http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 11 (1): 85-93, 2012
ISSN 1812-5638 / DOI: 10.3923/itj.2012.85.93

Conference Management and Speech Enhancement for Multiparty Video Conference over the MPLS Networks

C. Prabhu, C. Chellappan and R. Baskaran
Department of Computer Science and Engineering, Anna University,
Chennai-600 025, Tamil Nadu, India

Abstract: This study deals with both conference management and speech enhancement technique for Video Conference type of application. Moreover existing video conference systems do not make use of the IP multicasting technology available in most of the networks. Software based Video Conferencing solutions have not matured to a level where conference management is effortless. Our method makes use of the IP multicasting technology to create multi-party video conferencing system. In this method, the users divided into two categories such as Participants and Spectators. Participants can send both audio and video packets to all the members in the conference. However, Spectators cannot send any audio and video packets; they can only receive the audio and video packets. Using this technique we can reduce the total bandwidth, Server overloaded and latency between the Users. A method is also proposed to separate individual speech from a mixture of speech signals. Cross Spectral Density Matrix method is used to separate convolutive Blind Speech Sources (BSS). Experimental results conducted in real reverberant rooms and live recordings of speech data show the effectiveness of the proposed approach.

Key words: Blind speech separation, conference management, convolutive mixture, multicasting, participants, spectator

INTRODUCTION

Video conferencing tools already exist! This triggers a question "Why yet another tool?" Currently the tools popularly being used are unicast based tool. If a certain data is to be transmitted to multiple receivers, the same data is sent multiple times over the network there by inducing network congestion and delays. On the other hand multicast way of transmitting saves the server from redoing the transmission over and over again! The multicast enabled network (having routers that support multicast also called as m-routers) takes care of routing the packets to all the destinations, also makes sure that only a copy of the data travels through any link if at all the link is used (Pendakaris and Shi, 2001). The m-routers create copies of the data as and when required. When an application runs in a multicast enabled network a single copy of information is send to more than one destination nodes it reduces the transmission overhead on the sender and network (Deering and Cheriton, 1990). However it's important to note that multicast is not much different from a broadcast and therefore is not a controlled environment. On the other hand applications like video conferencing demand stricter access control and privacy. Now organizing a conference in an insecure environment is the challenge we are talking about! A multimedia packet takes large bandwidth, in network. In-order to reduce the bandwidth we are dividing the users into two categories as Participants and Spectators. Here, the Participant has more privileged users compared to the spectator. So that the Participant can send the audio and video packets to the other ends, where as the spectator cannot send the audio/video packets. They can only receive the audio/video packets that mean they can only listen the ongoing conference. Speech separation is carried out using Blind Speech Separation Technique (Solvang *et al*., 2009).

Blind Speech Separation (BSS) is the problem to separate independent sources from given mixed signals where the mixing process is unknown. The classical example is the cocktail party problem, where a number of people are talking simultaneously in a room and one is trying to follow one of the discussions.

Multicasting: In one-to-many or many-to-many communications, often a sender may need to send the same message to many receivers. Examples include audio webcasting, where the same audio stream is sent to many

Corresponding Author: Chakrapani Prabhu, Department of Computer Science and Engineering, Anna University, Chennai-600 025, Tamil Nadu, India

receivers and video conferencing or online gaming where any participant can generate data (audio, video, update messages) that need to be sent to all other participants. Traditional approaches such as using multiple one-to-one unicast connections or using a client-server approach are not scalable and will become bottlenecks and eventually collapse with increasing number of users in the system. Multicasting, on the other hand, allows a sender to send the message only once; the network would then deliver the message to all receivers in the group. The source sends a packet to the network and the network copies the packet at the routers such that each destination will receive a copy of the packet. This approach which is the main component of IP Multicast (Chen *et al*., 2004), will make the most efficient use of network resources compared to one-to-one or client-server approaches, where the packet has to be sent more than once either by the source or by a server.

Blind source extraction: The aim is to restore one source signal from the observation of a set of mixed sources. Among existing mixing models, we consider a Multiple-input/multiple-output (MIMO) context where non-observable source signals are mixed through a multidimensional convolutive channel (Lim *et al*., 2009). The separation is said to be performed blindly when the mixing system is unknown and cannot be identified. Such mixtures are considered in video conferencing and telecommunication applications like cellular communications. This is an issue in several application fields, of which the most famous is the cocktail party problem, where the name comes from the fact that we can hold a conversation at a cocktail party even though other people are speaking at the same time within an enclosed room or environment (Wang *et al*., 2005).

VIDEO CONFERENCING PROCESS IN MULTIPARTY VIDEO CONFERENCING SYSTEM IN MPLS NETWORK

Every user has to first register with the server by giving their user name and password. They use this to participate in the video conferencing, once the user gives the user name and password it is authenticated by the server. After successful authentication the server sends the list of on-going conference to the corresponding user, the user and chooses the conference of his preference then the server allows him to participate in that conference. After that the corresponding user's audio and video packets will be sending to all the participants in the conference (Kim *et al*., 2003). The user is allowed to create the conference of his own in our video conferencing system.

CONFERENCE MANAGEMENT SYSTEM

Conferences should always be held in a controlled environment. In a private conference, one would not like to have uninvited Participant/Spectator. The power of deciding who can be a Participant/Spectator rests with the conference host, who gives the list to the Server. The Server maintains the list of allowed Participants and allowed Spectators. The host of the conference can update the lists as and when required (Wu *et al*., 2007).

Conference life cycle

Hosting conference: As shown in the Fig. 1 to host a conference, a peer (Host) has to send a NEW_CONFERENCE message containing the conference name, allowed participants list and allowed spectators list. Since Host of a conference is also a Participant (Fig. 2) (private key and public key) pair has to be generated and the public key should be sent to the server for distribution to only those who are in the allowed participants or allowed spectators list.

Joining conference as a participant: A peer can join a conference as a Participant only if it is in the list of allowed participants specified by the conference Host. A peer intending to join a conference as a Participant generates a (private key, public key) pair and sends a JOIN_CONFERENCE message containing the conference name and its public key to the Server. The Server checks the list of allowed participants for that conference and if the peer is allowed then it sends the public key of the peer to other Participants and Spectators and also sends the public key of other Participants to the peer (Pendakaris and Shi, 2001). Now the peer can subscribe to multicast AV data (encrypted) of other Participants and decrypt it before rendering using corresponding public keys. Also other Participants and Spectators can subscribe to the multicast AV data (encrypted) of the new Participant and decrypt it before rendering using the public key of the new Participant.

Joining conference as a spectator: A peer can join a conference as a Spectator only if it is in the list of allowed spectators specified by the conference Host. A peer intending to join a conference as a Spectator sends a JOIN_CONFERENCE message containing just the conference name to the Server (Fig. 3). The Server checks

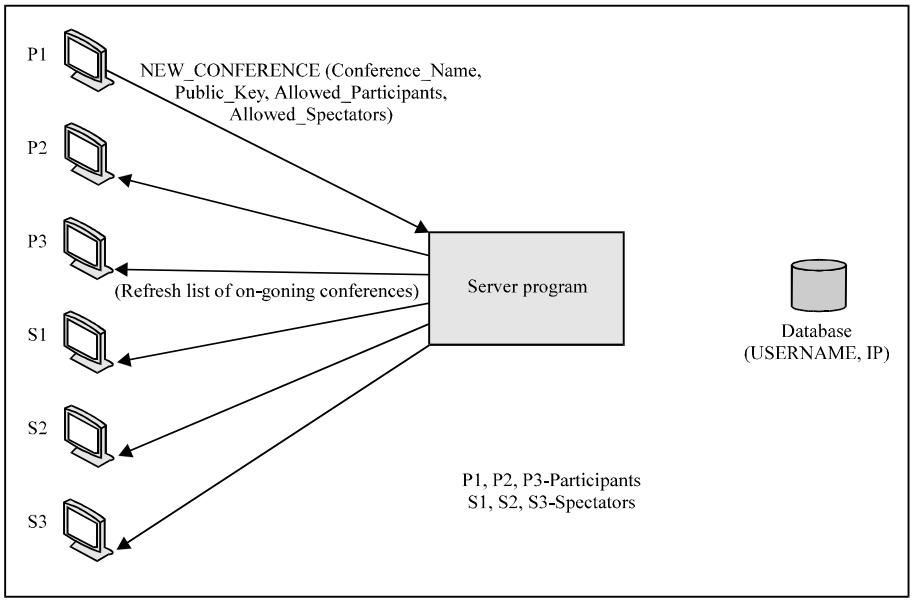

Fig. 1: Hosting conference

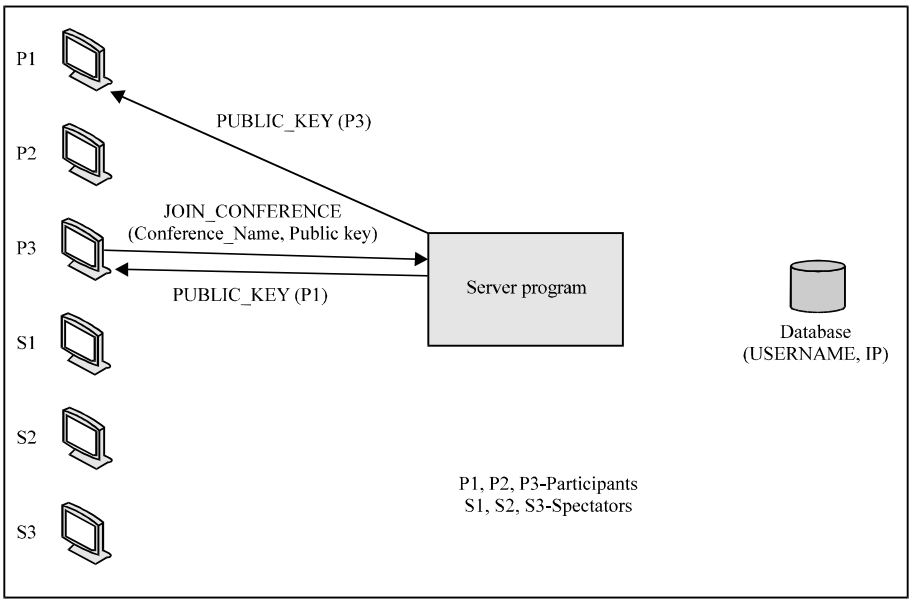

Fig. 2: Joining conference as a participant

the list of allowed spectators for that conference and if the peer is allowed then it sends the public key of all Participants to the Spectator (Pendakaris and Shi, 2001). Now the new Spectator can subscribe to the entire Participants' multicast AV data (encrypted) and decrypt it before rendering using the corresponding public key.

Leaving conference from participants list: The Participant sends a bye message LEAVE_CONFERENCE containing the conference name to the server. The server sends messages to all the spectators and other participants saying this particular Participant has left the conference (Chu *et al*., 2000). The spectators and other participants unsubscribe to the ex-participant's multicast data (Fig. 4).

Leave conference from spectators list: The Spectator can silently leave the conference unless the server wants to

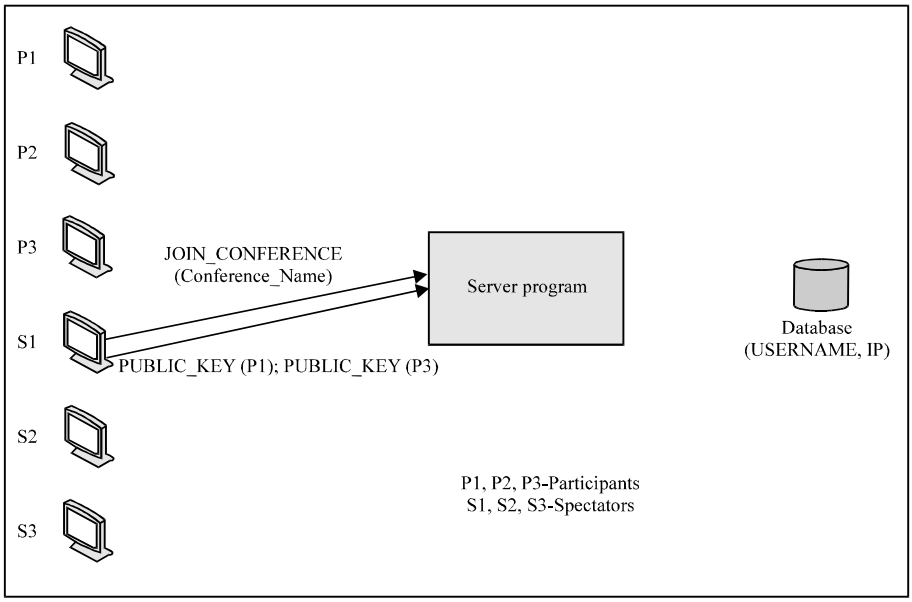

Fig. 3: Joining conference as spectators

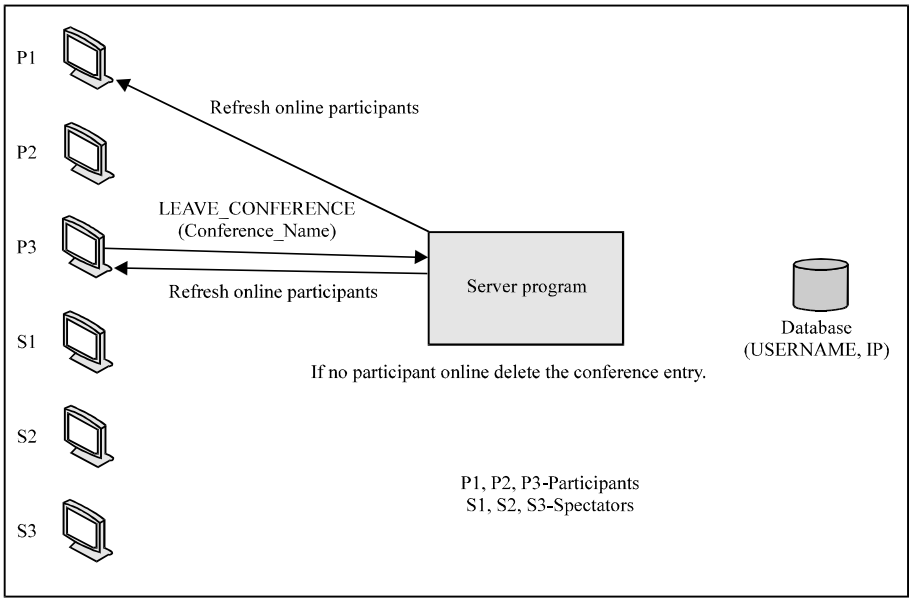

Fig. 4: Leaving conference

maintain a log of conference activities, in which case the Spectator sends a bye message LEAVE_CONFERENCE containing the conference name to the Server and the Server appropriately logs the incident (Chu *et al*., 2000).

PEER LIFE CYCLE

- Every peer has to get authenticated by the Authentication Server using its username and password. The server stores the IP of the peer's machine which can be used by other peers
- On successful login, the system starts audio/video capture and renders only video locally
- It continuously listens over a port for UDP update messages from the server. For e.g. list of online peers, list of on-going conferences
- The peer can interact with other peers or join a conference

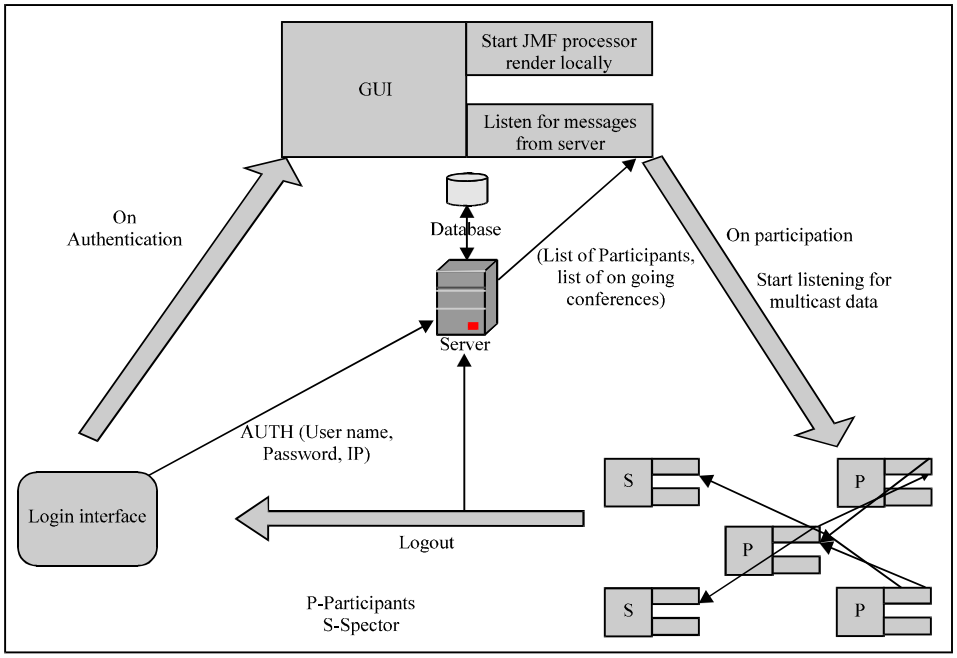

Fig. 5: Life cycle of conference

- On logout, peer should send a "BYE" message to the server and exit gracefully

Life Cycle of Conference has also been described in Fig. 5.

BLIND SPEECH SEPARATION METHOD

In blind speech separation, the objective is to separate multiple sources, mixed through an unknown mixing system (channel), using only the system output data (observed signals) and in particular without using any (or the least amount of) information about the sources of the system. In our paper, we use frequency-domain approach rather than time-domain approach because the frequency-domain algorithms have simpler implementation. We also assume that the noise vector as zero mean (i.e.,) the noise is assumed independent of the sources (Rahbar and Reilly, 2005).

CPSD matrix calculation: Cross power spectral density describes us how the power of the signals is distributed over each frequency. The CPSD matrix is the Fourier transform of cross covariance between two signals.

CPSD matrix is calculated for each epoch. “Epoch” means duration of time for which the source signals can be considered stationary within the epoch but non-stationary between two epochs. In the convolutive Blind speech separation permutation ambiguity problem still exists. We propose a new frequency domain approach to convolutive blind source separation method to resolve permutation problem (Rahbar and Reilly, 2005).

Method to solve permutation problem: Permutation ambiguity is an inherent limitation in independent component analysis which is a bottleneck in frequency-domain methods of convolutive source separation (Fig. 6). In this study we present a method for resolving this permutation ambiguity, where we can group vectors of estimated frequency responses into clusters in such a way that each cluster contains frequency responses associated with the same source. In contrast to existing methods, the proposed method does not require any prior information such as the geometric configuration of microphone arrays or distances between sources and microphones.

Let, X1 (t), X2 (t), X3 (t) be the mixed signals is converted to frequency domain using FFT. The mixed signals in the frequency bins are given to Blind Source Separation module. The output of BSS module is separated sources but still permutation problem exist separated sources. Various methods have been proposed to solve the permutation problem in frequency domain BSS. Liu *et al.* (2007) suggested a method which constrains the length of the filter but this is not suitable for real acoustic environments where the length of the

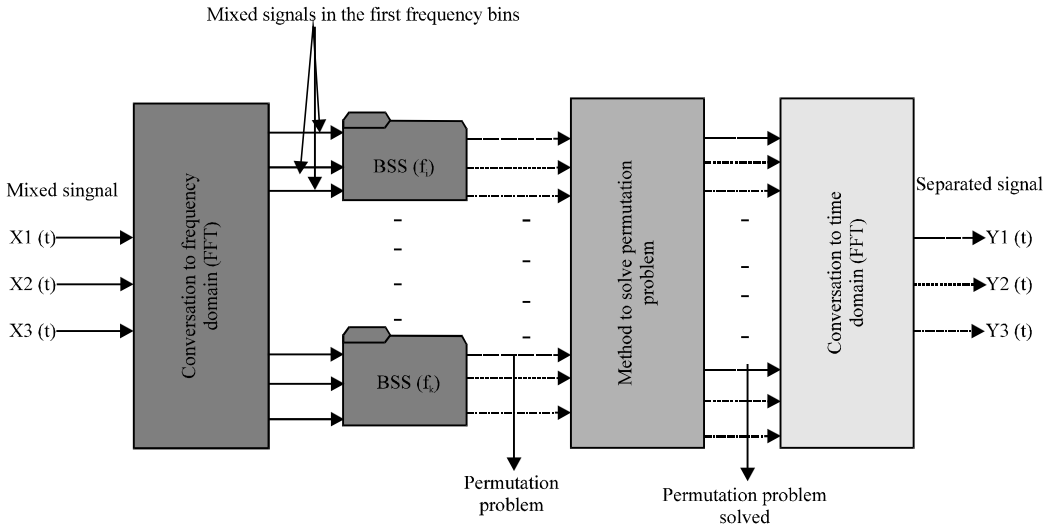

Fig. 6: Flow of frequency domain blind source separation

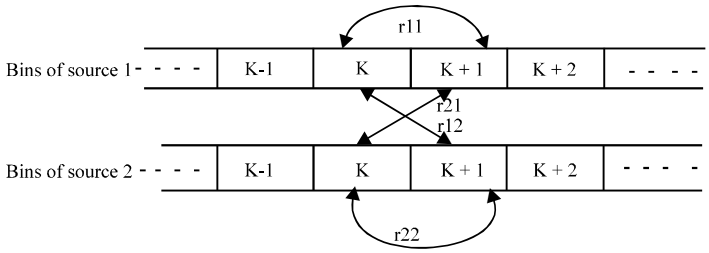

Fig. 7: Correlation between adjacent bins of sources

separation filter is of the order of thousands. Smoothing of the separation matrix is another method. Liu *et al.* (2007) the property that the adjacent bands are highly correlated for speech signals is utilized in to solve the permutation problem. In a correlation based method, the magnitude envelope of the DFT coefficient in each frequency bin is first calculated. Then the correlations, r_{pq}, between the magnitude envelopes of the kth and the (k+1)th bins are calculated as shown in Fig. 7. The permutation between the sources in the (k + 1)th bin are then solved in such a way that the sum of the correlation between the magnitude envelopes $|y_i (k, t)|$, $|y_i (k, t)|$ in the kth and the (k + 1)th bins of the same sources is maximum, i.e., for a two source case shown in Fig. 7, if r11 +r22<r12 +r21, the DFT coefficients in the (k +1)th bins are interchanged between the source; otherwise, the same permutation is kept. Kim and Choi (2006) show that instead of magnitude envelopes of the DFT coefficients, by using the power ratios between the signals, the performance can be improved. The power ratio of the ith separated signal in the kth bin and tth time frame is given in the equation below:

$$P_{r\,\text{ratio}Y_i} = \frac{\| Y_i(k,t) \|^2}{\sum_{j=1}^{Q} \| Y_j(k,t) \|^2} \quad (1)$$

The main disadvantage of the correlation method is its lack of robustness. Since the permutation in one bin is solved based on the permutation in the previous bins, if the method fails in one bin, the remaining bins will follow the same pattern and the method will fail completely. In the reliability of the correlation method is improved with the dyadic permutation sorting algorithm, as explained by Kim and Choi (2006). For the purpose of explanation, assume that there are only 8 frequency bins, k_0, k_1,…, k_7. To solve the permutation problem (Fig. 8), in the first level (Level 0), solve the permutation between the bins k_0 and k_1, k_2 and k_3, k_4 and k_5 and finally between k_6

and k_7. After solving permutation in Level 0, in Level 1, the correlation is calculated between the mean magnitude envelopes:

$$\frac{1}{2}(|Y(k_0,t)|+|Y(k_1,t)|)$$

and:

$$\frac{1}{2}(|Y(k_2,t)|+|Y(k_2,t)|)$$

Using the calculated correlations, solve the permutation between the groups $\{k_0, k_1\}$ and $\{k_2, k_3\}$ such that the permutation for the bins k_0, k_1, k_2 and k_3 are the same. Similarly, solve the permutation between the bins k_4, k_5, k_6 and k_7. In Level 2, correlation between the mean of the envelopes of the data of the bins k_0, k_1, k_2 and k_3 with that of k_4, k_5, k_6 and k_7 is calculated and accordingly the permutation is solved for the bins k_0, k_1,···, k_7. The same approach can be extended for K bins with total levels equal to $\log_2$ (K). Direction of Arrival (DOA) is another popular method, where the directivity pattern formed by the separation matrix in the frequency domain is used to estimate the directions of the sources and hence the permutation problem is solved (Mirtavoosi *et al*., 2009). However, the DOA method cannot solve the permutation problem in all the DFT bins especially for lower frequency bins combined the DOA method with correlation method to form a robust method for solving the permutation problem.

Which is suitable for all the cases except when the sources are very close to each other or collinear. Another approach is the combination of these two approaches, namely, time-frequency algorithm (Mirtavoosi *et al*., 2009). The algorithms defined in the time domain typically will not suffer any permutation problem even if they are implemented in the frequency domain and frequency domain implementation will improve the speed of computation. It is also shown that filter bank based BSS will improve the performance and solving the permutation problem between the filter banks will also be easy when compared to the frequency domain methods (Mirtavoosi *et al*., 2009). The convergence of the frequency domain algorithm greatly depends on the initial values of the unmixing matrices.

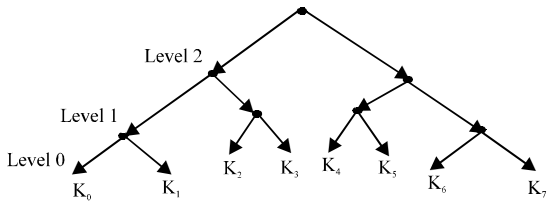

Fig. 8: Sorting to solve permutation problem

EXPERIMENTAL OUTPUT

We have implemented a software prototype of JMF-based video conferencing application based on Peer to Peer Application Level Multicast packet Protocols and demonstrating the algorithm capability. 4 PCs of Pentium IV 1.8 Ghz with 512 MB memory were setup in a fully-connected mesh topology as shown in Fig. 5.

Speech separation process: Recordings are conducted on a closed room of 5.2×3.4×2.9 m size in a less noisy environment. Experiments are conducted with various voice samples. For all experiment, sampling frequency is taken as 10 Khz and epoch as 5000 data samples (i.e.,) half second. First we gave different types of voices to the system as input. All inputs were of 8 sec. For example: One Male and one Female.

Individual source separation process: Figure 9 shows the speech of a male saying 'One Two Three …' for eight seconds in the Sensor 1.

Figure 10 shows the speech of a female saying 'One Two Three …' for eight seconds in the same Sensor 2.

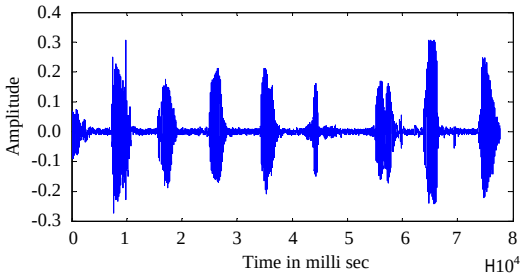

Fig. 9: Source 1

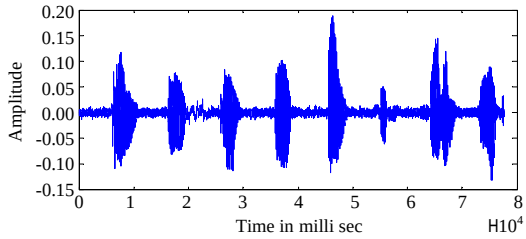

Fig. 10: Source 2

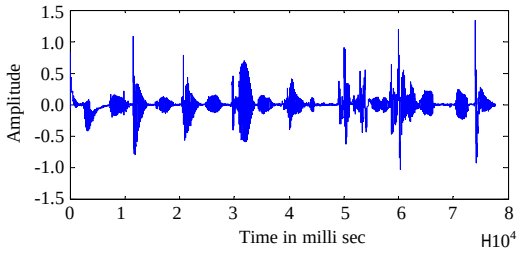

Fig. 11: Mixed source 1

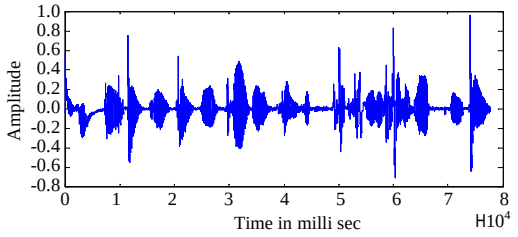

Fig. 12: Mixed source 2

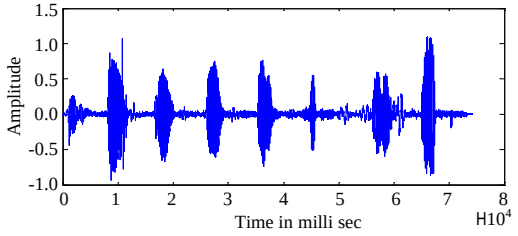

Fig. 13: Separated source 1

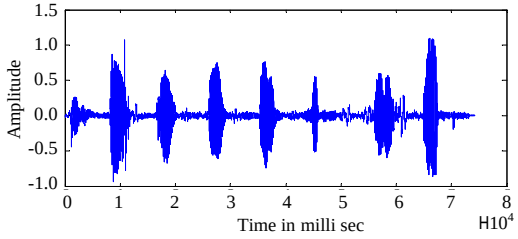

Fig. 14: Separated source 2

Mixed sources: Figure 11 and 12 show the mixed input of Source 1 and Source 2 in Sensor 1 and Sensor 2, respectively.

Separated sources: Figure 13 shows the output speech of a male saying 'One Two Three …' for eight seconds after applying the blind speech separation method to the Sensor output.

Figure 14 shows the output speech of a female saying 'One Two Three …' for eight seconds after applying the Blind Speech Separation method to the Sensor output.

CONCLUSION

In this study, we have discussed a new approach to reduced network bandwidth by using Participant and Spectator concept in conference management to avoided server overload and latency between the peers. We also proposed a new technique to separate the mixture of speech sources by using blind source separation method. This application also provides Text Chat, Online presentation and demos facilities to the participants.

ACKNOWLEDGMENT

This study is supported by the NTRO, Government of India. NTRO provides the fund for collaborative project "Smart and Secure Environment" and this study is modeled for this project. Authors would like to thank the project coordinators and the NTRO members.

REFERENCES

Chen, L., C. Luo, J. Li and S. Li, 2004. Digiparty-a decentralized multi-party Video conferencing system. IEEE Int. Conf. Multimedia Expo, 3: 1839-1842.

Chu, Y., S.G. Rao and H. Zhang, 2000. A case of end system multicast. Proceedings of ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (MMCS'00). ACM New York, NY., USA., pp: 1-12.

Deering, S. and D.R. Cheriton, 1990. Multicast routing in datagram internetworks and extended LANs. ACM Trans. Comput. Syst., 8: 85-110.

Kim, M.S., S.S. Lam and D.Y. Lee, 2003. Optimal distribution tree for network streaming media. Proceeding of the IEEE International Conference on Distributed Computing Systems, May 19-22, The University of Texas, Austin, pp: 116-116.

Kim, M. and S. Choi, 2006. ICA-Based clustering for resolving permutation ambiguity in frequency-domain convolutive source separation. Proceeding of the IEEE 18th International Conference on Pattern Recognition, Aug. 20-24, Pohang University of Science and Technology, Korea, pp: 950-954.

Lim, B.P., E.K. Karrupiah, E.S. Lin, T.K. Phan, N. Thoai, E. Muramoto and P.Y. Tan, 2009. Bandwidth fair application layer multicast for multi-party video conference application. Proceeding of the IEEE Consumer Communications and Networking Conference, Jan. 10-13, Panasonic Kuala Lumpur Lab., Kuala Lumpur, pp: 1-5.

Liu, Y.L., S. Xu and M.Q. Li, 2007. A Second-Order feature window method for blind separation of speech signals corrupted by color noise. Proceeding of the Machine Learning and Cybernetics, International Conference, Aug. 19-22, Hong Kong, pp: 3454-3458.

Mirtavoosi, S.A., M.R.A. Sahaf and V. Abutalebi, 2009. Combined noise reduction and joint diagonalization techniques for blind source separation. Proceeding of the 2nd International Conference on Computer and Electrical Engineering, Dec. 28-30, Iran, pp: 541-544.

Pendakaris, D. and S. Shi, 2001. ALMI: An application level multicast infrastructure. Proceedings of the 3rd USENIX Symposium on Internet Technologies and Systems, March 26-28, San Francisco, pp: 49-60.

Rahbar, K. and J.P. Reilly, 2005. A frequency domain method for blind source separation of convolutive audio sources. IEEE Trans. Speech Audio Process., 13: 832-844.

Solvang, H.K, Y. Nagahara, S. Araki, H. Sawada and S. Makino, 2009. Frequency-Domain pearson distribution approach for independent component analysis (fd-pearson-ica) in blind source separation. Proc. IEEE Trans. Audio Speech Lang., 17: 639-649.

Wang, W., S. Sanei and J.A. Chambers, 2005. Penalty function-based joint diagonalization approach for convolutive blind separation of nonstationary sources. IEEE Trans. Signal Process, 53: 1654-1669.

Wu, X., K. Dhara and V. Krishnaswamy, 2007. Enhancing application-layer multicast for p2p conferencing. Proceeding of the 4th IEEE Consumer Communications and Networking Conference, January 2007, Las Vegas, NV., USA., pp: 986-990.