Connection optimisation for two types of videoconference

Y.-W. Leung
T.-S. Yum

Indexing terms: Multipoint videoconferencing, Connection-path optimisation

Abstract: The connection optimisation problem for two types of multipoint videoconferences is formulated. The first type is called a selectable media conference. In this type each conferee can choose to receive a particular video composition. In the second type, called a common media conference, only one composite video is generated for all conferees in this type of conference. Algorithms that use a combination of look-up table and online processing are designed for computation of the optimal paths for connecting the conference sites. The blocking probabilities for these two types of videoconference in fully connected networks are derived and compared. The sensitivity of network throughput to conference-size distribution is also studied.

1 Introduction

Multipoint videoconference service allows users to conduct meetings without leaving offices. Sitting in front of their workstations or PCs, users can see each other via real-time videos and talk and listen via real-time audios. Public videoconference services have been provided in a number of countries for more than a decade [1]. However, this service is still unpopular because of its inconvenience in the sense that a user has to go to a public studio and that the meeting time most often has to be prescheduled. With the widespread laying of optical fibres which eventually might reach every office and home, the advances in image compression technology and the fall in hardware costs, videoconference services will become increasingly affordable in the near future.

Studies on the systems and human factor aspects of videoconferencing include the work by Watanabe *et al.* [2] who classified and analysed teleconference systems and reviewed the technologies that are vital to teleconferencing. Several types of multipoint videoconference systems were proposed in [3–7]. To maintain the continuous presence of all conferees, the possible terminal configurations fall into two main categories: those employing multiple monitors and those using a single monitor screen with segmented windows. On the network side, Ferguson and Mason [8] studied the design of optimal network topologies for multipoint videoconferences. Liao and Roberts [1] proposed a traffic model for a videoconference service where the users make reservation in advance and the conferences are between two studios.

The effectiveness of the videoconference service depends on how much it can emulate a face-to-face meeting. Audio should be unconstrainted because it occupies relatively small transmission bandwidth. On the other hand, transmission of good-quality compressed video requires much larger bandwidth (e.g. 1.5Mbit/s or more). The finding of good connection paths for a conference therefore is one of the most important problems in the design of a videoconference system. In this paper we describe two types of videoconference, namely with selectable media and with common media, and propose algorithms for determining the optimal connection paths for both types. The blocking probabilities of these two types of videoconferences in fully connected networks are also derived.

2 Videoconference services

2.1 The service

Videoconference calls can be classified into prescheduled calls and on-demand calls. Prescheduled calls require users to make reservations in advance. Access conflicts can be resolved in advance and efficient scheduling of channel usage can be done. On-demand calls, on the other hand, require immediate seizure of the necessary communication resources in the same way that telephone calls do. The determination of the connection paths must also be done in real time.

There are various ways to present video images of conferees at different locations. A promising method is to use one monitor with segmented windows [4, 8, 9] such that each window displays the video captured at each conference site. Using this method, each conference site only needs one monitor. The network provides a conference bridge for each conference. The conference bridge collects video and audio signals from each conference site, mixes the audio signals synthesises the composite video signals and distributes the resulting video and audio signals to the conference sites. By providing a conference bridge for each conference, each conference site needs only one incoming channel to receive one composite video showing the video images of all the conferees and it need not perform video and audio processing. This results in lower communication as well as processing costs.

IEE Proceedings online no. 19960344

Paper first received 15th May 1995 and in revised form 15th January 1996

Y.-W. Leung is with the Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

T.-S. Yum is with the Department of Information Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong

We consider two types of conferences called selectable media conferences and common media conferences. For a selectable media conference, each conferee can choose to receive a particular video composition. Fig. 1*a* shows an example in which conferees A and C of the same conference choose to receive different video compositions. In a selectable media conference, each conference site requires a dedicated incoming channel and the conference bridge must have sufficient processing power to synthesise one composite video for each conference site.

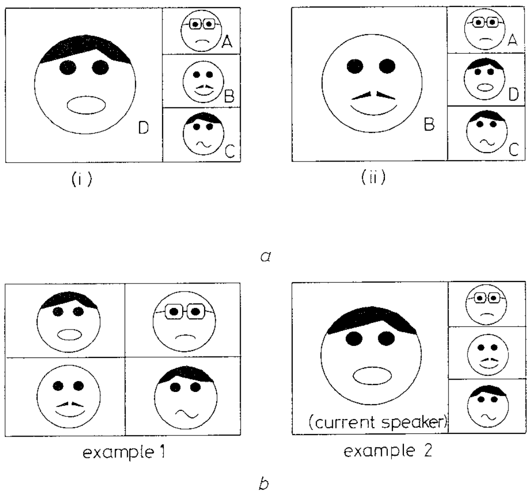

Fig. 1 *Examples of selectable media and common media conferences*
a Selectable media conference: individual conferee can change the layouts at will
(i) Composite video for conferee A who is intending to see conferee D
(ii) Composite video for conferee C who chooses to see conferee B
b Common media conference: same composite video is received by all conferees: two examples of window layout

For a common media conference, all conferees receive the same composite video. Fig. 1*b* shows two examples of common media conferences. In a common media conference, the conference bridge can multicast one composite video to all the conference sites and hence requires less bandwidth than the selectable media conference. Moreover, the conference bridge only needs to synthesise one composite video for each conference (i.e. smaller bridge cost).

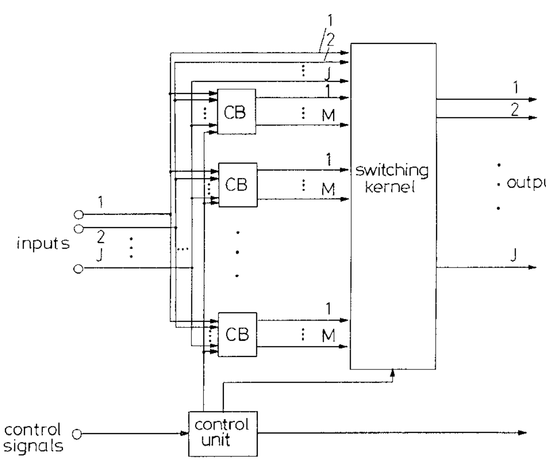

Fig. 2 *Internal structure of processing node*
CB: conference bridge
M: Max. number of conference sites of each conference
J: number of input/output channels of a node

2.2 The networks

The videoconference service is provided by a packet switched network. Let there be N nodes in the network and let a subset of these nodes be called processing nodes. Each processing node is equipped with a number of conference bridges. Fig. 2 shows the internal structure of a processing node. The conference bridge collects the video and voice packets from the conference sites, processes them, and distributes the processed packets to the conference sites via a switching kernel. For a common media conference, the processed packets are multicast to all the conference sites. Therefore the switching kernel must support point-to-multipoint transmission (e.g. embed a copy network in a Banyan network [10] or use the shared-media video switch proposed in [11].

The videos from all the conference sites are compressed, packetised and statistically multiplexed onto the network links. Each network link can be characterised in terms of the number of logical video channels by known statistical techniques [12, 13], which need as inputs the calibrated video quality in terms of packet loss statistics. For example, if we apply the video coding algorithm recommended in CCITT H.261 to compress a typical head-and-shoulder video frame sequence, the resulting source bit rate can be characterised by a eight-state Markov chain with mean 1.57Mbit/s and standard deviation 0.41Mbit/s [23]. If the data rate of each link is 100Mbit/s, then a link can support 56 logical video channels to ensure a packet loss probability 0.0001.

The conference sites are connected to the conference bridge through certain network links. This connection can be either fixed or changed dynamically (e.g. for load balancing).

3 Connection optimisation

Each conference site forwards its video/voice packets to the conference bridge via a connection path. Let the set of all connection paths from the conference sites to the conference bridge be called an inbound connection. Conferees receive composite videos through connection paths fanning out from the conference bridge. Let this set of paths be called the outbound connection. In this Section we determine the optimal inbound and outbound connections such that the total number of channels required is minimised (i.e. the total routing cost is minimised).

3.1 Selectable media conferences

For selectable media conferences, the optimal inbound and outbound connections, i.e. those with the minimum number of channels, have the same topology. Let V be the set of nodes in the network, A be the set of processing nodes and $|A|$ be the number of elements in A. For a particular conference, let the network nodes with one or more conference sites connected be called conference nodes. Note that the conference bridges are only located in the processing nodes and the optimal conference bridge need not be located at a conference node. Therefore we need to enumerate all the nodes in A to determine the optimal conference bridge location. Given the locations of the conference sites, the subroutine $CB(i)$ gives the sum of the shortest path lengths connecting node i and the conference sites. The algorithm for determining the optimal connection paths for selectable media conferences or the SM algorithm is

SM algorithm

$$\begin{bmatrix} \text{input:} & \text{locations of the conference sites} \\ \text{output:} & \text{the optimal conference bridge location } C \end{bmatrix}$$

$min \leftarrow \infty$

FOR $i = 1$ TO N DO

  IF $(i \in A)$ and $(CB(i) < (min)$ THEN $min \leftarrow CB(i)$ and $C \leftarrow i$;

After the optimal conference bridge location is determined, the optimal connection paths can then be constructed by connecting the conference sites to node C by the shortest paths.

Subroutine $CB(i)$ is executed a total of $|A|$ times. In each execution a fixed number (equal to the number of conference nodes in the conference concerned) of shortest paths need to be found. If Dijkstra's shortest path algorithm of time complexity $O(N^2)$ [14] is used, the time complexity of the SM algorithm is $O(N^3)$ as $|A| \leq N$.

For on-demand calls, a fast determination of the optimal connection path at call set-up time is required. For a network with a large number of possible conference sites it would not be feasible to store in tables all possible connection patterns. A combination of table look-up and online processing to reduce the computation time should therefore be used. In the SM algorithm, subroutine $CB(i)$ finds the shortest path lengths between node i and all other conference nodes. These shortest paths can be computed offline and stored in tables. For a network with N nodes, there are $(N^2 - N)/2$ shortest paths between all node pairs. With these tables available, the SM algorithm has a time complexity of $O(N)$.

3.2 Common media conferences

All conferees of a common media conference receive the same composite video. The inbound connection, through which the conferees forward video/voice packets to the conference bridge, is the same as that for selectable media conferences. The outbound connection, on the other hand, is a Steiner tree (or a multicast tree) [15, 16] through which the conference bridge multicasts the processed packets to the conferees. Fig. 3 shows an example of inbound and outbound connections for a common media conference when all nodes of the network are processing nodes.

The optimal conference bridge location must be found before determining the optimal connection paths. Since a conference bridge need not be located at a conference node, we need to check all processing nodes to determine the optimal location of the conference bridge. The search can be performed as follows. Let T^* be the minimum Steiner tree connecting a given set of conference nodes D. As the topology of T^* is invariant when any processing node in the minimum Steiner tree T^* (i.e. any node in $T^* \cap A$) assumes the role of conference bridge, the determination of the optimal conference bridge location can be divided into two stages. The first stage is to determine the Steiner tree T^*, select the best conference bridge location among the processing nodes in T^* and compute the number of channels in the resulting inbound and outbound connections. The second stage is to enumerate the processing nodes that have not been checked in stage 1 (i.e. enumerate all the processing nodes which are not included in the Steiner tree T^*) for conference bridge locations. In each enumeration the Steiner tree connecting the conference nodes and the conference bridge is determined and the number of channels needed in the resulting inbound and outbound connections is computed.

Let $Steiner(D)$ be a subroutine that finds a minimum Steiner tree connecting the given set of conference nodes D and has as outputs the set of Steiner tree nodes T_{node} and edges T_{edge}. As the determination of the minimum Steiner tree is NP-complete [17], finding an optimal solution is feasible only when the number of conference nodes is small (say, no more than five [16]). In this case, the optimal algorithm proposed in [16] can be used for $Steiner(D)$. When the number of conference nodes is large a good heuristic should be used. In particular, a fast heuristic should be used for ondemand conferences but a more elaborate heuristic can be used for prescheduled conferences. In addition, any multicast assumption will only affect $Steiner(D)$ but will not affect the procedures for connection optimisation. Therefore when we need to incorporate different multicast assumption we only need to modify $Steiner(D)$. The following algorithm summarises the procedures. Lines 1–6 perform the first stage and lines 7–15 perform the second stage.

CM algorithm

$$\begin{bmatrix} \text{input:} & \text{set of conference nodes} \\ \text{output:} & \text{optimal conference bridge location } C, \\ & \text{set of nodes } T_{node} \text{ and edges } T_{edge} \\ & \text{in outbound connection} \end{bmatrix}$$

1. call $Steiner(D)$;
2. $min \leftarrow \infty$;
3. FOR $i = 1$ TO N DO
4. IF node $i \in A \cap T_{node}$ THEN
5. IF $CB(i) < min$ THEN $min \leftarrow CB(i)$ and $C \leftarrow i$;
6. $min \leftarrow |T_{edge}| + min$;
7. FOR $i = 1$ TO N DO
8. IF node $i \in A - T_{node}$ THEN
9. BEGIN
10. $D \leftarrow D \cup \{$ node $i\}$;
11. call $Steiner(D)$;
12. $x \leftarrow |T_{edge}| + CB(i)$;
13. IF $x < min$ THEN $min \leftarrow x$ and $C \leftarrow i$;
14. $D \leftarrow D - \{$node $i\}$;
15. END.

To summarise, the inbound connection is the set of shortest paths connecting the conference sites to node C and the outbound connection is the tree defined by T_{node} and T_{edge}.

Subroutine $CB(i)$ is executed $|T_{node}|$ times in the first stage and $|A - T_{node}|$ times in the second stage. Since $|T_{node}| \leq N$ and $|A - T_{node}| \leq N$, the total execution time involving $CB(i)$ is $O(N^3)$. Subroutine $Steiner(D)$ is executed once in the first stage and $|A - T_{node}|$ times in the second stage. If Prim's minimum spanning tree heuristic of time complexity $O(N^2)$ [15] is used for $Steiner(D)$, the total execution time involving *Steiner*(*D*) is $O(N^3)$. Combining, the time complexity of the CM algorithm is found to be $O(N^3)$.

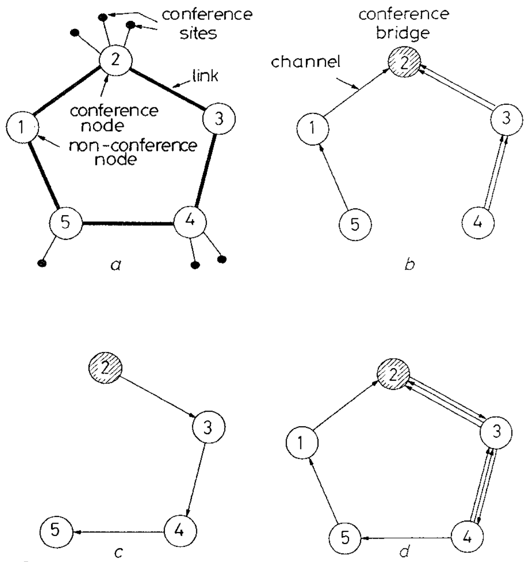

Fig. 3 *Inbound and outbound connections for common media conference*
a Locations of conferees
b minimum inbound connection
c minimum outbound connection
d optimal connection path

For ondemand calls, a combination of table lookup and online processing can be used. Recall that subroutine *CB*(*i*) gives the sum of the shortest path lengths connecting node *i* and the conference sites. There are $(N^2 - N)/2$ shortest paths between all node pairs and they can easily be computed and stored in tables. Subroutine *Steiner*(*D*) gives a Steiner tree connecting all the involved conference nodes and the processing node. Let *R* be the maximum number of involved conference nodes in any conference. Then the resulting number of Steiner trees is

$$\sum_{i=2}^{R+1} \binom{N}{i}$$

Table 1 shows this number for different *R* and *N*. Practically, *R* would not be very large, say in the order of five or less. Therefore computing and storing all the optimal Steiner trees should not be a problem provided *N* is not larger than say, 30. If all Steiner trees for connecting a small number of nodes are stored, larger size Steiner trees can be derived from a fast heuristic such as Prim's minimum spanning tree heuristic [15] or heuristic C in [16]. Note that the solutions given by heuristic C in [16] can be progressively improved. Hence, the optimality of the Steiner trees stored in memory can be progressively improved.

4 Performance analysis

The analysis of multipoint videoconference service in a general network appears to be very difficult and no known effective analytical technique is available. In this Section we restrict our attention to fully connected networks where all nodes are processing nodes. In the analysis the usual Poisson arrival of calls and the exponentially distributed holding time assumptions are made. Only the analysis for selectable media conferences is given. For common media conferences, simplex channels instead of full duplex channels are allocated and the derivation of blocking probabilities is similar to that for selectable media conferences. The details can be found in [18].

Table 1: Number of Steiner trees

R	Steiner trees			
	N = 20	*N* = 30	*N* = 40	*N* = 50
3	6.175×10^3	3.190×10^4	1.021×10^5	2.511×10^5
4	2.168×10^4	1.744×10^5	7.601×10^5	2.370×10^6
5	6.044×10^4	7.682×10^5	4.598×10^6	1.826×10^7
6	1.380×10^5	2.804×10^6	2.324×10^7	1.181×10^8
7	2.639×10^5	8.657×10^6	1.001×10^8	6.550×10^8
8	4.319×10^5	2.296×10^7	3.736×10^8	3.160×10^9
9	6.166×10^5	5.301×10^7	1.221×10^9	1.343×10^{10}
10	7.846×10^5	1.076×10^8	3.533×10^9	5.079×10^{10}

For a selectable media conference it can be proved that the optimal conference bridge is located at the node with the largest number of conferees attached. For a common media conference the minimum Steiner tree is constructed by connecting the conference nodes directly and hence the optimal conference bridge is also located at the conference node with the largest number of conferees. We denote the strategy which places the conference bridge at its optimal location the optimal strategy. For comparison, we also consider an alternate strategy in which the conference bridge is located at the call initiating node.

We model the conference calls as customers, the network links as facilities, and the channels in each link as servers. Since the number of conferees and the locations of the conferees in each conference are random variables the arrival of a conference call would mean a customer requesting a simultaneous possession of a random number of servers from several facilities. Kelly [19] investigated the blocking probability in circuit switched networks where each customer requests a fixed number of channels from a set of links. By decomposing the conference traffic (as explained in the following Section), Kelly's result can be applied to allocate a random number of servers. However, Kelly's model requires the enumerations of all possible paths in the networks, which is upper bounded by 2^x where *x* is the total number of links in the network. Therefore for any nontrivial network this approach cannot lead to numerical results. We observe that a properly designed videoconference service should have a low blocking probability, say no more than 10^{-2}. Under this condition the link occupancies can be assumed to be independent [20] and the maximum number of state variables can be reduced from a set of 2^x variables to *x* independent sets of $(M - 1)$ variables, where *M* is the maximum allowable number of conferees in a conference. The following is a derivation of the blocking probabilities for the reduced state space model.

In the network, let b_1, b_2, ... be the total number of conference subscribers at node 1, node 2, ..., and let the b_is be sufficiently large so that the arrivals of conference calls can be well approximated by a Poisson process just like that in the analysis of a telephone system. When the channel facilities are not all available for a conference call, this call is blocked and does not return. Define conference size $\tilde{W}$ as the total number of conferees in a conference and let the distribution of W be denoted as $w_i = P[W = i]$ (where $i = 2, 3, ..., M$) and be known. Knowledge of this distribution is necessary for a complete specification of the input traffic.

4.1 Analysis of alternate strategy

4.1.1 Decomposition of arrival process: The conferences initiated at a particular node, say node p, may involve conferees at the other nodes and thus may demand channels on the links connected to node p. Let λ_p be the arrival rate of conferences initiated at node p, $Poisson[\lambda_p]$ denotes a Poisson process with rate λ_p, and let $\beta_q(i)$ be the probability that a conference initiated at node p involves i conferees at node q. We decompose $Poisson\ [\lambda_p]$ into $M - 1$ processes $Poisson\ [\lambda_p\beta_q(1)]$, $Poisson\ [\lambda_p\beta_q(2)]$, ..., $Poisson[\lambda_p\beta_q(M - 1)]$ [21], where each arrival of $Poisson[\lambda_p\beta_q(i)]$ requests i channels on link pq. This decomposition is illustrated in Fig. 4.

Since a conference may be initiated at node p or node q, both $Poisson[\lambda_p]$ and $Poisson\ [\lambda_q]$ would load traffic on link pq Fig. 4. As arrivals of both $Poisson[\lambda_p\beta_q(i)]$ and $Poisson[\lambda_q\beta_p(i)]$ demand i channels on link pq, one can aggregate [21] these two processes to form $Poisson[\lambda_{pq}(i)]$ where $\lambda_{pq}(i) = \lambda_p\beta_q(i) + \lambda_q\beta_p(i)$.

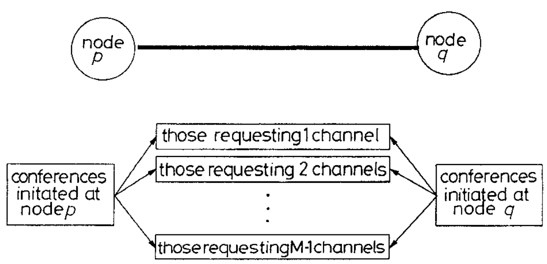

Fig. 4 *Decomposition of conference traffic*

4.1.2 Derivation of $\beta_q(i)$: Let all conferees have equal community interest on all the others. Then the probability γ_q that a conferee is located at node q is $\gamma_q = b_q/\Sigma_{i=1}^{N} b_i$. The probability $\beta_q(i)$ is

$$\beta_q(i) = \sum_{s=i+1}^{M} \binom{s-1}{i} \gamma_q^i (1-\gamma_q)^{s-1-i} w_s, \quad i = 0, 1, 2, \ldots, M-1$$

where M is the maximum allowable number of conferees in a conference and $w_1 = 0$.

4.1.3 Link occupancy distribution: The channel occupancy on link pq can be described by the state vector $\tilde{\mathbf{n}}_{pq} = (n_1, n_2, ..., n_{M-1})$ where n_i is the number of ongoing conferences on link pq occupying i channels each. The dependency of n_i on p and q is dropped for simplicity. Let $1/\mu$ be the mean conference duration and L_{pq} be the number of logical full-duplex channels on link pq. Since the total number of occupied channels on link pq cannot be larger than L_{pq}, the set Λ_{pq} of admissible states on link pq is

$$\Lambda pq = \left\{ \tilde{\mathbf{n}}_{pq} | \sum_{j=1}^{M-1} j n_j \le Lpq \right\}$$

The state probability $P[\tilde{\mathbf{n}}_{pq}]$ is given by a product form solution [19]

$$P[\tilde{\mathbf{n}}_{pq}] = \begin{cases} C_{pq} \frac{\rho_1^{n_1}}{n_1!} \frac{\rho_2^{n_2}}{n_2!} \cdots \frac{\rho_{M-1}^{n_{M-1}}}{n_{M-1}!} & \tilde{\mathbf{n}}_{pq} \in \Lambda_{pq} \\ 0 & \text{otherwise} \end{cases}$$

where ρ_i is defined as $\lambda_{pq}(i)/\mu$ (the dependency of ρ_i on p and q is also omitted here for simplicity) and C_{pq} is a normalisation constant and can be computed in the usual way [21]. Let $\tilde{\mathbf{N}}_{pq}$ be the channel occupancy on link pq. The distribution of $\tilde{\mathbf{N}}_{pq}$ can be expressed as

$$P[\tilde{N}_{pq} = k] = \sum_{\sum_j j n_j = k} P[\tilde{\mathbf{n}}_{pq}] \tag{1}$$

For the special case where there are only two conferees in all conferences eqn. 1 is reduced to the Erlang B formula.

4.1.4 Blocking probability: For a particular conference, let k_i be the number of conferees located at node i and $\mathbf{k} = (k_1, k_2, ..., k_N)$ be the distribution of conferees in the network. A conference call is blocked when at least one of the involved links does not have sufficient free channels. Given that a conference initiated at node p has s conferees, the blocking probability $B_p(s)$ can be found as

$$B_p(s) = 1 - \sum_{\Omega_1} \prod_{\substack{q=1 \\ q \ne p}}^{N} \left(P[\tilde{N}_{pq} \le L_{pq} - k_q] \times P\left[\mathbf{k} | \sum_{j=1}^{N} k_j = s \text{ and } k_p \ge 1\right] \right)$$

where

$$\Omega_1 = \left\{ \mathbf{k} | \sum_{j=1}^{N} k_j = s \text{ and } k_p \ge 1 \right\}$$

$$P\left[\mathbf{k} | \sum_{j=1}^{N} k_j = s \text{ and } k_p \ge 1\right] = \binom{s-1}{k_1, k_2, \ldots, k_p - 1, \ldots, k_N} \gamma_p^{k_p - 1} \prod_{\substack{i=1 \\ i \ne p}}^{N} \gamma_i^{k_i}$$

Removing the conditionings on s and p, the blocking probability B is obtained as

$$B = \frac{\sum_{p-1}^{N} \lambda_p \left[\sum_{s=2}^{M} B_p(s) w_s \right]}{\sum_{j=1}^{N} \lambda_j}$$

4.2 Analysis of optimal strategy

Let Θ_p denote the event that node p is chosen as the conference bridge and let $\chi_{pq}^i(j)$ be a composite event defined as

$\chi_{pq}^i(j)$ = {node p and $(i - 1)$ other nodes all have the same largest number of conferees attached; Θ_p; and node q has j conferees attached} $P[\chi_{pq}^1(j)]$ can be expressed in terms of the joint distribution of k_is as $P[\chi_{pq}^1(j)] = \Sigma_{\Omega 2} P[\mathbf{k}]$ where

$$\Omega_2 = \{\mathbf{k} | (2 \le k_1 + k_2 + \ldots + k_N \le M) \text{ and } (k_p > k_i \text{ for all } i \ne p) \text{ and } (k_q = j)\}$$

Similarly, $P[\chi_{pq}^2(j)]$ is given by

$$P[\chi_{pq}^2(j)] = \frac{1}{2} \sum_{\substack{u=1 \\ u \ne p}}^{N} \sum_{\Omega_3} P[\mathbf{k}]$$

where

$$\Omega_3 = \{\mathbf{k}|(2 \le k_1 + k_2 + \ldots + k_N \le M) \text{ and } (k_u = k_p \text{ and } k_p > k_i \text{ for all } i \ne u \ne p) \text{ and } (k_q = j)\}$$

$P[\chi^3_{pq}(j)]$, $P[\chi^4_{pq}(j)]$, ... can be found in a similar fashion. Finally, $P[\chi^N_{pq}(j)]$ is given by

$$P[\chi^N_{pq}(j)] = \frac{1}{N} P[\mathbf{k} = (j, j, \ldots, j)]$$

The probability $\sigma_{pq}(j)$ that a conference demands j channels on link pq is

$$\sigma_{pq}(j) = \sum_{i=1}^{N} \{P[\chi^i_{pq}(j)] + P[\chi^i_{qp}(j)]\}$$

Fig. 5 shows that under the optimal strategy the maximum number of channels required in any link is $\lfloor M/2 \rfloor$. Since the total rate of conference arrivals λ_T is the sum of the rates at individual nodes and is given by $\lambda_T = \Sigma_{i=1}^{N} \lambda_i$, the traffic loaded onto link pq by $Poisson[\lambda_T]$ can be decomposed into $Poisson[\lambda_T \sigma_{pq}(1)]$, $Poisson[\lambda_T \sigma_{pq}(2)]$, ..., $Poisson[\lambda_T \sigma_{pq}(\lfloor M/2 \rfloor)]$, where each arrival of $Poisson[\lambda_T \sigma_{pq}(i)]$ demands i channels on link pq. The state probabilities and link occupancies can be found in exactly the same manner as that for the alternate strategy in the previous Section. A conference call is blocked when all the optimal conference bridge locations cannot provide sufficient free channels for this call. Given that a conference has s conferees, the blocking probability $B(s)$ is given by

$$B(s) = 1 - \sum_{\Omega_4} \sum_{p=1}^{N} \prod_{\substack{q=1 \\ q \ne p}}^{N} \left(P[\tilde{N}_{pq} \le L_{pq} - k_q | k_q] \times P\left[\mathbf{k} \text{ and } \Theta_p | \sum_{n=1}^{N} k_n = s\right] \right) \quad (2)$$

where $\Omega_4 = \{\mathbf{k}|\Sigma_{n=1}^{N} k_n = s\}$ and

$$P\left[\mathbf{k} \text{ and } \Theta_p | \sum_{n=1}^{N} k_n = s\right] = P\left[\Theta_p | \sum_{n=1}^{N} k_n = s \text{ and } \mathbf{k}\right] P\left[\mathbf{k} | \sum_{n=1}^{N} k_n = s\right]$$

$$= \begin{cases} \frac{1}{j} \binom{s}{k_1, k_2, \ldots, k_N} \prod_{i=1}^{N} \gamma_i^{k_i} & \text{if node } p \text{ and } j-1 \text{ other nodes all have the same largest number of conferees and } \sum_{n=1}^{N} k_n = s \\ 0 & \text{otherwise} \end{cases}$$

Removing the condition on s, the blocking probability is $B = \Sigma_{s=2}^{M} B(s) w_s$.

4.3 Examples and discussions

In all the following examples a five-node fully connected network is considered and $\mu = 1$, $\lambda_i = \lambda$ and $L_{ij} = L$ for all i and j are assumed. We measure the total offered load in terms of Erlangs which is defined to be λ_T/μ [24].

Fig. 6 shows the blocking probabilities of selectable media conferences using the optimal and the alternate strategies against the total offered load λ_T/μ assuming the conference size distribution is of the truncated geometric type with a mean of three. We see that in general the optimal strategy gives about one to two orders of magnitude smaller blocking than the alternate strategy. At $B = 10^{-2}$ and for $L = 30$, the network using the alternate strategy has a throughput of 110 Erlangs while using the optimal strategy it increases to 132 Erlangs. This 20% increase of network throughput should well justify the use of optimal connection paths for setting up conferences.

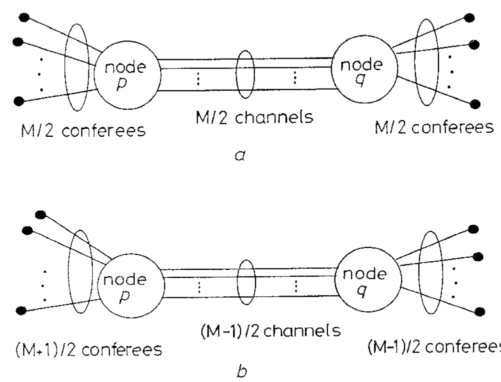

Fig.5 *Maximum channel requirements, optimal strategy*
a Even M
b Odd M

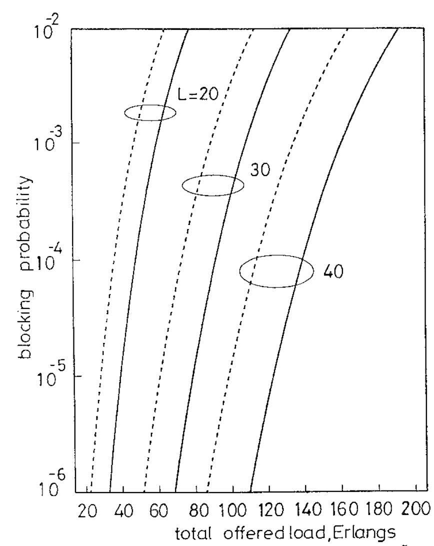

Fig.6 *Comparison of optimal and alternate strategies $E[\tilde{W}] = 3$*
——— optimal strategy
- - - - alternate strategy

To study the sensitivity of blocking probability to the conference size distribution, we consider three hypothetical distributions: truncated geometric distribution, uniform distribution, and delta distribution. For the same mean conference size, the truncated geometric distribution has the largest variance and the delta distribution has the smallest (zero) variance. Fig. 7 shows the throughput-blocking characteristics of selectable media conferences for these three distributions assuming the optimal strategy is used for placing conference bridges. It is seen that when the mean conference size is increased from three to four, the maximum load the network can take (or the network capacity) is decreased from 191 Erlangs to 123 Erlangs for the geometric distribution at $B = 10^{-2}$. The variance of the conference size distribution, on the other hand, is seen to have a small but non-negligible effect on the network throughput. For a given blocking requirement and a given mean conference size, the larger the variance the smaller is the throughput.

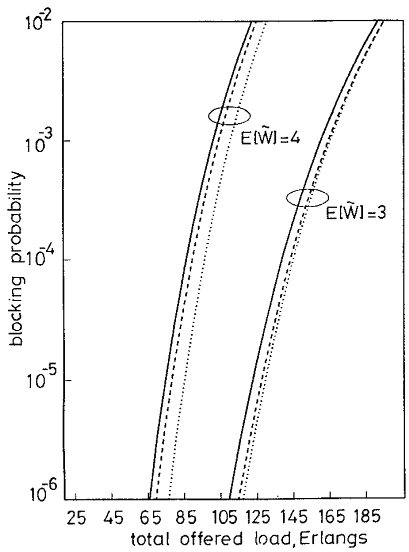

Fig. 7 *Comparison of different conference distributions (L = 40)*
——— truncated geometric
- - - - uniform
········ delta

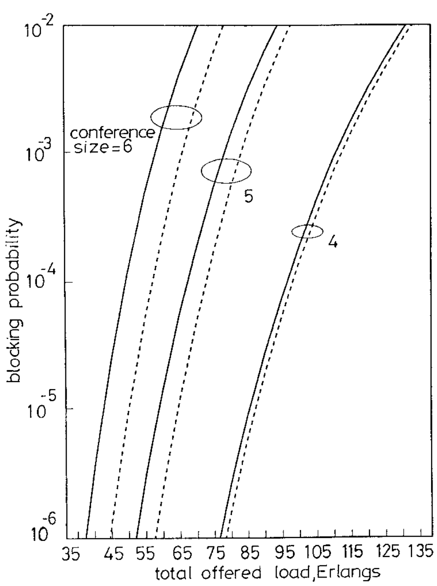

Fig. 8 *Performance comparison of selectable media and common media conferences (L = 40)*
——— selectable media
- - - - common media

Fig. 8 compares the performance of selectable media and common media conferences. The optimal strategy is used for placing conference bridges and the conference size is assumed to be a constant and equals four, five and six for the three sets of curves. The common media conferences require less channel resources from the network than selectable media conferences because sharing of channels in the outbound connection is possible. The amount of sharing, moreover, increases with the conference size. Thus at $B = 10^{-2}$ and conference size equal to five, the maximum network throughput for common media and selectable media conferences are 98 and 94 Erlangs respectively. This represents a 4.3% larger throughput for common media conferences. When the conference size is six a 11.4% increase of throughput is observed for common media conferences. In fully connected networks, only conferees connected to the same node can share a channel. In a general network those connected to different nodes may also be able to share channels in the outbound connection (i.e. the Steiner tree) [16]. Hence, in a general network, the network throughput for common media conferences is expected to be much higher than that for selectable media conferences.

5 Conclusions

The connection-path problems for two types of multipoint videoconferences, called selectable-media and common-media conferences, were formulated and algorithms were designed to find the connection paths with minimum number of channels. For on-demand calls which request immediate seizure of the transmission resources, a fast determination of the optimal connection path at call set-up time is required. A method was proposed that uses a combination of lookup table and online processing to reduce the computation time. The blocking probabilities as a function of network traffic for selectable media and common media conferences in fully connected networks were derived and compared. The network throughput for selectable media conferences was found to be smaller than that for common media conferences. However, selectable media conferences allow conferees to receive their own video compositions.

6 References

1 LIAO, K.Q., and ROBERTS, J.W.: 'Videoconference traffic and network design', *IEEE Trans.*, March 1987, **COM-35**, (3), pp. 275–282
2 WATANABE, K., MURAKAMI, S.I., ISHIKAWA, H., and KAMAE, T.: 'Audio and visually augmented teleconferencing', *Proc. IEEE*, April 1985, **73**, (4), pp. 656–669
3 SABRI, S., and PRASADA, B.: 'Video conferencing systems', *Proc. IEEE*, April 1985, **73**, (4), pp. 671–688
4 ROMAHN, G.: 'System aspects of multipoint videoconferencing'. Proceedings of IEEE GLOBECOM, 1987, pp. 723–725
5 ROBINSON, J., RUBINOV, E., TOULSON, C., PRASADA, B., SABRI, S., GOLDBERG, N., and VONDERWEIDT, G.: 'A multimedia interactive conferencing applications for personal workstations', *IEEE Trans.*, November 1991, **COM-39**, (11), pp. 1698–1708
6 VONDERWEIDT, G., ROBINSON, J., TOULSON, C., MASTRONARDI, J., RUBINOV, E., and PRASADA, B.: 'A multipoint communication service for interactive applications', *IEEE Trans.*, December 1991, **COM-39**, (12), pp. 1875–1885
7 KATO, S., KUBOTA, S., KAZAMA, H., and MORIKURA, M.: 'Novel satellite digital video TDMA system for business video communications', *IEEE J. Sel. Areas Commun.*, August 1992, **10**, (6), pp. 1054–1065
8 FERGUSON, M.J., and MASON, L.: 'Network design for a large class of teleconferencing systems', *IEEE Trans.*, July 1984, **COM-32**, (7), pp. 789–796
9 BODSON, D., and SCHAPHORST, R. (Ed.): 'Teleconferencing' (IEEE Press, 1989)
10 LEE, T.T.: 'Non-blocking copy networks for multicast packet switching', *IEEE J. Sel. Areas Commun.*, December 1988, **6**, (9), pp. 1455–1467
11 LEUNG, Y.W., and YUM, T.S.: 'A modular multirate video distribution system - design and dimensioning', *IEEE/ACM Trans. Netw.*, December 1994, **2**, (6), pp. 549–557
12 MAGLARIS, B., ANASTASSIOU, D., SEN, P., KARLSSON, G., and ROBBINS, J.D.: 'Performance models of statistical multiplexing in packet video communications', *IEEE Trans.*, July 1988, **COM-36**, (7), pp. 834–844
13 SEN, P., MAGLARIS, B., RIKLI, N.E., and ANASTASSIOU, D.: 'Models for packet switching of variable-bit-rate video sources', *IEEE J. Sel. Areas Commun.*, June 1989, **7**, (5), pp. 865–869

14 AHO, A.V., HOPCROFT, J.E., and ULLMAN, J.D.: 'Data structures and algorithms' (Addison–Wesley, 1983)
15 AHAMAD, M. (Ed.): 'Multicast communication in distributed systems' (IEEE Press, 1990)
16 LEUNG, Y.W., and YUM, T.S.: 'Efficient algorithms for multiple destinations routing', *Digit. Analog Commun. Syst.*, April 1993, **6**, (2), pp. 87–95
17 GAREY, M.R., and JOHNSON, D.S.: 'Computers and intractability: A guide to the theory of NP-completeness' (Freeman, San Fransisco, CA, 1979)
18 LEUNG, Y.W.: 'Routing and switching in teleconferencing networks'. PhD dissertation, Department of Information Engineering, The Chinese University of Hong Kong, 1992
19 KELLY, F.P.: 'Blocking probabilities in large circuit-switched networks', *Adv. Appl. Prob.*, 1986, **18**, pp. 473–505
20 WHITT, W.: 'Blocking when service is required from several facilities simultaneously', *AT & T Tech. J.*, October 1985, **64**, (8), pp. 1807–1856
21 KOBAYASHI, H.: 'Modelling and analysis: An introduction to system performance evaluation methodology' (Addison–Wesley, 1978), pp. 108–110
22 YUM, T.S., CHEN, M.S., and LEUNG, Y.W.: 'Bandwidth allocation for multimedia teleconferences', *IEEE Trans.*, February 1995, **COM-43**, (2)
23 HEEKE, H.: 'A traffic control algorithm for ATM networks', *IEEE Trans. Cir. Syst. Video Tech.*, June 1993, **3**, (3), pp. 182–189
24 KLEINROCK, L.: 'Queueing systems, Vol. 1: theory' (Wiley, 1975)