2004

On the provision of immersive audio communication to massively multi-player online games

C. D. Nguyen
University of Wollongong

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

D. Platt
University of Wollongong, dplatt@uow.edu.au

On the provision of immersive audio communication to massively multi-player online games

Abstract

This paper discusses issues in providing immersive audio communication to massively multi-player online games (MMOG). We focus on network and server architectures for creating immersive audio scenes for game clients. We examine advantages and limitations of several architectures, including central server, distributed servers, and peer-to-peer architectures. We focus on a dynamic centralized server architecture which is suitable for the current MMOG in terms of sealability at end users and resource efficiency. We propose algorithms to optimize the performance of a centralized server. Finally, a simulation study is carried out to evaluate the performance of our proposed centralized server architecture and provide some comparisons with others in term of delays and network resource usage.

Disciplines

Physical Sciences and Mathematics

Publication Details

This article was originally published as: Nguyen, CD, Safaei F & Platt, D, On the provision of immersive audio communication to massively multi-player online games, Proceedings ISCC 2004 Ninth International Symposium on Computers and Communications, 28 June-1 July 2004, vol 2, 1000-1005. Copyright IEEE 2004.

On the Provision of Immersive Audio Communication to Massively Multi-player Online Games

Cong Duc Nguyen, Farzad Safaei, Don Platt
Telecommunications and Information Technology Research Institute
University of Wollongong, Australia
Email {dcn01, farzad, don_platt}@uow.edu.au

Abstract

This paper discusses issues in providing immersive audio communication to massively multi-player online games (MMOG). We focus on network and server architectures for creating immersive audio scenes for game clients. We examine advantages and limitations of several architectures, including central server, distributed servers, and peer-to-peer architectures. We focus on a dynamic centralized server architecture which is a suitable for current MMOG in term of scalability at end users and resource efficiency. We propose algorithms to optimize the performance of a centralized server. Finally, a simulation study is carried out to evaluate the performance of our proposed centralized server architecture and provide some comparisons with others in term of delays and network resource usage.

1 Introduction

Massively multi-player online games (MMOG) have become attractive applications over the Internet. A recent report [13] released by Zona Inc. shows that the number of MMOG subscribers in 2002 is about six millions, resulting in a total subscription revenue in the order of a billion dollars, and by 2006, that number should be $2.7 billion. Current MMOG are role-playing games, which simulate a virtual world, where each game player is represented as an avatar.

Although the 3D graphics have been improved significantly in recent years, 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. We believe that these games will be far more attractive if *immersive* audio communication can be provided to game players. An immersive audio service means an avatar can hear voices of the avatars it talks to as well as the background sound of other avatars in its area of interest or hearing range. Creating a personalised mix of voices of other avatars produces the audio scene, spatially placed and attenuated according to distance to the listener. We divide the hearing range into *interactive zone* (a small area that an avatar communicates interactively) and *background zone* (a surrounding area), as shown in Fig. 1d.

In this paper, we focus on server architectures for providing immersive audio communication to MMOG. We introduce and briefly evaluate several architectures, including centralized server, distributed servers, and peer-to-peer architectures. We then propose a dynamic centralized server architecture and design algorithms to optimize the performance of this architecture. For dynamic centralized server architecture, we assume that a game service provider can have access to a number of virtual servers located over the Internet. Depending on the distribution of game participants, the centralized server can be relocated for better performance. A simulation study is carried out to evaluate the performance of these architectures.

The rest of this paper is organized as follows: Section 2 discusses possible network and server architectures for the immersive audio service. In section 3, we discuss objective and procedures for relocation of centralized server. We then describe the service model framework and optimization algorithm for centralized server architecture in Section 4. Simulation results are presented in section 5. We discuss related work in Section 6 and draw conclusion in section 7.

2 Network and server architectures

2.1 Network and server infrastructure for MMOG

Current internet service providers mainly provide bandwidth resources, while storage and computing resources are often provided by other parties such as content distribution networks [14], server farms, and grid computing [15]. In the near future, it is expected that internet service providers may combine provisioning of network and computing resources to support more complex applications that require large amount of computing resources as well as certain Quality of Service (QoS) and flexible functionalities in the underlying networks. An application which would require this infrastructure is immersive audio creation for MMOG. To deploy this application effectively, we envision that service providers may deploy or hire a number of potential processing servers all over the Internet. They may also have certain network supports such as control of routing between these servers or dedicated links (or tunnels) to connect these servers and edge routers if QoS is required. This infrastructure forms a server overlay network targeted for immersive audio scene creation for

0-7803-8623-X/04/$20.00 ©2004 IEEE

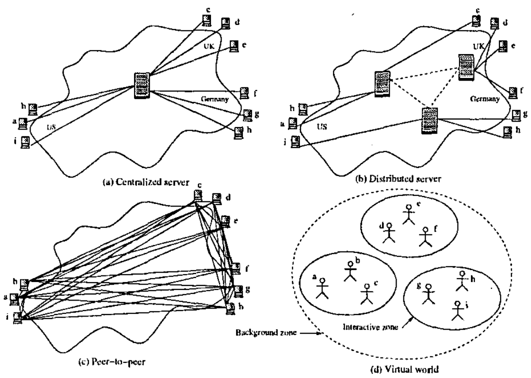


Figure 1. Server architectures for immersive audio scene creation

MMOG. Without having access to virtual resources, the service providers might be required to deploy hardware Internet wide, which would be costly. In the following sections, we examine advantages and trade-off of three architectures including: peer-to-peer, centralized server, distributed server, and dynamic centralized server, as shown in Fig. 1.

2.2 Peer-to-Peer

The main advantages of peer-to-peer are no single point of failure and low latency due to direct path between game participants. However, the deployment of this architecture for audio scene creation faces serious bandwidth inefficiency and scalability problems. If an avatar has a larger number of avatars in its hearing range, a large number of peer-to-peer audio flows are required, resulting in wastes of bandwidth, congestion, and scalability problems at end users, as shown in Fig. 1c. With limited access bandwidth such as modem, or ADSL, assuming reasonable quality audio data rate of 16 kb/s, these scenarios can cause serious scalability problems, especially on upload stream. If multicasting is used at edge nodes, a game client only need to send one audio flow, but still need to receive all audio streams from other clients in its hearing range.

2.3 Centralized server architectures

Using a central server to deliver the immersive audio scene is shown in Fig. 1a. In order to create the audio scene, 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. In this scheme, the core bandwidth usage scales linearly with respect to the number of participants and there are no scalability issues with respect to the access bandwidth. The implementation of security, privacy, billing and other policies is also possible.

The use of a central server for provision of an immersive audio service for a MMOG may introduce higher delays for voice streams since they are routed via one central server. However, it does not require multicasting at each edge routers and any sophisticated functions in the underlying network as peer-to-peer solution. In term of network resource efficiency, a centralized server only requires a single audio flow from each game client, as shown in Fig. 1a. Since all audio flows are processed at a centralized server, summarization of audio flows from a group of avatars to a single source can be reused (e.g an audio scene from avatars a,b, and c, can be mixed into a single source to be used for avatars in other interactive zones, as shown in Fig. 1d). As a result, the centralized server architecture also reduces total computing resources for audio scene creation compared to peer-to-peer architectures.

2.4 Distributed server architectures

There are several ways for distributing computation associated with immersive audio communication. For example, the virtual world can be partitioned to small chunks called locales, and audio streams in one or more locale can be processed by each server [6]. These servers are called distributed locale servers [1] as shown in Fig. 1b. Servers can also be located at edge nodes in a form of distributed proxies [10]. When servers are located in different parts of the Internet, distributed servers may provide better delay performance compared with a centralized server. However, distributed locale servers pose more complexity in control and coordination among servers.

In this paper, we focus on centralized server architecture and propose a dynamic relocation of the server for improving the performance.

3 Relocation of a centralized server

As will be shown later, it may be advantageous to change the location of a centralized server due to changes of game client population in the physical network or perhaps even changes in distribution of avatars in the virtual world. Time zone differences between countries can result in changes in the distribution of game participants in different parts of the network. A study in ref. [5] shows that the distribution of game players in different geographic regions, such as Europe, America, and Asia, has distinct peak patterns in six different 4-hour blocks during a day. In this situation, relocating the central servers would achieve better delay performance.

Naturally, if relocation of a centralized server involves hardware deployment in a different location, the time scale of relocation will be long and the cost will be significant. We assume that the game service provider can have access to a number of virtual servers located over the Internet. These virtual servers would represent some kind of potential server sites, where computation resources can be shifted to. Our primary purpose in relocating server would be improving delay performance. Since relocating a centralized server involves some cost, we only move the server when the improvement in delay performance outweighs the cost of moving the server.

Relocating a central server may take some time, in order of seconds or minutes. We expect this time since a study by Andersen *et. al.* [7] shows that fault detection and rerouting in a wide-area overlay network (RON) take several seconds. The transition time includes rerouting audio flows to a new server as well as moving audio scene creation states to this server. Rerouting can be approximately equal to rerouting in overlay networks, however, the challenge is to move the states from the old server to the new server. We expect to do it seamlessly by moving the states of a small part of the virtual world to the new server step by step. During this handover period, the centralized game state server may inform game clients to direct their audio flows to the new server. These two audio processing servers exchange related information since each server would certainly have clients that communicate with the other.

4 Service model framework

4.1 Model of a physical network and a virtual world

We model the network and server topology as a graph $G(V, E)$; where V denotes a set of vertices, E denotes a set of edges; a set $S \subseteq V$ denotes a set of potential processing server; $R \subseteq V$ is a set of Internet Service Provider Point of Presences (ISP POP). It is noted that R and S are disjoint sets and all nodes can support routing functions. Each ISP POP v_i has n_i game clients connected to. Each link $e_i \subseteq E$ has two metrics: a link cost for policy-based shortest path routing, and a link delay representing the propagation delay between the two nodes. The number of game players located at ISP POPs are randomly generated based on a uniform distribution.

Avatars are populated randomly in the virtual world which is modelled as a square area of certain size. The following avatar distributions can be considered.

- Uniform distribution: avatar (x,y) coordinates are set according to uniform random distribution. It results in uniform spread of avatars in the virtual world.
- Clustered distribution: After some cluster centers are randomly placed, a number of avatars are positioned in dense areas around each of these centers. The rest of avatars are populated sparsely in the entire virtual world area.

In the above distribution, we assume no correlation between avatar positions in the virtual world and game client locations in the physical world. However, in the real game, game players tend to communicate with other players within a particular location due to language, culture, and lifestyle preferences. We model this correlation by partitioning the virtual world into zones, each zone is associated with a particular ISP POP. The correlation parameter states the probability that an avatar is located in a zone that is associated with its ISP POP.

Each avatar has a interactive zone denoted as a circle diameter D_I, and background zone denoted by a circle diameter D_Z ($D_Z > D_I$). When the position of avatar is simulated, for each avatar, a set of avatars in its interactive zone and background zone are determined. This information can be used in accordance with locations of game clients in the physical world to optimize the server performance.

4.2 Optimization procedures for a central server

In this service model, a game application provider requests the service provider to hire a virtual central server for providing the audio service to the game. Depending on their requirements such as game client locations, the service provider will allocate an optimal server for the game. In order to measure delays between participating routers and server nodes, approaches such as IDMap services [9] can be used to estimate distance in term of latency between nodes in the Internet. It results in a virtual overlay network topology that consists of potential processing servers and participating routers. A service provider can use this topology to optimize the performance. We do not consider the delay from ISP POPs to client devices since this delay is unchanged regardless of the location of processing servers.

Given game client locations in the physical network and positions of avatars in the virtual world, the problem is to find an optimal central server for immersive audio scene creation. Since interactive voice communication is subject to delay between game clients and the server, we propose two simple optimization objectives for placing a central server, as would be defined later.

- *Minimize average latency.*
- *Minimize interactive delay.*

In a pre-processing step, we run Shortest Path First (SPF) from each potential server s to all ISP POPs. The distance from a server to an ISP POP node i denoted as $d(s, v_i)$, and the distance from ISP POP node i to j via the server, denoted as $d(v_i, s) + d(s, v_j)$, are pre-determined.

4.2.1 Minimize average latency

The average-latency is defined as the average delay from the central server to game clients. For each potential server s, we compute the latency from the server to each ISP POP, weighted by the number of game clients at that ISP POP, denoted as n_i. We iterate through the set of potential server and choose the server with lowest average latency as follows.

$$\frac{\sum_{i=1}^{N} d(s, v_i) n_i}{\sum_{i=1}^{N} n_i} \tag{1}$$

This procedure has the same complexity as finding shortest path in the graph which is $O(N^2)$ time, where N is the number of nodes. Since computing average-latency is based on the distance from the server to an ISP POP and the number clients

Time	Peak	Mid	Off-peak
1	US	Europe	Asia
2	US	Asia	Europe
3	Asia	US	Europe
4	Asia	Europe	US
5	Europe	Asia	US
6	Europe	US	Asia

Table 1. Game client distribution in a day period

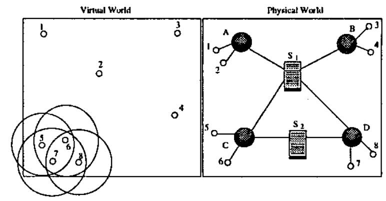


Figure 2. Physical/virtual world model.

at that node's proximity, the central server will be placed near the central mass of game client population in the physical network regardless of positions of avatars in the virtual world.

4.2.2 Minimize interactive delay

This solution requires both game client positions in the physical network as well as avatar positions in the virtual world. The optimization objective is to minimize the interactive delay metrics. The interactive delay of each game clients is defined as the average latency from this game client to other clients in her/his interactive zone as follows.

$$\frac{\sum_{k=1}^{n_{ij}}(d(v_i, s) + d(s, v_{jk}))}{n_{ij}} \tag{2}$$

where n_{ij} is the number of game client, which client j at node i needs to communicate in an interactive zone; v_{jk} is a set of nodes that this client communicate in an interactive zone. The interactive delay metric for a given server is the average interactive delay of game clients, defined as follows.

$$\frac{\sum_{i=1}^{N}\sum_{j=1}^{n_i}\sum_{k=1}^{n_{ij}}(d(v_i, s) + d(s, v_{jk}))}{\sum_{i=1}^{N}\sum_{j=1}^{n_i}\sum_{k=1}^{n_{ij}}} \tag{3}$$

We compute the interactive delay metrics for each server, and choose the server that obtain the lowest interactive delay metric.

The example in Fig. 2 shows the differences between these algorithms. It is noted that the distribution of avatars is not uniform, where some are in a sparse area, others are in a dense area with lots of interactive communication. Since *minimizing average latency* is based only on locations of game clients in the physical world, server S_1 is chosen. However, as only avatars 5, 6, 7, and 8, are in interactive communication, *minimizing interactive delay metric* chooses server S_2, instead.

5 Simulations

In this simulation study, we evaluate the performance of peer-to-peer, centralized server, and dynamic centralized server architectures. Firstly, we investigate the effect of changes in the geographic distribution of game players on a fixed centralized server in terms of network bandwidth cost and delay. We also investigate the effect of avatar distribution in the virtual world on the two proposed optimization algorithms for centralized servers. In addition, we compare the network bandwidth cost and the interactive delay metric of the architectures discussed earlier in a range of physical/virtual world correlation parameters.

We use GT-ITM topology generator to model the Internet topology. Several models are included in this package such as: random graph, hierarchical, and transit-stub. We use transit-stub model since it is a better model of the Internet [12]. Our intention is to model reasonably accurate network topology and game client distributions that reflect the current MMOG delivery scenarios in the Internet. A number of potential servers and ISP POPs are chosen randomly among a set of vertices. The topology generator parameters are chosen that the the maximum propagation delay in the shortest path between two edge nodes is 500ms.

5.1 Effect of changes in geographic locations of game clients

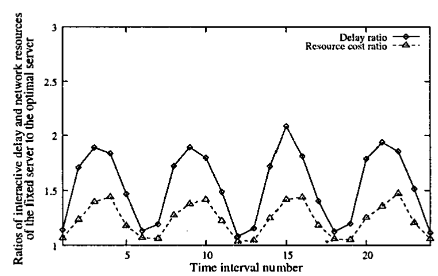


Figure 3. Effect of changes in game client distribution on the interactive delay and network resources

In this section, we simulate geographic changes in game clients population and investigate the effect of this change on performance of a central server over time. We use a transit-stub graph of 600 nodes, comprising of three transit domains, which reflect three main geographic regions: North America, Europe, and Asia. Each domain has on average eight transit nodes, each transit node connects to three stub Autonomous

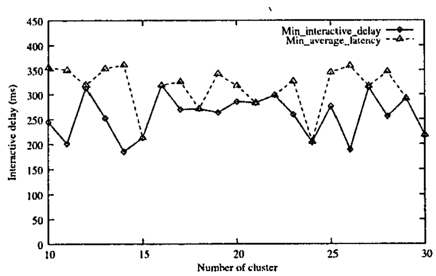
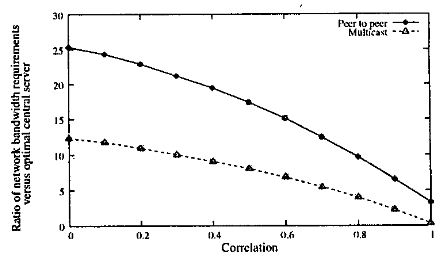
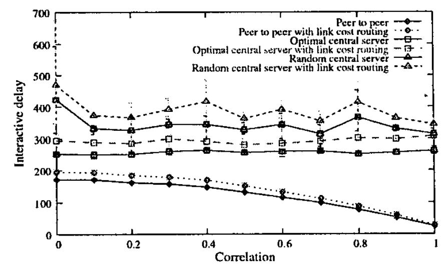


Figure 4. Effect of changes in avatar distribution on interactive delay



(a) Network resources



(b) Interactive delays

Figure 5. Effect of varying physical/virtual world correlation on interactive delay and network resource usage

System (AS), representing the connectivity of different ASs in each region. We randomly place 24 potential servers and 120 ISP POPs at these three regions. The number of game clients at ISP POPs in each region are varied in six four-hour blocks during a day, as shown in Tab. 1. These variations estimate the changes in game client distribution during a day, as described in [5].

In our simulation, the numbers of clients at each ISP POP are uniformly distributed with the mean of 50 for peak, 25 for mid (the period between peak and off peak), and 5 for off-peak. Avatars are uniformly populated in a 650x650m square, the interactive zone diameter D_I is 10m, the background zone diameter D_Z is 40m, and the average number of avatars in the interactive zone is 2.5 (referred as *interactive density*). At each time interval, we run the optimization algorithm to choose a new optimal central server and compare the performance of this server to the fixed server which is the optimal server at the first interval. We calculate the network bandwidth cost by the number of link usage, assuming these audio flows require the same bandwidth. Fig. 3 shows that the ratios of network bandwidth resource cost and interactive delay of a fix server to a dynamic optimal server in each time interval are from 1.1 to over 2. The periodic behaviour of the graph is as a result of the periodic distribution of game clients over time. For example, after six time intervals, the optimal server is close to the fixed server that was optimized for the first interval. The resource cost ratio is lower than the delay ratio since it is only based on the number of links rather than the propagation delays on the links. The same number of links can result in a large difference in delay. Therefore, it is more efficient to shift the fixed centralized server to a new optimal server every few hour interval during a day. A game service provider can monitor client distribution patterns and plan server relocations with specific times during a day.

5.2 Effect of avatars distribution in the virtual world on optimal central servers

In the previous simulation, the two optimization algorithms described earlier choose the same central server. It is due to uniform distribution of avatars with high density. As a result, game clients at different ISP POPs have equal probability of being in interactive communication, and the two algorithms get the same result. *Minimize interactive delay* chooses servers with lower latency when we use cluster distribution, in which some avatars are located in a few cluster points with high interactive density, while others are in sparse area. In Fig. 4, we use the same graph as before but populate avatars in clusters, and the number of clusters increases from 10 to 30. Since game client distribution is not changed, *minimize average latency* choose the same server. The result from Fig. 4 shows that *minimize interactive delay* can sometimes choose better server with much lower latencies. Therefore, depending on types of games and avatar distribution patterns, *minimize interactive delay* can be used to obtain a more optimal server location.

5.3 Effect of changing physical/virtual world correlation on network resources and delay metrics

In the following simulations, the number of avatars are equal to 10,000, the average number of avatars in background zone and interactive zone are 40 and 2.5, respectively. The number of ISP POPs is 25. We implement SPF network multicast routing to get results for multicast architecture.

Figure 5a shows the ratios of network bandwidth resource usage of multicast and peer-to-peer to the optimal central server in a range of correlation parameters. It is noted that these ratios increase as the correlation parameter reduces. This is expected since the lower the correlation is, the more unicast flows are needed across the networks, the more resources are used. Specifically, the cost ratio of peer-to-peer architecture increases from about 4 at a correlation of 1 to about 25 at no correlation. The cost of multicast is very small at a correlation of 1 but increases to nearly half of peer-to-peer cost at no correlation.

In Figure 5b, the interactive delay metric of different architectures are obtained in ten simulation runs. The delay curves show the average values and error bars. The interactive delay is calculated based on delay shortest path routing, and link cost shortest path routing. The latter use routing policy weights which is the current routing protocol for the Internet. As shown from the figure, the interactive delay metric of an architecture using delay shortest path routing (solid lines) is about 10% smaller than that using link cost shortest path routing (broken lines). Peer-to-peer architecture has smallest delay, and this delay increases when the correlation reduces, as expected. Delays of centralized servers do not depend on correlation. Random central server is subject to high interactive delay. Especially, when a random server is far from the optimal server, the interactive delay metric can be more than twice time that of the optimal server.

6 Related work

There are few papers in the literature that have considered audio communication in virtual environment. The work in [3] discusses adding voice to the Mimaze based on multicast. Another work in [4] 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 an "immersive audio viroment" that creates personalized audio scenes for each participant. This requirement has significant effect on the choice of delivery architectures. When the number of avatars in the hearing range is large, the peer-to-peer architecture may not be appropriate due to limited access bandwidth at end users. The work in [5] on geographic distribution of game clients indicates the need for our proposal of dynamic centralized server and provides some input for our simulation. For game state servers, most commercial MMOG development in [15] [13] are based on centralized server. Research work has been done on other architectures including: distributed server [8], proxy [10], and recently a network computation platform for supporting game servers called "booster box" [2]. Finally, research results on spatial audio rendering techniques [11] can be applied to our immersive audio scene creation servers.

7 Conclusion

In this paper, we examine advantages and limitation of a number of network and server architecture for providing immersive audio communication to current MMOG in the Internet. We then propose a dynamic centralized server architecture as a cost-effective delivery in terms of scalability at end users and resource efficiency. We also propose algorithms to optimize the performance of this architecture. Finally, we provide quantitative simulation study to support our proposal.

Acknowledgement

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

References

[1] Paul Boustead, Farzad Safaei, "Network and server infrastructure for the provision of immersive audio communication in distributed virtual environment," SITCRC technical report, available for download at http://www.titr.uow.edu.au/ paul/

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

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

[4] 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.

[5] W. Feng, W. Feng "On the geographic distribution of on-line game servers and players," in *Proc. Netgames*, 2003.

[6] D. B. Barrus, R. C. Waters, D. B. Anderson, "Locales: Supporting large multiuser virtual environments," *Computer Graphics and Applications, IEEE*, vol. 16, no. 6, pp. 50-57, 1996.

[7] D. Andersen, H. Balakrishnan, F. Kaashoek, R. Morris, "Resilient overlay networks," in *Proc. 18th ACM Symposium on Operating Systems Principles*, 2001, pp. 131-145.

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

[9] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, L. Zhang, "IDMaps: a global Internet host distance estimation service," *IEEE Trans. on Networking*, vol. 9, no. 5, pp. 525-540, 2001.

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

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

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

[13] Zona Inc. http://www.zona.net, 2003.

[14] Akamai. http://www.akamai.com, 2003.

[15] Butterfly Grid. http://www.butterfly.net, 2003.