2004

Performance evaluation of a proxy system for providing immersive audio communication to massively multi-player games

C. D. Nguyen
University of Wollongong

Farzad Safaei
University of Wollongong, farzad@uow.edu.au

P. Boustead
University of Wollongong, boustead@uow.edu.au

Performance evaluation of a proxy system for providing immersive audio communication to massively multi-player games

Abstract

In this paper, a distributed proxy architecture is introduced for the provisioning of an immersive audio communication service to massively multi-player online games. The immersive audio communication service enables each avatar to hear a realistic audio mix of conversations in its hearing range. In our earlier work, peer-to-peer and central server architectures have been proposed for this service. In this paper, a distributed proxy architecture with either using network multicast or unicast between proxies is introduced to address the limitations of the previous architectures. The main focus of this paper is to evaluate the bandwidth cost saving of network multicast in the distributed proxy architecture in different avatar grouping behaviours and distribution of game player scenarios. In addition, the effect of varying the number of proxy servers on communication delays and network bandwidth usages are investigated. We have developed a simulation environment that creates both the physical world (geographic distribution of participants and the Internet topology model) and the virtual world (distribution of avatars based on different avatar aggregation behaviors). Based on the simulation study, we provide recommendations on a cost-effective delivery architecture for this service.

Disciplines

Physical Sciences and Mathematics

Publication Details

This article was originally published as: Nguyen, CD, Safaei, F & Boustead, P, Performance evaluation of a proxy system for providing immersive audio communication to massively multi-player games, IEEE Global Telecommunications Conference Workshops, 29 November-3 December 2004, 192-199. Copyright IEEE 2004.

Performance Evaluation of a Proxy System for Providing Immersive Audio Communication to Massively Multi-Player Games

Cong Duc Nguyen, Farzad Safaei, Paul Boustead
Telecommunications and Information Technology Research Institute
University of Wollongong, Australia
Email {cong, farzad, paul}@titr.uow.edu.au

***Abstract*— In this paper, a *distributed proxy* architecture is introduced for the provisioning of an immersive audio communication service to massively multi-player online games. The immersive audio communication service enables each avatar to hear a realistic audio mix of conversations in its hearing range. In our earlier work, *peer-to-peer* and *central server* architectures have been proposed for this service. In this paper, a distributed proxy architecture with either using network multicast or unicast between proxies is introduced to address limitations of the previous architectures. The main focus of this paper is to evaluate the bandwidth cost saving of network multicast in the distributed proxy architecture in different avatar grouping behaviours and distribution of game player scenarios. In addition, the effect of varying the number of proxy servers on communication delays and network bandwidth usages are investigated. We have developed a simulation environment that creates both the physical world (geographic distribution of participants and the Internet topology model) and the virtual world (distribution of avatars based on different avatar aggregation behaviors). Based on the simulation study, we provide recommendations on a cost-effective delivery architecture for this service.**

I. Introduction

Natural voice communication in networked games is likely to enhance player satisfaction and improve market penetration of many games. Currently, there is a significant drive by game providers to add voice to both console- and PC-based games. Most of these systems provide a peer-to-peer audio communication capability among the members of a team or a single audio channel for the whole game. In the case of massively multiplayer games, inter-person audio communication is still clumsy and unnatural, primarily text-based and augmented (in some cases) by limited voice communications such as a party line.

In earlier work [7], the concept of *immersive* audio communication has been introduced for massively multiplayer games. The immersive audio communication service allows all the avatars to hear a realistic mix of voices of other avatars in their area of interest or hearing range. This audio scene is a personalised mix of voices of other avatars, spatially placed and attenuated according to distance to the listener. Although accuracy is desired, there is variation among the avatars about the importance of different audio signals. Let us define the *interactive zone* as the immediate vicinity of the avatar where active communicative interaction may take place, while the *background zone* as the region outside the interactive zone stretched to the limits of hearing range.

In some cases, the user may only be interested in the interactive zone. The background audio scene, therefore, appears as 'noise' and can either be blocked or simulated by a background chatter of suitable volume. However, there are situations where the multi-person voice communication is either the primary purpose of gathering in the virtual space or at least a very important means for achieving the actual goal. This will be particulary important for the genres of social games and interactive entertainment. Natural multi-person communication is often characterised by the presence of multiple simultaneous conversations among the people gathered in an environment such as a cocktail party, cafe, foyer of a conference, or market place. The ability to pick up interesting conversations in one's vicinity and join these groups or simply be aware of the peripheral discussions is critical to our sense of satisfaction of being in the presence of a crowd. The current style of audio teleconferencing, when a strict protocol of 'one at a time' for conversation has to be followed, may be too restrictive for crowded virtual spaces. We, therefore, believe that in certain situations, realistic presentation of the background audio zone is desirable, if not critical.

Scalable provision of the immersive audio communication service for large virtual environments and over a large-scale infrastructure is extremely demanding on network and processing resources. The audio mix must be suitably composed from the relevant sources (avatars in one's hearing range) and personalised based on the perspective of each individual. A separate mixing computation is therefore needed for each participant. All this should be done in real-time and delivered with tolerable delay and reasonable quality. The application is also highly dynamic as people move (in some cases very frequently) in the virtual environment. This implies changes in the sources of audio signal that needs to be mixed for any given avatar. Creation of each audio scene, therefore, requires *real-time composition of content from a dynamic set of dispersed sources*. This is in contrast to the more familiar web applications where the precomputed content is simply retrieved from a server.

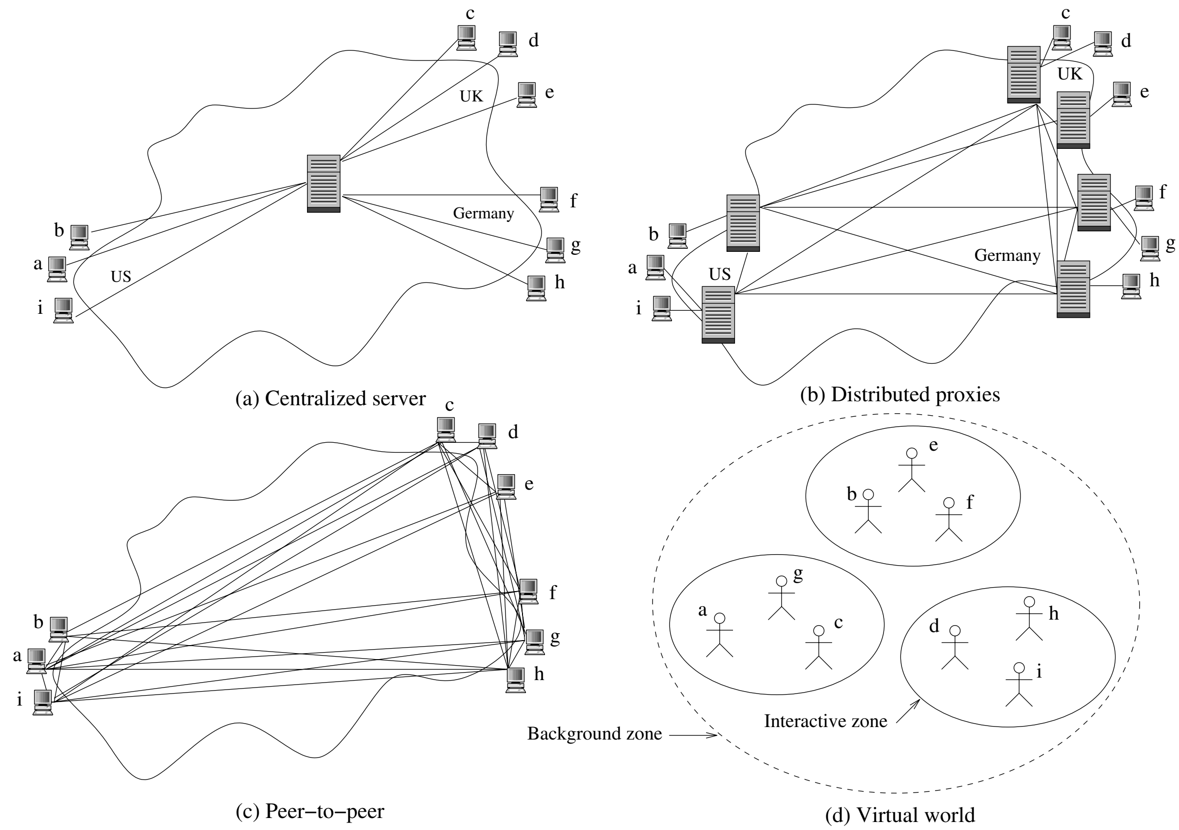


Fig. 1. Server architectures for immersive audio scene creation

In the work in [7], a peer-to-peer architecture and a central server architecture have been introduced for the immersive audio communication service. As shown in Fig. 1a, in the central server architecture, voice is streamed from each of the client devices to the central server. The central server uses these streams in conjunction with avatar position information to create an immersive audio scene from the perspective of every avatar, which is streamed back to each participant. This architecture has the ease of implementation for functionality, such as, security, privacy and billing. However, it is subject to high delay, processing scalability and single point of failure. As indicated in Fig. 1c, in the peer-to-peer architecture, all audio scenes are created on participants' devices. Each participant sends and receives audio signal flows associated with all other avatars in his/her hearing range. This solution will, in many cases, provide a best-case delay for the immersive audio communication environment (assuming shortest path and low congestion routes). However, this architecture faces serious access bandwidth scalability problems, especially on upstream bandwidth, security and privacy concerns due to the need for direct communication between users.

A. Objectives of this paper

In this paper, a distributed proxy architecture is introduced to overcome some limitations of the peer-to-peer and central server architectures. In this architecture, a group of participants

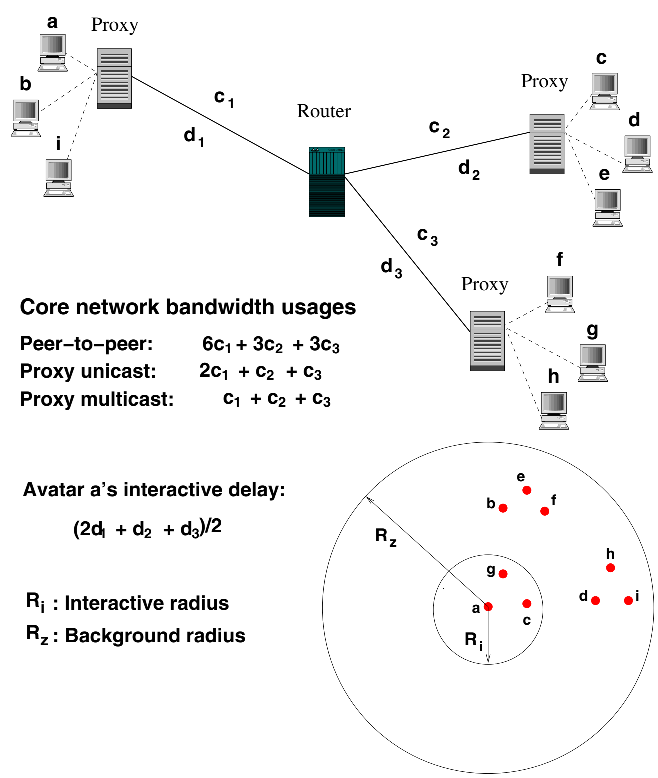


Fig. 2. Network bandwidth usages and interactive delay calculation

0-7803-8798-8/04/$20.00 ©2004 IEEE

in a given geographical location are assigned to a nearby server. This server, which is referred to as a proxy, is responsible for the creation audio scenes for its attached participants. These proxies can send audio flows either using unicast or multicast. The aim of this paper is to evaluate the bandwidth cost saving of network multicast in the distributed proxy architecture with different avatar grouping behaviours and distribution of game player scenarios. In addition, the effect of varying the number of proxy servers on communication delays and network bandwidth usages are investigated.

The rest of this paper is organized as follows: Section II describes the distributed proxy architecture and proposes a simulation model framework that captures a game virtual world and participants in the physical network. Simulation results are presented in Section III. We discuss related work in Section IV and draw conclusions in Section V.

II. Service models

A. Distributed proxy architecture

Fig. 1b shows the basic operation of a distributed proxy architecture, in which, game players in each geographic region are connected to the closest proxy. These proxies form an overlay network, connected in full-mesh. These proxies also communicate with the central game state server to obtain the relative position and state of the virtual environment, such as avatar positions. Based on this information, these proxies manage the audio mixing operation on behalf of their clients. In particular, each proxy is responsible for receiving audio streams, rendering, and mixing these streams to create an audio scene for each of its client. They also forward audio streams to other interested proxies that need these stream for their clients' audio scenes. Similar to the peer-to-peer architecture, the proxies can send audio streams in either unicast or multicast.

The distributed proxy architecture solves the access bandwidth problem of the peer-to-peer architecture since it has a similar access bandwidth requirement to the central server architecture. In addition, security and privacy of players is enhanced. In terms of delay, if a proxy is located at every POP, the distributed proxy architecture has a similar delay performance to the peer-to-peer architecture. As evaluated in [7], this delay performance is significantly better than that of the central server. Since computations are distributed, the proxy system is more scalable than the central server and has no single point of failure. If a proxy is failed, audio streams to/from game players associated with this proxy can be rerouted to the next closest proxy. However, it needs a very dynamic connectivity and interaction management between these proxies.

As an example, Fig. 2 compares the core network bandwidth usages of the peer-to-peer architecture with unicast routing (peer-to-peer), distributed proxy architectures using unicast routing (proxy unicast) and multicast routing (proxy multicast). In the figure, let c_1, c_2 and c_3 denote the bandwidth costs of links, the network bandwidth requirements of different architectures for delivering the audio stream from avatar *a* to all other avatars is shown. It is noted that the peer-to-peer architecture uses most resources, proxy unicast uses less, and proxy multicast uses least. The quantitative comparison of network bandwidth usages of these architectures in different game delivery scenarios will be provided in the simulation study in Section III.

B. Model of a physical network and a virtual world

We model the network topology as a graph $G(V, E)$; where V denotes a set of nodes, E denotes a set of edges; $R \subseteq V$ is a set of Internet Service Provider Points of Presence (ISP POPs). It is expected that potential proxy servers would be located at or near these ISP POPs. Therefore, we choose $S \subseteq R$ as a set of potential proxy servers. Each ISP POP has a number of game players connected to it. Each link has two metrics: a link cost for policy-based shortest path routing, and a link delay representing the propagation delay between two nodes. We do not consider the delays from ISP POPs to game players since these delays are fixed and can not be influenced by the choices of architectures.

Shortest path first (SPF) routing is implemented for unicast routing. This is the current routing protocol in the Internet. Network multicast routing is also implemented by using the SPF algorithm. In this case, each node in the graph is a multicast node. For each avatar, a multicast group is formed by the group of people that need the audio stream from that avatar for their audio scenes. Since end-to-end delay is critical for this application, we do not consider application layer multicast due higher delays in non-direct IP routing paths.

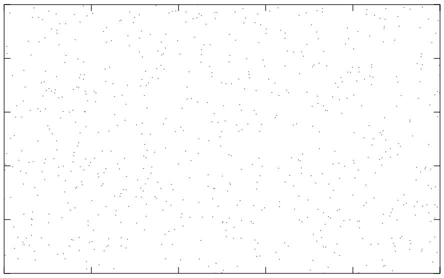
The virtual world is modelled as a square area of certain size, in which, avatars are distributed according to the following distributions.

- Uniform distribution: avatar (x,y) coordinates are set according to uniform random distribution. This results in a uniform spread of avatars in the virtual world.
- Clustered distribution: Cluster centers are randomly placed in the virtual world. At each cluster center, avatars are positioned around of the center according to normal distribution with the mean of 0.

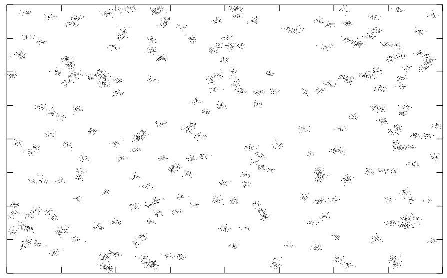
For different types of games, we envisage that there are a range of avatar grouping patterns, referred to as *game grouping behaviors*. The following terminology is proposed to capture this range of game grouping behaviors:

- *Loners* represent isolated avatars which are relatively far away from each other and have low chance of interaction.
- A *clan* represents a medium size group, such as a hunting group, which is very common in online game.
- A *crowd* represents a very large group, such as a stadium or a market place.

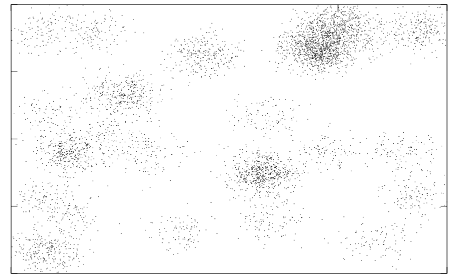
The above avatar distribution is demonstrated in Fig. 3. In Fig. 3a, we model the virtual world of games which mainly have loners as a uniform avatar distribution with low density. Clans are modelled as small clusters, which consist of up to about 30 avatars. Crowds are larger clusters, which consist of about 100 avatars or more. Fig. 3b and Fig. 3c shows a virtual world which consists of 250 clans and 25 crowds, respectively.



(a) Loners



(b) Clans



(c) Crowds

Fig. 3. Avatar distribution in different games

There may be a tendency for game players from close geographic regions to aggregate together due to language, culture and lifestyle preferences. As an example, people in Japan may prefer to gather together in some games for language and cultural reasons. A parameter loosely referred to as "correlation" parameter is introduced to specify how people aggregate in each clan/crowd based on their real world geographic locations. In particular, the correlation parameter states the probability that people in each crowd/clan reside in a particular geographic region. This region is modelled as an ISP POP or a number of close POPs.

To model this correlation, we use a correlation parameter x, $0 \leq x \leq 1$. Each time an avatar is populated in a clan/crowd, a random number r between 0 and 1 is generated. If $r > x$, the avatar is from a random chosen POP. If $r \leq x$, the avatar is from the POPs that are correlated to that clan/crowd. Therefore, the higher the correlation parameter is, the more people in a particular real-world geographical region group together in the same virtual location in the game.

As shown in Fig. 2, each avatar has an interactive zone, denoted as a circle radius R_I, and a background zone, denoted as a circle diameter R_Z ($R_Z > R_I$). The positions of avatars determine a subset of participants who are in the hearing range of each other. This information is used for forwarding audio flows between proxies. The interactive delay of each avatar is defined as the average delay from that avatar to all avatars in its interactive zone. As an example in Fig. 2, let d_1, d_2 and d_3 denotes propagation delays of network links, the interactive delay of avatar a is shown.

III. SIMULATIONS

A. Simulation setup

We use GT-ITM topology generator [12] to model the Internet topology. A transit-stub graph of 5000 nodes, with an average node degree of 3.9, is generated for the simulations. ISP POPs are randomly chosen among the set of nodes, and the number of game players located at these POPs are randomly generated based on a uniform distribution. The topology generator parameters are chosen such that the maximum propagation delay in the shortest path between two nodes is 300ms. Unless otherwise stated, the numbers of POPs and game players are 100 and 5000, respectively; a proxy server is located at each POP and the physical/virtual world correlation is 0. The interactive zone radius R_I and the background zone radius R_Z are 5m and 20m, respectively. The number of crowds/clans in a crowd/clan based virtual world is 50 and 250, respectively. In each architecture, the interactive delay metric is defined as the average interactive delay of all avatars. In the central server architecture, an optimal central server is chosen among a set of potential servers with the objective to minimize the average delay from all game players to that server. The bandwidth requirement is calculated based on the total number of links required by all audio flows. It is noted that when a proxy is located at each POP, the peer-to-peer architecture with multicast routing (peer-to-peer multicast) and the proxy multicast architecture have the same core network bandwidth usages. In this case, simulation results of proxy multicast can be applied for peer-to-peer multicast. In this simulation study, the following issues are investigated:

- Efficiency of network multicast.
- Impact of varying the number of proxies.
- Comparison of network bandwidth usages in different architectures.

B. Efficiency of network multicast in different game grouping behaviors and player distribution scenarios

In the first simulation, we investigate the bandwidth cost saving of multicast when varying virtual/physical world correlation in crowd/clan based games. As shown in Fig. 4, the bandwidth requirements of both peer-to-peer and proxy multicast increase when correlation reduces since more audio flows are needed across the networks. When using multicast, the network bandwidth usage is reduced by more than 50%, at low correlation, and by about 75%, at high correlation. Specifically, in crowd based games, the bandwidth usage ratio of peer-to-peer versus central server increases from about 8 at a correlation of 1 to about 28, (0.7 times the average number of avatars in hearing range which is 40) at no correlation. The bandwidth usage ratio of proxy multicast versus central server also increases from 2 to about 13 at no correlation. The bandwidth requirements in clan-based games are smaller (due to smaller numbers of avatars within hearing range) and follow a similar characteristic to crowd-based games.

In the second experiment, we study the effect of varying

avatar density on the network bandwidth requirements of peer-to-peer, proxy unicast and proxy multicast. It is noted that the avatar density does not affect the network bandwidth usage of a central server. As shown in Fig. 5a, in a loner based virtual world, the bandwidth usages of peer-to-peer and proxy unicast are similar. When the average number of avatars within hearing ranges is varied from 0.5 to 5, the ratio of bandwidth usage of peer-to-peer/proxy unicast versus the central server architecture increases linearly from under 0.5 to about 3. Proxy multicast bandwidth requirement increases almost linearly, and is only slightly lower than that of peer-to-peer. The bandwidth saving of multicast here is not significant due to a large number of small multicast groups. On the other hand, the bandwidth saving of multicast is more significant in a clan/crowd based game due to a larger number of avatars in each multicast group. As indicated in Fig. 5b, when the average number of avatars within hearing range in a clan/crowd based virtual world is varied from 5 to 50, the network bandwidth requirement ratio of peer-to-peer versus central server increases linearly from 3 to nearly 30. However, the bandwidth requirement of proxy multicast is significantly lower, especially, at high avatar densities. Specifically, at high densities, the network bandwidth usages of proxy unicast and proxy multicast are reduced by 25% and more than 50%, respectively, compared with the peer-to-peer architecture. In short, multicast is more suitable for games with crowds and large clans, and not effective for loner based games. In addition, when the number of avatars in hearing range is large, it is more likely that some of avatars are connected to the same proxy, therefore, the bandwidth usage of proxy unicast is lower than that of peer-to-peer.

We investigate the effect of game players distribution on bandwidth requirements of proxy unicast and proxy multicast architectures by varying the number of POPs that game players are connected to in a crowd based game. It is noted that the network bandwidth usage of the peer-to-peer architecture does not depend on the number of POPs. As indicated in Fig. 6b, the bandwidth cost saving of proxy multicast over peer-to-peer is less significant when game players are connected to a large number of POPs. When game players are connected to a small number of POPs, there are more overlaps in each multicast group, therefore, the advantage of multicast is more significant. Specifically, the network bandwidth usage of multicast reduces by more than three times when the number of POPs reduces from 500 to 10. In addition, as shown in Fig. 6a, the network bandwidth usage of proxy unicast also reduces by five times when the number of POPs reduces from 500 to 10. However, in most cases, the network bandwidth usage of proxy unicast architecture is about twice that of proxy multicast.

C. Effect of varying the number of proxies on interactive delays and network bandwidth usages

In Fig. 7, we investigate the effect of varying the number of proxy servers in a range of correlation parameters. The number of proxies is chosen as 10, 50, and 100 (the latter is equal to the number of POPs). While the delay of the

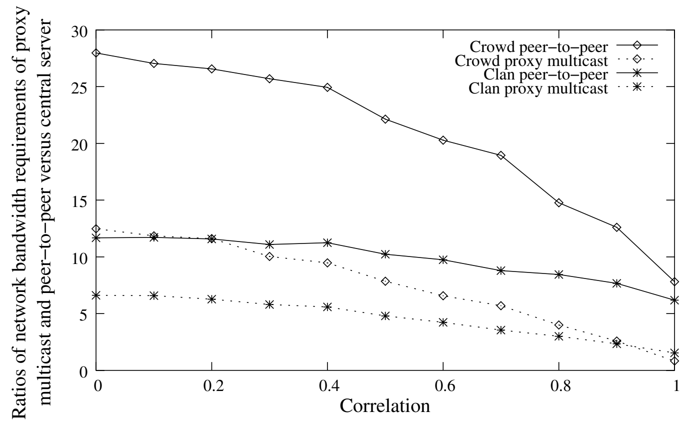


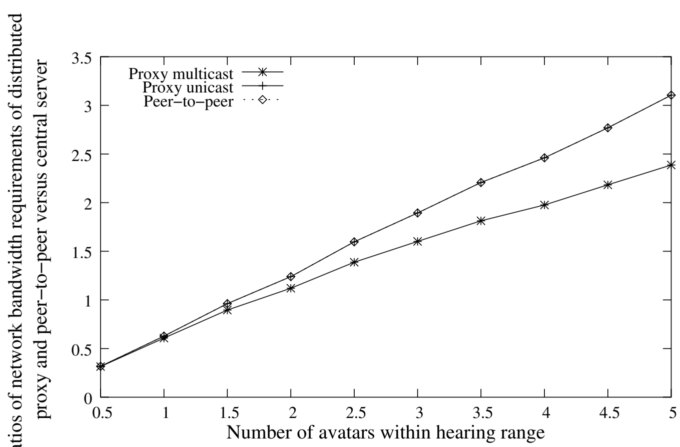
Fig. 4. Effect of varying virtual/physical world correlation on network bandwidth requirements in crowd/clan based games

central server does not depend on the correlation, the delay of the proxy architecture reduces when the correlation increases. When a proxy is located at every POP, the distributed proxy architecture has the smallest delay due to direct routing paths between POPs. When the number of proxies is half of the number of POPs (50 proxies), the interactive delay of the proxy architecture increases but is still significantly lower than that of the central server, especially, at high correlation. However, when the number of proxies is one tenth the number of POPs (10 proxies), this delay increases significantly and is even higher than the interactive delay of the central server when the correlation is under 0.5.

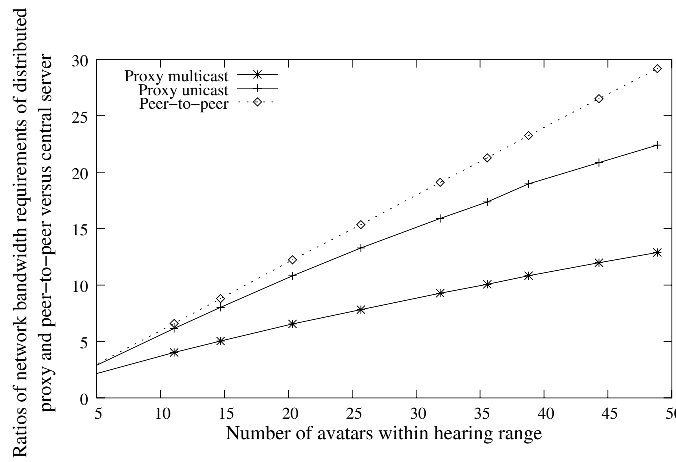
Fig. 8 investigates the interactive delay and network bandwidth requirements of the distributed proxy architecture in a crowd based game when the number of proxies is varied from 10 to 100 and the physical/virtual world correlation is 0.5. As indicated in Fig. 8a, when the number of proxies is reduced, the interactive delay increases significantly from just above half of the delay of the central server to nearly one. This is due to the fact that when the number of proxy servers is reduced, the distances from proxies to some POPs are increased. As a result, longer routing paths are required between two POPs when audio flows are routed through these proxies. On the other hand, as shown in Fig. 8b, when the number of proxies is reduced from 100 to 10, the network bandwidth requirements of proxy unicast and proxy multicast reduce linearly by about 60% and 50%, respectively. In this case, since only a single audio stream is required from a POP to a proxy, reducing number of proxies will reduce the bandwidth usages of proxy unicast and proxy multicast.

D. Summary of results and recommendation

In this study, we have investigated the performance of distributed proxies in a range of avatar grouping behaviours and player distribution scenarios. Our results indicate that if a larger number of proxy servers are available over the Internet, the distributed proxy architecture would be a cost effective

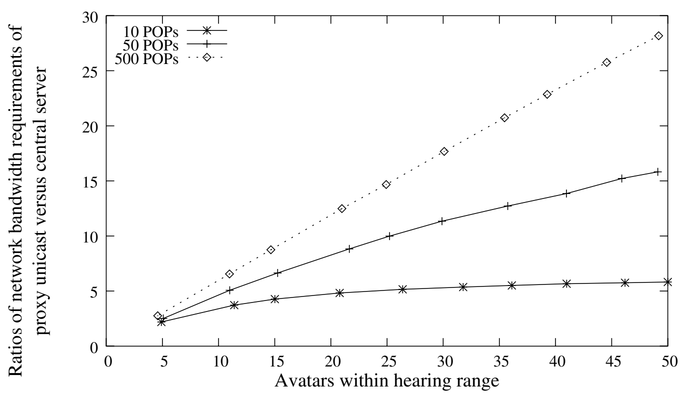


(a) Loners

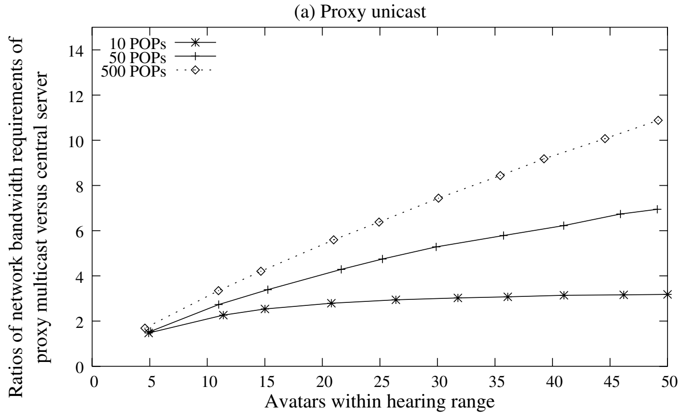


(b) Clans and Crowds

Fig. 5. Effect of varying density on network bandwidth requirements in loner and crowd based games



(a) Proxy unicast



(b) Proxy multicast

Fig. 6. Effect of varying the number of POPs on network bandwidth requirements of proxy multicast and unicast

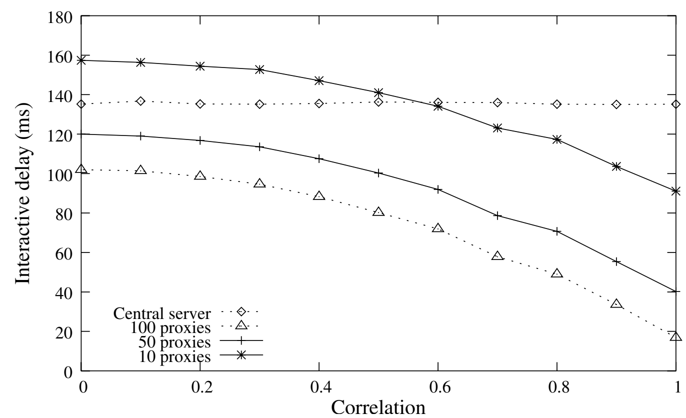
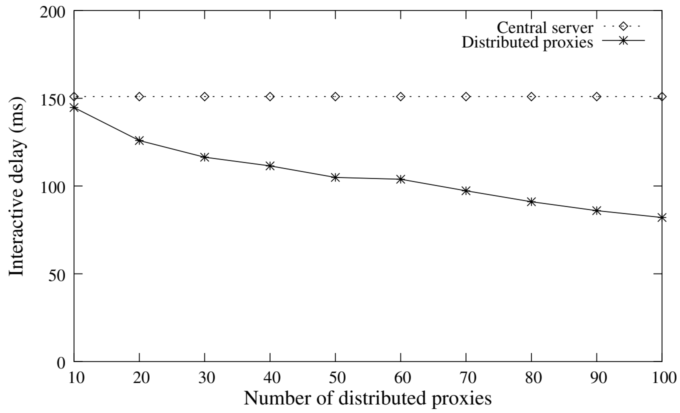


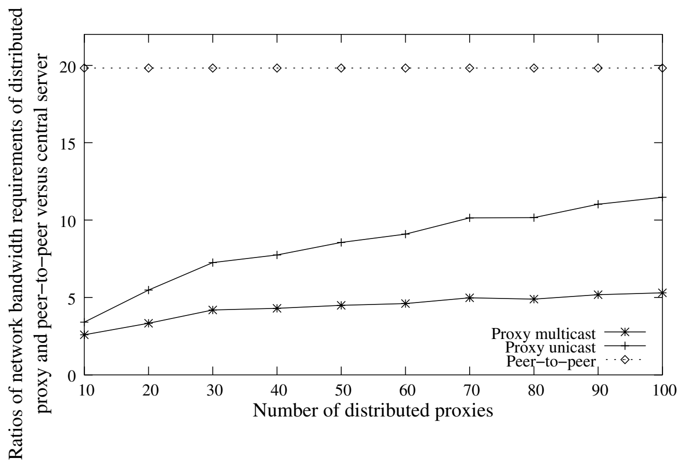
Fig. 7. Effect of varying virtual/physical world correlation on interactive delays

delivery architecture since it reduces the delay of the central server architecture and solves access bandwidth problems of the peer-to-peer architecture. This scenario is possible since ISPs are generally willing to put game servers (or proxy servers) in their networks in order to attract more game players (or Internet service subscribers), and increase their revenues. However, if this is not the case, other architectures should be considered. Therefore, the following recommendation is proposed:

- For loner based games, the peer-to-peer architecture would be favored due to low delays in direct paths between avatars and acceptable levels of access bandwidth, however, security/anonymity issues must be addressed.
- For games consisting of crowds and clans, the distributed proxy architecture is most appropriate due to improvements in interactive delays over the central server, especially when there is a correlation between the virtual world and the physical world. However, if the number of available proxy servers is small, central server may be preferred.
- In many games, where all different player characteristics may be apparent in different parts of a virtual world, a hybrid architecture would be most suitable, in which, parts of the virtual world that mainly consist of loners



(a) Interactive delay



(b) Network bandwidth requirements

Fig. 8. Effect of varying number of proxies on interactive delay and network bandwidth requirements of distributed proxy architectures

would use the peer-to-peer architecture, while crowds or clans would either use the distributed proxies or the central server.

Another recommendation is on the advantages of multicast and proxy unicast over peer-to-peer unicast in different game scenarios. Multicast is effective only when the average number of avatars in hearing ranges is large (i.e crowds). In these scenarios, when game players are connected to a small number of ISP POPs, the bandwidth cost saving of multicast is significant. However, when game players are widely spread, the efficiency of multicast reduces significantly. The proxy unicast architecture also has smaller bandwidth usages than the peer-to-peer architecture, especially when game players are connected to a small number of POPs. Although this bandwidth usage reduction is not as significant as multicast, the proxy unicast is worth being considered since the implementation of network multicast over large-scale infrastructure is a very complex task.

IV. Related work

There are few papers in the literature that have considered audio communication in virtual environment. The work in [1] discusses adding voice to the Mimaze based on multicast. Another work in [2] proposes an architecture called “distributed partial mixing” that effectively provides audio communication in collaborative virtual environment by adapting audio mixing functions to network congestion. In all these work, audio flows are sent in peer-to-peer or peer-to-peer multicast. Our work is different by defining “immersive audio environment” that creates a personalized audio scene for each participant. This requirement has significant effect on the choice of delivery architectures. When the number of avatars within hearing range is large, the peer-to-peer architecture may not be appropriate due to limited access bandwidth at end users.

Distributed proxies solve the access bandwidth problems of the peer-to-peer architecture while still maintain low latencies. A distributed proxy system has been proposed in [6] for game state information processing in networked games and has similar advantages to the architecture presented in this paper, including robustness, minimizing network delays and privacy. To increase server processing capabilities in massively multiplayer games, the work in [3][4] proposes an approach to deploy “booster boxes” at network edges, which act similarly as proxy servers in this paper. Techniques for disseminating avatar grouping information (i.e groups of avatars in hearing ranges) that is used for forwarding audio flows between proxies can be found in [11]. Multicast has been proposed and implemented for a multi-player online games in [5]. A general study on the cost of network multicast compared to unicast are presented in [10]. Our work here provides more specific study of multicast cost in different network game scenarios. Finally, research results on spatial audio rendering techniques [8][9] can be applied to our immersive audio scene creation servers.

V. Conclusions

In this paper, a distributed proxy system is proposed to provide an immersive audio communication service to massively multi-player online games. We have developed a model to capture different avatar aggregation behaviors in the virtual world and participant distribution in the real world. A quantitative simulation study has been carried out to evaluate the performance of the proxy architecture in terms of delays and network bandwidth requirements. The main contribution of this paper is to evaluate the bandwidth cost saving of network multicast in the distributed proxy architectures in different avatar grouping behaviours and distribution of game player scenarios. In addition, the effect of varying the number of proxy servers on communication delays and network bandwidth usages are investigated. Our study will be of benefit to future immersive audio service providers in the design of a cost effective delivery architecture for this service.

VI. Acknowledgments

This work is supported by Co-operative Research Centre for Smart Internet Technology (SITCRC), Australia.

REFERENCES

[1] J. C. Bolot, S. Fosse-Parisis, "Adding voice to distributed games on the Internet," in *Proc. IEEE Infocom*, 1998, pp. 480-487.

[2] M. Radenkovic, C. Greenhalgh, S. Benford, "Deployment issues for multi-user audio support in CVEs," in *ACM Symposium on Virtual Reality Software and Technology*, 2002, pp. 179-185.

[3] D. Bauer, S. Rooney, P. Scotton, "Network infrastructure for massively distributed games," in *Proc. Netgames*, 2002, pp. 36-43, ACM Press.

[4] Sean Rooney, Daniel Bauer, Paolo Scotton, "Efficient programmable middleboxes for scaling large distributed applications," in *Proc. IEEE OpenArch*, 2003, pp. 65-74.

[5] C. Diot, L. Gautier, "A distributed architecture for multiplayer interactive applications on the internet," *IEEE Networks magazine*, July/August 1999, pp. 6-15.

[6] M. Mauve, S. Fischer, J. Widmer, "A generic proxy system for networked computer games," in *Proc. Netgames*, 2002, pp. 25-28, ACM Press.

[7] Cong Nguyen, Farzad Safaei, Don Platt, "On the provision of immersive audio communication to massive multi-player online games," in *Proc. IEEE Symposium on Computers and Communications*, 2004, pp. 1000-1005.

[8] C. Kyriakakis, P. Tsakalides, T. Holman, "Surrounded by sound," in *IEEE Signal processing magazine*, vol. 16, no. 1, pp. 55-66, Jan 1999.

[9] Martin Naef, Oliver Staadt, Markus Gross, "Spatialized audio rendering for immersive virtual environments," in *ACM Symposium on Virtual Reality Software and Technology* 2002, pp. 65-72.

[10] J. C. -I. Chuang, M. A. Sirbu, "Pricing multicast communication: a cost-based approach.," *Telecommunication Systems*, vol. 17, no. 3, pp. 281-297, 2001.

[11] L. Zou, M. Ammar, C. Diot, "An evaluation of grouping techniques for state dissemination in networked multi-user games," in *Proc. MASCOT* 2001, pp. 25-28.

[12] GT-ITM: Georgia Tech internetwork topology models, available at http://www.cc.gatech.edu/fac/Ellen.Zegura/graphs.html