Downloaded from orbit.dtu.dk on: Mar 28, 2024



DTU Library

Graph Codes with Reed-Solomon Component Codes

Høholdt, Tom; Justesen, Jørn

Published in:
ISIT 2006

Link to article, DOI:
10.1109/ISIT.2006.261904

Publication date:
2006

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):
Høholdt, T., & Justesen, J. (2006). Graph Codes with Reed-Solomon Component Codes. In *ISIT 2006: 2006 IEEE International Symposium on Information Theory* IEEE Press. https://doi.org/10.1109/ISIT.2006.261904

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Graph Codes with Reed-Solomon Component Codes

Tom Høholdt
Department of Mathematics
Technical University of Denmark
Dk-2800, Lyngby, Denmark
Email: T.Hoeholdt@mat.dtu.dk

Jørn Justesen
COM
Technical University of Denmark
DK-2800 Lyngby, Denmark
Email: jju@com.dtu.dk

***Abstract*— We treat a specific case of codes based on bipartite expander graphs coming from finite geometries. The code symbols are associated with the branches and the symbols connected to a given node are restricted to be codewords in a Reed-Solomon code. We give results on the parameters of the codes and methods for their encoding.**

I. Introduction

We consider specific cases of the codes based on bipartite expander graphs described in [1] and [2] and based on earlier work [3]. The nodes are labeled by the points and lines of a finite geometry, and there is a branch connecting a line node to any node labeled by a point on the line. The code symbols are associated with the branches, and the symbols connected to a given node are restricted to be codewords in a Reed-Solomon (RS) code over the field that is used for constructing the geometry.

These codes can be seen as generalizations of products of RS codes (or concatenated codes with such codes as outer codes). They offer a very favorable trade-off between performance and complexity. Section II describes the properties of expander graphs derived from geometries. In particular a lower bound on the minimum distance is derived from the eigenvalues and eigenvectors of the graph. In Section III we study the rate of the codes. Section IV gives a method for encoding codes from Euclidean planes by evaluating certain polynomials. Section V describes how the code symbols can be organized into a square array to facilitate the processing. Codes from generalized quadrangles are particularly suitable for such a format. Finally Section VI contains some results on performance with iterative decoding.

II. Expander graphs from geometries

Certain bipartite graphs derived from generalized polygons have perfect expansion properties.[4]. The generalized polygons are incidence structures consisting of points and lines where any point is incident with the same number of lines and any line is incident with the same number of points. A generalized N-gon defines a bipartite graph G that satisfies the following conditions:

- For all nodes $u, v \in G$, $d(u,v) \leq N$, where $d(u,v)$ is the length of the minimum path connecting u and v.
- If $d(u,v) = h < N$, then there is a unique path of length h connecting u and v.
- Given a node $u \in G$ there exists a node $v \in G$ such that $d(u,v) = N$.

We note that this implies that the girth of the bipartite graph is at least $2N$. Most of this paper is concerned with graphs from finite planes, and in this context the 3-gons are derived from finite projective planes.

Let M be an incidence matrix for a projective plane with $m = q^2 + q + 1$ points, $(x : y : z)$, and $q^2 + q + 1$ lines of the form $ax + by + cz = 0$. The graph is invariant to an interchange of the two sets of variables.

The bipartite graph with m nodes in each set can be described by the adjacency matrix

$$A = \begin{pmatrix} 0 & M \\ M^T & 0 \end{pmatrix}$$

Thus each row has $q + 1$ 1s and the largest eigenvalue of A is $q + 1$ and the corresponding eigenvector is the all-ones vector. The graph may be seen as a simple expander graph: The eigenvalues are $\pm q + 1$ and $\pm\sqrt{q}$ (all real since A is symmetric).[4]

Starting from a node in the right set, $q+1$ nodes in the left set can be reached in one transition, and $q(q + 1)$ nodes in the right set can be reached from these nodes. The graph can be used to define a code by associating a symbol with each branch and letting all branches that meet in a node satisfy the parity checks of an (n, k, d) RS code where $n = q+1$. Thus the length of the total code is

$$N = mn = (q^2 + q + 1)(q + 1)$$

The rate of the code associated with the nodes is $r = \frac{k}{q+1}$, the total rate is bounded by

$$R \geq 2r - 1$$

The minimum distance is always lower bounded by

$$D \geq d(d(d-1)+1) = d(d^2 - d + 1)$$

Any nonzero node on the right side has at least d nonzero branches connecting to nodes in the left set, and these reach $d(d-1)$ nodes in the right set with nonzero branches. For large values of d, it follows from the expansion properties of the graph that the minimum distance is significantly larger [1].

Lemma 1: The size of subgraph with s nodes in each part where all nodes have degree at least j is

$$s \geq \frac{m(j-\lambda)}{n-\lambda}$$

1-4244-0504-1/06/$20.00 ©2006 IEEE

Authorized licensed use limited to: Danmarks Tekniske Informationscenter. Downloaded on November 28, 2009 at 16:22 from IEEE Xplore. Restrictions apply.

where λ is the second largest eigenvalue of A.

Proof: Let X be a vector with 1s in positions corresponding to nodes in the subset and 0 otherwise, and let Y be the corresponding balanced vector. Thus Y has s with value $1 - s/m$ and the rest are $-s/m$. The bound will be satisfied with equality if Y is an eigenvector with eigenvalue λ, since each node in the subset is then connected to j other nodes in the subset and $n - j$ nodes outside the subset.

In particular $j = d$ gives a lower bound on the minimum distance of the total code:[5]

$$D \geq \frac{d(q^2+q+1)(d-\lambda)}{q+1-\lambda}$$

We shall return to this property in Section VI.

It is sometimes more convenient to let M be an incidence matrix for an Euclidean plane with $m = q^2$ points, (x, y), and q^2 lines of the form $y = ax + b$. The lines of the form $x = c$ are omitted, and in this way the graph is invariant to an interchange of the two sets of variables.

Thus each row has q 1s and the eigenvalues are $\pm q$, $\pm\sqrt{q}$ and 0.

All branches that meet in a node satisfy the parity checks of an (n, k, d) RS code with $n = q$. Thus the length of the code is

$$N = q^3$$

Example 1: For $q = 16$, the projective plane gives codes of length $N = 4641$. The minimum distance is lower bounded by

$$D \geq 21d(d - 4)$$

A subgraph with 42 nodes of degree 5 can be found as a sub-plane over $q = 4$, and for this reason the lower bound is tight for $d \geq 5$.

In Section V we consider longer codes, in particular codes where the bipartite graph is derived from a generalized quadrangle.

III. The dimension of the code

The dimension of the graph code derived from a finite plane is lower bounded by

$$K \geq N - 2m(n - k)$$

since the last term is the total number of parity checks in the component codes. However, these checks are not all linearly independent. To find the actual dimension we must specify how the symbols of the component codes are mapped onto the branches. In the Euclidean plane, the node corresponding to a particular pair (a, b) connect to node (x, y) whenever $y = ax + b$. We choose the parity check matrix for the component code as

$$H = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ x_0 & x_1 & x_2 & \dots & x_{q-1} \\ {x_0}^2 & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ x_0^{q-k-1} & \dots & \dots & \dots & \dots \end{pmatrix}$$

Thus the codewords can be found by evaluating polynomials in x of degree less than k for all values of x. Since y is a linear function of x, we can also evaluate a polynomial in x and y of degree less than k in the q pairs. With this specification of the code we have:

Theorem 1: The dimension of the graph code based on a Euclidean plane over $\mathbb{F}_q$ is

$$\begin{array}{rl} k^3 & \text{for} \quad k \leq \frac{q}{2} \\ m(2k-n) + (n-k)^3 & \text{for} \quad k > \frac{q}{2} \end{array}$$

Lemma 2: The number of linearly independent monomials of degree $< k$ is k^3.

Proof: Clearly the total number of monomials of degree $< k$ in 2 variables is $\frac{k(k+1)}{2}$, and the number of monomials in 4 variables is the square of this number. However, polynomials that are equivalent modulo $y - ax - b$ have the same values on the branches of the graph. We can get a set of inequivalent monomials by not including terms that have ax as a factor. The number of monomials of degree less than k with a factor ax is simply found as the number with degree $< k - 1$. Thus the total number is

$$\frac{k^2(k+1)^2}{4} - \frac{k^2(k-1)^2}{4} = k^3$$

Proof of Theorem: We may think of the codewords as functions with arguments and values in the field. All such functions have a unique representation as polynomials of degree at most $q - 1$ in each variable. If the parity check matrices of the component codes are chosen as evaluations of low degree polynomials, the codewords are evaluations of polynomials of degree less than k. It then follows from the lemma that k^3 is a lower bound on the dimension. If two functions have the same values in the points that satisfy $y - ax - b = 0$, the difference is a multiple of this polynomial as long as the degrees are low enough. However, for polynomials of higher degrees, calculations modulo $y - ax - b$ and the identities $a^q - a = 0$ for elements in the field, give rise to equivalences between pairs of polynomials with different degrees in (x, y) and (a, b). We consider these equivalences in detail in the next section. But as long as the total degrees of the polynomials is $< q$, this situation cannot occur. In particular, if $degree(x, y) < q/2, degree(a, b) < q/2$, there is a one-to-one relation between codewords and polynomials, and thus the theorem holds for $k < q/2$. The parity check matrix of the graph code is a block matrix where the blocks are the parity check matrices for the component codes. Consider the rows of the parity check matrix corresponding to the nodes on the right. Since each block has nonzero entries only for the q positions associated with a particular node, and these $q-k$ nonzero rows are the parity check matrix of an RS code, it is clear that the $q^2(q - k)$ rows are linearly independent. Similarly the rows defining parity checks on the left are mutually independent. A vector which is spanned by both sets of rows can be described as follows: Each set of symbols corresponding to a set of branches that meet in a node on the right can be obtained by evaluating a polynomial in x of degree less than $q - k$. Similarly the set of positions corresponding to a left node can be found by evaluating a polynomial in a. Thus for these two descriptions to coincide, the total vector

Authorized licensed use limited to: Danmarks Tekniske Informationscenter. Downloaded on November 28, 2009 at 16:22 from IEEE Xplore. Restrictions apply.

has to be an evaluation of a polynomial in a and x or in (a, b) and (x, y) of degree less than $q - k$ in both sets of variables. However, the dimension of this space is given by the lemma, and the last part of the theorem follows.

The same result is true for projective planes when the component codes of length $n = q + 1$ are specified as evaluations of homogeneous polynomials. In this case the component codes can be described as evaluations where the highest degree coefficient of the information polynomial is included as an extra position (a point at infinity). Choose this point as a parity position in the component codes, and assume that the information positions of the total code is a subset of the information symbols for the component codes. When the line at infinity is deleted from the projective plane, the component codes are punctures to (q, k) RS codes, and we see that the dimension is the same in the Euclidean plane.

For longer codes the rates are closer to the simple lower bound.

IV. Encoding as evaluation

Since a graph code is described by the properties of a large parity check matrix, it is not immediately clear how encoding can be performed in a simple way [3]. Here we describe an encoding of codes from Euclidean planes based on evaluations of a suitable set of polynomials.

We represent an edge in the bipartite graph by a quadruple (x, y, a, b) in ${\mathbb{F}_q}^4$ where $y = ax + b$. A codeword is then obtained by evaluation of a polynomial from (a subset of) $\mathbb{F}_q[X, Y, A, B]$. We therefore have that polynomials which are equivalent modulo the ideal I spanned by $X^q - X, Y^q - Y, A^q - A, B^q - B, Y - AX - B$ evaluate to the same codeword and therefore we only have to consider polynomials in $V = \mathbb{F}_q[X, Y, A, B]/I$. Our first task is to find the dimension of V as a vector space over $\mathbb{F}_q$. This can be done by finding a Groebner basis of I with respect to some monomial order and then finding the leading monomials. The result is

Lemma 3: The dimension of V as a vector space over $\mathbb{F}_q$ is q^3.

Proof: For the purpose of the proof, we use the lexicographic ordering $y > x > b > a$. In this case the Groebner basis consists simply of the 5 original basis polynomials. The 'footprint' for this basis consists of all monomials in a, b, and x of degree $< q$ in each variable. Thus the dimension is q^3.

Alternative proof: The dimension of V equals the size of the "footprint" of a Groebner basis for the ideal I which is the same as the number of points in the algebraic closure of $\mathbb{F}_q$ on the variety $V(I)$. Clearly this number is q^3.

To obtain the set of polynomials that are evaluated to codewords we take a different ordering, the weighted degree ordering of the monomials with $weight(x, y) >> weight(a, b)$. In characteristic 2 we get a Groebner basis consisting of the original 5 polynomials and

$$
\begin{gathered}
ya^{q-1} + y + ba^{q-1} + b \\
y^2a^{q-2} + b^2a^{q-2} + bx + xy \\
\vdots \\
y^{q-1}a + y^{q-2}ba + \ldots + b^{q-1}a + bx^{q-2} + x^{q-2}y \\
x^{q-1}y + bx^{q-1} + y + b \\
x^{q-2}y^2 + x^{q-2}b^2 + ay + ab \\
\vdots
\end{gathered}
$$

Among the monomials in the footprint, all polynomials of degree less than k in (a, b) and (x, y) evaluate to codewords. In addition the weighted degree basis provides the monomials which have degree $< k$ in (x, y), but higher degree in (a, b). By reversing the total order to $degree(a, b) >> degree(x, y)$, we can reduce these monomials to polynomials with the lowest possible degree in (a, b) and find the space that has low degree in both representations. Thus these additional functions have two equivalent representations, one with degree $< k$ in (a, b) and another with degree $< k$ in (x, y). The procedure is illustrated in the example.

Example 2: For $q = 16$, $N = 2^{12}$, and the dimensions of the codes for $k = 1$ to 15 are

$$1, 8, 27, 64, 125, 216, 343, 512,$$
$$855, 1240, 1661, 2112, 2587, 3080, 3585.$$

Part of the basis for the code with $k = 12$ is obtained by evaluating all monomials of degree < 12 in both (x, y) and (a, b). It follows from the lemma that there $12^3 = 1728$ such monomials, and 384 additional basis functions are needed. Considering those polynomials in the Groebner basis with $degree(y) > 11$, we find

$$
\begin{gathered}
y^{12}a^4 + x^{11}y \\
= y^8b^4a^4 + y^4b^8a^4 + ba^{11} \\
y^{13}a^3 + y^{12}ba^3 + x^{12}y \\
= y^9b^4a^3 + y^8b^5a^3 + \ldots + b^{13}a^3 + bx^{12} \\
y^{14}a^2 + y^{12}b^2a^2 + x^{13}y \\
= y^{10}b^4a^2 + y^8b^6a^2 + \ldots + b^{14}a^2 + bx^{13} \\
y^{15}a + y^{14}ba + y^{13}b^2a + y^{12}b^3a + x^{14}y \\
= y^{11}b^4a + y^{10}b^5a + \ldots + b^{15}a + bx^{14}
\end{gathered}
$$

The terms on the left have degree at most 4 in (a, b), and the terms on the right degree at most 11 in (x, y). Thus we get codewords by multiplying each polynomial by a polynomial of degree at most 7 in (a, b). This gives $9 \cdot 8/2$ polynomials in each case, or at total of 144. An additional 144 are obtained by interchanging a and x. b and y. There are 72 polynomials of degree 12. However we can find more polynomials of degree 13 if we multiply the first equation by b and the second by a. When the equations are added, the highest order terms in (a, b) cancel, and we still have $degree(a, b) = 4$. If the result is multiplied by a polynomial of degree at most 6 in (a, b), the result is a linear combination of polynomials considered earlier, but the 8 terms obtained by multiplying by a degree 7 monomial are new. Thus there are a total of $8 \cdot 11 = 88$ polynomials of degree 13. The degree 14 polynomials similarly has a single term on the left with degree 4 in (a, b), and it can be cancelled by one of the lower degree polynomials. This gives $8 \cdot 13 = 104$ polynomials of degree

Authorized licensed use limited to: Danmarks Tekniske Informationscenter. Downloaded on November 28, 2009 at 16:22 from IEEE Xplore. Restrictions apply.

14. Finally we find $8 \cdot 15 = 120$ polynomials of degree 15, which completes the basis. Note that for polynomials of degree $< q/2$ this approach does not give multiple versions.

V. Codewords as square arrays

In this section we show how the graph can be used to obtain a regular organization of the code symbols as a square array. This significantly facilitates encoding and decoding, where accessing the relevant code symbols from the graph description can be a significant part of the total computational complexity. For a code protecting a large data set, it may also be an option to decode only a subset of current interest.

If the graph is derived from a Euclidean plane, we can index the array by the the values of the variables a and x. Thus a set of rows contains nodes corresponding to the parallel lines that have a particular value of a, but different values of b, and the columns are interpreted in a similar way. Each combination of a, x contains q code symbols, which can be represented by a square sub-array if q is a square (or in a third dimension). In this way the q symbols that are needed for processing a particular component code are always obtained by selecting a set of $\sqrt{q}$ rows (columns) and taking a symbol from each group of $\sqrt{q}$ columns (rows).

A graph for a longer code can be derived from a generalized quadrangle. The construction can be described in projective 3-space over $\mathbb{F}_q$ in the following way: A linear one-to-one mapping between points and planes is selected, and each such pair defines a node in the right set of the bipartite graph. The left set consist of all lines in any particular plane which pass through the corresponding point. The branches of the graph connect the lines to the planes that they are contained in (or points that they contain). There are $q^3 + q^2 + q + 1$ nodes in each set.

We can obtain a total code of length q^4 by restricting the projective space to a Euclidean 3-space. In this case we can organize the array in such a way that each row or column consists of q interleaved component codes of length q. First a special point and line in plane at infinity is chosen. The remaining plane contains q^2 points, which identify the rows of the array. Each such point is connected to q line nodes representing the interleaved component codes in that row. Similarly the q^2 lines which are not used and do not contain the special point identify the columns, and each line contains q point nodes representing the component codes. It follows from the girth of the graph that each coordinate in the array is assigned a unique symbol in this way.

Even longer codes can be constructed from generalized hexagons, but it is known that it is not possible to find larger n-gons with degree $q + 1$.

VI. Performance

Clearly it is desirable to base decoding of the graph code on the simple decoding algorithm for the component RS codes. One can also think of the codes as concatenated codes with an initial decoding step based on the (possibly redundant) binary image of the component codes.

In our discussion of iterative decoding we assume that decoding of the component RS code either corrects the errors or fails to produce a result. In the latter case, the received word is left unchanged. It is well-known that the probability of decoding when more than t errors have occurred is closely approximated by $1/t!$, and for moderate values of t these rare events have little influence on the performance. For short component codes with small t one can choose an even minimum distance to get sufficiently reliable decisions.

The error pattern can be described by a graph which is obtained from the original bipartite graph by including only branches containing errors. Iterative decoding is then described as a process of removing a node with at most t branches and any branches connecting to the node. It is well-known that the process terminates with an empty graph or with a subgraph where all nodes have degree at least $t + 1$.

It follows immediately from this description and Lemma 1 that any error pattern of weight less than $D/4$ is decoded in this way. This result is similar to the decoding of product codes by rows and columns, but it should be noted that for graph codes, the minimum distance increases linearly with the code length. If a set of nodes is not decoded, it is possible to erase the corresponding symbols and increase the number or errors that can be corrected to at least $D/2$.

However, as for standard product codes and concatenated codes, more errors are corrected in most cases. The performance of the graph codes under iteration of the decoding of the component codes can be analyzed using methods of random graphs [6]. It follows that the code will be successfully decoded with high probability even if the average number of errors in each component RS code is slightly larger than $(q - k)/2$. Thus in most cases $\frac{m(q-k)}{2}$ errors are decoded.

In the analysis of random graphs it is assumed that the branches are selected randomly. However, it is easily verified that the arguments are unchanged if the underlying structure is assumed to be a complete bipartite graph. Thus the cited result applies directly to iterative decoding of product codes. Thus it is clear that even though any remaining graphs with degree $t + 1$ with high probability are very large, there is a small probability of getting a graph with on the order t nodes. Moreover, the probability of failure does not decrease exponentially with the block length if t is fixed. The cited result can also be obtained by taking an average over bipartite graphs of fixed degree as long as the degree is significantly greater than t: Select the error positions randomly first, and then proceed to randomly select the remaining branches of the graph. Most graphs in this ensemble have good performance due to their expander properties, and the average second eigenvalue is of the order $\sqrt{q}$. Consequently most of them have minimum distances approximately satisfying the lower bound of Section II. The specific codes considered in this paper are slightly better than average in these respects, and simulations have verified that the number of errors corrected in typical cases coincides with the result of [6].

Example 3: For $q = 16$ and $k = 12$, the rate of the code from the Euclidean plane is 0.5156. The lower bound on the

Authorized licensed use limited to: Danmarks Tekniske Informationscenter. Downloaded on November 28, 2009 at 16:22 from IEEE Xplore. Restrictions apply.

minimum distance is 105, but we expect to be able to correct 512 symbol errors in most cases. For correcting binary errors we could represent each symbol as 5 bits with a parity symbol. The lower bound on the minimum distance is 210 in this case, but we expect to correct about 1000 binary errors.

REFERENCES

[1] G. Zèmor:"On expander codes" *IEEE Trans.Inform.Theory*(Special Issue on Codes on Graphs and iterative Algorithms) , vol.47, pp.835-837, Feb. 2001.

[2] A. Barg and G. Zèmor:"Error exponents of expander codes" *IEEE Trans.Inform.Theory*, vol.48, pp.1725-1729 , June 2002.

[3] M. Tanner:"A Recursive Approach to Low Complexity Codes" *IEEE Trans.Inform.Theory*, vol.27, pp.533-547, September 1981.

[4] M. Tanner:"Explicit Concentrators from Generalized N-Gons" *SIAM J.Alg.Disc.Meth.* Vol.5 No.3 pp.287-293, September 1984.

[5] M. Tanner:"Minimum-Distance Bounds by Graph Analysis" *IEEE Trans.Inform.Theory*, vol.47, pp.808-821, February 2001.

[6] B. Pittel, J. Spencer, and N. Wormald:"Sudden emergence of a giant k-core in a random graph", *J.Comb.Theory, Series B*, vol.67, pp. 11-151, 1996.

Authorized licensed use limited to: Danmarks Tekniske Informationscenter. Downloaded on November 28, 2009 at 16:22 from IEEE Xplore. Restrictions apply.