On the Bisection Width and Expansion of Butterfly Networks

Claudson Bornstein[*] Ami Litman[1†] Bruce M. Maggs[2*] Ramesh K. Sitaraman[3§] Tal Yatzkar[†]

Abstract

This paper proves tight bounds on the bisection width and expansion of butterfly networks with and without wraparound. Previously it was known that the bisection width of an n-input butterfly with wraparound is n. We show that without wraparound, the bisection width is $2(\sqrt{2}-1)n + o(n) \approx .82n$. This result is surprising because it contradicts the prior "folklore" belief that the bisection width is n in both cases. We also show that for every set A of k nodes in a butterfly with wraparound there are at least $(4 + o(1))k/\log k$ edges from A to $\bar{A}$, provided that $k = o(n)$.

1 Introduction

This paper analyzes the bisection width and expansion of a network called a *butterfly*. This network has been studied extensively and it, or one of its variants, has served as the routing network in several parallel computers and ATM switches. Surprisingly, however, the precise values of the butterfly's bisection width and expansion were not previously known. This paper proves tight upper and lower bounds on these parameters.

*Carnegie Mellon University, School of Computer Science, 5000 Forbes Ave., Pittsburgh, PA 15213, USA. bmm,cfb@cs.cmu.edu
†Computer Science Department, Technion, Technion City, Haifa 32000, Israel. litman,ytali@cs.technion.ac.il
§Dept. of Computer Science, University of Massachusetts, Amherst, MA 01003, USA. ramesh@cs.umass.edu

[1]Ami Litman is supported in part by BSF Grant No. 94-266.
[2]Bruce Maggs is supported in part by the Air Force Materiel Command (AFMC) and ARPA under Contract F196828-93-C-0193, by ARPA Contracts F33615-93-1-1330 and N00014-95-1-1246, and by an NSF National Young Investigator Award, No. CCR-94-57766, with matching funds provided by NEC Research Institute and Sun Microsystems.
[3]Ramesh Sitaraman is supported in part by NSF Grant No. CCR-94-10077.
The views and conclusions contained here are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either express or implied, of AFMC, ARPA, CMU, or the U.S. Government.

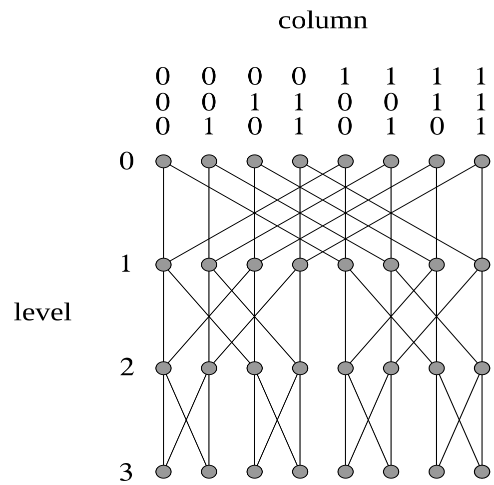

Figure 1. A 32-node butterfly network.

1.1 The butterfly and cube-connected cycles networks

Throughout this paper, we use the following terminology to describe butterfly networks. The $(\log n)$-dimensional butterfly B_n has $N = n(\log n+1)$ nodes arranged in $\log n+1$ levels of n nodes each. (All logarithms in this paper are base 2.) Each node has a distinct label $\langle w, i\rangle$ where i is the level of the node ($0 \le i \le \log n$) and w is a $\log n$-bit binary number that denotes the *column* of the node. All nodes of the form $\langle w, i\rangle$, $0 \le i \le \log n$, are said to belong to column w. Similarly, the ith level L_i consists of all of the nodes $\langle w, i\rangle$, where w ranges over all $\log n$-bit binary numbers. Two nodes $\langle w, i\rangle$ and $\langle w', i'\rangle$ are linked by an edge if $i' = i + 1$ and either w and w' are identical or w and w' differ only in the bit in position i'. (The bit positions are numbered 1 through $\log n$, the most significant bit being numbered 1.) The edges in the network are undirected. The nodes on level 0 are called the *input nodes* or just *inputs* of the network, and the nodes on level $\log n$ are called the *output nodes* or just *outputs*. A 32-node butterfly network is shown in Figure 1.

Sometimes the level 0 and $\log n$ nodes in each column are assumed to be the same node. In this case, the butterfly is said to *wrap around*. We use W_n to denote the $(\log n)$-dimensional butterfly with wraparound. This network has $n \log n$ nodes.

A number of properties of butterfly networks were known prior to this work. For example, it is not difficult to show that the diameter of B_n is $2 \log n$, and the diameter of W_n is $\lfloor (3 \log n)/2 \rfloor$, where the *diameter* of a network is the maximum, over all pairs of nodes u and v, of the length, in terms of edges, of the shortest path between u and v. Also, the VLSI layout area of B_n is $(1 \pm o(1))n^2$ [3] and the layout area of W_n is $\Theta(n^2)$. Furthermore, the three-dimensional layout volumes of B_n and W_n are $\Theta(n^{3/2})$ [14].

A network closely related to the butterfly is the *cube-connected cycles* [8, 21]. A $\log n$-dimensional cube-connected cycles network CCC_n consists of n cycles, each containing $\log n$ nodes. Each cycle has a distinct $\log n$-bit label, and within a cycle each node is labeled with its position, a number between 1 and $\log n$, inclusive. Taken together, these labels give each node a distinct label $\langle w, i \rangle$, where w is the label of its cycle, and i is its position in the cycle. Two nodes in different cycles are connected by an edge if and only if they share the same position i within their respective cycles, and their cycle labels differ only in the bit in position i. I.e., two nodes $\langle w, i \rangle$ and $\langle w', i \rangle$ are connected if w and w' differ in bit position i.

1.2 Bisection width

The *bisection width* of an N-node network $G = (V, E)$ is defined as follows. A *cut* $(S, \bar{S})$ of G is a partition of its nodes into two sets S and $\bar{S}$, where $\bar{S} = V - S$. The *capacity* of a cut, $C(S, \bar{S})$, is the number of (undirected) edges with one endpoint in S and the other in $\bar{S}$. A *bisection* of a network is a cut $(S, \bar{S})$ such that $|S| \le \lceil N/2 \rceil$ and $|\bar{S}| \le \lceil N/2 \rceil$. The *bisection width* $BW(G)$ is the minimum, over all bisections $(S, \bar{S})$, of $C(S, \bar{S})$. In other words, the bisection width is the minimum number of edges that must be removed in order to partition the nodes into two sets of equal cardinality (to within one node).

The bisection width of a network is an important indicator of its power as a communications network. As an example, suppose that an N-node network G is used to route messages between the processors in a general-purpose parallel computer, with one processor attached to each node. If each processor sends a message to another processor chosen at random, then the expected number of messages that cross the bisection is $N/2$. Assuming that each edge of the network can transmit one message in one time step, the time required by the network to route the messages is at least $N/2BW(G)$. Hence, the smaller the bisection width, the longer it will take to route these messages. In addition to this example, there are a large number of problems for which it is possible to prove some lower bound, I, on the number of messages that must cross a bisection of a parallel machine in order to solve the problem. In each case, $I/BW(G)$ is a lower bound on the time, T, to solve the problem.

The bisection width of a network also gives a lower bound on the VLSI layout area, A, of a network G. In particular, Thompson proved that $A \ge (BW(G))^2$ [24]. Combining this inequality with the inequality $T^2 \ge (I/BW(G))^2$ for any particular problem yields the so-called "AT^2" bound $AT^2 \ge \Omega(I^2)$. (See [24].)

1.3 Expansion

The *expansion* of a network G is defined as follows. The *edge expansion* of a set S of nodes is $C(S, \bar{S})$. We define the *edge-expansion function* $EE(G, k)$ of the network to be

$$EE(G, k) = \min_{S:|S|=k} C(S, \bar{S})$$

for $1 \le k \le N$. The set of neighbors $\mathcal{N}(S)$ of a set S are the nodes in $\bar{S}$ that are adjacent to nodes in S, i.e.,

$$\mathcal{N}(S) = \{v \in \bar{S} \mid \exists u \in S, (u, v) \in E\}.$$

The *node expansion* of a set S is $|\mathcal{N}(S)|$. We say that a network G has *node-expansion function* $NE(G, k)$ if, for $1 \le k \le N$,

$$NE(G, k) = \min_{S:|S|=k} \mathcal{N}(S).$$

The expansion of a network G is an indicator of the speed at which information can disseminate in G. In particular, if each node in a set of k nodes holds a small piece of information, they can increase the number of nodes holding the information to $k + NE(G, k)$ in a single step. Several load-balacing algorithms exploiting this property are reported in [9]. The expansion function can also be used to compare the computational powers of different networks. In particular, a difference in the expansion functions of a guest network and a host network has been used to prove lower bounds on the inefficiency of any emulation of the guest by the host [12, 22]. Finally, we observe that the only N-node bounded-degree networks known to be capable of routing and sorting deterministically in $O(\log N)$ time are those that incorporate some form of expansion (i.e., expansion functions of the form $NE(G, k) \ge (1 + \epsilon)k$, for some fixed $\epsilon > 0$) into their structures [1, 2, 15, 16, 26].

1.4 Lower bounds based on embeddings

Lower bounds on the bisection width and expansion of an N-node network H can often be proved by *embedding* the complete graph $G = K_N$ into H. In general, an embedding of a guest network G into a host network H is a mapping that maps nodes of G to nodes of H and edges of G to paths in H. The *load* l of an embedding is the maximum number of nodes of G mapped to any one node of H. The *congestion* c of the embedding is the maximum number of paths (corresponding to edges in G) that cross any one edge of H. The *dilation* d of an embedding is the length of the longest path. In proving lower bounds on bisection width and expansion, the embedding typically has load one, and routes the same number of paths, c, across each edge of H.

Given an embedding of K_N into H with load one and congestion c, a lower bound on $BW(H)$ is computed as follows. Let $(A, \overline{A})$ be a bisection of H with capacity $C(A, \overline{A}) = BW(H)$. Then removing the edges from K_N whose paths cross $(A, \overline{A})$ yields a bisection of K_N with capacity at most $c \cdot BW(H)$. Since $BW(K_N) = N^2/4$, we have $c \cdot BW(H) \geq N^2/4$, and hence $BW(H) \geq N^2/4c$. This approach readily yields $\Omega(n)$ lower bounds on the bisection widths of B_n and W_n, but without the right leading constants.

The same technique can be used to prove lower bounds on the edge expansion of a graph. Suppose that K_N is embedded in H with load one and congestion c. Let A be any set of k nodes in H. For each of the $EE(K_N, k)$ edges leading out of the corresponding set in K_N, a path must be routed out of A in H. Thus, we must have $c \cdot C(A, \overline{A}) \geq EE(K_N, k)$. Since the edge expansion of K_N is $EE(K_N, k) = k(N-k)$, we have $C(A, \overline{A}) \geq k(N-k)/c$. Since A was an arbitrary set of k nodes, $EE(K_N, k) \geq k(N-k)/c$. For $k \leq N/2$, $EE(K_N, k) \geq kN/2$, which gives $EE(H, k) \geq kN/2c$.

Prior to this work some bounds on $BW(B_n)$ were known, and $BW(W_n)$ had been analyzed exactly. It is not difficult to show that $BW(B_n) \leq n$ and $BW(W_n) \leq n$: partition the columns into those whose numbers start with a 0 and those that start with a 1. Similarly, $BW(CCC_n) \leq n/2$. For the cube-connected cycles network, Manabe, Hagihara, and Tokura [17] proved the converse, namely $BW(CCC_n) \geq n/2$. The same approach can be used to show that $BW(W_n) \geq n$. Hence $BW(CCC_n) = n/2$ and $BW(W_n) = n$. Since these results appear only in Japanese, we present original proofs in [7], but for the sake of brevity omit them here. Prior to this work, $BW(B_n)$ was known to be at least $n/2$. This lower bound is proved by embedding the graph $2K_N$ into B_n, where $2K_N$ is a variant of the complete graph in which any two nodes are connected by *two* parallel edges. There is an embedding of $2K_N = 2K_{n(\log n+1)}$ into B_n with load 1 and congestion $n(\log n + 1)^2$. Since $BW(2K_{n(\log n+1)}) = (n(\log n+1))^2/2, BW(B_n) \geq n/2$.

These embeddings also imply that the edge expansion functions of B_n and W_n satisfy $EE(B_n, k) = \Omega(k/\log n)$ and $EE(W_n, k) = \Omega(k/\log n)$, for $k \leq N/2$.

1.5 Related networks

Another network closely related to the butterfly is the Beneš network. A $\log n$-dimensional Beneš network consists of two back-to-back $\log n$-dimensional butterflies B_n and B'_n, where the ith node on level $\log n$ of B_n is identified with the ith node on level $\log n$ node of B'_n. The nodes on level 0 of B_n are called the input nodes of the Beneš network, and the nodes on level 0 of B'_n are called the output nodes. Typically each input node is viewed as having two input ports (i.e., connections for edges), and each output node is viewed as having two output ports. The Beneš network is called *rearrangeable* because it is possible to route edge-disjoint paths between its $2n$-input ports and $2n$-output ports in any permutation [5, 6, 27].

In addition to the cube-connected cycles and Beneš networks, the butterfly has been shown to be closely related to the hypercube and other bounded-degree variants of the hypercube, including the shuffle-exchange network and the de Bruijn network. For example, it is not difficult to prove that an N-node butterfly network can be embedded in an N-node hypercube with constant load, congestion, and dilation. In fact, Greenberg, Heath, and Rosenberg [11] proved that, for some sizes of N, the butterfly network is a subgraph of the hypercube. Also, Schwabe [23] showed that an N-node butterfly network can emulate T steps of any computation of an N-node shuffle-exchange network (or de Bruijn network) in $O(T)$ steps, and vice versa.

More information about the structural and algorithmic properties of butterflies can be found in the book by Leighton [13]. Some of the parallel computers that use butterfly networks or its variants are described in [4, 10, 18, 19, 20]. The butterfly network and its variants are also used as routing networks in modern ATM switches [25].

1.6 Our results

We begin in Section 2 by proving that the bisection width of the n-input butterfly network without wraparound, B_n, is $(2(\sqrt{2}-1) + o(1))n$. We show how to construct such a bisection and prove that no bisection is smaller. This result is surprising, because it contradicts the prior folklore belief that the bisection width was n. In Section 3 we give upper and lower bounds on the edge- and node-expansion functions of W_n and B_n. We show, for example, that for every set A of k nodes in W_n there are at least $(4 + o(1))k/\log k$ edges from A to $\overline{A}$, provided that $k = o(n)$.

2 The bisection width of the butterfly

In this section we show that the bisection width of the butterfly, $BW(B_n)$, satisfies: $2(\sqrt{2}-1)n < BW(B_n) \leq 2(\sqrt{2}-1)n + o(n)$.

We reach this result as follows. We begin by introducing a highly symmetric graph, the *mesh of stars*, and an embedding of the butterfly into this graph. We use the embedding and the (as of yet unknown) bisection width of the mesh of stars to establish tight lower and upper bounds on $BW(B_n)$. We conclude by computing the bisection width of the mesh of stars.

Henceforth, let n always be a power of 2. What follows is a list of properties of the butterfly that we use in our constructions; most of these properties are well known and are given with no proof.

Lemma 2.1 *There is an automorphism of B_n (i.e., an isomorphism of B_n to itself) that maps each level L_i onto $L_{\log n - i}$.*

Lemma 2.2 *Let v and v' be two nodes on the same level of B_n. Then there is a level-preserving automorphism π of B_n s.t. $\pi(v) = v'$. Moreover, let $v - u$ and $v' - u'$ be two edges of B_n s.t. v and v' are on the same level and u and u' are on the same level. Then there is a level-preserving automorphism π of B_n s.t. $\pi(v) = v'$ and $\pi(u) = u'$.*

Let p be a path in a graph G whose nodes have been partitioned into levels. We call p *monotonic* if p visits any level at most once.

Lemma 2.3 *Let v and u be nodes of B_n, $v \in L_0$ and $u \in L_{\log n}$. Then there is exactly one monotonic path linking v and u.*

For $0 \le i \le j \le \log n$, let $B_n[i, j]$ denote the subgraph of B_n induced by levels $L_i, L_{i+1}, \cdots, L_j$.

Lemma 2.4 *Let $0 \le i \le j \le \log n$. Then $B_n[i, j]$ has $n/2^{j-i}$ connected components; each component is isomorphic to $B_{2^{j-i}}$; and the k–th level of each component is a subset of the $(i + k)$–th level of B_n.*

Lemma 2.5 *Let $n > 1$. Then there is a partition of L_0, the first level of B_n, into two disjoint sets, I and O, each of cardinality $n/2$ s.t. if we assign two distinct "input ports" to each node of I and two distinct "output ports" to each node of O then the resulting network is rearrangeable. That is, for any bijection of the input ports to the output ports there is a set of n edge-disjoint paths that link each input port with its image output port.*

We say that a subset of nodes U is *compact* in G if for any given cut of G we can move all of U to one side of the cut without increasing its capacity. Formally, let $G = \langle V, E\rangle$ be a graph and $U \subseteq V$. Then U is compact in G if for any cut of G $g = (A, \overline{A})$, there is another cut $g' = (A', \overline{A'})$ s.t.:
1. either $U \subset A'$ or $U \subset \overline{A'}$,
2. $A \bigcap (V - U) = A' \bigcap (V - U)$, and
3. $C(g') \le C(g)$.

Lemma 2.6 *U is compact in G if U is compact in the subgraph of G induced by $U \bigcup \mathcal{N}(U)$.*

Lemma 2.7 *Let U be a compact set of nodes of a graph G. Then every connected component induced in G by U is also compact.*

Lemma 2.8 *$B_n[1, \log n]$ is compact in $B_n = B_n[0, \log n]$.*

Lemma 2.9 *Each connected component of $B_n[i, \log n]$ is compact in B_n, $1 \le i \le \log n$*

Proof: Let B' be a connected component of $B_n[i, \log n]$, and let B'' be the other connected component of $B_n[i, \log n]$ such that $\mathcal{N}(B'') = \mathcal{N}(B')$. Moreover, let G be the graph induced in B_n by $B' \cup B'' \cup \mathcal{N}(B')$. G is isomorphic to $B_{n/2^{i-1}}$, and by Lemma 2.8 $B' \cup B''$ is compact in $B_{n/2^{i-1}}$, and thus by Lemmas 2.6 and 2.7, B' and B'' are also compact in B_n. □

Lemma 2.10 *Let $0 \le i \le \log n$, $0 \le j$ and $k = n2^j$. Then there is an embedding π of B_k into B_n s.t.:*
1. The dilation of the embedding is 1.
2. The congestion of any edge is exactly 2^j.
3. π maps $B_k[0, i-1]$ on $B_n[0, i-1]$ with uniform load of 2^j.
4. π maps $B_k[i + 1 + j, \log n + j]$ on $B_n[i + 1, \log n]$ with uniform load of 2^j.
5. For each $l \in [i, i + j]$, π maps exactly 2^j nodes of the l–th level of B_k on each node of the i–th level of B_n.

2.1 Reducing the butterfly to a mesh of stars

The $j \times k$ *mesh of stars*, denoted $MOS_{j,k}$, is the graph obtained from the complete bipartite graph $K_{j,k}$ by replacing each edge with a path of length two. This graph has three levels that we refer to as M_1 (with j nodes), M_2 (with jk nodes), and M_3 (with k nodes).

Let $G = (V, E)$ be a graph, let $g = (A, \overline{A})$ be a cut of G, and let U be a subset of V. We say that g *bisects* U if $|A \bigcap U| \le |\overline{A} \bigcap U| \le |A \bigcap U| + 1$. The U*–bisection width of* G is defined by:

$$BW(G, U) = \min\{C(g) :\ g \text{ is a cut of } G \text{ that bisects } U\}.$$

In this section we show that

$$\frac{2BW(MOS_{n,n}, M_2)}{n^2} \le \frac{BW(B_n)}{n}$$
$$\le \frac{2BW(MOS_{f(n),f(n)}, M_2)}{f(n)^2} + o(1)$$

for some function f s.t. $\lim_{n\to\infty} f(n) = \infty$. Later we will compute the bisection width of the mesh of stars, which will give us lower and upper bounds on $BW(B_n)$

We establish both bounds on $BW(B_n)$ via the following embedding of butterflies into meshes of stars.

Lemma 2.11 *Let $j, k > 1$, and suppose jk divides n. Then there is an embedding π of B_n into $MOS_{j,k}$ s.t.:*
1. The dilation of the embedding is 1.
2. The congestion of any edge is exactly $2n/jk$.
3. π maps the first $\log k$ levels of B_n onto M_1 with uniform load.
4. π maps the last $\log j$ levels of B_n onto M_3 with uniform load.
5. π maps the other nodes of B_n onto M_2 with uniform load. Moreover, if $jk = n$ then $\pi^{-1}(\{v\})$ is compact for any node v of $MOS_{j,k}$, and the load of any node of M_2 is 1.

First, let us establish the lower bound on the bisection width of B_n in terms of $BW(MOS_{n,n}, M_2)$.

Lemma 2.12 *Let $n > 1$. Then:*
1. There is an i s.t. $0 \le i \le \log n$ and $BW(B_n, L_i) \le BW(B_n)$.
2. $BW(B_{n^2}, L_{\log n})/n^2 \le BW(B_n)/n$.

Proof: To establish (1), let $g = (A, \overline{A})$ be a bisection of B_n s.t. $C(g) = BW(B_n)$. Assume, without loss of generality, that $|A \bigcap L_0| \le n/2$. Then there is an i s.t. $|A \bigcap L_i| \le n/2 \le |A \bigcap L_{i+1}|$. Let $g' = (A', \overline{A'})$ be a cut of B_n s.t. $C(g') \le C(g)$, $|A' \bigcap L_i| \le n/2 \le |A' \bigcap L_{i+1}|$ and $|A' \bigcap L_{i+1}| - |A' \bigcap L_i|$ is minimal. (g' does not necessarily bisect B_n.) We establish (1) by showing that g' bisects either L_i or L_{i+1}. Assume otherwise. Since $|A' \bigcap L_i| < n/2 < |A' \bigcap L_{i+1}|$, there is a 4–cycle, $v{-}u{-}v'{-}u'{-}v$, with $v, v' \in L_i$ and $u, u' \in L_{i+1}$ s.t. $|A' \bigcap \{v, v'\}| < |A' \bigcap \{u, u'\}|$. Hence, either $|A' \bigcap \{v, v'\}| = 0$ or $|A' \bigcap \{u, u'\}| = 2$. In both cases we can modify g' by moving a single node from A' to $\overline{A'}$ or vice versa to reduce $|A' \bigcap L_{i+1}| - |A' \bigcap L_i|$ without increasing the capacity. A contradiction.

To prove (2), let $BW(B_n, L_i) \le BW(B_n)$ and let $g = (A, \overline{A})$ be a cut of B_n that bisects L_i and $C(g) = BW(B_n, L_i)$. Apply Lemma 2.10 with $j = \log n$ and $k = n^2$. Let π be the embedding of B_{n^2} into B_n given by this lemma. Define a cut of B_{n^2} by $g' = (\pi^{-1}(A), \pi^{-1}(\overline{A}))$. Since the congestion of each edge of B_n is exactly n, $C(g) \cdot n = C(G')$, and hence $C(g')/n^2 = C(g)/n$. Since g bisects the i–th level of B_n and $\log n \in [i, i + \log n]$, g' bisects the $(\log n)$–th level of B_{n^2}. Hence, $BW(B_{n^2}, L_{\log n})/n^2 \le C(g')/n^2 = C(g)/n = BW(B_n, L_i) \le BW(B_n)/n$. □

Lemma 2.13 $2BW(MOS_{n,n}, M_2)/n^2 \le BW(B_n)/n$.

Let us now establish the upper bound on the bisection width of B_n. Let $G = \langle V, E\rangle$ be a graph, $g = (A, \overline{A})$ a cut of G and $U \subset V$. We say that U *is amenable w.r.t.* cut g if the g can capture any number of nodes of U (but not necessarily any subset of U) from 0 to $|U|$ without increasing its capacity, i.e., U is amenable w.r.t. g in the graph G if for every $0 \le k \le |U|$ there is a cut $g' = (A', \overline{A'})$ s.t.:
1. $A \bigcap (V - U) = A' \bigcap (V - U)$,
2. $|A' \bigcap U| = k$, and
3. $C(g') \le C(g)$.

Lemma 2.14 *Let $g = (A, \overline{A})$ be a cut of G, U a set of nodes and $W = U \bigcup \mathcal{N}(U)$. Then U is amenable w.r.t. g in G iff U is amenable w.r.t. $g|_W \triangleq (A \bigcap W, \overline{A} \bigcap W)$ in the subgraph of G induced by W.*

Lemma 2.15 *Let $n > 2$, U be a connected component of $B_n[1, \log n - 1]$ and $g = (A, \overline{A})$ a cut of B_n s.t. $L_0 \bigcap \mathcal{N}(U) \subset A$ and $L_{\log n} \bigcap \mathcal{N}(U) \subset \overline{A}$. Then U is amenable w.r.t. g.*

Lemma 2.16 $BW(B_n)/n \le 2BW(MOS_{j,j}, M_2)/j^2 + 4/j$ *for any j, a power of 2, for all large enough n.*

2.2 The bisection width of the mesh of stars

In this subsection we show:

$$\sqrt{2} - 1 < \frac{BW(MOS_{j,j}, M_2)}{j^2} \le \sqrt{2} - 1 + o(1).$$

To this end, we use the symmetry of the graph $MOS_{j,j}$ which causes the capacity of an 'honest' $(A, \overline{A})$ cut to depend only on j, $|A \bigcap M_1|$, and $|A \bigcap M_3|$.

The real function $f(x, y) \triangleq x + y - \min(1, 2xy)$, defined on the closed domain $D = \{\langle x, y\rangle : 0 \le x, y \le 1 \text{ and } 1 \le x + y\}$, is relevant to $BW(MOS_{j,j}, M_2)$ as demonstrated by the next lemma.

Lemma 2.17 *Let j be an even integer and $\langle x, y\rangle \in D$ s.t. $x \cdot j$ and $y \cdot j$ are integers. Let B be the set of cuts $(A, \overline{A})$ of $MOS_{j,j}$ that bisect M_2 and satisfy $|A \bigcap M_1| = x \cdot j$ and $|A \bigcap M_3| = y \cdot j$. Then $f(x, y)j^2 = \min\{C(g) : g \in B\}$.*

Proof: Let $g = (A, \overline{A})$ be a cut in B s.t. $C(g)$ is minimal. The graph $MOS_{j,j}$ has $x(1 - y)j^2$ monotonic paths (of length two) leading from $A \bigcap M_1$ to $\overline{A} \bigcap M_3$, and $(1-x)yj^2$ monotonic paths leading from $\overline{A} \bigcap M_1$ to $A \bigcap M_3$. Each of these paths contributes one to the capacity of g.

In addition, there are $x \cdot y \cdot j^2$ monotonic paths from $A \bigcap M_1$ to $A \bigcap M_3$. Assume w.l.o.g. that $(1-x)(1-y) < xy$. Since g bisects M_2, if $\frac{1}{2} < xy$ then the middle node of exactly $(xy - \frac{1}{2})j^2$ of these paths is in $\overline{A}$. Each of these nodes contribute two to the capacity of g. Otherwise, if $xy \le \frac{1}{2}$, then these paths contribute nothing to the capacity of g. Since $(1-x)(1-y) < xy$, there are at most $j^2/2$ paths from $\overline{A} \bigcap M_1$ to $\overline{A} \bigcap M_3$, and hence these paths contribute nothing to the capacity of g.

In summary, $C(g)/j^2 = x(1 - y) + (1 - x)y + 2\max(xy - \frac{1}{2}, 0) = x + y - 2xy + \max(2xy - 1, 0) = x + y + \max(-1, -2xy) = x + y - \min(1, 2xy) = f(x, y)$. □

Lemma 2.18 *The function $f = x + y - \min(1, 2xy)$ is continuous in the domain $D = \{\langle x, y\rangle : 0 \le x, y \le 1$ and $1 \le x + y\}$, and $f\left(\sqrt{\frac{1}{2}}, \sqrt{\frac{1}{2}}\right) = \sqrt{2} - 1$ is a (global) minimum of f.*

Lemma 2.19 *For an even integer j:*

$$\sqrt{2} - 1 < BW(MOS_{j,j}, M_2)/j^2 \le \sqrt{2} - 1 + o(1).$$

Proof: Let $g = (A, \overline{A})$ be a cut of $MOS_{j,j}$ that bisects M_2 such that $C(g)$ is minimal. Assume, w.l.o.g., that $j \le |A \bigcap (M_1 \bigcup M_3)|$. (Otherwise, swap A and $\overline{A}$.) Let $x = |A \bigcap M_1|/j$ and $y = |A \bigcap M_3|/j$. Clearly, $\langle x, y\rangle \in D$.

The first inequality, $\sqrt{2} - 1 < BW(MOS_{j,j}, M_2)/j^2$, follows from Lemmas 2.17 and 2.18 and the fact that $\sqrt{2} - 1$ is irrational.

The second inequality, $BW(MOS_{j,j}, M_2)/j^2 \le \sqrt{2} - 1 + o(1)$, follows from the same two lemmas and the fact that, as j goes to infinity, $f(x, y)$ converges to the minimal value of f. □

Lemmas 2.13, 2.16 and 2.19 imply:

Theorem 2.20 *For* $1 < n$,

$$2(\sqrt{2} - 1)n < BW(B_n) \le 2(\sqrt{2} - 1)n + o(n).$$

3 Expansion of W_n and B_n

Complete proofs for the theorems in this section can be found in [7].

The *down-tree* T_u is an n-leaf complete binary tree rooted at node u of W_n. Let node u be in level i of W_n. Tree T_u is a subgraph of W_n such that the j^{th} level of T_u consists of nodes in level $(i + j) \bmod \log n$ of W_n. Note that the leaves of T_u also belong to level i of W_n. The *up-tree* T'_u is defined in a similar fashion.

Lemma 3.1 *The edge-expansion function* $EE(W_n, k)$ *is at most* $(4 + o(1))k/\log k$, *for* $1 \le k \le N$.

Proof: Let A be a sub-butterfly of W_n with k nodes. Each level of the sub-butterfly A has $(1 + o(1))k/\log k$ nodes. Each input and output node of sub-butterfly A has two incident edges that belong to cut $(A, \bar{A})$. Thus, the total number of edges in cut $(A, \bar{A})$ is $(4 + o(1))k/\log k$. Therefore, $EE(W_n, k) \le (4 + o(1))k/\log k$. □

Lemma 3.2 *The edge-expansion function* $EE(W_n, k)$ *is at least* $(4 + o(1))k/\log k$, *for* $k = o(n)$.

Proof: Let A be *any* set of $k = o(n)$ nodes of W_n. To prove the lemma, we use a credit distribution scheme to show that $C(A, \bar{A})$ is at least $(4 + o(1))k/\log k$. Each node $u \in A$ distributes 1 unit of credit to edges in cut $(A, \bar{A})$ using the following procedure. Let T_u be the down-tree rooted at node u. Furthermore, let the edges of T_u be directed from root to leaf. Node u passes $1/2$ unit of credit down the tree T_u using an iterative procedure. First, the two outgoing edges of u in tree T_u receive $1/4$ units of credit each. Iteratively, each tree edge (v, w) does one of the following. If tree edge (v, w) is an edge in cut $(A, \bar{A})$ or if w is a leaf of T_u, edge (v, w) *retains* all the credit it received. Otherwise, edge (v, w) retains none of the credit it received and passes half the credit it received to each of the two outgoing tree edges of w. In a similar fashion, node u distributes $1/2$ unit of credit via the up-tree T'_u rooted at u.

We bound the total amount of credit retained by the edges in cut $(A, \bar{A})$ as follows. Each node $u \in A$ distributes 1 unit of credit, of which some portion is retained by edges in cut $(A, \bar{A})$ and the rest is retained by edges (v, w) such that $w \in \bar{A}$ is a leaf of T_u or T'_u. If an edge (v, w) not in cut $(A, \bar{A})$ retains credit from node u then there is a path of length $\log n$ from node u to w such that every node in the path belongs to A. Note that a node w may appear as a leaf in both T_u and T'_u. Since there are at most k nodes $w \in A$, there are at most $2k$ edges not in cut $(A, \bar{A})$ that retain credit from node u, and each such edge retains $1/2^{\log n+1} = 1/(2n)$ units of credit from u. Thus the amount of credit from node $u \in A$ that is retained by edges in cut$(A, \bar{A})$ is at least $1 - 2k/2n = 1 - k/n$. Since there are k nodes in A, the total units of credit retained by the edges in cut $(A, \bar{A})$ is at least

$$k\left(1 - \frac{k}{n}\right) = (1 + o(1))k, \tag{1}$$

since $k = o(n)$.

Next, we show that each edge in cut $(A, \bar{A})$ retains a total of at most $(\lfloor \log k \rfloor + 1)/4$ units of credit. Let (v, w) be a cut edge such that $v \in A$ and $w \in \bar{A}$. Without loss of generality, let nodes v and w be in levels i and $(i+1) \bmod \log n$ of W_n respectively. Let T'_v be the up-tree rooted at node v. The cut edge (v, w) retains the most number of credits when *all* the k nodes in A are placed in the first $\lfloor \log k \rfloor + 1$ levels of tree T'_v as close to v as possible, i.e., when all nodes in levels 0 through $\lfloor \log k \rfloor - 1$ and some nodes in level $\lfloor \log k \rfloor$ of tree T'_v are in A. Since each node $u \in A$ at level j of tree T'_v contributes $1/2^{j+2}$ credits to cut edge (v, w), and since there are 2^j nodes in level j of tree T'_u, the total units of credit retained by cut edge (v, w) is at most

$$\sum_{j=0}^{\lfloor \log k \rfloor} \left(2^j \cdot \frac{1}{2^{j+2}}\right) = \frac{\lfloor \log k \rfloor + 1}{4} \tag{2}$$

It follows from Equations 1 and 2 that the number of edges in cut $(A, \bar{A})$ is at least

$$C(A, \bar{A}) \ge (1 + o(1))k \cdot \frac{4}{\lfloor \log k \rfloor + 1} = (4 + o(1))\frac{k}{\log k}$$

□

Theorem 3.3 *The node-expansion function* $NE(W_n, k)$ *is* $\Theta(k/\log k)$.

Theorem 3.4 *The edge-expansion function* $EE(B_n, k)$ *is* $(2 + o(1))k/\log k$, *when* $k = o(\sqrt{n})$.

Theorem 3.5 *The node-expansion function* $NE(B_n, k)$ *is* $\Theta(k/\log k)$.

The lower bound on $EE(W_n, k)$ cannot be extended to hold for all values of k up to $N/2$ because for $k = N/2$, the value of the expansion function cannot exceed the bisection width of W_n, which we showed in [7] to be $BW(W_n) = n = (1 + o(1))N/\log N$. Hence $EE(W_n, N/2) \le BW(W_n) = (2 + o(1))(N/2)/\log(N/2)$, which is smaller than our bound for $k = o(n)$ by a factor of about 2. For larger values of k, however, we can use the technique of embedding K_N into W_n, which gives a bound of $EE(W_n, k) = \Omega(k/\log n)$. For $k = n^\epsilon$, for any fixed $\epsilon > 0$, this lower bound is $\Omega(k/\log k)$. Hence, for all values of k, $EE(W_n, k) = \Theta(k/\log k)$. Our bounds for $NE(W_n, k)$ are not as tight. We show that $(1 + o(1))k/\log k \leq NE(W_n, k) \leq (3 + o(1))k/\log k$, for $k = o(n)$. For larger values of k we can again use the technique of embedding K_N into W_n, which yields $NE(W_n, k) = \Theta(k/\log k)$ over the whole range of k. A similar argument shows that the lower bound on $EE(B_n, k)$ cannot be extended to hold for all values of k up to $N/2$, but that for all values of k, $EE(B_n, k) = \Theta(k/\log k)$. Our bounds for $NE(B_n, k)$ are not as tight, but we can show that $NE(W_n, k) = \Theta(k/\log k)$ over the whole range of k.

References

[1] M. Ajtai, J. Komlós, and E. Szemerédi. Sorting in $c \log n$ parallel steps. *Combinatorica*, 3:1–19, 1983.

[2] S. Arora, F. T. Leighton, and B. M. Maggs. On-line algorithms for path selection in a non-blocking network. *SIAM Journal on Computing*, 25(3):600–625, June 1996.

[3] A. Avior, T. Calamoneri, S. Even, A. Litman, and A. L. Rosenberg. A tight layout of the butterfly network. In *Proceedings of the 8th ACM Symposium on Parallel Algorithms and Architectures*, pages 170–175, June 1996.

[4] Butterfly™ Parallel Processor Overview. BBN Report No. 6148, Version 1, BBN Advanced Computers, Inc., Cambridge, MA, Mar. 1986.

[5] B. Beizer. The analysis and synthesis of signal switching networks. In *Proceedings of the Symposium on Mathematical Theory of Automata*, pages 563–576, Brooklyn, NY, 1962. Brooklyn Polytechnic Institute.

[6] V. E. Beneš. Optimal rearrangeable multistage connecting networks. *Bell System Technical Journal*, 43:1641–1656, July 1964.

[7] C. F. Bornstein, A. Litman, B. M. Maggs, R. K. Sitaraman, and T. Yatzkar. On the bisection width and expansion of butterfly networks. Technical Report CMU–CS–97–132, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, May 1997.

[8] R. Feldmann and W. Unger. The cube-connected cycles network is a subgraph of the butterfly network. *Parallel Processing Letters*, 2(1):13–19, 1992.

[9] B. Ghosh, F. T. Leighton, B. M. Maggs, S. Muthukrishnan, C. G. Plaxton, R. Rajaraman, A. W. Richa, R. E. Tarjan, and D. Zuckerman. Tight analyses of two local load balancing algorithms. In *Proceedings of the 27th Annual ACM Symposium on the Theory of Computing*, pages 548–558, May 1995.

[10] A. Gottlieb. An overview of the NYU Ultracomputer Project. In J. J. Dongarra, editor, *Experimental Parallel Computing Architectures*, pages 25–95. Elsevier Science Publishers, B. V., Amsterdam, The Netherlands, 1987.

[11] D. S. Greenberg, L. S. Heath, and A. L. Rosenberg. Optimal embeddings of butterfly-like graphs in the hypercube. *Mathematical Systems Theory*, 23:61–77, 1990.

[12] R. Koch, T. Leighton, B. Maggs, S. Rao, and A. Rosenberg. Work-preserving emulations of fixed-connection networks. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 227–240, May 1989.

[13] F. T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays • Trees • Hypercubes*. Morgan Kaufmann, San Mateo, CA, 1992.

[14] F. T. Leighton, M. Lepley, and G. L. Miller. Layouts for the shuffle-exchange graph based on the complex plane diagram. *SIAM Journal of Algebraic and Discrete Methods*, 5(2):202–215, June 1984.

[15] F. T. Leighton and B. M. Maggs. Fast algorithms for routing around faults in multibutterflies and randomly-wired splitter networks. *IEEE Transactions on Computers*, 41(5):578–587, May 1992.

[16] B. M. Maggs and B. Vöcking. Improved routing and sorting on multibutterflies. In *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing*, pages 517–530, May 1997.

[17] Y. Manabe, K. Hagihara, and N. Tokura. The minimum bisection widths of the cube-connected cycles graph and the cube graph. *Transactions of the IEICE (D)*, J67-D(6):647–654, June 1984. In Japanese.

[18] T. Nakata, S. Matsushita, N. Tanabe, N. Kajihara, H. Onozuka, Y. Asano, and N. Koike. Parallel programming on Cenju: A multiprocessor system for modular circuit simulation. *NEC Research & Development*, 32(3):421–429, July 1991.

[19] W. Oed. Cray Y-MP C90: system features and early benchmark results. *Parallel Computing*, 18(8):947–954, Aug. 1992.

[20] G. F. Pfister, W. C. Brantley, D. A. George, S. L. Harvey, W. J. Kleinfelder, K. P. McAuliffe, E. A. Melton, V. A. Norton, and J. Weiss. An introduction to the IBM Research Parallel Processor Prototype (RP3). In J. J. Dongarra, editor, *Experimental Parallel Computing Architectures*, pages 123–140. Elsevier Science Publishers, B. V., Amsterdam, The Netherlands, 1987.

[21] F. P. Preparata and J. Vuillemin. The cube-connected cycles: a versatile network for parallel computation. *Communications of the ACM*, 24(5):300–309, May 1981.

[22] K. J. Rappoport. On the slowdown of efficient simulations of multibutterflies. In *Proceedings of the 8th ACM Symposium on Parallel Algorithms and Architectures*, pages 176–182, June 1996.

[23] E. J. Schwabe. On the computational equivalence of hypercube-derived networks. In *Proceedings of the 2nd Annual ACM Symposium on Parallel Algorithms and Architectures*, pages 388–397, July 1990.

[24] C. D. Thompson. *A Complexity Theory for VLSI*. PhD thesis, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, PA, 1980.

[25] J. Turner and N. Yamanaka. Architectural choices in large scale ATM switches. Technical Report WUCS–97–21, Department of Computer Science, Washington University, St. Louis, MO, 1997.

[26] E. Upfal. An $O(\log N)$ deterministic packet routing scheme. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, pages 241–250, May 1989.

[27] A. Waksman. A permutation network. *Journal of the ACM*, 15(1):159–163, Jan. 1968.