although it contains more traffic sources than the

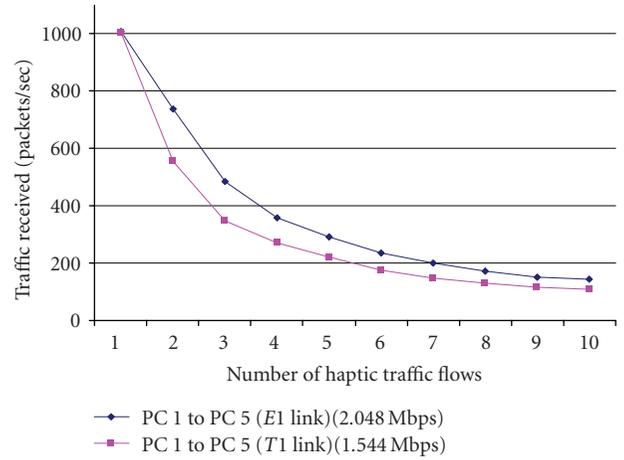


FIGURE 15: Received traffic rate of E1, T1 Link with multiple haptic flows.

experiment test bed. The haptic traffic is therefore able to improve its transmission quality if given a minimum amount of network bandwidth. This is shown in previous sections for both experiment and simulation results. From the test bed experiments, the user haptic perception is improved when CBWFQ is enabled in the network, as compared to a best-effort-only service.

We have also studied the consequence of using DSCP for haptic traffic. This is shown in Table 3. The haptic traffic is studied for maximum end-to-end delay under different AF and EF of DSCP Markings. EF with low-latency queue (LLQ) provides highest priority thus yielding lowest delay. Table 3 shows that the AF21-AF23, AF31-AF33, AF41-AF43, and EF have lower end-to-end delays compared to AF11-AF13. Therefore, AF11-AF13 are not recommended for transmission of haptic, audio or video traffic. This is in agreement with IETF recommendations (RFC4594) for the transport of video and voice traffic [6]. The maximum end-to-end delay also depends on the type of link used and the traffic loading. In this case, we used haptic traffic, real time audio and video streaming traffic plus the other multimedia traffic in our simulation model. A similar result was presented in our paper in Immerscom 2007 [25] in which haptic traffic was investigated with T1 (1.544 Mbps) and T3 (44.736 Mbps) link. BE and AF11-AF13 are not recommended for transmission of haptic traffic. This result confirms the recommendations are also valid for voice and video traffic.

The results in Figure 14 showed that the haptic traffic for our DHVE application has a throughput of 736 kbps. Therefore, it is important to reserve this minimum bandwidth in order for the haptic traffic to be effectively transmitted. This is comparable to the results obtained from the test bed which shows that a CBWFQ of 1 will guarantee more than enough bandwidth (736 kbps in our application) for haptic traffic and hence also reduce the delay of haptic traffic as is shown in Figure 11. The results confirm that haptic traffic is comparable to telephony or video classes but it is very

TABLE 3: . Maximum end-to-end delay for haptic, voice, and video traffic using different DiffServ Code Point AF and EF marking.⁽¹⁾

DSCP	End-to-end delay (millisecond)		
	Haptic	Voice	Video
BE	1206.8306	1254.7928	1023.2244
AF11	217.8382	284.8830	235.8420
AF12	217.8382	284.8330	235.8420
AF13	217.8382	284.8330	235.8420
AF21	1.7631	49.4548	6.5655
AF22	1.7631	49.4548	6.5655
AF23	1.7631	49.4548	6.5655
AF31	1.6851	49.3732	5.8811
AF32	1.6851	49.3732	5.8811
AF33	1.6851	49.3732	5.8811
AF41	1.6382	49.3348	5.5212
AF42	1.6382	49.3348	5.5212
AF43	1.6382	49.3348	5.5212
EF	1.5952	49.2930	4.9335
EFLQ	1.5868	49.2871	4.8084

⁽¹⁾ BE—best effort, AF—assured forward, EF—expedited forward, LLQ—low-latency queue, link T3—44.736 Mbps, 95% link utilisation.

TABLE 4: Proposed haptic class with DSCP marking scheme in addition to DiffServ service classes and DSCP marking scheme in [28].

Service class	Traffic characteristics	Tolerance to			Protocol	DSCP
		Loss	Delay	Jitter		
Haptic	Fixed packets, real-time, inelastic and constant rate flows	Very low	Very low	Extreme low	UDP	EF
Telephony	Fixed size small packets, inelastic and low-rate flows	Very low	Very low	Very low	UDP	EF
Multimedia streaming	Variable size packets, elastic with variable rate	Low-medium	Medium	Yes	UDP	AF31 AF32 AF33
Low-priority data	Nonreal time and elastic	High	High	Yes	N/A	BE

sensitive to jitter [8, 11]. Based on our findings, we proposed a DSCP marking scheme for haptic traffic. The requirement for using haptic traffic in a managed network by the network administrator is proposed in Table 4. The haptic class is proposed to have a DSCP marking of EF or at least AF21 and above.

7. CONCLUSIONS AND FUTURE WORK

This paper presents a study into the provision of QoS for (DHVEs) whenever they are provided over QoS-enabled packet switched networks such as the next generation Internet. Moreover the study is particularly relevant to DVHEs that are implemented as networked peers rather than traditional client-server architectures. A new peer-to-peer DHVE architecture that permits peers distributed across an IP network to perform collaborative and co-operative haptic tasks on virtual objects is presented. The provision of QoS for these types of applications is then investigated. The

approach taken employs an experimental test bed network to gather empirical data concerning the statistical distribution of haptic traffic generated by the networked peers. This is then used to generate a traffic model for haptic traffic which is used in a network simulation to analyze the performance of DHVE traffic flows across networks that are QoS-enabled. Haptic traffic is simulated along with voice (G711), video (MPEG-2) and other multimedia traffic. Suitable values for the network-level parameters for haptic traffic are then developed and recommendations are proposed to provide QoS for multimodal traffic flows. The work involves studies of haptic traffic under a best-effort IP network and a DiffServ IP network.

The results show that the network simulation model compares favourably with the physical network, and can be used to generate a scalable haptic network model where multiple connections carrying haptic traffic may be examined. Both approaches show that reducing network delay and jitter by providing “better-than-best-effort” service

which uses specific QoS classes for haptic traffic can lead to improvements in users' haptic experiences with distributed applications such as virtual environments.

The simulation results show that haptic throughput increases correspondingly to an increase in the queue scheduling weight. In the experimental test bed, the end-to-end delay experienced by haptic traffic is found to decrease from 200 milliseconds (best effort) to 40 milliseconds (CBWFQ) by running the haptic application in a DiffServ network. Both simulation and experimental results show that transmission of haptic traffic is improved with implementation of a traffic classification and prioritization mechanism (WFQ and CBWFQ, resp.). The simulation model can be used to simulate large numbers of haptic traffic flows. The results from this lead to the conclusion that WFQ and CBWFQ in DiffServ packet switched network improve network performance for transporting haptic traffic by proper setting of the DiffServ DSCPs and the packet schedulers in the routers. Subsequently, a haptic traffic class with DSCP marking scheme is proposed. This can be used as a reference for configuring a QoS-enabled network to support DHVE applications or multimodal traffic flows.

In the future, we intend to investigate haptic user perception tests on the possibility of a DiffServ-enabled IP QoS network that allows consistency force and position collaboration among multiple (>2) users. In addition, we will study the application of weight random early detection (WRED) and interleaving, which are specifically configured to improve haptic traffic under congestion conditions that may result in bursty packet loss in a network.

ACKNOWLEDGMENT

This work was supported by the Industrial Research and Technology Unit Northern Ireland SPUR scheme at Queen's University Belfast.

REFERENCES

- [1] K. Jeffay, T. Hudson, and M. Parris, "Beyond audio and video: multimedia networking support for distributed, immersive virtual environments," in *Proceedings of the 27th Euromicro Conference (EUROMICRO '01)*, pp. 300–307, Warsaw, Poland, September 2001.
- [2] R. Souayed, D. Gaiti, W. Yu, G. Dodds, and A. Marshall, "Experimental study of haptic interaction in distributed virtual environments," in *Proceedings of Eurohaptics*, pp. 260–266, Springer, Munich, Germany, June 2004.
- [3] K. M. Yap, A. Marshall, W. Yu, G. Dodds, Q. Gu, and R. T. Souayed, "Characterising distributed haptic virtual environment network traffic flows," in *Proceedings of the 4th IFIP Conference on Network Control and Engineering for QoS, Security and Mobility*, pp. 297–310, Lannion, France, November 2007.
- [4] P. Buttolo, R. Oboe, and B. Hannaford, "Architectures for shared haptic virtual environments," *Computers & Graphics*, vol. 21, no. 4, pp. 421–429, 1997.
- [5] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," RFC 2475, December 1998.
- [6] J. Babiarz, K. Chan, and F. Baker, "Configuration guidelines for diffServ service classes," RFC 4594, August 2006, <http://www.ietf.org/rfc/rfc4594.txt>.
- [7] R. Braden, D. Clark, and S. Shenker, "Integrated services in the internet architecture: an overview," RFC 1633, Network Information Center, SRI International, Menlo Park, Calif, USA, June 1994.
- [8] S. Matsumoto, I. Fukuda, H. Morino, K. Hikichi, K. Sezaki, and Y. Yasuda, "The influences of network issues on haptic collaboration in shared virtual environments," in *Proceedings of the 5th Phantom Users' Group Workshop (PUG '00)*, pp. 22–24, Aspen, Colo, USA, October 2000.
- [9] K. Hikichi, I. Arimoto, H. Morino, K. Sezaki, and Y. Yasuda, "The evaluation of adaptation control for haptics collaboration over the internet," in *Proceedings of International CQR Workshop*, Okinawa, Japan, May 2002.
- [10] K. M. Yap, A. Marshall, and W. Yu, "Supporting haptic collaboration across networked peers with real time force feedback in distributed virtual environments," in *Proceedings of the 14th IEEE International Conference on Telecommunications (ICT '07)*, pp. 494–499, Penang, Malaysia, May 2007.
- [11] F. Naït-Abdesselam and N. Agoulmine, "QoS control and adaptation in distributed multimedia systems," in *Proceedings of the 11th International Parallel Processing Symposium in Conjunction with the 13th International Parallel Processing Symposium and the 10th Symposium on Parallel and Distributed Processing (IPPS '97)*, pp. 375–383, Springer, San Juan, Puerto Rico, USA, April 1999.
- [12] J. C. de Oliveira, S. Shirmohammadi, and N. D. Georganas, "Collaborative Virtual Environment standards: a performance evaluation," in *Proceedings of the 3rd IEEE International Workshop on Distributed Interactive Simulation and Real-Time Applications*, pp. 14–21, Greenbelt, Md, USA, October 1999.
- [13] M. Eraslan, N. D. Georganas, J. R. Gallardo, and D. Makrakis, "A scalable network architecture for distributed virtual environments with dynamic QoS over IPv6," in *Proceedings of the 8th IEEE International Symposium on Computers and Communications (ISCC '03)*, vol. 1, pp. 10–15, Kemer-Antalya, Turkey, June-July 2003.
- [14] H. Yu, Q. Zhou, D. Makrakis, N. D. Georganas, and E. Petriu, "Quality of service support of distributed interactive virtual environment applications in IP networks," in *Proceedings of IEEE Pacific RIM Conference on Communications, Computers, and Signal Processing (PACRIM '01)*, vol. 1, pp. 196–199, Victoria, Canada, August 2001.
- [15] R. S. Allison, J. E. Zacher, D. Wang, and J. Shu, "Effects of network delay on a collaborative motor task with telehaptic and televisual feedback," in *Proceedings of the ACM SIGGRAPH International Conference on Virtual Reality Continuum and Its Applications in Industry (VRCAI '04)*, pp. 375–381, Singapore, June 2004.
- [16] HandshakeVR, TiDeC™ is trademark of HandshakeVR, <http://www.handshakeVR.com/>.
- [17] K. M. Yap, A. Marshall, and W. Yu, "Supporting haptic collaboration across networked peers with real time force feedback in distributed virtual environments," in *Proceedings of the 14th IEEE International Conference on Telecommunications (ICT '07)*, pp. 494–499, Penang, Malaysia, May 2007.
- [18] R. M. Traylor, D. Wilhelm, B. D. Adelstein, and H. Z. Tan, "Design considerations for stand-alone haptic interfaces communicating via UDP protocol," in *Proceedings of the 1st Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC '05)*, pp. 563–564, Pisa, Italy, March 2005.

- [19] C. Jay, M. Glencross, and R. Hubbard, "Modeling the effects of delayed haptic and visual feedback in a collaborative virtual environment," *ACM Transactions on Computer-Human Interaction*, vol. 14, no. 2, article 8, pp. 1–31, 2007.
- [20] P. Dev, D. Harris, D. Gutierrez, A. Shah, and S. Senger, "End-to-end performance measurement of internet based medical applications," in *Proceedings of the Annual Symposium of the American Medical Informatics Association*, pp. 205–209, San Antonio, Tex, USA, November 2002.
- [21] T. Hudson, M. C. Weigle, K. Jeffay, and R. M. Taylor II, "Experiments in best-effort multimedia networking for a distributed virtual environment," in *Multimedia Computing and Networking*, vol. 4312 of *Proceedings of SPIE*, pp. 88–98, San Jose, Calif, USA, January 2001.
- [22] H. Nishino, S. Yamabiraki, Y.-M. Kwon, Y.-M. Okada, and K. Utsumiya, "A distributed virtual reality framework for Korea-Japan high-speed network testbed," in *Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA '06)*, vol. 1, pp. 433–438, Vienna, Austria, April 2006.
- [23] J. Cheong, S.-I. Niculescu, and M. A. Srinivasan, "Motion synchronization in virtual environments with shared haptics and large time delays," in *Proceedings of the 1st Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC '05)*, pp. 277–282, Pisa, Italy, March 2005.
- [24] S. Lee and J. Kim, "Intra-media synchronization scheme for haptic interactions in distributed virtual environments," in *Multimedia Systems and Applications VIII*, vol. 6015 of *Proceedings of SPIE*, pp. 1–12, Boston, Mass, USA, October 2005.
- [25] K. M. Yap, A. Marshall, and W. Yu, "Providing QoS for distributed haptic virtual environments in IP networks," in *Proceedings of the ICST/ACM 1st International Conference on Immersive Telecommunications (Immerscom '07)*, Bussolengo, Italy, October 2007.
- [26] SenSable Technologies, <http://www.sensable.com/>.
- [27] D. P. Bertsekas, "Traffic behavior and queuing in a QoS environment," in *OPNETWORK*, Washington, DC, USA, August 2005.
- [28] J. Babiarz, K. Chan, and F. Baker, "Configuration guidelines for diffServ service classes. draft-baker-diffserv-basic-classes-03," July 2004, <http://www.tools.ietf.org/html/draft-baker-diffserv-basic-classes-03>.
- [29] Y. Ishibashi, K. Tomaru, S. Tasaka, and K. Inazumi, "Group synchronization in networked virtual environments," in *Proceedings of the IEEE International Conference on Communications (ICC '03)*, vol. 2, pp. 885–890, Anchorage, Alaska, USA, May 2003.
- [30] <http://www.en.wikipedia.org/wiki/Correlation>, January 2007.
- [31] IP Traffic Test Software, ZTI Telecom, <http://www.zti-telecom.com/pages/main-ip.htm>.
- [32] R. Iglesias, S. Casado, T. Gutiérrez, A. García-Alonso, W. Yu, and A. Marshall, "Simultaneous remote haptic collaboration for assembling tasks," *Multimedia Systems*, vol. 13, no. 4, pp. 263–274, 2008.
- [33] Cisco Systems, <http://www.cisco.com/>.
- [34] OPNET 11.0A, Product Documentation, Methodologies and Case Studies, <http://www.opnet.com/>.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

