LECTURE NOTES
ON THE AKS SORTING NETWORK

Vašek Chvátal
Department of Computer Science, Rutgers – The State University of New Jersey
New Brunswick, NJ 08903, USA

1 INTRODUCTION

For the basics of sorting networks, see Chapter 28 of Cormen, Leiserson, and Rivest (1990).

A *perfect halver* is a comparator network with output wires split into blocks B_L, B_R of equal sizes in such a way that, given any input consisting of a distinct keys, the network places the $a/2$ smaller keys in B_L and it places the $a/2$ larger keys in B_R. Perfect halvers may be used as modules to construct a sorting network with N wires such that $N = 2^d$ for some positive integer d. This network is a series composition of networks $N_0, N_1, \ldots, N_{d-1}$ such that each N_t is a parallel composition of 2^t perfect halvers of equal sizes; the 2^{d-t} output wires of each perfect halver in N_t are split into two blocks of equal sizes and each of these blocks carries the input of a perfect halver in N_{t+1}.

Unfortunately, this scheme yields sorting networks of depth $\Omega((\log N)^2)$: every perfect halver with a wires must have depth greater than $\lg(a/2)$. (To see this, consider an output wire y in B_R and let X denote the set of all the input wires from which a key can travel to y. We must have $|X| > a/2$: otherwise placing keys 1, 2, $\ldots |X|$ on the wires in X and placing keys $|X|+1$, $|X|+2$, $\ldots a$ on the remaining $a - |X|$ input wires yields an input that sends one of the $a/2$ smaller keys to the output wire y.) Fortunately, Ajtai, Komlós, and Szemerédi (1983a, 1983b) proved that a variation on this theme yields sorting networks of depth $O(\log N)$: instead of the perfect halvers, we can use weaker modules of constant depth, whose weakness is made up for by a more intricate way of piecing the network together.

Like the output wires of a perfect halver, the output wires of the weaker module are split into blocks B_L, B_R such that $|B_L| = |B_R| = a/2$. Unlike a perfect halver, the weaker module may misdirect a small fraction of the smaller $a/2$ input keys to B_R and it may misdirect a small fraction of the larger $a/2$ input keys to B_L. A partial compensation for this defect is an explicitly designated set F of output wires (typically about 5% of their total) such that (in a sense made precise at the end of section 2.1), for every input of a distinct keys, most of the smallest keys end up in $F \cap B_L$ and most of the largest keys end up in $F \cap B_R$.

Interconnections between these modules may be described in terms of a complete binary tree of depth d. At each time, the 2^d wires of the network are distributed throughout the nodes of the tree; in the beginning, all the wires are held in the root; the objective is to allocate one wire to each leaf in the end in such a way that the sequence of the 2^d keys held in the 2^d leaves is sorted.

In the network built from perfect sorters, this objective is accomplished simply: at each time $t = 0, 1, \ldots, d-1$, the 2^d wires are distributed throughout the t-th level of the tree. At this time, each node x on the t-th level holds 2^{d-t} wires; these wires are used as input of a perfect halver; between times t and $t+1$, the wires from output block B_j are sent down to the j-th child of x. At time d, each leaf of the tree holds a single wire and the sequence of the wires held in the 2^d leaves is sorted. Two clean features of this crude scheme are that

- at all times, all the keys held in a node are addressed below this node,
- as time progresses, wires keep moving steadily to the bottom of the tree.

The more intricate scheme of interconnecting imperfect modules approximates these features in the sense that

- at all times, most of the keys held in a node are addressed below this node,
- as time progresses, wires tend to wander erraticaly to the bottom of the tree.

At each time t, each node that holds any wires at all uses them as input wires of an imperfect module; between times t and $t+1$, it sends all the wires of the output block F to its parent, it sends all the wires of the output block $B_L - F$ to its left child, and it sends all the wires of the output block $B_R - F$ to its right child. Since the modules are not perfect halvers, a

small fraction of the keys may be sent down in a wrong direction at any time; these stray keys will eventually be sent back up again, so that they may correct the wrong turn.

Rather than follow the original Ajtai-Komlós-Szemerédi schedule of moving wires through the tree, we shall describe a simpler scheme proposed later by Paterson (1990) with a few missing details filled in. (A similar, but not quite the same, implementation of Paterson's proposal has been worked out by Pippenger (1990).) In section 2.1, we specify the number of wires held in each node of the tree at each time; in section 2.2, we prove that the resulting network sorts, provided that the quality of its modules is good enough. Construction of the modules will be taken up in section 3.

2 THE NETWORK

As in the preceding section, we shall write $N = 2^d$; it will be convenient to assume that

$$d \text{ is a multiple of four and } d \geq 8.$$

The sorting network that we are going to describe is the series composition of two components, which we will refer to as the *body* of the network and its *tail*. The body nearly sorts: in terms of the complete binary tree with N leaves, it distributes the N input keys through the $N/64$ nodes on level $d-6$ in such a way that

each of the 64 keys held in a node on level $d-6$ is addressed below this node.

The tail is just a parallel composition of $N/64$ copies of a sorting network on 64 wires.

2.1 Construction of the body

The body is the series composition of components indexed by variable t called *time.* At each time t, the N wires are distributed throughout the tree in such a way that the actual number of wires held in a node x depends only on t and on the depth i of x; we let $a(i,t)$ denote this number. To relocate the wires between times t and $t+1$, each node on level i sends $\pi(i,t)$ wires to its parent and it sends $\chi(i,t)$ wires to each of its two children.

	$t=0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
i	$a(i,t)$																
0	4096		256		64		16										
1		2048		512		128		32		8							
2			960		1008		252		64		16						
3				384		480		504		126		32		8			
4							192		240		252		64		16		
5										96		120		126		32	
6													48		60		64
7																24	
i	$\pi(i,t)$																
0	0		0		0		0										
1		128		32		8		0		0							
2			192		48		12		4		0						
3				384		96		24		6		0		0			
4							192		48		12		4		0		
5										96		24		6		0	
6													48		12		0
7																24	
i	$\chi(i,t)$																
0	2048		128		32		8										
1		960		240		60		16		4							
2			384		480		120		30		8						
3				0		192		240		60		16		4			
4							0		96		120		30		8		
5										0		48		60		16	
6													0		24		0
7																0	

The values of $a(i,t)$, $\pi(i,t)$, and $\chi(i,t)$ when $N = 4096$.

In the beginning, all the wires are held in the root:

$$a(0,0) = N. \tag{2.1}$$

Between $t = 0$ and $t = 1$, the root splits the set of N wires into two equal parts and sends them down to its two children:

$$\pi(0,0) = 0,\ \chi(0,0) = N/2,\ \text{ and }\ a(1,1) = N/2. \tag{2.2}$$

Between $t = 1$ and $t = 2$, each of the two nodes on level 1 sends $N/32$ of its $N/2$ wires back to the root and distributes the remaining wires evenly among its children:

$$\pi(1,1) = N/32,\ \chi(1,1) = 15N/64,\ \text{ and }\ a(0,2) = N/16,\ a(2,2) = 15N/64. \tag{2.3}$$

Let $\alpha(t)$ and $\omega(t)$ denote the top and the bottom level, respectively, that contain nonempty nodes at time t: formally, $\alpha(t)$ is the smallest i with $a(i,t) \neq 0$ and $\omega(t)$ is the largest i with $a(i,t) \neq 0$. In particular,

$$\alpha(0) = \omega(0) = 0; \quad \alpha(1) = \omega(1) = 1; \quad \alpha(2) = 0,\ \ \omega(2) = 2.$$

As t increases, we let the top first oscillate between levels 0 and 1,

$$\alpha(t) = \begin{cases} 0 & \text{if } 0 \le t \le d-5 \text{ and } t \text{ is even,} \\ 1 & \text{if } 0 \le t \le d-5 \text{ and } t \text{ is odd,} \end{cases}$$

and then begin a perodic zig-zag descent with period four and the average speed of one level per two iterations,

$$\alpha(t) = \begin{cases} (t-d+5)/2 & \text{if } t \ge d-6 \text{ and } t \equiv 1 \bmod 4, \\ (t-d+6)/2 & \text{if } t \ge d-6 \text{ and } t \equiv 2 \bmod 4, \\ (t-d+7)/2 & \text{if } t \ge d-6 \text{ and } t \equiv 3 \bmod 4, \\ (t-d+8)/2 & \text{if } t \ge d-6 \text{ and } t \equiv 0 \bmod 4. \end{cases}$$

We let the bottom descend steadily in a perodic zig-zag movement with period three and the average speed of one level per three iterations,

$$\omega(t) = \begin{cases} (t+2)/3 & \text{if } t \ge 1 \text{ and } t \equiv 1 \bmod 3, \\ (t+4)/3 & \text{if } t \ge 1 \text{ and } t \equiv 2 \bmod 3, \\ (t+6)/3 & \text{if } t \ge 1 \text{ and } t \equiv 0 \bmod 3. \end{cases}$$

These definitions imply that

- $\alpha(t) < \omega(t)$ whenever $2 \le t \le 3d - 21$, and
- $\alpha(3d - 20) = \omega(3d - 20) = d - 6$;

accordingly, the components whose series composition forms the body of the network are indexed by $t = 0, 1, \ldots, 3d - 21$; in terms of the binary tree, wires stop moving at time $t = 3d - 20$, when we have

$$a(d - 6, 3d - 20) = 64. \tag{2.4}$$

In addition, the definitions of $\alpha(t)$ and $\omega(t)$ imply that

- $\alpha(t) \equiv t \bmod 2$ and $\omega(t) \equiv t \bmod 2$ for all t;

numbers $a(i, t)$ will be defined so that

$$a(i, t) \neq 0 \quad \text{if and only if} \quad \alpha(t) \le i \le \omega(t) \text{ and } i \equiv t \bmod 2. \tag{2.5}$$

Numbers $a(i, t)$, $\pi(i, t)$, and $\chi(i, t)$ defined for all i and t such that

$2 \le t < 3d - 20$, $\alpha(t) \le i \le \omega(t)$, and $i \equiv t \bmod 2$

extend (2.1), (2.2), (2.3), and (2.4) into a feasible schedule for routing the N wires through the tree if and only if they satisfy the following conditions:

Integrality:

All of $a(i, t)$, $\pi(i, t)$, and $\chi(i, t)$ are integers.

Flow balance:

$$\begin{aligned} \pi(i, t) + 2\chi(i, t) &= a(i, t) && \text{whenever } 2 \le t < 3d - 20 \text{ and } a(i, t) \neq 0, \\ \chi(i - 1, t) + 2\pi(i + 1, t) &= a(i, t + 1) && \text{whenever } 2 \le t < 3d - 20 \text{ and } a(i, t + 1) \neq 0; \end{aligned}$$

Boundary conditions:

$a(0, 2) = N/16$, $a(2, 2) = 15N/64$, and $\chi(d - 7, 3d - 21) + 2\pi(d - 5, 3d - 21) = 64$.

The values that we shall use are defined in terms of numbers $c(i,t)$ which, in turn, are defined by

$$c(i,t) = N \cdot 2^{2i-t-2}$$

and may be thought of as the capacities of nodes on level i at time t.

Conditions on i	$a(i,t)$	$\pi(i,t)$	$\chi(i,t)$
$i = \alpha(t)$ and $\alpha(t+1) = i+1$	$c(i,t)$	0	$\frac{1}{2}\,a(i,t)$
$i = \alpha(t)$ and $\alpha(t+1) = i-1$	$c(i,t)$	$\frac{1}{16}\,a(i,t)$	$\frac{15}{32}\,a(i,t)$
$\alpha(t) < i < \omega(t)$ and $i \equiv t \bmod 2$	$\frac{63}{64}\,c(i,t)$	$\frac{1}{21}\,a(i,t)$	$\frac{10}{21}\,a(i,t)$
$i = \omega(t)$ and $t \equiv 1 \bmod 3$	$\frac{63}{64}\,c(i,t)$	$\frac{1}{21}\,a(i,t)$	$\frac{10}{21}\,a(i,t)$
$i = \omega(t)$ and $t \equiv 2 \bmod 3$	$\frac{15}{64}\,c(i,t)$	$\frac{1}{5}\,a(i,t)$	$\frac{2}{5}\,a(i,t)$
$i = \omega(t)$ and $t \equiv 0 \bmod 3$	$\frac{3}{64}\,c(i,t)$	$a(i,t)$	0

The values of $a(i,t)$, $\pi(i,t)$, and $\chi(i,t)$ when $2 \le t \le 3d-21$.

Verifying Flow balance and Boundary conditions for these values is a routine exercise; let us comment on Integrality. Since $2\alpha(t) \ge t - d + 5$ for all t, we have

$$c(\alpha(t),t) \ge 8 \quad \text{for all } t.$$

Two easy corollaries of this easy observation are that

$$c(\alpha(t),t) \ge 64 \quad \text{whenever } \alpha(t+1) = \alpha(t) - 1$$

(since $c(i+1,t) = 8c(i,t+1)$ and that

$$c(i,t) \ge 128 \quad \text{whenever } i > \alpha(t) \text{ and } i \equiv t \bmod 2$$

(since $c(i+2,t) = 16c(i,t)$). It follows not only that all of $a(i,t)$, $\pi(i,t)$, and $\chi(i,t)$ are integers, but also — a fact used in the construction of the network — that

all of $a(i,t)$ and $\pi(i,t)$ are even integers.

Two constants, ε_B and ε_F, control the quality of all the modules used throughout the network. By an (a, ε_B)-*halver,* we mean a comparator network on a wires with the output wires collected in equally sized blocks B_L, B_R such that, for every input of a distinct keys,

(i) the network places at most $\varepsilon_B a$ of its $a/2$ smallest input keys into output block B_R,

(ii) the network places at most $\varepsilon_B a$ of its $a/2$ largest input keys into output block B_L.

By an $(a, f, \varepsilon_B, \varepsilon_F)$-*separator,* we mean an (a, ε_B)-halver with designated blocks F_L, F_R of output wires such that

$$F_L \subseteq B_L, \quad F_R \subseteq B_R, \quad |F_L| = |F_R| = f/2$$

and such that, for every input of a distinct keys and for every $k = 1, 2, \ldots, f/2$,

(iii) the network places at most $\varepsilon_F k$ of its k smallest input keys outside output block F_L,

(iv) the network places at most $\varepsilon_F k$ of its k largest input keys outside output block F_R.

The module used in a node on level i at time t is an $(a(i,t), \pi(i,t), \varepsilon_B, \varepsilon_F)$-separator; the node sends the wires from $F_L \cup F_R$ to its parent, it sends the wires from $B_L - F_L$ to its left child, and it sends the wires from $B_R - F_R$ to its right child.

2.2 Analysis of the body

THEOREM 2.1 *If there is a positive δ such that*

$$\varepsilon_B \leq \frac{1}{128} - \frac{\delta}{2}\left(1 + \frac{1}{1-64\delta^2}\right), \tag{2.6}$$

$$\varepsilon_F \leq 2\delta(1-16\delta), \tag{2.7}$$

then, at time $3d-20$, each node on level $d-6$ holds 64 keys addressed below this node.

PROOF. An *outsider* is a key located in a node x but not addressed below x; an *outsider of order r* is an outsider that would remain an outsider even if it were moved to the ancestor of its current location that is r levels higher up in the tree. (In particular, "outsider" is synonymous with "outsider of order zero".) We aim to prove that, as long as some positive δ satisfies (2.6) and (2.7), nodes on level $d-6$ hold no outsiders at time $3d-20$. For this purpose, consider the predicate

$\mathbf{P}(t)$: For all $i = 0, 1, \ldots, d$ and for all $r = 0, 1, \ldots, d$,
each node on level i holds fewer than $\delta^r c(i,t)/64$ outsiders of order r at time t.

Since $c(d-6, 3d-20) = 64$, we only need prove $\mathbf{P}(3d-20)$; we will use induction on t to prove $\mathbf{P}(t)$ for all $t = 1, 2, \ldots, 3d-20$.

Let $M(i,t)$ denote the $(a(i,t), \pi(i,t), \varepsilon_B, \varepsilon_F)$-separator used in nodes on level i at time t.

As for the induction basis, all the outsiders in the two children of the root at time 1 were sent down from the root between $t=0$ and $t=1$; since $M(0,0)$ is an ε_B-halver, each of the two children receives at most $\varepsilon_B N$ outsiders, and so $\mathbf{P}(1)$ is implied by (2.6).

As for the inductive step, its bulk consists of showing that only a few (if any) of the keys a node u sends to its child v are not addressed below v; such misdirected keys are either outsiders in u or else addressed below the sibling w of v. By the inductive hypothesis, outsiders in u are scarce; since each $M(i,t)$ is an ε_B-halver, most of its input keys that are addressed below w get sent to w; we show first that u does not hold too many keys addressed

below w (Lemma 2.2). The argument relies on a simple formula for the total number of keys held in a node and all its descendants; this formula is used twice, and so we set it on its own as the following lemma.

LEMMA 2.1 *If $a(i,t) \neq 0$ and $i > \alpha(t)$, then*

$$\sum_{j=i}^{d} 2^{j-i} a(j,t) = N \cdot 2^{-i} - c(i,t)/64.$$

PROOF. Trivially,

$$\sum_{j=i}^{d} 2^{j-i} a(j,t) = \sum_{j=0}^{d} 2^{j-i} a(j,t) - \sum_{j=0}^{i-1} 2^{j-i} a(j,t) = N \cdot 2^{-i} - \sum_{j=0}^{i-1} 2^{j-i} a(j,t).$$

Writing $m = (i - \alpha(t))/2$, observe first that m is a positive integer and then that

$$\begin{aligned}
\sum_{j=0}^{i-1} 2^{j-i} a(j,t) &= \sum_{k=1}^{m} 2^{-2k} a(i-2k,t) \\
&= \frac{63}{64} \sum_{k=1}^{m-1} 2^{-2k} c(i-2k,t) + 2^{-2m} c(i-2m,t) \\
&= \frac{63}{64} \sum_{k=1}^{m-1} 64^{-k} c(i,t) + 64^{-m} c(i,t) \\
&= \frac{c(i,t)}{64}.
\end{aligned}$$

LEMMA 2.2 *Let t be an integer which satisfies $1 \leq t < 3d - 20$ and $\mathbf{P}(t)$; let u be a node of the tree, let i be the level of u, and let w be a child of u. Then u holds, at time t, fewer than*

$$\frac{1}{2} a(i,t) + \left(\frac{1}{128} + \frac{\delta}{2(1-64\delta^2)} \right) c(i,t)$$

keys addressed below w.

PROOF.

CASE 1: $i < \omega(t)$.

We may assume that $a(i,t) \neq 0$, since otherwise the conclusion is trivial. In the notation

x_1 = number of keys addressed below w,

x_2 = number of keys held (at time t) below w,

x_3 = number of keys held (at time t) below w and not addressed below w,

$x_1 - (x_2 - x_3)$ counts the number of keys addressed below w and not held below w; this quantity is an upper bound on the number of keys addressed below w and held in u.

Trivially,

$$x_1 = \frac{1}{2} N 2^{-i}.$$

Lemma 2.1, with $i+2$ in place of i, guarantees that

$$x_2 = 2(N \cdot 2^{-(i+2)} - c(i+2,t)/64) = \frac{1}{2} N 2^{-i} - \frac{1}{2}\, c(i,t).$$

$\mathbf{P}(t)$ guarantees that

$$x_3 \;<\; \sum_{j \geq 1} 2^{2j-1} \delta^{2j-1} c(i+2j,t)/64 \;=\; \frac{c(i,t)}{64} \sum_{j \geq 1} (2\delta)^{2j-1} 4^{2j} \;<\; \frac{c(i,t)}{64} \cdot \frac{32\delta}{1-64\delta^2}\,.$$

Altogether,

$$x_1 - (x_2 - x_3) \;<\; \left(\frac{1}{2} + \frac{\delta}{2(1-64\delta^2)} \right) c(i,t);$$

the assumption of this case guarantees that $a(i,t) \geq 63c(i,t)/64$, and so

$$\left(\frac{1}{2} + \frac{\delta}{2(1-64\delta^2)} \right) c(i,t) \;\leq\; \frac{1}{2}\, a(i,t) + \left(\frac{1}{128} + \frac{\delta}{2(1-64\delta^2)} \right) c(i,t).$$

Case 2: $i = \omega(t)$.

Lemma 2.1 guarantees that

$$\frac{1}{2} N 2^{-i} = \frac{a(i,t)}{2} + \frac{c(i,t)}{128};$$

again, the left-hand side of this equation is the number of keys addressed below w. □

LEMMA 2.3 *Let t be an integer which satisfies $1 \leq t < 3d-20$ and $\mathbf{P}(t)$; let v be a node of the tree and let $i+1$ be the level of v. Then v holds at time $t+1$ fewer than $c(i+1,t+1)/64$ outsiders.*

PROOF: Let X denote the set of all outsiders that are held in v at time $t+1$ and let us write

$x \in X_P$ if $x \in X$ and x is held in the parent of x at time t,

$x \in X_L$ if $x \in X$ and x is held in the left child of x at time t,

$x \in X_R$ if $x \in X$ and x is held in the right child of x at time t.

Furthermore, let u denote the parent of v and let w denote the sibling of v. Lemma 2.2 guarantees that u holds, at time t, fewer than

$$\frac{1}{2}\,a(i,t) + \left(\frac{1}{128} + \frac{\delta}{2(1-64\delta^2)}\right) c(i,t)$$

keys addressed below w; $\mathbf{P}(t)$ guarantees that u holds, at time t, fewer than $c(i,t)/64$ outsiders; all the remaining keys held in u at time t are addressed below v. It follows that a perfect halver in place $M(i,t)$ would send to v fewer than

$$\left(\frac{3}{128} + \frac{\delta}{2(1-64\delta^2)}\right) c(i,t)$$

keys that are not addressed below v; since $M(i,t)$ is only an ε_B-halver, it may misdirect up to $\varepsilon_B \cdot a(i,t)$ additional keys to v; it follows that

$$|X_P| < \left(\frac{3}{128} + \frac{\delta}{2(1-64\delta^2)} + \varepsilon_B\right) c(i,t).$$

$\mathbf{P}(t)$ guarantees that

$$|X_L| < \frac{\delta c(i+2,t)}{64} \quad \text{and} \quad |X_R| < \frac{\delta c(i+2,t)}{64}\ :$$

at time t, each key in $X_L \cup X_R$ is an outsider of order one in a child of v. Altogether, we have

$$|X| = |X_P| + |X_L| + |X_R| < \left(\left(\frac{3}{128} + \frac{\delta}{2(1-64\delta^2)} + \varepsilon_B\right) \cdot \frac{1}{2} \;+\; 2 \cdot \frac{\delta}{64} \cdot 8\right) c(i+1,t+1),$$

and so $|X| < c(i+1,t+1)/64$ by (2.6). □

LEMMA 2.4 *Let t be an integer which satisfies $1 \le t < 3d-20$ and $\mathbf{P}(t)$; let v be a node of the tree, let $i+1$ be the level of v, and let r be a positive integer. Then v holds at time $t+1$ fewer than $\delta^r c(i+1,t+1)/64$ outsiders of order r.*

PROOF. We may assume that $a(i+1,t+1) \neq 0$, since otherwise the conclusion is trivial. With u standing for the parent of v, let us first show that

(i) $\pi(i,t) \geq 3c(i,t)/64$, or else u holds no outsiders at time t.

For this purpose, let us assume that $\pi(i,t) < 3c(i,t)/64$, and so $i = \alpha(t)$ and $\alpha(t+1) = i+1$ by the definition of $\pi(i,t)$. Now if $i = 0$, then u holds no outsiders since it is the root; if $i \geq 1$, then $\alpha(t) \leq (t-d+7)/2$, and so $c(i,t) \leq 32$, in which case the absence of outsiders from u at time t is guaranteed by $\mathbf{P}(t)$.

Now let X denote the set of all outsiders of order r that are held in v at time $t+1$ and let us write

$x \in X_P$ if $x \in X$ and x is held in u at time t,

$x \in X_L$ if $x \in X$ and x is held in the left child of x at time t,

$x \in X_R$ if $x \in X$ and x is held in the right child of x at time t.

We claim that

(ii) $|X_P| < \varepsilon_F \delta^{r-1} c(i,t)/64$.

Each key in X_P is an outsider of order $r-1$ at time t, when it is held in u. Assumption $\mathbf{P}(t)$ guarantees that u holds fewer than $\delta^{r-1}c(i,t)/64$ outsiders of order $r-1$ at time t; in turn, claim (i) guarantees that the actual number of these outsiders is at most $\pi(i,t)/3$, and so module $M(i,t)$ places fewer than $\varepsilon_F \delta^{r-1} c(i,t)/64$ of them on its output wires outside its F.

Each key in $X_L \cup X_R$ is an outsider of order $r+1$ at time t, when it is held in a child of v; assumption $\mathbf{P}(t)$ guarantees that

(iii) $|X_L| < \delta^{r+1} c(i+2,t)/64$ and $|X_R| < \delta^{r+1} c(i+2,t)/64$.

Altogether, we have

$$|X| = |X_P| + |X_L| + |X_R| < \left(\left(\frac{\varepsilon_F \delta^{r-1}}{64} \right) \cdot \frac{1}{2} \; + \; 2 \cdot \frac{\delta^{r+1}}{64} \cdot 8 \right) c(i+1,t+1),$$

and so $|X| < \delta^r c(i+1,t+1)/64$ by (2.7). □

3 THE MODULES

3.1 Existence of expanders

The set of vertices of a *bipartite graph* is partitioned into two disjoint *parts* in such a way that each edge has one endpoint in each of the two parts; a *matching* is a set of pairwise disjoint edges; for each subset S of the vertex-set of a graph G, we write

$$N_G(S) = \{u : \ u \text{ is adjacent to at least one vertex in } S \ \}.$$

By a *bipartite* (n, d, μ)*-expander,* we shall mean a bipartite graph G such that

(i) G has n vertices in each part,

(ii) the edge-set that is the union of d matchings,

(iii) every nonempty set S of vertices in one part of G has $|N_G(S)| > \min\{\mu|S|\,, n - |S|\}$.

THEOREM 3.1 *If μ and d are positive integers such that*

$$(\mu + 1)\, e^{\,\mu+2} \left(\frac{\mu}{\mu+1}\right)^{d} < \frac{1}{3}\,, \tag{3.1}$$

then, for every positive integer n, there is a bipartite (n, d, μ)-expander.

PROOF. Consider arbitrary but fixed positive integers n, d, and μ and take two disjoint sets V_L, V_R of vertices such that $|V_L| = |V_R| = n$. By a *perfect matching*, we shall mean a set of n pairwise disjoint edges, each of which has one endpoint in V_L and the other endpoint in V_R. Let $\mathcal{G}$ denote the set of all unions of d perfect matchings and note that, as there are $n!$ distinct perfect matchings,

$$|\mathcal{G}| = (n!)^d.$$

For every positive integer s, write $t = \min\{\mu s, n-s\}$ and let $\mathcal{B}(s)$ denote the set of all graphs in $\mathcal{G}$ such that some nonempty set S of vertices in one part of G has $|S| = s$, $|N(S)| \le t$. Since $\mathcal{B}(s) = \emptyset$ whenever $s \ge n$ (in fact, $\mathcal{B}(s) = \emptyset$ whenever $s > n/2$),

$$\mathcal{G} - \bigcup_{s=1}^{n-1} \mathcal{B}(s)$$

is the set of all (n, d, μ)-expanders; this set is nonempty whenever

$$\frac{\sum_{s=1}^{n-1} |\mathcal{B}(s)|}{|\mathcal{G}|} < 1; \tag{3.2}$$

we are going to prove that (3.1) implies (3.2). More precisely, we are going to prove that (3.1) implies

$$\frac{|\mathcal{B}(s)|}{|\mathcal{G}|} < \frac{2}{3^s}. \tag{3.3}$$

For this purpose, note that every edge-set E of a graph in $\mathcal{B}(s)$ can be manufactured as follows:

(i) set $E = \emptyset$;

(ii) choose between $i = 1$ and $i = 2$;

(iii) choose a subset S of V_i such that $|S| = s$;

(iv) choose a subset T of the other V_j such that $|T| = t$
(we will have $N_G(S) \subseteq T$);

(v) repeat d times
(add to E a perfect matching M such that $N_M(S) \subseteq T$):

(vi) choose a subset S' of T such that $|S'| = |S|$;

(vii) choose s pairwise disjoint edges
with one endpoint in S and the other endpoint in S';

(viii) choose $n - s$ pairwise disjoint edges
with one endpoint in $V_i - S$ and the other endpoint in $V_L - S'$;

(ix) add the perfect matching chosen by (vi), (vii), and (viii) to E.

There are two choices in (ii); there are $\binom{n}{s}$ choices in (iii); there are $\binom{n}{t}$ choices in (iv); there are $\binom{t}{s}$ choices in (vi); there are $s!$ choices in (vii); there are $(n-s)!$ choices in (viii). Altogether, there are

$$2 \binom{n}{s} \binom{n}{t} \left(\binom{t}{s} s! \, (n-s)! \right)^d$$

different settings for the production line from start to finish; this quantity is an upper bound on $|\mathcal{B}(s)|$ since every edge-set of a graph in $\mathcal{B}(s)$ arises from at least one of the settings. It follows that

$$\frac{|\mathcal{B}(s)|}{|\mathcal{G}|} \;\le\; 2\binom{n}{s}\binom{n}{t}\left(\frac{\binom{t}{s}}{\binom{n}{s}}\right)^d;$$

using the inequalities $\binom{n}{s} \le n^s/s!$ (which follows directly from the definition) and $s! \ge (s/e)^s$ (which follows by induction on s from the inequality $1+x \le e^x$), and $\binom{t}{s}/\binom{n}{s} \le (t/n)^s$ (which is an elementary consequence of $t \le n$), we conclude that

$$\frac{|\mathcal{B}(s)|}{|\mathcal{G}|} \;\le\; 2\left(\frac{en}{s}\right)^s\left(\frac{en}{t}\right)^t\left(\frac{t}{n}\right)^{sd}. \tag{3.4}$$

In addition, note that $((\mu+1)/\mu)^d < e^{d/\mu}$, and so the left-hand side of (3.1) is greater than $(\mu+1)\,e^{\,\mu+2-d/\mu}$; in particular, (3.1) implies that

$$d > \mu(\mu+2). \tag{3.5}$$

We shall derive (3.3) from (3.1), (3.4), and (3.5).

Case 1: $t = \mu s$. In this case,

$$\left(\frac{en}{s}\right)^s\left(\frac{en}{t}\right)^t\left(\frac{t}{n}\right)^{sd} = \left(\frac{en}{s}\left(\frac{en}{\mu s}\right)^{\mu}\left(\frac{\mu s}{n}\right)^{d}\right)^s = \left(\mu\, e^{\,\mu+1}\left(\frac{\mu s}{n}\right)^{d-(\mu+1)}\right)^s$$

and $s \le n/(\mu+1)$; since $d-(\mu+1) \ge 0$ by (3.5), it follows that

$$\mu\, e^{\,\mu+1}\left(\frac{\mu s}{n}\right)^{d-(\mu+1)} \le \mu\, e^{\,\mu+1}\left(\frac{\mu}{\mu+1}\right)^{d-(\mu+1)} < \mu\, e^{\,\mu+2}\left(\frac{\mu}{\mu+1}\right)^{d} < \frac{1}{3}.$$

Case 2: $t = n - s$. In this case,

$$\frac{n}{s} \le \mu+1,\;\; t \le \mu s,\;\; \left(\frac{n}{t}\right)^t = \left(1+\frac{s}{n-s}\right)^{n-s} \le e^s,\;\; \frac{t}{n} \le \frac{\mu}{\mu+1},$$

and so

$$\left(\frac{en}{s}\right)^s\left(\frac{en}{t}\right)^t\left(\frac{t}{n}\right)^{sd} \le \left((\mu+1)\,e^{\,\mu+2}\left(\frac{\mu}{\mu+1}\right)^{d}\right)^s < \frac{1}{3^s}.$$

□

3.2 From expanders to strong $(2n, \varepsilon)$-halvers

By an *strong* $(2n, \varepsilon)$*-halver,* we shall mean a comparator network on $2n$ wires with the output wires collected in equally sized blocks B_L, B_R so that, for every $k = 1, 2, \ldots, n$,

(i) the network places at most εk of its k smallest input keys into output block B_R and

(ii) the network places at most εk of its k largest input keys into output block B_L.

THEOREM 3.2 *For every positive ε there is a positive integer d such that, for every positive integer n, there is a strong $(2n, \varepsilon)$-halver of depth d.*

PROOF. Given a positive ε, choose a positive integer μ such that $1/(\mu + 1) < \varepsilon$. Theorem 3.1 guarantees existence of a positive integer d such that, for every positive integer n, there is a bipartite (n, d, μ)-expander; for every choice of positive integer n, we are going to construct a strong $(2n, \varepsilon)$-halver of depth d.

Given a bipartite (n, d, μ)-expander G, we shall construct a strong $(2n, 1/(\mu + 1))$-halver H of depth d, whose $2n$ wires are identified with the $2n$ vertices of G. For this purpose, let V_L, V_R denote the two parts of the vertex-set of G and let $M_1, M_2, \ldots M_d$ denote the matchings whose union is the edge-set of G. The t-th layer in the series decomposition of H into d layers is defined by M_t: its comparators are precisely the edges of M_t and

($\star$) whenever two keys x and y come into the two wires of a comparator,
$\min\{x, y\}$ comes out on the wire in V_L and
$\max\{x, y\}$ comes out on the wire in V_R.

Wires in V_L form the output block B_L and wires in V_R form the output block B_R.

To see that H has defining property (i) of a strong $(2n, 1/(\mu + 1))$-halver, consider an arbitrary $k = 1, 2, \ldots, n$, let Q denote the set of k smallest input keys, and let S denote the set of all the wires in B_R that hold keys from Q as their output values: our aim is to prove that $|S| \leq k/(\mu + 1)$. For every u in $N_G(S)$, there are — by definition — a wire v in S and a subscript t such that uv is an edge in M_t; let u_t, v_t denote the keys held in wires u, v after

the first t layers of comparators and let u_d, v_d denote the keys held in wires u, v after all d layers of comparators. By definition of S, we have $v_d \in Q$; property $(\star)$ guarantees that $u_d \leq u_t \leq v_t \leq v_d$; it follows that $u_d \in Q$ and so, as u is an arbitrary element of $N_G(S)$,

$$|S| + |N_G(S)| \leq k.$$

Since G is a bipartite (n, d, μ)-expander G, we have

$$\min\{(1+\mu)|S|\,, n\} < |S| + |N_G(S)|\,.$$

comparing the two bounds and keeping in mind that $k \leq n$, we conclude that $(1+\mu)|S| < k$.

A mirror image of this argument shows that H has defining property (ii) of a strong $(2n, 1/(\mu+1))$-halver. □

3.3 From strong $(2n, \varepsilon)$-halvers to separators

THEOREM 3.3 *For every choice of positive ε_B, ε_F, and δ, there is a positive integer d such that, for every choice of positive even integers a and f such that $\delta a \leq f \leq a$, there is an $(a, f, \varepsilon_B, \varepsilon_F)$-separator of depth d.*

PROOF. We may assume that $\delta \leq 1$ since otherwise there is nothing to prove. Now let r be the smallest nonnegative integer such that $2^r\delta \geq 1$ and set $\varepsilon = \min\{\varepsilon_B, \varepsilon_F/(r+1)\}$. Theorem 3.2 guarantees existence of a positive integer d_0 such that, for every positive integer n, there is a strong $(2n, \varepsilon)$-halver of depth d_0; for every choice of positive even integers a and f such that $\delta a \leq f \leq a$, we are going to construct an $(a, f, \varepsilon_B, \varepsilon_F)$-separator of depth at most $(r+1)d_0$.

With t the smallest nonnegative integer such that $2^t f \geq a$, the separator is a series composition of $t+1$ layers. Layer 0 is a strong (a, ε)-halver of depth d_0; its output blocks B_L, B_R define output blocks B_L, B_R of the separator. Layer 1 is a parallel composition of two strong $(a - 2^{t-1}f, \varepsilon)$-halvers of depth d_0 ($a - 2^{t-1}f$ is positive and at most $a/2$); the left halver draws its input wires from B_L and collects its output wires in equally sized blocks

B^1_{LL}, B^1_{LR}; the right halver draws its input wires from B_R and collects its output wires in equally sized blocks B^1_{RL}, B^1_{RR}. Writing

$$B^1_L = B_L - B^1_{LR} \quad \text{and} \quad B^1_R = B_R - B^1_{RL},$$

note that $|B^1_L| = |B^1_R| = 2^{t-2}f$.

The subsequent layers are constructed recursively. Layer i is a parallel composition of two strong (a, ε)-halvers of depth d_0; a set B^{i-1}_L of $2^{t-i}f$ wires in B_L forms the input wires of the left halver and a set B^{i-1}_R of $2^{t-i}f$ wires in B_R forms the input wires of the right halver; the left halver and collects its output wires in equally sized blocks B^i_{LL}, B^i_{LR} and the right halver collects its output wires in equally sized blocks B^i_{RL}, B^i_{RR}; we set $B^i_L = B^{i-1}_L - B^i_{LR}$ and $B^i_R = B^{i-1}_R - B^i_{RL}$.

If $1 \leq k \leq f/2$, then this network places at most εk of the k smallest input keys in each of the sets $B_R, B^1_{LR}, B^2_{LR}, \ldots, B^t_{LR}$, and so it places at most $\varepsilon_F k$ of these k keys outside B^t_{LL}; similarly, the network places at most $\varepsilon_F k$ of the k largest input keys outside B^t_{RR}; we set $F_L = B^t_{LL}$ and $F_R = B^t_{RR}$. □

REFERENCES

- M. Ajtai, J. Komlós, and E. Szemerédi (1983a), An $O(n \log n)$ sorting network, *Proc. 15th Ann. ACM Symp. on Theory of Computing*, pp. 1–9.

- M. Ajtai, J. Komlós, and E. Szemerédi (1983b), Sorting in $c \log n$ parallel steps, *Combinatorica* **3**, 1–19.

- B.E. Batcher (1968), Sorting networks and their applications, *Proc. 32nd Ann. AFIPS Spring Joint Comp. Conf.*, pp.307–314.

- T.H. Cormen, C.E. Leiserson, and R.L. Rivest (1990), *Introduction to Algorithms,* MIT Press/McGraw-Hill.

- M.S. Paterson (1990), Improved sorting networks with $O(\log n)$ depth, *Algorithmica* **5**, 75–92.

- N. Pippenger (1990), Communication networks, in: *Handbook Of Theoretical Computer Science, Vol. A, Algorithms and Complexity* (J. van Leeuwen, ed.) The MIT Press/ Elsevier, Chapter 15, pp. 805–833.