

On Generalized Fat Trees *

Sabine R. Öhring Maximilian Ibel
Sajal K. Das

Department of Computer Science
University of North Texas
Denton, TX 76203-6886, USA
{oehring,ibel,das}@cs.unt.edu

Mohan J. Kumar

Department of Computer Science
Curtin University of Technology
Perth, WA 6001, Australia
kumar@cs.curtin.edu.au

Abstract

We introduce and analyze a new family of multiprocessor interconnection networks, called generalized fat trees, which include as special cases the fat trees used for the connection machine architecture CM-5, pruned butterflies, and various other fat trees proposed in the literature. The generalized fat trees provide a formal unifying concept to design and analyze a fat tree based architecture. The extended generalized fat tree network $XGFT(h; m_1, \dots, m_h; w_1, \dots, w_h)$ of height h has $\prod_{i=1}^h m_i$ leaf processors and the inner nodes serve only as switches or routers. Each non-leaf node in level i has m_i children and each non-root has w_{i+1} parent nodes. The generalized fat trees provide regularity, symmetry, recursive scalability, maximal fault-tolerance, logarithmic diameter, bisection scalability, and permit simple algorithms for fault tolerant self-routing and broadcasting. These networks are also versatile, since they can efficiently embed rings, meshes and tori, trees, pyramids and hypercubes.

Keywords: edge bisection, embedding, fat tree, interconnection network, routing.

1 Introduction

Several topologies have been proposed as interconnection networks for multicomputer systems [6]. Among these, the hypercube and the mesh topologies are two popular networks from a commercial point of view. However, although the hypercube is an efficient network because of its symmetry, regularity, logarithmic diameter, modularity and fault tolerance [14], it suffers from wirability and packing problems for VLSI implementation due to a non-constant node degree. The n -dimensional hypercube has an edge bisection of $\Theta(2^n)$ which implies a VLSI-layout area of at least $\Theta(2^{2n})$ [6]. Many problems in science and engineering such as matrix problems, multigrid methods [2, 10], and image processing algorithms have mesh-like communication patterns with constant node-degree. A mesh again has the drawback of a larger diameter and low edge bisection [5]. Therefore it is important to search for topologies which overcome the disadvantages of meshes and hypercubes, but efficiently simulate both topologies. The generalized fat tree concept allows to choose either a high edge bisection for efficient embeddings or a low edge bisection for reduced layout complexity, as desired, without

changing the number of nodes or the diameter.

Leiserson [8, 9] proposed fat trees as hardware-efficient, general-purpose interconnection networks. Several architectures including the Connection Machine CM-5 of Thinking Machines, the memory hierarchy of the KSR-1 parallel machine of Kendall Square Research [3], and Meiko supercomputer CS-2 [13, 15] are based on the fat trees. A different fat tree topology called "pruned butterfly" is proposed in [1], and other variants are informally described in [4], where the increase in channel bandwidth is modified compared to the original fat trees in [8].

In a fat tree architecture, the processing elements are located at the leaf nodes and the intermediate nodes serve as routers or switches. For example, the CM-5 Data Network is a fat tree, where each node has four children and two parents for level 0 (leaf level) upto level 2 and four parents from level 3 upwards. The fat trees have the advantage of being a recursively scalable and partitionable network with a simple routing algorithm [9]. Fat trees have low diameter and their edge bisection can be customized based on the algorithm, fault tolerance and cost requirements. In [8, 7] it has been shown that an area-universal fat tree of a given size is nearly the best routing network of that size. An extensive experimental performance evaluation of the communication capabilities of the CM-5 has been studied in [12]. The performance of random routing on fat trees was evaluated in [4].

In this paper, we generalize the concept of fat trees [8]. Our generalized fat tree $GFT(h, m, w)$ of height h consists of m^h processors in the leaf-level and routers or switching-nodes in the non-leaf levels. Each non-root has w parent nodes and each non-leaf has m children. We provide a more general definition in order to accommodate the CM-5 fat tree as a special case. This extended version of the generalized fat tree $XGFT(h, m_1, \dots, m_h, w_1, \dots, w_h)$ consists of $\prod_{i=1}^h m_i$ processors, where each node in level i , $0 \leq i \leq h-1$ has w_{i+1} parent nodes and m_i children for $1 \leq i \leq h$.

This concept provides an unifying approach to the existing variants of fat trees. Furthermore, an improved scalability of the network is achieved in two aspects. On the one hand, the system size is equal to $\prod_{i=1}^h m_i$ for arbitrary m_i and thus no more restricted to a power of m . On the other hand, we obtain some *bisection-scalability*, since the edge bisection of the generalized fat tree networks can be chosen independent of the number of nodes and its degree. A fat tree can there-

*This work is partially supported by Texas Advanced Technology Program Grant under Award No. TATP-003594031.

fore be adapted to efficiently utilize whatever edge bisection makes engineering sense in terms of cost and performance. This is impossible with meshes and hypercubes, where the edge bisection is $\min\{m, n\}$ for an $m \times n$ mesh and 2^{h-1} for a h -dimensional binary hypercube $Q(h)$. This feature is particularly important when studying the layout area of a graph, which is at least the square of the edge bisection asymptotically [17]. Another advantage of our approach is that we modeled the fat tree (unlike in [9]) with routing switches having a fixed degree, which allows to perform communication primitives faster than in a fat tree with switches having an exponentially growing degree.

The rest of this paper is organized as follows. Section 2 gives the definitions and notations used in the following sections. Section 3 defines the (extended) generalized fat tree networks. In Section 4, topological properties such as node degree, diameter, average distance, cost, modularity (recursive scalability) and symmetry, and implementation aspects such as edge bisection are discussed and compared with those of other popular networks. Simple communication algorithms for routing and broadcasting which tolerate upto (degree -1) node- and link-failures are presented in Section 5. The versatility of the proposed networks is shown by designing efficient embeddings of rings, tori, complete binary trees, hypercubes and pyramids in Section 6. Section 7 concludes the paper.

2 Definitions and Notations

An interconnection network can be modeled as an undirected graph $G = (V, E)$, where V is the node-set representing the processors and E the edge-set representing the communication links among the processors.

A *cartesian product* $G \times H = (V, E)$ of two graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$ is defined by $V = V_G \times V_H$ and $E = \{(x, y), (x', y') \mid x, x' \in V_G, y, y' \in V_H, x = x' \text{ and } \{y, y'\} \in E_H \text{ or } \{x, x'\} \in E_G \text{ and } y = y'\}$. A two-dimensional ($m \times n$)-mesh $M(m, n) = (V, E)$ has the node-set $V = \{(i, j) \mid 0 \leq i \leq m-1, 0 \leq j \leq n-1\}$ and there is an edge between two mesh nodes x and y if $|x - y| = 1$. A mesh $M(m, n)$ can be described as a cartesian product $L(m) \times L(n)$ where $L(m)$ is a linear array of length m . A *torus* $T(m, n)$ is defined as $R(m) \times R(n)$ where $R(m)$ is a ring of length m . An n -dimensional *binary hypercube* $Q(n) = (V_n, E_n) = Q(n-1) \times Q(1)$ is given by the node-set $V_n = \mathbf{Z}_2^n$, the set of binary strings of length n and there exists an edge $\{x, y\} \in E_n$ between two nodes x and y in $Q(n)$ if x and y differ in exactly one bit. The *generalized hypercube* GQ_n^b of base b and dimension n is analogously defined with the node-set $V = \mathbf{Z}_b^n$. Two nodes x and y , both strings of length n , are adjacent if they differ in at most one symbol.

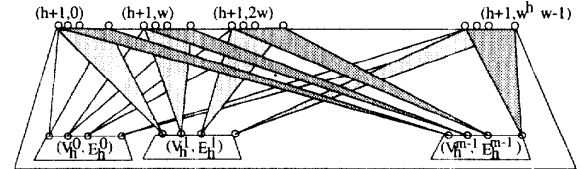
The *distance* between two nodes in an interconnection topology is the length of the shortest path between them. Two paths are said to be *node-disjoint* if they have no common nodes, except for the source and destination. The *diameter*, d , of a network is the maximum distance among all node-pairs. It is a measure of the worst-case communication delay. The *degree* of a node is the number of links incident to it. For a regular network of node-degree δ , its *cost* is defined as $C = d * \delta$. An important parameter for the VLSI-layout

area of a network G is its *edge bisection* [17], which is defined as the minimum number of edges whose removal splits G in two equal-sized disconnected graphs. The *node (edge)-connectivity* is the number of nodes (edges) whose removal results in a disconnected network. It is a measure of fault-tolerance of the network. A graph is *f-node (edge)-faulttolerant* if it remains connected after the removal of upto arbitrary f nodes (edges).

3 Generalized Fat Trees

Definition 1 : A generalized m -ary fat tree $GFT(h, m, w) = (V_h, E_h)$ of height h and edge bisection increasing factor w is an undirected graph.

Informally, $GFT(h+1, m, w)$ is recursively generated from m distinct copies of $GFT(h, m, w)$, denoted as $GFT^j(h, m, w) = (V_h^j, E_h^j)$, $0 \leq j \leq m-1$, and w^{h+1} additional nodes such that each top-level node $(h, k + j \cdot w^h)$ of each $GFT^j(m, w, h)$, for $0 \leq k \leq w^h - 1$, is adjacent to w consecutive new top-level-nodes (i.e. level $h+1$ nodes), given by $(h+1, k \cdot w), \dots, (h+1, (k+1) \cdot w - 1)$. The graph $GFT^2(h, m, w)$ is also called a *sub-fat tree* of $GFT(h+1, m, w)$. This construction is sketched in Figure 1.



$GFT(h+1, m, w) = (V_{h+1}, E_{h+1})$

Figure 1: Recursive construction of $GFT(h+1, m, w)$

The node set V_h of $GFT(h, m, w)$ is given by :
 $V_h := \{(l, i) \mid 0 \leq l \leq h \wedge 0 \leq i \leq m^{h-l} w^{l-1} - 1\}$, where l is the level of a node (the leaves being at level 0) and i denotes the position of this node in level l .

Since we will use the recursive scalability of the generalized fat trees, let us give a recursive definition of both the node- and edge-set of $GFT(h+1, m, w)$:

$$- V_0 := \{(0, 0)\}, E_0 := \emptyset.$$

$$- V_{h+1} := \bigcup_{j=0}^{m-1} V_h^j \cup \{(h+1, a) \mid 0 \leq a \leq w^{h+1} - 1\} \text{ and}$$

$$- E_{h+1} := \bigcup_{j=0}^{m-1} E_h^j \cup \{(h, a), (h+1, b) \mid a \bmod w^h = \lfloor \frac{b}{w} \rfloor\}, \text{ where}$$

$$V_h^j := \{(l, a) \mid 0 \leq l \leq h \wedge j \cdot \lambda \leq a \leq (j+1) \cdot \lambda - 1\}$$

$$\text{and } E_h^j := \{(l, a), (l+1, b) \mid 0 \leq l \leq h-1 \wedge (l, a), (l, b) \in V_h^j \wedge \{(l, a-j \cdot \lambda), (l+1, b-j \cdot \lambda)\} \in E_h\}.$$

Here $\lambda = m^{h-l} w^l$ is the number of level- l nodes in each sub-fat tree $GFT^j(h, m, w)$. The edges in E_h^j are simply the edges in $GFT^j(h, m, w)$, and an edge connecting some top-level node $(h+1, a)$ with a top-level node $(h, b + j \cdot w^h)$ in V_h^j where $b \in \{0, \dots, w^h - 1\}$ implies $a \in \{w \cdot b, w \cdot b + 1, \dots, w \cdot b + w - 1\}$.

Although a fat tree is a tree in graph-theoretic terms for only $w = 1$, we will denote all nodes with an out-degree of 0 as *leaves*, nodes with an in-degree of 0 as

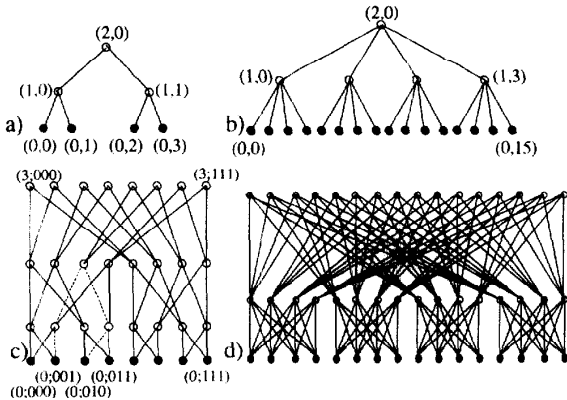


Figure 2: a) $GFT(2, 2, 1)$ b) $GFT(2, 4, 1)$ c) $GFT(3, 2, 2)$ d) $GFT(2, 4, 4)$

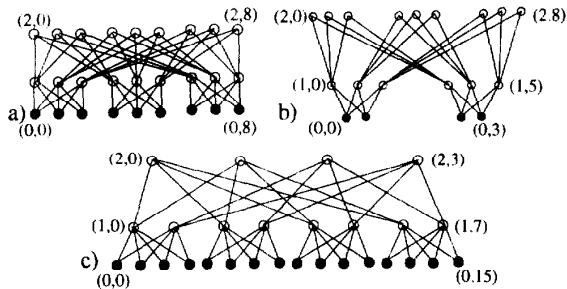


Figure 3: a) $GFT(2, 3, 3)$, b) $GFT(2, 2, 3)$ c) $GFT(2, 4, 2)$

top-level nodes (or roots) and all non-leaves as inner nodes. Nodes that are neither leaves nor roots will be called intermediate nodes.

As a special case, for $w = 1$ we obtain a complete m -ary tree of height h . Figures 2a) and b) show two examples for $m = 2$ (binary tree) and $m = 4$ (4-ary tree), both of height two. For $m = w$ we obtain a graph similar to the baseline network [6]. Figures 2c), 3a) and 2d) illustrate fat trees for $m = w = 2, 3$ and 4. It is even possible to choose $w > m$ (cf. Figure 3b). The fat tree used for the 16 node CM-5 machine corresponds to $GFT(2, 4, 2)$, shown in Figure 3c).

As a further generalization, the constants m and w can be chosen independently for each level.

Definition 2: An extended generalized fat tree $XGFT(h, m_1, \dots, m_h, w_1, \dots, w_h) = (V_h, E_h)$ of height h and $m = m_i$, $w = w_i$ at level i can be defined in the same way as $GFT(h, m, w)$, replacing m with m_1, m_2, \dots, m_h and w with w_1, w_2, \dots, w_h . For the rest of this paper, we abbreviate $XGFT(h, m_1, \dots, m_h, w_1, \dots, w_h)$ as $XGFT$.

Informally, $XGFT(h + 1, m_1, \dots, m_{h+1}, w_1, \dots, w_{h+1})$ is recursively generated from m_h distinct copies of an $XGFT(h, m_1, \dots, m_h, w_1, \dots, w_h)$, called $XGFT^j = (V_h^j, E_h^j)$, ($0 \leq j \leq m - 1$) and $w_1 \dots w_{h+1}$ additional nodes, such that each top-level node $(h, k + \lambda \cdot j)$ of each $XGFT^j$ ($0 \leq j \leq w_1 \dots w_h - 1$) is adjacent to w_{h+1} consecutive new top-level nodes $(h + 1, k \cdot w_{h+1}), \dots, (h + 1, (k + 1) \cdot w_{h+1} - 1)$, where

$\lambda = w_1 \dots w_h$ is the number of top-level nodes of each $XGFT^j$. Formally,

$$- V_h := \{(l, i) \mid 0 \leq l \leq h \wedge 0 \leq i \leq w_1 \dots w_l m_{l+1} \dots m_h - 1\},$$

$$- E_{h+1} = \bigcup_{j=0}^{m-1} E_h^j \cup \{(h, a), (h + 1, b) \mid a \bmod w_1 \dots w_h = \lfloor \frac{b}{w_{h+1}} \rfloor\}, \text{ where}$$

$$- E_h^j := \{(l, a), (l + 1, b) \mid 0 \leq l \leq h - 1 \wedge (l, a), (l, b) \in V_h^j \wedge \{(l, a - j \cdot \lambda), (l + 1, b - j \cdot \lambda)\} \in E_h\}.$$

Here λ is the number of level- l nodes in each sub-fat tree (V_h^j, E_h^j) , i.e. $\lambda = w_1 \dots w_l m_{l+1} \dots m_h$. The edges in E_h^j are therefore just the edges in $XGFT^j(h; m_1, \dots, m_h; w_1, \dots, w_h)$, and an edge connecting some top-level node $(h + 1, a)$ with a top-level node $(h, b + j \cdot \lambda)$ in V_h^j where $b \in \{0, \dots, w_1 \dots w_h\}$ implies $a \in \{w_{h+1} \cdot b, w \cdot b + 1, \dots, w_{h+1} \cdot b + w_{h+1} - 1\}$.

For example, the communication network of the existing Connection Machine CM-5 with 256 leaves [16] is nothing but $XGFT(4; 4, 4, 4, 4; 2, 2, 2, 4)$.

4 Topological Properties

In most hardware designs using fat trees (e.g., Connection Machine CM-5, KSRI, Meiko CS-2) between multiple processing elements (PE's), these PE's are located at the m^h (or $m_1 \dots m_h$, respectively) leaves of the fat tree, whereas the remaining inner nodes are simple routers or high-speed switches. Therefore, the usual definitions of "cost", "distance", "average distance" etc. have to be slightly modified, as follows.

The diameter of $GFT(h, m, w)$ is the maximum distance between two leaves. The average distance $\bar{d}_{(0,i)}$ of a leaf $(0, i)$ from other leaves is

$$\bar{d}_{(0,i)} := \frac{\sum_{j \neq i} \text{dist}((0,i), (0,j))}{m^h - 1},$$

where $\text{dist}((k, i), (l, j))$ denotes the distance between the nodes (k, i) and (l, j) in $GFT(h, m, w)$. The average distance, \bar{d} , of $GFT(h, m, w)$ is defined as the arithmetic mean of the average distances of all leaves,

i.e., $\bar{d} := \frac{\sum_{(0,i)} \bar{d}_{(0,i)}}{m^h}$. Similarly, the average distance $\bar{d}_{(0,i)}$ of a leaf $(0, i)$ in $XGFT$ is

$$\bar{d}_{(0,i)} := \frac{\sum_{j \neq i} \text{dist}((0,i), (0,j))}{m_1 \dots m_h - 1}.$$

Hence, the average distance in $XGFT$ is

$$\bar{d} := \frac{\sum_{(0,i)} \bar{d}_{(0,i)}}{m_1 \dots m_h}.$$

4.1 Node Degree and Number of Edges

In $GFT(h, m, w)$, by definition the degree of each leaf is w and the degree of each intermediate node is $m + w$. The degree of the roots, i.e., the nodes in level h , is m . For each non-root (l, i) , the parent nodes are $(l + 1, w \cdot i), \dots, (l + 1, w \cdot (i + 1) - 1)$. For each non-leaf (l, i) the child nodes are $(l - 1, \lfloor \frac{i}{w} \rfloor + 0 \cdot w^{l-1}), \dots, (l - 1, \lfloor \frac{i}{w} \rfloor + (m - 1) \cdot w^{l-1})$.

The generalized fat tree $GFT(h, m, w)$ has $|V_h| = \sum_{i=0}^h m^i w^{h-i}$ nodes, such that level i contains $m^i w^{h-i}$ nodes. The extended generalized fat tree of height h has

$m_1 \cdot m_2 \cdots m_h$ leaves, $w_1 \cdot m_2 \cdots m_h$ nodes in level 1, ..., and finally $w_1 \cdot w_2 \cdots w_h$ top-level nodes. Therefore the total number of nodes in $XGFT$ is

$$|V_h| = \sum_{i=0}^{h-1} \prod_{j=i+1}^h m_j \prod_{j=1}^i w_j.$$

The number of edges between the levels l and $l+1$ in $XGFT$ is by definition w_{l+1} times the number of nodes in level l , since every node (l, j) has w_{l+1} parents. Thus, the total number of edges in $XGFT$ and $GFT(h, m, w)$ $XGFT$ is respectively given by

$$|E_h| = \sum_{i=0}^{h-1} \prod_{j=i+1}^h m_j \prod_{j=1}^{i+1} w_j \text{ and } \sum_{i=0}^{h-1} m^{h-i} w^{i+1}.$$

4.2 Diameter

The distance between two leaves $(0, i_1)$ and $(0, i_2)$ of $GFT(h, m, w)$ or $XGFT$ is two times the height of a smallest generalized sub-fat tree of $GFT(h, m, w)$ which contains both of them.

To be more precise, consider the following string notation of the node-set V_h , which also simplifies our routing algorithms in Section 5. An arbitrary node $(0, i)$ of $XGFT$ is contained in exactly one of the sets $V_{h-1}^0, \dots, V_{h-1}^{m_h-1}$. Let $\alpha_h \in \{0, 1, \dots, m_h - 1\}$ be the number such that $(0, i) \in V_{h-1}^{\alpha_h}$. For the same reason, there exists exactly one number $\alpha_{h-1} \in \{0, \dots, m_{h-1} - 1\}$ such that the leaf $(0, i \bmod m_1 \cdots m_{h-1}) \in V_{h-2}^{\alpha_{h-1}}$. Recursively, the numbers $\alpha_{h-2} \in \{0, \dots, m_{h-2} - 1\}$, $\alpha_{h-3} \in \{0, \dots, m_{h-3} - 1\}$, ..., $\alpha_1 \in \{0, \dots, m_1 - 1\}$ are computed. In fact, $\alpha_1 = i \bmod m_1$.

The mapping $A : (0, i) \mapsto (0; \alpha_h^{(0,i)}, \dots, \alpha_1^{(0,i)})$ (also denoted as $(0; \alpha_h^{(0,i)} \dots \alpha_1^{(0,i)})$) is one-to-one, and $\text{dist}((0, i), (0, j)) = 2 \cdot h'$, where $\alpha_h^{(0,i)} = \alpha_h^{(0,j)}$, ..., $\alpha_{h'+1}^{(0,i)} = \alpha_{h'+1}^{(0,j)}$ and $\alpha_{h'}^{(0,i)} \neq \alpha_{h'}^{(0,j)}$. Thus, the diameter of $GFT(h, m, w)$ and $XGFT$ is $2h$.

For example, the distance between the leaves $(0; pqr) = (0; 001)$ and $(0; stu) = (0; 010)$ in $GFT(3, 2, 2)$, shown in Figure 2c) is $2 \cdot 2 = 4$, since $p = s = 0$ but $q \neq t$, i.e. the smallest sub-fat tree containing both leaves has height 2.

For the inner nodes of $GFT(h, m, w)$, we use a similar notation. We denote an arbitrary ancestor x of a leaf $a = (0; \alpha_h, \alpha_{h-1}, \dots, \alpha_1)$ as $(1; \alpha_h, \alpha_{h-1}, \dots, \alpha_2, \beta_1)$, where $0 \leq \beta_1 \leq w - 1$ and x is the β_1 th of the m ancestors of the leaf a . The nodes in level 2 are then denoted as $(2; \alpha_h, \dots, \alpha_3, \beta_2, \beta_1)$ and finally the top-level nodes are denoted by $(h; \beta_h, \dots, \beta_1)$. This notation is well defined. Using this notation, the w parents of a non-root $(l; \alpha_h, \dots, \alpha_{l+1}, x, b_{l-1}, \dots, b_1)$, $0 \leq l \leq h - 1$, $0 \leq \alpha_h, \alpha_{h-1}, \dots, \alpha_{l+1}, x \leq m - 1$, $0 \leq b_{l-1}, \dots, b_1 \leq w - 1$ are $(l+1; \alpha_h, \dots, \alpha_{l+1}, y, b_{l-1}, \dots, b_1)$ for $0 \leq y \leq w - 1$. Similarly, the m children of a non-leaf $(l; \alpha_h, \dots, \alpha_{l+1}, y, b_{l-1}, \dots, b_1)$, $1 \leq l \leq h$, $0 \leq \alpha_h, \alpha_{h-1}, \dots, \alpha_{l+1} \leq m - 1$, $0 \leq y, b_{l-1}, \dots, b_1 \leq w - 1$ are $(l+1; \alpha_h, \dots, \alpha_{l+1}, x, b_{l-1}, \dots, b_1)$ for $0 \leq x \leq m - 1$.

4.3 Average Distance

From an arbitrary leaf $(0, i)$ in $XGFT$, there exist $(m_1 - 1)$ other leaves at a distance two, $(m_2 - 1)m_1$ leaves at distance four, ..., and finally $(m_h - 1)m_{h-1} \cdots m_1$ leaves at distance $2h$. Thus, the average distance of this leaf is obtained as

$$\bar{d}_{(0,i)} = 2 \sum_{k=1}^h k \left(\frac{m_k - 1}{m_k m_{k-1} \cdots m_{k+1}} \right) \left(\frac{m_1 \cdots m_h}{m_1 \cdots m_h - 1} \right),$$

which is independent of the choice of $(0, i)$. Due to leaf-

symmetry, this formula also gives the average distance, \bar{d} , of $XGFT$. In $GFT(h, m, w)$, the average distance is given by $\bar{d} = 2 \left(\frac{m-1}{m^h-1} \right) \sum_{i=0}^{h-1} (i+1) \cdot m^i$. Since for $m \geq 2$ and $h \geq 1$, the identity

$$\sum_{i=0}^{h-1} (i+1) \cdot m^i = \frac{hm^{h+1} - (h+1)m^h + 1}{(m-1)^2}$$

holds, it follows that $\bar{d} = 2h - \frac{m^h - hm^h + h - 1}{m^{h+1} - m^h - m + 1}$. For $m = 1$, $\bar{d} = 0$ due to only one leaf.

4.4 Leaf-Symmetry

The networks $GFT(h, m, w)$ and $XGFT$ are *leaf-symmetric*, because there exists an automorphism which maps each level onto itself and the leaf $a = (0; a_h, a_{h-1}, \dots, a_1)$ onto $b = (0; a_h, a_{h-1}, \dots, a_{k+1}, b_k, \dots, b_1)$ for any arbitrary choice of a, b . Such an automorphism is given by $\tau : (l; c_h, \dots, c_1) \mapsto (l; c_h, \dots, c_{k+1}, c_k \oplus (b_k \ominus a_k), \dots, c_1 \oplus (b_1 \ominus a_1))$, where \oplus_i and \ominus_i denote the addition and subtraction of integers modulo m_i , for $1 \leq i \leq k$.

4.5 Edge Bisection

Theorem 1 : *The edge bisection of $GFT(h, m, w)$ is at most mw^h .*

Proof : W.l.o.g., we show the result for even m . First we split V_h into the two equal sized blocks $B_1 = \bigcup_{i=0}^{\frac{m}{2}-1} V_{h-1}^i \cup \{(h, i) | 0 \leq i \leq \frac{w}{2} - 1\}$ and $B_2 = V_h - B_1$. Obviously, the only edges between B_1 and B_2 are the $\frac{mw^h}{2}$ edges from level h in B_1 to level $h - 1$ in B_2 , and the $\frac{mw^h}{2}$ edges from level h in B_2 to level $h - 1$ in B_1 . Therefore, mw^h is an upper bound of the edge bisection. \square

4.6 Comparison

Table 1 summarizes several topological properties, namely the number of nodes (N), degree (δ) which is equal to the connectivity (κ) or fault tolerance, and edge bisection (EB) of the generalized fat trees. These parameters are compared with those of the equal-sized hypercubes and tori.

Tori have a fixed degree and connectivity, while these values in a hypercube are equal to the logarithm of the number of nodes. Also the edge bisection of both topologies is fixed once the number of nodes is chosen. On the other hand, the generalized fat tree $GFT(h, m, w)$ allows us to choose the parameter w equal to the node-degree and connectivity, independent of the number of nodes. Also the edge bisection $EB = \frac{mw^h}{2}$ and the cost $C = 2wh$ is influenced from the value of w . Thus, the generalized fat trees provide a so called *bisection scalability*, contrary to the meshes, tori and hypercubes. The same holds for the extended generalized fat tree $XGFT(h; m_1, \dots, m_h; w_1, \dots, w_h)$, but additionally the values for w_i , $2 \leq i \leq h$, can be chosen such that the value for the edge bisection is adapted to the desired value. For instance, $XGFT(h; 4, \dots, 4; 2h, 1, \dots, 1)$ has the same number of nodes, degree and diameter as the binary hypercube $Q(2h)$, but a smaller edge bisection of $4h$ compared to 2^{2h-1} for $Q(2h)$.

5 Communication Issues

As defined in Section 4.2, the nodes in level i of the generalized fat tree $GFT(h, m, w)$ can be represented

Table 1: Comparison of topological properties

Network	N	$\delta = \kappa$	d	$C = \delta \cdot d$	EB
Hypercube $Q(2h)$	4^h	$2h$	$2h$	$4h^2$	2^{2h-1}
Generalized Hypercube GQ_h^m	m^h	$(m-1)h$	h	$(m-1)h^2$	$\frac{m^{h+1}}{4}$
Tori $T(n_1, n_2)$	$n_1 \cdot n_2$	4	n_1+n_2-2	$4(n_1+n_2)-8$	$\min\{n_1, n_2\}$
$T(2^h, 2^h)$	4^h	4	$2^{h+1}-2$	$2^{h+3}-8$	2^h
$T(m_1, \dots, m_h)$	$\prod_{i=1}^h m_i$	$2h$	$2(\sum_{i=1}^h m_i) - h$	$2h(\sum_{i=1}^h m_i) - 4h^2$	$\min_i \{\prod_{j \neq i} m_j\}$
Fat Trees $GFT(h, m, w)$	m^h	w	$2h$	$2hw$	$\frac{m \cdot w^h}{2}$
$GFT(h, 4, 4)$	4^h	4	$2h$	$8h$	2^{2h+1}
$GFT(h, 4, 2h)$	4^h	$2h$	$2h$	$4h^2$	$2 \cdot (2h)^h$
$XGFT(h; m_1, \dots, m_h; w_1, \dots, w_h)$	$\prod_{i=1}^h m_i$	w_1	$2h$	$2hw_1$	$\frac{m_h \cdot w_1 \cdot \dots \cdot w_h}{2}$
$XGFT(h; 4, \dots, 4; 2h, 1, \dots, 1)$	4^h	$2h$	$2h$	$4h^2$	$4h$
$XGFT(h; 4, \dots, 4; w_1, \dots, w_h)$	4^h	w_1	$2h$	$2hw_1$	$2w_1 \cdot \dots \cdot w_h$
$XGFT(h; m, \dots, m; (m-1)h, 1, \dots, 1)$	m^h	$(m-1)h$	$2h$	$2h^2(m-1)$	$\frac{m(m-1)h}{2}$

by strings $(i; x_h \dots x_{i+1} x_i \dots x_1)$ where $0 \leq x_j \leq m-1$ for $i+1 \leq j \leq h$ and $0 \leq x_j \leq w-1$ for $1 \leq j \leq i$.

5.1 Message Routing

In the following, we describe a simple self-routing scheme between any two processors in $GFT(h, m, w)$. Let $(0; a_h \dots a_1)$ denote the source node and $(0; b_h \dots b_1)$ the destination node. W.l.o.g., it is sufficient to give a routing between the source and the top-level node $t = (h; 0 \dots 0)$ since we can route the message from the source to the node t and from t to the destination node.

A possible routing from $(0; a_h \dots a_1)$ to $(h; 0 \dots 0)$ is the following path : $p = ((0; a_h \dots a_1), (1; a_h \dots a_2 0), (2; a_h \dots a_3 00), \dots, (h-1; a_h 0 \dots 0), (h; 0 \dots 0))$.

Even w node-disjoint paths $p_i, 0 \leq i \leq w-1$, between the source and destination nodes can be given as : $p_i = ((0; a_h \dots a_1), (1; a_h \dots a_2 i), (2; a_h \dots a_3 0i), \dots, (h-1; a_h 0 \dots 0i), (h; 0 \dots 0i), (h-1; b_h 0 \dots 0i), (h-2; b_h b_{h-1} 0 \dots 0i), \dots, (1; b_h \dots b_2 i), (0; b_h \dots b_1))$.

These w paths are node disjoint as well as edge disjoint. Figure 4a) shows the two node disjoint paths in $GFT(3, 2, 2)$ between $(0; 000)$ and $(0; 101)$.

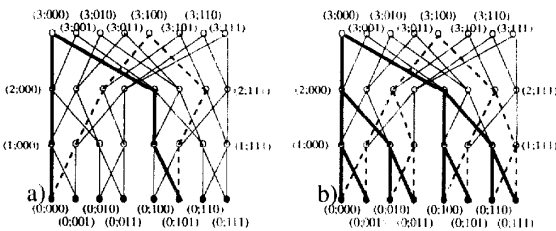


Figure 4: a) Node disjoint paths between $(0; 000)$ and $(0; 101)$ in $GFT(3, 2, 2)$ b) Two edge disjoint spanning trees in $GFT(3, 2, 2)$ with root $(0; 000)$

The routing and the construction of w_1 node disjoint paths between any two nodes can be shown analogously in the $XGFT$ network. As a consequence, the following theorem and corollary are obtained.

Theorem 2 : *The generalized fat trees $GFT(h, m, w)$ and $XGFT$ have a node-connectivity of w and w_1 , respectively. This implies a node- and edge-fault tolerance of $w-1$ and w_1-1 , respectively.*

Corollary 1 : *Even with upto $w-1$ (or w_1-1) node- or link-failures, a routing between any two operational*

processors in $GFT(h, m, w)$ and $XGFT$, respectively, can be guaranteed.

5.2 Broadcasting

Broadcasting, i.e., sending a message from a source node to all other nodes of a network in $GFT(h, m, w)$ can be done by sending the message from the source node to the top-level node $t = (h; 0 \dots 0)$ w.l.o.g., and afterwards sending the message from the node t to all other processors in the network. The routing from the source to node t is done as described in Section 5.1. For the broadcasting, we construct a spanning tree with root $t = (h; 0 \dots 0)$ as illustrated in Figure 4b). Under the all port communications model, broadcasting from any node in $GFT(h, m, w)$ takes at most $2h$ time steps.

In $GFT(h, m, w)$, w edge disjoint leaf-spanning trees can be constructed as follows. Let $a^* \in \{0, \dots, w-1\}$. For each leaf $(0; a_h \dots a_1)$, the a^* -th leaf spanning tree T_{a^*} (in Figure 5a) contains exactly the paths $((h; 0 \dots 0 a^*), (h; a_h 0 \dots 0 a^*), \dots, (1; a_h a_{h-1} \dots a_2 a^*), (0; a_h \dots a_1))$. These w leaf-spanning trees $T_{a^*}, 0 \leq a^* \leq w-1$ are obviously edge disjoint because of the differing values for a^* in the last position. So, they can be arranged as a multiple spanning trees graph, MST , with the root $r = (0; 0 \dots 0)$ and the w spanning trees T_{a^*} as subtrees (cf. Figure 5). The individual trees in MST are edge-disjoint and there exist w node disjoint paths between the root r and any other processor in $GFT(h, m, w)$. Thus, MST can be used to broadcast a message from the root to any other node on w node- and edge disjoint paths. Consequently, the broadcasting of a message from the root still works in the presence of upto $w-1$ faulty nodes or links. So we obtain:

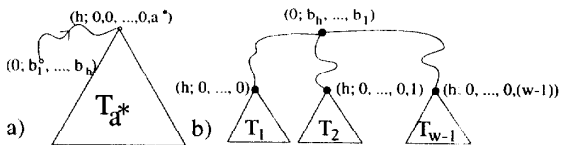


Figure 5: a) Spanning tree T_{a^*} b) Multiple spanning trees graph MST

Theorem 3 : *In $GFT(h, m, w)$ and $XGFT$, there exists a w - (or w_1 -) multiple spanning trees graph MST of height $2h$ which has w (or w_1) edge-disjoint subtrees and provides w (or w_1) node disjoint paths between the*

root and any other processor in $GFT(h, m, w)$. This allows broadcasting which tolerates upto $w-1$ (or w_1-1) faulty links or nodes.

6 Network Embeddings

Simulation of one network by another or mapping a task graph of a problem onto a multicomputer can be described by a *graph embedding*. An embedding of a guest graph $G = (V_G, E_G)$ into a host graph $H = (V_H, E_H)$ is formally defined by the tuple (f, g) with a node-mapping $f : V_G \rightarrow V_H$ and an edge-mapping $g : E_G \rightarrow \mathcal{P}(H)$ (pathset of H), where $g(\{u, v\})$ connects $f(u)$ and $f(v)$ in V_H , for all $\{u, v\} \in E_G$.

The *dilation* of the embedding (f, g) is the maximum distance in the host between the images of adjacent guest-nodes. The *expansion* is the ratio $\frac{|V_H|}{|V_G|}$. The *load* is the maximum, over all host-nodes, of the number of guest-nodes mapped onto a host-node. The *edge congestion* is the maximum number of edges of G that are routed by the mapping g over a single edge of H .

6.1 Linear Arrays and Rings

Theorem 4 : Let $w \geq 2$. In $GFT(h, m, w)$ with even m , a ring $R(m^h)$ can be embedded with optimal load one, dilation $2h$ and edge congestion one. If m is odd, then a linear array $L(m^h)$ can be embedded with optimal load one, dilation $2h$ and edge congestion one. In the latter case, a ring $R(m^h)$ can be embedded in $GFT(h, m, w)$ with load one, dilation $2h$ and edge congestion two. For $w = 1$, the edge congestion doubles.

Proof : For $w = 1$ the theorem is trivial, since $GFT(h, m, 1)$ is a m -ary complete tree of height h . Therefore, we consider the case $w \geq 2$ and odd m .

Let an \mathcal{H} -path in $GFT(h, m, w)$ be a path which starts at the root $(h, 0)$, routes to the leaf $(0, 0)$, then connects all leaves by a path of length at most $2h$, finally reaches the leaf $(0, m^h - 1)$ and routes to $(h, m^h - 1)$. Also, no edge is allowed to appear more than once in an \mathcal{H} -path.

For $h = 1$, we choose the \mathcal{H} -path as $H = ((1, 0), (0, 0), (1, 1), (0, 1), (1, 0), (0, 2), \dots, (0, m-2), (1, 0), (0, m-1), (1, 1))$. This is illustrated in Figure 6a).

For the induction step $(h+1)$, we embed in each (V_h^j, E_h^j) an \mathcal{H} -path H_j starting with the node (h, jw^h) and ending with $(h, jw^h + w^{h-1} - 1)$ where $0 \leq j \leq m-1$. Let H_j^{-1} denote the trace of H_j traversed in the reversed order. Then the new \mathcal{H} -path, H_{new} , is built by glueing all small \mathcal{H} -paths together, using H_j^{-1} for odd j .

$$H_{new} = ((h+1, 0), (h, 0)) \circ H_0 \circ ((h, w^{h-1}), (h+1, w^h), (h, w^h + w^{h-1})) \circ (H_1^{-1})^{-1}((h, w^h), (h+1, 0), (h, 2w^h)) \circ \dots \circ (H_{m-2})^{-1}((h, (m-2)w^h + w^{h-1}), (h+1, 0), (h, (m-1)w^h)) \circ H_{m-1} \circ ((h, (m-1)w^h + w^{h-1}), (h+1, 0), (h+1, w^h)),$$

where \circ denotes the concatenation of paths. This is sketched in Figure 6b). For even m , the \mathcal{H} -paths begin and end at $(h, 0)$. \square

Figures 6c) and d) illustrate embeddings of $L(9)$ in $GFT(2, 3, 2)$ and $L(16)$ in $GFT(2, 4, 2)$. The leaves are numbered according to their positions in the list.

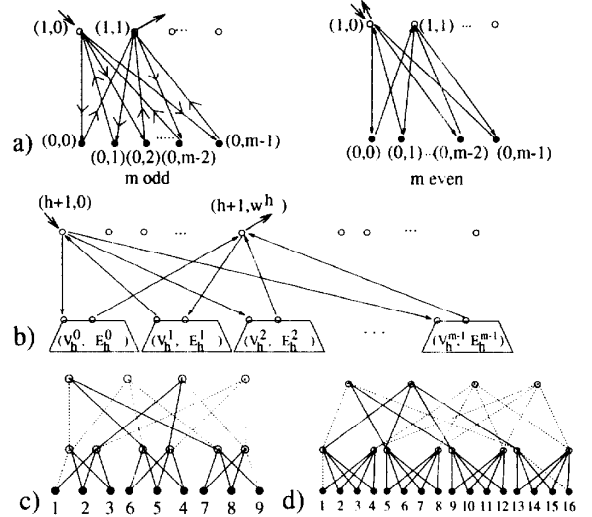


Figure 6: $L(m^h)$ in $GFT(h, m, w)$: a) Induction base b) Induction Step c) $L(9)$ in $GFT(2, 3, 2)$ d) $L(16)$ in $GFT(2, 4, 2)$

6.2 Complete Binary Trees and Meshes

For embeddings of meshes and trees, as well as to compute a lower bound of their edge congestion, we prove first the following lemma.

Lemma 1 : a) For each leaf $(0, i)$ in $GFT(h, w, w)$ there exist w paths $p_{i,1}, \dots, p_{i,w}$ starting at $(0, i)$ and ending at some root (h, j_i) . All paths in the set $\{p_{i,j} \mid (0, i) \text{ leaf}, 1 \leq j \leq w\}$ are edge disjoint.

b) For each h, m, w there exists an embedding of w^h paths $p_{i,j}$, $1 \leq j \leq w$ for each leaf $(0, i)$ in $GFT(h, m, w)$ to a top-level node with edge congestion $(\lceil \frac{m}{w} \rceil)^{h-1}$. On the other hand, the edge congestion $\mathcal{E}(h)$ is in $O(\lceil (\frac{m}{w})^{h-1} \rceil)$ such that $\mathcal{E}(h) = \Theta(\lceil (\frac{m}{w})^{h-1} \rceil)$.

Proof : Note that a) is just a special case of b). W.l.o.g., we assume that w divides h . Each of the w paths from each of the m^h leaves has to arrive at the root-level. Since there are only w^h top level nodes, even with an optimal balancing at least one of those will be an endpoint of at least $\frac{wm^h}{w^h} = w(\frac{m}{w})^h$ paths. Since each top-level node has m successors, the edge congestion is greater than or equal to $\frac{w}{m}(\frac{m}{w})^h = (\frac{m}{w})^{h-1}$.

To construct an optimal embedding, we enumerate all w paths starting from the leaf $l = (0; \alpha_h, \alpha_{h-1}, \dots, \alpha_1)$. For some arbitrarily chosen $\alpha_0 \in 0, \dots, m-1$, the w paths are $(0; \alpha_h, \alpha_{h-1}, \dots, \alpha_1), (1; \alpha_h, \alpha_{h-1}, \dots, \beta_2, \beta_1), \dots, (h; \beta_h, \dots, \beta_1)$, with $\beta_{i+1} = \lfloor \frac{\alpha_i}{w} \rfloor$, $0 \leq i \leq h-1$. Obviously, the congestion of edges among nodes in levels 0 and 1 is one. The edge congestion increases per level by $\frac{m}{w}$. This completes the proof \square

Theorem 5 : A complete binary tree $CBT(h)$ of height h can be embedded in $GFT(h+1, 2, 2)$ with an optimal load one, expansion $\frac{2^{h+1}}{2^{h+1}-1} \approx 1$, an optimal dilation $2(h+1)$ and edge congestion three.

Proof : See [11] \square

Theorem 6 : A two dimensional $m^l \times m^h$ -mesh $M(m^l, m^h)$ can be embedded in $GFT(h+l, m, m)$ with load and expansion one, dilation $2(h+l)$ and edge congestion three. If m is even, a two-dimensional $(m^h \times m^l)$ -torus can be embedded with the same values for load, expansion, edge congestion and dilation.

Proof : The expansion and load are equal to one and the dilation is equal to $2(h+l)$.

We show by induction that there exists an embedding of the mesh $M(m^l, m^h) = L(m^l) \times L(m^h)$ with two additional paths starting at each of the first and last m^l leaves $(0, i)$, $0 \leq i \leq m^l - 1$ and $m^{h+l} - m^l \leq i \leq m^{h+l} - 1$, and ending at the root $(h+l, (i \bmod m^l) \cdot w^{h-1})$ with edge congestion three. For $h = 1$, we obtain the result from Lemma 1 and Theorem 4. For the induction step, we embed m distinct $(m^l \times m^h)$ -meshes in the m sub-fat trees $GFT^0(h+l, m, m), \dots, GFT^{m-1}(h+l, m, m)$. We link the end nodes of the additional paths to the top-level nodes $GFT^j(h+l, m, m)$ via the corresponding root in the fat tree $GFT(h, m, m)$ (i.e., the node $(h, j \cdot m^{h+l+k})$ is connected with $(h+l, (j+1) \cdot m^{h+l+k})$ via the new root $(h+l+1, w \cdot k)$. This can be achieved with edge congestion two between the levels $h+1$ and h . By induction hypothesis, the edge congestion in each sub-fat tree is at most three. This construction is illustrated in Figure 7. For even m , the wraparound edges can be routed via the top-level nodes analogously to the proof of Theorem 4. \square

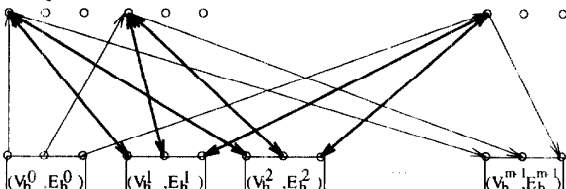


Figure 7: Embedding of $M(m^l, m^{h+1})$ in $GFT(h+1, m, w)$

A generalization of the last theorem yields :

Theorem 7 : A k -dimensional $(m^{h_1} \times \dots \times m^{h_k})$ -mesh can be embedded in $GFT(h_1 + h_2 + \dots + h_k, m, m)$ with load and expansion one, dilation $2(h_1 + \dots + h_k)$ and edge congestion $2k-1$. If m is even, a torus of the same size can be embedded with the same values for load, dilation, edge congestion and expansion.

6.3 Hypercubes

Theorem 8 : The h -dimensional hypercube $Q(h)$ can be embedded in $GFT(h, 2, 2)$ with load and expansion one, dilation $2h$, and edge congestion $\lceil \frac{h}{2} \rceil$.

Proof : We prove this result recursively starting with $h = 1$. Figure 8a) illustrates the embedding of the hypercube $Q(1)$ in $GFT(1, 2, 2)$. For the induction we use the recursiveness of both the hypercube and the fat tree $GFT(h, 2, 2)$. Thus, the hypercube $Q(2)$ is embedded in $GFT(2, 2, 2)$ as shown in Figure 8b).

For the construction of a hypercube $Q(h+1)$ in $GFT(h+1, 2, 2)$, one utilizes two instances of an embedded $Q(h)$ in $GFT(h, 2, 2)$. The corresponding nodes in the two instances of the hypercube are connected in the fat tree using the path which routes over the least common ancestor in the fat tree (cf. Figure 8c) and

Figure 9). If there are two unused least common ancestors, one chooses the left one. This is the case for h odd. For even h , we choose the unique least common ancestor which is unused.

Thus, the load and expansion are equal to one. The dilation is equal to the diameter $2h+2$ in the fat tree $GFT(h+1, 2, 2)$ which is obviously optimal. The edge congestion increases only for odd h by one, i.e., in each second step of this recursive construction. Hence, the edge congestion obtained is equal to $\lceil \frac{h+1}{2} \rceil$. \square

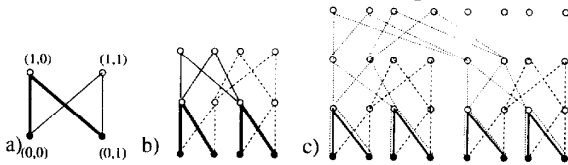


Figure 8: Embedding of a) $Q(1)$ in $GFT(1, 2, 2)$, b) $Q(2)$ in $GFT(2, 2, 2)$ and c) $Q(3)$ in $GFT(3, 2, 2)$

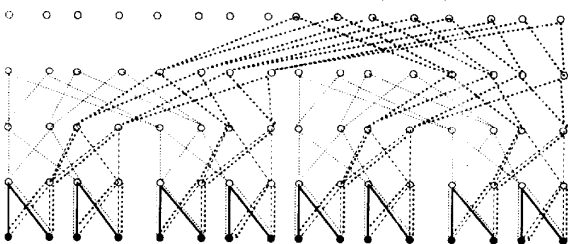


Figure 9: $Q(4)$ in $GFT(4, 2, 2)$

6.4 Pyramids

A pyramid $PR(n)$ of height n is a 4-ary complete tree of height n and the nodes at each level form a square mesh. Pyramids are very useful data structures in image processing and scientific multigrid computations.

Theorem 9 : In a fat tree $GFT(h+1, 4, 4)$, three instances of a pyramid $PR(h)$ of height h can be embedded with load one, dilation $2h+2$, expansion $\frac{4^{h+1}}{4^{h+1}-1} \approx 1$, and edge congestion at most 5.

Proof : The pyramid $PR(l)$ has $\sum_{i=0}^l 4^i = \frac{4^{l+1}-1}{3}$ nodes. Thus an embedding of three instances of $PR(l)$ in $GFT(1, 4, 4)$ leads to an expansion of $\frac{4^{h+1}}{4^{h+1}-1} \approx 1$.

Figure 10 embeds three instances of $PR(1)$ in $GFT(2, 4, 4)$. This embedding has load 1, expansion $\frac{16}{15}$, dilation 4, and edge congestion 1.

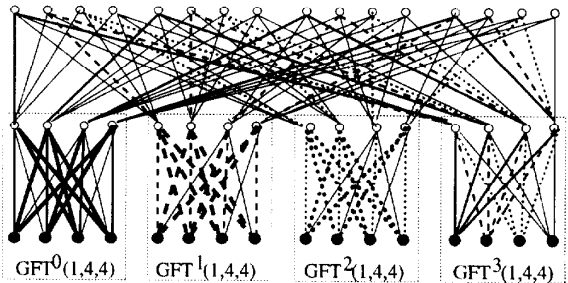


Figure 10: Embedding of three $PR(1)$'s in $GFT(2, 4, 4)$

The embedding of three instances of $PR(h+1)$ uses the embedding of three instances of $PR(h)$ in $GFT(h+$

1, 4, 4). A scheme is shown in Figure 11. The detailed proof is provided in [11]. \square

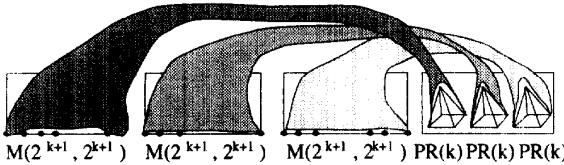


Figure 11: Embedding of $PR(h+1)$ in $GFT(h+2, 4, 4)$

6.5 Lower bound for the edge congestion

Since for $m > w$ the ratio between roots and leaves in $GFT(h, m, w)$ decreases exponentially with respect to the height h , it seems that every embedding of a guest graph with an edge bisection increasing linearly with the number of nodes must have necessarily a high edge congestion. The following theorem gives a relation between the edge bisection of the guest graph, the parameters h, m, w and the minimum edge congestion. It shows that an exponentially increasing edge congestion is necessary for every embedding where the edge bisection of the guest graph is in $\Omega(\# \text{nodes})$. Thus, for guest graphs with linearly increasing edge bisection, it is desirable to have $m \leq w$.

Theorem 10 : *Let $G = (V, E)$ be a guest graph to be embedded in $GFT(h, m, w)$, and b the edge bisection of G . Then the minimum edge congestion in every possible embedding of G in $GFT(h, m, w)$ is at least $O(\frac{2b}{mw^k})$.*
Proof : For $m = 1$, the theorem is trivial. W.l.o.g. assume m even. Then in each possible embedding, from the left $\frac{m}{2}$ subtrees V_{h-1}^j to the $\frac{m}{2}$ right subtrees there have to be at least b connections, consuming each two links. These $2b$ links must be distributed over the mw^h edges connecting level h and $h-1$ of $GFT(h, m, w)$. Thus, at least one edge must have a congestion of at least $\frac{2b}{mw^k}$. \square

Since the edge bisection of $Q(h)$ is 2^{h-1} , we obtain:

Corollary 2 : *Every embedding of a $Q(2h)$ in a $GFT(h, 4, w)$ has an edge congestion of at least $O((\frac{1}{w})^h)$ (e.g. = $O(2^h)$ for the fat tree $GFT(h, 4, 2)$)*

7 Conclusions

We have introduced a family of (extended) generalized fat tree interconnection networks which include as special cases Leiserson's fat trees used in the CM-5 machine, pruned butterflies and a few other variants. Our concept provides an unifying approach to define and analyze these fat tree based architectures. The generalized fat tree $GFT(h, m, w)$ has m^h PE's, degree w , diameter $2h$, and edge bisection $\frac{mw^h}{2}$. Thus, the degree, connectivity, cost and edge bisection can be chosen independent of the number of PE's.

The extended generalized fat tree $XGFT(h, m_1, \dots, m_h, w_1, \dots, w_h)$ with $m_1 \dots m_h$ PE's has a connectivity equal to its degree of w_1 , a diameter of $2h$, and an edge bisection of $\frac{m_1 \dots m_h}{2} (w_1 \dots w_h)$. The edge bisection can additionally be chosen independent of the degree. Thus, we can construct fat trees tailored to the application.

As part of our future research, we intend to analyze several implementation aspects of the generalized fat trees, such as constructing a VLSI-layout and analyzing other lower bound indicators such as the crossing

number, the bandwidth and the vertex bisection of the fat trees. To improve the scalability of the network model, we plan to design an incomplete fat tree network providing an arbitrary number of processors. Furthermore, it is worthwhile to investigate fault-tolerant embeddings and communication algorithms.

References

- [1] P. Bay and G. Bilardi. Deterministic on-line routing on area universal networks. In *Proceedings of the Annual Symp. on Foundations of Computer Science*, pages 297-306, Oct 1990.
- [2] T.F. Chan and Y. Saad. Multigrid algorithms on the hypercube multiprocessor. In *IEEE Transactions on Computers*, volume C-35, No. 11, pages 969-977, November 1986.
- [3] S. Frank, J. Rothnie, and H. Burkhardt. The KSR1 : Bridging the gap between shared memory and mpps. In *Proceedings Comcon'93*, San Francisco, CA, February 1993.
- [4] R.I. Greenberg and C.E. Leiserson. Randomized routing on fat trees. In Silvio Micali, editor, *Advances in Computing Research, Book 5: Randomness and Computation*, pages 345-374. JAI Press, Greenwich, CT, 1989.
- [5] K. Hwang and J. Ghosh. Hypernet: A communication-efficient architecture for constructing massively parallel computers. *IEEE Trans. Comput.*, 36:1450-1467, Dec 1987.
- [6] F.T. Leighton. *Introduction to Parallel Algorithms and Architectures : Arrays - Hypercubes*. Morgan Kaufmann Publishers, San Mateo, CA, 1992.
- [7] F.T. Leighton, B.M. Maggs, A.G. Ranade, and S.B. Rao. Randomized routing and sorting on fixed-connection networks. *Journal of Algorithms*, 17(1):157-205, July 1994.
- [8] C.E. Leiserson. Fat-trees : Universal networks for hardware-efficient supercomputing. *IEEE Transactions on Computers*, C-34(10):892-901, October 1985.
- [9] C.E. Leiserson, Z.S. Abumadeh, D.C. Douglas, C.R. Feynman, M.N. Ganmukhi, J.V. Hill, W.D. Hillis, R.C. Kuszmaul, M.A. St. Pierre, D.S. Wells, M.C. Wong, S.W. Yang, and R. Zak. The network architecture of the connection machine CM-5. In *Proceedings of the Symposium on Parallel Algorithms and Architectures*, pages 272 - 285, 1992.
- [10] S. Öhring and S.K. Das. Mapping dynamic data and algorithm structures into product networks. In *Proceedings of the 4th International Symposium on Algorithms and Computation (ISAAC'93)*, Hong Kong, *Lecture Notes in Computer Science*, vol. 762, pages 147-156, Dec 1993.
- [11] S.R. Öhring, M. Ibel, S.K. Das, and M. Kumar. *On generalized fat trees*. Technical Report CRPDC-94-18, University of North Texas, Denton, Oct 1994.
- [12] R. Ponnusamy, R. Thakur, A. Choudhary, K. Velamakanni, Z. Bozkus, and G. Fox. Experimental performance evaluation of the CM-5. *Journal of Parallel and Distributed Computing*, 19:192-202, 1993.
- [13] G. Ramanathan and J. Oren. Survey of commercial parallel machines. *ACM SIGARCH Computer Architecture News*, 21(3):13-33, June 1993.
- [14] Y. Saad and M.H. Schultz. Topological properties of hypercubes. *IEEE Trans. Comp.*, 37(7):867-872, 1988.
- [15] Klaus E. Schauer and Chris J. Scheiman. Experiments with active messages on the Meiko CS-2. To appear in the *Proceedings of the 9th International Parallel Processing Symposium*, Santa Barbara, April, 1995.
- [16] Thinking Machines Corporation. *The Connection Machine System -- CM-5 Technical Summary*, November 1993.
- [17] C.D. Thompson. Area-time complexity for VLSI. In *Proc. of the ACM Symposium on Theory of Computing*, 1979.