The Asymptotic Complexity of Merging Networks *
(Extended Abstract)

Peter Bro Miltersen † Mike Paterson ‡ Jun Tarui ‡

January 16, 1995

Abstract

Let $M(m,n)$ be the minimum number of comparators in a comparator network that merges two ordered chains $x_1 \leq x_2 \leq \ldots \leq x_m$ and $y_1 \leq y_2 \ldots \leq y_n$, where $n \geq m$. Batcher's odd-even merge yields the following upper bound:

$$M(m,n) \leq \frac{1}{2}(m+n)\log_2(m+1) + O(n), \text{ e.g., } M(n,n) \leq n\log_2 n + O(n).$$

Floyd (for $M(n,n)$), and then Yao and Yao (for $M(m,n)$) have shown the following lower bounds:

$$M(m,n) \geq \frac{1}{2}n\log_2(m+1); \quad M(n,n) \geq \frac{1}{2}n\log_2 n + O(n).$$

We prove a new lower bound that matches the upper bound asymptotically:

$$M(m,n) \geq \frac{1}{2}(m+n)\log_2(m+1) - O(m), \text{ e.g., } M(n,n) \geq n\log_2 n - O(n).$$

Our proof technique extends to give similarly tight lower bounds for the size of monotone Boolean circuits for merging, and for the size of switching networks capable of realizing the set of permutations that arise from merging.

1 Introduction and Overview

Merging networks (for a definition, see Section 2) together with sorting networks, have been studied extensively. ([Knu73, pages 220–246] is a good reference on the subject.)

Let $M(m,n)$ denote the minimum number of comparators in a comparator network that merges two input sequences $x_1 \leq x_2 \leq \ldots \leq x_m$ and $y_1 \leq y_2 \leq \ldots \leq y_n$ into the sequence $z_1 \leq z_2 \leq \ldots \leq z_{m+n}$. Batcher's odd-even merge [Knu73, pp. 224–226] provides the best known upper bound for $M(m,n)$ for *all* values of m,

*This work was partially supported by the ESPRIT II BRA Programme of the EC under contract # 7141 (ALCOM II).

†Department of Computer Science, Aarhus University, Ny Munkegade, 8000 Aarhus C, Denmark (**pbmiltersen@daimi.aau.dk**).

‡Department of Computer Science, University of Warwick, Coventry, CV4 7AL, United Kingdom (Paterson: **msp@dcs.warwick.ac.uk**, Tarui: **jun@dcs.warwick.ac.uk**).

$n \geq 1$. Throughout the paper we assume $n \geq m$, and all logarithms have base 2. If $C(m,n)$ denotes the number of comparators in Batcher's network for (m,n) then

$$M(m,n) \leq C(m,n) = \frac{1}{2}(m+n)\log m + O(n),$$

and, in particular,

$$M(n,n) \leq n\log n + O(n).$$

The previous best lower bounds for $M(n,n)$ and $M(m,n)$ are due to Floyd [Knu73, pp. 230–232] and to Yao and Yao [YY76] who proved, respectively,

$$M(n,n) \geq \frac{1}{2}n\log n + O(n) \qquad \text{and} \qquad M(m,n) \geq \frac{1}{2}n\log(m+1).$$

We close this long-standing factor-of-two gap between the previous best lower and upper bounds for $M(n,n)$ and show that the asymptotic value of $M(n,n)$ is $n\log n$, by proving the following lower bound:

$$M(m,n) \geq \frac{1}{2}(m+n)\log(m+1) - 0.73m.$$

Our lower bound arguments only involve the total path length, and thus we can extend the result to the general framework considered by Pippenger and Valiant [PV76], showing that any graph with in-degree two, which is capable of realizing all the merging patterns, has many vertices. In particular, our lower bound for merging networks also holds for the number of switches in a switching network that can realize all the connections from inputs to outputs that arise from merging. We also obtain a tight lower bound for the size of monotone Boolean circuits for merging, improving the best previous lower bound essentially by a factor of two, in the same way that our lower bound for $M(n,n)$ improves Floyd's lower bound.

1.1 Overview of Proof

The main ideas involved in the proof of our main theorem (Theorem 1) are first described informally. For simplicity, we explain our arguments in terms of merging networks.

Assume that two input sequences $x_1 < x_2 < \ldots < x_m$ and $y_1 < y_2 < \ldots < y_n$ are given and that $x_i \neq y_j$ for all $1 \leq i \leq m, 1 \leq j \leq n$. Imagine the x_i's and y_j's actually moving through a merging network to their destination z_k's. Let $\mathrm{Merge}(m,n)$ be the set of $\binom{m+n}{m}$ possible merging patterns. We define a probability distribution on $\mathrm{Merge}(m,n)$ such that the expected total path length, which equals the sum over the z_k's of the expected length of the path reaching z_k, can be shown to be large. It follows that there exists a merging pattern under which the total path length is large and, since only two inputs go through each comparator, there must be many comparators. For each z_k there is a certain probability distribution on the set of x_i's and y_j's that arrive at z_k, and the *expected length* of the path reaching z_k is at least the *entropy* of this distribution.

There is the natural bijection from $\mathrm{Merge}(m,n)$ onto the set of up-or-right paths from $(0,0)$ at the lower-left corner to (m,n) at the upper-right corner of the $n \times m$ grid, and the probability distribution on $\mathrm{Merge}(m,n)$ can be thought of as a *unit flow*

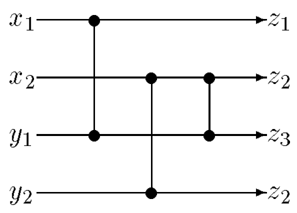

Figure 1: Batcher's odd-even network for $m = n = 2$

on the grid. Conversely, a unit flow on the grid can be converted to a probability distribution on Merge(m, n) by considering a Markov random walk on the grid according to the flow. Which x_i or y_j reaches z_k is determined by which edge is traversed at the diagonal corresponding to z_k.

We would like to determine the unit flow F on the grid which maximizes the sum of the entropies, $H(F)$. First we consider the flow U that maintains the uniform distribution on the *vertices* along each diagonal, and evaluate $H(U)$. All the lower bounds in this extended abstract are based on $H(U)$. We can show that there is a unique optimal flow Z maximizing the total entropy. The improvement of $H(Z)$ over $H(U)$ is only in terms of reducing the coefficient of the linear term in Theorem 1 and its corollaries, i.e., we can get about $-1.3m$ instead of $-1.45m$ in Theorem 1. Let Z_n be the optimal flow on the $n \times n$ grid. We have no closed formula for Z_n or $h(n) = H(Z_n)$, but we can establish two recurrence inequalities for $h(n)$, bounding it rather accurately from above and below. Detailed discussions of the optimal flow and $h(n)$ are omitted from this extended abstract.

1.2 Outline of Paper

In Section 2 we explain the general framework in which we work, state the main theorem (Theorem 1), and explain how our lower bounds follow as its corollaries. In Section 3 we prove the main theorem. In Section 4 we discuss the optimal flow and the slight improvement of lower bounds that it yields. Finally in Section 5 we state some open problems related to this work.

2 Results

A *comparator network* is a directed graph in which there are k vertices $s_1, \ldots, s_k$ of in-degree 0 and out-degree 1 called *inputs* and k vertices $t_1, \ldots, t_k$ of in-degree 1 and out-degree 0 called *outputs*, and the rest of the vertices, called *comparators*, have in-degree two and out-degree two. (See Figures 1 and 2.)

We shall denote by s_i (t_j) both an input (output) vertex and the value assigned to s_i as input (or to t_j as output). For an arbitrary totally ordered set D and $s_1, \ldots, s_k \in D$, each edge in a comparator network, and hence each of $t_1, \ldots, t_k$, can be assigned some value in D in the natural way: if a and b are the values computed by the two incoming edges of a comparator C, one outgoing edge of C

computes $\max\{a,b\}$ and the other computes $\min\{a,b\}$. An (m,n)-*merging network* is a comparator network with $m+n$ inputs $x_1,\ldots,x_m$, $y_1,\ldots,y_n$ and $m+n$ outputs $z_1,\ldots,z_{m+n}$ such that if $x_1 \leq x_2 \leq \cdots \leq x_m$ and $y_1 \leq y_2 \leq \cdots \leq y_n$ then $z_1 \leq z_2 \leq \cdots \leq z_{m+n}$.

Our main theorem is in terms of the following general framework considered by Pippenger and Valiant [PV76].

Let $G=(V,E)$ be a directed graph, and $S=\{s_1,\ldots,s_k\}$ and $T=\{t_1,\ldots,t_k\}$ be disjoint sets of vertices. We say that G *realizes* a set M of bijections from T onto S if for each $\pi \in M$ there are k vertex-disjoint paths $p_1,\ldots,p_k$ in G, where p_i is from $\pi(t_i)$ to t_i, for $1 \leq i \leq k$.

If $S=\{x_1,\ldots,x_m,y_1,\ldots,y_n\}$ and $T=\{z_1,\ldots,z_{m+n}\}$, then *Merge*(m,n) is the set of $\binom{m+n}{m}$ bijections from T onto S that arise in the following way. If D is a totally ordered set and $f: S \rightarrow D$ is an injective map assigning values to vertices so that $f(x_1) < \cdots < f(x_m)$ and $f(y_1) < \cdots < f(y_n)$, then we get a bijection $\pi \in Merge(m,n)$ defined by: $\pi(z_i)=$the unique $w \in S$ with $\text{rank}(w)=i$. A graph G together with $S,T \subseteq V$ $(S \cup T = \emptyset, |S|=|T|=m+n)$ is an (m,n)-*merging graph* if it realizes $\text{Merge}(m,n)$.

We can now state our main theorem.

Theorem 1. *If $G=(V,E)$ together with S, $T \subseteq V$ is an (m,n)-merging graph with in-degree at most two, then*

$$\begin{aligned} |V-S| &\geq (m+n)\log_2(m+1) - (\log_2 e)m \\ &\geq (m+n)\log_2(m+1) - 1.45m. \end{aligned}$$

Applications of our main theorem become obvious when we consider min-max circuits. A *min-max circuit* is a combinatorial circuit with gates of fan-in two and of unbounded fan-out, where each gate is either a MIN gate or a MAX gate that computes the minimum or the maximum of two inputs respectively. A min-max circuit with inputs $x_1,\ldots,x_m$ and $y_1,\ldots,y_n$ and outputs $z_1,\ldots,z_{m+n}$ is said to (m,n)-*merge* if it computes the merge of the x_i's and the y_j's at the z_k's.

The following observations are easy.

Lemma 1. *If a min-max circuit (m,n)-merges, then its underlying graph is an (m,n)-merging graph with in-degree at most 2.*

Proof : Omitted from this abstract.

Fact 1. *A merging network N can be converted to a min-max merging circuit C by replacing each comparator by a MIN and a MAX gate. (See Figure 2.) The number of gates in C is twice the number of comparators in N.*

From Lemma 1 and Fact 1, we get our lower bound for merging networks as a corollary of Theorem 1.

Corollary 1.

$$M(m,n) \geq \frac{1}{2}(m+n)\log(m+1) - 0.73m.$$

The same bound holds even when we allow outgoing edges of comparators to branch.

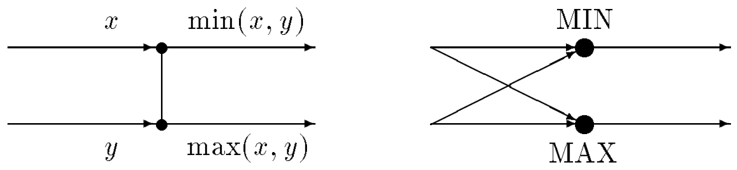

Figure 2: A comparator and an equivalent pair of MIN/MAX gates

2.1 Monotone Circuit Complexity of Merging

Consider a monotone Boolean circuit with $m+n$ inputs and $m+n$ outputs that computes the merge of two sequences of lengths m and n. The following two observations yield a lower bound on the number of AND and OR gates needed.

Fact 2. *If C is a monotone Boolean circuit for Boolean merging, we can transform C to a min-max circuit of the same size that merges Boolean inputs, by replacing each AND or OR gate with a MIN or MAX gate respectively.*

The next lemma is the "0-1 principle" for merging.

Lemma 2. *If a min-max circuit merges every pair of Boolean sequences of length m and n, then it is an (m, n)-merging min-max circuit.*

Using Fact 2, Lemma 2, and Lemma 1, we obtain the following as a corollary of Theorem 1.

Corollary 2. *Any monotone Boolean circuit that computes the merge of two Boolean sequences of length m and n has at least $(m+n)\log m - 1.45m$ gates.*

The previous best lower bounds are due to Lamagna [Lam] and, independently, to Pippenger and Valiant [PV76]. Their bounds are essentially half our bounds when $m = n$, as in the case of merging networks.

3 Proof of Theorem 1

In this section we prove Theorem 1 by relating merging graphs with a network flow problem.

3.1 Entropies

Suppose that $G = (V, E)$ with $S = \{x_1, \ldots, x_m, y_1, \ldots, y_n\}$ and $T = \{z_1, \ldots, z_{m+n}\}$ is an (m, n)-merging graph, and let $M = \mathrm{Merge}(m, n)$. For each $\pi \in M$, fix a sequence $\langle P_i^\pi : i = 1, \ldots, m+n \rangle$ of $m+n$ vertex-disjoint paths in G, where P_i^π is a path from $\pi(z_i)$ to z_i, for $1 \le i \le m+n$. For each vertex of in-degree two in G, fix arbitrarily which incoming edge is "left" and which is "right." For each $\pi \in M$ and $i \in \{1, \ldots, m+n\}$, encode P_i^π by following the path in the reverse direction from z_i to $\pi(z_i)$ and using, say, 0 for left and 1 for right. Let C_i^π be the binary code for P_i^π obtained this way. For each $i \in \{1, \ldots, m+n\}$, the set $\{C_i^\pi : \pi \in M\}$ gives an instantaneous decipherable binary coding for $\{\pi(z_i) : \pi \in M\}$. (There may be more than one code for some x_j or y_k.) Let a probability distribution on M be given and

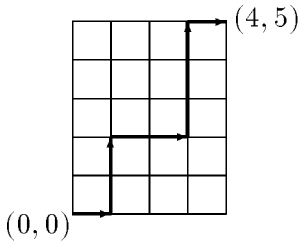

Figure 3: The merge $x_1 < y_1 < y_2 < x_2 < x_3 < y_3 < y_4 < y_5 < x_4$

consider $\pi(z_i)$ and P_i^π, for $1 \le i \le m+n$, as random variables accordingly. For a random variable X, let $H(X)$ denote the entropy of X measured in bits, and let $E[X]$ denote its expectation. For a path P in a graph and a binary code C, let $|P|$ and $|C|$ denote their lengths. Then

$$\begin{aligned} \sum_{i=1}^{m+n} H(\pi(z_i)) \quad \le \quad \sum_{i=1}^{m+n} E[|C_i^\pi|] \quad \le \quad \sum_{i=1}^{m+n} E[|P_i^\pi|] \quad = \quad E\left[\sum_{i=1}^{m+n} |P_i^\pi|\right] \\ \le \quad \max_{\pi \in M} \sum_{i=1}^{m+n} |P_i^\pi| \quad \le \quad |V - S|, \end{aligned}$$

where the first inequality is by the well-known Shannon's Theorem for a noiseless channel and a discrete memoryless source (see any textbook on information theory), and the equality is by the linearity of expectations.

We obtain our lower bound for $|V - S|$ by defining a certain distribution on M and evaluating $\sum_{i=1}^{m+n} H(\pi(z_i))$ with respect to it.

3.2 Unit Flow on a Grid

Consider the grid with coordinates as shown in Figure 3, and let M' be the set of directed paths from $(0,0)$ to (m,n) of length $m+n$ that move right or upward from each vertex. We will simply say 'path' when we mean such a path from $(0,0)$ to (m,n). The natural bijection from M onto M' is illustrated in Figure 3.

Any distribution on M induces a distribution on M'. Under this induced distribution on M', define $\alpha_{i,j}$ for $1 \le i \le m$, $0 \le j \le n$, to be the probability that path p passes through the edge from $(i-1,j)$ to (i,j), and similarly define $\beta_{i,j}$, for $0 \le i \le m$, $1 \le j \le n$, to be the probability that p passes through the edge from $(i,j-1)$ to (i,j). (See Figure 4.) For convenience, we define $\alpha_{i,j} = 0$ for $i = 0, m+1$ and $0 \le j \le n$, and $\beta_{i,j} = 0$ for $j = 0, n+1$ and $0 \le i \le m$. Jointly $\alpha_{i,j}$ and $\beta_{i,j}$ define a *unit flow* from $(0,0)$ to (m,n), that is, the following equations are satisfied:

$$\alpha_{1,0} + \beta_{0,1} = 1; \quad \alpha_{m,n} + \beta_{m,n} = 1;$$

$$\alpha_{i,j} + \beta_{i,j} = \alpha_{i+1,j} + \beta_{i,j+1} \quad \text{for } (i,j) \ne (0,0), (m,n).$$

The equations above express the fact that one unit goes out of the source $(0,0)$, one unit goes into the sink (m,n), and the flow is conserved at the other vertices.

We define $\gamma_{i,j}$, the flow through vertex (i,j), as follows:

$$\gamma_{i,j} = \alpha_{i,j} + \beta_{i,j} \quad \text{for } (i,j) \ne (0,0), \quad \text{and } \gamma_{0,0} = 1.$$

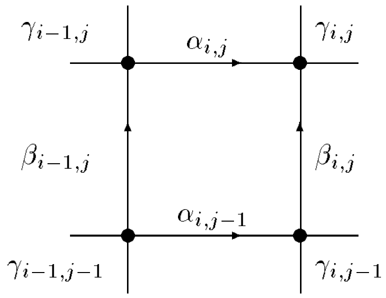

Figure 4: Names of edge- and vertex-flows

Note that:

$$\alpha_{i,j} = \mathrm{Prob}[\pi(z_{i+j}) = x_i]; \quad \beta_{i,j} = \mathrm{Prob}[\pi(z_{i+j}) = y_j];$$

$$\gamma_{i,j} = \mathrm{Prob}[\{\pi(z_k) : 1 \le k \le i+j\} = \{x_1, \ldots, x_i, y_1, \ldots, y_j\}].$$

From the first two equations we get:

$$\begin{aligned} \sum_{k=1}^{m+n} H(\pi(z_i)) &= -\sum_{k=1}^{m+n} \sum_{i+j=k} \alpha_{i,j} \log \alpha_{i,j} + \beta_{i,j} \log \beta_{i,j} \\ &= -\sum_{i=1}^{m} \sum_{j=0}^{n} \alpha_{i,j} \log \alpha_{i,j} - \sum_{i=0}^{m} \sum_{j=1}^{n} \beta_{i,j} \log \beta_{i,j}. \end{aligned}$$

(As usual we take $x \log x$ to be 0 when $x = 0$.) For a unit flow $F = (\alpha, \beta)$ from $(0,0)$ to (m,n), define the *entropy*, $H(F)$, to be the quantity expressed above.

Above we have described the map Γ from the set of distributions on the paths of the grid to the set of unit flows. To see that this map Γ is surjective, let $F = (\alpha, \beta)$ be any unit flow. Consider a random walk from $(0,0)$ to (m,n) that behaves as follows. At vertex (i,j), visited with probability $\gamma_{i,j}$, move right with probability $q = \alpha_{i+1,j}/(\alpha_{i+1,j} + \beta_{i,j+1})$ and move upward with probability $1-q$. It is easy to see that the distribution on paths defined by this random walk gets mapped by Γ to the original flow F.

Thus $\sup\{H(F) : F \text{ unit flow}\}$ is a lower bound for $|\{v \in V : \text{in-degree}(v) = 2\}|$. There exists in fact a unique *optimal* flow Z that attains the supremum. In this extended abstract, we obtain our lower bound from U, a nice near-optimal flow, such that $H(U)$ and $H(Z)$ differ only in the coefficient of the linear term. In Section 4 we sketch the additional arguments needed to determine $H(Z)$ for the $n \times n$ grid.

3.3 Diagonally Uniform Flow

Consider the following flow $U = (\alpha, \beta)$. From $(0,0)$, U maintains a uniform distribution on the diagonals $i+j = 1, 2, \ldots$, i.e., $\alpha_{i,j} + \beta_{i,j} = \gamma_{i,j} = 1/(i+j+1)$, until the diagonal $i+j = m$. Then U maintains the flow of $1/(m+1)$ along each vertical line until the diagonal $i+j = n$. U "converges" to (m,n) from this diagonal in the same way that U "diverges" from $(0,0)$ to the diagonal $i+j = m$. (See Figure 5.)

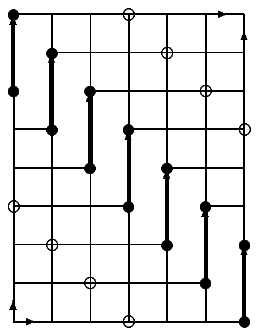

Figure 5: The flow U on the 8×6 grid.

More precisely U can be expressed as follows:

$$\alpha_{i,j} = \beta_{j,i} = \alpha_{m+1-i,n-j} = \beta_{m-j,n+1-i} = \frac{i}{(i+j)(i+j+1)} \quad \text{for } 1 \le i+j \le m;$$

$$\alpha_{i,j} = 0, \quad \beta_{i,j} = 1/(m+1), \quad \text{for } m < i+j \le n.$$

It is easy to verify that these equations do indeed define a unit flow. By symmetry,

$$\begin{aligned}
H(U) &= 4\sum_{d=1}^{m}\sum_{i=1}^{d}(-\beta_{i,d-i}\log\beta_{i,d-i}) - (n-m)(m+1)\frac{1}{m+1}\log\frac{1}{m+1} \\
&= 4\sum_{d=1}^{m}\sum_{i=1}^{d}\frac{-i}{d(d+1)}\log\frac{i}{d(d+1)} + (n-m)\log(m+1) \\
&= 2\sum_{d=1}^{m}(\log d + \log(d+1)) - 4\sum_{d=1}^{m}\frac{1}{d^2+d}\sum_{i=1}^{d} i\log i + (n-m)\log(m+1) \\
&= 4\log(m+1)! + (n-m-2)\log(m+1) - 4\sum_{d=1}^{m}\frac{1}{d^2+d}\sum_{i=1}^{d} i\log i.
\end{aligned}$$

Evaluating the summation, we get

$$\begin{aligned}
&\sum_{d=1}^{m}\frac{1}{d^2+d}\sum_{i=1}^{d} i\log i \\
\le\ &\sum_{d=1}^{m}\frac{1}{d^2+d}\sum_{i=1}^{d}\left(\frac{(i+1)i}{2}\log\frac{i+1}{\sqrt{e}} - \frac{i(i-1)}{2}\log\frac{i}{\sqrt{e}}\right) \\
=\ &\sum_{d=1}^{m}\frac{1}{d^2+d}\frac{d^2+d}{2}\log\frac{d+1}{\sqrt{e}} \\
=\ &\frac{1}{2}\log(m+1)! - \frac{1}{4}m\log e.
\end{aligned}$$

So, using Stirling's formula, we bound $H(U)$ as follows:

$$\begin{aligned}
H(U) &\ge 2\log(m+1)! + (n-m-2)\log(m+1) + m\log e \\
&\ge 2(m+1)(\log(m+1) - \log e) + \log(2\pi(m+1)) \\
&\qquad +(n-m-2)\log(m+1) + m\log e \\
&\ge (m+n)\log(m+1) - m\log e \quad \text{for} \quad m \ge 1.
\end{aligned}$$

The proof of Theorem 1 is complete.

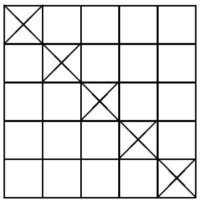

Figure 6: Squares where U does *not* satisfy the local condition on the 5×5 grid.

4 Optimal Flow

4.1 Characterization of Optimal Flow

By analytic arguments, we can show the following.

Proposition 1. *A flow $F = (\alpha, \beta)$ is optimal if and only if $F = (\alpha, \beta)$ satisfies the following local conditions: for $1 \le i \le m, 1 \le j \le n$,*

$$\alpha_{i,j-1}\beta_{i,j} = \beta_{i-1,j}\alpha_{i,j}.$$

There is a unique optimal unit flow U such that $H(U) = \sup\{H(F) : F \ \text{ unit flow}\}$.

Proof sketch : Here we only prove "only if", our main intention being to explain where the local condition above comes from. Let $F = (\alpha, \beta)$ be an optimal flow. We can show that F has nonzero value on every edge. Suppose that $1 \le i_0 \le m$ and $1 \le j_0 \le n$, and let $F(t) = (\alpha(t), \beta(t))$ be the flow defined as follows: $\alpha_{i,j}(t) = \alpha_{i,j}$ and $\beta_{i,j}(t) = \beta_{i,j}$ for all (i, j) except the following four pairs, where

$$\begin{aligned} \alpha_{i_0,j_0}(t) &= \alpha_{i_0,j_0} - t, & \alpha_{i_0,j_0-1}(t) &= \alpha_{i_0,j_0-1} + t, \\ \beta_{i_0,j_0}(t) &= \beta_{i_0,j_0} + t, & \beta_{i_0-1,j_0}(t) &= \beta_{i_0-1,j_0} - t. \end{aligned}$$

$F(t)$ is defined for $|t| \le \min\{\alpha_{i_0,j_0}, \beta_{i_0,j_0}, \alpha_{i_0,j_0-1}, \beta_{i_0-1,j_0}\}$, and corresponds to a local change by t of the flow around the cell with (i_0, j_0) at its upper-right corner (see Figure 4).

Since F is optimal, the derivative of $H(F(t))$ with respect to t at 0 must be 0. But

$$\frac{dH(F(t))}{dt}(0) = -\log \alpha_{i_0,j_0-1} - \log \beta_{i_0,j_0} + \log \beta_{i_0-1,j_0} + \log \alpha_{i_0,j_0},$$

and so F satisfies the local condition above for each i and j. □

4.2 Improvement by Optimal Flow

Let Z_n be the unique optimal flow on the $n \times n$ grid, and let $h(n) = H(Z_n)$. Let U_n be the uniform flow considered in Section 3.3, and recall that $H(U_n) \approx 2n \log n - 1.45n$. The flow U_n is not optimal since it does not satisfy the local condition above for the cells on the main diagonal, where $i + j = n$. The condition *is* satisfied at all the other cells. (See Figure 6.)

Although we have no closed formula for Z_n or $h(n)$, we can show that $h(n) = 2n \log n - cn + o(n)$, where $c \approx 1.3$. Thus using Z instead of U, we can slightly improve our lower bound in Theorem 1 and its corollaries.

5 Conclusion and Open Problems

It has been conjectured that Batcher's (m, n)-network exactly optimal for all m, n. Yao and Yao [YY76] have shown that $M(2, n) = C(2, n)$, and so Batcher's networks are optimal for $m = 2$, however the *exact* behavior of $M(m, n)$ for $m > 2$ remains an open problem.

The results proved in this paper take a major step towards establishing the conjecture. We have shown that the asymptotic value of $M(m, n)$ is $(m+n)\log(m+1)$, and hence that Batcher's networks are *asymptotically* optimal.

References

[Knu73] D. Knuth, *The Art of Computer Programming*, Volume 3: Sorting and Searching, Addison-Wesley, 1973.

[Lam] E. Lamagna, "The complexity of monotone networks for certain bilinear forms, routing problems, sorting and merging," *IEEE Trans. on Comp.*, 28(1979), 773–782.

[PV76] N. Pippenger and L. Valiant, "Shifting graphs and their applications," *J. Assoc. Comput. Mach.*, 23(1976), 423–432.

[YY76] A. Yao and F. Yao, "Lower bounds on merging networks," *J. Assoc. Comput. Mach.*, 23(1976), 566–571.