Lower Bounds on Correction Networks

Grzegorz Stachowiak

Institute of Computer Science, University of Wrocław, Przesmyckiego 20, 51-151 Wrocław, Poland (gst@ii.uni.wroc.pl)

Abstract. Correction networks are comparator networks that sort inputs differing from sorted sequences of length N in a small number of positions. The main application of such networks is producing fault-tolerant sorting networks. We show the lower bound $1.44\log_2 N$ on the depth of correction networks settling an open problem from [7]. This bound is tight since the upper bound $1.44\log_2 N$ is known.

1 Introduction

Sorting is one of the most fundamental problems of computer science. A classical approach to sort a sequence of keys is to apply a comparator network. Apart from a long tradition, comparator networks are particularly interesting due to potential hardware implementations. They can be also implemented as sorting algorithms for parallel computers.

In our approach the elements (keys) to be sorted are stored in registers $r_1, r_2, \ldots, r_N$. A *comparator* $[i : j]$ is a simple device connecting registers r_i and r_j $(i < j)$. It compares the keys they contain and if the key in r_i is bigger, it swaps the keys. The general problem is the following. At the beginning of the computations the input sequence of keys is placed in the registers. Our task is to sort the sequence of keys applying a sequence of comparators. The sequence of comparators is the same for all possible inputs. We assume that comparators connecting disjoint pairs of registers can work in parallel. Thus we arrange the sequence of comparators into a series of comparator *layers* which are sets of comparators connecting disjoint pairs of registers. The total time needed by such a network to perform its computations is proportional to the number of layers of called the network's *depth*.

Much research concerning sorting networks was done in the past. Its main goals were to minimize the depth and the total number of comparators. The most famous results are asymptotically optimal AKS [1] sorting network of depth $O(\log N)$ and more 'practical' Batcher [2] network of depth $\sim \frac{1}{2}\log^2 N$ (logarithms in this paper are binary). Due to a very large constant hidden behind big O in AKS, Batcher network has much smaller depth for practical input sizes N. Another well known result is Yao's [8] construction of an almost optimal network to select t smallest (or largest) entries of a given input of size N (*t-selection* problem). His network has depth $\log N + (1+o(1))\log t \log\log N$ and $\sim N \log t$ comparators which matches lower bounds for that problem ($t \ll N$). The analysis of comparator networks is most often based on the following useful lemma [4]

Lemma 1.1 (zero–one principle). *A comparator network is a sorting network if and only if it can sort any input consisting only of 0s and 1s.*

This lemma is the reason, why we consider inputs consisting only of 0s and 1s in the analysis of comparator networks. In this paper we deal with the problem of sorting sequences that differ by a small number of modifications t from a sorted sequence. These modifications can be either swaps between pairs of elements or changes on single positions. Such sequences we call *t-disturbed*. A comparator network that sorts such sequences we call a *t-correction* network. There are some potential applications in which we have to deal with sequences that differ not much from a sorted one. Assume we have a large sorted database with N entries. In some period of time we make t modifications of the database and want to have it sorted back by a comparator network. It can be better to design a specialized network of small depth to 'repair' the ordering and avoid using costly general sorting networks.

For the analysis of a t-correction network we can consider only t-disturbed sequences consisting of 0s and 1s. We note, that 0-1 sequence $x_1,\dots,x_N$ is t disturbed if for some index b called the *border* at most t entries in $x_1,\dots,x_b$ are 1s and at most t entries in $x_{b+1},\dots,x_N$ are 0s. These 1s (0s) we call *displaced*. We have a useful analog of zero-one principle for t-correction networks.

Lemma 1.2. *A comparator network is a t-correction network if it can sort any t-disturbed input consisting of 0s and 1s.*

The problem of construction t-correction networks arose when construction of fault-tolerant sorting networks was concerned. A comparator suffers a *passive fault* if it does nothing instead of sorting two elements. Assume that our aim is to construct a sorting network that is resistant to passive faults of any t (or less) of its comparators. One of approaches to this problem is to notice that if a 0-1 input is processed by an arbitrary sorting network suffering t passive faults, then the resulting output is t disturbed (see [6, 9]). The idea is to add to this sorting network an additional t-correction unit in order to produce a fault tolerant sorting network. An additional requirement for this unit is that it has to be resistant to errors present in itself. Detailed conditions assuring fault tolerance of the whole network can be found in [7].

A nice construction of such a unit of depth $\sim 2\log N$ for one fault can be found in Shimmler and Starke paper [6]. Later Piotrów [5] pipelined this network to obtain a unit for $t > 1$ of depth $O(\log N + t)$ having $O(Nt)$ comparators. The exact constants hidden behind these big O-s were not determined, but since Piotrów uses network [6] in his construction the constant in front of $\log N$ in $O(\log N + t)$ is at least 2. The best result as the constant in front of $\log N$ is concerned is included in [7] in which a network of depth $\alpha \log N(1+\varepsilon) + ct/\varepsilon$ is described for any $\varepsilon > 0$. In this paper we denote $\Phi = \frac{1+\sqrt{5}}{2}$, $\alpha = 1/\log\Phi = 1.44\ldots$.

Any unit making a sorting network resistant to t faults has to be a t-correction network, but we can have better t-correction networks. A nice result here is network of Kik, Kutyłowski, Piotrów [3] of depth $4\log N + O(\log^2 t \log\log N)$. The best result concerning t-correction networks if we consider the leading constant is included in [7]. The network described there has depth $\alpha \log N(1+\varepsilon) + c(\varepsilon)\log^2 t \log^2 N$.

As we mentioned our goal is to reduce the constant in front of $\log N$ in the depth of correction networks, which is most essential if t is small and N big. The best results concerning constructions of correction networks suggest α as a candidate for the tight bound for this constant. We prove in this paper that it is the case. Doing this we settle an open problem from [7].

2 One Displaced Element

In this section we consider the simplest possible case of correction network. Consider 0-1 inputs $x_1, \ldots, x_N$ that differ from sorted sequences by a single displaced 1. We call a comparator network sorting all such inputs a *simple correction network*. Note that sorting such an input means moving the single displaced 1 to the border register r_b in which at the beginning we have a 0. In this section we remind the construction from [7] of a simple correction network F_N of depth $\sim \alpha \log N$. Obviously a t-correction network for any t has to be a simple correction network. Any lower bound on a simple correction network is also a lower bound on any correction network. Conversely all the constructions of correction networks with leading constant α are based on the construction of F_N. This section serves mainly to introduce the denotations. Note, that in this section we describe F_N and the upper bound it gives in a more precise manner, than in [7].

To make easier understanding of the construction we first introduce the notion of *odd-even comparator network*. We view the registers with the smallest indexes to be on the top and those with the largest indexes to be on the bottom. A comparator $[i : j]$ has two end-registers: top r_i and bottom r_j. A comparator network is odd-even if it fulfills the following conditions:

– every comparator connects an even indexed register with an odd indexed register;
– in odd layers the bottom register of each comparator has an odd index;
– in even layers the bottom register of each comparator has an even index.

We assume, that our simple correction network F_N for inputs having a single displaced 1 is an odd-even network. Let the *distance* between registers r_i and r_j be the value $|j - i|$. Let the *length* of a comparator $[i : j]$ be the distance between r_i and r_j. To construct simple correction network F_N we make additional assumptions how it should look like:

– layer l consists of all possible comparators of length a_l (whose bottom register has parity of l);
– the sequence of comparator lengths a_l decreases as quickly as some geometric progression.

Following these additional assumptions we can formulate a necessary and sufficient condition for such a network to sort any input with a single displaced 1. Let the distance of displaced 1 contained in register r_i from the border b be the value $b - i$ (b is the border index). Assume that in some layer l the displaced 1 is not moved by the comparator of length a_l because it is compared with a non displaced 1. In the next layer it is also not moved because it can be only in the bottom register of a comparator. Starting from the next layer it can start moving layer by layer with comparators of lengths $a_{l+2}, a_{l+3}, \ldots$.

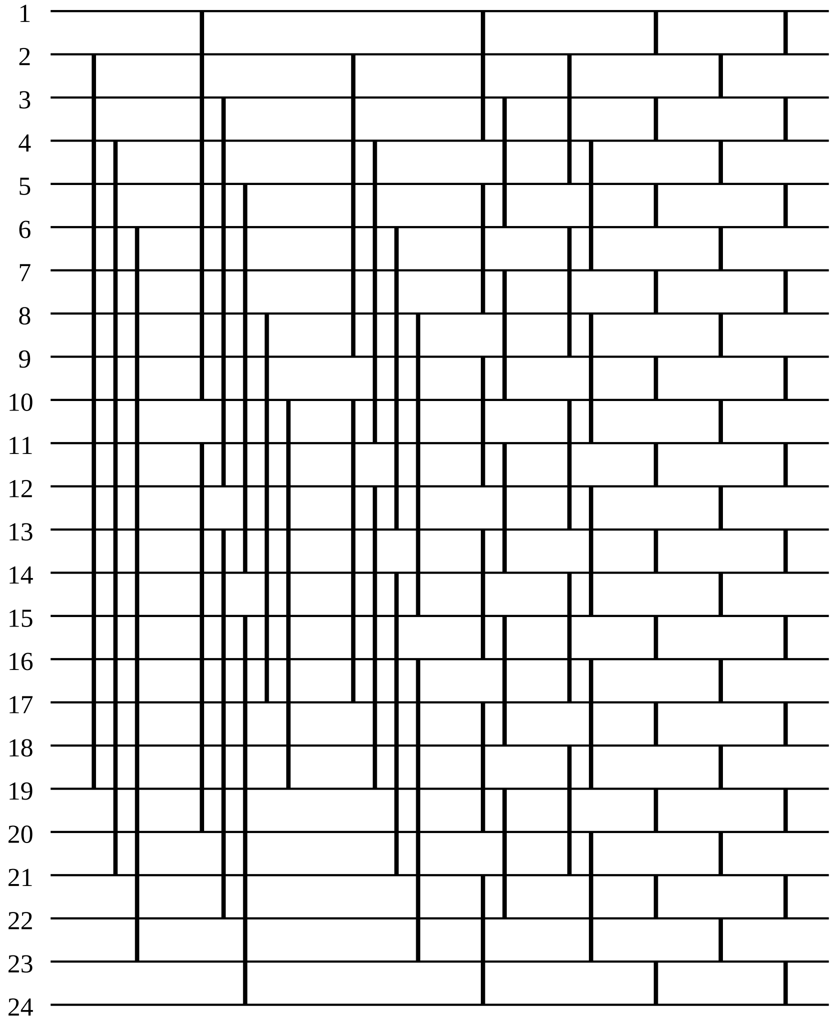

Fig. 1. Network F_{24}

The network sorts the input if and only if this 1 always gets to the border (or covers the distance $a_l - 1$). This can be expressed by the inequality

$$a_l - 1 \leq a_{l+2} + a_{l+3} + \cdots + a_d$$

Now we should assign explicit values to the sequence a_l which is the sequence of comparator lengths.

First we recall the definition of Fibonacci numbers f_k:

$$f_0 = f_1 = 1,$$

$$f_k = f_{k-2} + f_{k-1}.$$

We define the numbers φ_k and ψ_k behaving similarly to f_k:

$$\varphi_0 = \varphi_1 = \psi_0 = \psi_1 = 1,$$

$$\psi_k = \varphi_{k-2} + \psi_{k-1} = 1 + \sum_{i=0}^{k-2} \varphi_i,$$

$$\varphi_k = \text{ the largest odd number smaller or equal } \psi_k.$$

Define $\mathrm{LG}(n)$ to be the smallest k such that $\psi_k \geq n$. The network F_N consists of $d = \mathrm{LG}(N)$ subsequent layers. The length of comparators in layer l is $a_l = \varphi_{d-l}$. We can check that our inequality

$$a_l - 1 \leq a_{l+2} + a_{l+3} + \cdots + a_d$$

now holds, because

$$\varphi_{d-l} - 1 \leq \psi_{d-l} - 1 = \varphi_{d-l-2} + \varphi_{d-l-3} + \cdots + \varphi_0.$$

At the beginning of the computations each displaced 1 is able to reduce its distance to the border to 0, because the largest distance to be covered is $N - 1$ and the latest layer on which a 1 has chance to be moved is 2. So we can check that

$$N - 1 \leq \psi_d - 1 \leq \varphi_{d-2} + \varphi_{d-3} + \cdots + \varphi_0.$$

This short justification proves, that F_N is indeed a simple correction network. We can note that the length φ_{d-1} of the comparators of the first layer was not used in the proof. In practical implementations we can exchange layer 1 of F_N by a layer consisting of comparators of length 1 which can reduce the layout of the network.

The depth d of F_N is $\sim \alpha \log N$ because of the following fact

Fact 2.1 $d = \mathrm{LG}(N) < \alpha \log N + \alpha \log(5)/2 + 1$

Proof. We have $N \leq \psi_d$ and $\psi_{d-1} < N$. It is easy to prove the following inequalities: $f_{k-1} \leq \psi_k \leq f_k$. Thus $f_{d-2} \leq \psi_{d-1} < N$. But f_k is the integer closest to $\Phi^{k+1}/\sqrt{5}$. So $\Phi^{d-1}/\sqrt{5} < N$ and $d < \alpha \log N + \alpha \log(5)/2 + 1$. ∎

So we can formulate the following theorem

Theorem 2.2. *The network F_N is a simple correction network of depth d such that $d < \alpha \log N + \alpha \log(5)/2 + 1$.*

3 Lower bounds

We prove two lower bounds. First we prove, that any odd-even simple correction network has at least as many layers as F_N. Then we prove that any correction network has the depth bigger than

$$\alpha \log N - \frac{\alpha}{2} \log 3$$

Theorem 3.1. *The network F_N has the smallest depth amongst all odd-even simple correction networks.*

The proof is based on the following lemma.

Lemma 3.2. *Assume that an odd-even comparator network is a simple correction network. Assume that b_{d-k} is the maximum length of a comparator that is used in layer $d-k$ to move a 1 for some 0-1 input with a single displaced 1. Then we have $b_{d-k} \leq \varphi_k$.*

Proof. Induction on k. Assume the lemma is true for $k', k' < k$ and prove it for k. Let the maximum length comparator in layer $d-k$ is $[i:j]$. There is some input in which the single displaced 1 is the x_m entry and this 1 is moved by comparator $[i:j]$. In such case $j \leq b$ where b is the border index. Let us take a new input in which the displaced 1 is x_m, but the border is $j-1$. For such an input the displaced 1 is again in r_i, when $[i:j]$ from layer $d-k$ is used. But the 1 is not moved by this comparator. Then this 1 is forced to cover the distance $b_{d-k}-1 = j-i-1$ using the next layers. It is not moved by the layer $d-k+1$. It can be moved by any of further layers $d-k'$ by the distance at most $b_{d-k'} \leq \varphi_{k'}$. So

$$b_{d-k} - 1 \leq b_{d-(k-2)} + b_{d-(k-3)} + \cdots + b_d \leq \varphi_{k-2} + \varphi_{k-3} + \cdots + \varphi_0 \leq \psi_k - 1.$$

Since b_{d-k} is an odd number it has to be not bigger, than φ_k. □

Proof of the Theorem. Assume that we have an input having $N-1$ 0s and a single 1 in r_1. The 1 is not moved by the first layer of an odd-even network because it cannot be on the top of a comparator at this layer. So it has to be moved to r_N by the next layers. Thus

$$N-1 \leq b_2 + b_3 + \ldots b_d \leq \varphi_{d-2} + \varphi_{d-3} + \cdots + \varphi_0 \leq \psi_d - 1.$$

But $d = \mathrm{LG}(N)$ is the minimal depth, such that $N-1 \leq \psi_d - 1$. □

Now we formulate the main result of this paper. Note that the previous lower bound is exact and this bound is not. There is some constant gap between $\alpha \log N + \alpha \log(5)/2 + 1$ which the upper bound given by F_N and the lower bound we find. In fact there are examples of simple correction networks of slightly smaller depth than that of F_N.

Theorem 3.3. *Any simple correction network has depth bigger than*

$$\alpha \log N - \frac{\alpha}{2} \log 3$$

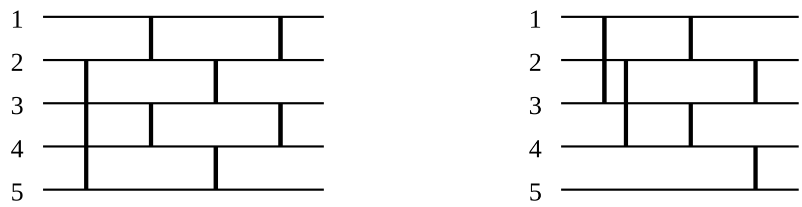

Fig. 2. The network F_5 has depth 4 and the optimal simple correction network for $N = 5$ has depth 3.

Proof of the Theorem. Assume we have an arbitrary (not necessarily odd-even) simple correction network on N registers. Its depth is d. Following the reduction made by Loryś (personal communication) we assign a value $v_{i,l}$ to each pair (i,l), where r_i is a register and l is a layer of the network. This value is the number of indexes j such that if the single displaced 1 is in r_j at the beginning of computations, then for some setting of border b it is in r_i just after the layer l. Let $v_{i,0} = 1$ is this value at the beginning of computations. Note, that $v_{i,d} = i$ for all i.

Fact 3.4 *If in l-th layer the comparator* $[i : j]$ *is present, then*

$$v_{i,l} \le v_{i,l-1} \quad \textit{and} \quad v_{j,l} \le v_{i,l-1} + v_{j,l-1}$$

Proof. Only these 1s that were in r_i or r_j just before layer l can be in r_j just after layer l. □

In fact $v_{i,l} = v_{i,l-1}$, but we do not need this equality and its proof would require a few words of explanation.

Lemma 3.5. *If in l-th layer the comparator* $[i : j]$ *is present, then*

$$v_{i,l}^2 + v_{j,l}^2 \le \Phi^2(v_{i,l-1}^2 + v_{j,l-1}^2)$$

Proof. Consider a linear operator $A(v,w) \stackrel{\text{def}}{=} (w, v+w)$ on R^2. Euclidean norm of a vector (v,w) is defined as follows

$$\|(v,w)\| = \sqrt{v^2 + w^2}.$$

Note, that if $(v,w) = (v_{j,l-1}, v_{i,l-1})$, then

$$\sqrt{v_{i,l}^2 + v_{j,l}^2} \le \sqrt{v_{i,l-1}^2 + (v_{i,l-1} + v_{j,l-1})^2} = \|A(v_{j,l-1}, v_{i,l-1})\|$$

It is easy to check, that A has two eigenvalues: $\Phi, -1/\Phi$ for orthogonal eigenvectors: $(1,\Phi), (-\Phi, 1)$. Due to this fact

$$\|A(v,w)\| \le \Phi \|(v,w)\|$$

Thus

$$v_{i,l}^2 + v_{j,l}^2 \leq \|A(v_{j,l-1}, v_{i,l-1})\|^2 \leq \Phi^2 \|(v_{j,l-1}, v_{i,l-1})\|^2 = \Phi^2 (v_{i,l-1}^2 + v_{j,l-1}^2).$$

□

Summing up inequalities from the last lemma over all comparators of the layer and adding values for not connected registers we get the following corollary

Corollary 3.6.

$$\sum_i v_{i,l}^2 \leq \Phi^2 \sum_i v_{i,l-1}^2$$

To end up the proof of the theorem we note that $\sum_i v_{i,0}^2 = N$ and

$$\sum_i v_{i,d}^2 = \sum_i i^2 = \frac{1}{3}N^3 + \frac{1}{2}N^2 + \frac{1}{6}N > \frac{1}{3}N^3.$$

From the corollary we have

$$\sum_i v_{i,d}^2 \leq \Phi^{2d} \sum_i v_{i,0}^2$$

or

$$\frac{1}{3}N^3 < \Phi^{2d} N$$

so

$$d > \alpha \log N - \frac{\alpha}{2} \log 3$$

□

4 Conclusions

We proved, that a correction network for a single displaced 1 has depth bigger, than $\alpha \log N - \frac{\alpha}{2} \log 3$. An open question remains how many layers we have to add to deal with 2,3,4,... displaced 1s. In [7] we give a construction of 2-correction network of depth

$$\alpha \log N + c\sqrt{\log N}.$$

Does any 2-correction network have to have at least $\alpha \log N + c\sqrt{\log N}$ layers?

Acknowledgments

Author wishes to thank Mirek Kutyłowski, Krzysiek Loryś and Marek Piotrów for presenting the problems, helpful discussions and their encouragement to write this paper. Author wishes also to thank Maciek Warchoł for motivating him to keep on working on this problem.

References

1. M. Ajtai, J. Komlós, E. Szemerédi, Sorting in $c \log n$ parallel steps, *Combinatorica* 3 (1983), 1-19.
2. K.E. Batcher, Sorting networks and their applications, in *AFIPS Conf. Proc.* 32 (1968), 307-314.
3. M. Kik, M. Kutyłowski, M. Piotrów, Correction Networks, in *Proc. of 1999 ICPP*, 40-47.
4. D. E. Knuth, *The Art of Computer Programming*, Vol 3, 2nd edition, Addison Wesley, Reading, MA, 1975.
5. M. Piotrów, Depth Optimal Sorting Networks Resistant to *k* Passive Faults in *Proc. 7th SIAM Symposium on Discrete Algorithms* (1996), 242-251 (also accepted for *SIAM J. Comput.*).
6. M. Schimmler, C. Starke, A Correction Network for *N*-Sorters, *SIAM J. Comput.* 18 (1989), 1179-1197.
7. G. Stachowiak, Fibonacci Correction Networks, in *Algorithm Theory – SWAT 2000* , M Halldórsson (Ed.) , LNCS 1851, Springer 2000, 535-548.
8. A.C. Yao, Bounds on Selection Networks, *SIAM J. Comput.* 9 (1980), 566-582.
9. A.C. Yao, F.F. Yao, On Fault-Tolerant Networks for Sorting, *SIAM J. Comput.* 14 (1985), 120-128.