DBF-MLT: NP-Completeness of Specialized Voice Service in Networked Virtual Environment a Peer-to-Peer Approach

SK Alamgir Hossain
Computer Science and Engineering Discipline
Khulna University
Khulna, Bangladesh
Email: alamgir@cseku.ac.bd

Tamanna Tabassum
CFS Limited
Khulna, Bangladesh
Email: tamanna@thecodeandfix.com

***Abstract*—Several researchers have successfully developed realistic technique for specialized voice service in networked virtual enthronement (*NVE*) like Second Life. Specialized voice service in virtual environment enables users to speak to each other naturally and seamlessly. The spatial 3D avatar location and the voice direction in the virtual environment are used to generate realistic sound. Although with this specialized audio or voice streaming in virtual environment the real user can more easily identify who is talking, if several people are taking simultaneously. But this is a challenging task to stream the voice or audio data to the users' machine by considering the user bandwidth limit and the tight latency constraint and this can be formalized as a NP-complete problem. In this paper, we presented the proof of NP-completeness of the specialized audio streaming problem called *DBF-MLT*.**

***Keywords*—NP-Completeness, Networked Virtual Environment, Second Life, Specialized Voice Service**

I. Introduction

Plato introduced the concept of virtual reality in 380 BC, through the famous writing *the cave* [1]. Technology realizes Plato's writing of the *the cave* through networked virtual environment. Networked virtual environment, is a computer simulated environment that reproduces a real world where human control their activities inside the world through their digital representation called *avatar*. One of the most popular and rapidly spreading examples of networked virtual environment is Linden Lab's *Second Life*[1]. In Second Life, once connected the users can view their avatars in the 3D virtual world and they can participate in real-time in task-based games, play animation, communicate with other avatars ([2] [3]) through instant messaging and specialized voice services. The goal of this specialized voice services provided in a virtual world is to enable users to speak to each other naturally and seamlessly. The most important disadvantage of existing audio service in networked virtual environment is that they are not designed the spatial information of a virtual avatar. With specialized voice service the real user can more easily identify who is talking, if several people are taking simultaneously. In our real life we are familiar with this type of activity. If several people are talking from several distance then the sound receive by a person depends on different factors like sound level, distance, frequency, direction etc. Similarly, for specialized voice service the audio stream depend on the avatar virtual position, direction and the user bandwidth limit.

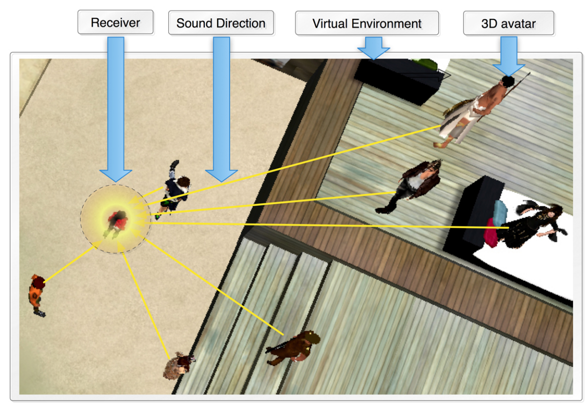

Fig. 1: Relative Distance of Avatars in the Networked Virtual Environment and the physical positions of real users. The specialized voice service depends on the avatar virtual position, direction and the user bandwidth limit.

By utilizing a peer-to-peer based approach which can alleviate the scalability problems caused by the client-server design the audio streams disseminate using application level multicast trees built on overlay networks that take into consideration heterogeneous bandwidth limits of users and tight latency constraints. Because the upload and download bandwidths of each node in a peer-to-peer network are limited and heterogeneous, the task to effectively utilize the network resources when disseminating several separate audio streams while keeping the overlay latency bounded becomes very challenging. Some unsuccessful receivers may not be able to receive the audio stream that they supposed to receive. Similarly some successful receivers may have higher delay

[1]Second Life, http://secondlife.com

or latency because of the non-optimal dissemination path of the audio stream. Hence, the problem aims to:

- Maximize the number of successful receivers of corresponding audio streams and
- Minimize the latency for those successful receivers.

Specialized voice service and audio rendering technique in networked virtual environment is not new, many researcher around the world are working on different services specially realistic audio streaming service in networked virtual environment. A notable work is [4], the author presented the audio rendering technique in virtual environment when the audio sources are moving in different directions. They used the spatial level of details for the 3D audio rendering in complex virtual environment. Based on their studies they argued that by incorporating simple spatial information the audio rendering can be noticeable improve. Another, spatial audio rendering technique is [5], where the authors nicely introduced the audio streaming problem in the networked virtual environment and the optimized technique for rendering a sufficient number of dynamically moving sound sources in the virtual environment. Yamazak et. al [6] also explore the use of spacial audio streaming and the user's experience in multiuser virtual environments. They developed a system called *chatspace* for exploring spatial audio conferencing functionality in multiuser virtual environments.

Although various audio streaming technique have been devised but considering the bandwidth and processing speed peer-to-peer based data delivery is better that the client server based approach [7]. The author presented that the very nature of 3D streaming necessarily places a heavy burden on the streaming server providing the service and peer-to-peer based approach is more useful for distributed networked virtual environment.

Audio streaming performance not only depends on the spatial direction of voice but also depends on the server load. Lui et. al [8] presented how to assign workload to different servers in virtual environment and at the same time, reducing the overall workload and the communication cost. They showed this partitioning problem as a NP-complete problem and discussed several heuristics to improve the service.

Our contribution in this paper is two-fold. First, we provide the problem formulation of an interesting problem in network virtual environment. Second, we presented the NP-completeness proof by reduction of an existing NP-complete problem.

The remainder of this paper is organized as the following. At first, in Section II we formulate the problem statement. Next in section III we presents the NP-completeness proof of *DBF-MLT* problem. At the end we provide conclusion of the paper in Section IV and state some possible future work directions.

II. Problem Formulation

The users (avatars) in a virtual environment can move in the system at will and any two users can communicate with each other when their distance is less than *r*, the radius of a node's area of interest (*AoI*). Every user can speak anytime and at any location. Each user *j* has a limited upload bandwidth u_j^{max}. The upload capacity is measured in units, where one unit is defined as the required bandwidth for a single audio stream and measured in bits/s. We model the system as a digraph *G = (V, E)*. Where, *V* is the set of nodes representing users (avatars) in the system and *E* is the set of edges. An edge *e(i, j)* indicates that node n_i is willing to send (if n_i is a speaker) or forward an audio stream to n_j.

When a node starts speaking, all its neighbour nodes are candidate receivers for the audio stream sent from the speaker. Due to the bandwidth limits of the neighbour nodes and the speaker, only a fraction of the candidate receivers, called successful receivers, may actually be able to receive the stream. All other nodes outside of this range will not be able to hear or forward the audio stream. When a node n_i starts to speak, it will construct a temporal multicast tree T_0^i rooted at n_i. All candidate receivers are members of T_0^i, that is, T_0^i is initially constructed without considering the bandwidth limits of nodes. T_0^i is a Shortest Path Tree (*SPT*) which is a combination of shortest paths from the speaker to all nodes within its *AoI* (see figure 2 for more details).

From figure 2, it is clear that we need to construct a forest that consists of multicast trees rooted at all the speakers (avatar), provided that we have to ensure the bandwidth limits of nodes and latency or delay. We define the problem as constructing a forest of degree-bounded minimum latency trees called here, *DBF-MLT*.

III. NP-Completeness of DBF-MLT

We can formulate our *DBF-MLT* problem as a decision problem. Here we have a set of nodes (avatars) n_1, n_2, ... ,n_m and a distance matrix (d_{ij}) giving the distance between avatar n_i and n_j. Consider the following notation in Table I.

TABLE I: Symbols and notations

Symbol	Meaning
L(j, k)	the latency of the path from node n_j to node n_k
S	the set of speakers
R_i	set of successful receivers of speaker n_i

$$Tour,\ \zeta = \frac{1}{R^S} \sum_{j \in S} \sum_{k \in R_j} L(j,k) \tag{1}$$

$$|R^S| = \sum_{i \in S} |R_i| \tag{2}$$

We wish to construct a tour that minimizes Eq 1. In this equation if the value $|R^S|$ is maximum then ζ will be minimum. We also need to satisfy the upload bandwidth limit of user *x*, i.e. $u_x \leq u_x^m, m \in V$. So our question is we have multi-cast trees rooted at all the speakers and is the tree is optimal with at most the total latency is *B*, i.e. is it satisfy our needs (maximize the number of receiver

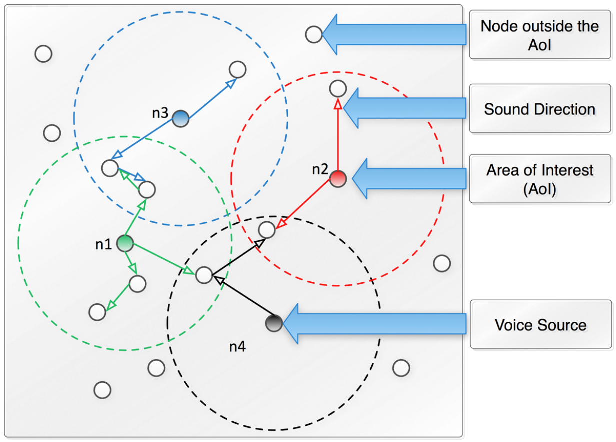

Fig. 2: Audio Streaming Trees within the AoI of Four Speaker Avatars, n1, n2, n3 and n4.

avatar with minimizing the latency by considering the user bandwidth limit).

The process of proving NP-completeness ([9] [10]) of *DBF-MLT* (degree-bounded minimum latency trees with proper bandwidth limit) passes through the following steps :

- Show *DBF-MLT* is in *NP*.
- Select a known *NP-complete* problem such as Freeze-Tag Problem (*FTP*) [11] [12].
- Construct a reduction *f* from *FTP* to *DBF-MLT*.
- Prove the reduction *f* is in polynomial time solvable.
- Finally, prove that for all $I \in D_{DBF-MLT}$, $I \in Y_{FTP}$, iff $f(I) \in Y_{DBF-MLT}$.

Following are the above five steps for proving the NP-Completeness of audio streaming problem in *NVE* that discussed so forth. The first step is to show *DBF-MLT* problem is in *NP*, because it can be decided by a nondeterministic Turing machine [13], in time $O(n^2)$. This machine, on an input containing a representation of the *DBF-MLT* instance, goes on to write an arbitrary sequence of symbols, no longer than its input. When this writing stops, the machine goes back and checks to see whether the string written is the representation of a permutation of the nodes, and, if so, whether the permutation is a traverse with latency B or less. Both tasks can easily be carried out in $O(n^2)$ time. If the string indeed encodes a traverse cheaper than B, the machine accepts; otherwise, it rejects. This machine indeed decides *DBF-MLT*, because it accepts its input if and only if it encodes a "yes" instance of *DBF-MLT*. If the input is a "yes" instance, this means that there is a tour of the nodes with latency B or less. Therefore, there will be a computation of this machine that "guesses" precisely this permutation, checks that its cost is indeed below B, and thus ends up accepting. It does not matter that other computations of the machine may have guessed very costly tours, or plain nonsense strings, and will therefore reject; a single accepting computation is all that is needed for the machine to accept. Conversely, if there is no tours of latency less than B, there is no danger that a computation will accept: All computations will discover that their guess is wrong, and will reject. Thus, the input will be rejected. This will fulfill the first step that *DBF-MLT* problem is in *NP*.

The next step is to choose a known *NP-complete* problem and reduce the problem to our *DBF-MLT* problem. For the reduction here Freeze-Tag Problem (*FTP*) [11] [12] is taken as a known *NP-complete* problem. Freeze-Tag is a very popular children's tag game. A person who contain "it" can tag the other and the tagged player become "frozen". Other player who are not already "frozen" can unfreeze a "frozen" player. The game end when all the players are become "frozen". This tag game can be modeled in case of robot awake problem. Given a complete graph *G' = (V', E')*, the Freeze-Tag Problem can be modeled as a set of robots *V'*, in which initially one robot *S'* awake and others are *R'* asleep. The robots which are awake can be select *B'* sleeping robots and awaken them, provided that the task must be completed in optimal awakening schedule. Once awaken, each new robot is available to assist in rousing another set of *B'* robots. There exists latency between each pair of robots. Thus, a solution to the *FTP* can be described by a wake-up tree *T* which is a directed *B'-ary* tree (means is a rooted tree in which each node has no more than *B* children) rooted at *S'*, spanning all robots *R'*. The objective is to minimize the make span of *T*, i.e., the time *t'* when the last robot awakens.

Now, let the formulation of digraph *G = (V, E)* as shown in Figure 2 in *DBF-MLT* be identical to *G'* in *FTP*, and let the audio streaming rate be one (1) bits/s and all the nodes *V* in *G*, the maximum upload capacity is $B' = u_j^{max} bits/s$, means every node can stream audio up to *B'* childrens. So the resulting topology becomes a *B'-ary* spanning tree *T*, i.e. the resulting tree is a rooted tree in which each node has maximum *B'* children. The maximum end-to-end audio streaming delay (*t*) is the latency of the last peer in *T*. So, the optimal solution (maximize the number of receiver avatar with minimizing the latency by considering the user bandwidth limit) to the *DBF-MLT* problem can also solve the *FTP* problem optimally (i.e. minimize the make span of *T*). In other words, the *FTP* will have a tree *T* in *G'* with minimum delay of min *t'* if and only if the same *T* in *G* provides the minimum-delay spanning tree and the resulting min-max latency of *DBF-MLT* equals min *t'*. So the reduction is complete and clearly the simple reduction can be done in polynomial time.

Now suppose that unsuccessful receiver set U_i (all nodes that have no more available in degrees) is removed from the list of nodes (users). The multicast tree is constructed based on the remaining nodes, and rooted at the sender. In order to achieve low latency we construct a minimum latency tree (*MLT*), constructed by combining all shortest paths from the sender to its neighbours and for maximize the number of successful receivers a heuristic called *MulticastTreeAdjust* used to fulfill our requirements. From this multicast tree which is also a minimum latency tree the audio stream transfer from the sender to the receiver in a minimum delay d. If the user j bandwidth limit is $u_j^{max} bits/s$ and if the audio stream rate is one $bits/s$ then clearly the sender has at most u number of children or it is a *u-ary* tree. If we consider the tree node as a robot then surely the *u-ary* tree is an instance of a *FTP*, where the robot which is awake will select u sleeping robots and awaken them. Once awaken, each new robot is available to assist in rousing another set of u robots. As the tree is a minimum latency tree so wake-up tree *T'* of *FTP* rooted at the sender must be minimize.

So we have proved both sides of equivalence. A *DBF-MLT* of latency less than *B* exists in *G* if and only if a Freeze-Tag Problem (*FTP*) of *B'-ary* tree exists in *G'*. Since we know that Freeze-Tag Problem (*FTP*) is an *NP-complete* problem, *DBF-MLT* is also *NP-complete*. This part proves the last step that for all instance *I*, $I \in Y_{FTP}$, iff $f(I) \in Y_{DBF-MLT}$. Thus all steps of showing *NP-Completeness* of *DBF-MLT* has done. So *DBF-MLT* is in *NP-Complete*.

IV. Conclusion

In this paper, we presented the proof of NP-completeness of specialized audio streaming in networked virtual environment. With specialized audio or voice streaming in virtual environment the real user can more easily identify who is talking, if several people are taking simultaneously. Like the real life the streaming service depends on the virtual avatar position, the direction of sound and the user bandwidth limits. For the proof we used the well-known NP-complete problem called Freeze-Tag problem. Although, it is hard to measure some of the parameters in the virtual world for example the avatar voice direction. In this work we investigate the preliminary characteristic of the audio service; we would like to address the issues more details in future. However, we believe that our proof will remain as a motivation for further research in this area.

References

[1] R. L. Huard, *Plato's Political Philosophy: The Cave*. Algora Publishing, 2007.

[2] S. Hossain, A. Rahman, and A. El Saddik, "Interpersonal haptic communication in second life," in *IEEE International Symposium on Haptic Audio-Visual Environments and Games (HAVE)*, Oct 2010, pp. 1–4.

[3] S. A. Hossain, A. Rahman, and A. El Saddik, "Measurements of multimodal approach to haptic interaction in second life interpersonal communication system," *IEEE Transactions on Instrumentation & Measurement*, vol. 60, no. 11, pp. 3547–3558, Nov 2011.

[4] N. Tsingos, E. Gallo, and G. Drettakis, "Perceptual audio rendering of complex virtual environments," in *ACM SIGGRAPH 2004 Papers*, ser. SIGGRAPH '04. New York, NY, USA: ACM, 2004, pp. 249–258. [Online]. Available: http://doi.acm.org/10.1145/1186562.1015710

[5] M. Naef, O. Staadt, and M. Gross, "Spatialized audio rendering for immersive virtual environments," in *Proceedings of the ACM symposium on Virtual reality software and technology*, ser. VRST '02. New York, NY, USA: ACM, 2002, pp. 65–72. [Online]. Available: http://doi.acm.org/10.1145/585740.585752

[6] Y. Yamazaki and J. Herder, "Exploring spatial audio conferencing functionality in multiuser virtual environments (poster session)," in *Proceedings of the third international conference on Collaborative virtual environments*, ser. CVE '00. New York, NY, USA: ACM, 2000, pp. 207–208. [Online]. Available: http://doi.acm.org/10.1145/351006.351051

[7] S.-Y. Hu, "A case for 3d streaming on peer-to-peer networks," in *Proceedings of the eleventh international conference on 3D web technology*, ser. Web3D '06. New York, NY, USA: ACM, 2006, pp. 57–63. [Online]. Available: http://doi.acm.org/10.1145/1122591.1122599

[8] J. Lui, M. Chan, O. So, and T. Tam, "Balancing workload and communication cost for a distributed virtual environment," in *Advances in Multimedia Information Systems*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1998, vol. 1508, pp. 130–135. [Online]. Available: http://dx.doi.org/10.1007/3-540-49651-3_13

[9] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1979.

[10] M. Sipser, *Introduction to the Theory of Computation*, 1st ed. International Thomson Publishing, 1996.

[11] E. M. Arkin, M. A. Bender, S. P. Fekete, J. S. B. Mitchell, and M. Skutella, "The freeze-tag problem: how to wake up a swarm of robots," in *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, ser. SODA '02. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2002, pp. 568–577. [Online]. Available: http://portal.acm.org/citation.cfm?id=545381.545457

[12] M. Sztainberg, E. Arkin, M. Bender, and J. Mitchell, "Theoretical and experimental analysis of heuristics for the "freeze-tag" robot awakening problem," *Robotics, IEEE Transactions on*, vol. 20, no. 4, pp. 691 – 701, 2004.

[13] Wikipedia, "Non deterministic turing machine, http://en.wikipedia.org/wiki/non-deterministic_turing_machine," Tech. Rep., Last Accessed, March 2011.