Abstract

This paper reports on the performance of Distributed Haptic Virtual Environment (DHVE) applications deployed over policy-based IP networks. Our previous work reported that a best effort service in IP networks is inadequate for satisfying the Quality of Service (QoS) requirements of such applications. The current work tries to find out whether applying differentiated Services (DiffServ) satisfies DHVE QoS requirements or at least improves them. This work also determines which policies should be applied on the network. A set of experiments was conducted over a network connecting two departments of the Queen’s University of Belfast. The results are presented in this paper and compared with our previous work results.

Keywords

Haptic Virtual Environment, PHANToM, Force Feedback, QoS, DiffServ, IP Networks.

1. Introduction

Distributed Haptic Virtual Environments (DHVEs) [2] are systems that provide a user an illusion of sharing the same mediated space with other people who may be geographically distributed all over the world. These environments support haptic devices that reinforce, in addition to video and audio, the user’s feeling of telepresence. DHVEs include training and simulation of telemedicine, telerobotics/teleoperation, etc.

Unlike non-interactive media (audio/ video media) which changes its state over time without correspondence to external events, DHVEs are classified as interactive media where the system state changes in response to external events to the media or over time [8]. This leads to more complex mechanisms for loss detection and reliability than those for non-interactive media. DHVE applications also have more critical networking requirements in terms of packet loss, delay, jitter and bandwidth. They may also require multicasting capabilities over the network, as well as a certain level of security according to the transmitted data type. Failure in satisfying these requirements may lead to destabilization of a control system, desynchronization of the environment between participants, or possibly to a disaster when it concerns teleoperation for example. Multicasting and security issues are part of our future work.

The standard Internet Protocol (IP) provides by default a best effort data delivery which maintains simplicity in the core network and requires more complexity in the end-hosts. As the network becomes loaded, the service level degrades, and applications such as DHVE cannot adapt to inconsistent service levels. In order to provide an adequate and consistent service to such sensitive applications, qualitative/quantitative services must be implemented to distinguish traffic with strict requirements from other traffic. In other words, sensitive traffic must be prioritized. The most common
method for achieving prioritization is by implementing Differentiated Services (DiffServ) in network devices (routers and switches). A network device classifies each packet according to its 5-tuplet (source IP address, destination IP address, source port number, destination port number, and transport protocol type) and services it accordingly. Packets are serviced according to the scheduling mechanisms used for guaranteeing certain level of QoS for each flow. Network resources such as bandwidth are apportioned to a flow according to predetermined policies, which are then applied in the network device.

Our study has been conducted through experiments over a fast Ethernet link in Queen’s University of Belfast connecting the School of Electronic and Electrical Engineering, located in the Ashby building, and the Virtual Engineering Centre (VEC), located in the Northern Ireland Technology Centre (NITC). For the purpose of this study, we have developed a DHVE program based on a peer-to-peer approach. Results on the DHVE program performance over the policy-based network are presented in this paper and compared with our previous work results.

In this paper, we briefly explain the network parameters. A description of the visual and haptic update rate problem is detailed. We also describe the experimental design and the network topology over which the experiments are conducted, including the operation of DiffServ. We present the results and compare them with our previous work. Finally, we conclude by stating some related perspectives.

II. Network Parameters

DHVE requirements differ according to the mode of operation, the adopted architecture, and the task to accomplish. In free motion mode, tracking the cursor position is important but moderate precision is acceptable. When approaching an object, the precision and the system performance should be higher. While in contact with an object, the more a user feels the object mass, more he/she can handle it [5]. Thus, the need for different performance level and consequently different QoS requirements rises. The main idea is to adapt the communication quality with the task requirements. Any parameter adaptation should be a tradeoff between the parameter to adapt and the end-to-end delay [6].

In order to determine the requirements for data transmission in a DHVE system with haptic feedback, any network effects need to be parameterized. Since, network parameters (delay, Jitter, bandwidth and packet loss) are already determined, we will examine their suitability of reflecting the performance requirements for DHVE systems.

When transmitted over networks, packets experience delays caused by queuing, processing, transmission and the propagation over links. The delay varies in proportion to the physical distance and also according to the load in the network. Compared to the other sources of delay, queuing delay represents the major part of waiting in network devices. In addition, queuing delay in network devices is random. Thus packets generated from one source and destined to the same node may experience different delays. This fluctuation of delay is called jitter and depends on the occupancy of the device buffers which in turn is dependent on the network load and the queuing mechanism employed. Jitter effect on teleoperation systems for example, increases with the amplitude and the frequency of the control signal [10]. Furthermore, previous work has shown that the network induced jitter for haptic flows should not exceed 3 ms [15]. Mechanisms such as “Delaying Playout” are deployed to reduce Jitter. As its name indicates, this mechanism delays playing out packets at the receiver for a time equal to $2J$ where $J$ is the maximum experienced jitter. However, “Delaying playout” increases the end-to-end delay; hence there is a trade-off between end-to-end delay and jitter.

DHVE systems are bandwidth-sensitive. The bandwidth required depends on the system resolution, the sampling rate of position and force feedback signals and the overhead of the used transport protocol. DHVE systems usually have multimodalities: audio, video and haptics that all need to be delivered within certain timescales. It should be noted that audio and video media requires much higher bandwidth than haptic information. Human performance is highly dependent on the impression of different modalities arriving on time.

Packet loss is due to shortage in network resources caused by congested network and depends on queuing strategies adopted in network devices. A single packet loss does not destabilize the system but affects the task being performed especially if the packet involves control commands. However, losing a number of consecutive packets in a flow can destabilize a system. It has been shown that the retransmission mechanisms implemented in TCP [13] does not suit DHVE systems [15]. An alternative solution is to use redundant retransmission in UDP protocol [12] associated with duplicate packet detection at the receiver side.

III. Visual and Haptic Update Rate Problem

In DHVE, all scenegraphes must be consistent in order to reinforce the feeling of tele-presence. The visual renderer handles the visual aspect of the DHVE by transforming the geometric information into images for display. On the other hand, the haptic scenegraph is used to calculate forces applied to objects in the DHVE and the reaction forces that should be fed back to the haptic device.

In general, the frame updates of the visual renderer ranges from 10 to 60 Hz depending on the performance of the computational resources (CPU, graphic card, RAM, etc). However, haptic devices such as the PHANToM Desktop perform updates at 1 KHz in order for the user to accurately experience the force feedback. This update rate is greater than the visual renderer rate by two orders of magnitude. This involves different
threads to handle the differing rates of visual and haptic processing. However, as explained above, audio, video, and haptic multimodalities need to be delivered within certain timescales. Therefore, the haptic event loop must be coupled to the rendering loop (Figure 1) [11]. Thus haptic events may proceed at either the same speed or less speed than the visual frame rate. In DHVE, haptic events are not only processed locally but are also sent via network to other participants. When network delay is combined with low update rate, the visual and haptic scenegraphs becomes quickly desynchronized. An alternative solution of the different update rate problem is to employ a multi-threaded architecture [11]. In our experiments, 3D visualisation is not provided for the user. He/she can monitor the local and the remote haptic rate as well as the local and the remote communication rate. Instead a Virtual Network Computing program is used to simulate the display data flow of a DHVE. Providing 3D visual media along with haptics is part of our future work.

Figure 1. Parallel data structure within a DHVE system [11]

IV. Differentiated Services (DiffServ)

Participants in a DHVE exchange command messages and real-time interactive flows that are consuming in terms of network resources and capabilities. Administering traffic by managing allocation of network resources as the flows transit over the network improves DHVE performance and consequently improves users’ feeling of telepresence. However, different types of data are communicated between hosts in a DHVE. Data may be in the form of control commands, haptic, rendering, etc. and exchanges can be frequent or less frequent. This makes DHVE data more sensitive to loss since subsequent packets are unlikely to compensate any lost information. Hence, it is important to discriminate between the different data types (flows) transmitted over the network, and to manage the amount of network resources allocated to each flow in order to guarantee the required service level for each type.

The differentiated services model (DiffServ) [1] is based on the concept of aggregated flows. These flows are classified, identically marked and transported between two adjacent nodes. The transport does not need signalling for resource reservation but Per Hop Behaviour (PHB). DiffServ (currently) defines two per-hop behaviors that effectively represent two service levels (traffic classes):

- **Expedited Forwarding (EF):** Has a single codepoint (DiffServ value). EF minimizes delay and jitter and provides the highest level of aggregate quality of service. Any traffic that exceeds the traffic profile (which is defined by local policy) is discarded [7]. This service is defined as a forwarding treatment applicable to a DiffServ aggregate where its throughput at the output of any node must be greater than or equal to a configurable value. EF corresponds to a critical service similar to a Virtual Leased Line (VLL) in which an instantaneous throughput, instantaneous delay, low jitter and low error rate are guaranteed. EF has strong priority in nodes and yet must be controlled so that the sum of flows coming from different sources and crossing the same node must not exceed the capacity of the output link. Excess traffic is dropped.

- **Assured Forwarding (AF):** Has four classes and three drop-precedences within each class (so a total of twelve codepoints). Excess AF traffic is not delivered with as high probability as the traffic “within profile,” which means it may be demoted but not necessarily dropped [4]. The AF service is a way for a domain operator to provide different levels of forwarding guarantees that are associated to packets coming from a particular source domain. Unlike EF, in AF delay and jitter are not quantified.

Per-hop behaviours are applied by the conditioner to traffic at a network ingress point (network border entry) according to pre-determined policy criteria. The traffic may be marked at this point, and routed according to the marking, then unmarked at the network egress (network border exit). Originating hosts can also apply the DiffServ marking. DiffServ assumes the existence of a service level agreement (SLA) between networks that share a border. The SLA establishes the policy criteria, and defines the traffic profile. Traffic is policed and smoothed at egress points according to the SLA. Any “out of profile” traffic (i.e. above the upper bounds of bandwidth usage stated in the SLA) at an ingress point will have no guarantees (or may incur extra costs such as being dropped/delayed, according to the SLA). When applied, the protocol mechanism that the service uses are bit patterns in the “DS-byte,” which for IPv4 is Type-of-Service (TOS) octet and for IPv6 is the Traffic Class octet.

Best effort PHB, the basic service available on the Internet, must also exist on each node to serve non-sensitive packets. A small amount of the link bandwidth at the output of each node is usually reserved for such packets in order not to totally penalise non-sensitive flows.

Once packets have been classified and processed accordingly, they are queued according to the PHB.
They are then transmitted over the output link by scheduling algorithms that determine the order in which each packet should be served (Figure 2) [3]. In our experiments, we have implemented Priority Queuing (PQ) and weighted Round Robin (WRR) mechanisms. PQ guarantees low delay for high priority packets but it may starve the lower priority flows whenever its throughput is considerable. The WRR mechanism provides different bandwidths for different flows and guarantees inter-flow protection.

---

**Figure 2. Traffic conditioning block: classification, policing, and scheduling**

---

**V. Experiment Design**

The experiments objective is to study the performance of the DHVE whenever policies are applied on network devices and QoS mechanisms are implemented. We focus on the effect of constant background traffic on the haptic flows when background flows coexist in the same network.

Our previous work shows that peer-to-peer approach performs better than the client/server one. Hence, the experiments test the peer-to-peer architecture deployed over a policy-based network. The scenario conducted on the experiments consists of a peer-to-peer distributed haptic virtual environment. The virtual environment, which contains a solid wall, was running on one computer. The PHANToM connected to the other computer was used to explore the remote virtual environment. The PHANToM computer did not have a copy of the virtual environment and instead it only communicated with the remote computer and maintained the servol loop of the PHANToM based on the received data. Only one participant was required for this experiment and she was requested to explore the remote virtual environment in a 7 minutes period.

For the purpose of our study, a program termed Remote Haptic Interaction (RHI) was developed. RHI is based on the User Datagram Protocol (UDP) and provides haptic interaction with remote virtual objects. The PHANTOM [14], haptic device, is used to interact with a virtual object on the remote computer (Figure 3). PHANTOM positional information is sent over the network and processed on the remote computer. Collision detection is performed in order to determine whether the PHANTOM position is in contact with the virtual object. If this is the case, the position of the PHANTOM is updated and prevented from penetrating the object surface then sent back. If there is no collision, the original position is sent back to the PHANTOM. Therefore, the user is able to detect the virtual object and touch its surface.

In the experiments, RHI is run with another program called Virtual Network Computing (VNC) to simulate the display data flow of a DHVE, as it can generate similar continuous data flows over the network. VNC, which is based on TCP protocol, is a remote display application that consists of a server and a viewer program. To allow users to see what happens on the server computer, the screen image on the server is continuously captured and compressed before being sent to the client. Once the data reaches the client, the compressed image is restored and displayed on the client screen. To allow users having remote control over the server in addition to displaying the server screen on the client computer, users can enter their commands, destined for the server computer, through the local keyboard and mouse. The VNC data flow combined with the RHI form the typical traffic in the DHVE. These programs have been used in different network conditions in our previous work [15].

---

**Figure 3. Remote Haptic Interactions**

---

**VI. Network Topology**

Our experiments have been conducted over an experimental fast packet switched network with QoS and policy capabilities (Figure 4) in Queen’s University. It consists of three 100 Mbps Ethernet segments, two 1000Mb/s segments and three PCs: A, B, and C. Computer C is connected to a haptic device (PHANToM). A and B are connected to switch 1, a Business policy Switch 2000, via 100 Mb/S Fast Ethernet link while C is connected to switch 2 also via a
100Mb/s link. Segments 3 and 4, connecting the switches to the router, are 100Mb/s fibre optic links. For simulating network congestion, segment 3 is replaced by a 100Mb/s Ethernet link.

Each computer runs a traffic generation/network analysis tool, called “IP Traffic- Test & Measure” [16]. The computers’ internal clocks are synchronised by using the NTP protocol (Network Time Protocol) [9]. In order to have accurate results in our experiments, we connect to the parallel port of each computer a very accurate clock, called Zclock. This guarantees that each packet sent or received will be time-stamped with a clock variation not higher than 1ms per day. Thus, our worst-case measurement resolution is (± 1 ms). All the experiments are run over the link between A and C while B is used to load the network with background traffic based on UDP protocol.

ZCLOCK is used by the “IP Traffic” software as the following: At the beginning, “IP Traffic” gets the PC time ($T_i$) and the ZCLOCK value ($V_i$). When a packet is sent or received, it is time-stamped by a value equal to $T_i + \delta$, where $\delta = V_i - V_0$. $V_i$ is the time extracted from ZCLOCK at the moment of sending or receiving a packet. It should be noted that the Zclock should be initialised with a time reference, which is in our case the PC’s internal clock.

In order to study whether DiffServ satisfies DHVE flows requirements, the programs RHI and VNC are run for around 7 minutes and their data is exchanged between computers A and C. Data flows are captured and replayed and statistics are collected to calculate packet loss and throughputs of haptic flows. The experiments are repeated for each of the following two cases: (i) Only flows of each program are present on the network, (ii) In addition to these flows, constant background traffic is generated from computer B to C making a bottleneck on segment 3. VNC and RHI flows are classified and queued in the same queue. They are allocated the premium service while the background traffic is served as best effort. A Priority queueing management mechanism is implemented to serve premium flows while a weighted Round Robin mechanism is implemented to serve the other queues. Flows are not metered or shaped at this stage. The throughput of the constant background traffic generated during experiments ranges from 0% to 80% of segment 3’s bandwidth. VNC average throughput sent by the server is 5Mb/s. It should be noted that, during our previous work, where no policy were applied on the network, the VNC server throughput was reduced to 331.422 kb/s for experimental purposes. The size of PHANToM and force feedback packets is 60 bytes.

### VII. Experimental Results

The results collected during the experiments showed no packet loss in haptic flows. This is not the case when haptic flows were run over the network without applying any kind of policy (previous work [15]). At 75% of UDP background traffic, the packet loss rate experienced for haptic flows without applying policies is $2.1 \times 10^{-4}$ (Figure 5).

The new results show that when policies are applied, the sense of touch is affected by the VNC flow. Despite the PHANToM having a local rate of only 1 KHz, the two DHVE hosts (computer A acting as VNC server and computer C to which a PHANToM is connected) failed to maintain the same rate as the PHANToM. Whenever, the VNC throughput is reduced, the computers’ communication rate increases and subsequently, the sense of touch improves. The results also show that as the UDP background traffic increases the PHANToM and force feedback throughputs decrease despite the fact that both haptic flows are very
low compared to VNC and UDP background throughputs. The haptic average throughput while no other flow is present on the network is 461.243 Kb/s. At 50% of background traffic and while running with VNC, the haptic average throughput is 332.47 Kb/s while it decreases to 226.78 Kb/s at 75 % of background traffic load (Figure 6).

Figure 6. Haptic throughput in presence of background traffic

VIII. Conclusion and Future Work

This work leads to a number of conclusions. Firstly, DiffServ improves the DHVE performance but more restricted policies should be applied. Secondly, some more improvements are needed in order to reduce the effect of VNC flows on haptic flows. Finally, haptic flows should be queued in a separate queue from VNC flows. VNC flows should be metered and possibly shaped. These modifications will be our next step as well as providing 3D visual display along with haptics.

In DHVE, accomplishing a task depends heavily on human performance, which in turn depends on the perceived multimodalities of the environment. While the work presented in this paper focused on the QoS issues at the network level, our future goal is to assess the DHVE performance from the user’s point of view. Users will be requested to evaluate the environment during realistic trials. One user per task on the first stage would navigate the remote environment by using the Desktop PHANToM. Evaluation subjects would be the sense of touch perceived by the user, the sense of tele-presence, the ability to control an object in the remote environment, and the degree of the user’s global satisfaction while performing on the DHVE.

In addition to QoS, the next generation Internet has other challenging issues including the provision of high security levels while transmitting data. Security in DHVEs is a very important issue. Malicious users’ do not only cause material and financial damage but can also inflict physical damage, possibly putting lives in danger. Users’ capacity to modify control type data should be rendered difficult. In future work we will study the threats the DHVEs may face while deployed over the Internet.

IX. References


* This work has been achieved during Rima Tfaily’s training at Queen’s University of Belfast.