Marc Fossorier Hideki Imai
Shu Lin Alain Poli (Eds.)

Applied Algebra, Algebraic Algorithms and Error-Correcting Codes

13th International Symposium, AAECC-13
Honolulu, Hawaii, USA, November 15-19, 1999
Proceedings

Springer

Volume Editors

Marc Fossorier
Shu Lin
University of Hawaii, Department of Electrical Engineering
2540 Dole St., # 483, Honolulu, HI 96822, USA
E-mail: marc@aravis.eng.hawaii.edu; slin@spectra.eng.hawaii.edu

Hideki Imai
University of Tokyo, Institute of Industrial Science
7-22-1, Roppongi, Minato-ku, Tokyo 106, Japan
E-mail: imai@iis.u-tokyo.ac.jp

Alain Poli
Université Paul Sabatier, AAECC/IRIT
118, Route de Narbonne, F31067 Toulouse Cedex, France
E-mail: poli@cict.fr

Cataloging-in-Publication data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Applied algebra, algebraic algorithms and error-correcting codes
: 13th international symposium ; proceedings / AAECC 13, Honolulu, Hawaii, USA, November 15 - 19, 1999. Marc Fossorier . . . (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ; Paris ; Singapore ; Tokyo : Springer, 1999
(Lecture notes in computer science ; Vol. 1719)
ISBN 3-540-66723-7

CR Subject Classi cation (1998): E.4, I.1, E.3, G.2, F.2

ISSN 0302-9743
ISBN 3-540-66723-7 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera-ready by author
SPIN 10704606 06/3142 – 5 4 3 2 1 0 Printed on acid-free paper

New Sequences of Linear Time Erasure Codes Approaching the Channel Capacity

M. Amin Shokrollahi

Bell Labs, Room 2C-353, 700 Mountain Ave, Murray Hill, NJ 07974, USA
amin@research.bell-labs.com

Abstract. We will introduce a new class of erasure codes built from irregular bipartite graphs that have linear time encoding and decoding algorithms and can transmit over an erasure channel at rates arbitrarily close to the channel capacity. We also show that these codes are close to optimal with respect to the trade-off between the proximity to the channel capacity and the running time of the recovery algorithm.

1 Introduction

A linear error-correcting code of block length n and dimension k over a finite field $\mathbb{F}_q$—an $[n,k]_q$-code for short—is a k-dimensional linear subspace of the standard vector space $\mathbb{F}_q^n$. The elements of the code are called codewords. To the code C there corresponds an *encoding map* Enc which is an isomorphism of the vector spaces $\mathbb{F}_q^k$ and C. A sender, who wishes to transmit a vector of k elements in $\mathbb{F}_q$ to a receiver uses the mapping Enc to encode that vector into a codeword. The *rate* k/n of the code is a measure for the amount of real information in each codeword. The minimum distance of the code is the minimum Hamming distance between two distinct codewords. A linear code of block length n, dimension k, and minimum distance d over $\mathbb{F}_q$ is called an $[n,k,d]_q$-code.

Linear codes can be used to reliably transmit information over a noisy channel. Depending on the nature of the errors imposed on the codeword during the transmission, the receiver then applies appropriate algorithms to *decode* the received word. In this paper, we assume that the receiver knows the position of each received symbol within the stream of all encoding symbols. We adopt as our model of losses the *erasure channel*, introduced by Elias [3], in which each encoding symbol is lost with a fixed constant probability p in transit independent of all the other symbols. As was shown by Elias [3], the capacity of this channel equals $1-p$.

It is easy to see that a code of minimum distance d is capable of recovering $d-1$ or less erasures. In the best case, it can recover from any set of k coordinates of the encoding which means that $d-1=n-k$. Such codes are called MDS-codes. A standard class of MDS-codes is given by Reed-Solomon codes [10]. The connection of these codes with polynomial arithmetic allows for encoding and decoding in time $O(n\log^2 n\log\log n)$. (See, [2, Chapter 11.7] and [10, p. 369]). However, these codes do not reach the capacity of the erasure channel, since there is no infinite sequence of such codes over a fixed field.

Marc Fossorier et al. (Eds.): AAECC-13, LNCS 1719, pp. 65–76, 1999.

Elias [3] showed that a random linear code can be used to transmit over the erasure channel at any rate $R < 1 - p$, and that encoding and decoding can be accomplished with $O(n^2)$ and $O(n^3)$ arithmetic operations, respectively. Hence, we have on the one hand codes that can be encoded and decoded faster than general linear codes, but do not reach the capacity of the erasure channel; and on the other hand we have random codes which reach the capacity but have encoding and decoding algorithms of higher complexity.

The paper [1] was the first to design codes that could come arbitrarily close to the channel capacity while having linear time encoding and decoding algorithms. Improving these results, the authors of [8] took a different approach and designed fast linear-time algorithms for transmitting just below channel capacity. For all $\epsilon > 0$ they were able to produce rate $R = 1 - p(1 + \epsilon)$ codes along with decoding algorithms that could recover from the random loss of a p fraction of the transmitted symbols in time proportional to $n \ln(1/\epsilon)$ with high probability, where n is the block length. These codes could also be encoded in time proportional to $n \ln(1/\epsilon)$. They belong to the class of low-density parity check codes of Gallager [4]. In contrast to Gallager codes, however, the graphs used to construct the asymptotically good codes obtained in [8] are highly irregular.

The purpose of the present paper is twofold. First, we prove a general trade-off theorem between the proximity of a given Gallager code to the channel capacity in terms of the loss fraction and the running time of the recovery algorithm of [8]. We show that in this respect, the codes constructed in that paper are close to optimal. Next, we exhibit a different sequence of asymptotically close to optimal codes which have better parameters than the codes in [8]. An interesting feature of these codes is that the underlying bipartite graphs are *right regular*, i.e., all nodes on the right hand side of the graph have the same degree. Since they are theoretically better than their peers, we expect them to also perform better in practice.

The organization of the paper is as follows. In the next section we will review the construction of Gallager codes. Next, we prove upper bounds on the maximum tolerable loss fraction in terms of the running time of the decoding algorithm. The last two sections are concerned with the derivation of the sequence of right regular erasure codes.

2 Codes from Bipartite Graphs

In this section, we will briefly review the class of codes we are interested in, and the erasure recovery algorithm associated to them.

Our codes are similar to the Gallager codes [4] in that they are built from sparse bipartite graphs. In contrast to Gallager codes, however, our codes will be constructed from graphs that have a highly irregular degree pattern on the left.

Let G be a bipartite graph with n nodes on the left and $n - k$ nodes on the right. G gives rise to a binary code of block-length n and dimension $\geq k$ in the

following way: let the adjacency matrix of the graph G be given as

$$A = \left(\begin{array}{c|c} 0 & H^\top \\ \hline H & 0 \end{array}\right),$$

where H is some $(n-k)\times n$ matrix describing the connections in the graph. The code defined by the graph is the code with parity check matrix H. A different way of describing the code is as follows: we index the coordinate positions of the code with the n nodes on the left hand side of the graph. The code consists of all binary vectors $(c_1,\ldots,c_n)$ such that for each right node in the graph the sum of the coordinate places adjacent to it equals zero. The block-length of this code equals n, and its dimension is at least k since we are imposing $n-k$ linear conditions on the coordinates of a codeword. Expressed in terms of the graph, the fraction of redundant symbols in a codeword is at most a_L/a_R where a_L and a_R are the average node degrees on the left and the right hand side of the graph, respectively. In other words, the rate of the code is at least $1-a_L/a_R$. This description of the rate will be useful in later analysis. In the following, we will assume that the rate is in fact equal to this value. This is because the statements we will prove below will become even stronger if the rate is larger.

The above construction needs asymptotically $O(n^2)$ arithmetic operations to find the encoding of a message of length k, if the graph is sparse. One can apply a trick to reduce the running time to $O(n)$ by a modification of the construction. Details can be found in [8].

Suppose now that a codeword $(c_1,\ldots,c_n)$ is sent and that certain erasures have occurred. The erasure recovery algorithm works as follows. We first initialize the contents of the right hand nodes of G with zero. Then we collect the non-erased coordinate positions, add their value to the current value of their right neighbors, and delete the left node and all edges emanating from it from the graph. After this stage, the graph consists of the erased nodes on the left and the edges emanating from these nodes. In the next step we look for a right node in the graph of degree one, i.e., a node that has only one edge coming out of it. We transport the value of this node to its unique left neighbor ℓ, thereby recovering the value of c_ℓ. We add c_ℓ to the current value of all the right neighbors of ℓ, delete the edges emanating from ℓ, and repeat the process until we cannot find a node of degree one on the right, or until all nodes on the left have been recovered.

It is obvious that, on a RAM with unit cost measure, the amount of arithmetic operations to finish the algorithm is at most proportional to the number of edges in the graph, i.e., to na_L, where a_L is the average node degree on the left. The aim is thus to find graphs with constant a_L for which the recovery algorithm finishes successfully.

The main contribution of [8] was to give an analytic condition on the maximum fraction of tolerable losses in terms of the degree distribution of the graph. More precisely, define the *left* and the *right* degree of an *edge* in the graph as the degree of the left, resp. right node it is emanating from. Further, denote by λ_i and ρ_i the fraction of edges of left, resp. right degree i, and consider the generating functions $\lambda(x) := \sum_i \lambda_i x^{i-1}$ and $\rho(x) := \sum_i \rho_i x^{i-1}$. In the following we will call the pair (λ,ρ) a *degree distribution*.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.