Correspondence

On Expander Codes

Gilles Zémor

***Abstract*—Sipser and Spielman have introduced a constructive family of asymptotically good linear error-correcting codes—expander codes—together with a simple parallel algorithm that will always remove a constant fraction of errors. We introduce a variation on their decoding algorithm that, with no extra cost in complexity, provably corrects up to 12 times more errors.**

***Index Terms*—Decoding, expander code, Ramanujan graph.**

I. Introduction

In [1], Sipser and Spielman introduced a constructive family of asymptotically good linear error-correcting codes together with a simple parallel algorithm that will always remove a constant fraction of errors.

More precisely, they prove [1, Theorem 19].

Theorem 1 (Sipser–Spielman): For all δ_0 such that $1-2H(\delta_0)>0$, where $H(\cdot)$ is the binary entropy function, there exists a polynomial-time constructible family of expander codes of rate $1-2H(\delta_0)$ and minimum relative distance arbitrarily close to δ_0^2 in which any $\alpha < \delta_0^2/48$ fraction of error can be corrected by a circuit of size $O(N \log N)$ and depth $O(\log N)$ where N is the length of the code.

In one of the two open questions of [1], Sipser and Spielman ask whether one can obtain better constants from the construction of the above theorem.

To obtain such an improvement we shall introduce a variation on their decoding algorithm. It will enable us to replace in Theorem 1 the correction of "any $\alpha < \delta_0^2/48$" fraction of error by "any $\alpha < \delta_0^2/4$." Formally, Theorem 1 is improved to the following.

Theorem 1A: For all δ_0 such that $1-2H(\delta_0) > 0$, where $H(\cdot)$ is the binary entropy function, there exists a polynomial-time constructible family of expander codes of rate $1-2H(\delta_0)$ and minimum relative distance arbitrarily close to δ_0^2 in which any $\alpha < \delta_0^2/4$ fraction of error can be corrected by a circuit of size $O(N \log N)$ and depth $O(\log N)$.

II. Context and Main Result

A. Codes and Graphs

The following construction of a binary code was first proposed by Tanner [2]. Take a bipartite graph with vertex set $E \cup V$ where edges exist only between vertices of E and vertices of V: suppose that every vertex of V is adjacent to exactly Δ vertices of E. Number the vertices of E, i.e., let $E = \{1, 2, \dots N\}$. Let C_0 be a linear code of length Δ. For any vertex $v \in V$ define $v(1), v(2), \dots, v(\Delta)$ to be some ordering of the Δ vertices of E adjacent to v. Define the code C to be the set of binary vectors $x = (x_1, x_2, \dots, x_N)$ of $\{0, 1\}^N$ such that, for every vertex v of V, $(x_{v(1)}, x_{v(2)}, \dots, x_{v(\Delta)})$ is a codeword of C_0.

In this correspondence, we shall be concerned solely with the special case when every vertex of E is adjacent to exactly two vertices of V. Equivalently, this means that V and E make up, respectively, the vertex set and edge set of a Δ-regular graph G. Let us call the code C constructed in this way a (G, C_0)-code. Note that for a given graph G and a given code C_0 there are several (G, C_0)-codes since there are several ways of numbering the edges of G and several ways of ordering the edges incident to a given vertex v, i.e., $v(1), v(2), \dots, v(\Delta)$.

Graphs G will be assumed to be connected and without multiple edges.

B. Expander Codes

Suppose the code C_0 has length Δ, dimension k_0, redundancy $r_0 = \Delta - k_0$, and minimum distance d_0. Then the dimension of a (G, C_0)-code C is at least $N(1-2r_0/\Delta)$ [2]. Sipser and Spielman [1] have found an elegant lower bound on the minimum distance D of C of the form

$$D \geq N\delta_0^2(1-\varepsilon) \tag{1}$$

where $\delta_0 = d_0/\Delta$ is the relative minimum distance of C_0 and ε depends only on d_0 and graphical parameters of G. More precisely, ε is a function of d_0, Δ, and λ, the second largest eigenvalue of the adjacency matrix of G. Furthermore, ε is such that $\varepsilon \to 0$ when $\lambda/d_0 \to 0$. This result becomes especially interesting when one brings in the Ramanujan graphs of [3], [4]. These families of graphs are constructive, have an arbitrarily large number of vertices for fixed degrees Δ, and satisfy $\lambda \leq 2\sqrt{\Delta - 1}$. By choosing G to be Ramanujan and fixing a large enough Δ we can, therefore, make ε as small as we like and obtain constructions of asymptotically good (G, C_0)-codes that Sipser and Spielman named "expander codes" in reference to the expanding properties of Ramanujan graphs.

For example, if C_0 is chosen to be a shortened extended Bose–Chaudhuri–Hocquenghem (BCH) code of length $\Delta = 224$, dimension $k_0 = 115$, and minimum distance proved to satisfy $d_0 \geq 30$ [5] then it can be checked that Ramanujan graphs G of degree 224 (constructed in [4], [3]) yield asymptotically good (G, C_0) codes.

Furthermore, Sipser and Spielman exhibit a decoding algorithm of low complexity, namely, time $O(\log N)$ for a circuit of size $O(N \log N)$ that, for any fixed $\alpha < 1$, will always return the original codeword provided the error vector has weight less than $\alpha N\delta_0^2(1-\varepsilon)/48$: again, ε is a quantity that depends only on d_0, Δ, and λ and is such that $\varepsilon \to 0$ when $\lambda/d_0 \to 0$.

Since Δ can be fixed and N can grow to infinity one can choose the best known codes for C_0, i.e., on the Varshamov–Gilbert bound: this yields Theorem 1.

We shall study a variation on their decoding scheme for (G, C_0)-codes that yields the following.

Theorem 2: Let $\alpha < 1$ be fixed. When G is a bipartite graph there is a decoding algorithm for (G, C_0) codes that can be implemented as a circuit of size $O(N \log N)$ and depth $O(\log N)$, that always returns the original codeword provided the error vector has weight less than $\alpha N\delta_0^2(1-\varepsilon)/4$: ε is a quantity that depends only on d_0, Δ, and λ, and is such that $\varepsilon \to 0$ when $\lambda/d_0 \to 0$.

Manuscript received December 13, 1999; revised September 11, 2000.

The author is with École Nationale Supérieure des Télécommunications, 75 634 Paris 13, France (e-mail: zemor@infres.enst.fr).

Communicated by D. A. Spielman, Guest Editor.

Publisher Item Identifier S 0018-9448(01)00734-9.

Hence, Theorem 1A. The restriction that G be bipartite is not a problem since this is the case of about half the known constructions of Ramanujan graphs.

III. Decoding

A. The Decoding Algorithm

This algorithm works when G is a *bipartite* Δ-regular graph.

Let the set of vertices of G be $V = A \cup B$ where $|A| = |B| = n$ and where every edge of G has one endpoint in A and one in B.

For any vertex v of G the subset of edges incident to v is

$$E_v = \{v(1), v(2), \ldots, v(\Delta)\}.$$

Because G is bipartite, the set A of vertices induces the partition of the edge set $E = \bigcup_{v \in A} E_v$. The set B induces similarly a second partition, namely, $E = \bigcup_{v \in B} E_v$.

Let $x \in \{0, 1\}^N$ be the received vector, and recall that $N = \Delta n$. The first iteration of the algorithm consists of applying complete decoding for the code induced by E_v for every $v \in A$. This means replacing, for every $v \in A$, the vector $(x_{v(1)}, x_{v(2)}, \ldots, x_{v(\Delta)})$ by one of the closest codewords of C_0 (which might not be unique). Because the subsets of edges E_v are disjoint for $v \in A$, the decoding of these n subvectors of x may be done in parallel.

This iteration yields a new vector y. The next iteration consists of applying the preceding procedure to y but with A replaced by B. In other words, it consists of decoding all the subvectors induced by the vertices of B. The next iterations repeat those two steps, alternately applying parallel decoding to the subvectors induced by the vertices of A and to the subvectors induced by the vertices of B.

Theorem 6 of the next section will give a sufficient condition on the number of corrupted bits for this algorithm to converge.

Remark: If $\Delta = n$ and G is the complete bipartite graph, then C is a product code of C_0 with itself and the above algorithm reduces to the natural hard iterative decoding of product codes.

B. Analysis

In Sipser and Spielman's analysis of (G, C_0)-codes the basic tool is the following result of Alon and Chung [6] upper-bounding the average degree of an induced subgraph. The *degree* of a vertex is the number of edges incident to it. If S is a subset of vertices of G the *subgraph induced by* S is the graph G_S with vertex set S and edge set E_S where E_S is the set of all the edges of G that have both endpoints in S. The *average degree* of G_S is $\overline{d}_S = 2|E_S|/|S|$. We have the following lemma.

Lemma 3 (Alon–Chung): Let G be a Δ-regular graph on n vertices with second largest eigenvalue λ. Let S be a subset of vertices. Then the average degree $\overline{d}_S$ of the subgraph induced by S satisfies

$$\overline{d}_S \le \Delta \frac{|S|}{n} + \lambda \left(1 - \frac{|S|}{n}\right).$$

We shall need the following generalization of Lemma 3.

Lemma 4: Let G be a Δ-regular bipartite graph with vertex set $A \cup B$ where $|A| = |B| = n$ and where every edge has one endpoint in A and one in B. Let $S \subset A$ and $T \subset B$. Then the average degree $\overline{d}_{ST}$ of the subgraph induced by $S \cup T$ satisfies

$$\overline{d}_{ST} \le \frac{2|S||T|}{|S|+|T|} \frac{\Delta}{n} + \lambda - \frac{\lambda}{n} \frac{|S|^2 + |T|^2}{|S|+|T|}.$$

Note that this last upper bound reduces to that of Lemma 3 when $|S| = |T|$ but is more precise otherwise.

Since the proof of Lemma 4 is fairly independent of the decoding issues we postpone it to the next section. For our purposes its principal consequence is the following lemma which summarizes the technical part of our analysis.

Lemma 5: Suppose $d_0 \ge 3\lambda$. Let S be a subset of vertices of A such that

$$|S| \le \alpha n \left(\frac{\delta_0}{2} - \frac{\lambda}{\Delta}\right) \tag{2}$$

where $\alpha < 1$. Let T be a subset of vertices of B and suppose that there exists a set $Y \subset E$ of edges such that

1) every edge of Y has one of its endpoints in S;
2) every vertex of T is incident to at least $d_0/2$ edges of Y.

Then

$$|T| \le \frac{1}{2-\alpha} |S|.$$

Proof: Let $W \subset Y$ consist of those edges of Y that have one endpoint in T. Then W must be included in the set of edges of $G_{S \cup T}$, the subgraph induced by $S \cup T$. Therefore, $G_{S \cup T}$ has the average degree greater or equal to $2|W|/(|S|+|T|)$ and by point 2) we must have $|T|d_0/2 \le |W|$, therefore, by Lemma 4

$$\frac{|T|d_0}{|S|+|T|} \le \frac{2|S||T|}{|S|+|T|} \frac{\Delta}{n} + \lambda$$

whence, applying (2) and $\delta_0 \Delta = d_0$

$$|T|d_0 \le \alpha(d_0 - 2\lambda)|T| + \lambda(|S| + |T|)$$
$$|T| \le \frac{\lambda}{d_0(1-\alpha) + \lambda(2\alpha - 1)} |S|.$$

Hence the result, using $d_0 \ge 3\lambda$. □

We are now ready to prove the following theorem.

Theorem 6: Suppose $d_0 \ge 3\lambda$. If the weight of the error vector x satisfies

$$|x| \le \alpha \frac{\delta_0}{2} \left(\frac{\delta_0}{2} - \frac{\lambda}{\Delta}\right) N \tag{3}$$

for any $\alpha < 1$, then the algorithm of Section III-A converges to the initial codeword in a number of parallel steps logarithmic in N.

Proof: Because of linearity we may suppose, without loss of generality, that the initial uncorrupted codeword is the zero codeword. Let x be the error vector and let us identify it with the corresponding set of edges $X = \{i, x_i = 1\}$. Let y be the vector obtained from x after the first decoding step (induced by A), and let $Y = \{i, y_i = 1\}$ be the corresponding set of edges. Let z be the vector obtained after the *second* decoding step and let Z be the corresponding set of edges.

We start by looking at the partitions of X and of Y induced by $(E_v)_{v \in A}$. Let $v \in A$. The key observation is that if v is incident to less than $d_0/2$ edges of X, then these will be totally erased by the decoding procedure, i.e., $E_v \cap Y = \emptyset$. Let S be the set of vertices v of A such that $E_v \cap Y \ne \emptyset$. We have just observed that

$$v \in S \quad \text{implies} \quad |E_v \cap X| \ge d_0/2. \tag{4}$$

Similarly, let T be the set of vertices v of B such that $E_v \cap Z \ne \emptyset$; we also have, for the same reason

$$v \in T \quad \text{implies} \quad |E_v \cap Y| \ge d_0/2. \tag{5}$$

Let us now check that S, T, and Y satisfy the conditions of Lemma 5.

Observation (4) implies $|X| \geq |S|d_0/2$. Together with (3) and since $N = \Delta n$, this implies that $|S|$ satisfies (2). Point 1) of Lemma 5 holds by the definition of S. Point 2) of Lemma 5 holds by (5).

Therefore, the conclusion of Lemma 5 holds and we have

$$|T| \leq \beta |S|$$

with $\beta = 1/(2-\alpha) < 1$.

The proof of convergence consists of repeating the argument. For $i \geq 1$ let $X^{(i)}$ be the set of edges in error after decoding step i. For $i \geq 0$, let $S^{(i)}$ be the set of vertices defined as

- $S^{(i)} = \{v \in A,\ E_v \cap X^{(i+1)} \neq \emptyset\}$ if i is even
- $S^{(i)} = \{v \in B,\ E_v \cap X^{(i+1)} \neq \emptyset\}$ if i is odd.

We have just proved that $|S^{(1)}| \leq \beta|S|$. Therefore, $S^{(1)}$ also satisfies (2) and we have $|S^{(2)}| \leq \beta|S^{(1)}|$ and more generally $|S^{(i)}| \leq \beta^i |S|$. When $S^{(i)} = \emptyset$ then $X^{(i+1)} = \emptyset$ and the decoding is complete. □

Remark: The above proof consisted of showing that the sets $S^{(i)}$ have strictly decreasing cardinalities. The weight of the error vector, however, does not necessarily decrease at each iteration.

Theorem 2 is a direct consequence of Theorem 6.

IV. A Proof of Lemma 4

The proof is very much in the spirit of [6].

Let $\boldsymbol{A} = (\boldsymbol{a_{ij}})$ be the $2n \times 2n$ adjacency matrix of the bipartite graph G, i.e., $a_{ij} = 1$ if the vertex indexed by i is adjacent to the vertex indexed by j and $a_{ij} = 0$ otherwise; a fixed ordering of the vertices is assumed but does not influence the computations to come. Let $\boldsymbol{X}_{ST}$ be the column vector of length $2n$ such that every coordinate indexed by a vertex of S or of T equals 1 and the other coordinates equal 0. It is straightforward to check that

$$^t\boldsymbol{X}_{ST}\boldsymbol{A}\boldsymbol{X}_{ST} = \sum_{\boldsymbol{v} \in \boldsymbol{S} \cup \boldsymbol{T}} \boldsymbol{d}_{\boldsymbol{G}_{\boldsymbol{S}\cup\boldsymbol{T}}}(\boldsymbol{v}) \tag{6}$$

where $d_{G_{S\cup T}}(v)$ stands for the degree of v in the subgraph induced by $S \cup T$, i.e., the number of neighbors of v that belong to S or to T.

Now let $\boldsymbol{j}$ be the all-one vector and let $\boldsymbol{k}$ be the vector such that every coordinate indexed by a vertex of A equals 1 and every coordinate indexed by a vertex of B equals -1. $\boldsymbol{j}$ and $\boldsymbol{k}$ are eigenvectors of $\boldsymbol{A}$ associated to the eigenvalues Δ and $-\Delta$, respectively. Next define $\boldsymbol{Y}_{ST}$ as the vector such that

$$\boldsymbol{X}_{ST} = \frac{|\boldsymbol{S}| + |\boldsymbol{T}|}{\boldsymbol{2n}}\,\boldsymbol{j} + \frac{|\boldsymbol{S}| - |\boldsymbol{T}|}{\boldsymbol{2n}}\,\boldsymbol{k} + \boldsymbol{Y}_{ST}.$$

It is straightforward to check that $\boldsymbol{Y}_{ST}$ is orthogonal to $\boldsymbol{j}$ and $\boldsymbol{k}$. Because the eigenspaces of $\boldsymbol{A}$ are orthogonal we can therefore write

$$^t\boldsymbol{X}_{ST}\boldsymbol{A}\boldsymbol{X}_{ST} = \left(\frac{|S| + |T|}{2n}\right)^2 \Delta \boldsymbol{j}\cdot\boldsymbol{j} - \left(\frac{|\boldsymbol{S}| - |\boldsymbol{T}|}{\boldsymbol{2n}}\right)^2 \Delta \boldsymbol{k}\cdot\boldsymbol{k} + {}^t\boldsymbol{Y}_{ST}\boldsymbol{A}\boldsymbol{Y}_{ST}$$

which reduces to, since $\boldsymbol{j}\cdot\boldsymbol{j} = \boldsymbol{k}\cdot\boldsymbol{k} = \boldsymbol{2n}$,

$$^t\boldsymbol{X}_{ST}\boldsymbol{A}\boldsymbol{X}_{ST} = {}^t\boldsymbol{Y}_{ST}\boldsymbol{A}\boldsymbol{Y}_{ST} + \boldsymbol{2}\frac{|\boldsymbol{S}||\boldsymbol{T}|}{\boldsymbol{n}}\Delta.$$

Now, since $\boldsymbol{Y}_{ST}$ is orthogonal to $\boldsymbol{j}$ and since the eigenspace associated to the eigenvalue Δ is of dimension one (G is connected) we have $^t\boldsymbol{Y}_{ST}\boldsymbol{A}\boldsymbol{Y}_{ST} \leq \lambda \|\boldsymbol{Y}_{ST}\|^2$. Together with (6) we obtain therefore

$$(|S| + |T|)\overline{d}_{ST} \leq \lambda \|\boldsymbol{Y}_{ST}\|^2 + \boldsymbol{2}\frac{|\boldsymbol{S}||\boldsymbol{T}|}{\boldsymbol{n}}\Delta \tag{7}$$

where $\overline{d}_{ST}$ is the average degree in the induced subgraph $G_{S\cup T}$. There remains the computation of $\|\boldsymbol{Y}_{ST}\|^2$. Note that $\boldsymbol{Y}_{ST}$ has $|S|$ coordinates equal to $1 - |S|/n$, $|T|$ coordinates equal to $1 - |T|/n$, $n - |S|$ coordinates equal to $-|S|/n$, and $n - |T|$ coordinates equal to $-|T|/n$. After some rearranging we obtain

$$\|\boldsymbol{Y}_{ST}\|^2 = |\boldsymbol{S}| + |\boldsymbol{T}| - \frac{|\boldsymbol{S}|^2 + |\boldsymbol{T}|^2}{\boldsymbol{n}}.$$

Together with (7) this yields Lemma 4.

References

[1] M. Sipser and D. A. Spielman, "Expander Codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1710–1722, Nov. 1996.

[2] M. Tanner, "A recursive approach to Low-complexity codes," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 533–547, Sept. 1981.

[3] A. Lubotsky, R. Philips, and P. Sarnak, "Ramanujan graphs," *Combinatorica*, vol. 8, no. 3, pp. 261–277, 1988.

[4] G. A. Margulis, "Explicit group theoretical constructions of combinatorial schemes and their application to the design of expanders and concentrators," *Probl. Inform. Transm.*, vol. 24, no. 1, pp. 39–46, 1988.

[5] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: Elsevier, 1977.

[6] N. Alon and F. R. K. Chung, "Explicit construction of linear sized tolerant networks," *Discr. Math.*, vol. 72, pp. 15–19, 1988.

Probability Propagation and Decoding in Analog VLSI

Hans-Andrea Loeliger, *Member, IEEE*, Felix Lustenberger, *Student Member, IEEE*, Markus Helfenstein, and Felix Tarköy, *Member, IEEE*

***Abstract*—The sum-product algorithm (belief/probability propagation) can be naturally mapped into analog transistor circuits. These circuits enable the construction of analog-VLSI decoders for turbo codes, low-density parity-check codes, and similar codes.**

***Index Terms*—Analog circuits, belief propagation, factor graphs, iterative decoding, turbo codes.**

I. Introduction

It has recently been observed that a number of important algorithms in error-control coding, signal processing, and computer science can be interpreted as instances of a general "sum-product algorithm" which operates by message passing on a graph (the factor graph [1], [2]; see also Aji and McEliece [3]). These algorithms include the forward–backward [Bahl–Cocke–Jelinek–Raviv (BCJR)] algorithm

Manuscript received December 16, 1999; revised September 9, 2000. The work of H.-A. Loeliger was supported by the Swiss National Science Foundation under Grant 21-49619.96. The material in this paper was presented in part at the 1998 IEEE International Symposium on Information Theory, Cambridge, MA, August 16–21, 1998, and at several other conferences.

H.-A. Loeliger was with Endora Tech AG, Basel, Switzerland. He is now with the Signal and Information Processing Laboratory (ISI), ETH Zentrum, CH-8092 Zürich, Switzerland.

F. Lustenberger is with the Signal and Information Processing Laboratory (ISI), ETH Zentrum, CH-8092 Zürich, Switzerland.

M. Helfenstein was with ISI/ETH Zurich, Switzerland. He is now with Globespan Semiconductor Inc., Red Bank, NJ 07701 USA.

F. Tarköy is with Endora Tech AG, CH-4051 Basel, Switzerland.

Communicated by R. Koetter, Guest Editor.

Publisher Item Identifier S 0018-9448(01)00718-0.