Expanders that Beat the Eigenvalue Bound: Explicit Construction and Applications*

Avi Wigderson † David Zuckerman ‡

Abstract

For every n and $0 < \delta < 1$, we construct graphs on n nodes such that every two sets of size n^δ share an edge, having essentially optimal maximum degree $n^{1-\delta+o(1)}$. Using known and new reductions from these graphs, we derive new explicit constructions of:

1. A k round sorting algorithm using $n^{1+1/k+o(1)}$ comparisons.
2. A k round selection algorithm using $n^{1+1/(2^k-1)+o(1)}$ comparisons.
3. A depth 2 superconcentrator of size $n^{1+o(1)}$.
4. A depth k wide-sense nonblocking generalized connector of size $n^{1+1/k+o(1)}$.

All of these results improve on previous constructions by factors of $n^{\Omega(1)}$, and are optimal to within factors of $n^{o(1)}$. These results are based on an improvement to the extractor construction of Nisan & Zuckerman: our algorithm extracts an asymptotically optimal number of random bits from a defective random source using a small additional number of truly random bits.

1 Introduction

1.1 Expanders

A graph is called an expander if any subset of its nodes of a certain size (or sizes) has many neighbors. Varying the meaning of "certain size" and "many neighbors" give different notions of expansion, as we will see below. Expander graphs have had numerous applications in a wide range of areas of computer science (e.g. [AKS1, AKS2, FFP, GIL+, Tom, Val2]).

It is not hard to show that a random graph is an expander. Yet the problem of deterministically constructing expanders has proved to be difficult; the construction of constant-degree expanders was considered a breakthrough [Mar, GG].

The eigenvalue method has proved particularly useful in designing expander graphs. This method works by looking at the adjacency matrix A of an undirected graph $G = (V, E)$. To simplify matters for the

*A preliminary version of this paper appeared in the *25th ACM Symposium on Theory of Computing*, 1993, pp. 245-251.

†Institute of Computer Science, Hebrew University of Jerusalem, Israel, 91904. avi@cs.huji.ac.il. This research was supported by USA-Israel BSF grant 92-00106 and by a Wolfeson research award administered by the Israeli Academy of Sciences.

‡Dept. of Computer Sciences, The University of Texas at Austin, Austin, TX 78712. diz@cs.utexas.edu. Part of this research was supported by NSF NYI Grant No. CCR-9457799, a David and Lucile Packard Fellowship for Science and Engineering, and an Alfred P. Sloan Research Fellowship. Most of this research was done while the author was affiliated with MIT and supported by an NSF Postdoctoral Fellowship, NSF Grant No. 92-12184 CCR, and DARPA Grant No. N00014-92-J-1799. Part of this research was done while the author visited Princeton University, partially supported by DIMACS.

moment, suppose that G is d-regular, so A has d as its largest eigenvalue. Then G is an expander if and only if the other eigenvalues of A are bounded away from d [Alo2, Tan, AM].

The problem with this equivalence is that it is not tight. For a random d-regular graph, small sets S have roughly $(d-1)|S|$ neighbors, yet bounding the second eigenvalue can only be used to show the existence of roughly $(d/2)|S|$ neighbors [Kah].

The situation gets much worse for larger degree and stronger expansion. A definition that captures such strong expansion is:

Definition 1.1 *[Pip3] An undirected graph is* a-expanding *if any two disjoint sets of vertices, each containing at least a vertices, are joined by an edge. Equivalently, every set with a vertices has more than $n-a$ neighbors.*

It is easy to see that every a-expanding graph must have $d \geq \frac{n}{a}$, and it is not too hard to show that random $\frac{n}{a}\log n$-regular graphs are a-expanding. To see the best upper bound on d that can be obtained using the eigenvalue method, let λ be the second largest eigenvalue of A, and $E(S,T)$ denote the number of edges between $S, T \subseteq V$. The basic inequality (simplified slightly from the best inequality obtainable) is

$$E(S,T) \geq \frac{d|S||T|}{n} - \lambda\sqrt{|S||T|} \tag{1}$$

Thus if $|S| = |T| = a$, to ensure $E(S,T) > 0$ we must have $da \geq \lambda n$.[1] But it is known that if $d \leq n/2$ then $\lambda \geq \sqrt{d/2}$, which forces $d \geq \frac{1}{2}(\frac{n}{a})^2$. This is useless when $a < \sqrt{n}$, and even when $a \geq \sqrt{n}$, it forces roughly a quadratic loss compared to the probabilistic existence bound $d = \frac{n}{a}\log n$.

In this paper, we show how to construct graphs that come within an $n^{o(1)}$ factor of optimal:

Theorem 1.2 *There is a Logspace algorithm that, on input n (in unary) and δ, where $0 < \delta = \delta(n) < 1$, constructs n^{δ}-expanding graphs on n nodes with maximum degree $n^{1-\delta+o(1)}$.*

Remark: In fact, our $n^{o(1)}$ factors will be bounded by $\exp((\log n)^{2/3+o(1)})$.

Our result is obtained by improving the extractor construction of [NZ]. The motivation for extractors is that there are many fast and useful randomized algorithms. The extractor allows us to compute efficiently if the random source is defective, as long as we have a small number of truly random bits available. (In fact, even if we don't have any truly random bits, we can cycle through all possibilities – see [NZ, Zuc2] for more details.) Our model for defective random source will essentially be the most general:

Definition 1.3 *[Zuc1] A distribution D on $\{0,1\}^n$ is called a δ-source if for all $x \in \{0,1\}^n$, $D(x) \leq 2^{-\delta n}$.*

Note that a particular type of δ-source is the uniform distribution on a subset $A \subseteq \{0,1\}^n$, $|A| \geq 2^{\delta n}$. We can now define:

Definition 1.4 *[NZ] $E : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$ is called an $(n, m, t, \delta, \epsilon)$-extractor if for every δ-source D, the distribution of $E(x,y) \circ y$ induced by choosing x from D and y uniformly in $\{0,1\}^t$ is within statistical distance ϵ of the uniform distribution. Here $\circ$ denotes concatenation.*

For now, think of $\delta \leq 1/2$ as a fixed constant and $\epsilon = 1/n^c$ for some constant c. In [NZ] an efficient extractor was described requiring $t = (\log n)^{O(1)}$ additional random bits and outputting $m = \Omega(n)$ nearly-random bits. Here we show how to improve the output length to asymptotically the right value: our construction gives $t = (\log n)^{O(1)}$ and $m = (\delta - o(1))n$. This near-optimal output length is necessary for our graphs to have near-optimal expansion. We do pay a price, however, in that our t is larger than in [NZ].

The improvememnt allowing near optimal output length comes from a simple but crucial observation, namely that we can apply a given extractor to a defective source until we get essentially all of its entropy out. It is made precise in lemmas 2.4 and 2.5. This idea was key for later results as well, e.g. the near-optimal samplers of Zuckerman [Zuc3].

[1] Actually, using Tanner's inequality [Tan], it suffices to have the degree slightly less: $da \geq \lambda n/(1+\lambda/d)$.

The only tools we use are hash functions and k-wise independence. Our construction builds heavily on the one in [NZ], which in turn builds upon ideas in [Zuc1, Zuc2]. Indeed, the explicit construction of expanders that beat the eigenvalue bound in a different scenario were first obtained in [Zuc1].

1.2 Applications

Our graphs improve many explicit constructions. In all cases, our results improve upon previous constructions by factors of $n^{\Omega(1)}$, and are optimal to within factors of $n^{o(1)}$. Therefore, it will be convenient to ignore $n^{o(1)}$ factors using the following notation.

Definition 1.5 *$O^*(f(n))$ denotes $f(n)n^{o(1)}$.*

Still, we stress that in our bounds the $n^{o(1)}$ factor is really $\exp((\log n)^{2/3+o(1)})$. In the probabilistic and optimal bounds they are at most $\log^2 n$, so there is still a gap to close.

Sorting and Selecting in Rounds

Sorting and selecting in rounds has been an area of intensive study. This is the worst-case complexity in Valiant's comparison-tree model [Val1] using a constant number k of rounds. For sorting, $\Omega(n^{1+1/k}(\log n)^{1/k})$ comparisons are necessary [AA], and $O(n^{1+1/k}\log n)$ comparisons are sufficient [BT]. This last result, however, is non-constructive. Pippenger [Pip3] showed a slightly worse non-explicit construction of $O(n^{1+1/k}(\log n)^{2-2/k})$, but his construction depends only on the existence of n^{δ}-expanding graphs with an optimal number of edges. Thus, applying our construction, we obtain a near-optimal explicit algorithm using $O^*(n^{1+1/k})$ comparisons. The best known previously was $O^*(n^{1+2/(k+1)})$ [Pip3].

The situation for selecting is very similar. The non-constructive upper bound of $O(n^{1+1/(2^k-1)}(\log n)^{2-\frac{2}{2^k-1}})$ comparisons [Pip3] is close to the lower bound of $\Omega(n^{1+1/(2^k-1)}(\log n)^{\frac{2}{2^k-1}})$ [AA]. Again we obtain a nearly optimal constructive upper bound of $O^*(n^{1+1/(2^k-1)})$, improving the previous best of $O^*(n^{1+2^{k-2}/(3^{k-1}-2^{k-2})})$ [Pip3].

A related problem is "almost-sorting" in 1 round: how many comparisons are necessary to find the relations of all but r of the pairs of elements. Several papers have analyzed the case $r = o(n^2)$ (e.g. [AKSS, AA]), but it is natural to study the question for general r, such as $r = n^{2-\epsilon}$, $0 < \epsilon < 1$. For such r, the non-constructive upper bound of $O(n^{1+\epsilon}\log^2 n)$ comparisons [AKSS, AA] is close to the lower bound of $\Omega(n^{1+\epsilon}\log n)$ [AA]. Here we give the first nearly optimal constructive upper bound of $O^*(n^{1+\epsilon})$, improving the previous best of $O(n^{1+2\epsilon}\log n)$ [AKSS, AA].

Superconcentrators and Nonblocking Networks

Our graphs are also useful in explicitly constructing various networks. An (n, m)-network is a directed acyclic graph with n distinguished vertices called inputs and m other distinguished nodes called outputs. An (n, n)-network is also called an n-network. The size of a network is the number of edges, and the depth is the length of the longest path from an input to an output.

One important example is a superconcentrator. An n-superconcentrator is an n-network such that for every subset A of the inputs and B of the outputs such that $|A| = |B|$, there exist vertex-disjoint paths joining the vertices in A to the vertices in B. Superconcentrators have proved very useful in complexity theory (e.g [Tom, Val2]). Indeed, superconcentrators were the original motivation for constructing expander graphs.

While linear-sized superconcentrators have been explicitly constructed (e.g. [GG]), these all have logarithmic depth. The best known explicit constructions for depth 2 is $O(n^{3/2})$ [Mes], and for depth $2k+1$ are of size $O(n^{(k+3)/(k+2)})$ [Alo1]. On the other hand, non-explicit constructions were known of size $O(n\log^2 n)$ for depth 2 [Pip2], and $O(n\lambda(k, n))$ for depth $2k$, $k \geq 2$, for an extremely slowly growing $\lambda(k, n)$ (e.g. $\lambda(2, n) = \log^* n$) [DDPW].

Here, we give an explicit construction for depth 2 of size $O^*(n)$. This is our biggest improvement: a factor of $O^*(\sqrt{n})$. We use this construction to give the first explicit construction of a linear-sized superconcentrator with sublogarithmic depth (namely, depth $(\log n)^{2/3+o(1)}$).

The main tool in most superconcentrator constructions is the concentrator, which is interesting in its own right (e.g. [Mor]). An (n, m, l)-concentrator is an (n, m)-network such that every set of at most l inputs is connected by vertex-disjoint paths to outputs. Concentrators of depth 1 are usually built with expanders, with the exception of [Mor]. The best previous construction of depth 1 $(n, n^\delta, \Omega(n^\delta))$ concentrators has size $O(n^{1+\min\{\delta/2,(1-\delta)\}})$ (see e.g. [FFP]). Here we construct a generalization of these concentrators with size $O^*(n)$.

We use this generalized construction to give a construction of wide-sense nonblocking generalized connectors. To motivate this, think of routing telephone calls from inputs to outputs: any input-output pair can be requested at any time and the callers may "hang up" at any time, at which time these new inputs and outputs are free to be requested. A wide-sense nonblocking generalized connector, roughly speaking, is one where the router need never be stuck (we define it precisely later). Feldman, et.al. [FFP] gave non-explicit constructions for depth k wide-sense nonblocking generalized connectors of size $O(n^{1+1/k}(\log n)^{1-1/k})$, essentially matching the $\Omega(n^{1+1/k})$ lower bound [PY]. They also gave explicit constructions for depth 2 of size $O(n^{5/3})$, for depth 3 of size $O(n^{11/7})$, and for depth k of size $O(n^{1+2/k})$. Here we give an explicit construction for depth k of size $O^*(n^{1+1/k})$.

2 The Construction

For ease of reading, we ignore integrality constraints, assuming when needed that a number is an integer. It is not hard to see that this does not affect the validity of our arguments. We start by clarifying what we mean by "within statistical distance ϵ of the uniform distribution" in the definition of extractor.

Definition 2.1 *A probability distribution D on a set S is* uniform within ϵ *if for all $X \subseteq S$, $|D(X) - |X|/|S|| \leq \epsilon$. Here $D(X)$ denotes the probability of the set X according to distribution D.*

From the definition of extractor, it is clear that the smaller δ and ϵ are, the harder it is to construct an extractor. In [NZ] (see the final version for slightly improved parameter dependence), an extractor is constructed for essentially all reasonable δ and ϵ (although upper bounds are placed to make the expressions simpler):

Lemma 2.2 *[NZ] For any parameters $\delta = \delta(n)$ and $\epsilon = \epsilon(n)$ with $1/n \leq \delta \leq 1/2$ and $2^{-\delta n} \leq \epsilon \leq 1/n$, there exists a polynomial-time, linear-space computable $(n, m, t, \delta, \epsilon)$-extractor $E : \{0,1\}^n \times \{0,1\}^t \to \{0,1\}^m$, where $t = O(\log \epsilon^{-1} \log^2 n \log \delta^{-1}/\delta)$ and $m = \Omega(\delta^2 n / \log \delta^{-1})$.*

The idea for constructing our graphs is fairly simple, and described by the following lemma:

Lemma 2.3 *If there is an $(n, m, t, \delta, 1/4)$-extractor computable in linear space, then there is an N^δ-expanding graph on $N = 2^n$ nodes with maximum degree $N2^{1+2t-m}$ constructible in Logspace.*

Proof: An extractor E naturally defines a bipartite graph H on $V \times W$, where $V = \{0,1\}^n$ and $W = \{0,1\}^m$. Namely, connect $x \in \{0,1\}^n$ to $z \in \{0,1\}^m$ if and only if there is a $y \in \{0,1\}^t$ such that $E(x, y) = z$. Let $N = |V| = 2^n$ and $M = |W| = 2^m$. Then E being an extractor implies that a subset A of V of size at least N^δ has at least $(1-\epsilon)M = 3M/4$ neighbors. Otherwise, if more than ϵM vertices in W have probability 0 of being hit, then $E(x, y)$ cannot be uniform within ϵ, for x chosen uniformly from A and y from $\{0,1\}^t$. Thus, any two sets of size N^δ have a common neighbor (in fact, at least $M/2$ common neighbors). In other words, the graph H^2, with vertex set V and edges corresponding to paths of length 2 in H, is N^δ-expanding.

We must be somewhat careful, however. Nothing in the definition of extractor prevents every vertex in V from having the neighbor 0^m (say) in H, making H^2 a clique. We therefore form the graph G from H by deleting vertices in W that have degree more than twice the average for nodes in W, i.e. more than $2N2^t/M$. In this way, we retain a set of undeleted nodes W', which has size $M' \geq (1-\epsilon)M = 3M/4$ (if more

than ϵM nodes had twice the average probability of being hit, then $E(x,y)$ could not be uniform within ϵ). Moreover, any two subsets of V of size N^δ have at least $M/4$ common neighbors. Thus the graph G^2 is the one we seek. □

Substituting in the extractor of [NZ] gives a construction which beats the eigenvalue bound, but is not near-optimal. To get a near-optimal construction, we need to make m close to δn, while keeping t small. The idea for doing this is to repeatedly apply existing, sub-optimal extractors to the same defective source, with fresh values of y. As long as we didn't get all the entropy out from the source, even when we condition on what we output so far, some entropy remained and we can apply the extractor again to get more out. This idea is formalized in the next two lemmas. In them, the word "efficient" can have almost any reasonable meaning, such as computable in Logspace, NC, or polynomial-time.

Lemma 2.4 *Given an efficient $(n, m_1, t_1, \delta, \epsilon_1)$-extractor E_1 and an efficient $(n, m_2, t_2, \delta-(m_1+k)/n, \epsilon_2)$-extractor E_2, we can construct an efficient $(n, m_1+m_2, t_1+t_2, \delta, \epsilon_1+\epsilon_2+2^{-k})$-extractor E.*

Proof: We define $E(x, y_1 \circ y_2) = E_1(x, y_1) \circ E_2(x, y_2)$. Suppose X is output according to a δ-source on n bits, and $Y_1 \circ Y_2$ is chosen uniformly from $\{0,1\}^{t_1+t_2}$. Let D denote the distribution of the random variable $W_1 = E_1(X, Y_1) \circ Y_1$, which is uniform within ϵ_1. Clearly, the probability a random value w of W_1 chosen according to D has probability weight $D(w) \leq 2^{-(m_1+t_1+k)}$ is at most $2^{m_1+t_1} \cdot 2^{-(m_1+t_1+k)} = 2^{-k}$. But for w satisfying $D(w) \geq 2^{-(m_1+t_1+k)}$, when we condition on $E_1(X, Y_1) \circ Y_1 = w$, the distribution of X is a $(\delta-(m_1+k)/n)$-source, so $W_2 = E_2(X, Y_2) \circ Y_2$ is uniform within ϵ_2. Removing the conditioning, we conclude that $W_1 \circ W_2$ is uniform within $\epsilon_1 + 2^{-k} + \epsilon_2$, as required. □

This allows us to recurse as in the following lemma:

Lemma 2.5 *Fix positive integers n and k. Suppose there is a family of efficient $(n, m(\delta), t(\delta), \delta, \epsilon(\delta))$-extractors E_δ, one for each $\delta \in [\eta, 1]$, where t and ϵ are non-increasing functions of δ. Let $f(\delta) = m(\delta)/(\delta n)$, and suppose f is non-decreasing. Then we can construct an efficient $(n, (\delta-\eta)n-k, r \cdot t(\eta), \delta, r(\epsilon(\eta)+2^{-k}))$-extractor, where $r = 1 + \ln(\delta/\eta)/f(\eta)$. If f grows at least linearly (i.e. $f(c\delta) \geq cf(\delta)$ for $c > 1$), then we can take $r = \lceil 1/f(\eta) \rceil$.*

Proof: We recurse using Lemma 2.4 with parameter k, defining extractors inductively as follows. Let $E^{(1)} = E_\delta$, and for $i > 1$ let

$$E^{(i+1)}(x, y^{(i)} \circ y_{i+1}) = E^{(i)}(x, y^{(i)}) \circ E_{\delta_{i+1}}(x, y_{i+1}),$$

where we define δ_{i+1} inductively as follows. Namely, let $E^{(i)}$ output $m^{(i)}$ bits. Then $\delta_{i+1} = \delta - (m^{(i)}+k)/n$. We stop recursing at the integer s where $\delta_{s+1} < \eta$, and hence $m^{(s)} > (\delta-\eta)n-k$.

Let m_i, t_i, ϵ_i denote $m(\delta_i)$, $t(\delta_i)$, and $\epsilon(\delta_i)$, respectively, so that E_{δ_i} is an $(n, m_i, t_i, \delta_i, \epsilon_i)$-extractor. Applying Lemma 2.4 inductively shows that $E^{(i)}$ is an $(n, m^{(i)}, t^{(i)}, \delta, \epsilon^{(i)})$-extractor, where $m^{(i)} = \sum_{j=1}^{i} m_j$, $t^{(i)} = \sum_{j=1}^{i} t_j \leq r \cdot t(\eta)$, and $\epsilon^{(i)} = (\sum_{j=1}^{i} \epsilon_j) + (r-1)2^{-k} \leq r(\epsilon(\eta)+2^{-k})$.

We show that $s \leq r$, from which the lemma follows. Subtracting expressions for δ_{i+1} and δ_i gives $\delta_{i+1} = \delta_i - m_i/n = \delta_i(1-f(\delta_i))$. To see $s \leq 1+\ln(\delta/\eta)/f(\eta)$, we use $f(\delta_i) \geq f(\eta)$, so $\delta(1-f(\eta))^{s-1} \geq \delta_s \geq \eta$.

If f grows at least linearly, let $\gamma = f(\eta)/\eta$. By the linear growth of f, $f(\delta_i) \geq \delta_i\gamma$, and hence $\delta_{i+1} \leq \delta_i(1-\gamma\delta_i)$. Now let $b_i = 1/\delta_i$. Then $b_{i+1} \geq b_i(1-\gamma/b_i)^{-1} \geq b_i(1+\gamma/b_i) = b_i + \gamma$. Hence $b_s \leq 1/\eta$ implies $s \leq \lceil (\gamma\eta)^{-1} \rceil = \lceil 1/f(\eta) \rceil$, as required. □

We can now prove our main theorem, which we restate:

Theorem 1.2: There is a Logspace algorithm that, on input N (in unary) and δ, where $0 < \delta = \delta(N) < 1$, constructs an N^δ-expanding graph on N nodes with maximum degree $O^*(N^{1-\delta})$.

Proof: Assume without loss of generality that N is a power of 2, so $N = 2^n$. Set $\eta = (\log^5 n/n)^{1/3}$, $\epsilon = 1/n$, and $k = \log n$. If $\delta < \eta$, then the complete graph satisfies the theorem.

Otherwise, apply Lemma 2.5 to the extractor given by Lemma 2.2. Since $f(\delta) = \Omega(\delta/\log\delta^{-1})$, which grows at least linearly, $r = O(\log\eta^{-1}/\eta) = O(\log n/\eta)$. Therefore Lemma 2.5 yields an $(n, m, t, \delta, \epsilon)$-extractor

with $m = (\delta - \eta)n - \log n$, $t = O(\log^5 n/\eta^2)$, and $\epsilon = o(1)$. Then Lemma 2.3 gives an N^δ-expanding graph, where the logarithm of the maximum degree is

$$n + 1 + 2t - m = (1-\delta)n + O((\log^5 n)/\eta^2 + \eta n) = (1-\delta)n + n^{2/3+o(1)},$$

as needed. □

3 Sorting and Selecting in Rounds

The following is implicit in Pippenger's work:

Lemma 3.1 *[Pip3] Suppose that for all $1/2 \le \delta < 1$ there are explicitly-constructible n^δ-expanding graphs with maximum degree $n^{1-\delta} f(n)$. Then there are explicit algorithms for sorting and selecting in k rounds using $O(n^{1+1/k} f(n) \log n)$ and $O(n^{1+1/(2^k-1)} f(n) \log n)$ comparisons, respectively.*

Proof: Use $a = n^{1-1/k}/\log n$ and $a = n^{1-1/(2^k-1)}$ in Pippenger's proofs of Theorems 2 and 1, respectively. □

This immediately yields:

Theorem 3.2 *There are explicit algorithms for sorting and selecting in k rounds using $O^*(n^{1+1/k})$ and $O^*(n^{1+1/(2^k-1)})$ comparisons, respectively.*

Proof: Use Lemma 3.1 with the graphs constructed in Theorem 1.2. □

The following lemma about almost-sorting in 1 round appears in [AKSS]:

Lemma 3.3 *[AKSS] If G is an a-expanding graph, then after performing the comparisons according to G, all relations will be known except for $O(an \log n)$.*

This immediately gives:

Theorem 3.4 *There are explicit algorithms to find all relations except $n^{2-\epsilon}$ in one round using $O^*(n^{1+\epsilon})$ comparisons.*

Proof: Perform comparisons according to a $cn^{1-\epsilon}/\log n$-expanding graph constructed via Theorem 1.2. □

4 Superconcentrators

In this section, we explicitly construct superconcentrators of depth 2 and size $O^*(n)$. In order to construct our networks, we use as building blocks n^δ-expanding weak concentrators:

Definition 4.1 *An a-expanding weak (n,m)-concentrator is an (n,m)-network of depth 1 in which every subset of the inputs of size a expands to more than $m - a$ outputs.*

Note that these are not concentrators in the usual sense.

Lemma 4.2 *For all n, $0 < \delta = \delta(n) < 1$, $2 \le r = r(n) \le O(n^{1-\delta})$, there are explicitly-constructible n^δ-expanding weak (n, rn^δ)-concentrators of size $O^*(rn)$*

Proof: By Theorem 1.2, we can construct an n^δ-expanding graph G on $n + rn^\delta$ nodes with maximum degree $O^*(n^{1-\delta})$. Form an (n, rn^δ)-network $H = (V \cup W, E)$ by letting the outputs W be any rn^δ vertices, and V the rest. Remove all edges not between V and W. Since G is n^δ expanding, H is an n^δ-expanding weak (n, rn^δ)-concentrator. Moreover, $|E| \le rn^\delta O^*(n^{1-\delta}) = O^*(rn)$. □

It is convenient to use the following characterization of depth 2 superconcentrators, due to [Mes]. Let $N = (I \cup M \cup O, F)$ be an n-network of depth 2 with inputs I, middle layer M, and outputs O. For $X \subseteq I$ and $Y \subseteq O$, define

$$\Gamma^+(X) = \{z \in M : (x,z) \in F \text{ for some } x \in X\},$$
$$\Gamma^-(Y) = \{z \in M : (z,y) \in F \text{ for some } y \in Y\}.$$

Lemma 4.3 *[Mes] N is a superconcentrator if and only if for any $1 \leq k \leq n$ and $X \subseteq I$, $Y \subseteq O$ such that $|X| = |Y| = k$, $|\Gamma^+(X) \cap \Gamma^-(Y)| \geq k$.*

This motivates the following definition.

Definition 4.4 *An (a, b)-partial n-superconcentrator of depth 2 is an n-network $N = (I \cup M \cup O, F)$ of depth 2, such that for any $a \leq k \leq b$ and $X \subseteq I$, $Y \subseteq O$ with $|X| = |Y| = k$, $|\Gamma^+(X) \cap \Gamma^-(Y)| \geq k$.*

Lemma 4.5 *For all n, $0 < \delta = \delta(n) < 1$, $2 \leq r = r(n) \leq O(n^{1-\delta})$, there are explicitly-constructible (n^δ, rn^δ)-partial n-superconcentrators of depth 2 having size $O^*(rn)$.*

Proof: By Lemma 4.2, we can construct H, an $n^\delta/2$-expanding weak $(n, (r+1)n^\delta)$-concentrator of size $O^*(rn)$. The network $N = (I \cup M \cup O, F)$ satisfying the conditions of the lemma will have a copy of H between I and M and a copy of the reverse of H between M and O. Suppose $X \subseteq I$, $Y \subseteq O$ with $|X| = |Y| = k$, $n^\delta \leq k \leq rn^\delta$. Then both $|\Gamma^+(X)|$ and $|\Gamma^-(Y)|$ are at least $(r+1)n^\delta - n^\delta/2$. Thus $|\Gamma^+(X) \cap \Gamma^-(Y)| \geq rn^\delta \geq k$. □

Theorem 4.6 *For all n, there are explicitly-constructible n-superconcentrators of depth 2 and size $O^*(n)$.*

Proof: Construct the union of $(2^{i-1}, 2^i)$-partial n-superconcentrators of depth 2, $i = 1, \ldots \lg n$. Lemma 4.3 implies that this is a superconcentrator. □

We now show how this construction can be used to achieve linear-sized superconcentrators with sublogarithmic depth.

Lemma 4.7 *If there are explicitly-constructible n-superconcentrators of size an and depth k, then there are explicitly-constructible n-superconcentrators of linear size and depth $k + O(\log a)$.*

Proof: (Sketch) Use the recursive superconcentrator construction developed by Pippenger [Pip1]. After $O(\log a)$ levels, we need an n/a-superconcentrator. Assuming $a = a(n)$ is a non-decreasing function of n, we use the n/a-superconcentrator of size at most n. □

Theorem 4.8 *For all n, there are explicitly-constructible n-superconcentrators of linear size and depth $(\log n)^{2/3+o(1)}$.*

Proof: Use Lemma 4.7 and Theorem 4.6. □

5 Concentrators and Non-Blocking Networks

In this section we show how similar ideas can be used to explicitly construct non-blocking networks. Before we do this, we define wide-sense nonblocking generalized connectors, following [FFP].

A *route* in a network is a directed path from an input to an output. A *state* of a network is a set of vertex-disjoint routes. The states of a network are partially ordered by inclusion; *above* and *below* refer to this partial order. A *connection request* is an input-output pair. A connection request (v, w) is *legal* with respect to a state s if v and w are not in any route contained in s. A connection request (v, w) is *satisfied* by a route if the route begins at v and ends at w.

Finally, a *wide-sense nonblocking generalized n-connector* is an n-network for which there exists a set of distinguished states, called *safe* states, with the following properties:

1. the empty set is safe;
2. any state below a safe state is safe;
3. given any safe state s and any legal connection request (v, w) with respect to s, there exists a safe state above s containing a route satisfying (v, w).

The key to our result is to construct a certain generalization of concentrators, and then apply a lemma of [FFP].

Definition 5.1 An (n, m, l)-concentrator with expansion e *is an* (n, m)*-network such that every set of* $t \leq l$ *inputs expands to at least* et *outputs.*

Note that a concentrator with expansion 1 is a concentrator in the usual sense.

Theorem 5.2 *For all* n, $0 < \delta = \delta(n) < 1$, $1 \leq e = e(n) \leq n^{1-\delta}/2$, *there are explicitly-constructible* $(n, 2en^{\delta}, n^{\delta})$*-concentrators with expansion* e *of size* $O^*(en)$ *and depth 1.*

Proof: The concentrator C we seek is the union of 2^i-expanding weak $(n, 4e2^i)$-concentrators C_i, $i = 0, \ldots, \lg(n^{\delta}/2)$; the outputs of C_i are, say, the first $4e2^i$ outputs of C. Suppose we have a set of t inputs, and say $t \in [2^i, 2^{i+1}]$. Then in C_i this set must expand to $(4e-1)2^i > et$. □

The following lemma is implicit in [FFP]:

Lemma 5.3 *[FFP] If for all* $1/2 \leq \delta < 1$ *there are explicitly-constructible* $(n, 4n^{\delta}, n^{\delta})$*-concentrators with expansion 2 and size* $O^*(n)$, *then there are efficiently constructible wide-sense nonblocking generalized* n*-connectors of size* $O^*(n^{1+1/k})$ *and depth* k.

Thus, the following theorem is immediate.

Theorem 5.4 *For all* k *and* n, *there are efficiently constructible wide-sense nonblocking generalized* n*-connectors of size* $O^*(n^{1+1/k})$ *and depth* k.

6 Subsequent Work

Subsequent to this work, the $n^{o(1)}$ factors have been improved twice [SZ, TS96], by constructing stronger extractors and applying our methods. In the most recent improvement, Ta-Shma [TS96] obtained expanders where the $n^{o(1)}$ factors are $\exp((\log\log n)^{O(1)})$. Hence all the applications have these new $n^{o(1)}$ factors and the depth of the linear-sized superconcentrator is $(\log\log n)^{O(1)}$.

Acknowledgements

We thank Noga Alon, Nabil Kahale, Nick Pippenger, and Greg Plaxton for helpful comments and discussions. We also thank the anonymous referees for helpful suggestions.

References

[AKS1] M. Ajtai, J. Komlos, and E. Szemeredi, "Sorting in $c \log n$ Parallel Steps," *Combinatorica* 3 (1983), pp. 1-19.

[AKS2] M. Ajtai, J. Komlos, and E. Szemeredi, "Deterministic Simulation of Logspace." In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, 1987, pp. 132-140.

[AKSS] M. Ajtai, J. Komlos, W. Steiger, and E. Szemeredi, "Almost Sorting in One Round," *Advances in Computing Research*, Vol. 5, 1989, pp. 117-125.

[Alo1] N. Alon, "Eigenvalues, Geometric Expanders, Sorting in Rounds, and Ramsey Theory," *Combinatorica* 6 (1986), pp.207-219.

[Alo2] N. Alon, "Eigenvalues and Expanders," *Combinatorica* 6 (1986), pp. 83-96.

[AA] N. Alon and Y. Azar, "Sorting, Approximate Sorting, and Searching in Rounds," *SIAM J. Disc. Math.* 1 (1988), pp. 269-280.

[AM] N. Alon and V.D. Milman, "λ_1, Isoperimetric Inequalities for Graphs and Superconcentrators," *J. Combinatorial Theory Ser. B* 38 (1985), pp. 73-88.

[AP] Y. Azar and N. Pippenger, "Parallel Selection," *Discrete Appl. Math.* 27 (1990), pp. 49-58.

[BT] B. Bollobas and A. Thomason, "Parallel Sorting," *Discrete Appl. Math.* 6 (1983), pp. 1-11.

[DDPW] D. Dolev, C. Dwork, N. Pippenger, and A. Wigderson, "Superconcentrators, Generalizers, and Generalized Connectors with Limited Depth." In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, 1983, pp. 42-51.

[FFP] P. Feldman, J. Friedman, and N. Pippenger, "Wide-Sense Nonblocking Networks," *SIAM J. Disc. Math.*, 1 (1988), pp. 158-173.

[GG] O. Gabber and Z. Galil, "Explicit Construction of Linear Sized Superconcentrators," *J. Comp. and Sys. Sci* 22 (1981), pp. 407-420.

[GIL+] O. Goldreich, R. Impagliazzo, L. Levin, R. Venkatesan, and D. Zuckerman, "Security Preserving Amplification of Hardness." In *Proceedings of the 31st Annual IEEE Symposium on Foundations of Computer Science*, 1990, pp. 318-326.

[Kah] N. Kahale, "Eigenvalues and expansion of regular graphs," *Journal of the ACM*, 42 (1995), pp. 1091-1106.

[LPS] A. Lubotzky, R. Philips, P. Sarnak, "Ramanujan Graphs," *Combinatorica* 8 (1988), pp. 261-277.

[Mar] G.A. Margulis, "Explicit Construction of Concentrators," *Problems of Inform. Transmission*, pp. 325-332.

[Mes] R. Meshulam, "A Geometric Construction of a Superconcentrator of Depth 2," *Theoretical Computer Science* 32 (1984), pp. 215-219.

[Mor] M. Morgenstern, "Explicit Construction of Natural Bounded Concentrators." In *Proceedings of the 32nd Annual IEEE Symposium on Foundations of Computer Science*, 1991, pp. 392-397.

[NZ] N. Nisan and D. Zuckerman, "Randomness is Linear in Space," *Journal of Computer and System Sciences*, 52 (1996), pp. 43-52.

[Pip1] N. Pippenger, "Superconcentrators," *SIAM J. Comput.* 6 (1977), pp. 298-304.

[Pip2] N. Pippenger, "Superconcentrators of Depth 2," *J. Comp. and Sys. Sci.* 24 (1982), pp. 82-90.

[Pip3] N. Pippenger, "Sorting and Selecting in Rounds," *SIAM J. Comput.* 16 (1987), pp. 1032-1038.

[PY] N. Pippenger and A.C. Yao, "Rearrangeable Networks with Limited Depth," *SIAM J. Algebraic and Discrete Methods* 3 (1982), pp. 411-417.

[SZ] A. Srinivasan and D. Zuckerman, "Computing with Very Weak Random Sources." In *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science*, 1994, pp. 264-275. To appear in *SIAM Journal on Computing*.

[Tan] R.M. Tanner, "Explicit Construction of Concentrators from Generalized N-gons," *SIAM J. Alg. Discr. Meth.* 5 (1984), pp. 287-293.

[TS96] A. Ta-Shma, "On Extracting Randomness from Weak Random Sources." In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, 1996, pp. 276-285.

[Tom] M. Tompa, "Time Space Tradeoffs for Computing Functions, Using Connectivity Properties of Their Circuits," *J. Comp. and Sys. Sci*, 20 (1980), pp. 118-132.

[Val1] L.G. Valiant, "Parallelism in Comparison Problems," *SIAM J. Comput.* 4 (1975), pp. 348-355.

[Val2] L.G. Valiant, "Graph Theoretic Properties in Computational Complexity," *J. Comp. and Sys. Sci.* 13 (1976), pp. 278-285.

[Zuc1] D. Zuckerman, "General Weak Random Sources." In *Proceedings of the 31st Annual IEEE Symposium on Foundations of Computer Science*, 1990, pp. 534-543.

[Zuc2] D. Zuckerman, "Simulating BPP Using a General Weak Random Source," *Algorithmica*, 16 (1996), pp. 367-391.

[Zuc3] D. Zuckerman. "Randomness-Optimal Oblivious Sampling," *Random Structures and Algorithms*, 11 (1997), pp. 345-367.