Serdar Boztaş
Igor E. Shparlinski (Eds.)

LNCS 2227

Applied Algebra, Algebraic Algorithms and Error-Correcting Codes

14th International Symposium, AAECC-14
Melbourne, Australia, November 2001
Proceedings

Springer

Serdar Boztaş Igor E. Shparlinski (Eds.)

Applied Algebra, Algebraic Algorithms and Error-Correcting Codes

14th International Symposium, AAECC-14
Melbourne, Australia, November 26-30, 2001
Proceedings

Springer

Volume Editors

Serdar Boztaş
RMIT University, Department of Mathematics
GPO Box 2476V, Melbourne 3001, Australia
E-mail: serdar@rmit.edu.au

Igor E. Shparlinski
Macquarie University, Department of Computing
NSW 2109, Australia
E-mail: igor@comp.mq.edu.au

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Applied algebra, algebraic algorithms and error correcting codes :
14th international symposium ; proceedings / AAECC 14, Melbourne,
Australia, November 26 - 30, 2001. Serdar Boztaş ; Igor E. Shparlinski (ed.). -
Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ;
Paris ; Tokyo : Springer, 2002
(Lecture notes in computer science ; Vol. 2227)
ISBN 3-540-42911-5

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

ISSN 0302-9743
ISBN 3-540-42911-5 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

http://www.springer.de

Typesetting: Camera-ready by author, data conversion by Steingräber Satztechnik GmbH, Heidelberg
Printed on acid-free paper SPIN: 10840981 06/3142 5 4 3 2 1 0

Preface

The AAECC Symposia Series was started in 1983 by Alain Poli (Toulouse), who, together with R. Desq, D. Lazard, and P. Camion, organized the first conference. Originally the acronym AAECC meant "Applied Algebra and Error-Correcting Codes". Over the years its meaning has shifted to "Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes", reflecting the growing importance of complexity in both decoding algorithms and computational algebra.

AAECC aims to encourage cross-fertilization between algebraic methods and their applications in computing and communications. The algebraic orientation is towards finite fields, complexity, polynomials, and graphs. The applications orientation is towards both theoretical and practical error-correction coding, and, since AAECC 13 (Hawaii, 1999), towards cryptography. AAECC was the first symposium with papers connecting Gröbner bases with E-C codes. The balance between theoretical and practical is intended to shift regularly; at AAECC-14 the focus was on the theoretical side.

The main subjects covered were:

- Codes: iterative decoding, decoding methods, block codes, code construction.
- Codes and algebra: algebraic curves, Gröbner bases, and AG codes.
- Algebra: rings and fields, polynomials.
- Codes and combinatorics: graphs and matrices, designs, arithmetic.
- Cryptography.
- Computational algebra: algebraic algorithms.
- Sequences for communications.

Six invited speakers covered the areas outlined:

- Robert Calderbank, "Combinatorics, Quantum Computers, and Cellular Phones"
- James Massey, "The Ubiquity of Reed-Muller Codes"
- Graham Norton, "Gröbner Bases over a Principal Ideal Ring"
- Vera Pless, "Self-dual Codes – Theme and Variations"
- Amin Shokrollahi, "Design of Differential Space-Time Codes Using Group Theory"
- Madhu Sudan, "Ideal Error-Correcting Codes: Unifying Algebraic and Number-Theoretic Algorithms".

Except for AAECC-1 (*Discrete Mathematics* 56, 1985) and AAECC-7 (*Discrete Applied Mathematics* 33, 1991), the proceedings of all the symposia have been published in Springer-Verlag's *Lecture Notes in Computer Science* (Vols. 228, 229, 307, 356, 357, 508, 539, 673, 948, 1255, 1719).

It is a policy of AAECC to maintain a high scientific standard, comparable to that of a journal. This has been made possible thanks to the many referees involved. Each submitted paper was evaluated by at least two international researchers.

AAECC-14 received and refereed 61 submissions. Of these, 1 was withdrawn, 36 were selected for publication in these proceedings, while 7 additional works contributed to the symposium as oral presentations. Unrefereed talks were presented in a "Recent Results" session.

The symposium was organized by Serdar Boztaş, Tom Høholdt, Kathy Horadam, Igor E. Shparlinski, and Branka Vucetic, with the help of Asha Baliga, Pride Conference Management (Juliann Smith), and the Department of Mathematics, RMIT University. It was sponsored by the Australian Mathematical Society.

We express our thanks to the staff of Springer-Verlag, especially Alfred Hofmann and Anna Kramer, for their help in the preparation of these proceedings.

August 2001

Serdar Boztaş and Igor E. Shparlinski

Organization

Steering Committee

General Chair:	Kathy Horadam (RMIT Univ., AUS)
Conference Co-chair:	Tom Høholdt (Technical Univ. of Denmark, DK)
Program Chair:	Igor Shparlinski (Macquarie Univ., AUS)
Program Co-chair:	Branka Vucetic (Sydney Univ., AUS)
Publication:	Serdar Boztaş (RMIT Univ., AUS)

Conference Committee

J. Calmet
M. Clausen
G. Cohen
P.G. Farrell
G.L. Feng
M. Giusti
J. Heintz
T. Høholdt
K. Horadam
H. Imai
H. Janwa
J.M. Jensen
R. Kohno
H.W. Lenstra Jr.
S. Lin
O. Moreno
H. Niederreiter
A. Poli
T.R.N. Rao
S. Sakata
P. Solé

Program Committee

I.F. Blake
J. Calmet
C. Carlet
P. Charpin
M. Clausen
P.G. Farrell
M. Fossorier
M. Giesbrecht
M. Giusti
J. Gutierrez
J. Heintz
T. Helleseth
H. Imai
E. Kaltofen
T. Kasami
L. Knudsen
S. Litsyn
A. Nechaev
H. Niederreiter
D. Panario
S. Sakata
P. Solé
H. van Tilborg
C. Xing

Local Organizing Committee

Asha Baliga
Serdar Boztaş
Kathy Horadam

Referees

D. Augot
A. Baliga
I.F. Blake
A. Bonnecaze
N. Boston
F. Boulier
S. Boztaş
J. Calmet
C. Carlet
P. Charpin
M. Clausen
G. Cohen

R. Cramer
I. Damgård
M. Dichtl
C. Ding
I. Duursma
P.G. Farrell
G-L. Feng
H.C. Ferreira
M. Fossorier
T. Fujiwara
P. Gaborit
J. Galati
S. Galbraith
S. Gao
V.P. Gerdt
M. Giesbrecht
M. Giusti
F. Griffin
J. Gutierrez
Y.S. Han
C. Hao
T. Hashimoto
J. Heintz
T. Helleseth
K. Horadam
X-D. Hou
H. Imai
J. Jensen
G. Kabatiansky
E. Kaltofen
T. Kasami
F. Keqin
T. Kløve
L. Knudsen
L. Kulesz
T. Laihonen
S. Ling
S. Litsyn
F. Morain
R. Morelos-Zaragoza
S. Murphy
V.K. Murty
A. Nechaev
H. Niederreiter
D. Panario
L. Pecquet
V. Rijmen
S. Sakata
P. Sarkar
H.G. Schaathun
I. Shparlinski
B. Shung
A. Silverberg
P. Solé
B. Stevens
H. van Tilborg
B. Vucetic
J.L. Walker
K. Yang
C. Xing

Sponsoring Institutions

Australian Mathematical Society

Table of Contents

Invited Contributions

Block Codes

Code Constructions

Codes and Algebra: Rings and Fields

Codes and Algebra: Algebraic Geometry Codes

Sequences

Cryptography

Algorithms

Algorithms: Decoding

Algebraic Constructions

The Ubiquity of Reed-Muller Codes

James L. Massey

ETH-Zürich and Lund University
Trondhjemsgade 3 2TH, DK-2100 Copenhagen East
JamesMassey@compuserve.com

Abstract. It is argued that the nearly fifty-year-old Reed-Muller codes underlie a surprisingly large number of algebraic problems in coding and cryptography. This thesis is supported by examples that include some new results such as the construction of a new class of constant-weight cyclic codes with a remarkably simple decoding algorithm and a much simplified derivation of the well-known upper bound on the linear complexity of the running key produced by a nonlinearly filtered maximal-length shift-register.

1 Introduction

The Reed-Muller codes, which were actually discovered by Muller [1], were the first nontrivial class of multiple-error-correcting codes. Reed [2] gave a simple majority-logic decoding algorithm for these binary codes that corrects all errors guaranteed correctable by their minimum distance; he also gave an insightful description of these codes that has been adopted by most later researchers and that we will also follow here.

Nearly 50 years have passed since the discovery of the Reed-Muller codes. It is our belief that when one digs deeply into almost any algebraic problem in coding theory or cryptography, one finds these venerable codes (or closely related codes) lying at the bottom. We illustrate this "ubiquity" of the Reed-Muller codes in what follows with a number of examples that include some new results.

In Section 2, we describe the two matrices whose properties underlie the construction and theory of the Reed-Muller codes. The codes themselves are introduced in Section 3. In Section 4 we show how the Reed-Muller codes have been used in a natural way to measure the nonlinearity of a binary function of m binary variables, a problem that arises frequently in cryptography. In Section 5 we use Reed-Muller coding concepts to construct a new class of constant-weight cyclic codes that have an astonishingly simple decoding algorithm. The cyclic Reed-Muller codes are introduced in Section 6 where we also describe an "unconventional" encoder for these codes. This encoder is seen in Section 7 to be the same as the running-key generator for a stream cipher of the type called a nonlinearly filtered maximal-length shift register, which leads to an extremely simple derivation of a well-known upper bound on the linear complexity of the resulting running key. We conclude with some remarks in Section 8.

S. Boztaş and I.E. Shparlinski (Eds.): AAECC-14, LNCS 2227, pp. 1–12, 2001.

2 Two Useful Matrices

In this section we describe two matrices whose properties will be exploited in the sequel.

Let $\mathbf{M}_m$ denote the $2^m \times 2^m$ binary matrix in which the entries in row $i+1$ are the coefficients of $(1+x)^i$ in order of ascending powers of x for $i = 0, 1, 2, \ldots, 2^m - 1$. For $m = 3$, this matrix is

$$\mathbf{M}_3 = \begin{bmatrix} 1\,0\,0\,0\,0\,0\,0\,0 \\ 1\,1\,0\,0\,0\,0\,0\,0 \\ 1\,0\,1\,0\,0\,0\,0\,0 \\ 1\,1\,1\,1\,0\,0\,0\,0 \\ 1\,0\,0\,0\,1\,0\,0\,0 \\ 1\,1\,0\,0\,1\,1\,0\,0 \\ 1\,0\,1\,0\,1\,0\,1\,0 \\ 1\,1\,1\,1\,1\,1\,1\,1 \end{bmatrix} .$$

Some Properties of $\mathbf{M}_m$:

1. The i-th row of $\mathbf{M}_m$ is the i-th row of Pascal's triangle with entries reduced modulo 2. Equivalently, each row after the first is obtained by adding the previous row to its own shift right by one position.
2. The Hamming weight of row $i+1$, *i.e.*, the number of nonzero coefficients in $(1+x)^i$, is equal to the Hamming weight $W_2(i)$ of the radix-two representation of the integer i for $i = 0, 1, 2, \ldots, 2^m - 1$, cf. Lemma 1 in [3].
3. The matrix $\mathbf{M}_m$ is its own inverse, cf. [4].
4. The sum of any selection of rows of the matrix $\mathbf{M}_m$ has Hamming weight at least that of the uppermost row included in the sum, cf. Theorem 1.1 in [3].

Of special interest to us here will be the submatrix $\mathbf{A}_m$ of $\mathbf{M}_m$ consisting of the m rows with Hamming weight 2^{m-1}. For $m = 3$, this matrix is

$$\mathbf{A}_3 = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} = \begin{bmatrix} 1\,1\,1\,1\,0\,0\,0\,0 \\ 1\,1\,0\,0\,1\,1\,0\,0 \\ 1\,0\,1\,0\,1\,0\,1\,0 \end{bmatrix}$$

where here and hereafter we denote the rows of $\mathbf{A}_m$ as $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m$.

Some Properties of $\mathbf{A}_m$:

1. The j-th column of $\mathbf{A}_m$, when read downwards with its entries considered as integers, contains the radix-two representation of the integer $2^m - j$ for $j = 1, 2, \ldots, 2^m$.
2. The i^{th} row $\mathbf{a}_i$ of $\mathbf{A}_m$, when treated as the function table of a binary-valued function of m binary variables in the manner that the entry in the j^{th} column is the value of the function $f(x_1, x_2, \ldots, x_m)$ when $x_1, x_2, \ldots, x_m$ considered as integers is the radix-two representation of the integer $2^m - j$, corresponds to the function $f(x_1, x_2, \ldots, x_m) = x_i$ for $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, 2^m$.
3. Cyclic shifting the rows of $\mathbf{A}_m$ in any way (*i.e.*, allowing different numbers of shifts for each row) is equivalent to a permutation of the columns of $\mathbf{A}_m$.

4. Every $m \times 2^m$ binary matrix whose columns are all different can be obtained from $\mathbf{A}_m$ by a permutation of the columns.

3 Reed-Muller Codes

Following Reed's notation [2] for the Reed-Muller codes, we use juxtaposition of row vectors to denote their term-by-term product, which we will refer to as the *Hadamard product* of these row vectors. For instance, for $m = 3$, $\mathbf{a}_1\mathbf{a}_3 = \begin{bmatrix} 1\,0\,1\,0\,0\,0\,0\,0 \end{bmatrix}$ and $\mathbf{a}_1\mathbf{a}_2\mathbf{a}_3 = \begin{bmatrix} 1\,0\,0\,0\,0\,0\,0\,0 \end{bmatrix}$. We also write $\mathbf{a}_0$ to denote the all-one row vector of length 2^m.

Let RM(m, μ), where $1 \leq \mu < m$, denote the μ^{th}-order Reed-Muller code of length $n = 2^m$. RM(m, μ) can be defined as the linear binary code for which the matrix $\mathbf{G}_m^\mu$, which has as rows $\mathbf{a}_0$, $\mathbf{a}_1$, $\ldots$ $\mathbf{a}_m$ together with all Hadamard products of $\mathbf{a}_1$, $\mathbf{a}_2$, $\ldots$ $\mathbf{a}_m$ taken μ or fewer at a time, is a generator matrix. For instance, the second-order Reed-Muller code RM$(3, 2)$ has the generator matrix

$$\mathbf{G}_3^2 = \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \\ \mathbf{a}_1\mathbf{a}_2 \\ \mathbf{a}_1\mathbf{a}_3 \\ \mathbf{a}_2\mathbf{a}_3 \end{bmatrix} = \begin{bmatrix} 1\,1\,1\,1\,1\,1\,1\,1 \\ 1\,1\,1\,1\,0\,0\,0\,0 \\ 1\,1\,0\,0\,1\,1\,0\,0 \\ 1\,0\,1\,0\,1\,0\,1\,0 \\ 1\,1\,0\,0\,0\,0\,0\,0 \\ 1\,0\,1\,0\,0\,0\,0\,0 \\ 1\,0\,0\,0\,1\,0\,0\,0 \end{bmatrix}.$$

It is also convenient to define the 0^{th}-order Reed-Muller code RM$(m, 0)$ as the binary linear code with generator matrix $\mathbf{G}_m^0 = [\mathbf{a}_0]$. For instance for $m = 3$,

$$\mathbf{G}_3^0 = \begin{bmatrix} 1\,1\,1\,1\,1\,1\,1\,1 \end{bmatrix}.$$

The following proposition is a direct consequence of Properties 3 and 4 of the matrix $\mathbf{M}_m$.

Proposition 1. *The Reed-Muller code RM(m, μ) of length $n = 2^m$, where $0 \leq \mu < m$, has dimension $k = \sum_{i=0}^{\mu} \binom{m}{i}$ and minimum distance $d = 2^{m-\mu}$. Moreover, its dual code is the Reed-Muller code RM$(m, m - 1 - \mu)$.*

4 Measuring Nonlinearity

It is often the case in cryptography that one wishes to find a binary-valued function $f(x_1, x_2, \ldots, x_m)$ of m binary variables that is "highly nonlinear". Rueppel [5] showed that the Reed-Muller codes can be used to measure the amount of nonlinearity in a very natural way. His approach is based on the following proposition, which is an immediate consequence of Property 2 of the matrix $\mathbf{A}_m$ and of the facts that

$$\mathbf{G}_m^1 = \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{A}_m \end{bmatrix}$$

and that $\mathbf{a}_0$ is the function table of the constant function 1.

Proposition 2. *The codewords in the first-order Reed-Muller code of length 2^m, RM(m, 1), correspond to the function tables of all linear and affine functions of m binary variables when the entry in the j^{th} position is considered as the value of the function $f(x_1, x_2, \ldots, x_m)$ where $x_1, x_2, \ldots, x_m$ give the radix-two representation of the integer $2^m - j$ for $j = 1, 2, \ldots, 2^m$.*

Rueppel, cf. pp. 127–129 in [5], exploited the content of Proposition 2 to assert that the best linear or affine approximation to a binary function $f(x_1, x_2, \ldots, x_m)$ with function table $\mathbf{y} = \begin{bmatrix} y_1 \; y_2 \; \ldots \; y_{2^m} \end{bmatrix}$ has as its function table the codeword in RM(m, 1) closest (in the Hamming metric) to $\mathbf{y}$. If e is the number of errors in this best approximation, *i.e.*, the Hamming distance from this closest codeword to $\mathbf{y}$, then $e/2^m$ is the error rate of this best linear or affine approximation to $f(x_1, x_2, \ldots, x_m)$.

Sometimes in cryptography one knows only that the function to be approximated is one of a set of t functions. In this case, Rueppel suggested taking the best linear or affine approximation to be the function corresponding to the codeword in RM(m, 1) at the smallest average Hamming distance to the function tables $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_t$ and to use the smallness of the average error rate as the measure of goodness. As an example of this method, Rueppel showed that the best linear or affine approximation to the most significant input bit of "S-box" S_5 of the Data Encryption Standard (DES) [6] from the four different output functions $f(x_1, x_2, \ldots, x_{16})$ determined by the two "control bits" for this S-Box is the affine function $1 + x_1 + x_2 + x_3 + x_4$ and has an error rate of only 12/64 or 18.8%. It is hardly surprising that, seven years later, Matsui [7] built his "linear cryptanalysis" attack against DES on this "linear weakness" in S-box S_5.

5 Easily Decodable Constant-Weight Cyclic Codes

There are many ways to combine binary vectors to obtain another binary vector in addition to summing and to taking their Hadamard product. One of the most interesting ways when the number of vectors is odd is by *majority combining* in each bit position. For instance, majority combining of the three rows in

$$\mathbf{A}_3 = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} = \begin{bmatrix} 1\,1\,1\,1\,0\,0\,0\,0 \\ 1\,1\,0\,0\,1\,1\,0\,0 \\ 1\,0\,1\,0\,1\,0\,1\,0 \end{bmatrix}$$

gives the row vector

$$\mathbf{v}_3 = \begin{bmatrix} 1\,1\,1\,0\,1\,0\,0\,0 \end{bmatrix}.$$

The sequence $\mathbf{v}_m$ obtained by majority combining the rows of $\mathbf{A}_m$ was introduced by Stiffler [8] as one period of a periodic "ranging sequence" with the property that, when corrupted by additive noise, it could be synchronized by serial processing with a single correlator much faster than could any previously proposed ranging sequence of the same period. We adopt a coding viewpoint here and, for odd m at least 3, take $\mathbf{v}_m$ and its $2^m - 1$ cyclic shifts to be the codewords in a binary cyclic constant–weight code, which we denote by S_m and call a *Stiffler code*.

Proposition 3. *For every odd m at least 3, the Stiffler code S_m is a cyclic constant-weight binary code with length $n = 2^m$ having n codewords of weight $w = 2^{m-1}$ and minimum distance $d = 2\binom{m-1}{(m-1)/2}$.*

For instance, the $n = 8$ codewords

$$[1\,1\,1\,0\,1\,0\,0\,0], [0\,1\,1\,1\,0\,1\,0\,0], [0\,0\,1\,1\,1\,0\,1\,0], [0\,0\,0\,1\,1\,1\,0\,1],$$
$$[1\,0\,0\,0\,1\,1\,1\,0], [0\,1\,0\,0\,0\,1\,1\,1], [1\,0\,1\,0\,0\,0\,1\,1], [1\,1\,0\,1\,0\,0\,0\,1]$$

in S_3 have weight $w = 4$ and are easily checked to have minimum distance $d = 2\binom{2}{1} = 4$. Because the codewords in S_m form a single cyclic equivalence class, the code has a well-defined distance distribution. The distance distribution for S_3 is $D_0 = 1$, $D_4 = 5$ and $D_6 = 2$ where D_i is the number of codewords at distance i from a fixed codeword.

Before proving Proposition 3, it behooves us to say why the Stiffler codes are interesting. From a distance viewpoint, they are certainly much inferior to the first-order Reed-Muller code RM$(m, 1)$ which have $n = 2^m$, dimension $k = m+1$ (and thus $2n$ codewords), and minimum distance $d = 2^{m-1}$. The saving grace of the Stiffler codes is that they can be decoded up to their minimum distance much more simply than even the first-order Reed-Muller codes.

To prove Proposition 3, we first note that row $\mathbf{a}_1$ of $\mathbf{A}_m$ affects the majority combining that produces $\mathbf{v}_m$ only in those $2\binom{m-1}{(m-1)/2}$ columns where the remaining $m-1$ rows of $\mathbf{A}_m$ contain an equal number of zeroes and ones. Complementing row $\mathbf{a}_1$ of $\mathbf{A}_m$ and then majority combining with the remaining rows would thus produce a new row vector at distance $2\binom{m-1}{(m-1)/2}$ from $\mathbf{v}_m$–but this complementing of the first row of $\mathbf{A}_m$ without changing the remaining rows is equivalent to cyclic shifting *all* rows of $\mathbf{A}_m$ by 2^{m-1} positions so that this new row vector is the cyclic shift of $\mathbf{v}_m$ by 2^{m-1} positions and is thus also a codeword in S_m. It follows that the minimum distance of S_m cannot exceed $2\binom{m-1}{(m-1)/2}$.

We complete the proof of Proposition 3 by showing that the following decoding algorithm for S_m corrects all patterns of $\binom{m-1}{(m-1)/2} - 1$ or fewer errors and either corrects or detects every pattern of $\binom{m-1}{(m-1)/2}$ errors, which implies that the minimum distance cannot be less than $2\binom{m-1}{(m-1)/2}$. We first note, however, that every row of $\mathbf{A}_m$, say row $\mathbf{a}_i$, agrees with $\mathbf{v}_m$ in exactly $2^{m-1} + \binom{m-1}{(m-1)/2}$ positions, *i.e.*, in all $2\binom{m-1}{(m-1)/2}$ positions where $\mathbf{a}_i$ affects the majority combining and in exactly half of the remaining $2^m - 2\binom{m-1}{(m-1)/2}$ positions. We note also that by the *decimation by 2* of a vector of even length, say $\mathbf{r} = [r_1\, r_2\, r_3\, r_4 \ldots r_{2L-1}\, r_{2L}]$, is meant the vector $[r_1\, r_3 \ldots r_{2L-1}\, r_2\, r_4 \ldots r_{2L}]$ whose two subvectors $[r_1\, r_3 \ldots r_{2L-1}]$ and $[r_2\, r_4 \ldots r_{2L}]$ are called the *phases* of this decimation by 2 of $\mathbf{r}$.

Decoding algorithm for S_m:
Let $\mathbf{r} = \left[r_1 \, r_2 \ldots r_{2^m}\right]$ be the binary received vector.
Step 0: Set $i = m$ and $\tilde{\mathbf{r}} = \mathbf{r}$.
Step 1: If the Hamming distance from $\mathbf{a}_m = \left[1\,0\,1\,0 \ldots 1\,0\right]$ to $\tilde{\mathbf{r}}$ is less than $n/2 = 2^{m-1}$ or greater than $n/2 = 2^{m-1}$, set δ_i to 0 or 1, respectively. If this distance is equal to $n/2 = 2^{m-1}$, announce a detected error and stop.
Step 2: If $i = 1$, stop and announce the decoding decision as the right cyclic shift of $\mathbf{v}_m$ by $\delta = \delta_1 2^{m-1} + \delta_2 2^{m-2} + \ldots + \delta_m$ positions.
Step 3: If $\delta_i = 1$, shift $\tilde{\mathbf{r}}$ cyclically left by one position.
Step 4: Replace $\tilde{\mathbf{r}}$ by its decimation by 2, decrease i by 1, then return to Step 1.

Example of Decoding for S_3 :
Suppose that $\mathbf{r} = \left[1\,1\,0\,0\,0\,1\,1\,1\right]$ is the received vector.

We begin by setting $i = 3$ and $\tilde{\mathbf{r}} = \left[1\,1\,0\,0\,0\,1\,1\,1\right]$.

Because the Hamming distance from $\mathbf{a}_3 = \left[1\,0\,1\,0\,1\,0\,1\,0\right]$ to $\tilde{\mathbf{r}}$ is 5, which exceeds $n/2 = 4$, we set $\delta_3 = 1$ and then shift $\tilde{\mathbf{r}}$ cyclically to the left by one position to obtain $\tilde{\mathbf{r}} = \left[1\,0\,0\,0\,1\,1\,1\,1\right]$. We then decimate $\tilde{\mathbf{r}}$ by 2 to obtain $\tilde{\mathbf{r}} = \left[1\,0\,1\,1\,0\,0\,1\,1\right]$, after which we decrease i to 2.

Because the Hamming distance from $\mathbf{a}_3$ to $\tilde{\mathbf{r}}$ is 3, which is less than $n/2 = 4$, we set $\delta_2 = 0$. We then decimate $\tilde{\mathbf{r}}$ by 2 to obtain $\tilde{\mathbf{r}} = \left[1\,1\,0\,1\,0\,1\,0\,1\right]$, after which we decrease i to 1.

Because the Hamming distance from $\mathbf{a}_3$ to $\tilde{\mathbf{r}}$ is 7, which exceeds $n/2 = 4$, we set $\delta_1 = 1$.

We now announce the decoding decision as the right shift of $\mathbf{v}_m$ by $\delta = 4\delta_1 + 2\delta_2 + \delta_3 = 5$ positions, *i.e.*, as the codeword $\left[0\,1\,0\,0\,0\,1\,1\,1\right]$, which we note is at Hamming distance 1 from the received word so that we have corrected an apparent single error.

To justify this decoding algorithm, which is an adaptation to the decoding problem for S_m of the algorithm given by Stiffler [8] for synchronization of the periodic ranging sequence with pattern $\mathbf{v}_m$ within one period, we argue as follows:

Suppose that the transmitted codeword is the right cyclic shift of $\mathbf{v}_m$ by an *even* number of bit positions. Because $\mathbf{a}_m$ is unchanged by a right cyclic shift by an *even* number of bit positions, $\mathbf{a}_m$ will agree with the transmitted codeword in the same number of bit positions as it agrees with $\mathbf{v}_m$, *i.e.*, in $2^{m-1} + \binom{m-1}{(m-1)/2}$ bit positions. Thus, if $\binom{m-1}{(m-1)/2} - 1$ or fewer errors occur, $\mathbf{a}_m$ will agree with the transmitted codeword in more than $n/2 = 2^{m-1}$ positions–and in at least $n/2 = 2^{m-1}$ positions if exactly $\binom{m-1}{(m-1)/2}$ errors occur. Suppose conversely that the transmitted codeword is the right cyclic shift of $\mathbf{v}_m$ by an *odd* number of bit positions. Because $\mathbf{a}_m$ is complemented by a right cyclic shift by an *odd* number of bit positions, $\mathbf{a}_m$ will disagree with the transmitted codeword in the same number of bit positions as it agrees with $\mathbf{v}_m$, *i.e.*, in $2^{m-1} + \binom{m-1}{(m-1)/2}$ bit positions. Thus, if $\binom{m-1}{(m-1)/2} - 1$ or fewer errors occur, $\mathbf{a}_m$ will disagree with the transmitted codeword in more than $n/2 = 2^{m-1}$ positions–and in at least $n/2 = 2^{m-1}$ positions if exactly $\binom{m-1}{(m-1)/2}$ errors occur. It follows that the value