Strictly Nonblocking Conference Networks Using High-Dimensional Meshes

Yi Du, Gerald M. Masson

Department of Computer Science, The Johns Hopkins University, Baltimore, Maryland 21218

Received September 1997; accepted December 1997

Abstract: This paper introduces a conferencing server design based on an innovative configurable computing architecture to support the information transmission and distributed processing associated with creating and maintaining simultaneous disjoint conferences among sets of N conferees, where r-dimensional meshes can be used as the conferencing components. An r-dimensional conferencing mesh network is strictly nonblocking if, regardless of the existing conferences implemented, a new conference among any subset of the idle conferees can be implemented using a connected set of idle processing elements without any disturbance to the existing conferences. Using arguments employing the isoperimetric ratios of the sizes of edge and node sets in a graph, we give necessary and sufficient conditions such that an r-dimensional conferencing mesh of M nodes provides strictly nonblocking conferencing to N conferees. We show that a necessary and sufficient condition for r-dimensional meshes to be strictly nonblocking, when dimension r is fixed, is that $M = O(N^{(r+1)/r})$. For general r-dimensional meshes, $M = O(r^{(r-1)/r}N^{(r+1)/r})$ nodes are sufficient to support strictly nonblocking capabilities. A fundamental relationship is established between the requirements on M for strictly nonblocking conferencing among N conferees using certain graph structures and the isoperimetric ratios for those structures. Networks 33: 293–308, 1999

1. INTRODUCTION

In this paper, we study constructions of a novel architecture for a configurable conferencing server which supports the simultaneous transmission of processed messages among conference participants, referred to as conferees. Each conferee uses a generic send/receive device as a conferencing instrument providing access to/from the server. Depending on the application, the messages exchanged by the conferees can involve audio, video, graphics, data, or essentially any such information type, including combinations thereof.

Correspondence to: Y. Du
Contract grant sponsor: Compunetix Inc. and DARPA

The configurable conference servers that we will consider consist of two components: a bidirectional $N \times M$ switching network used for routing information to/from an M node conferencing network comprising processing elements which perform the actual signal processing of the information associated with the conference. Figure 1 illustrates the general structure of the configurable conferencing servers that we are studying. As shown, a conference among a specified set of conferees is implemented on the conferencing network by the processing of information exchanged among a set of directly connected processing elements, each of which is linked to exactly one of the send/receive devices associated with the conferees using the switching network.

We are interested in conferencing applications in which upon setting up a requested conference among a

 CCC 0028-3045/99/040293-16

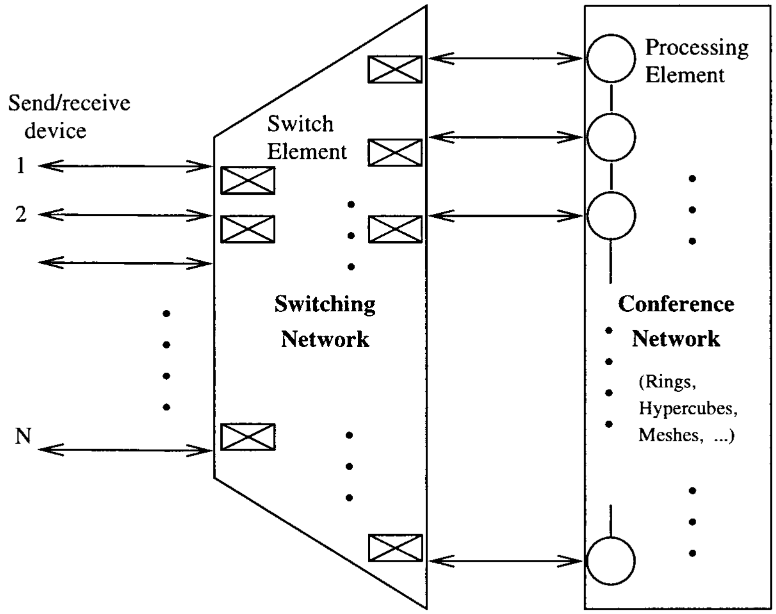


Fig. 1. A general structure of a configurable conferencing server.

group of conferees each conferee will remain in the conference until the entire conference is terminated. This is a reasonable assumption as often the conference message streams being exchanged, such as video and audio streams for so-called video-conference application or data and graphics information for distributed simulation environments, will demand a specified upper bound on the delays for acceptable message delivery, together possibly with certain communication bandwidth requirements. In conferencing application demanding a high quality of service, these message streams can be quite sensitive to rearrangements and/or transmission disturbances resulting from reconfigurations of existing conferences. Thus, disruptions of these message streams resulting from additions or deletions of conferees within an established conference cannot always be tolerated.

Similarly, to supporting a conference of, say, k conferees, we assume that a connected component of k processing elements are allocated which are not involved in any way with other existing conferences being supported in the server. Thus, these k processing elements associated with the k conference form a connected component of nodes in the conference component of the server. This means that the bandwidth of the links connecting processing elements as well as the processing power of the elements allocated are entirely available to supporting a given conference.

Figure 2 illustrates an implementation of two conferences in this conferencing server architecture. In this scenario, conference #1 comprises three conferees, namely A, C, and E, supported by the connected component of processing elements 2, 3, and 4, respectively. To illustrate the processing and exchange of information associated with the conference, the figure shows, for example, that conferee (user) C provides an input stream, denoted as "C," to the allocated processing elements for the conference and receives the output stream denoted as "$A + B$." Also shown is conference #2 comprising the conferees B, D, and F being supported by the connected component of processing elements 6, 7, and 8, respectively. The configurations of the processing elements supporting each of the shown conferences is in each case a connected series of three processing elements in this ring network.

In this paper, we will consider r-dimensional conferencing meshes as the conference network component of our server architecture for providing strictly nonblocking conferencing capability. Clearly, the $N \times M$ switching network component of our architecture must be strictly nonblocking. However, this can be provided in numerous ways [3], employing, for example, the well-known Cantor network construction [5]. Hence, we give this issue no particular attention in our discussions.

Our main focus will be the strictly nonblocking capability of the conference network component of our server which realizes the conferencing applications. An r-dimensional conferencing mesh is *strictly nonblocking* if, regardless of the existing conferences implemented on the r-dimensional mesh, a new conference among any subset of the idle conferees can be implemented using a connected set of idle processing elements without any disturbance (rearrangement) to the existing conferences and without any information being passed-through processing elements allocated to other conferences. Using arguments employing the so-called isoperimetric ratios of

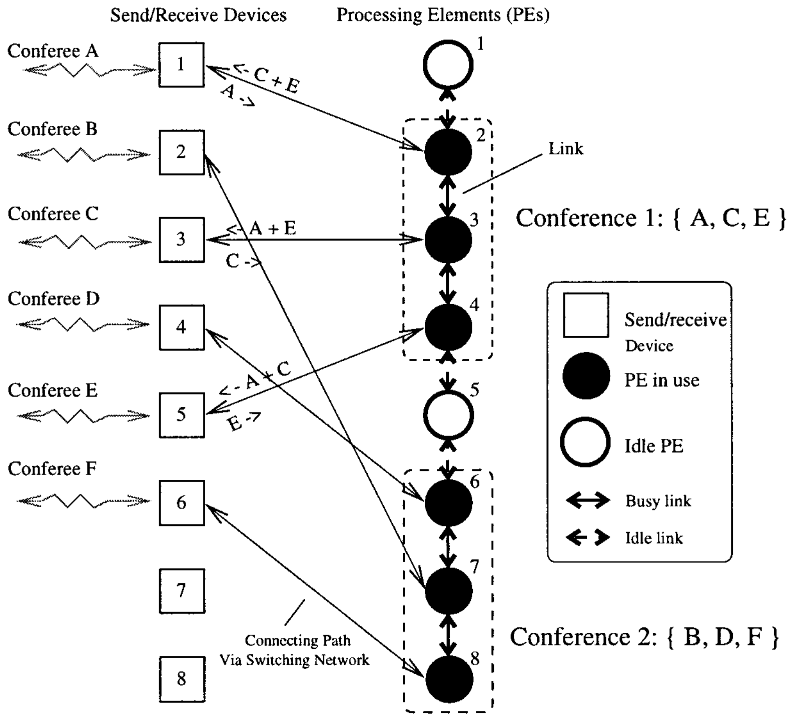


Fig. 2. Assignment of conferees in a conference network.

the sizes of edge and node sets in an r-dimensional mesh, we will show that a necessary and sufficient condition such that an r-dimensional conferencing mesh of M processing elements provides strictly nonblocking conferencing for a set of N conferees is that $M = O(N^{(r+1)/r})$ processing elements, for fixed dimension r. For general r-dimensional meshes, $M = O(r^{(r-1)/r} N^{(r+1)/r}$ nodes are sufficient to support strictly nonblocking conferencing among N conferees. Our analyses will be seen to give further insight into related results relative to the use of r-dimensional meshes in configurable conferencing server architectures. Finally, we will consider algorithms for the allocation of processing elements to satisfy conference requests in our configurable conferencing server design.

2. PREVIOUS RESEARCH ON CONFERENCING SERVER ARCHITECTURES

Configurable conferencing server architectures similar to the general structure being considered in this paper have been previously explored. Yang and Masson [13] considered the use of a sequential ring of $M = O(N)$ processing elements for a multicast conference network in which rearrangements of existing conferencing configurations are permitted to satisfy a request to provide for the addition of a new conferee to a conference. Also relative to the use of a series of processing elements in the shape of a ring as the conferencing network component, the results of Woodall [12] show that $M = O(N^2)$ processing element nodes are necessary and sufficient to support strictly nonblocking conferencing. Hypercube constructions for the conferencing network component have also been studied [9, 10] as conferencing networks, and it has been shown that $M = O(N\sqrt{\log N})$ nodes are sufficient for a hypercube to support a strictly nonblocking conferencing for N conferees. The study in [11] considers the special case of a 2-D mesh design and shows that $O(N^{3/2})$ is necessary and sufficient for strictly nonblocking conferencing.

3. CONDITIONS FOR A STRICTLY NONBLOCKING *r*-DIMENSIONAL MESH

In this section, we will determine the necessary and sufficient conditions on the number of processing elements (nodes) required for a strictly nonblocking conferencing network. In particular, we are interested in the number, M, of nodes in an r-dimensional mesh which can support conferences comprising subsets of N conferees when a single link between any two nodes has the capacity to accommodate at most one conference. Our proof will be based on a general technique that can be used to obtain upper and lower bounds on M for general conferencing network structures with certain properties.

We begin by giving the necessary condition on M for

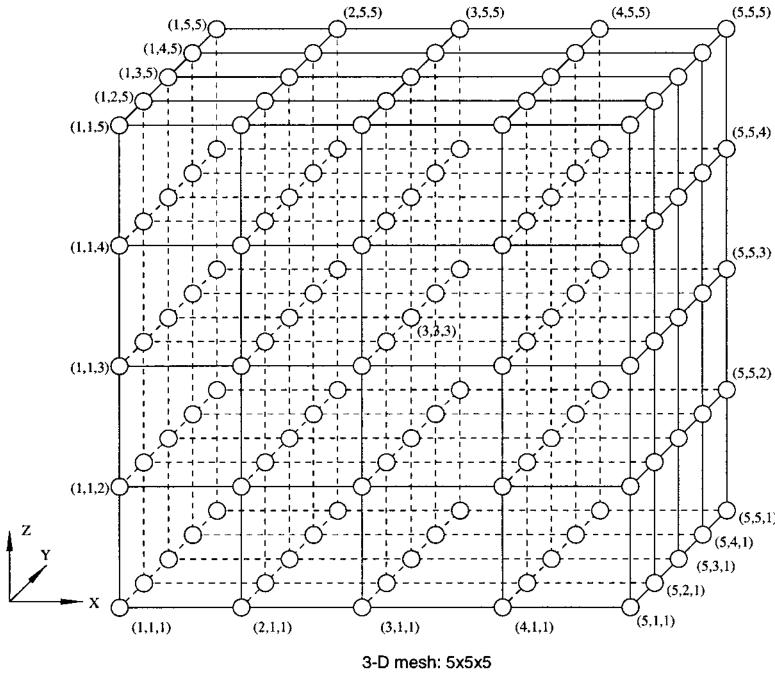


Fig. 3. Node coordinates in 3-D mesh with 125 nodes with length $L = 5$.

r-dimensional meshes for a strictly nonblocking conferencing capability.

3.1. Necessary Condition on M for a Strictly Nonblocking r-Dimensional Mesh Serving N Conferees

Consider an r-dimensional (i.e., an r-D) mesh with M nodes. Let L be the length of any side of the r-D mesh. This length is measured in terms of the number of nodes in a specific dimension of the mesh; it follows that $L = M^{1/r}$. It should also be noted that, depending on its location in an r-D mesh, every node has an edge (or link) degree between r and $2r$.

The location of each node v in the r-D mesh can be specified using a vector listing its coordinates in each dimension: $v = \langle i_1^v, i_2^v, \ldots, i_j^v, \ldots, i_r^v \rangle$, where $1 \le i_1^v, i_2^v, \ldots, i_r^v \le L$. Thus, i_j^v indicates the coordinate of node v in the j dimension. Two nodes $v = \langle i_1^v, i_2^v, \ldots, i_r^v \rangle$ and $w = \langle i_1^w, i_2^w, \ldots, i_r^w \rangle$ are connected by a link in an r-D mesh if there exists a k, where $1 \le k \le r$, such that for all $j \ne k$ we have $i_j^v = i_j^w$ and $i_k^v = i_k^w + 1$ or $i_k^v = i_k^w - 1$.

An *ordering* of nodes in the r-D mesh can be determined as follows: Node $v = \langle i_1^v, i_2^v, \ldots, i_r^v \rangle$ is said to be *smaller* than $w = \langle i_1^w, i_2^w, \ldots, i_r^w \rangle$ if there exists k where $1 \le k \le r$, such that for all $j < k$, $i_j^v = i_j^w$ and $i_k^v < i_k^w$.

Figure 3 illustrates a 3-D mesh network with 125 nodes with length $L = 5$. For this case, each node v can be identified by a triple $\langle i_1^v, i_2^v, i_3^v \rangle$, where $1 \le i_1^v, i_2^v, i_3^v \le 5$. For this 3-D case, we can also use the X-, Y- and Z-coordinates in 3-D Euclidean space in the identification of nodes. Each node v is labeled as $\langle x_v, y_v, z_v \rangle$. As an example of the node ordering in this 3-D mesh, the smallest node is labeled $\langle 1, 1, 1 \rangle$, the largest node is labeled as $\langle 5, 5, 5 \rangle$, and the center node is labeled $\langle 3, 3, 3 \rangle$.

A *hyperplane* in the r-D mesh consists of a set of nodes with a coordinate of one dimension having the same value. For example, all the nodes with coordinates having the same value x $(1 \le x \le L)$ in dimension j (i.e., all nodes such that $i_j = x$) are contained in a hyperplane in dimension j. We will denote as $H_j(x)$ the hyperplane of nodes each with coordinate x in dimension j. In Figure 3, each of the three shown surfaces with darkened links are hyperplanes, denoted as $H_X(3)$, $H_Y(3)$, and $H_Z(3)$, since each corresponds to a set of nodes which all have the same value (namely, 3) as one of its X, Y, or Z coordinates.

It is obvious that nodes in a hyperplane compose a connected graph. Thus, in satisfying a conference, assigning conferees in a conference to all the nodes in the same hyperplane is an acceptable node allocation. However, it should be noted that doing so partitions the remaining nodes in the r-D mesh so that the size of future conferences is limited. This is a key observation to establishing a necessary condition on M for a strictly nonblocking capability.

Thus, to obtain a lower bound on the number M of nodes needed for a strictly nonblocking r-D mesh, we will consider a specific conference request sequence and a method to allocate nodes to conferees to satisfy these requests. In particular, we will refer to a *hyperplane allocation* of nodes for satisfying conference requests wherein the conferees associated with a conference request are assigned all the nodes in some hyperplanes in the r-D mesh. When the nodes in some hyperplanes are all assigned to a conference, the mesh becomes disconnected into several disjoint *connected components.* Obviously, to guarantee a strictly nonblocking capability for future conferences that might be requested by idle conferees, at least one of these connected components (namely, the one with most nodes) must be of sufficient size to support any remaining conference request.

To support our proof of the main theorem for a lower bound on M, consider the following lemma which presents a regular scheme by which a hyperplane can separate an r-D mesh into several mutually disjoint submeshes, each similar in structure but with a smaller size relative to the original mesh. It will be useful in this lemma to denote as $B(L) = \cup_{j=1}^{r} H_j(L/2)$ the total number of nodes in the r hyperplanes $H_j(L/2)$, $1 \le j \le r$ in a r-D mesh of length L in each dimension. The total number of nodes, $B(L)$, in these hyperplanes will also be obtained from this lemma.

Lemma 1. *An r-D mesh of length L in each dimension can be separated into exactly $p = 2^r$ disjoint connected components by the r hyperplanes $H_j(\lceil L/2 \rceil) = \{\langle i_1, i_2, \ldots, i_j, \ldots, i_r \rangle : i_j = L/2\}$, where $1 \le j \le r$. Each of these disjoint connected components is a smaller r-D mesh with length $(L-1)/2$ and with $((L-1)/2)^r$ nodes. Finally, $B(L) = L^r - (L-1)^r$. Moreover, $B(L)$ is increasing on L.*

Proof. A hyperplane $H_j(L/2)$ $(1 \le j \le L)$ separates the r-D mesh into two parts: In one part, each coordinate $i_j < L/2$, and in the other part, $i_j > L/2$. It follows then that a collection of r hyperplanes, namely, $H_1(L/2)$, $H_2(L/2), \ldots, H_r(L/2)$, would separate the mesh into 2^r-connected components. The coordinates of each of the nodes in each of these components satisfy a system of inequalities: $i_1 > (<) L/2, i_2 > (<) L/2, \ldots, i_r > (<) L/2$. It can be observed that each of these components is also an r-D mesh with length at most $\lceil (L-1)/2 \rceil$. As an example, Figure 4 illustrates three hyperplanes used for partitioning a 3-D mesh. Figure 5 gives more details of this partitioning. Shown in this figure are the nodes on these hyperplanes, as well as nodes in the resulting components separated by these hyperplanes.

We can now count the number of nodes in these hyperplanes. Note that some of these hyperplanes contain common nodes. Hence, our counting will be based on an inclusion and exclusion procedure. Consider hyperplanes $H_j(L/2) = \{\langle i_1, i_2, \ldots, i_j, \ldots, i_r \rangle : i_j = L/2\}$, where $1 \le j \le r$. Clearly, the number of nodes in $H_j(L/2)$ is $|H_j(L/2)| = L^{r-1}$. The intersection of k hyperplanes $H_{j_1}(L/2), H_{j_2}(L/2), \ldots, H_{j_k}(L/2)$ is $\cap_{l=1}^{k} H_{j_l}(L/2)$, where $|\cap_{l=1}^{k} H_{j_l}(L/2)| = L^{r-k}$.

Since in this counting procedure we are always considering hyperplanes where the fixed coordinate is $i_j = L/2$, where $1 \le j \le r$, we will simply the notation in the following to more compactly write $H_{j_k}(L/2) = H_{j_k}$.

By the inclusion–exclusion principle, the total number of nodes in $\cup_{j=1}^{r} H_j$ is

$$
\begin{aligned}
|\cup_{j=1}^{r} H_j| &= \sum_{j=1}^{r} |H_j| - \sum_{j_1, j_2} |H_{j_1} \cup H_{j_2}| + \cdots \\
&\quad + (-1)^{k-1} \sum_{j_l = j_1, \ldots, j_k} |\cap H_{j_l}| + \cdots \\
&\quad + (-1)^{r-1} \sum_{j_l = j_1, \ldots, j_r} |\cap H_{j_l}| \\
&= \binom{r}{1} L^{r-1} - \binom{r}{2} L^{r-2} + \cdots \\
&\quad + (-1)^{k-1} \binom{r}{k} L^{r-k} + \cdots \\
&\quad + (-1)^{r-1} \binom{r}{r} L^{r-r} = L^r - (L-1)^r.
\end{aligned}
$$

Since the derivative of $B(L)$ is always positive, it follows that $B(L)$ increases with L. ■

The following lemma is useful for proving Theorem 1 giving a necessary condition on M.

Lemma 2. *If $a \ge b$ ce 0, for integer $n \ge 0$, we have $(a - b)^n \ge a^n - na^{n-1}b$.*

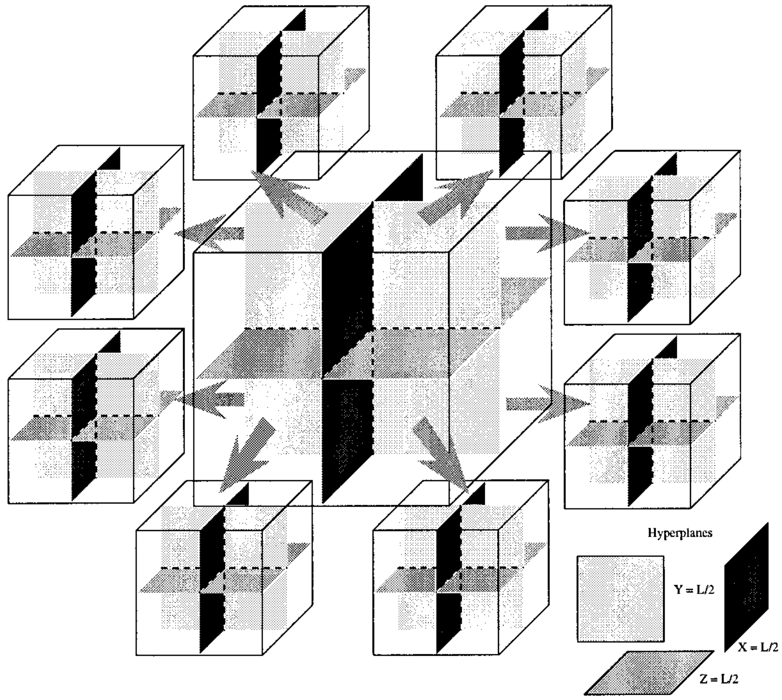
Proof. We can prove this by induction on n. The claim obviously holds for $n = 0$.

Suppose that it holds for $n = k$, that is, $(a-b)^k \ge a^k - ka^{k-1}b$. Consider $n = k + 1$. We have

$$
\begin{aligned}
(a-b)^{k+1} &= (a-b)(a-b)^k \\
&\ge (a-b)(a^k - ka^{k-1}b) = a^{k+1} - (k+1)a^k b \\
&\quad + \mathrm{k}a^k b^2 \ge a^{k+1} - (k+1)a^{(k+1)-1} b,
\end{aligned}
$$

that is, the claim holds for $n = k + 1$. ■

Now we can prove the necessary condition on M by considering the requirement on the number of nodes necessary to support strictly nonblocking conferencing among N conferees when a sequence of requests for con-



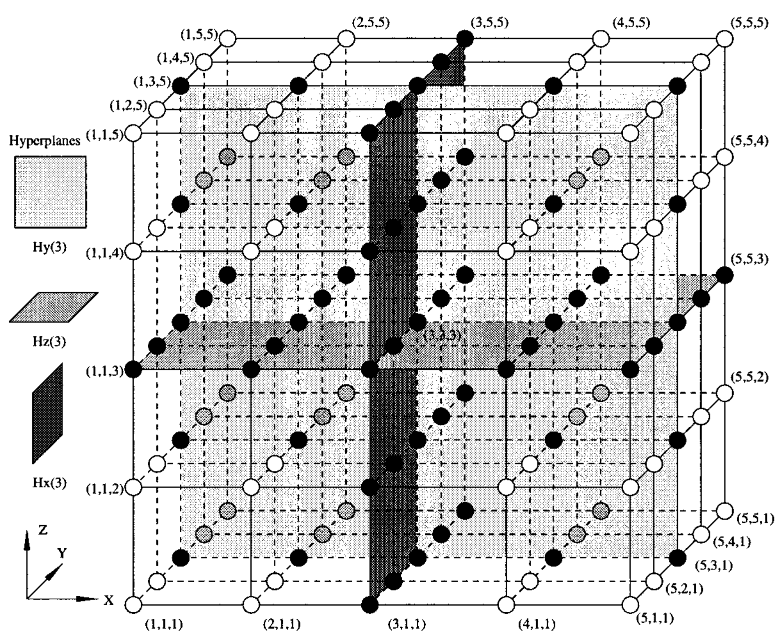
Recursive Partitioning of a 3-Dimensional Mesh: 8 sub-meshes obtained after each step

Fig. 4. Partitioning 3-D mesh using three hyperplanes.

ferences is satisfied using a repeated hyperplane allocation of nodes.

Theorem 1. $M = \Omega((N/r)^{(r+1)/r})$ *nodes are necessary for an r-D mesh to support nonblocking conferencing for N conferees. More precisely, the lower bound is* $[N/(r+1)]^{(r+1)/r}$.

Proof. Assume that an r-D mesh with M nodes is strictly nonblocking for N conferees. Suppose that from



Separate 2-D mesh using 3 hyperplanes: 8 sub-meshes obtained

Fig. 5. Locations of nodes in 3-D mesh after hyperplane partitioning.

the start no conferences have been realized on the r-D mesh, so that all M nodes are idle. Now suppose an initial conference request is made of size $B_1 = B(L) = L^r - (L - 1)^r$. Since this is the number of nodes in the hyperplanes $H_1(L/2), H_2(L/2), \ldots, H_r(L/2)$, we can allocate all the nodes in these hyperplanes to the conferees for this requested conference. As discussed above, we refer to this as a hyperplane allocation of nodes. By the above lemma, such an allocation of nodes in these hyperplanes partitions the r-D mesh of M nodes with length $L = M^{1/r}$ into 2^r disjoint smaller r-D meshes, each with $((L - 1)/2)^r$ nodes. Since we are assuming that the r-D mesh is strictly nonblocking, all additional conference requests must be possible to satisfy in a strictly nonblocking manner.

Let $S_1 = ((L - 1)/2)^r$ denote the number of nodes in each of the smaller meshes resulting from the first hyperplane allocation used to satisfy the initial request. Clearly, additional conferencing requests among the remaining $N - B_1$ idle conferees must now be satisfied within the 2^r disjoint r-D meshes, each with $((L - 1)/2)^r$ nodes. Thus,

$$S_1 \geq N - B_1,$$

where

$$S_1 \leq M/2^r$$

and

$$B_1 = B(L) = L^r - (L - 1)^r.$$

Next, suppose there are $p = 2^r$ additional conference requests, each for a conference of size $B_2 = B[(L - 1)/2]$ conferees. We can satisfy these new conference requests with a second set of $p = 2^r$ hyperplane allocations of nodes, each within one of the 2^r disjoint r-D meshes of $((L - 1)/2)^r$ nodes resulting from the first hyperplane allocation. The total number of nodes used to satisfy these $p = 2^r$ new conference requests is $B_2 = 2^r B[(L - 1)/2]$, where

$$\begin{aligned} B_2 &= pB((L - 1)/2) \leq pB(L/2) \\ &= 2^r\left(\left(\frac{L}{2}\right)^r - \left(\frac{L}{2} - 1\right)^r\right) = L^r - (L - 2)^r. \end{aligned}$$

Also, note that using a second round of hyperplane allocations of nodes within each of the $p = 2^r$ disjoint r-D meshes results in another partitioning of each of these smaller meshes into 2^r still smaller disjoint r-D meshes, each now having $((L - 1)/2^2)^r$ nodes. Let $S_2 = ((L - 1)/2^2)^r$ denote the number of nodes in each of the still smaller meshes obtained by the second round of hyperplane allocations used to satisfy the second set of $p = 2^r$ conference requests. Clearly, because the overall r-D mesh is assumed to be strictly nonblocking, additional conferencing requests among the remaining $N - B_1 - B_2$ idle conferees must now be satisfied within the still smaller 2^r disjoint r-D meshes, each with $((L - 1)/2^2)^r$ nodes. Thus,

$$S_2 \geq N - B_1 - B_2,$$

where

$$S_2 \leq \frac{S_1}{2^r} \leq \frac{M}{2^r}\frac{1}{2^r} \leq \frac{M}{(2^2)^r}.$$

If we repeat the above conference request and hyperplane allocation process k times for incoming conference requests of appropriate sizes, we would then have $p^k = 2^{kr}$ disconnected submeshes each with S_k nodes, where B_k is the number of conferees assigned nodes in the k-th step (which is the same number of nodes in the hyperplanes used in the conference assignment in this step). Also, since the overall r-D mesh is strictly nonblocking,

$$S_k \geq F^k = N - B^1 - B^2 - \cdots - B^k = N - \sum_{i=1}^{i} B^i,$$

where

$$S_k \leq \frac{M}{(2^k)^r}$$

$$\begin{aligned} B_k \leq p^{k-1}B(L/2^{k-1}) &= (2^r)^{k-1}((L/2^{k-1})^r \\ &\quad - (L/2^{k-1} - 1)^r) = L^r - (L - 2^{k-1})^r \end{aligned}$$

$$B_i \leq L^r - (L - 2^{i-1})^r, \quad 1 \leq i \leq k.$$

Thus, for all k we need

$$\begin{aligned} \frac{M}{(2^k)^r} &\geq N - \sum_{i=1}^{k} 2^{i-1} r M^{(r-1)/r} \\ &= \mathrm{N} - \sum_{i=1}^{k} L^r - (L - 2^{i-1})^r \\ &= \mathrm{N} - (kL^r - \sum_{i=1}^{k} (L - 2^{i-1})^r) \end{aligned}$$

or

$$\frac{M}{(2^k)^r} + (kL^r - \sum_{i=1}^{k} (L - 2^{i-1})^r) \geq N.$$

Consider

$$f(k) = \frac{M}{(2^k)^r} + (kL^r - \sum_{i=1}^{k} (L - 2^{i-1})^r),$$

$$f(k) \le \frac{M}{(2^k)^r} + kL^r - \sum_{i=1}^{k} (L^k - rL^{r-1}2^{i-1})$$

$$\le \frac{M}{(2^k)^r} + rL^{r-1}(2^k - 1) \le \frac{M}{(2^k)^r}$$

$$+ rL^{r-1}2^k = \frac{M}{(2^k)^r} + rM^{(r-1)/r}2^k$$

$$= \mathrm{M}^{(r-1)/r}\left(\frac{M^{1/r}}{(2^k)^r} + r2^k\right) = g(k).$$

In the above, inequality $(a - b)^n \ge a^n - na^{n-1}b$ is used as it holds for any positive integer n when $a \ge b \ge 0$, by Lemma 2.

$g(k)$ achieves its minimum when $M^{1/r} = (2^k)^{r+1}$, or $2^k = M^{1/[r(r+1)]}$, that is,

$$\min g(k) = M^{(r-1)/r}\left(\frac{M^{1/r}}{M^{1/[r(r+1)]})^r} + rM^{1/[r(r+1)]}\right)$$

$$= (r + 1)M^{r/(r+1)}.$$

However, since for each k we must have $g(k) \ge f(k) \ge N$, we have

$$(r + 1)M^{r/(r+1)} \ge N$$

or

$$M \ge \left(\frac{N}{r+1}\right)^{(r+1)/r} = \Omega\left(\left(\frac{N}{r}\right)^{(r+1)/r}\right). \qquad \blacksquare$$

Before considering sufficient conditions for r-dimensional meshes, we can consider the necessary condition for the case of an r-dimensional torus. An r-D torus is similar to an r-D mesh, except that every node now has an identical degree of $2r$, since the special situation of a ''border'' has been eliminated in the r-D torus. Thus, the r-D torus has more connectivity than that of a corresponding r-D mesh.

It is not difficult to see that our above arguments can be extended to this case. Besides the r hyperplanes required to separate the r-D mesh in Lemma 1, another r hyperplanes ($x_1 = 1$, $x_2 = 1$, . . . , $x_r = 1$) would be needed to partition the torus into 2^r submeshes. Thus, for the first step, we need twice as many as nodes to separate the torus into smaller meshes. After doing so, all the other arguments remain the same. Thus, essentially, the same reasoning as in the proof of Theorem 1 can be used to obtain the following theorem:

Theorem 2. *$M = \Omega((N/r)^{(r+1)/r})$ nodes are necessary for an r-D torus to support nonblocking conferencing for N conferees.*

3.2. Sufficient Conditions on *M* for a Strictly Nonblocking *r*-Dimensional Mesh Serving *N* Conferees

In this section, we determine the number of nodes, M, for an r-D mesh sufficient to support strictly nonblocking N-node conferencing. To do this, we first determine the minimum number of boundary edges (or links) of a connected component of nodes of a specified size which can separate that component from all others. In general, consider component C of graph $G(V, E)$. C is a subset of nodes in the graph, that is, $C \subset V$. The *edge boundary* $BE(C)$ of C is $BE(C) = \{(v, w) : (v, w) \in E(G), v \in C, w \notin C\}$, and each edge $e \in BE(C)$ is called *edge boundary* for component C. The *node boundary* $BN(C)$ is $BN(C) = \{w : (v, w) \in E, v \in C, w \notin C, w \in V\}$, and each node $v \in BN(C)$ is called a *boundary node* for component C.

For a graph, a lower bound can be determined for the ratio of the number of boundary edges (nodes) versus the number of nodes in the component. The *edge* (*node*) *isoperimetric ratio* [4] $\lambda_e(C)(\lambda_v(C))$ of a component C of certain size k is

$$\lambda_e(C) = \min_{|C|=k} \frac{|BE(C)|}{|C|}$$

$$\lambda_v(C) = \min_{|C|=k} \frac{|BN(C)|}{|C|}.$$

For some graphs, this ratio is known to be lower-bounded by a constant which does not depend on the component size k. For example, expanders and superconcentrators are of such classes of graphs [1, 2]. However, for other graphs, the isoperimetric ratio depends on the component size k. This property is particularly useful to us, as given a component set of nodes of certain size, we can determine as a lower bound the number of edges necessary to separate the component from other nodes of the graph. In other words, if a fixed upper bound is known on the number of edges that can be used to separate a component, the size of the component cannot be arbitrarily small, in the sense that this component size thus has a lower bound depending on the number of boundary edges used. This observation is the basis of our proof of the theorem to obtain an upper bound on M for the strictly nonblocking r-D mesh.

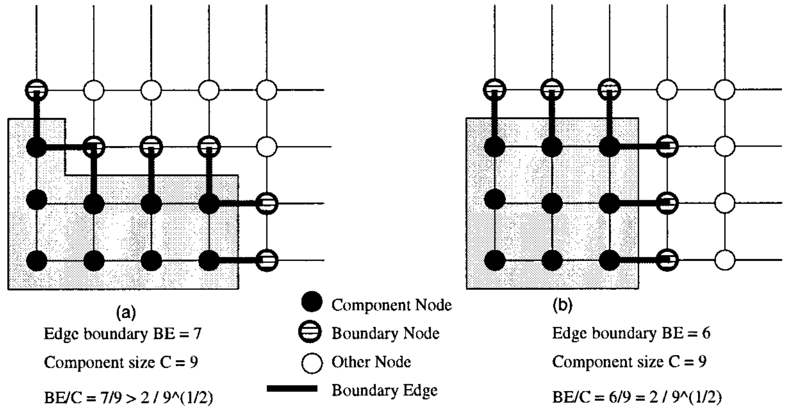


Fig. 6. Isoperimetric number: 2-D mesh.

As an example of the edge isoperimetric ratio for meshes, consider the simple case of a 2-D mesh. It can be shown that the edge isoperimetric ratio for component C in a 2-D mesh is $\lambda_e(C) = 2/|C|^{1/2}$, that is, for a component C of size $|C|$, any edge boundary set BE separating C from other components must satisfy the following inequality: $|BE|/|C| \geq \lambda_e(C) = 2/|C|^{1/2}$, where equality holds when the component is a 2-D square. A rather simple (and intuitive) argument for this bound is that, when component C is a rectangle with sides of size a and b, at least $a + b$ edges are needed to separate two sides from other components. This is necessary since the rectangle will have at least two sides adjacent to other components if it is located on a border. Thus, $|BE| \geq a + b$. We also have $|C| = ab$ for a rectangle, so $|BE|/|C| = (a + b)/(ab) \geq (2\sqrt{ab})/(ab) = 2/(\sqrt{ab}) = 2/\sqrt{|C|}$, where the equality holds when $a = b$, that is, when C is a square.

When C is not a rectangle, let R denote the rectangle with the smallest number of nodes such that C is enclosed by R in the sense that all nodes in C are contained within the rectangle R. Suppose that R has sides of size a and b. Again, at least $a + b$ edges are needed to separate C from other components, while now the component size of C is not larger than that of the enclosing rectangle R. In this case, $|BE| \geq a + b$ and $|C| \leq ab$. So, $BE \geq 2\sqrt{ab} \geq 2\sqrt{|C|}$. Thus, $|BE|/|C| \geq 2/\sqrt{|C|}$.

Figure 6 illustrates two configurations for the 2-D mesh. Both these configurations contain a component C of size 9. The (a) configuration uses $|BE| = 7$ edges to separate it from other components, while the (b) configuration uses only $|BE| = 6$ edges, the smallest possible number since the component is a 3×3 square. In both cases, the ratio of $|BE|$ and $|C|$ is no less than $2/\sqrt{|C|} = 2/\sqrt{9} = \frac{2}{3}$, as determined above.

From the above discussions, it is thus interesting to extend the edge boundary result for a 2-D mesh to more general cases such as r-D meshes. For a higher-dimensional mesh, it is known that the isoperimetric ratio depends on the component size. The following theorem by Bollobas and Leader [4] presents a key relationship between the size of a component and the size of its edge boundary.

Theorem 3 [Bollobas and Leader, 1991]. *Let A be a subset of connected nodes in an r-D mesh $[L]^r$ with $A \leq L^r/2$. Then,*

$$|BE(A)| \geq \min\{|A|^{1-1/k} k L^{(r/k)-1} : k = 1, 2, \ldots, r\}.$$

We can derive the isoperimetric ratio from this theorem. Consider an r-D mesh with a *fixed dimension*, that is, r is a constant. We have the following theorem on isoperimetric ratio of any component A with size no bigger than certain constant fraction of the mesh size $L^r = M$ when N is sufficiently large, that is, when $M/|A| \geq e^r$ for big N.

Theorem 4. *Given connected components in an r-D mesh with size M, consider component C_i of size $|C_i|$ such that $M/|C_i| \geq e^r$. The edge boundary BE_i used to separate C_i from other components has size $|BE_i|$ satisfying the following inequality:*

$$\frac{|BE_i|}{|C_i|} \geq \lambda_e(C_i) = \frac{r}{|C_i|^{1/r}}.$$

Proof. From Theorem 3, for an r-D mesh with size M and length $L = M^{1/r}$, we have

$$|BE_i| \geq \min\{|C_i|^{1-1/k} k L^{(r/k)-1} : k = 1, 2, \ldots, r\}.$$

Thus,

$$\frac{|BE_i|}{|C_i|} \geq \min k \left(\frac{L^r}{|C_i|} \right)^{1/k} \frac{1}{L} = \min k \left(\frac{M}{|C_i|} \right)^{1/k} \frac{1}{L}.$$

Consider $f(k) = kd^{1/k}$. Its derivative $f'(x) = d^{1/k}[1 - (\ln d)/k]$. For $k = 1, 2, \ldots, r$, if $\forall k$, $1 - (\ln d)/k \le 0$, that is, $1 - (\ln d)/r \le 0$, or $d \ge e^r$, then function $kd^{1/k}$ reaches a minimum at $k = r$.

Thus, for C_i with $M/|C_i| \ge e^r$, we have

$$\frac{|BE_i|}{|C_i|} \ge r\left(\frac{L^r}{|C_i|}\right)^{1/r}\frac{1}{L} = \frac{r}{|C_i|^{1/r}}. \qquad \blacksquare$$

Now we consider the sufficient condition on M for an r-D mesh. The basic idea to obtain the upper bound is to establish a lower bound on the maximum-size idle node component (the one with the largest size) such that we can always find a component to allocate to the remaining conferees. Since at most N nodes are allocated to conferees, we only need to consider the case where all components have size less than N. This fact allows the use of the above theorem to determine the maximum component size, since, for any $|C_i|$, we have that $M/|C_i| \ge M/N$.

Theorem 5. *An r-D mesh with $M = O(N^{(r+1)/r})$ nodes can support strictly nonblocking conferences for N conferees, where dimension r is fixed.*

Proof. The basic idea used to find the upper bound is to consider any configuration where a certain number of conferees (less than N, of course) are allocated arbitrarily to nodes. We want to establish a lower bound on the size of the maximum-connected idle component among all those separated by the allocated conferee nodes. If the size of this maximum component is sufficiently large as indicated by its lower bound for any number of allocated nodes, we can guarantee the existence of a component to allocate to any future conference request, thus providing a strictly nonblocking conferencing capability.

Consider an r-D mesh with M nodes which is to support strictly nonblocking conferencing applications. Denote X as the set of nodes already allocated to conferees, and denote C as the set of components of idle nodes, which will be referred to as *idle components.* Without loss of generality, suppose that C consists of k ($k \ge 1$)-connected components, separated by nodes in X. So, $C = \{C_1, C_2, \ldots, C_k\}$. Since these components are disjoint, we have $|\cup C_i| = \Sigma\, |C_i|$. We also have $M = |X| + |\cup C_i| = |X| + \Sigma\, |C_i|$. Note that all nodes in $\cup C_i$ are idle (unallocated nodes), and each node v in X falls into one of two categories:

- A nonboundary allocated node: Node v is connected only to nodes in X itself.
- A boundary allocated node: Node v is connected to some idle nodes.

Each connected idle component is separated from other idle components by a set of boundary edges. Denote such an edge boundary set as BE_i for component C_i if it is composed of edges with one end in X and the other end in C_i. The edge boundary BE_i separates C_i from other components.

Figure 7 illustrates a conferencing configuration within a 9×9 two-D mesh. Nodes are numbered from (1, 1) to (9, 9). Here, boundary nodes are drawn as solid disks, while idle nodes are shown as empty disks. Nonboundary conferee nodes are drawn as solid disks with an outside circle. Boundary edges are drawn as heavy lines. In this configuration, we have two conferences shown established. The node set $Conf_1$ representing conference 1 has 14 conferees, where 13 of these are boundary nodes and 1 node ((5, 3)) is a nonboundary node. The node set representing conference 2 ($Conf_2$) has six nodes, and all are boundary nodes. We list the nodes in $Conf_2$ as $Conf_2 = \{(3, 6), (4, 6), (4, 7), (5, 7), (4, 8), (4, 9)\}$. Here, the set X of assigned nodes is $X = Conf_1 \cup Conf_2$.

There are five idle components, namely, C_1, C_2, C_3, C_4, and C_5. Components C_1 and C_2 are separated by boundary edges connected to boundary nodes in $Conf_1$. The idle components C_3 and C_4 are separated by boundary edges connected to $Conf_2$. Component C_5, however, is separated from other components by boundary edges connected to $Conf_1$ and $Conf_2$. We see that $C_5 = \{(3, 4), (4, 4)\}$, and the edge boundary set for C_5 is $BE_5 = \{\langle(2, 4) \leftrightarrow (3, 4)\rangle, \langle(4, 4) \leftrightarrow (5, 4)\rangle, \langle(3, 3) \leftrightarrow (3, 4)\rangle, \langle(4, 3) \leftrightarrow (4, 4)\rangle, \langle(3, 4) \leftrightarrow (3, 5)\rangle, \langle(4, 4) \leftrightarrow (4, 5)\rangle\}$. The first four edges listed in BE_5 are related to boundary nodes in $Conf_1$, and the last two edges are related to boundary nodes in $Conf_2$.

Since each boundary node must be an allocated node, we have $\cup_{i=1}^{k} B_i \subseteq X$. Each $v \in \cup_{i=1}^{k} B_i$ can be incident to at most $2r$ boundary edges, while a boundary edge is always connected to a boundary node. Thus, we have

$$|\cup_{i=1}^{k} B_i| \le |X|$$

and

$$\sum_{i=1}^{k} |BE_i| \le 2r|\cup_{i=1}^{k} B_i|.$$

Thus,

$$|X| \ge \frac{1}{2r}\sum_{i=1}^{k} |BE_i| \tag{1}$$

and the number of idle nodes $M - |X|$ in the mesh is

$$M - |X| = \sum_{i=1}^{k} |C_i|. \tag{2}$$

For each component C_i, consider its edge boundary

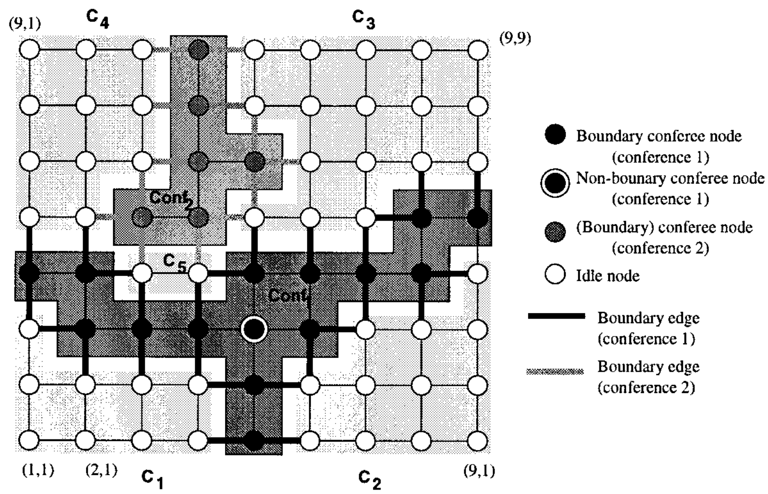


Fig. 7. Nodes separation in 2-D mesh.

set BE_i. For a strictly nonblocking r-D mesh with size M, we know from the previous subsection that $M = \Omega[(1/r)N^{(r+1)/r}]$. If there exists C_i such that $|C_i| \geq N$, then we have a component with sufficient size to realize any conference of no more than N conferees. Obviously, the mesh can support any feasible request in this case. Hence, we only need to consider the case when $|C_i| \leq N$. In this case, we have $M/|C_i| \geq (N/(r+1))^{(r+1)/r})/N = N^{1/r}/(r+1)^{(r+1)/r}$. When N is sufficiently large [$N \geq c_0$, where $c_0 = ((r+1)^{(r+1)/r}e^r)^r$ is a constant], $N^{1/r}/(r+1)^{(r+1)/r} \geq e^r$, so $M/|C_i| \geq e^r$. By Theorem 4, for such C_i, we have

$$\frac{|BE_i|}{C_i} \geq \frac{r}{|C_i|^{1/r}}$$

or

$$\frac{|BE_i|}{|C_i|} \geq \frac{r}{|C_{max}|^{1/r}}, \tag{3}$$

where $C_{max} = C_j$ and $|C_j| = \max_{i=1}^{k} |C_i|$.

Note that, in general, for $a, b, c, d, e > 0$, if $a/b \geq e$ and $c/d \geq e$, then $(a+c)/(b+d) \geq e$. Thus, by Eqs. (1)–(3), we have

$$\frac{|X|}{M - |X|} \geq \frac{\frac{1}{2r}\sum_{i=1}^{k} |B_i|}{\sum_{i=1}^{k} |C_i|} \geq \frac{1}{2r} \frac{\sum_{i=1}^{k} |B_i|}{\sum_{i=1}^{k} |C_i|}$$

$$\geq \frac{1}{2r} \frac{r}{|C_{max}|^{1/r}} = \frac{1}{2|C_{max}|^{1/r}},$$

where

$$|C_{max}| \geq \left(\frac{M - |X|}{2|X|}\right)^r. \tag{4}$$

For every configuration, if the remaining largest idle component C_{max} is large enough to support any conference among the remaining idle nodes, the mesh is strictly nonblocking, that is, we need to guarantee that

$$|C_{max}| \geq N - |X|, \forall |X| \leq N. \tag{5}$$

The reason why condition (5) guarantees a strictly nonblocking mesh is as follows: For any set X_s of assigned conferees, regardless of the groupings in which they arrived and the allocation of nodes to these groups of conferences, if inequality (5) holds for all X, the remaining largest idle component would always be of sufficient size such that any next conference request (with at most $N - |X_s|$ conferees) can be satisfied. Once X_s conferees have been assigned to nodes, there are two possibilities for the next event: namely, either a new conference request would arrive or an existing conference would depart. Consider the first case when the next event is a conference request of size $|X_r|$. If inequality (5) holds, we can always find an idle component to allocate to the conferees in this request. Any such assignment, however, generates a new configuration where $|X'|$ nodes have been allocated, where X' is the union of X_r and X_s, $X' = X_r \cup X_s$. The second case is when an existing conference of size $|X_r|$ would depart. Here, the allocated node set X'' resulting from the conference departure becomes smaller as the allocated nodes of X_r are released. Thus, $X'' = X_s - X_r$. In both cases, a new assigned node set X (either X' for first case or X'' for second case) is obtained. Since inequality (5) holds for all possible X, in this new configuration, a component with a size large enough to realize any next request is guaranteed to exist. Thus, regardless of the current configuration, we can guarantee the existence of idle components to allocate to conferences.

The above condition (5) is satisfied when the following holds:

$$\min_X \left(\frac{M - |X|}{2|X|} \right)^r \geq N - |X|,$$

and from inequality (4), it is sufficient that

$$\min_X \left(\frac{(M - |X|)}{2|X|} \right)^r + |X| \geq N. \qquad (6)$$

The above inequality (6) holds when the following is satisfied:

$$\min_X f(|X|) \geq N, \quad f(|X|) = \left(\frac{M - N}{2|X|} \right)^r + |X|.$$

The derivative $f'(x)$ of function $f(x)$ is $f'(x) = 1 - [(M - N)/2]^r (r/x^{r+1})$, so $f(|X|)$ reaches its minimum when $|X|^{r+1} = r[(M - N)/2]^r$, or $|X| = (r[(M - N)/2]^r)^{1/(r+1)}$, and

$$\min_X f(X) = \left(\frac{M - N}{2\left(r \left(\frac{M - N}{2} \right)^r \right)^{1/(r+1)}} \right)^r + \left(r \left(\frac{M - N}{2} \right)^r \right)^{1/(r+1)}$$

$$= \left(r^{1/(r+1)} + \frac{1}{r^{r/(r+1)}} \right) \left(\frac{M - N}{2} \right)^{r/(r+1)}.$$

Thus, we need

$$(r^{1/(r+1)} + r^{-r/(r+1)}) \left(\frac{M - N}{2} \right)^{r/(r+1)} \geq N$$

or

$$M \geq 2\left(\left(\frac{N}{r^{1/(r+1)} + r^{-r/(r+1)}} \right)^{(r+1)/r} \right) + N$$

$$= 2\left(\left(\frac{1}{r^{r/(r+1)} + r^{-[(r^2/(r+1)]}} \right)^{(r+1)/r} N^{(r+1)/r} \right) + N.$$

Note that for $r > 1$ we have

$$\left(\frac{1}{r^{1/(r+1)} + r^{-[r/(r+1)]}} \right)^{(r+1)/r} \leq \left(\frac{1}{r^{1/(r+1)}} \right)^{r+1/r} = \frac{1}{r^{1/r}},$$

Thus, it suffices that

$$M \geq 2\left(\frac{1}{r^{1/r}} N^{(r+1)/r} + N \right) = O\left(\frac{1}{r^{1/r}} N^{(r+1)/r} + N \right).$$

Since r is constant, $N = o(N^{(r+1)/r})$, so

$$M = O(N^{(r+1)/r}). \qquad \blacksquare$$

Note that the above theorem just gives an asymptotic upper bound for the size of r-D mesh to support strictly nonblocking conferencing. For a given number N of conferees, we can solve inequality (6) for all X where $1 \leq |X| \leq N$. This can be accomplished by checking whether an M value satisfies the system of inequalities, and the smallest such M would suffice to guarantee a strictly nonblocking mesh.

Thus, when the mesh dimension r is a fixed integer, we know from Theorem 1 that the necessary condition is $\Omega(((N/r)^{(r+1)/r}) = \Omega(N^{(r+1)/r})$. Together with Theorem 5, we have the following result:

Theorem 6. *For an r-D mesh with a fixed dimension r, $M = \Theta(N^{(r+1)/r})$ nodes are necessary and sufficient to support a strictly nonblocking conference application for N conferees.*

3.3. Variable Mesh Dimensionality

The above discussions considered conditions for an r-D mesh using the assumption that the dimensionality r is a fixed constant. We will now consider the implications of loosening this assumption such that r can be varied, that is, it will only required that $L^r = M$ is satisfied for r-D mesh with M nodes and length L.

For this case, we have the following theorem on the edge isoperimetric ratio for r-D mesh with an arbitrary r value. Note that this lower bound is smaller compared to the case when r is a fixed constant:

Theorem 7. *For any connected component C_i in an r-D mesh with size M, the edge boundary BE_i used to separate C_i from others has size $|BE_i|$ satisfying the following inequality:*

$$\frac{|BE_i|}{|C_i|} \geq \frac{1}{|C_i|^{1/r}}.$$

Proof. From Theorem 3, for an r-D mesh with M nodes and length $L = M^{1/r}$, we have

$$|BE_i| \geq \min\{|C_i|^{1-1/k} kL^{(r/k)-1} : k = 1, 2, \ldots, r\},$$

and

$$\frac{BE_i}{C_i} \geq \min k|C_i|^{-1/k} L^{(r/k)-1} \frac{|C_i|^{1/r}}{|C_i|^{1/r}}$$
$$= \min k|C_i|^{(k-r)/rk} L^{(r-k)/k} \frac{1}{|C_i|^{1/r}}$$
$$= \min k\left(\frac{L^r}{|C_i|}\right)^{(r-k)/rk} \frac{1}{|C_i|^{1/r}}.$$

Since $M \geq |C_i|$, we have $L^r/|C_i| \geq 1$. For $k = 1, 2, \ldots, r$, we have $r \geq k \geq 1$, or $(r-k)/(rk) \geq 0$; thus,

$$k\left(\frac{L^r}{|C_i|}\right)^{(r-k)/rk} \geq 1.$$

Finally,

$$\min k\left(\frac{L^r}{|C_i|}\right)^{(r-k)/rk} \frac{1}{|C_i|^{1/r}} \geq \frac{1}{|C_i|^{1/r}}.$$

Hence, we have the desired result. ■

Using a very similar argument as in Theorem 5, we can prove the following more general result for an r-D mesh. Here, no constraint is placed on the dimensionality r:

Theorem 8. *An r-D mesh with $M = O(r^{(r-1)/r} N^{(r+1)/r})$ nodes can support a strictly nonblocking conference for N conferees.*

Proof. The proof is similar to the fixed r case. We use the isoperimetric ratio for the general high-dimensional mesh determined by Theorem 7.

For general r, inequality (3) in the proof of Theorem 5 becomes

$$\frac{|BE_i|}{|C_i|} \geq \frac{1}{|C_{max}|^{1/r}}. \tag{7}$$

Similar to inequality (4), for general r, we have the following bound for the size of the largest idle component:

$$|C_{max}| \geq \left(\frac{M - |X|}{2r|X|}\right)^r. \tag{8}$$

It suffices to have

$$\min_X \left(\frac{(M - |X|)}{2r|X|}\right)^r + |X| \geq N. \tag{9}$$

The above is satisfied if

$$\min_X f(|X|) \geq N, \quad f(|X|) = \left(\frac{M-N}{2r|X|}\right)^r + |X|.$$

$f(|X|)$ reaches minimum when

$$|X| = \left(r\left(\frac{M-N}{2r}\right)^r\right)^{1/(r+1)},$$

and

$$\min_X f(|X|) = \left(r^{1/(r+1)} + \frac{1}{r^{r/(r+1)}}\right)\left(\frac{M-N}{2r}\right)^{r/(r+1)}.$$

We need

$$(r^{1/(r+1)} + r^{-[r/(r+1)]})\left(\frac{M-N}{2r}\right)^{r/(r+1)} \geq N.$$

Thus, it suffices that

$$M \geq 2r\left(\frac{1}{r^{1/r}} N^{(r+1)/r}\right) + N = O(r^{(r-1)/r} N^{(r+1)/r}). \quad ■$$

Note that this upper bound can be tighter than the bound for the fixed dimensionality case if an appropriate r value is selected. Thus, Theorem 5 can be viewed as a special case of this more general theorem.

We are interested in a nonblocking r-D mesh which uses the fewest number of nodes to support N conferees. We get the following bound when using higher-dimensional meshes:

Corollary 1. *There exists r-D meshes for N conferees which use as few as $M = O(N \log N)$ nodes. This bound is reached when the mesh dimension r is $\Theta(\log N)$.*

Proof. By the above theorem, an r-D mesh for N conferees is strictly nonblocking if it has $O(rN^{(r+1)/r}$ nodes for variable r values.

Let $f(r) = rN^{(r+1)/r}$. Its derivative $f'(r)$ is

$$f'(r) = rN^{(r+1)/r} + rN^{(r+1)/r}\ln N\left(-\frac{1}{r^2}\right)$$

$$= rN^{(r+1)/r}\left(1 - \frac{\ln N}{r}\right).$$

Thus, $f(r)$ reaches its minimum when $(\ln N)/r = 1$ or $r = \ln N$:

$$\min f(r) = N^{(\ln N+1)/\ln N}\ln N = O(N \log N).$$

In fact, when $r = c \log N = \Theta(\log N)$, where c is any constant, we have

$$f(r) = (c \log N)^2 N^{(c\log N+1)/(c\log N)} = O(N \log N). \quad \blacksquare$$

In the following, we consider the sufficient conditions for the nonblocking r-D torus.

3.4. Results for the *r*-Dimensional Torus

Now consider the case for r-D torus. The torus and the mesh have very similar structures, while the torus has slightly more edges. Bollobas and Leader [4] provided the following result for the r-dimensional torus:

Lemma 3 [Bollobás and Leader, 1991]. *Let A be a subset of nodes of an r-D torus $[L]^r$ with $A \le L^r/2$. Then,*

$$|BE(A)| \ge \min\{2|A|^{1-1/k}kL^{(r/k)-1} : k = 1, 2, \ldots, r\}.$$

Note that only a constant factor of 2 is added for the lower bound when compared to the bound for the mesh. A similar edge isoperimetric ratio result for the r-D torus is as follows:

Lemma 4. *For connected components in an r-D torus with size M, consider component C_i of size $|C_i|$ such that $M/|C_i| \ge e^r$. The edge boundary BE_i used to separate C_i from others has size $|BE_i|$, satisfying the following inequality:*

$$\frac{|BE_i|}{|C_i|} \ge \lambda_e(C_i) = \frac{2r}{|C_i|^{1/r}}.$$

In general, for any component C_i, the following inequality is satisfied:

$$\frac{|BE_i|}{|C_i|} \ge \lambda_e(C_i) = \frac{2}{|C_i|^{1/r}}.$$

Using similar reasoning to that in the proof of Theorem 5, we have the following theorem:

Theorem 9. *An r-D torus with $M = O(N^{(r+1)/r})$ nodes can support strictly nonblocking conferencing for N conferees, when dimension r is fixed. For variable dimension r, $M = O(r^{(r-1)/r}N^{(r+1)/r})$ nodes are sufficient for a strictly nonblocking r-D torus conference network for N conferees.*

When combined with Theorem 2, we have

Theorem 10. *For an r-D torus with a fixed dimension r, $M = \Theta(N^{(r+1)/r})$ nodes are necessary and sufficient to support a strictly nonblocking conference network.*

4. ALGORITHM TO ALLOCATE NODES TO CONFERENCES

In this section, we consider an algorithm for allocating nodes to conference requests for strictly nonblocking r-D meshes.

For any existing configuration, a strictly nonblocking mesh guarantees the existence of a component with sufficient size to realize any valid request, that is, the idle component with the maximum size can always be used to accommodate the conference request. A conferee assignment algorithm for allocation of nodes to the conferees would use the largest component to realize the request. Before doing that, however, we need to determine the current set of idle components and identify the largest component.

Finding connected components in a graph with nodes N and edges E can be accomplished using a variation of depth-first-search techniques. Such an algorithm, called the *conference assignment algorithm,* is depicted in Figure 8 as follows:

This algorithm would consider each edge at most once since each of its end nodes would check the other end node, and this checking is performed at most once. For the first step to determine the components, the running time is $O(V + E)$. To allocate the nodes in the largest component to the requested conference, we also could use the same depth-first search procedure. The only difference is that the procedure will stop when k nodes are found. As a result, the algorithm runs in time $O(V + E)$ for each on-line conference request.

Consider an r-D mesh for N-conferees, when dimension r is fixed. There are $V = O(N^{(r+1)/r})$ nodes and $E = O(N^{(r+1)/r})$ edges. Thus, the running time for this case is $O(N^{(r+1)/r})$. If r is a variable, $V = O(rN^{(r+1)/r})$ and $E = O(r^2N^{(r+)/r})$. Thus, the running time is $O(r^2N^{(r+1)/r})$.

The time needed for obtaining the current idle compo-

Conference Assignment Algorithm

Input: Graph $G = (V, E)$, conference request from k conferees;

1. For each non-allocated node v in G

 1.1 If v is marked, continue;

 1.2 If v is not marked, we perform a depth-first-search starting from v to find the component containing v.

 All nodes found in the same component are marked with the component number. A node is in the component when it is not assigned or marked, and there is a feasible (unused) edge reaching it from a previously reached node in the same idle component.

2. Allocate k nodes in the largest component to the request;

Fig. 8. Conference assignment algorithim.

nents would be reduced if we update information regarding these components after each on-line request assignment. For example, for a requested allocation, either the largest component exactly accommodating the request (i.e., all nodes are used) or a subset of the nodes in the component are used. In each case, the other remaining components are not affected. So, we only need to determine the smaller components within the largest component resulting from the node allocation. When a conference departs, we can check the neighboring components of the released nodes and merge them. However, these improvements in the algorithm running time do not reduce the asymptotic bound for the algorithm, since in the worst case, the edges that need to be checked could be as many as all the edges in the graph. Also, most importantly, the time needed for node allocation could take $O(V + E)$ if a search algorithm is used to find a tree of size k.

To improve on the running time needed for on-line requests, we can take advantage of some particular constructions of meshes, which could use more nodes than the smallest number of nodes possible. Consider an example for a 2-D mesh. We can use the following construction which requires $5N^{3/2}$ nodes: In this construction, a total $5N$ submeshes, each of size $N^{1/2}$, are used. The first N of such submeshes are used for a request size no larger than $N^{1/2}$, as no more than N requests are present at any instant, and the remaining $4N$ submeshes are used for realizing requests larger than $N^{1/2}$. Note that in this latter case any request with size $k_i > N^{1/2}$ is realized by $\lceil k_i/N^{1/2} \rceil$ submeshes. The total number of such submeshes needed is $\Sigma \lceil k_i/N^{1/2} \rceil \leq 2N^{1/2}$; thus, $(2N^{1/2})^2 = 4N$ submeshes suffice to realize conferences in any grouping for these $2N^{1/2}$ submeshes.

Since the upper bound on the number of nodes for a strictly nonblocking capability for a 2-D mesh is approximately $2N^{3/2}$, this construction does not use the smallest number of nodes possible. However, using more nodes has allowed us to utilize a more efficient algorithm, since we have a predetermined way to assign submeshes to requests based on their sizes. Thus, we only need to maintain the current ''status'' for each submesh, instead of maintaining information regarding all components which could involve searching through the entire mesh. There are $5N = O(N)$ such small submeshes. Thus, we effectively reduce the information storage requirement. Also, we can find a feasible submesh using $O(N)$ time, which is an improvement over the $O(N^{3/2})$ time for the algorithm described in Figure 8.

5. CONCLUSION

In this paper, we considered r-dimensional conferencing meshes as a configurable conference server component. Using an argument employing the isoperimetric ratio of the sizes of edge and node sets in a graph, we showed that a necessary and sufficient condition such that an r-dimensional conferencing mesh of M processing elements can provide a strictly nonblocking conferencing capability among a set of N conferees is that $M = \Omega(N^{(r+)/r})$ processing elements, when r is fixed. We also give a general bound of $M = O(r^{(r-1)/r}N^{(r+1)/r})$ for a strictly nonblocking r-dimensional mesh for N conferees for variable r values.

We considered algorithms for allocating processing elements to satisfy conference requests in our configurable conferencing server design and analyzed their running times. We also gave examples of explicit mesh constructions to allow more efficient algorithms for satisfying on-line conference requests.

It is important to note that our approach to establishing the sufficient conditions on M for strictly nonblocking r-D meshes can be extended to general graph structures, such as, for example, expanders and lattices, or, indeed, to any graphs for which the isoperimetric ratios can be determined [7]. Hence, we have established a fundamental relationship between the size requirements on M for strictly nonblocking conferencing among N conferees using certain graph structures and the isoperimetric ratios for those structures.

Conference network models in which a communication link can support more than one conference at the same time are of particular interest also. Called *Multicapacity Conference Networks*, this class of conference networks possess many attractive properties [6]. Further research on these networks would shed insight on the resource requirement for satisfying conference requests [8].

One of author's research (G.M.M.) was partially supported by Compunetix Inc., Monroeville, PA [14], and the other author's research (Y.D.) was partially supported by DARPA.

REFERENCES

[1] N. Alon, Eigenvalues and expanders, Combinatorica 6 (1986), 85–98.

[2] N. Alon and V.D. Milman, λ_1, isoperemetric inequalities for graphs, and superconcentrators, J Combin Theory Ser B 38 (1995), 73–88.

[3] V.E. Beneš, Mathematical theory of connecting networks and telephone traffic, Academic Press, New York, 1965.

[4] B. Bollobás and I. Leader, Edge-isoperimetric inequalities in the grid, Combinatorica 11 (1991), 299–314.

[5] D.G. Cantor, On non-blocking switching networks, Networks 1 (1972), 367–377.

[6] Y. Du and G.M. Masson, Practical constructions for conference networks, Proc 31st Annual Conf on Information Sci and Sys, 1997.

[7] Y. Du and G.M. Masson, Isoperimetric inequalities and general theorem on sufficient conditions for strictly non-blocking conference networks, Technical Report TR-1997-14, Department of Computer Science, The Johns Hopkins University, 1997.

[8] Y. Du and G.M. Masson, Wide-sense non-blocking multi-capacity mesh conferencing servers, IEEE Trans Comp, (submitted 1998).

[9] J.F. Houlahan, L.M. Cowen, and G.M. Masson, Hypercube sandwich approach to conferencing, J Supercomput 10 (1996), 271–283.

[10] J.F. Houlahan, L.M. Cowen, and G.M. Masson, Wide-sense non-blocking hypercube sandwich conference networks, Proc 30th Ann Conf Information Sci Syst (1996), 222–227.

[11] Y. Li and G.M. Masson, Strictly non-blocking mesh conference networks, Proc 30th Ann Conf Information Sci Syst (1996), 228–233.

[12] D.R. Woodall, The Bay Restaurant—A linear storage problem, Am Math Monthly March (1974), 240–246.

[13] Y. Yang and G.M. Masson, Broadcast ring sandwich networks, IEEE Trans Comput 44 (1995), 1169–1180.

[14] J. Driscol, E. Law, M. Kancel, and G. Masson, Programmable conference modules for switchable ring arrays, U.S. Pat. 4,094,948 (1990) (assigned to Compunetix, Inc., Monroeville, PA).