Algebraic Soft- and Hard-Decision Decoding of Generalized Reed–Solomon and Cyclic Codes

Alexander Zeh

▶ To cite this version:

Alexander Zeh. Algebraic Soft- and Hard-Decision Decoding of Generalized Reed–Solomon and Cyclic Codes. Computational Complexity [cs.CC]. Ecole Polytechnique X, 2013. English. NNT : . pastel-00866134

HAL Id: pastel-00866134

https://pastel.hal.science/pastel-00866134

Submitted on 11 Oct 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ulm university universität uulm

ÉCOLE POLYTECHNIQUE ParisTech

Algebraic Soft- and Hard-Decision Decoding of Generalized Reed–Solomon and Cyclic Codes

Alexander Zeh

alex@codingtheory.eu

DISSERTATION

Joint Doctoral Supervision between

Fakultät für Ingenieurwissenschaften
und Informatik der Universität Ulm
zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
Acting Dean: Prof. Dr.-Ing. Klaus Dietmayer

École Polytechnique
ParisTech
pour obtenir le grade de
Docteur de l'École Polytechnique
Spécialité: Mathématiques - Informatique

Supervisor:	Prof. Dr.-Ing. Martin Bossert, Universität Ulm, Ulm, Germany
Supervisor:	Prof. Dr. Daniel Augot, École Polytechnique, Paris, France
Reporter:	Prof. Dr. Margreta Kuijper, University of Melbourne, Melbourne, Australia
Reporter:	Prof. Dr. Pascale Charpin, INRIA Rocquencourt, Paris, France
Examiner:	Prof. Dr.-Ing. Wolfgang Menzel, Universität Ulm, Ulm, Germany
Examiner:	Prof. Dr. Jean-Pierre Tillich, INRIA Rocquencourt, Paris, France
Examiner:	Dr. Clément Pernet, Université Joseph Fourier, Grenoble, France

Date of Submission:	June 7, 2013
Date of Defense:	September 2, 2013

Für die größte Errungenschaft meiner Dissertationszeit.

Acknowledgments

This bi-nationally supervised doctoral dissertation contains parts of my work obtained as a PhD student at the Institute of Communications Engineering (former Institute of Telecommunications and Applied Information Theory) at Ulm University, Germany and in the Team Grace of INRIA–Saclay-Île-de-France and École Polytechnique, Paris-Palaiseau, France from November 2008 until September 2013.

I am very grateful to my two doctoral advisers Martin Bossert (Institute of Communications Engineering, Ulm University) and Daniel Augot (Team Grace, INRIA–Saclay-Île-de-France and École Polytechnique).

Martin's organizational genius and his sensibility for potential research directions and researchers turn the institute at Ulm University into a lighthouse of science on an international level. His spirit and fascination for teaching and science, led to my decision to come to Ulm after finishing my Master's thesis under the supervision of Daniel Augot.

Daniel is a fascinating researcher and an enthusiastic scientist. He is the source of various—sometimes crazy—ideas. During scientific discussions Daniel is very precise; thanks for rapping my knuckles when I was unclear!

I would like to thank the reviewers of my thesis Margreta Kuijper (Department of Electrical & Electronic Engineering, The University of Melbourne) and Pascale Charpin (Team Secret, INRIA–Paris-Rocquencourt) for their interest in my work, the constructive feedback and the discussions. Moreover, I would like to thank my three examiners Wolfgang Menzel (Institute of Microwave Techniques, Ulm University), Jean-Pierre Tillich (Team Secret, INRIA–Paris-Rocquencourt) and Clément Pernet (Université Joseph Fourier, Grenoble and Team AriC, INRIA-Grenoble) for reading my thesis, their feedback and for coming to Ulm for my defense.

I am very thankful for the pleasant discussions with Vladimir R. Sidorenko. They have led to various new insights; not only in coding theory, but also for the anti-gravitation problem.

Next, I want to mention those of my co-authors who were not included in the above acknowledgments. I am very thankful for the lively discussions—at the end of the thesis mostly via video-telephony—on coding problems with Sergey V. Bezzateev from St. Petersburg State University for Aerospace Instrumentation. I would like to thank Maximilien Gadouleau (School of Engineering & Computing Sciences, Durham University), Johan S. R. Nielsen (Technical University of Denmark, DTU), Christian Gentner (German Aerospace Center, DLR), and my co-authors from Ulm: Sabine Kampf, Wenhui Li, Steffen Schober as well as Christian Senger with whom I shared the "President Room".

I am especially beholden to Iwan Duursma for indicating the link to cyclic product codes at ISIT 2012.

Moreover, I want to thank my colleagues for all the discussions in mathematics, (bio-)information theory, communications engineering, cryptography and for the nice coffee breaks. At Ulm University: Carolin Huppert, Katharina Mir, Johannes Klotz, Frederic Knabe, David Kracht, Mostafa Hosni Mohamed and Henning Zörlein. In Paris: Cécile Gonçalves, Morgan Barbier, Alain Couvreur, Guillaume Quintin and Ben Smith. Thanks also to all (former) members of the Secret team at INRIA–Paris-Rocquencourt and to Alexey Frolov, Konstantin Kondrashov and Pavel Rybin of the Institute for Information Transmission Problems (IPPI) in Moscow.

During the period of this dissertation, I supervised (or co-supervised) several Master's and Bachelor's

theses. It was a pleasure to work with such open-minded and nice persons as (chronologically ordered) Christian Gentner, Uwe Kraft, Wenhui Li, Fan Yu, Markus Stinner, Markus Ulmschneider, Youssef Hashad, Harris Hafeez, Salma Hesham Badredlin, Georg Fischer and Thomas Jerkovits.

Finally, I want to thank the German Research Council (Deutsche Forschungsgemeinschaft, DFG), the German Academic Exchange Service (Deutscher Akademischer Austauschdienst, DAAD), the German-French University (Deutsch-Französische Hochschule, DFH) and the French Institute for Research in Computer Science and Control (Institut National de Recherche en Informatique et en Automatique, INRIA) for their financial support.

Alexander Zeh, Ulm, Germany,
September 2013

Abstract and Résumé

TWO challenges in algebraic coding theory are addressed within this dissertation. The first one is the efficient hard- and soft-decision decoding of Generalized Reed–Solomon codes over finite fields in Hamming metric. The motivation for this more than 50 years old problem was renewed by the discovery of a polynomial-time interpolation-based decoding principle up to the Johnson radius by Guruswami and Sudan at the end of the 20th century. First syndrome-based error/erasure decoding approaches by Berlekamp–Massey and Sugiyama–Kasahara–Hirasawa–Namekawa for Generalized Reed–Solomon codes were described by a Key Equation, i.e., a polynomial description of the decoding problem. The reformulation of the interpolation-based approach in terms of Key Equations is a central topic of this thesis. This contribution covers several aspects of Key Equations for Generalized Reed–Solomon codes for both, the hard-decision variant by Guruswami–Sudan, as well as for the soft-decision approach by Kötter–Vardy. The obtained systems of linear homogeneous equations are structured and efficient decoding algorithms are developed.

The second topic of this dissertation is the formulation and the decoding up to lower bounds on the minimum Hamming distance of linear cyclic block codes over finite fields. The main idea is the embedding of a given cyclic code into a cyclic (generalized) product code. Therefore, we give an extensive description of cyclic product codes and code concatenation. We introduce cyclic generalized product codes and indicate how they can be used to bound the minimum distance. Necessary and sufficient conditions for lowest-rate non-primitive binary cyclic codes of minimum distance two and a sufficient condition for binary cyclic codes of minimum distance three are worked out and their relevance for the embedding-technique is outlined. Furthermore, we give quadratic-time syndrome-based error/erasure decoding algorithms up to some of our proposed bounds.

DEUX défis de la théorie du codage algébrique sont traités dans cette thèse. Le premier est le décodage efficace (dur et souple) de codes de Reed–Solomon généralisés sur les corps finis en métrique de Hamming. La motivation pour résoudre ce problème vieux de plus de 50 ans a été renouvelée par la découverte par Guruswami et Sudan à la fin du 20ème siècle d'un algorithme polynomial de décodage jusqu'au rayon Johnson basé sur l'interpolation. Les premières méthodes de décodage algébrique des codes de Reed–Solomon généralisés faisaient appel à une équation clé, c'est à dire, une description polynomiale du problème de décodage. La reformulation de l'approche à base d'interpolation en termes d'équations clés est un thème central de cette thèse. Cette contribution couvre plusieurs aspects des équations clés pour le décodage dur ainsi que pour la variante décodage souple de l'algorithme de Guruswami–Sudan pour les codes de Reed–Solomon généralisés. Pour toutes ces variantes un algorithme de décodage efficace est proposé.

Le deuxième sujet de cette thèse est la formulation et le décodage jusqu'à certaines bornes inférieures sur leur distance minimale de codes en blocs linéaires cycliques. La caractéristique principale est l'intégration d'un code cyclique donné dans un code cyclique produit (généralisé). Nous donnons donc une description détaillée du code produit cyclique et des codes cycliques produits généralisés. Nous prouvons plusieurs bornes inférieures sur la distance minimale de codes cycliques linéaires qui permettent d'améliorer ou de généraliser des bornes connues. De plus, nous donnons des algorithmes de décodage d'erreurs/d'effacements [jusqu'à ces bornes] en temps quadratique.

Contents

List of Symbols and Acronyms

List of Symbols

$\mathbb{N}$	Set of natural numbers $\{0, 1, 2, \dots\}$
$\mathbb{Z}$	Set of integers $\{\dots, -2, -1, 0, 1, 2, \dots\}$
$\mathbb{Z}_n$	Ring of integers $\{0, 1, 2, \dots, n-1\}$
$\mathbb{F}_q$	Finite field of q elements $\{\beta_0, \beta_1, \dots, \beta_{q-1}\}$
$\mathbb{F}_q^n$	Vector space of dimension n over $\mathbb{F}_q$
$\mathbb{F}_q^*$	Multiplicative group of $\mathbb{F}_q$
$\mathbb{F}_q[X]$	Polynomial ring with indeterminate X over $\mathbb{F}_q$
$\mathbb{F}_q[X, Y]$	Polynomial ring with indeterminates X and Y over $\mathbb{F}_q$
$[a, b)$	The set of integers $\{a, a+1, \dots, b-1\}$
$[b)$	The set of integers $[0, b) = \{0, 1, \dots, b-1\}$
$[a]^+$	The maximum of a and 0
$\mathbf{a}$	Vector $\mathbf{a} = (a_0\ a_1\ \dots\ a_{n-1})$ of length n
$\mathbf{A} = (A_{i,j})_{i \in [m)}^{j \in [n)}$	A $m \times n$ matrix $\mathbf{A}$ with entries $A_{i,j}$
$d(\mathbf{a}, \mathbf{b})$	Hamming distance between $\mathbf{a}$ and $\mathbf{b}$
$\mathrm{wt}(\mathbf{a})$	Hamming weight of $\mathbf{a} \in \mathbb{F}_q^n$
$\mathrm{supp}(\mathbf{a})$	Support of $\mathbf{a} \in \mathbb{F}_q^n$
$[n, k]_q$	Parameters length n and dimension k of a linear code over $\mathbb{F}_q$
$[n, k, d]_q$	Parameters length n and dimension k of a linear code over $\mathbb{F}_q$ with minimum Hamming distance d
R	Code-rate $R = k/n$
$\mathrm{gcd}(a, b)$	Greatest common divisor of a and b
$\mathrm{lcm}(a, b)$	Least common multiple of a and b
τ	Decoding radius
m	Interpolation multiplicity
$\mathrm{Cost}(\mathbf{m})$	Costs of a multiplicity matrix $\mathbf{m}$
$\mathrm{Score}_{\mathbf{m}}(\mathbf{a})$	Score of a vector $\mathbf{a}$ w.r.t. the matrix $\mathbf{m}$
$\Lambda(X)$	Error-locator polynomial
$\Omega(X)$	Error-evaluator polynomial
$\mathbf{c}$, $c(X)$	Codeword, code polynomial

$\mathbf{e}$, $e(X)$	Error vector, error polynomial
E, ε	Set of errors positions $\operatorname{supp}(\mathbf{e})$, number of errors $\lvert\operatorname{supp}(\mathbf{e})\rvert$
$\mathbf{z}$, $z(X)$	Erasure vector, erasure polynomial
Z, ζ	Set of erasures positions $\operatorname{supp}(\mathbf{z})$, number of erasures $\lvert\operatorname{supp}(\mathbf{z})\rvert$
$\mathbf{r}$, $r(X)$	Received vector, received polynomial
$\Psi(X)$	Erasure-locator polynomial
$\Phi(X)$	Erasure-evaluator polynomial
$\mathcal{RS}(\boldsymbol{\alpha}, k)$	A normalized Reed–Solomon code over $\mathbb{F}_q$ of length n and dimension k with support $\boldsymbol{\alpha} \in \mathbb{F}_q^n$
$\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$	A Generalized Reed–Solomon code over $\mathbb{F}_q$ of length n and dimension k with support $\boldsymbol{\alpha} \in \mathbb{F}_q^n$ and scaling factors $\overline{\boldsymbol{\upsilon}} \in \mathbb{F}_q^n$
$\mathcal{CRS}(q, n, b, k)$	A cyclic Reed–Solomon code over $\mathbb{F}_q$ of length n and dimension k with consecutive zeros $\alpha^b, \alpha^{b+1}, \dots, \alpha^{b+n-k-1}$
$\mathcal{IGRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, \mathbf{k})$	An Interleaved Generalized Reed–Solomon code of interleaving order s with one common support set $\boldsymbol{\alpha} \in \mathbb{F}_q^n$, sets of scalar factors $\overline{\boldsymbol{\upsilon}} \in \mathbb{F}_q^{sn}$ and of dimension $\mathbf{k} \in \mathbb{N}^s$
D	Defining set of a cyclic code
$\mathcal{C}(D)$	A cyclic code $\mathcal{C}$ with defining set D
$g(X)$	Generator polynomial of a cyclic code
$M_{r,q}^{\langle n \rangle}$	Cyclotomic coset
$m_{r,q}^{\langle n \rangle}(X)$	Minimal polynomial
$\mathcal{A} \otimes \mathcal{B}$	Product code of $\mathcal{A}$ and $\mathcal{B}$
$\bigoplus_{i=0}^{s-1} \mathcal{C}_i$	Direct sum code of codes $\mathcal{C}_0, \mathcal{C}_1, \dots, \mathcal{C}_{s-1}$

List of Acronyms

BCH	**B**ose–Ray-**C**haudhuri–**H**ocquenghem (bound for the minimum distance of a cyclic code)
BMD	**B**ounded **M**inimum **D**istance (decoding)
EEA	**E**xtended **E**uclidean **A**lgorithm
FIA	**F**undamental **I**terative **A**lgorithm
GMD	**G**eneralized **M**inimum **D**istance
GRS	**G**eneralized **R**eed–**S**olomon (code)
IGRS	**I**nterleaved **G**eneralized **R**eed–**S**olomon (code)
IRS	**I**nterleaved **R**eed–**S**olomon (code)
MDS	**M**aximum **D**istance **S**eparable (code)
RS	(Normalized) **R**eed–**S**olomon (code)

1

"Computer science is no more about computers than astronomy is about telescopes."

Edsger W. Dijkstra (1930–2002)

Introduction

The publications of Claude E. Shannon [A-Sha48] and Richard W. Hamming [A-Ham50] marked the advent for the theory of "reliable communication in the presence of noise" and "error correcting codes". Shannon developed information entropy as a measure for uncertainty and defined the capacity C of a noisy channel. He showed that for any fixed rate $R < C$, there exist codes of rate R with small decoding error probability. Furthermore, he showed that longer codewords are more likely to be recovered. His theory is based on statistics and leads to information theory.

The theory of error correcting codes can be considered as an area of combinatorial mathematics. Hamming defined a notion of distance between codewords over finite fields—which we call now Hamming distance—and he observed that this is a metric—the Hamming metric. Furthermore, he constructed an explicit family of codes.

This dissertation deals with linear block codes over finite fields in Hamming metric. We recommend the tutorials of Berlekamp [A-Ber72], Sudan [O-Sud00; I-Sud01] and Costello–Forney [A-CF07], which inspired the following paragraphs.

In 1960, Irving S. Reed and Gustave Solomon [A-RS60] defined a class of algebraic codes that are probably the most extensively-used codes in practice, and as a consequence very well studied. This is due to several good attributes of Reed–Solomon (RS) codes and the fact that they lie in the intersection of numerous code families.

RS codes are standardized for magnetic and optical storage systems like hard drives, Compact-Discs (CDs), Digital-Versatile-Discs (DVDs), Blu-Ray-Discs (BDs) [B-Wic99], in Redundant Arrays of Inexpensive Disks (RAID) systems (see [A-Pla97; A-BHH13] for RAID-6), in communication systems like Digital Subscriber Line (DSL), in wireless communications standards like WiMax and broadcasting systems like Digital Video Broadcasting (DVB), in bar-codes like the nowadays popular 2D Quick-Response (QR) codes and in code-based crypto-systems like the McEliece public-key approach [O-McE78]. They are part of several code constructions e.g., rate-less Raptor codes [A-Sho06], interleaved and folded RS codes [A-Kra97; A-GR08] and concatenated schemes.

RS codes are Maximum-Distance-Separable (MDS) codes, i.e., they attain the Singleton bound with equality. They belong to the class of linear codes and if they are cyclic, RS codes are the super-codes of Bose–Ray-Chaudhuri–Hocquenghem codes [A-BRC60; A-Hoc59]. Algebraic-Geometry codes [A-BHHW98], Reed–Muller codes [A-Ree54; A-Mul54] and ideal-based/number-field codes as the Chinese-Remainder-Theorem codes [A-Man76] can be seen as generalizations of RS codes. Counterparts are defined over other algebraic structures as e.g., Galois rings [A-Arm10; O-Qui12] and in other metrics as e.g., Gabidulin codes [A-Del78; A-Gab85; A-Rot91] in rank-metric, which attract nowadays a lot of interest due to network-coding theory [A-KK08; A-SKK08].

Peterson [A-Pet60] developed the first decoding algorithm for RS codes which had cubic time complexity in code length. The decoding method of Berlekamp [B-Ber68] and Massey [A-Mas69] scales quadratic in time. The Berlekamp–Massey algorithm as well as the modification of the Extended Euclidean Algorithm by Sugiyama–Kasahara–Hirasawa–Namekawa [A-SKHN75] became standard approaches of decoding RS codes for decades. The discrete Fourier transform and its inverse operation can be used to describe elementary properties and the classic decoding of RS codes (see [B-Bla83, Chapter 8] and [B-Bos13, Chapter 3]). Delsarte [A-Del75] extended the definition of RS codes to so-called Generalized Reed–Solomon (GRS) codes, which we consider throughout this thesis.

The discovery of a polynomial-time interpolation-based decoding principle up to the Johnson radius [A-Joh62; A-Bas65] of Guruswami and Sudan [A-Sud97; A-GS99] for GRS and Algebraic-Geometry codes at the end of the 20th century revived the interest in algebraic coding theory. The simplicity of their approach inspired many researchers to re-think about list decoding of aforementioned related code families like Reed–Muller codes [A-FT08; A-DKT07], Hadamard codes [A-GRS00b], Chinese-Remainder-Theorem codes [A-GRS00a; I-GSS00; I-LS12] and Gabidulin codes [A-Wac13]. The impact of a feasible list decoding algorithm on applications, where GRS or Algebraic-Geometry codes are used, is in the focus of several investigations (see e.g., [O-Bar11] for the McEliece crypto-system).

Guruswami and Sudan proposed a new soft-decision decoding variant for GRS codes (and related code families) by assigning different multiplicities for the interpolation step. Their approach was elaborated by Kötter and Vardy [A-KV03a] for RS codes. The Kötter–Vardy decoding approach is believed to provide the best performance in terms of coding gain among all polynomial-time soft-decision decoding approaches for GRS codes and due to complexity-reducing techniques, as the re-encoding transformation [A-KMV11], its wide-spread deployment in practical systems is probable.

Cyclic codes were first introduced by Prange [O-Pra57] and the first difference to RS codes is that their distance is not obvious from their length and dimension. The second is that they are defined over the base field and therefore have some advantages. Cyclic codes are often referred to as Bose–Ray-Chaudhuri–Hocquenghem (BCH) codes, which is somehow misleading. The BCH bound was the first lower bound on the minimum distance of cyclic codes. The Berlekamp–Massey as well as the Sugiyama–Kasahara–Hirasawa–Namekawa [A-SKHN75] algorithm can be used to decode up to the BCH bound. A challenge is to find good lower bounds on the minimum distance of cyclic codes and to develop efficient, i.e., at most quadratic-time, hard- and soft-decision decoding algorithms. Feng and Tzeng [A-FT89; A-FT91b] generalized the approach of Berlekamp–Massey and Sugiyama–Kasahara–Hirasawa–Namekawa to decode up to the Hartmann–Tzeng bound [A-HT72], which was the first generalization of the BCH bound. Several lower bounds and decoding algorithms exist and we refer to them when appropriate.

This dissertation is structured as follows.

In Chapter 2, we give necessary preliminaries for linear (cyclic) block codes in Hamming metric and bivariate polynomials over finite fields. The Hartmann–Tzeng bound [A-HT72] for cyclic codes is proven. Combining methods for linear (cyclic) codes that lead to Slepian's product codes [A-Sle60], Forney's concatenated codes [O-For66a] and Blokh and Zyablov's generalized concatenation [A-BZ74] are discussed in Chapter 2. Furthermore, we define GRS codes and Interleaved GRS codes (the description of GRS codes is close to Roth's [B-Rot06, Chapter 5]).

Chapter 3 describes hard- and soft-decision decoding approaches for linear codes in general and GRS codes in particular. We derive the Key Equation for syndrome-based error/erasure Bounded Minimum Distance (BMD) decoding of GRS codes from the simplest interpolation-based approach [O-WB86]. The modification of the Extended Euclidean Algorithm is outlined, while the Fundamental Iterative Algorithm (FIA) is discussed extensively. Furthermore, we outline the collaborative decoding of Interleaved GRS codes, the interpolation-based principle of Guruswami–Sudan [A-Sud97; A-GS99] and the soft-decision variant of Kötter and Vardy [A-KMV11] for GRS codes.

Chapters 4, 5 and 6 cover new results, in parts already published, and therefore appropriately referenced.

Two variants of Key Equations for decoding GRS codes beyond half the minimum distance are given in Chapter 4. The derivation of both is done in detail and the adaption of the FIA is described. The correctness of the FIA is proven and its complexity is analyzed. Furthermore, some future research directions are given.

The univariate reformulation of the bivariate Kötter–Vardy soft-decision interpolation problem for GRS codes is derived in Chapter 5 and the obtained set of Key Equations is given. We investigate the re-encoding transformation [A-KMV11] and give a modified set of Key Equations in Chapter 5. The adaption of the FIA for this case is roughly outlined.

In Chapter 6, we propose four new bounds on the minimum distance of linear cyclic codes, denoted by bound I-a, I-b, II and III. Bound I-a is very close to bound I-b. While bound I-a is based on the association of a rational function, the embedding of a given cyclic code into a cyclic product code is the basis of bound I-b. The idea of embedding a code into a product code is extended by bound II and III. We prove the main theorems for the bounds and give syndrome-based error/erasure decoding algorithms up to bounds I-a, I-b and II. Good candidates for the embedding-technique are discussed and, as a first result, conditions for non-primitive lowest-code-rate binary codes of minimum Hamming distance two and three are given. The work is based on the contributions of Charpin, Tietäväinen and Zinoviev [A-CTZ97; A-CTZ99]. Furthermore, we outline how embedding a given cyclic code into a cyclic product code can be extended to the embedding into a cyclic variant of generalized product codes, which has not been defined before.

We summarize and conclude this contribution in Chapter 7.

2

"Science is facts; just as houses are made of stones, so is science made of facts; but a pile of stones is not a house and a collection of facts is not necessarily science."

Henri Poincaré (1854–1912)

Linear Block Codes over Finite Fields

Necessary properties of linear block codes over finite fields in Hamming metric are covered in this chapter. In the next section, we recall the Lagrange interpolation theorem and define relevant properties of bivariate polynomials over finite fields.

In Section 2.2, we define basic properties of linear and cyclic block codes. We prove the Hartmann–Tzeng [A-Har72; A-HTC72; A-HT72; A-HT74] bound, which was the first generalization of the Bose–Ray-Chaudhuri-Hocquenghem (BCH), [A-BRC60; A-Hoc59] lower bound on the minimum Hamming distance of a cyclic code.

The product of linear codes is introduced in Section 2.3. Cyclic product codes and generalized concatenated codes are special cases of product codes. We give the conditions for the product code to be cyclic and illustrate the defining set with an example. In Section 2.4, we define Generalized Reed–Solomon (GRS) codes as introduced by [A-Del75] based on the work of Reed and Solomon [A-RS60]. The notation of normalized, cyclic and primitive RS codes is given. In addition, we define Interleaved Generalized Reed–Solomon (IGRS) codes and connect them to product codes.

2.1 Basic Notations

2.1.1 Hamming Metric and Polynomials over Finite Fields

Let $\mathbb{N}$ denote the set of natural numbers, $\mathbb{Z}$ the set of integers and $\mathbb{F}_q$ the finite field of order q. Let $\mathbb{F}_q[X]$ be the polynomial ring over $\mathbb{F}_q$ with indeterminate X. The polynomial ring with indeterminates X and Y over $\mathbb{F}_q$ is denoted by $\mathbb{F}_q[X, Y]$. For two given integers a and b, we denote by $[a, b)$ the set $\{a, a+1, \dots, b-1\}$ and by $[b)$ the set $[0, b)$.

A vector of length n is denoted by a bold letter as $\mathbf{a}= (a_0 \; a_1 \; \dots \; a_{n-1}) \in \mathbb{F}_q^n$. An $m \times n$ matrix is denoted by a bold letter as $\mathbf{A} = (A_{i,j})_{i\in[m]}^{j\in[n]}$ in $\mathbb{F}_q^{m\times n}$. A set of n elements $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ is denoted by a capital letter as $D = \{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\}$.

A linear $[n, k]_q$ code of length n and dimension k over $\mathbb{F}_q$ is denoted by a calligraphic letter like $\mathcal{C}$. We also use $[n, k, d]_q$ to include the minimum Hamming distance d (see Definition 2.11). The code-rate is denoted by $R = k/n$.

We denote a univariate polynomial in $\mathbb{F}_q[X]$ by $A(X) = \sum_i A_i X^i \in \mathbb{F}_q[X]$. A bivariate polynomial in $\mathbb{F}_q[X, Y]$ is $B(X, Y) = \sum_i B_i(X) Y^i$, where $B_i(X) \in \mathbb{F}_q[X]$. We denote $B(X, Y) = \sum_i \sum_j B_{i,j} X^j Y^i$.

The greatest common divisor of two elements a, b in a Euclidean Domain is denoted by $\gcd(a, b)$ and their least common multiple by $\operatorname{lcm}(a, b)$.

The support of $\mathbf{v} \in \mathbb{F}_q^n$ is the set $\mathrm{supp}(\mathbf{v}) = \{i : v_i \neq 0\}$. The Hamming weight of a vector $\mathbf{v}$ is the cardinality of its support and denoted by $\mathrm{wt}(\mathbf{v}) = |\mathrm{supp}(\mathbf{v})|$.

> **Definition 2.1 (Hamming Distance)**
> Given two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{F}_q^n$, the Hamming distance $d(\mathbf{a}, \mathbf{b})$ is defined as:
>
> $$d(\mathbf{a}, \mathbf{b}) = \mathrm{wt}(\mathbf{a} - \mathbf{b}) = \left| \left\{ i : a_i \neq b_i, \quad \forall i \in [n) \right\} \right|.$$

The Hamming distance is a metric, the so-called Hamming metric, because it fulfills for any three vectors $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{F}_q^n$:

1. $d(\mathbf{a}, \mathbf{b}) \geq 0$,
2. $d(\mathbf{a}, \mathbf{b}) = d(\mathbf{b}, \mathbf{a})$,
3. $d(\mathbf{a}, \mathbf{b}) = 0 \Leftrightarrow \mathbf{a} = \mathbf{b}$,
4. $d(\mathbf{a}, \mathbf{c}) \leq d(\mathbf{a}, \mathbf{b}) + d(\mathbf{b}, \mathbf{c})$.

The scalar (or inner) product of two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{F}_q^n$ is:

$$\langle\, \mathbf{a}, \mathbf{b} \,\rangle = \mathbf{a}\mathbf{b}^T = \sum_{i=0}^{n-1} a_i b_i.$$

2.1.2 The Univariate Lagrange Interpolation Problem

Given n distinct elements $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ in $\mathbb{F}_q$. Define:

$$L(X) \overset{\text{def}}{=} \prod_{i=0}^{n-1} (X - \alpha_i), \tag{2.1}$$

and let

$$L_i(X) \overset{\text{def}}{=} \frac{L(X)}{X - \alpha_i} = \prod_{\substack{j=0 \\ j \neq i}}^{n-1} (X - \alpha_j). \tag{2.2}$$

The following formula was given by Lagrange in 1795 and we restrict ourselves here to points in $\mathbb{F}_q$ and a univariate polynomial in $\mathbb{F}_q[X]$.

> **Theorem 2.2 (Univariate Lagrange Interpolation)**
> Let $n < q$ distinct elements $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ in $\mathbb{F}_q$ and n (not necessarily distinct) elements $r_0, r_1, \dots, r_{n-1} \in \mathbb{F}_q$ be given. The unique Lagrange interpolation polynomial $R(X) \in \mathbb{F}_q[X]$ of degree smaller than n with:
>
> $$R(\alpha_i) = r_i, \quad \forall i \in [n)$$
>
> is given by:
>
> $$R(X) = \sum_{i=0}^{n-1} r_i \frac{L_i(X)}{L_i(\alpha_i)}. \tag{2.3}$$

2.1.3 Bivariate Polynomials over Finite Fields

We first define the weighted degree of a bivariate polynomial with coefficients in $\mathbb{F}_q$.

Definition 2.3 (Weighted Degree)
Let two integers a and b be given. The (a, b)-weighted degree of a monomial $X^iY^j \in \mathbb{F}_q[X, Y]$ is defined as:

$$\mathrm{wdeg}_{\mathrm{a,b}}\, X^iY^j \overset{\mathrm{def}}{=} ai + bj,$$

and the weighted degree of a bivariate polynomial $Q(X, Y) \in \mathbb{F}_q[X, Y]$ with $Q(X, Y) = \sum_i \sum_j Q_{i,j}X^iY^j$ is defined as:

$$\mathrm{wdeg}_{\mathrm{a,b}}\, Q(X, Y) \overset{\mathrm{def}}{=} \max_{\substack{(i,j),\\ Q_{i,j}\neq 0}} (ai + bj).$$

We define the Hasse derivative in the following. It is sometimes referred to as hyper-derivative.

Definition 2.4 (Mixed Hasse Derivative [A-Has36, Equation (2)])
Let two integers a and b be given. The (a, b)-th Hasse derivative of $Q(X, Y) = \sum_i \sum_j Q_{i,j}X^iY^j$ in $\mathbb{F}_q[X, Y]$ is defined as:

$$Q^{[a,b]}(X, Y) \overset{\mathrm{def}}{=} \sum_{i\geq a} \sum_{j\geq b} \binom{i}{a}\binom{j}{b} Q_{i,j}X^{i-a}Y^{j-b}.$$

and we introduce the following short-hand notation. Let

$$Q^{[b]}(X, Y) \overset{\mathrm{def}}{=} Q^{[0,b]}(X, Y)$$

denote the b-th Hasse derivative of $Q(X, Y)$ with respect to the variable Y.

Definition 2.5 (Multiplicity of a Bivariate Polynomial)
A bivariate polynomial $Q(X, Y) \in \mathbb{F}_q[X, Y]$ has multiplicity m at $(\alpha, \beta) \in \mathbb{F}_q^2$ if

$$Q^{[a,b]}(\alpha, \beta) = 0, \quad \forall a, b \text{ with } a + b < m.$$

Let us introduce Taylor's expansion for a bivariate polynomial in $\mathbb{F}_q[X, Y]$.

Theorem 2.6 (Taylor's Formula)
Let $Q(X, Y) = \sum_i \sum_j Q_{i,j}X^iY^j \in \mathbb{F}_q[X, Y]$. For any $(\alpha, \beta) \in \mathbb{F}_q^2$, we have:

$$Q(X + \alpha, Y + \beta) = \sum_a \sum_b Q^{[a,b]}(\alpha, \beta)X^aY^b.$$

Proof We have:

$$\begin{aligned}
Q(X+\alpha, Y+\beta) &= \sum_i \sum_j Q_{i,j}(X+\alpha)^i (Y+\beta)^j \\
&= \sum_i \sum_j Q_{i,j} \left(\sum_{a=0}^{i} \binom{i}{a} X^a \alpha^{i-a} \right) \left(\sum_{b=0}^{j} \binom{j}{b} Y^b \beta^{j-b} \right) \\
&= \sum_a \sum_b X^a Y^b \left(\sum_i \sum_j \binom{i}{a} \binom{j}{b} Q_{i,j} \alpha^{i-a} \beta^{j-b} \right) \\
&= \sum_a \sum_b Q^{[a,b]}(\alpha, \beta) X^a Y^b.
\end{aligned}$$

■

The following corollary is a direct consequence of Theorem 2.6.

Corollary 2.7 (Multiplicity of Bivariate Polynomials)
A bivariate polynomial $Q(X,Y) = \sum_i \sum_j Q_{i,j} X^i Y^j \in \mathbb{F}_q[X,Y]$ has multiplicity m at $(\alpha,\beta) \in \mathbb{F}_q^2$ if and only if the shifted polynomial

$$Q(X+\alpha, Y+\beta) = \sum_i \sum_j Q'_{i,j} X^i Y^j$$

has no term of total degree less than m, i.e., $Q'_{i,j} = 0$, if $i+j < m$. Equivalently, we can say that $Q(X+\alpha, Y+\beta)$ has multiplicity of order m at $(0,0)$.

The following lemma is essential for the re-encoding transformation technique for decoding Generalized Reed–Solomon codes (see Section 5.2).

Lemma 2.8 (Multiplicity with One Zero-Coordinate)
A bivariate polynomial $Q(X,Y) = \sum_j Q_j(X) Y^j \in \mathbb{F}_q[X,Y]$ has multiplicity m at the point $(\alpha, 0)$ if and only if the univariate polynomials $Q_j(X)$ are divisible by $(X-\alpha)^{m-j}$ for all $j \in [m)$.

Proof The translated bivariate polynomial is

$$Q(X+\alpha, Y+0) = \sum_j Q_j(X+\alpha) Y^j,$$

and the first $m-j-1$ coefficients of $Q_j(X+\alpha)$ are zero according to Definition 2.5. In other words, the univariate polynomial $Q_j(X+\alpha)$ has multiplicity $m-j$ at 0. This implies that $X^{m-j} | Q_j(X+\alpha)$ and therefore $(X-\alpha)^{m-j} | Q_j(X)$. ■

For $m \in \mathbb{Z}$, define $[m]^+ \overset{\text{def}}{=} \max(m, 0)$.

Corollary 2.9 (Multiplicity with One Zero-Coordinate)
A bivariate polynomial $Q(X,Y) = \sum_j Q_j(X) Y^j \in \mathbb{F}_q[X,Y]$ has multiplicities $m_0, m_1, \dots, m_{k-1}$ at the points $(\alpha_0, 0), (\alpha_1, 0), \dots, (\alpha_{k-1}, 0)$ if and only if the univariate polynomials $Q_j(X)$ are divisible by $\prod_{i=0}^{k-1} (X-\alpha_i)^{[m_i - j]^+}$.

We define the inner product of two polynomials $A(X) = \sum_i A_i X^i$ and $B(X) = \sum_i B_i X^i$ in $\mathbb{F}_q[X]$ as:

$$\langle\, A(X), B(X) \,\rangle \overset{\text{def}}{=} \sum_i A_i B_i. \tag{2.4}$$

For two bivariate polynomials $A(X,Y) = \sum_i \sum_j A_{i,j} X^i Y^j$ and $B(X,Y) = \sum_i \sum_j B_{i,j} X^i Y^j$ in $\mathbb{F}_q[X,Y]$ the inner product is

$$\langle\, A(X,Y), B(X,Y) \,\rangle \overset{\text{def}}{=} \sum_{i,j} A_{i,j} B_{i,j}. \tag{2.5}$$

2.2 Basics of Linear Block Codes and Cyclic Codes

2.2.1 Basics of Linear Block Codes

Definition 2.10 (Linear Code)
A code $\mathcal{C}$ over $\mathbb{F}_q$ is called linear if $\mathcal{C}$ is a linear subspace of $\mathbb{F}_q^n$, i.e., for any two codewords $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$ and two elements $\alpha, \beta \in \mathbb{F}_q$ we have

$$\alpha\mathbf{c} + \beta\mathbf{c}' \in \mathcal{C}.$$

Let $\mathcal{C}$ denote such a linear $[n,k]_q$ code. The mapping

$$\begin{array}{rccl} \text{enc:} & \mathbb{F}_q^k & \to & \mathbb{F}_q^n \\ & \mathbf{m} & \mapsto & \text{enc}(\mathbf{m}) = \mathbf{mG} \end{array}$$

defines the encoding of a linear code $\mathcal{C}$. The generator matrix $\mathbf{G} = (G_{i,j})_{i\in[k]}^{j\in[n)}$ of $\mathcal{C}$ is a $k \times n$ matrix with $G_{i,j} \in \mathbb{F}_q$, whose rows form a basis of the code. The generator matrix $\mathbf{G}$ is not unique and its rank k equals the dimension of the code $\mathcal{C}$. A parity-check matrix $\mathbf{H}$ has rank $n-k$ and is (in most cases) a $(n-k) \times n$ matrix over $\mathbb{F}_q$ such that for every:

$$\mathbf{c} \in \mathcal{C} \iff \mathbf{Hc}^T = \mathbf{0}$$

holds.

Definition 2.11 (Minimum Hamming Distance of a Linear Code)
Let $\mathcal{C}$ be an $[n,k]_q$ linear code. The minimum Hamming distance d of $\mathcal{C}$ is:

$$d \overset{\text{def}}{=} \min_{\substack{\mathbf{c},\mathbf{c}' \in \mathcal{C} \\ \mathbf{c} \neq \mathbf{c}'}} \Big(d(\mathbf{c}, \mathbf{c}') \Big) = \min_{\substack{\mathbf{c},\mathbf{c}' \in \mathcal{C} \\ \mathbf{c} \neq \mathbf{c}'}} \Big(\text{wt}(\mathbf{c} - \mathbf{c}') \Big),$$

and due to linearity $\mathbf{c}' = \mathbf{0} \in \mathcal{C}$ and thus

$$d = \min_{\mathbf{c} \in \mathcal{C} \setminus \{\mathbf{0}\}} \text{wt}(\mathbf{c}).$$

We denote a linear $[n,k]_q$ code $\mathcal{C}$ with minimum Hamming distance d as an $[n,k,d]_q$ code.

Let us define the dual of a code.

Definition 2.12 (Dual Code)
Let $\mathcal{C}$ be an $[n, k, d]_q$ linear code with an $(n-k) \times n$ parity-check matrix $\mathbf{H}$. The dual code of $\mathcal{C}$ contains all vectors $\mathbf{a} \in \mathbb{F}_q^n$, such that:

$$\mathbf{a} \cdot \mathbf{c}^T = 0, \quad \forall \mathbf{c} \in \mathcal{C}. \tag{2.6}$$

Then, the linear $[n, n-k, d^\perp]_q$ code with generator matrix $\mathbf{H}$ is the dual of the code $\mathcal{C}$ and is denoted by $\mathcal{C}^\perp$.

Note that $\mathcal{C} = (\mathcal{C}^\perp)^\perp$.

The following definition is relevant for the description of (generalized) concatenated codes in Subsection 2.3.3.

Definition 2.13 (Direct Sum of Linear Codes)
Let s linear $[n, k_i, d_i]_q$ codes $\mathcal{C}_i, \ \forall i \in [s)$ over the same alphabet $\mathbb{F}_q$ and of equal length n be given. Furthermore, let $\sum_{i=0}^{s-1} k_i < n$ and let

$$\bigcap_{i=0}^{s-1} \mathcal{C}_i = \{\mathbf{0}\}. \tag{2.7}$$

Then, the direct sum code is defined as:

$$\bigoplus_{i=0}^{s-1} \mathcal{C}_i = \left\{ \sum_{i=0}^{s-1} \mathbf{c}_i \ : \ \mathbf{c}_i \in \mathcal{C}_i, \ \forall i \in [s) \right\}. \tag{2.8}$$

The following theorem gives the essential properties of a direct sum code.

Theorem 2.14 (Linearity and Minimum Distance of a Direct Sum Code)
Let $\mathcal{C} = \bigoplus_{i=0}^{s-1} \mathcal{C}_i$ be a direct sum of s linear $[n, k_i, d_i]_q$ codes $\mathcal{C}_i$ as in Definition 2.13. Then $\mathcal{C}$ is a linear $[n, \sum_{i=0}^{s-1} k_i, d]_q$ code with minimum distance:

$$d \leq \min_{i \in [s)} (d_i). \tag{2.9}$$

Proof Linearity follows from the definition. The dimension is guaranteed by Condition (2.7), because then the $\sum_{i=0}^{s-1} k_i$ rows of the generator matrix of $\mathcal{C}$ are linearly independent. The distance follows from the fact that every $\mathcal{C}_i$ is a subset of $\mathcal{C}$. ■

The following corollary plays an important role for the construction of generalized concatenated codes (see Subsection 2.3.3).

Corollary 2.15 (Direct Sum Code)
Let s linear $[n, k_i, d_i]_q$ codes $\mathcal{C}_i, \ \forall i \in [s)$ with:

$$\mathcal{C}_0 \supset \mathcal{C}_1 \supset \cdots \supset \mathcal{C}_{s-1}$$

be given. Then, we have

$$\mathcal{C}_0 = (\mathcal{C}_0 \backslash \mathcal{C}_1) \oplus (\mathcal{C}_1 \backslash \mathcal{C}_2) \oplus \cdots \oplus \mathcal{C}_{s-1}.$$

2.2.2 Cyclic Codes

Cyclic $[n, k, d]_q$ codes are extensively discussed in the literature and they are well studied. We refer to [B-PWBJ12, Chapter 7], [B-MS88a, Chapter 7 and 8], [O-Cha98], [B-LC04, Chapter 5], [B-PW72, Chapter 8] and [B-Bos13, Chapter 4].

For a given $[n, k, d]_q$ cyclic code $\mathcal{C}$, we denote a codeword by $\mathbf{c} = (c_0 \, c_1 \, \ldots \, c_{n-1}) \in \mathcal{C}$ and equivalently in polynomial form by $c(X) = \sum_{i=0}^{n-1} c_i X^i$ in $\mathbb{F}_q[X]$. Let us first define a cyclic code over $\mathbb{F}_q$.

Definition 2.16 (Cyclic Code)
A linear $[n, k, d]_q$ code $\mathcal{C}$ is called cyclic over $\mathbb{F}_q$ if every cyclic shift of a codeword in $\mathcal{C}$ is also a codeword, i.e.:

$$c(X) \in \mathcal{C} \quad \Rightarrow \quad X \cdot c(X) \mod (X^n - 1) \in \mathcal{C}. \tag{2.10}$$

A linear $[n, k, d]_q$ cyclic code $\mathcal{C}$ is then an ideal in the ring $\mathbb{F}_q[X]/(X^n - 1)$ generated by $g(X)$. The generator polynomial $g(X)$ has roots in the splitting field $\mathbb{F}_{q^l}$ of $X^n - 1$, where $n \mid (q^l - 1)$.

Definition 2.17 (Cyclotomic Coset and Minimal Polynomial)
Let three integers r, n, q with $\gcd(n, q) = 1$ and $r < n$ be given. A cyclotomic coset $M_{r,q}^{\langle n \rangle}$ is defined as:

$$M_{r,q}^{\langle n \rangle} \overset{\text{def}}{=} \left\{ rq^j \bmod n \,|\, j \in [n_r) \right\}, \tag{2.11}$$

where n_r is the smallest integer such that

$$rq^{n_r} \equiv r \mod n.$$

Let α be an n-th root of unity of $\mathbb{F}_{q^l}$. The minimal polynomial $m_{r,q}^{\langle n \rangle}(X)$ of the element α^r is given by:

$$m_{r,q}^{\langle n \rangle}(X) \overset{\text{def}}{=} \prod_{i \in M_{r,q}^{\langle n \rangle}} (X - \alpha^i), \tag{2.12}$$

and it is well-known that $m_{r,q}^{\langle n \rangle}(X) \in \mathbb{F}_q[X]$.

Let α be an n-th root of unity. The defining set D of an $[n, k, d]_q$ cyclic code $\mathcal{C}$ is defined as:

$$D \overset{\text{def}}{=} \left\{ 0 \leq i \leq n-1 \,|\, g(\alpha^i) = 0 \right\}. \tag{2.13}$$

Therefore, we denote $\mathcal{C}(D)$ for a linear cyclic code $\mathcal{C}$ with defining set D. Clearly, we have

$$g(X) = \prod_{i \in D} (X - \alpha^i)$$

and $\deg g(X) = |D| = n - k$. Furthermore, we introduce the following short-hand notations for a given set D and a non-zero integer z:

$$(D \cdot z)_n \overset{\text{def}}{=} \Big\{(i \cdot z) \bmod n \mid i \in D\Big\}, \tag{2.14}$$

$$(D + z)_n \overset{\text{def}}{=} \Big\{(i + z) \bmod n \mid i \in D\Big\}, \tag{2.15}$$

$$(D + z) \overset{\text{def}}{=} \Big\{(i + z) \mid i \in D\Big\}. \tag{2.16}$$

Let us prove the Hartmann–Tzeng [A-Har72; A-HT72] lower bound on the minimum distance d of an $[n, k, d]_q$ cyclic code. We present it in polynomial form, which we use later on.

> **Theorem 2.18 (Hartmann–Tzeng Bound [A-Har72; A-HT72])**
> Let an $[n, k, d]_q$ cyclic code $\mathcal{C}(D)$ with defining set D be given. Let α denote an n-th root of unity. Let four integers f, m, δ and ν with $m \neq 0$ and $\gcd(n, m) = 1$, $\delta \geq 2$ and $\nu \geq 0$ be given, such that:
>
> $$\begin{aligned}\Big(&\{0, m, 2m, \dots, (\delta-2)m\}\\ &\cup \{1, 1+m, 1+2m, \dots, 1+(\delta-2)m\}\\ &\qquad\qquad\qquad \ddots\\ &\qquad \cup \{\nu, \nu+m, \nu+2m, \dots, \nu+(\delta-2)m\}\Big)_n \subseteq (D+f)_n.\end{aligned} \tag{2.17}$$
>
> Then, $d \geq d^*_{\mathsf{HT}} \overset{\text{def}}{=} \delta + \nu$.

Proof Equivalently, we can state that for the four parameters f, m, δ and ν with $m \neq 0$, the following:

$$\begin{aligned}\sum_{i=0}^{\infty} c(\alpha^{f+im+j})X^i &= c(\alpha^{f+j}) + c(\alpha^{f+m+j})X + c(\alpha^{f+2m+j})X^2 + \dots\\ &\equiv 0 \mod X^{\delta-1}, \quad \forall j \in [\nu+1)\end{aligned} \tag{2.18}$$

holds for all $c(X) \in \mathcal{C}$.

Let $c(X) \in \mathcal{C}$ and let $Y = \{i_0, i_1, \dots, i_{y-1}\}$ denote the support of $c(X)$, where $y \geq d$ holds for all codewords except the all-zero codeword. We linearly combine these $\nu + 1$ sequences (or equations) as in (2.18). The scalar factors for each power series as in (2.18) is $\lambda_i \in \mathbb{F}_{q^l}$ for $i \in [\nu+1)$. We obtain from (2.18):

$$\sum_{i=0}^{\infty}\sum_{j=0}^{\nu} \lambda_j c(\alpha^{f+im+j})X^i \equiv 0 \mod X^{\delta-1}$$

$$\sum_{i=0}^{\infty}\sum_{j=0}^{\nu}\sum_{u \in Y} \lambda_j \big(c_u \alpha^{u(f+im+j)}\big)X^i \equiv 0 \mod X^{\delta-1}.$$

We re-order it according to the codeword coefficients:

$$\begin{aligned}\sum_{i=0}^{\infty}\sum_{u\in Y}\sum_{j=0}^{\nu}\lambda_j\big(c_u\alpha^{u(f+im+j)}\big)X^i &= \sum_{i=0}^{\infty}\sum_{u\in Y}\Big(c_u\alpha^{u(f+im)}\sum_{j=0}^{\nu}\alpha^{uj}\lambda_j\Big)X^i\\ &\equiv 0 \mod X^{\delta-1}.\end{aligned} \tag{2.19}$$

We want to annihilate the first ν terms of $c_{i_0}, c_{i_1}, \dots, c_{i_{y-1}}$. From (2.19), the following linear system of equations with $\nu+1$ unknowns is obtained:

$$\begin{pmatrix} 1 & \alpha^{i_0} & \alpha^{i_0 2} & \cdots & \alpha^{i_0\nu}\\ 1 & \alpha^{i_1} & \alpha^{i_1 2} & \cdots & \alpha^{i_1\nu}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ 1 & \alpha^{i_\nu} & \alpha^{i_\nu 2} & \cdots & \alpha^{i_\nu\nu}\end{pmatrix}\cdot\begin{pmatrix}\lambda_0\\ \lambda_1\\ \vdots\\ \lambda_\nu\end{pmatrix} = \begin{pmatrix}0\\ \vdots\\ 0\\ 1\end{pmatrix}, \tag{2.20}$$

and it is guaranteed to find a unique non-zero solution, because the $(\nu+1)\times(\nu+1)$ matrix in (2.20) is a Vandermonde matrix and therefore has full rank.

Let $\widetilde{Y} \overset{\text{def}}{=} Y \setminus \{i_0, i_1, \dots, i_{\nu-1}\}$. Then, we can rewrite (2.19):

$$\sum_{i=0}^{\infty}\left(\sum_{u\in\widetilde{Y}} c_u\alpha^{u(f+im)}\sum_{j=0}^{\nu}\alpha^{uj}\lambda_j\right)X^i \equiv 0 \mod X^{\delta-1}.$$

This leads to:

$$\sum_{u\in\widetilde{Y}}\frac{c_u\alpha^{uf}\sum_{j=0}^{\nu}\alpha^{uj}\lambda_j}{1-\alpha^{mu}X} \equiv 0 \mod X^{\delta-1},$$

and we can bring it to the least common denominator:

$$\frac{\sum\limits_{u\in\widetilde{Y}}\Big(c_u\alpha^{uf}\sum_{j=0}^{\nu}\alpha^{uj}\lambda_j \prod\limits_{h\in\widetilde{Y}\setminus\{u\}}(1-\alpha^{mh}X)\Big)}{\prod\limits_{u\in\widetilde{Y}}(1-\alpha^{mu}X)} \equiv 0 \mod X^{\delta-1},$$

where the degree of the numerator is smaller than or equal to $y-1-\nu$ and has to be at least $\delta-1$. Therefore for $y \geq d$, we have:

$$\begin{aligned} d-1-\nu &\geq \delta-1,\\ d &\geq \delta+\nu. \end{aligned}$$

■

Note that for $\nu = 0$, the Hartmann–Tzeng bound d^*_{HT} becomes the Bose–Ray-Chaudhuri–Hocquenghem (BCH) bound [A-Hoc59; A-BRC60] and is denoted by d^*_{BCH}.

2.3 Product Codes

2.3.1 Basic Principle

Elias introduced so-called i-iterated codes in [A-Eli54] that coincide for $i = 2$ with the established construction of product codes. The term "product code" was first used by Slepian in 1960 [A-Sle60]. Slepian used the Kronecker product to define the generator matrix of a linear product code. They are discussed in [B-MS88a, Chapter 18] and an overview of iterative decoding approaches for product codes can be found in the survey paper of Kschischang [O-Ksc03]. We use cyclic product codes (see Subsection 2.3.2) to bound the minimum distance of cyclic codes. In the following, we shortly introduce basic properties of product codes.

In this subsection, let $\mathcal{A}$ denote an $[n_a, k_a, d_a]_q$ code and $\mathcal{B}$ denote an $[n_b, k_b, d_b]_q$ code over the same field $\mathbb{F}_q$. For simplicity, we assume that the first k_a and k_b symbols of a codeword are the information symbols of $\mathcal{A}$ and $\mathcal{B}$, respectively.

Definition 2.19 (Product Code)
Let $\mathcal{A}$ be an $[n_a, k_a, d_a]_q$ code with a $k_a \times n_a$ generator matrix $\mathbf{G}^{\langle a \rangle}$ and let $\mathcal{B}$ be an $[n_b, k_b, d_b]_q$ code with a $k_b \times n_b$ generator matrix $\mathbf{G}^{\langle b \rangle}$. The code with the $(k_a k_b) \times (n_a n_b)$ generator matrix

$$\begin{aligned}\mathbf{G}^{\langle b \rangle} \otimes \mathbf{G}^{\langle a \rangle} &= \left(G_{i,j}^{\langle b \rangle} \mathbf{G}^{\langle a \rangle} \right)_{i \in [k_b)}^{j \in [n_b)} \\ &= \begin{pmatrix} G_{0,0}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} & G_{0,1}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} & \dots & G_{0,n_b-1}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} \\ G_{1,0}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} & G_{1,1}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} & \dots & G_{1,n_b-1}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} \\ \vdots & \vdots & \ddots & \vdots \\ G_{k_b-1,0}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} & G_{k_b-1,1}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} & \dots & G_{k_b-1,n_b-1}^{\langle b \rangle}\mathbf{G}^{\langle a \rangle} \end{pmatrix}\end{aligned} \tag{2.21}$$

is called direct product code and is denoted by $\mathcal{A} \otimes \mathcal{B}$.

Therefore, the mapping:

$$\begin{aligned}\text{enc-pc:}\quad \mathbb{F}_q^{k_a \cdot k_b} &\rightarrow \mathbb{F}_q^{n_a \cdot n_b} \\ \mathbf{m} &\mapsto \text{enc-pc}(\mathbf{m}) = \mathbf{m}\left(\mathbf{G}^{\langle b \rangle} \otimes \mathbf{G}^{\langle a \rangle}\right)\end{aligned}$$

defines the encoding of a product code $\mathcal{A} \otimes \mathcal{B}$.

Notice that the Kronecker product of two matrices is non-commutative, i.e., in general $\mathbf{G}^{\langle b \rangle} \otimes \mathbf{G}^{\langle a \rangle} \neq \mathbf{G}^{\langle a \rangle} \otimes \mathbf{G}^{\langle b \rangle}$, but both matrices generate the same code. If the generator matrix of the product codes is $\mathbf{G}^{\langle b \rangle} \otimes \mathbf{G}^{\langle a \rangle}$, then the following encoding procedure is applied: first the k_b rows of the $k_b \times k_a$ information block are encoded k_b times by the code $\mathcal{A}$. Afterwards the n_a columns are encoded n_a times by $\mathcal{B}$ (see Figure 2.1). The second encoding procedure works as follows: first, the k_a columns are encoded k_a times by $\mathcal{B}$, then the obtained n_b rows are encoded n_b times by $\mathcal{A}$. The generator matrix for the second encoding procedure is $\mathbf{G}^{\langle a \rangle} \otimes \mathbf{G}^{\langle b \rangle}$.

Theorem 2.20 (Distance of a Product Code)
Let two $[n_a, k_a, d_a]_q$ and $[n_b, k_b, d_b]_q$ codes $\mathcal{A}$ and $\mathcal{B}$ be given. The product code $\mathcal{A} \otimes \mathcal{B}$ over $\mathbb{F}_q$ as in Definition 2.19 has minimum distance $d_a d_b$.

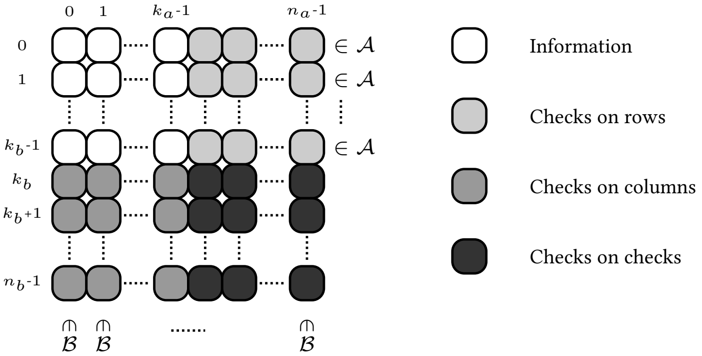


Figure 2.1: Illustration of an $[n_a n_b, k_a k_b, d_a d_b]_q$ product code $\mathcal{A} \otimes \mathcal{B}$. First the k_b rows are encoded by an $[n_a, k_a, d_a]_q$ code $\mathcal{A}$ and afterwards the n_a columns by an $[n_b, k_b, d_b]_q$ code $\mathcal{B}$.

Proof Let us w.l.o.g. assume that the first k_b rows were encoded with $\mathcal{A}$ and afterwards the n_a columns are encoded with $\mathcal{B}$. Then, each of the first k_b non-zero rows of the corresponding $n_a \times n_b$ matrix has weight at least d_a. After encoding with $\mathcal{B}$, each non-zero column has weight at least d_b and therefore the minimum Hamming distance of the product code is greater than or equal to $d_a d_b$. To prove the equality, we have to show that a codeword with (exactly) weight $d_a d_b$ exists. Let $\mathbf{c}_a \in \mathcal{A}$ and $\mathbf{c}_b \in \mathcal{B}$ be two codewords with weight d_a and d_b respectively. Then $\mathbf{c}_a^T \mathbf{c}_b \in \mathbb{F}_q^{n_b \times n_a}$ is a codeword of the product code $\mathcal{A} \otimes \mathcal{B}$ and has weight $d_a d_b$. ■

The simplest two-step decoding procedure decodes first the rows (or columns) separately and afterwards the columns (or rows). Clearly, the second step fails, if a decoding failure (see Chapter 3 for definition) occurred in the first one. There exist error patterns with weight less than $\lfloor (d_a d_b - 1)/2 \rfloor$ that cannot be corrected by this two-step approach (see one of the first works for decoding product codes [A-Abr68]). Product codes are suited to be decoded by iterative methods and a variety of literature exists on it (see e.g., [O-Ksc03]).

2.3.2 Cyclic Product Codes

Burton and Weldon [A-BW65] considered cyclic product codes first. Their work was extended by and Lin and Weldon [A-LW70]. We recall some basic properties of [A-BW65; A-LW70] in the following and give an example.

> **Theorem 2.21 (Cyclic Product Code [A-BW65, Theorem I])**
> Let $\mathcal{A}$ be an $[n_a, k_a, d_a]_q$ cyclic code and let $\mathcal{B}$ be an $[n_b, k_b, d_b]_q$ cyclic code. The product code $\mathcal{C} = \mathcal{A} \otimes \mathcal{B}$ is an $[n_a n_b, k_a k_b, d_a d_b]_q$ cyclic code provided that the two lengths n_a and n_b are relatively prime. Let the $n_b \times n_a$ matrix $\mathbf{M} = (M_{i,j})_{i \in [n_b)}^{j \in [n_a)}$ be a codeword of $\mathcal{C}$ (as in Definition 2.19). Then, the polynomial $c(X) = \sum_{i=0}^{n_a n_b - 1} c_i X^i \in \mathbb{F}_q[X]$ with
> $$c_i = M_{i \bmod n_b, i \bmod n_a}, \quad \forall i \in [n_a n_b)$$
> is a codeword of the cyclic product code $\mathcal{C}$, which is an ideal in the ring $\mathbb{F}_q[X]/(X^{n_a n_b} - 1)$.

Let us outline how the defining set $D_{\mathcal{C}}$ of $\mathcal{C} = \mathcal{A} \otimes \mathcal{B}$ can be obtained from defining set $D_{\mathcal{A}}$ and $D_{\mathcal{B}}$ of its component codes $\mathcal{A}$ and $\mathcal{B}$.

Theorem 2.22 (Defining Set of a Cyclic Product Code, [A-LW70, Theorem 4])
Let $\mathcal{A}$ and $\mathcal{B}$ be an $[n_a, k_a, d_a]_q$ respectively an $[n_b, k_b, d_b]_q$ cyclic code with defining sets $D_{\mathcal{A}}$ and $D_{\mathcal{B}}$ and generator polynomials $g_a(X)$ and $g_b(X)$. For some integers u and v, let $un_a + vn_b = 1$. Then, the generator polynomial $g(X)$ of the cyclic product code $\mathcal{A} \otimes \mathcal{B}$ is:

$$g(X) = \gcd\Big(X^{n_a n_b} - 1, g_a(X^{vn_b}) \cdot g_b(X^{un_a})\Big). \tag{2.22}$$

Let $B_{\mathcal{A}} \overset{\text{def}}{=} (D_{\mathcal{A}} \cdot v)_{n_a}$ and let $A_{\mathcal{B}} \overset{\text{def}}{=} (D_{\mathcal{B}} \cdot u)_{n_b}$ as defined in (2.14). The defining set of the cyclic product code $\mathcal{C}$ is:

$$D_{\mathcal{C}} = \left\{ \bigcup_{i=0}^{n_b-1} (B_{\mathcal{A}} + in_a) \right\} \cup \left\{ \bigcup_{i=0}^{n_a-1} (A_{\mathcal{B}} + in_b) \right\}.$$

Let us consider an example of a binary cyclic product code.

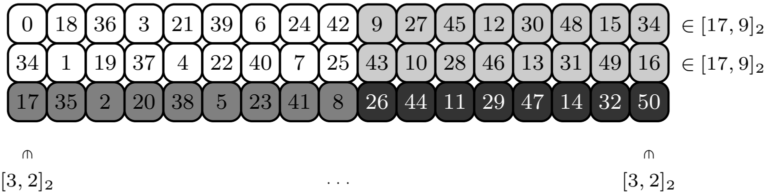
Example 2.23 (Cyclic Product Code)
Let $\mathcal{A}$ be the binary $[17, 9, 5]_2$ cyclic code with defining set

$$\begin{aligned} D_{\mathcal{A}} = M_{17,2}^{\langle 3 \rangle} &= \{-8 \cdot 3, -4 \cdot 3, -2 \cdot 3, -1 \cdot 3, 1 \cdot 3, 2 \cdot 3, 4 \cdot 3, 8 \cdot 3\} \\ &= \{1, 2, 4, 8, 9, 13, 15, 16\}. \end{aligned}$$

Let $\mathcal{B}$ be the binary $[3, 2, 2]_2$ single-parity check code with defining set $D_{\mathcal{B}} = M_{3,2}^{\langle 0 \rangle} = \{0\}$ and let

$$\underbrace{-1}_{u} \cdot 17 + \underbrace{6}_{v} \cdot 3 = 1$$

be a given Bézout's relation. The binary product code $\mathcal{A}(M_{17,2}^{\langle 3 \rangle}) \otimes \mathcal{B}(M_{3,2}^{\langle 0 \rangle})$ is illustrated in the following figure. The numbers in the symbols are the indexes of the coefficients c_i of the univariate polynomial $c(X)$ of the $[51, 18, 10]_2$ cyclic product code as stated in Theorem 2.21. As previously discussed, we encode first the two rows by the $[17, 9, 5]_2$ code $\mathcal{A}$ and afterwards the columns by the binary $[3, 2, 2]_2$ single-parity check code $\mathcal{B}$.



0	18	36	3	21	39	6	24	42	9	27	45	12	30	48	15	34	$\in [17, 9]_2$
34	1	19	37	4	22	40	7	25	43	10	28	46	13	31	49	16	$\in [17, 9]_2$
17	35	2	20	38	5	23	41	8	26	44	11	29	47	14	32	50	

$\rotatebox[origin=c]{90}{$\in$}$
$[3, 2]_2$
$\cdots$
$\rotatebox[origin=c]{90}{$\in$}$
$[3, 2]_2$

According to Theorem 2.22, we obtain the following defining sets:

$$\begin{aligned} B_{\mathcal{A}} &= (D_{\mathcal{A}} \cdot 6)_{17} = \{-8,-4,-2,-1,1,2,4,8\} = \{1,2,4,8,9,13,15,16\}, \\ A_{\mathcal{B}} &= (D_{\mathcal{B}} \cdot -1)_3 = \{0\}. \end{aligned}$$

A subset of the unions of the sets $B_{\mathcal{A}}$ and $A_{\mathcal{B}}$ is shown in the following table.

$(B_{\mathcal{A}} + 2 \cdot 17) \cup B_{\mathcal{A}}$	..	□	□	13	□	15	16	□	1	2	□	4	□	..
$\cup_{i \in \{15,16,1,2\}} (A_{\mathcal{B}} + 3i)$	..	0	□	□	0	□	□	0	□	□	0	□	□	..
$D_{\mathcal{A}\otimes\mathcal{B}}$	..	45	□	47	48	49	50	0	1	2	3	4	□	..

The corresponding subset of the defining set of the $[51,18,10]_2$ cyclic product code $D_{\mathcal{A}\otimes\mathcal{B}}$ is shown in the third row. The dashed line indicates the start of a defining set of the $[3,2,2]_2$ single-parity check code $\mathcal{B}$. The sequence $47, 48, \dots, 4$ in the defining set $D_{\mathcal{A}\otimes\mathcal{B}}$ of the cyclic product has length nine and is the longest consecutive one.

According to the BCH bound, the minimum distance of the cyclic product codes is then at least ten, which is the true minimum distance of the product code. Notice that the BCH bound is not tight for the minimum distance of $\mathcal{A}(M_{17,2}^{\langle 3 \rangle})$ and gives four.

2.3.3 Generalized Concatenated Codes

In 1966, Forney introduced concatenated block codes in [O-For66a, Section I.2]. A so-called outer code is concatenated with an inner code. Furthermore, the model of the super channel, i.e., the concatenation of the inner encoder, the channel and the inner decoder (see Figure 2.2), was proposed. Basically, a concatenated code is a product code $\mathcal{A} \otimes \mathcal{B}$, but it is assumed that the $[n_a, k_a, d_a]_{q^l}$ code $\mathcal{A}$—the row-code —is a code over an extension field $\mathbb{F}_{q^l}$ and the $[n_b, k_b, d_b]_q$ code $\mathcal{B}$—the column-code—is over

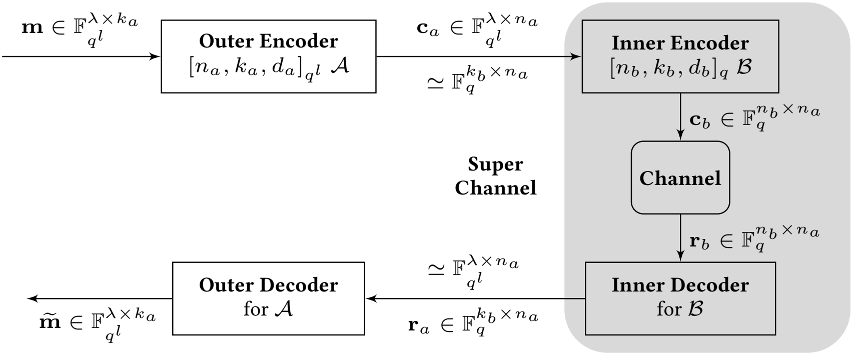

$\mathbf{m} \in \mathbb{F}_{q^l}^{\lambda \times k_a}$
Outer Encoder
$[n_a, k_a, d_a]_{q^l}$ $\mathcal{A}$
$\mathbf{c}_a \in \mathbb{F}_{q^l}^{\lambda \times n_a}$
$\simeq \mathbb{F}_q^{k_b \times n_a}$
Inner Encoder
$[n_b, k_b, d_b]_q$ $\mathcal{B}$
$\mathbf{c}_b \in \mathbb{F}_q^{n_b \times n_a}$
Super Channel
Channel
$\mathbf{r}_b \in \mathbb{F}_q^{n_b \times n_a}$
$\simeq \mathbb{F}_{q^l}^{\lambda \times n_a}$
Outer Decoder
for $\mathcal{A}$
Inner Decoder
for $\mathcal{B}$
$\widetilde{\mathbf{m}} \in \mathbb{F}_{q^l}^{\lambda \times k_a}$
$\mathbf{r}_a \in \mathbb{F}_q^{k_b \times n_a}$

Figure 2.2: The super channel in a classic concatenated code scheme with an $[n_a, k_a, d_a]_{q^l}$ outer code $\mathcal{A}$ and an $[n_b, k_b, d_b]_q$ inner code $\mathcal{B}$ where $k_b = \lambda \cdot l$.

$\mathbb{F}_q$. Furthermore, we require that

$$k_b = \lambda \cdot l,$$

where $\lambda \geq 0$ is an integer (see Figure 2.2). In terms of the product code as in Figure 2.1, first the λ rows of length k_a over $\mathbb{F}_{q^l}$ are encoded by $\mathcal{A}$ and the obtained λ codewords $\mathbf{c}_{a,0}, \mathbf{c}_{a,1}, \dots, \mathbf{c}_{a,\lambda-1}$ are in $\mathbb{F}_{q^l}^{n_a}$. Each element $\mathbf{c}_{a,i}$ in $\mathbb{F}_{q^l}^{n_a}$ can be represented uniquely as $l \times n_a$ matrix over $\mathbb{F}_q$. The obtained $\lambda \times n_a$ code matrix

$$\mathbf{c}^{\langle a \rangle} = (\mathbf{c}_{a,0}\, \mathbf{c}_{a,1}\, \dots\, \mathbf{c}_{a,\lambda-1})$$

in $\mathbb{F}_{q^l}^{\lambda \times n_a}$ is equivalent to k_b vectors in $\mathbb{F}_q^{n_a}$. These vectors are then encoded column-wisely by $\mathcal{B}$, the inner code (or column code). The obtained n_b vectors $\mathbf{c}_{b,0}, \mathbf{c}_{b,1}, \dots, \mathbf{c}_{b,n_b-1} \in \mathbb{F}_q^{n_a}$ are transmitted over the channel, then decoded separately by an inner decoder for $\mathcal{B}$, represented as λ received vectors $\mathbf{r}_{a,0}, \mathbf{r}_{a,1}, \dots, \mathbf{r}_{a,\lambda-1} \in \mathbb{F}_{q^l}^{n_a}$ and decoded by an outer decoder for $\mathcal{A}$.

> **Definition 2.24 (Concatenated Code)**
> Let an $[n_a, k_a, d_a]_{q^l}$ code $\mathcal{A}$ over $\mathbb{F}_{q^l}$ with a $k_a \times n_a$ generator matrix $\mathbf{G}^{\langle a \rangle}$ and an $[n_b, k_b, d_b]_q$ code $\mathcal{B}$ over $\mathbb{F}_q$ with a $k_b \times n_b$ generator matrix $\mathbf{G}^{\langle b \rangle}$ be given. Let $k_b = \lambda l$. The $k_a l \times n_a$ matrix $\overline{\mathbf{G}}^{\langle a \rangle}$ is the $k_a \times n_a$ matrix $\mathbf{G}^{\langle a \rangle}$ represented over $\mathbb{F}_q$. The code with generator matrix
>
> $$\mathbf{G} = \mathbf{G}^{\langle b \rangle} \otimes \overline{\mathbf{G}}^{\langle a \rangle} \tag{2.23}$$
>
> is an $[n_a n_b, k_a \lambda l, d \geq d_a d_b]_q$ concatenated code and denoted by $\overline{\mathcal{A}} \otimes \mathcal{B}$.

The mapping

$$\begin{array}{rccl} \text{enc-cc:} & \mathbb{F}_q^{k_a \lambda l} & \rightarrow & \mathbb{F}_q^{n_a n_b} \\ & \mathbf{m} & \mapsto & \text{enc-cc}(\mathbf{m}) = \mathbf{m}\left(\mathbf{G}^{\langle b \rangle} \otimes \overline{\mathbf{G}}^{\langle a \rangle}\right) \end{array}$$

defines the encoding of a concatenated code.

Equivalently, we can represent the row-code $\mathcal{B}$ over the extension field $\mathbb{F}_{q^l}$ and first encode column-wisely and then row-wisely by $\mathcal{A}$.

> **Theorem 2.25 (Minimum Distance of a Concatenated Code)**
> Let $\mathcal{A}$ denote the $[n_a, k_a, d_a]_{q^l}$ row code and $\mathcal{B}$ denote the $[n_b, k_b, d_b]_q$ column code of a concatenated code $\overline{\mathcal{A}} \otimes \mathcal{B}$ as in Definition 2.24. The minimum distance of $\overline{\mathcal{A}} \otimes \mathcal{B}$ is:
>
> $$d \geq d_a d_b.$$

Proof The proof is similar to the first part of the proof of Theorem 2.20. Two different codewords of $\mathcal{A}$ differ in at least d_a positions leading to at least d_b different symbols of the column code $\mathcal{B}$. ■

The counterpart concept of the super channel considers a concatenation of the outer and inner encoder as one element—the super encoder—and the concatenation of the inner and outer decoder as super decoder respectively.

Forney's code concatenation [O-For66a] was generalized by Blokh and Zyablov [A-BZ74; A-ZSB99] and a non-linear construction was proposed by Zinoviev [A-Zin76]. Generalized code concatenation is also referred to as multilevel code concatenation. Dumer gives an introduction in his chapter in the Handbook of Coding Theory [O-Dum98]. In addition, Lin and Costello [B-LC04, Chapter 15] and Bossert [B-Bos13, Chapter 9] deal with generalized code concatenation.

We give the definition of a generalized concatenated code, because we use it in Chapter 6 to bound the minimum distance of cyclic codes.

Definition 2.26 (Generalized Concatenated Code [A-Zin76])
Let s outer (or row) $[n_a, k_{a,i}, d_{a,i}]_{q^{l_i}}$ codes $\mathcal{A}_0, \mathcal{A}_1, \dots, \mathcal{A}_{s-1}$ over $\mathbb{F}_{q^{l_i}}$ with $k_{a,i} \times n_a$ generator matrices $\mathbf{G}^{\langle a_i \rangle}$ for all $i \in [s)$ be given. Let $\mathcal{B}_0, \mathcal{B}_1, \dots, \mathcal{B}_{s-1}$ denote s inner (or column) $[n_b, k_{b,i}, d_{b,i}]_q$ codes over $\mathbb{F}_q$ for all $i \in [s)$. Furthermore let

$$\mathcal{B}_0 \supset \mathcal{B}_1 \supset \dots \supset \mathcal{B}_{s-1}$$

as in Corollary 2.15 hold. Let $\mathbf{G}^{\langle b_i \backslash b_{i+1} \rangle}$ denote the $(k_{b,i} - k_{b,i+1}) \times n$ generator matrix of the code $\mathcal{B}_i \backslash \mathcal{B}_{i+1}$ code for all $i \in [s-1)$. Let the dimensions $k_{b,i} - k_{b,i+1} = \lambda_i l_i$ for all $i \in [s-1)$ and $k_{b,s-1} = \lambda_{s-1} l_{s-1}$.

The $k_{a,i} l_i \times n_a$ matrices $\overline{\mathbf{G}}^{\langle a_i \rangle}$ are the corresponding representations of $\mathbf{G}^{\langle a_i \rangle}$ in the base field $\mathbb{F}_q$. Then, the code with generator matrix:

$$\mathbf{G} = \begin{pmatrix} \mathbf{G}^{\langle b_0 \backslash b_1 \rangle} \otimes \overline{\mathbf{G}}^{\langle a_0 \rangle} \\ \mathbf{G}^{\langle b_1 \backslash b_2 \rangle} \otimes \overline{\mathbf{G}}^{\langle a_1 \rangle} \\ \vdots \\ \mathbf{G}^{\langle b_{s-2} \backslash b_{s-1} \rangle} \otimes \overline{\mathbf{G}}^{\langle a_{s-2} \rangle} \\ \mathbf{G}^{\langle b_{s-1} \rangle} \otimes \overline{\mathbf{G}}^{\langle a_{s-1} \rangle} \end{pmatrix} \tag{2.24}$$

is an $[n_a n_b, \sum_{i=0}^{s-1} k_{a,i} \lambda_i l_i]_q$ generalized concatenated code of order s denoted by $\left(\bigoplus_{i=0}^{s-2} \left(\overline{\mathcal{A}}_i \otimes (\mathcal{B}_i \backslash \mathcal{B}_{i+1}) \right) \right) \oplus (\overline{\mathcal{A}}_{s-1} \otimes \mathcal{B}_{s-1})$.

The mapping with $\mathbf{G}$ as in (2.24):

$$\begin{aligned} \text{enc-gcc:} \quad \mathbb{F}_q^{\sum_{i=0}^{s-1} k_{a,i} \lambda_i l_i} &\to \mathbb{F}_q^{n_a n_b} \\ \mathbf{m} &\mapsto \text{enc-gcc}(\mathbf{m}) = \mathbf{m}\mathbf{G}, \end{aligned}$$

defines the encoding of a generalized concatenated code. Similar to the code concatenation, every i-th sub-code can be equivalently formed by representing the generator matrix $\mathbf{G}^{\langle b_i \backslash b_{i+1} \rangle}$ over the corresponding extension field $\mathbb{F}_{q^{l_i}}$ and building the product code first column-wisely and then row-wisely.

Theorem 2.27 (Minimum Distance of a Generalized Concatenated Code)
Let $\mathcal{A}_0, \mathcal{A}_1, \dots, \mathcal{A}_{s-1}$ be s $[n_a, k_{a,i}, d_{a,i}]_{q^{l_i}}$ inner (or row) codes and let $\mathcal{B}_0, \mathcal{B}_1, \dots, \mathcal{B}_{s-1}$ be s $[n_b, k_{b,i}, d_{b,i}]_q$ outer (or column) with

$$\mathcal{B}_0 \supset \mathcal{B}_1 \supset \dots \supset \mathcal{B}_{s-1}.$$

Let the generalized concatenated code

$$\mathcal{C} = \left(\bigoplus_{i=0}^{s-2} \left(\overline{\mathcal{A}}_i \otimes (\mathcal{B}_i \backslash \mathcal{B}_{i+1}) \right) \right) \oplus \left(\overline{\mathcal{A}}_{s-1} \otimes \mathcal{B}_{s-1} \right)$$

as in Definition 2.26. The minimum distance of $\mathcal{C}$ is:

$$d \geq \min_{i \in [s)} \left(d_{a,i} \cdot d_{b,i} \right)$$

Proof A codeword $\mathbf{a}_i$ of $\mathcal{A}_i$ with minimal Hamming weight $d_{a,i}$ affects a sub-code $\mathcal{B}_{i+1}$ of $\mathcal{B}_i$ having at least weight $d_{b,i}$.

We refer to [A-ZSB99; O-BGMZ99] and [O-Gri02, Chapter 2] for further information on generalized concatenated codes and their decoding.

2.4 Generalized Reed–Solomon Codes

2.4.1 Definition and Notation

Delsarte [A-Del75] introduced Generalized Reed–Solomon codes 15 years after their initial definition by Reed and Solomon in [A-RS60]. They are extensively described in [B-MS88a, Chapter 10.8], [B-Bos98; B-Bos99, Chapter 3.1] and [B-Rot06, Chapter 5] as well as in [O-PHB98a, Section 8] and [O-Huf98, Section 2].

Let $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ denote $n < q$ non-zero[1] distinct elements of the finite field $\mathbb{F}_q$ and let

$$\boldsymbol{\alpha} = (\alpha_0 \, \alpha_1 \, \dots \, \alpha_{n-1}).$$

Let

$$\overline{\boldsymbol{v}} = (\overline{v}_0 \, \overline{v}_1 \, \dots \, \overline{v}_{n-1})$$

contain n non-zero (not necessarily distinct) elements of $\mathbb{F}_q$. For some univariate polynomial $f(X) \in \mathbb{F}_q[X]$, let

$$\begin{aligned} \text{eval:} \quad \mathbb{F}_q[X] &\to \mathbb{F}_q^n \\ f(X) &\mapsto \text{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha}) = \left(\overline{v}_0 f(\alpha_0) \;\; \overline{v}_1 f(\alpha_1) \;\; \dots \;\; \overline{v}_{n-1} f(\alpha_{n-1}) \right) \end{aligned} \tag{2.25}$$

denote the evaluation of $f(X)$ at all n points α_i scaled by $\overline{v}_i$.

Definition 2.28 (Generalized RS (GRS) Code)
Let $\boldsymbol{\alpha} = (\alpha_0 \, \alpha_1 \, \dots \, \alpha_{n-1})$ consist of n distinct non-zero elements in $\mathbb{F}_q$ with $n < q$ and let $\overline{\boldsymbol{v}} = (\overline{v}_0 \, \overline{v}_1 \, \dots \, \overline{v}_{n-1})$ consist of n non-zero elements in $\mathbb{F}_q$. An $[n, k]_q$ GRS code is given by:

$$\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k) \overset{\text{def}}{=} \left\{ \text{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha}) : f(X) \in \mathbb{F}_q[X] \text{ and } \deg f(X) < k \right\}. \tag{2.26}$$

[1] We restrict ourselves to this case, but in general the set of code locators can contain the zero-element.

We denote $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ in the sense of a function prototype. The vectors $\overline{\boldsymbol{v}}$ and $\boldsymbol{\alpha}$ give indirectly the length n of the code and the field size q. Therefore, the parameter n and q are not explicitly given.

The elements $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ are also called code locators or support of an GRS code. GRS codes are Maximum Distance Separable (MDS) codes, i.e., their minimum Hamming distance is $d = n - k + 1$. Therefore, we give only the length n and dimension k as tuple $[n, k]_q$ in the context of GRS codes (instead of $[n, k, d]_q$).

The generator matrix $\mathbf{G} \in \mathbb{F}_q^{k \times n}$ of an $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ is:

$$\mathbf{G} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_0 & \alpha_1 & \cdots & \alpha_{n-1} \\ \alpha_0^2 & \alpha_1^2 & \cdots & \alpha_{n-1}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_0^{k-1} & \alpha_1^{k-1} & \cdots & \alpha_{n-1}^{k-1} \end{pmatrix} \cdot \begin{pmatrix} \overline{v}_0 & & \\ & \overline{v}_1 & & 0 \\ 0 & & \ddots & \\ & & & \overline{v}_{n-1} \end{pmatrix}, \tag{2.27}$$

and the parity-check matrix $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ is given by:

$$\mathbf{H} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_0 & \alpha_1 & \cdots & \alpha_{n-1} \\ \alpha_0^2 & \alpha_1^2 & \cdots & \alpha_{n-1}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_0^{n-k-1} & \alpha_1^{n-k-1} & \cdots & \alpha_{n-1}^{n-k-1} \end{pmatrix} \cdot \begin{pmatrix} v_0 & & \\ & v_1 & & 0 \\ 0 & & \ddots & \\ & & & v_{n-1} \end{pmatrix}. \tag{2.28}$$

The elements $v_0, v_1, \dots, v_{n-1}$ of the parity-check matrix in (2.28) are the so-called column multipliers of the GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$. Since $\mathbf{G}\mathbf{H}^T = \mathbf{0}$, we can relate the column multipliers $v_0, v_1, \dots, v_{n-1}$ and the column multipliers $\overline{v}_0, \overline{v}_1, \dots, \overline{v}_{n-1}$ of the dual $[n, n-k]_q$ GRS code $\mathcal{GRS}(\boldsymbol{v}, \boldsymbol{\alpha}, n-k)^\perp$ in the following lemma.

Lemma 2.29 (Column Multipliers of the Dual GRS Code)
Let $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ be an $[n, k]_q$ GRS code. The dual (see Definition 2.12) of $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ is an $[n, n-k]_q$ GRS code

$$\mathcal{GRS}(\boldsymbol{v}, \boldsymbol{\alpha}, n-k) = \mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)^\perp$$

with

$$v_i^{-1} = \overline{v}_i L_i(\alpha_i), \quad \forall i \in [n), \tag{2.29}$$

where $L_i(X)$ is as given in (2.2).

Proof The proof follows an idea of Huffman's Chapter 17 in the Handbook of Coding Theory [O-Huf98] and in the lecture notes of Hall [O-Hal12, Chapter 5].

Let $\mathbf{c} = \text{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ be a codeword of the given $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ code and let $\overline{\mathbf{c}} = \text{eval}(\overline{f}(X), \boldsymbol{v}, \boldsymbol{\alpha})$ be a codeword of the dual $[n, n-k]_q$ GRS code $\mathcal{GRS}(\boldsymbol{v}, \boldsymbol{\alpha}, n-k)$ as in Definition 2.12. The polynomial $f(X)$ has degree less than k and $\overline{f}(X)$ has degree less than $n-k$. Therefore, the product $f(X)\overline{f}(X)$ has degree at most $n-2$. The Lagrange interpolation formula (2.3) for $f(X)\overline{f}(X)$ gives:

$$\sum_{i=0}^{n-1} f(\alpha_i)\overline{f}(\alpha_i) \frac{L_i(X)}{L_i(\alpha_i)} = f(X)\overline{f}(X). \tag{2.30}$$

We consider the (highest) coefficient of X^{n-1} and for the LHS of (2.30) we have:

$$\begin{aligned}\sum_{i=0}^{n-1} f(\alpha_i)\overline{f}(\alpha_i)\frac{1}{L_i(\alpha_i)} &= \sum_{i=0}^{n-1} \overline{\upsilon}_i f(\alpha_i)\overline{f}(\alpha_i)\frac{1}{\overline{\upsilon}_i L_i(\alpha_i)}\\ &= \sum_{i=0}^{n-1} \overline{\upsilon}_i f(\alpha_i)\overline{f}(\alpha_i)\upsilon_i\\ &= \mathbf{c}\cdot\overline{\mathbf{c}}^T.\end{aligned}$$

The RHS of (2.30) gives zero and therefore $\mathbf{c}\cdot\overline{\mathbf{c}}^T$ is zero, too. The condition of duality is fulfilled (see (2.6) in the Definition 2.12 of a dual code). ■

We consider some special classes of GRS codes in the following example.

Example 2.30 (Dual of a Primitive GRS Code)
Let $\overline{\boldsymbol{\upsilon}} = (\overline{\upsilon}_0\,\overline{\upsilon}_1\,\ldots\,\overline{\upsilon}_{n-1})$ and let $\boldsymbol{\alpha} = (\alpha_0\,\alpha_1\,\ldots\,\alpha_{n-1})$ be the code locators of an $[n = q-1, k]_q$ primitive GRS code $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$ over $\mathbb{F}_q$. Let α be a primitive element in $\mathbb{F}_q$ and let

$$\alpha_i = \alpha^i, \quad \forall i \in [n).$$

We need to calculate explicitly $L_i(\alpha_i) = L_i(\alpha^i)$ as given in (2.29). We obtain from (2.1) and (2.2):

$$L_i(X) = \frac{L(X)}{X-\alpha_i} = \frac{X^n - 1}{X-\alpha_i}.$$

Applying L'Hôpital's rule leads to:

$$L_i(\alpha^i) = \frac{n\alpha^{i(n-1)}}{1} = n\alpha^{-i}.$$

And for $n = q-1$ we get:

$$L_i(\alpha^i) = n\alpha^{-i} = -\alpha^{-i}.$$

Then, the column multipliers of $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$ are $\upsilon_i = -\alpha^i/\overline{\upsilon}_i$. It is common to set $\upsilon_i = \alpha^i/\overline{\upsilon}_i$ without the factor -1.

We define normalized GRS codes in the following.

Definition 2.31 (Normalized GRS Code)
Let the support set $\boldsymbol{\alpha} = (\alpha_0\,\alpha_1\,\ldots\,\alpha_{n-1})$ consist of n distinct non-zero elements in $\mathbb{F}_q$ with $n < q$ and let $\mathbf{1} \in \mathbb{F}_q^n$ denote the all-one vector. An $[n,k]_q$ normalized GRS code is denoted by $\mathcal{RS}(\boldsymbol{\alpha}, k)$ and defined by $\mathcal{RS}(\boldsymbol{\alpha}, k) \overset{\text{def}}{=} \mathcal{GRS}(\mathbf{1}, \boldsymbol{\alpha}, k)$. More explicitly:

$$\mathcal{RS}(\boldsymbol{\alpha}, k) \overset{\text{def}}{=} \Big\{\text{eval}(f(X), \mathbf{1}, \boldsymbol{\alpha}) : f(X) \in \mathbb{F}_q[X] \text{ and } \deg f(X) < k\Big\}. \tag{2.31}$$

Let us consider the dual of a normalized GRS code.

Example 2.32 (Dual of a Primitive Normalized RS Code)
Let $\mathcal{RS}(\boldsymbol{\alpha}, k)$ be an $[n = q-1, k]_q$ primitive RS code over $\mathbb{F}_q$ with support set $\boldsymbol{\alpha}$. Furthermore, we have $\alpha_i = \alpha^i, \ \forall i = [n)$. The column multipliers of the dual code of $\mathcal{RS}(\boldsymbol{\alpha}, k)$ are:

$$\upsilon_i = -\alpha^i, \quad \forall i \in [n).$$

This follows directly from Example 2.30. Notice that the dual of a primitive normalized RS code is not necessarily a normalized RS code.

We consider cyclic RS codes *inter alia* in Chapter 6 and therefore define them in the following.

Definition 2.33 (Conventional/Cyclic RS Code)
Let n be a positive integer with $n|(q-1)$ and let β denote the primitive element of $\mathbb{F}_q$. Then, the element:

$$\alpha = \beta^{\frac{q-1}{n}}$$

is an n-th root of unity of $\mathbb{F}_q$. Let b be a positive integer. Then, an $[n, k]_q$ GRS code with support $\boldsymbol{\alpha} = (\alpha_0 \, \alpha_1 \, \dots \, \alpha_{n-1})$, where

$$\alpha_i = \alpha^i, \quad \forall i \in [n),$$

and with column multipliers $\overline{\boldsymbol{\upsilon}} = (\overline{\upsilon}_0 \, \overline{\upsilon}_1 \, \dots \, \overline{\upsilon}_{n-1})$, where

$$\overline{\upsilon}_i = \alpha^{i(1-b)}, \quad \forall i \in [n)$$

is called cyclic RS code and therefore denoted by $\mathcal{CRS}(q, n, b, k)$. More explicitly, we have:

$$\mathcal{CRS}(q, n, b, k) \overset{\text{def}}{=} \Big\{ \text{eval}(f(X), \boldsymbol{\alpha}, \overline{\boldsymbol{\upsilon}}) : f(X) \in \mathbb{F}_q[X], \deg f(X) < k \Big\}.$$

We denote $\mathcal{CRS}(q, n, b, k)$ again in the sense of a function prototype. In contrast to GRS codes, the field size q, the length n is not given by the set of locators. The characteristic parameter b is given explicitly.

Let us investigate the cyclic property of a CRS code. According to (2.29), we obtain the following column multipliers:

$$\upsilon_i = \frac{1}{\upsilon_i^{-1} L_i(\alpha_i)} = \frac{1}{n} \frac{\alpha^i}{\alpha^{i(1-b)}} = \frac{1}{n} \alpha^{ib}.$$

and therefore we have the following $(n-k) \times n$ parity-check matrix for an $[n, k]_q$ CRS code $\mathcal{CRS}(q, n, b, k)$:

$$\mathbf{H} = n^{-1} \cdot \begin{pmatrix} 1 & \alpha^b & \cdots & \alpha^{(n-1)b} \\ 1 & \alpha^{b+1} & \cdots & \alpha^{(n-1)(b+1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{b+n-k-1} & \cdots & \alpha^{(n-1)(b+n-k-1)} \end{pmatrix}. \tag{2.32}$$

We know that for all $\mathbf{c} \in \mathcal{CRS}(q, n, b, k)$:

$$\mathbf{H}\mathbf{c}^T = 0. \tag{2.33}$$

Let us associate for every codeword $\mathbf{c} = (c_0\, c_1\, \ldots\, c_{n-1}) \in \mathcal{CRS}(q, b, n, k)$ a univariate polynomial $c(X) \in \mathbb{F}_q[X]$ with $c(X) = \sum_{i=0}^{n-1} c_i X^i$. Due to the special form of $\mathbf{H}$ as in (2.32), we can rewrite (2.33) in polynomial form and obtain:

$$\mathbf{c} \in \mathcal{CRS}(q, n, b, k) \quad \Longleftrightarrow \quad c(\alpha^j) = 0, \quad \forall j \in [b, b+n-k).$$

and therefore, the element $\alpha^b, \alpha^{b+1}, \ldots, \alpha^{b+n-k-1}$ are the roots of the generator polynomial $g(X)$ of $\mathcal{CRS}(q, n, b, k)$. Our presentation of normalized and cyclic RS codes is very close to the one of Roth [B-Rot06, Chapter 5].

2.4.2 Interleaved Generalized Reed–Solomon Codes

Interleaved block codes over $\mathbb{F}_q$ of interleaving order s are a special case of product codes, where the column-code $\mathcal{B}$ is the trivial $[s, s, 1]_q$ code. We focus on Interleaved RS (IRS) codes. The existing literature of Krachkovsky [A-Kra97; I-Kra98; A-Kra03], Bleichenbacher [O-BKY03; A-BKY07] and Schmidt *et al.* [A-SSB09] considers interleaved normalized RS codes (as in Definition 2.31). Analog to them, we introduce Interleaved GRS (IGRS) codes. We focus on IGRS codes, where each sub-code has the same support.

Definition 2.34 (Interleaved GRS (IGRS) Code)
Let $\mathbf{k} = (k_0\, k_1\, \ldots\, k_{s-1})$ consist of s integers, where all $k_i < n$. Let $\overline{\boldsymbol{v}} = (\overline{\boldsymbol{v}}_0\, \overline{\boldsymbol{v}}_1\, \ldots\, \overline{\boldsymbol{v}}_{s-1})$, where each $\overline{\boldsymbol{v}}_i$ contains $n < q$ non-zero (and not necessarily distinct) elements of $\mathbb{F}_q$ and let $\boldsymbol{\alpha}_0 = (\alpha_0\, \alpha_1\, \ldots\, \alpha_{n-1})$ of n distinct non-zero elements in $\mathbb{F}_q$ with $n < q$ be given. Let $\boldsymbol{\alpha}$ denote

$$\boldsymbol{\alpha} = \underbrace{(\boldsymbol{\alpha}_0\, \boldsymbol{\alpha}_0\, \ldots\, \boldsymbol{\alpha}_0)}_{s \text{ times}}.$$

Then, an $[sn, \sum_{i=0}^{s-1} k_i]_q$ IGRS code $\mathcal{IGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, \mathbf{k})$ of interleaving order s is given by:

$$\mathcal{IGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}_0, \mathbf{k}) = \mathcal{IGRSD}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, \mathbf{k}).$$

Again in the sense of a function prototype, the length n and the interleaving order s of the IGRS code are indirectly given by the parameters $\overline{\boldsymbol{v}} \in \mathbb{F}_q^{sn}$ and $\mathbf{k} \in \mathbb{N}^s$.

IGRS code are called heterogeneous in general and if $k_i = k, \ \forall i \in [s)$, they can be called homogeneous.

Furthermore, we note that it is possible to extend the Definition 2.34 to IGRS codes with different support sets.

3

"Basic research is like shooting an arrow into the air and, where it lands, painting a target."

Homer Burton Adkins (1892 - 1949)

Algebraic Decoding Principles for Linear Block Codes

The decoding of linear block codes is a difficult task in general, since a native decoding algorithm, i.e., an exhaustive search, has exponential time complexity. The definition of the decoding problem is not unified in the literature and many various formulations exists. In this chapter, we describe elementary decoding principles of linear block codes in Hamming metric. The refinement for algebraic block codes and in particular for GRS codes is outlined.

In a first step, we give relevant definitions for hard-decision decoding problems in Section 3.1. For each problem, we identify when a decoder fails. In the second step, the hard-decision decoding problem is generalized to the case where the channel, in addition to the received vector, outputs information on the reliability of the received symbol, the so-called soft-information.

Syndrome-based hard- and soft-decision decoding approaches for GRS codes are considered in Section 3.2. We show in Section 3.3, how the Extended Euclidean Algorithm (EEA), as originally modified by Sugiyama, Kasahara, Hirasawa and Namekawa [A-SKHN75; A-SKHN76] for Goppa codes, can be used for error/erasure decoding of GRS codes. The Fundamental Iterative Algorithm (FIA) of Feng and Tzeng [A-FT85; A-FT91a] finds the minimal number of linearly dependent columns (and the corresponding vanishing linear combination) of a given arbitrary matrix. The FIA can be suited to a structured matrix like a Hankel matrix. The homogeneous linear equations that originates from a so-called Key Equation, i.e., a polynomial equation that connects algebraically the input and the output of the unique decoding problem of GRS codes, is of Hankel structure. Therefore, we show the adjustment of the FIA and prove the complexity reducing initialization rule, which is generalized in Chapter 4 and 5. We illustrate the native and the adjusted FIA, when used for Bounded Minimum Distance (BMD) decoding of a GRS code.

In Section 3.4, collaborative decoding of IGRS codes, as defined in Subsection 2.4.2, is considered. We define the model of burst-errors and give the set of Key Equations for the collaborative decoding scheme. The definition of decoding failure is given and the corresponding homogeneous set of equations is outlined.

We prove the main theorem of the interpolation-based hard-decision list decoding of GRS codes by Guruswami and Sudan [A-Sud97; A-GS99] in Section 3.5. We show that the Johnson radius is achieved asymptotically. Furthermore, the extension of Kötter–Vardy [A-KV03a] to a soft-decision scenario is discussed. The comparison of existing realizations for the interpolation step concludes this chapter.

3.1 Decoding Principles and Complexity Issues

3.1.1 Hard-Decision Decoding of Linear Block Codes and Generalized Reed–Solomon Codes

Throughout this subsection, let $\mathcal{C} \subset \mathbb{F}_q^n$ be a linear $[n,k,d]_q$ block code and let $\mathbf{c} \in \mathcal{C}$. Let $\mathbf{e} \in \mathbb{F}_q^n$ denote the error with $\varepsilon = \mathrm{wt}(\mathbf{e})$. Let $\mathbf{r} = \mathbf{c} + \mathbf{e}$ denote the received word.

> **Definition 3.1 (Bounded Distance Decoder)**
> A bounded distance decoder for a given linear block code $\mathcal{C}$ is a function that returns one codeword or a decoding failure for a given decoding radius τ and a given received vector $\mathbf{r}$, i.e.:
>
> $$\begin{aligned} \text{BDD:} \quad (\mathbb{F}_q^n, \mathbb{N}) &\rightarrow \mathcal{C} \cup \{\textsc{Decoding Failure}\} \\ (\mathbf{r}, \tau) &\mapsto \mathrm{BDD}(\mathbf{r}, \tau). \end{aligned}$$
>
> The bounded distance decoder returns one codeword $\mathbf{c}$ from a given received vector if
>
> $$|\mathcal{B}_\tau(\mathbf{r}) \cap \mathcal{C}| = 1, \tag{3.1}$$
>
> where $\mathcal{B}_\tau(\mathbf{r})$ is the Hamming ball of radius τ around the vector $\mathbf{r}$, and otherwise it declares a decoding failure.

With Definition 3.1, we can easily define a Bounded Minimum Distance (BMD) decoder.

> **Definition 3.2 (Bounded Minimum Distance (BMD) Decoder)**
> A Bounded Minimum Distance (BMD) decoder for an $[n,k,d]_q$ block code is a bounded distance decoder as in Definition 3.1 with decoding radius $\tau = \lfloor (d-1)/2 \rfloor$.

If the number of errors ε is at most $\lfloor (d-1)/2 \rfloor$, a BMD decoder does not fail. For a bounded distance decoder with decoding radius $\tau > \lfloor (d-1)/2 \rfloor$ a decoding failure can occur if $\varepsilon > \lfloor (d-1)/2 \rfloor$ and therefore includes cases where the number of errors is smaller than the decoding radius.

For algebraic block codes, BMD decoding was first realized by syndrome-based decoding algorithms. The first polynomial-time approaches were proposed by Peterson [A-Pet60], Gorenstein–Zierler [A-GZ61] and Chien [A-Chi64]. Many efficient BMD decoding algorithms for GRS codes exist. The different steps for syndrome-based decoding have quadratic or even sub-quadratic time and space complexity.

We derive the Key Equation for syndrome-based BMD decoding of GRS codes as in Definition 3.2 from the simplest interpolation-based approach in Section 3.2. It is a special case of the derivation of the Key Equation for the Sudan principle by Roth and Ruckenstein [I-RR98; A-RR00] and resembles the ones of Welch–Berlekamp [O-WB86] and Gao [O-Gao03]. We consider the derivation for the case of erasures, which was not considered in [A-RR00].

A decoder capable to decode GRS codes beyond half the minimum distance is given in Section 4.1 and was first developed by Schmidt, Sidorenko and Bossert [I-SSB06; O-Sch07; A-SSB10]. It is not clear if it is a bounded distance decoder as in Definition 3.1 or if it declares in some cases a failure even if (3.1) is fulfilled. The Schmidt–Sidorenko–Bossert decoding approach is based on a virtual extension of a GRS code to an IGRS code. Therefore, we investigate first the collaborative decoding principle of IGRS codes in Section 3.4.

Definition 3.3 (List Decoder)
A list decoder for a linear block code $\mathcal{C}$ is a function that returns a list of up to ℓ codewords or a decoding failure for a given received vector $\mathbf{r}$, i.e.:

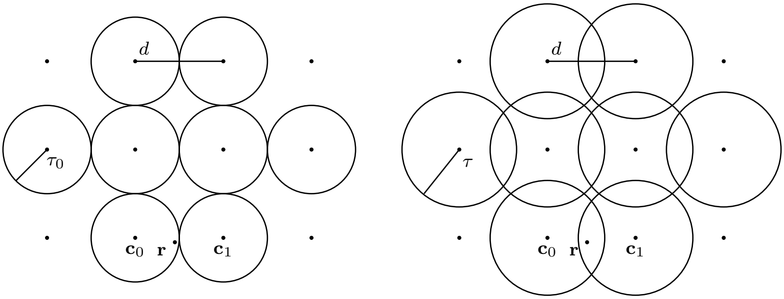
$$\begin{aligned} \text{LD:}\quad (\mathbb{F}_q^n, \mathbb{N}) &\rightarrow \{\mathcal{C} \cup \emptyset\}^{\ell} \setminus \{\emptyset\}^{\ell} \cup \{\textsc{Decoding Failure}\} \\ (\mathbf{r}, \ell) &\mapsto \text{LD}(\mathbf{r}, \ell). \end{aligned}$$

The decoding radius τ is such that

$$|\mathcal{B}_{\tau}(\mathbf{r}) \cap \mathcal{C}| \leq \ell,$$

where $\mathcal{B}_{\tau}(\mathbf{r})$ is the Hamming ball of radius τ around the vector $\mathbf{r}$. The list decoder returns at most ℓ codewords $\mathbf{c}_0, \mathbf{c}_1, \ldots, \mathbf{c}_{\ell-1} \in \mathcal{C}$ from a given received vector $\mathbf{r}$. If there is no codeword $\mathbf{c}$, such that $d(\mathbf{r}, \mathbf{c}) \leq \tau$ holds, i.e., the list is empty, a decoding failure is declared.

Similar to a bounded distance decoder, a list decoder with decoding radius $\tau > \lfloor (d-1)/2 \rfloor$ can return a decoding failure if $\varepsilon > \lfloor (d-1)/2 \rfloor$. The decoding spheres for BMD decoder and a decoder with higher decoding radius are shown in Figure 3.1.



(a) BMD decoding spheres for $\tau_0 \leq \lfloor (d-1)/2 \rfloor$ **(b)** Decoding spheres for $\tau > \lfloor (d-1)/2 \rfloor$

Figure 3.1: Comparison of the decoding spheres of BMD (Subfigure 3.1a) and a decoder with radius larger than $\lfloor (d-1)/2 \rfloor$ (Subfigure 3.1b). In the case of list decoding the illustrated received vector $\mathbf{r}$ can be mapped to the codewords $\mathbf{c}_0$ and $\mathbf{c}_1$.

The principle of list decoding was first considered in the work of Elias [O-Eli57] and Wozencraft [O-Woz58]. The first polynomial-time list decoder for GRS and Algebraic-Geometry codes was developed by Sudan [A-Sud97], Shokrollahi–Wasserman [A-SW99] and extended by Guruswami–Sudan [A-GS99]. We give an introduction to their interpolation-based principle in Section 3.5.

A nearest-codeword decoder returns the closest codeword, i.e., a codeword with smallest Hamming distance to the received word (and therefore generalizes the bounded distance decoder of Definition 3.1). The definition of a maximum likelihood decoder coincides with the one of a nearest-codeword decoder in Hamming metric for channels, where an error word with higher Hamming weight is less probable than one of lower Hamming weight.

Maximum likelihood decoding of linear codes, in general, and RS codes, in particular, is NP-hard [A-BMVT78; A-GV05]. It remains an open problem to find polynomial-time decoding algorithms with near maximum likelihood performance for GRS as well as for linear block codes.

3.1.2 Soft-Decision Decoding of Generalized Reed–Solomon Codes

Soft-decision decoding algorithms process side information to recover the transmitted codeword $\mathbf{c}$. The source of this additional information can be, e.g., the demodulator in a communication system or the inner decoder in concatenated coding schemes (see Subsection 2.3.3).

The first soft-decision algorithms for GRS codes used the available side information to map some received symbols in $\mathbb{F}_q$ to so-called erasures (see Figure 3.2 for a q-ary symmetric error-erasure channel). The position of an erasure is known but the value not. BMD decoders are able to correct ε errors and ζ erasures as long as

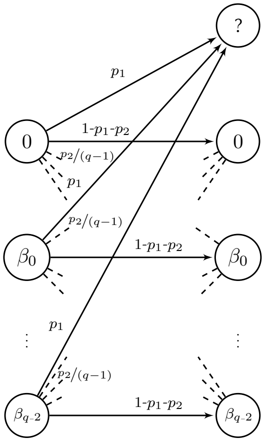


Figure 3.2: The q-ary symmetric error/erasure channel: The symbols of the q-ary alphabet are mapped to a $(q+1)$-ary alphabet with erasure probability p_1 and with error probability p_2.

$$2\varepsilon + \zeta < d.$$

The proof is quite simple: the ζ erased positions are neglected and the decoding of a punctured GRS code with minimum distance $d - \zeta$ is performed. Forney [A-For66b] first introduced a Generalized Minimum Distance (GMD) decoder that successively erases the least reliable symbols and performs for each step error/erasure decoding. The principle was among others refined by Chase [A-Cha72]. An overview of adaptive single- and multi-trial GMD decoding algorithms can be found in the work of Senger [O-Sen11]. We outline the syndrome-based algebraic error/erasure decoding of GRS codes in Section 3.2.

Another soft-decision principle for GRS codes uses the representation of $\mathbf{r} \in \mathbb{F}_{p^l}^n$ over the base field $\mathbb{F}_p$ and applies iterative algorithms (as, e.g., the belief propagation algorithm that is used originally for the decoding of Low-Density-Parity-Check codes). The initial work is from Jiang and Narayanan [A-JN04; A-JN06]. See also the work of Bellorado *et al.* [A-BKMP10; A-BK10] on this topic.

The interpolation-based decoding algorithm of Guruswami and Sudan [A-GS99] allows a new soft-decision variant of decoding GRS codes, the so-called Kötter–Vardy algorithm. We discuss the Kötter–Vardy principle in Section 3.5. Some simulation results can be found in [A-GKKG06] and [I-KV03b]. Furthermore, a variety of publications on an optimal weight calculation [I-Kö06; A-EKM06] and many suboptimal Kötter–Vardy-based algorithms exist (see e.g. [B-Che09; A-SLX12; I-NZ13]).

3.2 Syndrome-Based Decoding of Generalized Reed–Solomon Codes

3.2.1 Welch–Berlekamp Approach as List-One Decoder and Explicit Syndromes

In this subsection, we derive the classical Key Equation for syndrome-based error-only decoding of GRS codes up to half the minimum distance. The starting point of the derivation is the simplest interpolation-based approach, known as the Welch–Berlekamp algorithm (see [O-WB86], [A-GS92, Problem 9], [A-YB94, Section 2], [A-DB95, Section II]) or Gao algorithm [O-Gao03]. We consider the scenario where an $[n,k]_q$ GRS code as in Definition 2.28 is affected by errors (and not by erasures). The algorithm is based on the following lemma (see also [B-JH04, Chapter 5.2]).

> **Lemma 3.4 (Welch–Berlekamp Approach as a List-One Decoder)**
> Let $\mathbf{c} = \mathrm{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ be a codeword of a given $[n,k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$. Let $\mathbf{r} = (r_0\, r_1\, \dots\, r_{n-1}) = \mathbf{c} + \mathbf{e}$ be the received word with $\mathbf{e} \in \mathbb{F}_q^n$. Let $d(\mathbf{r}, \mathbf{c}) \leq \lfloor (n-k)/2 \rfloor$. Let
>
> $$Q(X,Y) = Q_0(X) + Q_1(X)Y,$$
>
> be a non-zero polynomial in $\mathbb{F}_q[X,Y]$ such that:
>
> C1) $Q(\alpha_i, r_i/\overline{v}_i) = 0, \quad \forall i \in [n]$,
>
> C2) $\deg Q_0(X) < n - \tau$,
> $\deg Q_1(X) < n - \tau - (k-1)$.
>
> Then $f(X) = -Q_0(X)/Q_1(X)$.

Proof We first proof the existence of a non-zero solution. From C2 we have $n - \tau + n - \tau - k + 1$ unknown coefficients of $Q(X,Y)$ and from C1 n linear constraints on $Q(X,Y)$. The system of equation has a non-zero solution if the number of unknowns is greater than the number of linear equations, i.e.:

$$\begin{aligned} 2(n-\tau) - (k-1) &> n \\ \tau &< \frac{n-k+1}{2}. \end{aligned}$$

Any interpolation polynomial $Q(X,Y)$ satisfies $Q(\alpha_i, c_i/\overline{v}_i) = 0$ for at least $n-\tau$ positions (due to C1). But $Q(X, f(X))$ has degree at most $n-\tau-1$ (due to C2), so $Q(X, f(X)) = 0$ and therefore $Q_0(X) + Q_1(X)f(X) = 0$. To prove the uniqueness of $f(X)$, let us consider a second interpolation polynomial $Q'(X,Y) = Q'_0(X) + Q'_1(X)Y$ that satisfies C1 and C2. We have:

$$Q_0(\alpha_i) + Q_1(\alpha_i)\frac{r_i}{\overline{v}_i} = 0, \tag{3.2}$$

$$Q'_0(\alpha_i) + Q'_1(\alpha_i)\frac{r_i}{\overline{v}_i} = 0, \tag{3.3}$$

for all $i \in [n]$. From (3.3), we have that $r_i/\overline{v}_i = -Q'_0(\alpha_i)/Q'_1(\alpha_i)$ and substitute this in (3.2) leads to:

$$Q_0(\alpha_i)Q'_1(\alpha_i) = Q_1(\alpha_i)Q'_0(\alpha_i),$$

for all $i \in [n)$. The degree of $Q_0(X)Q_1'(X)$ is at most $n-\tau-1+n-\tau-k = 2n-2\tau-k-1 = n-1$ and due to the Lagrange interpolation theorem (see Theorem 2.2) the polynomial is unique, hence $f(X) = -Q_0(X)/Q_1(X) = -Q_0'(X)/Q_1'(X)$. ■

Lemma 3.5 (Univariate Reformulation of Welch–Berlekamp)
Let $R(X) \in \mathbb{F}_q[X]$ with $\deg R(X) < n$ be the Lagrange interpolation polynomial, such that $R(\alpha_i) = r_i/\overline{\upsilon}_i,\ \forall i \in [n)$ holds (as in Theorem 2.2). Let $L(X) = \prod_{i=0}^{n-1}(X-\alpha_i)$ as in (2.1). Then, the interpolation polynomial $Q(X,Y)$ satisfies the conditions C1 and C2 of Lemma 3.4 if and only if there exists a polynomial $B(X) \in \mathbb{F}_q[X]$ such that:

$$Q(X, R(X)) = B(X) \cdot L(X), \tag{3.4}$$

where $\deg B(X) < n-k-\tau$ holds.

Proof From C1 of Lemma 3.4 the univariate polynomial $Q(X, R(X)) \in \mathbb{F}_q[X]$ vanishes at all n points α_i and thus $L(X)|Q(X,R(X))$. The degree of $Q(X,R(X))$ is at most $\deg Q_1(X) + \deg R(X) - \deg L(X) < n-\tau-k+1+n-1-n = n-k-\tau$. ■

We introduce the following polynomials:

$$\begin{aligned}
\overline{R}(X) &\overset{\text{def}}{=} X^{n-1}R(X^{-1}), \\
\overline{L}(X) &\overset{\text{def}}{=} X^{n}L(X^{-1}), \\
\Omega(X) &\overset{\text{def}}{=} X^{n-k-\tau-1}B(X^{-1}), \\
\Lambda_t(X) &\overset{\text{def}}{=} X^{n-\tau-t(k-1)-1}Q_t(X^{-1}), \quad t = 0, 1.
\end{aligned} \tag{3.5}$$

Note that, these polynomials are not necessarily the reciprocal polynomials, because, e.g., for the received polynomial the degree can be smaller than $n-1$. Inverting the order of the coefficients of (3.4) leads to:

$$X^{n-\tau+n-k-1}\left(Q_0(X^{-1}) + Q_1(X^{-1})R(X^{-1})\right) = X^{n-k-\tau-1}B(X^{-1})X^{n}L(X^{-1}).$$

Inserting the polynomials of (3.5), we obtain:

$$X^{n-k}\Lambda_0(X) + \Lambda_1(X)\cdot\overline{R}(X) = \Omega(X)\cdot\overline{L}(X).$$

We can consider the previous equation modulo X^{n-k} and obtain:

$$\Lambda_1(X)\cdot\overline{R}(X) \equiv \Omega(X)\cdot\overline{L}(X) \mod X^{n-k}. \tag{3.6}$$

The formal power series $S^{\infty}(X)$ is defined as follows:

$$S^{\infty}(X) \overset{\text{def}}{=} \sum_{i=0}^{\infty} S_i X^i = \frac{\overline{R}(X)}{\overline{L}(X)}. \tag{3.7}$$

In the following lemma, we give an explicit expression for the syndromes $S_i \in \mathbb{F}_q$.

Lemma 3.6 (Explicit Syndromes for GRS Codes)
Let $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ be an $[n,k]_q$ GRS code and let $\boldsymbol{v} = (v_0\, v_1\, \dots\, v_{n-1})$ denote its column multipliers as in Lemma (2.29). Let the power series $S^{\infty}(X) = \sum_{i=0}^{\infty} S_i X^i$ be defined as in (3.7). Let $\mathbf{r} = (r_0\, r_1\, \dots\, r_{n-1}) = \mathbf{c} + \mathbf{e}$ be the received word in $\mathbb{F}_q^n$, where $\mathbf{c} \in \mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$. Then, the coefficients of $S^{\infty}(X)$ are given by:

$$S_i = \sum_{j=0}^{n-1} r_j v_j \alpha_j^i. \tag{3.8}$$

Proof The reciprocal polynomial of $R(X)$ is explicitly:

$$\begin{aligned}\overline{R}(X) = X^{n-1} \cdot R(X^{-1}) &= X^{n-1} \sum_{j=0}^{n-1} \frac{r_j}{\overline{v}_j} L_j(\alpha_j)^{-1} \prod_{\substack{i=0\\ i\neq j}}^{n-1} (X^{-1} - \alpha_i)\\ &= \sum_{j=0}^{n-1} \frac{r_j}{\overline{v}_j} L_j(\alpha_j)^{-1} \prod_{\substack{i=0\\ i\neq j}}^{n-1} (1 - \alpha_i X).\end{aligned} \tag{3.9}$$

With $v_j^{-1} = \overline{v}_j L_j(\alpha_j)$ from (2.29) for the column multipliers, we get:

$$\overline{R}(X) = \sum_{j=0}^{n-1} \frac{r_j}{\overline{v}_j} L_j(\alpha_j)^{-1} \prod_{\substack{i=0\\ i\neq j}}^{n-1} (1 - \alpha_i X) = \sum_{j=0}^{n-1} r_j v_j \prod_{\substack{i=0\\ i\neq j}}^{n-1} (1 - \alpha_i X). \tag{3.10}$$

The reciprocal of $L(X)$ is:

$$\overline{L}(X) = X^n \cdot L(X^{-1}) = \prod_{i=0}^{n-1} (1 - \alpha_i X). \tag{3.11}$$

Thus, with (3.10) and with (3.11), we can write (3.7) more explicitly and obtain:

$$\frac{\overline{R}(X)}{\overline{L}(X)} = \frac{\sum\limits_{j=0}^{n-1} r_j v_j \prod\limits_{\substack{i=0\\ i\neq j}}^{n-1} (1 - \alpha_i X)}{\prod\limits_{i=0}^{n-1} (1 - \alpha_i X)} = \sum_{j=0}^{n-1} \frac{r_j v_j}{(1 - \alpha_j X)}. \tag{3.12}$$

And with the geometric progression, we obtain from (3.12)

$$\frac{\overline{R}(X)}{\overline{L}(X)} = S^{\infty}(X) = \sum_{i=0}^{\infty} S_i X^i = \sum_{i=0}^{\infty} \sum_{j=0}^{n-1} r_j v_j (\alpha_j X)^i.$$

and therefore $S_i = \sum_{j=0}^{n-1} r_j v_j \alpha_j^i$, for all $i \in \mathbb{N}$. ■

Thus, dividing (3.6) by $\overline{L}(X)$ and with $S(X) \overset{\text{def}}{\equiv} S^{\infty}(X) \bmod X^{n-k}$, we get

$$\Lambda_1(X) \cdot S(X) \equiv \Omega(X) \mod X^{n-k}, \tag{3.13}$$

which corresponds exactly to the classical Key Equation. Therefore, we denote $\Lambda_1(X)$ as $\Lambda(X)$. Given $\mathbf{e} \in \mathbb{F}_q^n$, the so-called error-locator polynomial $\Lambda(X)$ is the univariate polynomial in $\mathbb{F}_q[X]$ of minimal degree, such that for all $i \in \operatorname{supp}(\mathbf{e}) \Leftrightarrow \Lambda(\alpha^i) = 0$. For $\operatorname{wt}(\mathbf{e}) = \tau$, the degree of $\Lambda(X)$ is τ, and the degree of $\Omega(X)$ is smaller than τ.

Let us at this point shortly summarize the classical derivation of the Key Equation based on the syndrome definition similar to the description of [B-Rot06, Chapter 6]. From the $(n-k) \times n$ parity-check matrix $\mathbf{H}$ of an $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$ as defined (2.28), we know that

$$\mathbf{H}\mathbf{c}^T = \mathbf{0}$$

for $\mathbf{c} \in \mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$. More explicitly, we have:

$$\sum_{j=0}^{n-1} c_j \upsilon_j \alpha_j^i = 0, \quad \forall i \in [n-k).$$

Therefore, the syndrome expression of (3.8) simplifies to:

$$S_i = \sum_{j=0}^{n-1} r_j \upsilon_j \alpha_j^i = \sum_{j=0}^{n-1} e_j \upsilon_j \alpha_j^i = \sum_{j \in E} e_j \upsilon_j \alpha_j^i,$$

where $E = \operatorname{supp}(\mathbf{e})$. Define the error-locator polynomial $\Lambda(X)$ and the error-evaluator polynomial $\Omega(X)$ in $\mathbb{F}_q[X]$ as follows:

$$\Lambda(X) \overset{\text{def}}{=} \prod_{j \in E} (1 - \alpha_j X), \tag{3.14}$$

$$\Omega(X) \overset{\text{def}}{=} \sum_{j \in E} e_j \upsilon_j \prod_{i \in E \setminus \{j\}} (1 - \alpha_i X), \tag{3.15}$$

and from (3.12) we get the following relation:

$$\begin{aligned} S(X) &\equiv \sum_{j=0}^{n-1} \frac{r_j \upsilon_j}{(1 - \alpha_i X)} \mod X^{n-k} \\ &\equiv \sum_{j \in E} \frac{e_j \upsilon_j}{(1 - \alpha_i X)} \mod X^{n-k} \\ &\equiv \frac{\Omega(X)}{\Lambda(X)} \mod X^{n-k}. \end{aligned}$$

We consider only the terms of the polynomials with highest degree, when we represent (3.13) in matrix form. From (3.13) we get the τ homogeneous linear equations of the following form:

$$\sum_{i=0}^{n-\tau-k} \Lambda_i \cdot S_{j-i} = 0, \qquad j \in [n-k-\tau, n-k). \tag{3.16}$$

Let us assume that $\tau = |E| = \lfloor (n-k)/2 \rfloor$. Reverting the coefficients of (3.16) leads to:

$$\sum_{i=0}^{n-\tau-k} \Lambda_{\tau-i} \cdot S_{i+j} = 0, \quad j \in [\tau),$$

$$\iff \begin{pmatrix} S_0 & S_1 & \dots & S_\tau \\ S_1 & S_2 & \dots & S_{\tau+1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{\tau-1} & S_\tau & \dots & S_{2\tau-1} \end{pmatrix} \cdot \begin{pmatrix} \Lambda_\tau \\ \Lambda_{\tau-1} \\ \vdots \\ \Lambda_0 \end{pmatrix} = \mathbf{0}. \tag{3.17}$$

The $\tau \times (\tau+1)$ syndrome matrix $\mathbf{S} = (S_{i+j})_{i\in[\tau)}^{j\in[\tau+1)}$ in (3.17) is a Hankel matrix, i.e., $S_{i,j} = S_{i+j}$ holds for all $i \in [\tau+1), j \in [\tau)$.

3.2.2 Error/Erasure Decoding of Generalized Reed–Solomon Codes

We shortly outline the syndrome-based error/erasure decoding procedure for GRS codes as it was first investigated by Forney [A-For65] and the modification of the EEA, which was introduced by Sugiyama, Kasahara, Hirasawa and Namekawa [A-SKHN76].

Let ? mark an erasure. For the transmission over an error/erasure channel as in Figure 3.2, the received vector is denoted by $\widetilde{\mathbf{r}}$, where each $\widetilde{r}_j$ is in the alphabet $\mathbb{F}_q \cup \{?\}$. Let

$$\widetilde{r}(X) = \sum_{j=0}^{n-1} \widetilde{r}_j X^j$$

be the received polynomial for the error/erasure case.

Let the set $E = \{i_0, i_1, \dots, i_{\varepsilon-1}\}$ of cardinality $|E| = \varepsilon$ be the set of erroneous positions and let the set $Z = \{j_0, j_1, \dots, j_{\zeta-1}\}$ of cardinality $|Z| = \zeta$ be the set of erased positions.

In the first step of the decoding process, the erasures in $\widetilde{r}(X)$ are substituted by an arbitrary element from $\mathbb{F}_q$. For simplicity, it is common to choose the zero-element. Thus, the corresponding erasure polynomial in $\mathbb{F}_q[X]$ is denoted by $z(X) = \sum_{i\in Z} z_i X^i$, where $c_i + z_i = 0, \ \forall i \in Z$. Let the modified received polynomial $r(X) \in \mathbb{F}_q[X]$ be

$$r(X) \overset{\text{def}}{=} \sum_{i=0}^{n-1} r_i X^i = c(X) + e(X) + z(X), \tag{3.18}$$

where $c(X)$ is a codeword of the GRS code $\mathcal{GRS}(\boldsymbol{\overline{\upsilon}}, \boldsymbol{\alpha}, k)$ with column multipliers $\boldsymbol{\upsilon} = (\upsilon_0\, \upsilon_1\, \dots\, \upsilon_{n-1})$ and $e(X) = \sum_{i\in E} e_i X^i$ in $\mathbb{F}_q[X]$ is the error polynomial.

The syndrome polynomial $S(X) = \sum_{i=0}^{n-k-1} S_i X^i \in \mathbb{F}_q[X]$ with

$$(S_0\, S_1\, \dots\, S_{n-k-1})^T = \mathbf{H}\mathbf{r}^T$$

is then:

$$S(X) \overset{\text{def}}{=} \sum_{i=0}^{n-k-1} \sum_{j=0}^{n-1} r_j \upsilon_j \alpha_j^i X^i. \tag{3.19}$$

To obtain a Key Equation for error/erasure decoding, we write the syndrome polynomial as power series expansion:

$$S(X) \equiv \sum_{j=0}^{n-1} \frac{r_j \upsilon_j}{(1-\alpha_j X)} \mod X^{n-k}$$

as in the error-only case. In the case of errors and erasures, we obtain with (3.18) for the received vector/polynomial:

$$S(X) \equiv \sum_{j \in E \cup Z} \frac{(e_j + z_j)\upsilon_j}{(1-\alpha_j X)} \mod X^{n-k}. \tag{3.20}$$

Since we know the positions of the erasures, we can compute an erasure-locator polynomial.

Definition 3.7 (Erasure-Locator Polynomial)
Let $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$ be an $[n,k]_q$ GRS code with column-multipliers $\boldsymbol{\upsilon}$ as in Definition 2.28. Let the set Z with $|Z| = \zeta$ denote the erasure set. The erasure-locator polynomial $\Psi(X)$ in $\mathbb{F}_q[X]$ is defined as:

$$\Psi(X) \overset{\text{def}}{=} \prod_{j \in Z} (1 - \alpha_j X). \tag{3.21}$$

Now, we relate the syndrome definition to the erasure-locator polynomial $\Psi(X)$. From (3.20) we obtain:

$$\begin{aligned} S(X) &\equiv \sum_{j \in E \cup Z} \frac{(e_j + z_j)\upsilon_j}{(1-\alpha_j X)} \mod X^{n-k} \\ &\equiv \sum_{j \in E} \frac{e_j \upsilon_j}{(1-\alpha_j X)} + \sum_{j \in Z} \frac{z_j \upsilon_j}{(1-\alpha_j X)} \mod X^{n-k} \\ &\overset{\text{def}}{\equiv} \frac{\Omega(X)}{\Lambda(X)} + \frac{\Phi(X)}{\Psi(X)} \mod X^{n-k}, \end{aligned} \tag{3.22}$$

where $\Omega(X)$ is the error-evaluator polynomial as defined in (3.15) and $\Phi(X)$ is the erasure-evaluator polynomial:

$$\Phi(X) \overset{\text{def}}{=} \sum_{j \in Z} z_j \upsilon_j \prod_{i \in Z \setminus \{j\}} (1 - \alpha_i X) \tag{3.23}$$

of degree at most $\zeta - 1$. To obtain a "combined" Key Equation, a further modification is necessary. Let us modify the syndrome polynomial.

Lemma 3.8 (Modified Syndrome Polynomial)
Let $S(X)$ be the syndrome polynomial (3.19) and $\Phi(X)$ the erasure-locator polynomial (3.23). Let us define a modified syndrome polynomial as:

$$\widetilde{S}(X) \overset{\text{def}}{\equiv} \Psi(X) \cdot S(X) \mod X^{n-k}. \tag{3.24}$$

Then the highest $n - k - \zeta$ coefficients of $\widetilde{S}(X)$ depend only on the error polynomial $e(X)$.

Proof The statement follows directly from (3.19) and from the degree of $\Psi(X)$. ■

Inserting (3.22) into (3.24) yields:

$$\begin{aligned}
\widetilde{S}(X) &\equiv \Psi(X) \cdot S(X) \mod X^{n-k} \\
\widetilde{S}(X) &\equiv \Psi(X) \left(\frac{\Omega(X)}{\Lambda(X)} + \frac{\Phi(X)}{\Psi(X)} \right) \mod X^{n-k}, \\
\widetilde{S}(X) &\equiv \frac{\Psi(X)\Omega(X) + \Phi(X)\Lambda(X)}{\Lambda(X)} \mod X^{n-k},
\end{aligned}$$

and with the combined error/erasure evaluator polynomial

$$\widetilde{\Omega}(X) \overset{\text{def}}{=} \Omega(X)\Psi(X) + \Phi(X)\Lambda(X),$$

we obtain the Key Equation for error/erasure decoding of GRS codes:

$$\widetilde{S}(X) \equiv \frac{\widetilde{\Omega}(X)}{\Lambda(X)} \mod X^{n-k}, \tag{3.25}$$

where $\deg \Lambda(X) = \varepsilon$ and $\deg \widetilde{\Omega}(X) \leq \varepsilon + \zeta - 1$. In the erasure-free case, $\widetilde{\Omega}(X)$ becomes the error-evaluator polynomial $\Omega(X)$, with $\deg \Omega(X) \leq \varepsilon - 1$.

3.2.3 Welch–Berlekamp-like Approach for Error/Erasure Decoding

For the interpolation-based decoding approach, the ζ positions are neglected and the reduced interpolation problem of $n - \zeta$ points is solved. The evaluation polynomial $f(X)$ of the sent codeword $\mathbf{c} = \text{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ of an $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ is directly obtained and therefore an error/erasure-evaluation is not necessary.

To obtain the Key Equation (3.25) for error/erasure decoding from the interpolation-based starting point, we have to modify the derivation as in Lemma 3.4.

Let the reciprocal of the erasure-locator polynomial as in Definition 3.7 be

$$\overline{\Psi}(X) \overset{\text{def}}{=} X^{\zeta}\Psi(X^{-1}) = \prod_{i \in Z}(X - \alpha_i). \tag{3.26}$$

Let $R(X)$ be the Lagrange polynomial, such that $R(\alpha_i) = r_i/\overline{v}_i$ for all $i \in [n)$. Clearly $R(X)$ has multiplicity one at all erasure positions and therefore a unique polynomial $R_-(X)$ with degree less than $n - \zeta - 1$ exists, such that:

$$R(X) = \overline{\Psi}(X) \cdot R_-(X).$$

Furthermore, let $L_-(X) \in \mathbb{F}_q[X]$ of degree $n - \zeta$ be such that $L(X) = \overline{\Psi}(X)L_-(X)$. Then the univariate reformulation of Lemma 3.5 becomes:

$$\begin{aligned}
Q(X, R(X)) &= B(X) \cdot L(X), \\
Q_0(X) + Q_1(X)\overline{\Psi}(X)R_-(X) &= B(X) \cdot \overline{\Psi}(X)L_-(X),
\end{aligned}$$

with deg $B(X) < n - k - \varepsilon$ and by reverting the coefficients as previously and with $\overline{R}_-(X) \overset{\text{def}}{=} X^{n-\zeta-1}R_-(X^{-1})$ and $\overline{L}_-(X) \overset{\text{def}}{=} X^{n-\zeta}L_-(X^{-1})$, we obtain:

$$\Lambda_0(X)X^{n-k} + \Lambda_1(X)\Psi(X)\overline{R}_-(X) = \overline{B}(X) \cdot \Psi(X)\overline{L}_-(X)$$

and with $S(X) \equiv \overline{R}_-(X)/\overline{L}_-(X) \mod X^{n-k}$ (that coincides with the syndrome definition of (3.19)) we obtain the Key Equation as in (3.25):

$$\Lambda_1(X)\widetilde{S}(X) \equiv \overline{B}(X) \cdot \Psi(X) \mod X^{n-k}, \tag{3.27}$$

where the degree of $\overline{B}(X) \cdot \Psi(X)$ is less than $n - k - \varepsilon - 1 + \zeta = \varepsilon - 1 + \zeta$ and it corresponds to $\widetilde{\Omega}(X)$ of (3.25). We use the Key Equation (3.27) in Algorithm 3.2 in the next section for error/erasure decoding of GRS codes with the EEA.

3.3 Decoding Algorithms Based on the Key Equation

3.3.1 Overview

The Key Equation for syndrome-based decoding of GRS codes can be solved by the well-known Berlekamp–Massey algorithm [B-Ber68; A-Mas69] or the Sugiyama–Kasahara–Hirasawa–Namekawa algorithm [A-SKHN75] based on the Extended Euclidean Algorithm (EEA). Several publications discuss the parallels of these two algorithms (see [A-Dor87; A-JH00; O-AO09], [O-Hey01, Chapter 2]).

We present the Fundamental Iterative Algorithm (FIA) of Feng and Tzeng [A-FT85; A-FT91a], that can solve a system of homogeneous linear equations. The FIA generalizes well to a structured system of linear equations derived from the interpolation-based algorithms of Sudan and Guruswami–Sudan (see Chapter 4 and 5).

3.3.2 Extended Euclidean Algorithm and Error/Erasure Decoding of Generalized Reed–Solomon Codes

The EEA is discussed e.g., in [B-Lip81, Chapter VII], [B-GG03, Chapter 3] and [B-MS88a, Chapter 12 §8]. We present the EEA for the sake of completeness, but do not prove all necessary properties. Algorithm 3.1 is the EEA here for two elements a and b in a Euclidean domain $\mathbb{D}$ and the function d denotes the degree function of $\mathbb{D}$. We initialize the remainders u_{-1} and u_0 with the elements a and b and the coefficients s_i, t_i for $i = -1, 0$.

Algorithm 3.1: $(u_{i-1}, s_{i-1}, t_{i-1}) = \text{EEA}(a, b, \mathsf{crit})$

Input: Elements a, b with $d(a) > d(b)$ in a Euclidean Domain, stopping criteria crit
Output: $u_{i-1}, s_{i-1}, t_{i-1}$

Initialize: $\begin{pmatrix} u_{-1} \\ u_0 \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$ and $\begin{pmatrix} s_{-1} & t_{-1} \\ s_0 & t_0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, i = 0$

```
while crit do
    $i = i + 1$
    $q_i = \lfloor u_{i-2}/u_{i-1} \rfloor$
    $\begin{pmatrix} u_i & s_i & t_i \end{pmatrix} = \begin{pmatrix} 1 & -q_i \end{pmatrix} \cdot \begin{pmatrix} u_{i-2} & s_{i-2} & t_{i-2} \\ u_{i-1} & s_{i-1} & t_{i-1} \end{pmatrix}$
```

If the stopping criteria is crit = $\{u_i \neq 0\}$, then the EEA terminates in the $(i+1)$-th step and returns the greatest common divisor u_{i+1} of a and b, i.e.:

$$u_{i+1} = s_{i+1}a + t_{i+1}b.$$

For the proof of correctness of Algorithm 3.1 and the complexity analysis, the interested reader is referred to the literature (e.g. [B-Lip81, Chapter VII]).

Algorithm 3.2 summarizes the different steps for error/erasure BMD decoding of GRS codes based on the EEA.

```
Algorithm 3.2: $c(X) = \text{EE-DECODER}(\widetilde{r}(X), \boldsymbol{v}, \boldsymbol{\alpha}, k)$

Input: Received word $\widetilde{r}(X) \in \mathbb{F}_q[X] \cup \{?\}$, parameters $\boldsymbol{v}, \boldsymbol{\alpha}, k$ of $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$
Output: Estimated codeword $c(X)$ or DECODING FAILURE

1  Substitute erasures from $\widetilde{r}(X)$ by zero to obtain $r(X)$
2  Save positions of erasures in $Z = \{i_0, i_1, \dots, i_{\zeta-1}\}$
3  Calculate erasure-polynomial $\Psi(X)$ as in (3.21)
4  Calculate $\widetilde{S}(X)$ as in (3.24)                      // Syndrome calculation
5  Set crit $= \{\deg u_i < (n-k+\zeta)/2 - 1\}$
6  ␣, $\Lambda(X), \widetilde{\Omega}(X) = \text{EEA}\big(X^{n-k}, \widetilde{S}(X), \text{crit}\big)$
7  Find all $i$, where $\Lambda(\gamma_i) = 0 \Rightarrow E = \{i_0, i_1, \dots, i_{\varepsilon-1}\}$      // Chien-like search
8  if $\varepsilon < \deg \Lambda(X)$ then
9      Declare DECODING FAILURE
10 else
11     Determine error/erasure values $e_{i_0}, e_{i_1}, \dots, e_{i_{\varepsilon-1}}$ and $z_{i_0}, z_{i_1}, \dots, z_{i_{\zeta-1}}$
12     $e(X) \leftarrow \sum_{i \in E} e_i X^i$ and $z(X) \leftarrow \sum_{i \in Z} z_i X^i$
13     $c(X) \leftarrow r(X) - e(X) - z(X)$
```

In the following, we shortly outline how to solve (3.25) by the EEA as described in [A-SKHN75; A-SKHN76] to decode classical Goppa codes. In Line 6 of Algorithm 3.2, the EEA is called and the sign ␣ indicates that the returned polynomial is not needed for further calculations.

Theorem 3.9 (Error/Erasure Decoding, [B-MS88a, Chapter 12, Theorem 16])
Assume $\zeta < n-k$ erasures occurred. Let $\widetilde{S}(X)$ with $\deg \widetilde{S}(X) < n-k$ as in (3.24) be given. If

$$\varepsilon = |E| \leq \left\lfloor \frac{n-k-\zeta}{2} \right\rfloor,$$

then there exists a unique solution of (3.25) and Algorithm 3.2 with the input polynomials $u_{-1}(X) = X^{n-k}$ and $u_0(X) = \widetilde{S}(X)$ determines the error-locator polynomial $\Lambda(X)$ and the error/erasure-evaluation polynomial $\widetilde{\Omega}(X)$ as in (3.14). The following stopping criteria crit for Algorithm 3.2 guarantees the correct solution:

$$\deg u_{i-1} \geq \frac{n-k+\zeta}{2} \quad \text{and} \quad \deg u_i \leq \frac{n-k+\zeta}{2} - 1. \tag{3.28}$$

The determination of the error and erasure values as in Line 11 of Algorithm 3.2 can be done by Forney's formula [A-For65]. See Chapter 6 for the Forney formula in the case of error-evaluation for decoding cyclic codes.

3.3.3 The Fundamental Iterative Algorithm

In this subsection, we explain the basic idea of the Fundamental Iterative Algorithm (FIA) of Feng and Tzeng [A-FT85; A-FT91a]. The outline follows the description of [O-Kö96b, Chapter 4].

Given an arbitrary $m \times n$ matrix $\mathbf{M} = (M_{i,j})_{i \in [m)}^{j \in [n)}$ with $n > m$ over $\mathbb{F}_q$, the FIA outputs the minimal number of the $\mu + 1$ first linearly dependent columns together with the polynomial $T(X) = \sum_{i=0}^{\mu} T_i X^i$ in $\mathbb{F}_q[X]$, with $T_\mu \neq 0$, such that

$$\sum_{j=0}^{\mu} T_j M_{i,j} = 0, \quad i \in [m).$$

The FIA scans the μ-th column of the matrix $\mathbf{M}$ row-wise in the order $M_{0,\mu}, M_{1,\mu}, \dots$ and uses previously stored polynomials to update the current polynomial $T(X)$. Let μ be the index of the current column of matrix $\mathbf{M}$ under inspection. Let $T(X) = \sum_{j=0}^{\mu} T_j X^j$ be the current candidate polynomial and let κ be the greatest row-index such that:

$$\sum_{j=0}^{\mu} T_j M_{i,j} = 0, \quad \forall i \in [\kappa). \tag{3.29}$$

We denote, where it is appropriate, $\kappa(\mu)$ for the greatest κ in column μ, such that (3.29) holds. In other words, the coefficients of the polynomial $T(X)$ give us the vanishing linear combination of the matrix consisting of the first κ rows and the first $\mu + 1$ columns of the matrix $\mathbf{M}$. The discrepancy

$$\Delta = \sum_{j=0}^{\mu} T_j M_{\kappa,j} \tag{3.30}$$

for the next row $\kappa + 1$ is non-zero. In the case $\Delta \neq 0$ and there is no discrepancy Δ_κ stored, the current discrepancy Δ is stored as Δ_κ. The corresponding auxiliary polynomial is stored as $T_\kappa(X)$. Then, the FIA examines a new column $\mu + 1$. Let us define the case, when the FIA examines a new column.

> **Definition 3.10 (Core Discrepancy of FIA)**
> Let the row $\kappa < (m-1)$ and the column $\mu < n$ of a $m \times n$ matrix $\mathbf{M}$ over $\mathbb{F}_q$ with $n > m$ be examined by the FIA. Let the calculated discrepancy as in (3.30) be non-zero and no other non-zero discrepancy be stored for the row κ. Then, the FIA stores the current discrepancy Δ as Δ_κ, the current polynomial $T(X)$ as $T_\kappa(X)$ and examines a new column $\mu + 1$. We call this case a core discrepancy.

If there exists a previously stored polynomial $T_\kappa(X)$ and a non-zero discrepancy $\Delta_\kappa \in \mathbb{F}_q$, which corresponds to row κ, then the current polynomial $T(X)$ is updated in the following way:

$$T(X) \leftarrow T(X) - \frac{\Delta}{\Delta_\kappa} T_\kappa(X). \tag{3.31}$$

The following lemma proves the proposed update rule (3.31).

Lemma 3.11 (Update Rule of the FIA)
Let the FIA examine a $m \times n$ matrix $\mathbf{M}$ over $\mathbb{F}_q$ and $n > m$. Let $\mu < n$ be the current column of $\mathbf{M}$ under inspection and let κ be maximal such that (3.29) holds. Let $T_\kappa(X) = \sum_i T_{\kappa,i} X^i \in \mathbb{F}_q[X]$ and $\Delta_\kappa \in \mathbb{F}_q$ be a previously stored connection polynomial and discrepancy for row κ. Let the current discrepancy Δ as in (3.30) be non-zero. Then for $\overline{T}(X) \overset{\text{def}}{=} T(X) - \frac{\Delta}{\Delta_\kappa} T_\kappa(X)$:

$$\sum_{j=0}^{\mu} \overline{T}_j M_{i,j} = 0, \quad \forall i \in [\kappa + 1)$$

holds.

Proof The proof of the above update rule is straightforward see [A-FT91b, Lemma 1]. We have:

$$\begin{aligned} \sum_{j=0}^{\mu} \overline{T}_j M_{i,j} &= \sum_{j=0}^{\mu} T_j M_{i,j} - \sum_{j=0}^{\mu} \frac{\Delta}{\Delta_\kappa} T_{\kappa,j} M_{i,j} \\ &= \begin{cases} 0 - \dfrac{\Delta}{\Delta_\kappa} \cdot 0, & \forall i \in [\kappa), \\ \Delta - \dfrac{\Delta}{\Delta_\kappa} \cdot \Delta_\kappa, & \text{for } i = \kappa. \end{cases} \end{aligned}$$ ■

Lemma 3.12 (Rank and Core Discrepancy of the FIA)
Let a $m \times n$ matrix $\mathbf{M} = (\mathbf{M}_0^T \; \mathbf{M}_1^T \; \dots \; \mathbf{M}_{n-1}^T)$, where each $\mathbf{M}_i \in \mathbb{F}_q^m$, be examined by the FIA. Let μ be an integer and $\mu < n$. The rank of the sub-matrix $(\mathbf{M}_0^T \; \mathbf{M}_1^T \; \dots \; \mathbf{M}_\mu^T)$ is equal to the number of encountered non-zero core discrepancies (as in Definition 3.10), which the FIA has found when examining columns 0 to μ of matrix $\mathbf{M}$.

Proof See for instance [A-FT91a, Lemma 2]. Let $\Delta_{\kappa(0)}, \Delta_{\kappa(1)}, \dots, \Delta_{\kappa(\mu)}$ be the stored core discrepancies and let $T_{\kappa(0)}(X), T_{\kappa(1)}(X), \dots, T_{\kappa(\mu)}(X)$ be the corresponding $\mu + 1$ stored auxiliary polynomials, for the rows $\kappa(0), \kappa(1), \dots, \kappa(\mu)$ after the FIA examined the first $\mu + 1$ columns of a $m \times n$ matrix $\mathbf{M}$. Let the $\mu + 1$ vectors of length $\mu + 1$ be defined as:

$$\mathbf{T}_i \overset{\text{def}}{=} (T_{i,0} \; T_{i,1} \; \dots \; T_{i,\kappa(i)} \; 0 \; \dots \; 0), \quad \forall i \in [\mu + 1).$$

Let the $\mu + 1$ vectors of length m be defined as:

$$\mathbf{D}_i \overset{\text{def}}{=} \left(\mathbf{M}_0^T \; \mathbf{M}_1^T \; \dots \; \mathbf{M}_\mu^T\right) \mathbf{T}_i = \left(0 \; \dots \; 0 \; \Delta_{\kappa(i)} \; \star \; \dots \; \star\right)^T, \quad i \in [\mu + 1),$$

where $\star \in \mathbb{F}_q$. Then the $(\mu+1) \times (\mu+1)$ discrepancy matrix is defined as:

$$\mathbf{D} \overset{\text{def}}{=} \left(\mathbf{D}_0 \, \mathbf{D}_1 \, \dots \, \mathbf{D}_\mu\right) = \begin{pmatrix} 0 & 0 & \dots & 0 \\ \vdots & \vdots & & \Delta_{\kappa(\mu)} \\ 0 & & & \star \\ \Delta_{\kappa(0)} & 0 & & \\ \star & \Delta_{\kappa(1)} & \vdots & \vdots \\ \vdots & \star & & \\ \vdots & \vdots & & \\ \star & \star & \dots & \star \end{pmatrix}. \tag{3.32}$$

Since all core discrepancies occurred at different rows, the columns of the discrepancy matrix $\mathbf{D}$ as in (3.32) can be re-ordered into lower-triangular form and thus $\mathbf{D}$ has rank $\mu+1$. Equivalently, the rank of the $(\mu+1) \times (\mu+1)$ matrix $(\mathbf{T}_0 \, \mathbf{T}_1 \dots \mathbf{T}_\mu)$ is $\mu+1$. From (3.32), we have:

$$\mathbf{D} = \left(\mathbf{M}_0^T \, \mathbf{M}_1^T \, \dots \, \mathbf{M}_\mu^T\right) \left(\mathbf{T}_0 \, \mathbf{T}_1 \dots \mathbf{T}_\mu\right),$$

and with

$$\operatorname{rank}(\mathbf{D}) \leq \min\left(\operatorname{rank}\left(\mathbf{M}_0^T \, \mathbf{M}_1^T \, \dots \, \mathbf{M}_\mu^T\right), \operatorname{rank}\left(\mathbf{T}_0 \, \mathbf{T}_1 \dots \mathbf{T}_\mu\right)\right),$$

we conclude that the $(\mu+1) \times (\mu+1)$ sub-matrix $(\mathbf{M}_0^T \, \mathbf{M}_1^T \, \dots \, \mathbf{M}_\mu^T)$ of $\mathbf{M}$ is also of full rank $\mu+1$. ■

> **Theorem 3.13 (Correctness and Complexity)**
> Let the FIA examine a $m \times n$ matrix $\mathbf{M}$ with $n > m$ and entries in $\mathbb{F}_q$. If the last row $m-1$ of $\mathbf{M}$ is examined, the polynomial $T_\mu(X)$ corresponds to a valid linear combination of the first $\mu+1$ columns of $\mathbf{M}$. The time complexity of the FIA is $\mathcal{O}(m^3)$.

Proof The correctness follows from Lemma 3.12. For the complexity analysis: Each discrepancy calculation has complexity at most $\mathcal{O}(m)$ and is performed at most m times in each of m columns. ■

It is more difficult to prove that the FIA returns the shortest linear combination (see [A-FT91a, Theorem 1]) and that it can be used to prove the correctness of multi-sequence shift-register synthesis, especially of different length (see [O-Sch07, Chapter 4] and [A-SS11, Section 3.3]). We do not use this property and therefore do not investigate it here.

Algorithm 3.3: $T(X) = \texttt{FIA-ONEHANKEL}(S(X))$

```
Input: Syndrome polynomial $S(X) \in \mathbb{F}_q[X]$ with $\deg S(X) < 2\tau$;
Output: Univariate polynomial $T(X) \in \mathbb{F}_q[X]$;

Data structures:
  Column pointer $\mu \in [\tau+1)$, Row pointer $\kappa \in [\tau)$;
  Array $D$ of $\tau$ entries in $\mathbb{F}_q$, Array $A$ of $\tau$ entries in $\mathbb{F}_q[X]$;
  Variable $\Delta \in \mathbb{F}_q$, variable compute $\in \{\mathsf{true}, \mathsf{false}\}$;

Initialize:
  for every $i \in [\tau)$: $D[i] \leftarrow 0$;
  $\mu \leftarrow 0$, $\kappa \leftarrow 0$;
  $T(X) \leftarrow 1$; compute $\leftarrow$ false;

while $\kappa < \tau$ do
    if compute then
        $\Delta \leftarrow \langle X^{\kappa} \cdot T(X), S(X) \rangle$        // Discrepancy calculation
    else
        if $\kappa < 1$ then
            $T(X) \leftarrow X^{\mu}; \Delta \leftarrow S_{\mu}; \kappa \leftarrow 0$
        else
            $T(X) \leftarrow X \cdot T(X); \kappa \leftarrow \kappa - 1$
        compute $\leftarrow$ true
    if $\Delta = 0$ or $D[\kappa] \neq 0$ then
        if $\Delta \neq 0$ then
            $T(X) \leftarrow T(X) - \frac{\Delta}{D[\kappa]} \cdot A[\kappa](X)$        // Update
        $\kappa \leftarrow \kappa + 1$
    else        // Core discrepancy $\Delta \neq 0$ and $D[\kappa] = 0$
    $A[\kappa](X) \leftarrow T(X); D[\kappa] \leftarrow \Delta; \mu \leftarrow \mu + 1$
    compute $\leftarrow$ false
```

In the following, we adjust the FIA to a Hankel matrix denoted by $\mathbf{S}$ instead of $\mathbf{M}$. Furthermore, we refine also the dimension of the matrix $\mathbf{S}$ to draw easily the connection to a univariate polynomial in $\mathbb{F}_q[X]$. First, we state the problem in terms of the inner product.

Problem 3.14 (Hankel Matrix System)
Let $\mathbf{S} = (S_{i,j})_{i \in [\tau)}^{j \in [\tau+1)}$ be a $\tau \times (\tau+1)$ Hankel matrix with entries $S_{i,j} \in \mathbb{F}_q$. Let $S(X) = \sum_{i=0}^{2\tau-1} S_i X^i$ be the associated univariate polynomial in $\mathbb{F}_q[X]$, such that:

$$S_{i,j} = S_{i+j}, \quad \forall i \in [\tau), j \in [\tau+1).$$

We search a non-zero polynomial $T(X) \in \mathbb{F}_q[X]$ that fulfills:

$$\langle\, X^{\kappa} T(X), S(X) \,\rangle = 0, \quad \forall \kappa \in [\tau),$$

where $\deg T(X) \leq \tau$.

Algorithm 3.3 is the FIA adjusted to a Hankel matrix and it returns a polynomial $T(X)$ solving

Problem 3.14. Similar to the formulation in Problem 3.14, the discrepancy calculation as in (3.30) for the FIA can be given in terms of the inner product for the case of a Hankel matrix (see Line 3 of Algorithm 3.3). The column pointer μ indexes the column and the row pointer κ indexes the row of the Hankel matrix $\mathbf{S}$ under inspection. The variable Δ is used to calculate the current discrepancy according to (3.30) (see Line 3). These values in $\mathbb{F}_q$ are stored in array D and the corresponding intermediate polynomials are stored in the array A in the case of a core discrepancy as in Definition 3.10.

The Boolean variable *compute* $\in \{\mathsf{true}, \mathsf{false}\}$ in Line 2 splits the FIA into two cases. It becomes true, when a discrepancy calculation has to be executed. The polynomial $T(X)$ is updated according to (3.31) in Line 12. The value of *compute* is false, when a new column (see Lines 5-8) is entered and no computation of the discrepancy has to be executed.

The initialization in Line 8 of Algorithm 3.3 is (besides the way of calculating the discrepancy) the main difference of the FIA, adjusted to one Hankel matrix $\mathbf{S}$, to the FIA for an arbitrary matrix. In the more general case of an arbitrary matrix, the row pointer κ would be set to zero when entering a new column. Due to the Hankel structure of the matrix $\mathbf{S}$, Algorithm 3.3 can start examining the new column at the $(\kappa - 1)$-th row. The following lemma proves this modification.

Lemma 3.15 (Initialization Rule)
Suppose Algorithm 3.3 examines column $\mu - 1$ of a $\tau \times (\tau + 1)$ Hankel matrix $\mathbf{S} = (S_{i,j})_{i \in [\tau)}^{j \in [\tau+1)}$ over $\mathbb{F}_q$ or equivalent a polynomial $S(X) = \sum_{i=0}^{2\tau-1} S_i X^i \in \mathbb{F}_q[X]$ with

$$S_{i,j} = S_{i+j}, \quad \forall i \in [\tau), j \in [\tau + 1).$$

A core discrepancy was obtained in row κ. Let $A[\kappa](X)$ be the previously stored polynomial for that row κ, i.e.:

$$\langle\, X^i A[\kappa](X), S(X) \,\rangle = \sum_{j=0}^{\mu-1} A_j \cdot S_{i+j} = 0, \quad \forall i \in [\kappa).$$

We can start examining the next column μ of $\mathbf{S}$ with the initial value $T(X) \leftarrow X \cdot A[\kappa](X)$ and set the row pointer to $\kappa \leftarrow \kappa - 1$.

Proof We have the following relation:

$$\begin{aligned}\langle\, X^i T(X), S(X) \,\rangle &= \langle\, X^{i+1} A[\kappa](X), S(X) \,\rangle \\ &= \sum_{j=0}^{\mu-1} A_j \cdot S_{i+j+1} \\ &= 0, \quad \forall i \in [\kappa - 1).\end{aligned}$$ ■

The FIA, adjusted to one Hankel matrix, enters the next column by examining the row $\kappa - 1$ instead of row 0. We summarize the properties of Algorithm 3.3 in the following theorem.

Theorem 3.16 (FIA for One Hankel Matrix)
Given a $\tau \times (\tau + 1)$ Hankel matrix $\mathbf{S} = (S_{i,j})_{i \in [\tau)}^{j \in [\tau+1)}$ over $\mathbb{F}_q$, or equivalently, the polynomial $S(X) = \sum_{i=0}^{2\tau-1} S_i X^i \in \mathbb{F}_q[X]$, such that

$$S_{i,j} = S_{i+j}, \quad \forall i \in [\tau), j \in [\tau + 1).$$

Algorithm 3.3 outputs the polynomial $T(X) = \sum_{i=0}^{\mu} T_i X^i \in \mathbb{F}_q[X]$ such that:

$$\langle\, X^{\kappa} T(X), S(X) \,\rangle = 0, \quad \kappa \in [\tau),$$

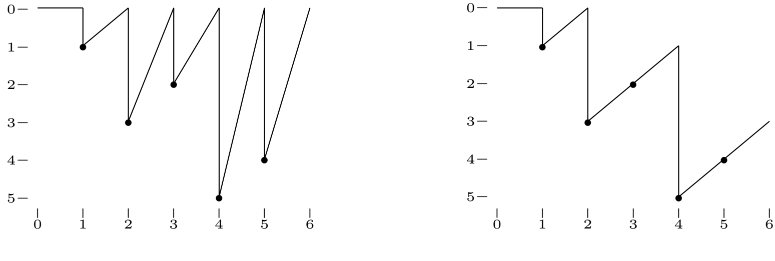
with time complexity $\mathcal{O}(\tau^2)$ in $\mathbb{F}_q$.

Proof The correctness of Algorithm 3.3 follows from the correctness of the basic (unadjusted) FIA as in Theorem 3.13 and from the initialization rule as stated in Lemma 3.15. The proof of the complexity is as follows. Let the triple (μ, κ, δ) consist of the column pointer μ, row pointer κ and a counter for the number of core discrepancies δ. We distinguish two events, when Algorithm 3.3 examines a Hankel matrix $\mathbf{S}$:

1. No core discrepancy: Algorithm 3.3 remains in the same column μ, increases the row pointer κ and the number of calculated core discrepancies δ remains unchanged. The triple is updated as follows:
$$(\mu, \kappa, \delta) \leftarrow (\mu, \kappa + 1, \delta).$$
2. Core discrepancy: Algorithm 3.3 enters next column $\mu + 1$, decreases the row pointer and the number of calculated core discrepancies is increased. Therefore, the triple becomes:
$$(\mu, \kappa, \delta) \leftarrow (\mu + 1, \kappa - 1, \delta + 1).$$

For both cases the sum over the triple $\mu + \kappa + \delta$ increases only by one (in contrast to the unadjusted FIA). The initial value of the triple is $(0, 0, 0)$ and the final value is bounded by $(\tau - 1, \tau - 1, \tau - 1)$, for a $\tau \times (\tau + 1)$ input Hankel matrix. Therefore, the maximal number of iterations of Algorithm 3.3 is of order $\mathcal{O}(\tau) + \mathcal{O}(\tau) + \mathcal{O}(\tau) = \mathcal{O}(\tau)$. Each discrepancy calculation costs at most $\mathcal{O}(\tau)$ operations and therefore the overall time complexity is $\mathcal{O}(\tau^2)$. ■

Let us illustrate the discrepancy calculation of Algorithm 3.3, when it is used for BMD decoding of GRS codes up to $\lfloor (n - k)/2 \rfloor$ errors. The BMD error correcting radius of a $[16, 4]_{17}$ GRS code is $\lfloor (n - k)/2 \rfloor = 6$. The syndrome matrix $\mathbf{S}$ as in (3.17) for six errors is a 6×7 Hankel matrix. To



(a) FIA without adaption

(b) FIA with adaption

Figure 3.3: Illustration of the row pointer κ of the classic FIA (Subfigure 3.3a) and of the adjusted Algorithm 3.3 (Subfigure 3.3b), when both algorithms are applied to a 6×7 Hankel syndrome matrix of a $[16, 4]_{17}$ GRS code. The dots indicate the calculation of a non-zero discrepancy and where an update of the interim polynomial is not possible (core discrepancy, see Definition 3.10). Then, both algorithms enter a new column with different initialization of their row pointers.

illustrate the complexity reduction of the FIA adjusted to a Hankel matrix (compared to the original, unadjusted FIA), we trace the examined rows for each column in Figure 3.3. Sub-figure 3.3a shows the values of the row pointer κ of the FIA without any adaption. The row pointer κ of the adapted FIA is traced in Sub-figure 3.3b. The points in both figures indicate the case of a core discrepancy (see Definition 3.10).

3.4 Collaborative Decoding of Interleaved Generalized Reed–Solomon Codes

3.4.1 Error Model

Let s codewords

$$\mathbf{c}_t \overset{\text{def}}{=} \text{eval}(f_t(X), \overline{\boldsymbol{\upsilon}}_t, \boldsymbol{\alpha}), \quad \forall t \in [s)$$

be s sub-codewords in $\mathbb{F}_q^n$ of an IGRS code $\mathcal{IGRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, \mathbf{k})$ as in Definition 2.34. They are corrupted by s error words $\mathbf{e}_0, \mathbf{e}_1, \dots, \mathbf{e}_{s-1} \in \mathbb{F}_q^n$ of weight $\text{wt}(\mathbf{e}_t) = \varepsilon_t, \forall t \in [s)$. We denote each received word by

$$\mathbf{r}_t \overset{\text{def}}{=} \mathbf{c}_t + \mathbf{e}_t = (r_{t,0}\; r_{t,1}\; \dots\; r_{t,n-1}), \quad \forall t \in [s).$$

We associate to each received vector $\mathbf{r}_t$ a polynomial in $\mathbb{F}_q[X]$ and it is denoted by

$$r_t(X) = \sum_{i=0}^{n-1} r_{t,i} X^i, \quad \forall t \in [s),$$

respectively.

We assume (as usual for interleaved codes) that the channel adds so-called burst errors (see Figure 3.4). Let

$$E_t \overset{\text{def}}{=} \text{supp}(\mathbf{e}_t).$$

We assume that ε burst errors occurred, i.e., the union of the s sets of error positions

$$E \overset{\text{def}}{=} \bigcup_{t=0}^{s-1} E_t \tag{3.33}$$

has cardinality $|E| = \varepsilon$.

Clearly, the error–correction capability of each $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}_t, \boldsymbol{\alpha}, k_t)$ code is $\lfloor (n-k_t)/2 \rfloor$ and successful unambiguous decoding for an $\mathcal{IGRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, \mathbf{k})$ code is possible by sub-code-wise decoding if $|E_t| \leq \lfloor (n-k_t)/2 \rfloor$ for all $t \in [s)$.

3.4.2 Syndromes and Collaborative Decoding Algorithms

Joint or collaborative decoding of IGRS codes makes use of the special structure of the burst error. In the first step, s syndrome polynomials $S_0(X), S_1(X), \dots, S_{s-1}(X) \in \mathbb{F}_q[X]$ of degree smaller than $n-k_0, n-k_1, \dots, n-k_{s-1}$ are calculated. The coefficients of $S_t(X) = \sum_{i=0}^{n-k_t-1} S_{t,i} X^i$ are given in Lemma 3.6, i.e.:

$$S_{t,i} = \sum_{j=0}^{n-1} r_{t,j} \upsilon_{t,j} \alpha_j^i, \quad \forall i \in [n-k_t), t \in [s),$$

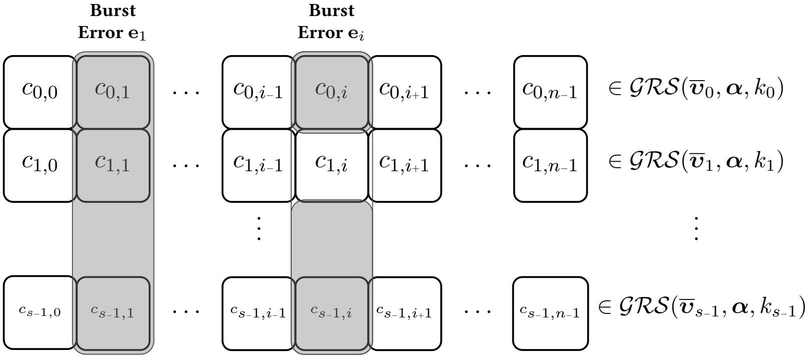

Burst Error $\mathbf{e}_1$
Burst Error $\mathbf{e}_i$
$c_{0,0}$
$c_{0,1}$
$c_{0,i\text{-}1}$
$c_{0,i}$
$c_{0,i+1}$
$c_{0,n\text{-}1}$
$\in \mathcal{GRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k_0)$
$c_{1,0}$
$c_{1,1}$
$c_{1,i\text{-}1}$
$c_{1,i}$
$c_{1,i+1}$
$c_{1,n\text{-}1}$
$\in \mathcal{GRS}(\overline{\boldsymbol{v}}_1, \boldsymbol{\alpha}, k_1)$
$c_{s\text{-}1,0}$
$c_{s\text{-}1,1}$
$c_{s\text{-}1,i\text{-}1}$
$c_{s\text{-}1,i}$
$c_{s\text{-}1,i+1}$
$c_{s\text{-}1,n\text{-}1}$
$\in \mathcal{GRS}(\overline{\boldsymbol{v}}_{s\text{-}1}, \boldsymbol{\alpha}, k_{s\text{-}1})$

Figure 3.4: Illustration of an IGRS code $\mathcal{IGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, \mathbf{k})$, where each sub-code is a GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}_t, \boldsymbol{\alpha}, k_t)$, for all $t \in [s)$. Two burst errors $\mathbf{e}_1, \mathbf{e}_i \in \mathbb{F}_q^s$ occurred at position 1 and i. The second burst error $\mathbf{e}_i$ has one zero component $e_{i,1}$.

where $\boldsymbol{v}_0, \boldsymbol{v}_1, \dots, \boldsymbol{v}_{s-1}$ are the column multipliers of the GRS codes $\mathcal{GRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k_0), \mathcal{GRS}(\overline{\boldsymbol{v}}_1, \boldsymbol{\alpha}, k_1)$, $\dots, \mathcal{GRS}(\overline{\boldsymbol{v}}_{s-1}, \boldsymbol{\alpha}, k_{s-1})$. These syndromes provide s Key Equations with one common error-locator polynomial $\Lambda(X)$:

$$\Lambda(X) \cdot S_t(X) \equiv \Omega_t(X) \mod X^{n-k_t}, \quad \forall t \in [s),$$

where $\deg \Omega_t(X) < \varepsilon$ for all $t \in [s)$. Similar to (3.16), let us consider only the terms with degree at least ε:

$$\sum_{i=0}^{\varepsilon} \Lambda_i \cdot S_{t,j-i} = 0, \quad \forall j \in [\varepsilon, n-k_t), t \in [s). \tag{3.34}$$

The combined system of linear equations, where the coefficients of $\Lambda(X) = \Lambda_0 + \Lambda_1 X + \dots + \Lambda_\varepsilon X^\varepsilon$ are the unknowns, and the s Hankel matrices:

$$\mathbf{S}^{\langle t \rangle}_{i,j} \overset{\text{def}}{=} S_{t,i+j}, \qquad \forall i \in [n-k_t-\varepsilon-1), j \in [\varepsilon+1), t \in [s),$$

or more explicitly:

$$\mathbf{S}^{\langle t \rangle} = \begin{pmatrix} S_{t,0} & S_{t,1} & \dots & S_{t,\varepsilon} \\ S_{t,1} & S_{t,2} & \dots & S_{t,\varepsilon+1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{t,n-k_t-\varepsilon-1} & S_{t,n-k_t-\varepsilon} & \dots & S_{t,n-k_t-1} \end{pmatrix}, \quad \forall t \in [s), \tag{3.35}$$

can be written as follows:

$$\begin{pmatrix} \mathbf{S}^{\langle 0 \rangle} \\ \mathbf{S}^{\langle 1 \rangle} \\ \vdots \\ \mathbf{S}^{\langle s-1 \rangle} \end{pmatrix} \cdot \begin{pmatrix} \Lambda_\varepsilon \\ \vdots \\ \Lambda_1 \\ \Lambda_0 \end{pmatrix} = \mathbf{0}. \tag{3.36}$$

A unique solution of (3.36) up to a scalar factor of the system of linear homogeneous equations as in (3.36) is given if the nullity of the syndrome matrix $\left(\mathbf{S}^{\langle 0 \rangle} \; \mathbf{S}^{\langle 1 \rangle} \; \ldots \; \mathbf{S}^{\langle s-1 \rangle}\right)^T$ is one, i.e., the difference between the number of columns and the rank of $\left(\mathbf{S}^{\langle 0 \rangle} \; \mathbf{S}^{\langle 1 \rangle} \; \ldots \; \mathbf{S}^{\langle s-1 \rangle}\right)^T$.

The decoding fails if for a given ε, the syndrome matrix $\left(\mathbf{S}^{\langle 0 \rangle}, \mathbf{S}^{\langle 1 \rangle}, \ldots, \mathbf{S}^{\langle s-1 \rangle}\right)^T$ has not full rank. Furthermore, we assume that all syndrome sequences are long enough to form the required rows of length $\varepsilon + 1$ of the matrices $\mathbf{S}^{\langle 0 \rangle}, \mathbf{S}^{\langle 1 \rangle}, \ldots, \mathbf{S}^{\langle s-1 \rangle}$ as in (3.35). We obtain for the maximal decoding radius:

$$\begin{aligned}
\varepsilon + 1 - \operatorname{rank}(\mathbf{S}) &= 1 \\
\sum_{t=0}^{s-1} (n - k_t - \varepsilon) &= \varepsilon \\
sn - \sum_{t=0}^{s-1} k_t &= (s+1)\varepsilon \qquad (3.37) \\
\left\lfloor \frac{s}{s+1} \left(n - \frac{1}{s} \sum_{t=0}^{s-1} k_t \right) \right\rfloor &= \varepsilon.
\end{aligned}$$

We refer to [A-Kra03, Theorem 2] for further informations.

An efficient solution of the system of equations (3.36) can be obtained by multi-sequence shift-register synthesis as proposed by Schmidt and Sidorenko [I-SS06] or by a generalized EEA [A-FT89; A-ZW11].

3.5 Interpolation-Based Decoding of Generalized Reed–Solomon Codes

3.5.1 Overview

We describe the Guruswami–Sudan principle for GRS codes, as originally proposed in [A-GS99] for GRS and Algebraic-Geometry codes as an generalization of Sudan's original work [A-Sud97]. The recently published books of Roth [B-Rot06, Chapter 9], Moon [B-Moo05, Chapter 7.6], Justesen and Høholdt [B-JH04, Chapter 12] and Kabatiansky *et al.* [B-KKS05, Chapter 4.5] cover the Guruswami–Sudan procedure extensively. Furthermore, the survey papers of McEliece [I-McE03] and Augot [A-Aug04] give a substantial introduction to the Guruswami–Sudan principle for decoding GRS codes.

We give the main theorem of the interpolation step of Guruswami–Sudan in the case where all the points have same multiplicity m. Furthermore, we prove that the decoding radius of Guruswami–Sudan reaches the Johnson bound asymptotically.

A modification of the Guruswami–Sudan principle allows a new soft-decision variant for decoding GRS codes which is known as Kötter–Vardy algorithm [A-KV03a]. We give the basic idea in Subsection 3.5.3. Recently, Kötter and Vardy presented a generalization of the complexity-reduction technique called re-encoding [A-KMV11]. We describe the re-encoding technique in Chapter 5.

In Sub-section 3.5.4 we summarize some existing realizations for Guruswami–Sudan and Kötter–Vardy and compare them.

3.5.2 The Guruswami–Sudan Principle

We introduce the notations of the Guruswami–Sudan algorithm that are extensively used in Chapter 4 and Chapter 5.

Let us state the main theorem for the interpolation step of Guruswami–Sudan for GRS codes.

Theorem 3.17 (Guruswami–Sudan for GRS Codes [A-GS99, Theorem 8])
Let $\mathbf{c} = \text{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ be a codeword of a given $[n,k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ over $\mathbb{F}_q$. Let the positive integers τ, m and ℓ be given. Let $\mathbf{r}$ be the received word. Let

$$Q(X,Y) = Q_0(X) + Q_1(X)Y + \cdots + Q_\ell(X)Y^\ell,$$

be a polynomial in $\mathbb{F}_q[X,Y]$ such that:

C1) $Q^{[a,b]}(\alpha_i, r_i/\overline{v}_i) = 0, \quad \forall i \in [n)$ and $\forall a, b$ with $a+b<m$,

C2) $\text{wdeg}_{1,\text{k-1}}\, Q(X,Y) < m(n-\tau)$.

Then $(Y - f(X)) | Q(X,Y)$.

Proof The interpolation polynomial $Q(X,Y)$ satisfies $Q^{[a,b]}(\alpha_i, c_i/\overline{v}_i) = 0$ for $a+b<m$ and for at least $n-\tau$ positions (due to C1). According to Corollary 2.7 the polynomial $(X-\alpha_i)^m$ divides $Q(X+\alpha_i, Y+f(\alpha_i))$ for these $n-\tau$ error-free positions. However, $Q(X, f(X))$ has degree at most $m(n-\tau)-1$, so $Q(X,f(X))=0$ and therefore $(Y-f(X))|Q(X,Y)$. ■

There exists a non-zero interpolation polynomial $Q(X,Y)$ if the number of unknowns, i.e., the number of monomials of $Q(X,Y)$, is larger than the number of constraints, i.e., $\binom{m+1}{2}n$ linear homogeneous equation by Condition C1 of Theorem 3.17. From C2, we know that the number of monomials of each univariate polynomial $Q_t(X) \in \mathbb{F}_q[X]$ for all $t \in [\ell+1)$ is at most:

$$N_t \overset{\text{def}}{=} m(n-\tau) - t(k-1), \quad \forall t \in [\ell+1). \tag{3.38}$$

The list size is the integer ℓ such that $N_\ell > 0$ and $N_{\ell+1} \leq 0$. Therefore, we can state the following bound on the list size:

$$\ell < \frac{m(n-\tau)}{k-1} \leq \ell + 1. \tag{3.39}$$

Let us calculate an upper bound on the decoding radius τ of Theorem 3.17 (that coincides asymptotically with the Johnson radius [A-Joh62; A-Bas65]).

Lemma 3.18 (The Guruswami–Sudan Decoding Radius)
Let $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ be an $[n,k]_q$ GRS code. For a given non-zero multiplicity m, there exists a non-zero interpolation polynomial $Q(X,Y)$ as in Theorem 3.17, if the normalized decoding radius is:

$$\frac{\tau}{n} < 1 - \sqrt{\frac{(k-1)}{n}\left(1 + \frac{1}{m}\right)}.$$

Proof The number of unknowns, i.e., the number of coefficients of an interpolation polynomial $Q(X, Y)$ according to Theorem 3.17 is:

$$\sum_{t=0}^{\ell} N_t = \sum_{t=0}^{\ell} m(n-\tau) - t(k-1)$$
$$= (\ell+1)\left(m(n-\tau) - \frac{1}{2}\ell(k-1)\right).$$

With (3.39), we obtain:

$$\sum_{t=0}^{\ell} N_t > \frac{m(n-\tau)}{k-1}\left(m(n-\tau) - \frac{1}{2}m(n-\tau)\right)$$
$$\geq \frac{(m(n-\tau))^2}{2(k-1)}. \tag{3.40}$$

The number of unknowns as bounded in (3.40) should be greater than the constraints on $Q(X, Y)$ given by Theorem 3.17:

$$\frac{(m(n-\tau))^2}{2(k-1)} > \frac{1}{2}m(m+1)n$$
$$\Leftrightarrow (m(n-\tau))^2 > m(m+1)n(k-1)$$
$$\Leftrightarrow (n-\tau)^2 > n(k-1)\left(1+\frac{1}{m}\right). \tag{3.41}$$

Dividing (3.41) by n^2 leads to:

$$\left(1-\frac{\tau}{n}\right)^2 > \frac{(k-1)}{n}\left(1+\frac{1}{m}\right),$$
$$\frac{\tau}{n} < 1 - \sqrt{\frac{(k-1)}{n}\left(1+\frac{1}{m}\right)}. \tag{3.42}$$

■

Figure 3.5 shows the normalized decoding radius τ/n for the asymptotic case ($n \to \infty$) as a function of the code-rate $R = k/n$. For $m = 1$ we obtain for τ/n with (3.42) for the normalized decoding radius $1 - \sqrt{2R}$ and for $m \to \infty$ we get $1 - \sqrt{R}$.

3.5.3 Soft-Decision Decoding Based on Guruswami–Sudan

The interpolation-based decoding approach of Guruswami–Sudan can be modified for the case where soft-information is available. Guruswami and Sudan mentioned this as “weighted curve fitting” (see [A-GS99, Subsection III-D] and [B-Gur04, Subsection 6.2.10]). Kötter and Vardy provided a framework to translate the soft-information given by the channel into algebraic interpolation constraints. Therefore, the soft-decision variant based on the Guruswami–Sudan approach is referenced to as Kötter–Vardy algorithm and was first mentioned in the preprint [O-KV00] and published in [A-KV03a].

We focus on the resulting algebraic decoding problem and state it as a generalization of Theorem 3.17. Let $\boldsymbol{\alpha} = (\alpha_0\ \alpha_1\ \dots\ \alpha_{n-1})$ be the support of a given GRS code and let $\beta_0, \beta_1, \dots, \beta_{q-1}$ denote the q distinct elements of $\mathbb{F}_q$.

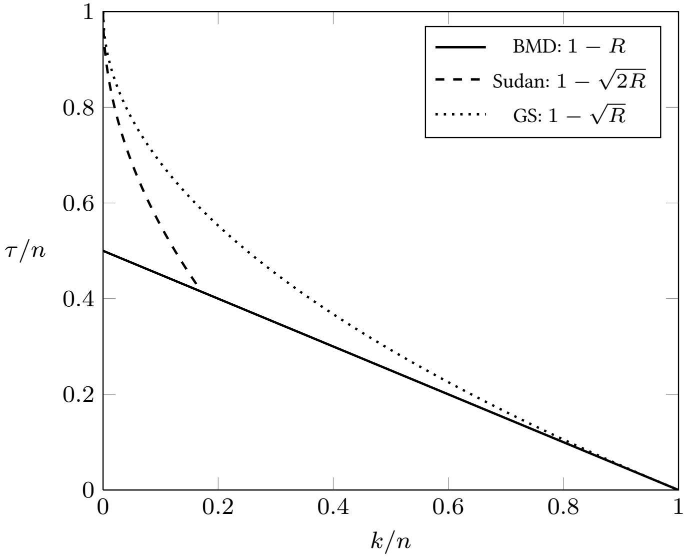


Figure 3.5: Illustration of the normalized decoding radius τ/n for BMD, Sudan ($m = 1$) and Guruswami–Sudan ($m \to \infty$) decoding as a function of the code-rate k/n for the asymptotic case ($n \to \infty$).

In the scenario of soft-information for Kötter–Vardy a $q \times n$ reliability matrix $\mathbf{P} = (P_{i,j})_{i \in [q)}^{j \in [n)}$, where $P_{i,j}$ is a real number between 0 and 1, is given, e.g., by the Euclidean distance of the received symbol to other points of the modulation scheme or by the inner code in a concatenated code, instead of "simply" a received vector with error/erasures. The entry $P_{i,j}$ gives the probability of the j-th symbol to be equal $\beta_i \in \mathbb{F}_q$.

We assume that a $q \times n$ multiplicity matrix $\mathbf{m} = (m_{i,j})_{i \in [q)}^{j \in [n)}$ with $m_{i,j} \in \mathbb{N}$ approximates the $q \times n$ reliability matrix given by the channel model. A native algorithm to obtain the multiplicity matrix from the channel probabilities is e.g., [A-KV03a, Algorithm A].

The number of constraints on the bivariate interpolation polynomial is based on the multiplicity matrix and we define the cost of such a matrix.

Definition 3.19 (Cost of a Multiplicity Matrix)
Given a $q \times n$ matrix $\mathbf{m} = (m_{i,j})_{i \in [q)}^{j \in [n)}$ with entries $m_{i,j} \in \mathbb{N}$, the cost of $\mathbf{m}$ is defined as:

$$\mathrm{Cost}(\mathbf{m}) \overset{\mathrm{def}}{=} \frac{1}{2} \sum_{i=0}^{q-1} \sum_{j=0}^{n-1} m_{i,j}(m_{i,j}+1). \tag{3.43}$$

To measure the "distance" of a received word to a codeword we introduce the score of a vector, when a multiplicity matrix is given.

Definition 3.20 (Score of a Vector)
Let $\beta_0, \beta_1, \dots, \beta_{q-1}$ be the distinct elements of $\mathbb{F}_q$ and let $\mathbf{v} = (v_0\, v_1\, \dots\, v_{n-1}) \in \mathbb{F}_q^n$. The score with respect to a given $q \times n$ multiplicity matrix $\mathbf{m} = (m_{i,j})_{i\in[q)}^{j\in[n)}$ is defined as:

$$\mathrm{Score}_{\mathbf{m}}(\mathbf{v}) \overset{\text{def}}{=} \sum_{\substack{i,j:\\ v_j=\beta_i}} m_{i,j}. \tag{3.44}$$

Definition 3.20 is equivalent to the inner product $\langle\, \mathbf{m}, |\mathbf{v}|\, \rangle$, where $|\mathbf{v}|$ is the $q \times n$ matrix $(v_{i,j})_{i\in[q)}^{j\in[n)}$, where $v_{i,j} = 1$ if $v_j = \beta_i$, and $v_{i,j} = 0$ otherwise (see [A-KV03a, Definition 4]).

Theorem 3.21 (Kötter–Vardy for GRS Codes [A-KV03a, Theorem 3])
Let $\mathbf{c} = \mathrm{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ be a codeword of a given $[n,k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ and let a $q \times n$ multiplicity matrix $\mathbf{m} = (m_{i,j})_{i\in[q)}^{j\in[n)}$ with $m_{i,j} \in \mathbb{N}$ be given. Let

$$Q(X,Y) = Q_0(X) + Q_1(X)Y + \dots + Q_\ell(X)Y^\ell,$$

be a polynomial in $\mathbb{F}_q[X,Y]$ such that:

C1) $Q^{[a,b]}(\alpha_j, \beta_i/\overline{v}_j) = 0, \quad \forall i \in [q), \forall j \in [n)$ and $\forall a, b$ with $a + b < m_{i,j}$,

C2) $\mathrm{wdeg}_{1,\text{k-1}}\, Q(X,Y) < \delta + 1$.

If $\mathrm{Score}_{\mathbf{m}}(\mathbf{c}) > \delta$, then $(Y - f(X))|Q(X,Y)$.

Proof Let $j(i)$ be such that $c_i = \beta_{j(i)}/\overline{v}_i$ for all $i \in [n)$. The interpolation polynomial $Q(X,Y)$ satisfies $Q^{[a,b]}(\alpha_i, c_i/\overline{v}_i) = 0$ for $a + b < m_{j(i),i}$ for all $i \in [n)$ (due to C1) and according to Corollary 2.7 the polynomial $(X - \alpha_i)^{m_{j(i),i}}$ divides $Q(X + \alpha_i, Y + f(\alpha_i))$. But $Q(X, f(X))$ has degree at most δ (due to C2), so $Q(X, f(X)) = 0$ and therefore $(Y - f(X))|Q(X,Y)$. ■

Such a non-zero interpolation polynomial exists if the number of coefficients of $Q(X,Y)$, i.e., $\sum_{t=0}^{\ell} \delta + 1 - t(k-1)$, is greater than $\mathrm{Cost}(\mathbf{m})$ and similar by Lemma 3.18 a maximal radius in case of given multiplicities $m_{i,j}$ can be derived (see [B-Gur04, Corollary 3.7, Section 3.4]).

With

$$N_t \overset{\text{def}}{=} \delta + 1 - t(k-1),$$

we get similar to Theorem 3.17 that ℓ is the largest integer, such that $N_\ell > 0$ and $N_{\ell+1} \leq 0$. Therefore, we have:

$$N_\ell > 0 \quad \Leftrightarrow \quad \ell = \left\lfloor \frac{\delta+1}{k-1} \right\rfloor.$$

Algorithm 3.4 is the interpolation-based soft-decision decoding variant for decoding an $[n,k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ with given multiplicity matrix $\mathbf{m}$. We initialize the two sets L and $\widetilde{L}$, which store the list of possible outputs, to zero. The first calculations (Line 1-3) are as derived in Theorem 3.21. After the interpolation step, we determine all roots of the polynomial $Q(X,Y)$ of the form $Y - f(X)$, where $\deg f(X) < k$ in Line 4 of Algorithm 3.4 and store them in $\widetilde{L}$. It is guaranteed that $|\widetilde{L}| \leq \ell$.

Algorithm 3.4: $L = \text{KOETTERVARDY}(\mathbf{m}, \overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$

```
Input: Parameters of $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$, Multiplicity matrix $\mathbf{m} = (m_{i,j})_{i \in [q]}^{j \in [n]} \in \mathbb{N}^{q \times n}$
Output: List $L = \{f_0(X), f_1(X), \dots\}$ or DECODING FAILURE
Initialize: $\widetilde{L} \leftarrow \emptyset$, $L \leftarrow \emptyset$
1 Calculate $\text{Cost}(\mathbf{m})$ according to (3.43)
2 Calculate minimal $\delta$, such that $\sum_{t=0}^{\lfloor(\delta+1)/(k-1)\rfloor} N_t > \text{Cost}(\mathbf{m})$
3 Determine $Q(X,Y)$ with $\text{wdeg}_{1,\text{k-1}} < \delta + 1$ and
  $Q^{[a,b]}(\alpha_j, \beta_i/\overline{v}_j) = 0, \quad \forall j \in [n], \forall i \in [q]$ and $\forall a, b$ with $a + b < m_{i,j}$
4 Find all roots $(Y - f_i(X)) | Q(X,Y)$ with $\deg f_i(X) < k$ and store them in $\widetilde{L}$
5 for $f(X) \in \widetilde{L}$ do
6     if $\text{Score}_{\mathbf{m}}(eval(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})) > \delta$ then
7         $L \leftarrow L \cup \{f(X)\}$
8 if $L = \emptyset$ then
9     Declare DECODING FAILURE
```

The true number of valid codewords is $|L|$ and we need to check the Score in Line 5 and Line 7 of Algorithm 3.4. This corresponds to the verification if

$$d\Big(\text{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha}), \mathbf{r}\Big) \leq \tau$$

in the hard-decision scenario.

3.5.4 Some Realizations of Guruswami–Sudan for Generalized Reed–Solomon Codes

Table 3.1 shows some existing realization of the interpolation step for list decoding GRS codes.

The properties of the interpolation step are compared with the original work of Sudan [A-Sud97] and Guruswami–Sudan [A-GS99]. The second column indicates if the interpolation multiplicity can be greater than one. When an adaption to different multiplicities was considered, it is marked in column three of Table 3.1. In the last column of Table 3.1, it is listed if the proposed algorithm takes advantage of a speed-up based on the Divide-and-Conquer principle.

Alekhnovich [I-Ale02; A-Ale05] uses a module minimization technique and applied it to the case of different multiplicities for each interpolation point.

Trifonov [A-Tri07; A-Tri10] proposed in his original work a fast iterative interpolation algorithm for Guruswami–Sudan GRS decoding, where all points have the same multiplicity. Trifonov give the complexity of his approach, but not how it is decreased by considering the re-encoding transformation.

Wu [A-Wu08] uses the output of the Berlekamp–Massey algorithm and formulates a trivariate interpolation problem. For high-rate GRS codes a reduction of the necessary interpolation multiplicity, compared to the original interpolation problem, is achieved.

Beelen *et al.* [A-BHNW13; O-Nie13] modified Wu's approach and used the Extended Euclidean Algorithm as preliminary step for the rational interpolation. Lee-O'Sullivan [I-LO06; A-LO09] formulated the interpolation problem in terms of Gröbner bases.

Beelen and Brander [O-BH08b; A-BB10b] combined the idea of using a Key Equation (based on [O-BH08a; I-AZ08]) and Alekhnovich's algorithm.

Reference	**Multiplicity** $\mathbf{m > 1}$	**Soft**	**Re-Encoding**	**Divide & Conquer**
Sudan [A-Sud97]	No	No	No	No
Guruswami–Sudan [A-GS99]	Yes	(Yes)	No	No
Alekhnovich [I-Ale02; A-Ale05]	Yes	Yes	No	Yes
Trifonov [A-Tri07; A-Tri10]	Yes	No	Yes	No
Wu [A-Wu08]	Yes	No	No	No
Beelen–Brander [O-BH08b; A-BB10b]	Yes	No	No	Yes
Lee–O'Sullivan [I-LO06; A-LO09]	Yes	Yes	No	Yes
Kötter–Vardy [A-KMV11]	Yes	Yes	Yes	No
Roth–Ruckenstein [A-RR00]	No	No	No	No

Table 3.1: Some realizations of the interpolation step of Guruswami–Sudan for decoding GRS codes and their properties.

Kötter and Vardy developed in [A-KMV11] (and in forgoing conference publications [I-KV03b; I-KMVA03]) the re-encoding transformation for the scenario of different multiplicities as in Theorem 3.21. In [A-KMV11] Kötter and Vardy adapted the so-called Kötter algorithm [O-Kö96b; A-Kö96a], which solves the original bivariate interpolation, for the reduced problem after re-encoding (see Chapter 5).

Kuijper *et al.* [A-KP04; A-AK11] investigated the Guruswami–Sudan principle for GRS codes from a system-theoretic point of view and gave a Gröbner basis description.

Roth and Ruckenstein proposed in [A-RR00; O-Ruc01], besides a fast root-finding procedure, an Extended Key Equation for the interpolation step of multiplicity $m = 1$.

In Chapter 4, we generalize the idea of a Key Equation of Roth and Ruckenstein [A-RR00] to the case of higher multiplicity. We outline the univariate reformulation in Chapter 5 for the soft-decision scenario as in Theorem 3.21 and after the re-encoding transformation technique of [A-KMV11].

4

"Mathematicians are a sort of Frenchmen; if you talk to them, they translate it into their own language, and then it is immediately something quite different."

Johann Wolfgang v. Goethe (1749–1832)

Key Equations for Decoding of Generalized Reed–Solomon Codes Beyond Half the Minimum Distance

We consider two different approaches capable to decode GRS codes beyond half the minimum distance in this chapter. In Section 4.1, a decoding principle for GRS codes, based on a virtual extension to an IGRS codes, is introduced. Schmidt, Sidorenko and Bossert [I-SSB06; O-Sch07; A-SSB10] proposed this approach in 2006 reaching a similar (but not equal) decoding radius for RS codes as Sudan's interpolation-based list decoding approach [A-Sud97] (multiplicity $m > 1$). Our contribution covers the generalization of the algorithm to GRS codes and a small modification of the bound on the failure probability (see [I-ZWB12a]).

Further, we modify the derivation of Roth and Ruckenstein [I-RR98; A-RR00; O-Ruc01] for Sudan's approach in Section 4.2. Roth–Ruckenstein denoted their result "Extended Key Equation". This modified reformulation gives us a proper basis for a comparison to our reformulation of the Guruswami–Sudan approach. We present the derivation of the Key Equation for $m = 1$ and outline the adaption of the FIA for the corresponding non-reduced (n instead of τ) set of homogeneous equations.

Section 4.3 covers the univariate reformulation of the interpolation-based Guruswami–Sudan approach and the adjustment of the FIA to the obtained structured system of homogeneous linear equations. The univariate reformulation was first published in [I-AZ08] and the adjustment of the FIA in [I-ZGB09], summarized in the journal version [A-ZGA11]. In addition, we give further directions in Section 4.4 in order to find an explicit syndrome expression of the Guruswami–Sudan interpolation problem.

Beelen and Høholdt in [O-BH08a; I-HB08] used a reformulation of the Guruswami–Sudan interpolation problem in terms of matrices. A fast algorithm, which uses Alekhnovich's [I-Ale02; A-Ale05] module interpretation, was given by Beelen and Brander [A-BB10b; A-BB10a; O-Bra10].

We give open research problems in Section 4.5 and conclude this chapter.

4.1 A Unique Decoding Approach for Generalized Reed–Solomon Codes Beyond Half the Minimum Distance

4.1.1 Basic Idea

In 2006, Schmidt, Sidorenko and Bossert [I-SSB06; O-Sch07; A-SSB10] proposed an decoding approach for RS codes with a similar decoding radius as Sudan's interpolation-based list decoding approach. We recall this idea in Subsection 4.1.2 and describe it for the slightly more general case of GRS codes. Clearly, the presentation for GRS codes does not give new insights, but we use it as basis to draw the connection to the univariate reformulation of Sudan's principle by Roth and Ruckenstein in Section 4.2, which leads to the same explicit expression for the syndromes. Furthermore, we provide a slight generalization of the bound of the failure probability of [I-SSB06; A-SSB10].

4.1.2 Virtual Extension of a Generalized Reed–Solomon Code to an Interleaved Generalized Reed–Solomon Code

We generalize the scheme of [I-SSB06; A-SSB10] to the case of IGRS codes (see Definition 2.34) and give the corresponding parameters. Let $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ be an $[n, k]_q$ GRS code with code-rate $R < 1/3$. We show that $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ can be virtually extended to an IGRS code of interleaving order $s > 1$. This specific IGRS code is denoted by $\mathcal{VGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k, s)$, where $\overline{\boldsymbol{v}}$ and $\boldsymbol{\alpha}$ are the original parameters of the given GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ and the parameter s denotes the order of (virtual) interleaving. Let a vector $\mathbf{c} = (c_0\, c_1\, \dots\, c_{n-1}) \in \mathbb{F}_q^n$ and an integer $t > 1$ be given. Let the following mapping be defined as:

$$\begin{array}{rcl} \text{pow}: \quad (\mathbb{F}_q^n, \mathbb{N}) & \to & \mathbb{F}_q^n \\ \big((c_0\, c_1\, \dots\, c_{n-1}), t\big) & \mapsto & \text{pow}\big((c_0\, c_1\, \dots\, c_{n-1}), t\big) = (c_0^t\, c_1^t\, \dots\, c_{n-1}^t). \end{array}$$

The virtual IGRS code is obtained as follows.

Definition 4.1 (Virtual Extension to an IGRS code)
Let $\mathbf{c} = \text{eval}(f(X), \overline{\boldsymbol{v}}_0, \boldsymbol{\alpha})$ be a codeword of an $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k)$ as in Definition 2.28. Let $s \in \mathbb{N}$ with $s > 1$ be a given (virtual) interleaving order. Then, the VGRS code is:

$$\mathcal{VGRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k, s) \overset{\text{def}}{=} \left\{ \begin{pmatrix} \mathbf{c} \\ \text{pow}\,(\mathbf{c}, 2) \\ \vdots \\ \text{pow}\,(\mathbf{c}, s) \end{pmatrix} : \quad \mathbf{c} \in \mathcal{GRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k) \right\}.$$

We derive the parameters of the specific IGRS code. Let $\overline{\boldsymbol{v}} = (\overline{\boldsymbol{v}}_0\, \overline{\boldsymbol{v}}_1\, \dots\, \overline{\boldsymbol{v}}_{s-1}) \in \mathbb{F}_q^{sn}$ with

$$\overline{\boldsymbol{v}}_t \overset{\text{def}}{=} \text{pow}\,(\overline{\boldsymbol{v}}, t+1)\,, \quad \forall t \in [1, s).$$

Let $\mathbf{k} = (k_0\, k_1\, \dots\, k_{s-1})$ with

$$k_t \overset{\text{def}}{=} (t+1)(k-1)+1, \quad \forall t \in [s).$$

The virtually extended GRS code $\mathcal{VGRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k, s)$ of extension order s can be seen as a sub-code of an IGRS code, i.e.:

$$\mathcal{VGRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k, s) \subseteq \mathcal{IGRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, \mathbf{k}),$$

or more explicitly:

$$\mathcal{VGRS}(\overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}, k, s) = \left\{ \begin{pmatrix} \text{eval}(f(X), \overline{\boldsymbol{v}}_0, \boldsymbol{\alpha}) \\ \text{eval}(f(X)^2, \overline{\boldsymbol{v}}_1, \boldsymbol{\alpha}) \\ \vdots \\ \text{eval}(f(X)^s, \overline{\boldsymbol{v}}_{s-1}, \boldsymbol{\alpha}) \end{pmatrix} : \begin{array}{c} f(X) \in \mathbb{F}_q[X] \\ \deg f(X)^t < k_t \\ \forall t \in [s) \end{array} \right\}.$$

The following theorem shows the relation between the scalar factors $\overline{\boldsymbol{v}}_0, \overline{\boldsymbol{v}}_1, \dots, \overline{\boldsymbol{v}}_{s-1} \in \mathbb{F}_q^n$ and the column multipliers $\boldsymbol{v}_0, \boldsymbol{v}_1, \dots, \boldsymbol{v}_{s-1} \in \mathbb{F}_q^n$ of the s GRS sub-codes of the virtually created IGRS code.

> **Theorem 4.2 (Column Multipliers)**
> Let $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ be an $[n, k]_q$ GRS code as in Definition 2.28 and let $\boldsymbol{v} = (v_0 \, v_1 \, \dots \, v_{n-1})$ be the column multipliers as in Lemma 2.29.
>
> Furthermore, let $\mathcal{VGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k, s)$ be the IGRS code as in Definition 4.1. Then, the column multipliers of the t-th $[n, t(k-1)+1]_q$ sub-code $\mathcal{GRS}(\overline{\boldsymbol{v}}_t, \boldsymbol{\alpha}, t(k-1)+1)$ are given by
>
> $$v_{t,i} = \frac{v_i}{(\overline{v}_i)^t}, \quad \forall i \in [n], t \in [1, s). \tag{4.1}$$

Proof We have for the t-th sub-code $\mathcal{GRS}(\overline{\boldsymbol{v}}_t, \boldsymbol{\alpha}, k_t)$ of the virtually created IGRS code $\mathcal{VGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k, s)$ that $\overline{v}_{t,i} = (\overline{v}_i)^{t+1}$, $\forall i \in [n], t \in [1, s)$. With Lemma 2.29 we get:

$$\begin{aligned} \overline{v}_{t,i} &= \Big(v_{t,i} L_i(\alpha_i)\Big)^{-1} \\ (\overline{v}_i)^{t+1} &= \Big(v_i L_i(\alpha_i)\Big)^{-(t+1)} \\ &\Leftrightarrow \\ v_{t,i} &= \frac{(v_i)^{t+1} L_i(\alpha_i)^{t+1}}{L_i(\alpha_i)} = (v_i)(v_i)^t L_i(\alpha_i)^t = \frac{v_i}{(\overline{v}_i)^t}. \end{aligned}$$ ■

4.1.3 Decoding and Failure Probability

We consider the error-only unique decoding beyond half the minimum distance in the following. Let $\mathbf{r} = \mathbf{c} + \mathbf{e}$, where $\mathbf{c} = \mathrm{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ is a codeword of an $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ and $\mathbf{e} \in \mathbb{F}_q^n$. Let an integer s with $s > 1$ be given and we assume that $(t+1)(k-1)+1 < n$ for all $t \in [s)$. Let

$$\mathbf{r}_t \overset{\text{def}}{=} \mathrm{pow}(\mathbf{r}, t+1), \quad \forall t \in [s), \tag{4.2}$$

and $\mathbf{e}_0 \overset{\text{def}}{=} \mathbf{e}$. With (4.2), we obtain a virtually created

1. burst error in $\mathbb{F}_q^s$ and
2. a VIRS code as defined in Definition 4.1 of interleaving order s.

More explicitly, we have with $\mathbf{r}_{t-1} = (r_{t-1,0} \, r_{t-1,1} \, \dots \, r_{t-1,n-1})$, where each component can be decomposed as follows:

$$\begin{aligned} r_{t-1,i} &= r_i^t \\ &= (c_i + e_i)^t \\ &= c_i^t + e_{t-1,i}, \quad \forall i \in [n], \forall t \in [1, s+1), \end{aligned}$$

where:

$$e_{t-1,i} \overset{\text{def}}{=} \sum_{j=1}^{t} c_i^{t-j} \binom{t}{j} e_i^j, \quad \forall i \in [n], \forall t \in [1, s+1). \tag{4.3}$$

KEs for GRS Decoding

The virtual created error $e_{t-1,i}$ is zero if e_i is zero for all $t \in [2, s+1)$ and for all $i \in [n)$. The virtual burst error is $(e_{0,i}\, e_{1,i}\, \dots\, e_{s-1,i})^T \in \mathbb{F}_q^s$. Note that $e_{0,i} \neq 0$ does not imply that $e_{t,i} \neq 0$ for $t \in [1, s)$. The virtual burst error can be used to increase the error-correcting capability of a given low-rate GRS codes (see [I-SSB06; A-SSB10]). The decoding radius of a virtual created IGRS code $\mathcal{VGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k, s)$ is the same as in [A-SSB10, Equation (10)] for RS codes. Let $\varepsilon = \mathrm{wt}(\mathbf{e}) = |E|$ and let:

$$\overline{k} \overset{\mathrm{def}}{=} \frac{1}{s}\sum_{t=0}^{s-1} k_t = \frac{1}{s}\sum_{t=0}^{s-1}\Big((t+1)(k-1)+1\Big) = \frac{(k-1)(s+1)}{2} + 1.$$

We obtain from (3.37) a maximal decoding radius:

$$\tau_{\mathsf{VGRS}} \overset{\mathrm{def}}{=} \left\lfloor \frac{s}{s+1}\left(n - \overline{k}\right)\right\rfloor, \tag{4.4}$$

where we choose s such that τ_{VGRS} is maximized (for detailed analysis see [A-SSB10]).

We obtain s Key Equations as in (3.13) with a common error-locator polynomial:

$$\Lambda(X) \cdot S_t(X) \equiv \Omega_t(X) \mod X^{n-k_t}, \quad \forall t \in [s), \tag{4.5}$$

where $\deg \Omega_t(X) < \varepsilon$ holds for all $t \in [s)$. The syndromes are:

$$S_{t,i} = \sum_{j=0}^{n-1} r_j^{t+1} \frac{\upsilon_j}{(\overline{\upsilon}_j)^t}\alpha_j^i, \quad \forall t \in [s), i \in [n-k_t). \tag{4.6}$$

Let the s $(n - k_t - \varepsilon) \times \varepsilon$ Hankel matrices be:

$$\mathbf{S}^{\langle t \rangle} = \big(S_{i,j}^{\langle t \rangle}\big)_{i \in [n-k_t-\varepsilon)}^{j \in [\varepsilon)} = \big(S_{t,i+j}\big)_{i \in [n-k_t-\varepsilon)}^{j \in [\varepsilon)}, \quad \forall t \in [s). \tag{4.7}$$

Let

$$\mathbf{S} \overset{\mathrm{def}}{=} (\mathbf{S}^{\langle 0 \rangle}\ \mathbf{S}^{\langle 1 \rangle}\ \dots\ \mathbf{S}^{\langle s-1 \rangle})^T$$

be the $s(n - \overline{k} - \varepsilon) \times \varepsilon$ syndrome matrix, where each sub-matrix $\mathbf{S}^{\langle i \rangle}$ is an $(n - k_i - \varepsilon) \times \varepsilon$ Hankel matrix as defined in (4.7).

We search a unique solution for the error-locator polynomial $\Lambda(X)$ of the virtually extended GRS code. Therefore, a decoding failure is declared if the system of equations (3.36) has more than one solution. In the following, we derive an upper bound on the failure probability for $s = 2$, which is the same as for the virtual extension of the RS codes used in [A-SSB10] and therefore independent of the column multipliers of the GRS code. We consider the corresponding heterogeneous system of equation with ε unknowns for the analysis of the failure probability.

We bound the probability that the $s(n - \overline{k} - \varepsilon) \times \varepsilon$ syndrome matrix $\mathbf{S}$ does not have full rank ε and denote the failure probability, if ε errors occurred, by:

$$P_f(\varepsilon) \leq P\Big\{(\mathrm{rank}(\mathbf{S}) < \varepsilon) \mid (|E| = \varepsilon)\Big\}.$$

Let us recall [A-SSB10, Theorem 3] as an upper bound on the failure probability for virtual interleaving order $s = 2$.

Theorem 4.3 (Upper Bound on the Failure Probability [A-SSB10, Theorem 3])
Let $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ be virtually extended to an IGRS code $\mathcal{VGRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k, s)$ of extension order $s = 2$ as in Definition 4.1.

Let a codeword of a given $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ code be corrupted by an error of weight $\varepsilon \leq \tau_{\mathsf{VGRS}}$, where τ_{VGRS} is as in (4.4) for $s = 2$.

For decoding we solve the system of equations from (3.36). The probability for a decoding failure is upper bounded by:

$$P_f(\varepsilon) \leq \left(\frac{q}{q-1} + \frac{1}{q}\right)^{\varepsilon} \cdot \frac{q^{-3(\tau_{\mathsf{VGRS}}-\varepsilon)}}{q-1}.$$

Proof As in the proof of [A-SSB10, Theorem 3], this is equivalent to the case that there exists a non-zero vector $\mathbf{u} \in \mathbb{F}_q^{\varepsilon}$, such that

$$\exists\, \mathbf{u} \neq \mathbf{0} \ : \ \mathbf{S}^{\langle t \rangle} \cdot \mathbf{u}^T = \mathbf{0}, \quad \forall t \in [s). \tag{4.8}$$

Each syndrome matrix can be decomposed into five matrices (in [A-SSB10], the decomposition consists only of four matrices):

$$\mathbf{S}^{\langle t \rangle} = \mathbf{H}^{\langle t \rangle} \cdot \overline{\boldsymbol{v}}_t \cdot \mathbf{F}^{\langle t \rangle} \cdot \mathbf{D} \cdot \mathbf{V}, \quad \forall t \in [s),$$

where $\mathbf{D}$ and $\mathbf{V}$ are the same full-rank $\varepsilon \times \varepsilon$ matrices as in [A-SSB10, Proof of Theorem 3] and

$$\begin{aligned}
\mathbf{H}^{\langle t \rangle} &= \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_{j_0} & \alpha_{j_1} & \cdots & \alpha_{j_{\varepsilon-1}} \\ \alpha_{j_0}^2 & \alpha_{j_1}^2 & \cdots & \alpha_{j_{\varepsilon-1}}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{j_0}^{n-k_t-\varepsilon-1} & \alpha_{j_1}^{n-k_t-\varepsilon-1} & \cdots & \alpha_{j_{\varepsilon-1}}^{n-k_t-\varepsilon-1} \end{pmatrix}, \quad \forall t \in [s), \\
\overline{\boldsymbol{v}}_t &= \operatorname{diag}\left(\overline{v}_{t,j_0},\ \overline{v}_{t,j_1},\ \ldots,\ \overline{v}_{t,j_{\varepsilon-1}}\right), \quad \forall t \in [s), \\
\mathbf{F}^{\langle t \rangle} &= \operatorname{diag}\left(e_{t,j_0},\ e_{t,j_1},\ \ldots,\ e_{t,j_{\varepsilon-1}}\right), \quad \forall t \in [s),
\end{aligned}$$

where e_{t,j_ν} is as in (4.3). Since $\overline{\boldsymbol{v}}_t$ and $\mathbf{F}^{\langle t \rangle}$ are both diagonal matrices, $\overline{\boldsymbol{v}}_t \mathbf{F}^{\langle t \rangle} = \mathbf{F}^{\langle t \rangle} \overline{\boldsymbol{v}}_t$. The matrices $\overline{\boldsymbol{v}}_t, \mathbf{D}, \mathbf{V}$ are nonsingular and there is a one-to-one mapping from $\mathbf{u}$ to $\mathbf{v}_t$, where $\mathbf{v}_t^T = \overline{\boldsymbol{v}}_t \cdot \mathbf{D} \cdot \mathbf{V} \cdot \mathbf{u}^T$. Hence, (4.8) is equivalent to

$$\exists\, \mathbf{v}_t \neq \mathbf{0} : \quad \mathbf{H}^{\langle t \rangle} \cdot \mathbf{F}^{\langle t \rangle} \cdot \mathbf{v}_t^T = \mathbf{0}, \quad \forall t \in [s). \tag{4.9}$$

■

This is similar to [A-SSB10, Proof of Theorem 3, Equation (22)] and using Lemma 4.4 for arbitrary q, which is an extension of [A-SSB10, Lemma 4] to arbitrary fields, the rest of the proof is analog.

Lemma 4.4 (Independence for q-ary fields)
Let v, e, c be three non-zero elements in $\mathbb{F}_q$. Then, the set

$$V = \left\{ \begin{pmatrix} v \cdot e \\ v \cdot e_2 \end{pmatrix} \ : \ v, e \in \mathbb{F}_q^* \right\},$$

with $e_2 = 2c \cdot e + e^2$, forms the set of all full weight vectors of length two, i.e.:

$$V = \left\{ \mathbf{v} \in \mathbb{F}_q^2 : \mathrm{wt}(\mathbf{v}) = 2 \right\} = \left(\mathbb{F}_q^*\right)^2.$$

Proof It is sufficient to show that all $(q-1)^2$ possible vectors $\mathbf{v} \in V$ are pairwise different. For any fixed $c \in \mathbb{F}_q^*$, consider two vectors $\mathbf{v}, \overline{\mathbf{v}} \in V$, and assume that $\mathbf{v} = \overline{\mathbf{v}}$, then

$$v \cdot e = \overline{v} \cdot \overline{e}, \tag{4.10}$$

$$v \cdot e_2 = \overline{v} \cdot \overline{e}_2. \tag{4.11}$$

Dividing (4.11) by (4.10) yields:

$$\frac{e_2}{e} = \frac{\overline{e}_2}{\overline{e}}$$
$$\frac{2c \cdot e + e^2}{e} = \frac{2c \cdot \overline{e} + \overline{e}^2}{\overline{e}}.$$

Therefore, $2c + e = 2c + \overline{e}$ and hence, $e = \overline{e}$. Inserting this into (4.10), we obtain $v = \overline{v}$. Thus, for any c, two different pairs $(v, e) \neq (\overline{v}, \overline{e})$ always result in two different vectors $\mathbf{v}, \overline{\mathbf{v}}$. ■

Thus, the upper bound on the failure probability is independent of the column multipliers and in particular independent of using GRS codes or the normalized RS codes from [A-SSB10].

4.2 Key Equation for the Sudan Principle

4.2.1 Modified Univariate Reformulation

In this section, we recall parts of the work of Roth and Ruckenstein [I-RR98; A-RR00; O-Ruc01] for the interpolation step of the Sudan [A-Sud97] principle, i.e., a special case of Theorem 3.17 where the interpolation multiplicity $m = 1$. We present here a slightly modified version of [A-RR00], to see the generalization of our reformulation of the Guruswami–Sudan case, where the interpolation multiplicity is $m > 1$. As in Theorem 3.17, let $\mathbf{c} = \text{eval}(f(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ be a codeword of an $[n,k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ and let $\mathbf{r} = \mathbf{c} + \mathbf{e}$ be the received vector. The aimed decoding radius is denoted by τ and the corresponding list size is ℓ.

Similar to Lemma 3.4 for BMD decoding, Roth and Ruckenstein [A-RR00] proved the following.

> **Lemma 4.5 (Univariate Reformulation of Sudan [A-RR00, Lemma 3.1])**
> Let $R(X) \in \mathbb{F}_q[X]$ be the Lagrange interpolation polynomial, such that $R(\alpha_i) = r_i/\overline{v}_i, \ \forall i \in [n)$ (as in Theorem 2.2) with $\deg R(X) < n$. Let $L(X) = \prod_{i=0}^{n-1}(X - \alpha_i)$ as in (2.1).
>
> For given parameters n, k, ℓ, τ, the bivariate interpolation polynomial
>
> $$Q(X,Y) = \sum_{t=0}^{\ell} Q_t(X) Y^t$$
>
> satisfies Conditions C1 and C2 of Theorem 3.17 for a multiplicity $m = 1$ if and only if there exists a univariate polynomial $B(X) \in \mathbb{F}_q[X]$ of degree smaller than $\ell(n-k) - \tau$, such that
>
> $$Q(X, R(X)) = B(X) \cdot L(X). \tag{4.12}$$

For the proof see [A-RR00, Lemma 3.1] or the proof of Theorem 4.10 for $m = 1$. We modify the reformulation in the following. Let as in (3.38) for $m = 1$:

$$N_t \overset{\text{def}}{=} n - \tau - t(k-1), \quad \forall t \in [\ell + 1). \tag{4.13}$$

Define the polynomials as:

$$\begin{aligned}
\overline{R}(X) &\overset{\text{def}}{=} X^{n-1}R(X^{-1}),\\
\overline{L}(X) &\overset{\text{def}}{=} X^{n}L(X^{-1}) = \prod_{i=0}^{n-1}(1-\alpha_i X),\\
\overline{B}(X) &\overset{\text{def}}{=} X^{\ell(n-k)-\tau-1}B(X^{-1}),\\
\Lambda_t(X) &\overset{\text{def}}{=} X^{N_t-1}Q_t(X^{-1}), \quad \forall t \in [\ell+1).
\end{aligned}$$

Note that, these polynomials are not necessarily the reciprocal polynomials, because, e.g., for the received polynomial the degree can be smaller than $n-1$.

Reverting the coefficients of both sides of (4.12) leads to:

$$X^{n-\tau+\ell(n-k)-1}\sum_{t=0}^{\ell} Q_t(X^{-1})R(X^{-1})^t = X^{n-\tau+\ell(n-k)-1}B(X^{-1})L(X^{-1}) \tag{4.14}$$

and inserting $\overline{R}(X), \Lambda_0(X), \Lambda_1(X), \dots, \Lambda_\ell(X), \overline{L}(X)$ and $\overline{B}(X)$ into (4.14) gives us (as in [A-RR00]):

$$\sum_{t=0}^{\ell} \Lambda_t(X)X^{(\ell-t)(n-k)}\overline{R}(X)^t = \overline{B}(X)\cdot\overline{L}(X). \tag{4.15}$$

Let the polynomials $U_t(X)$ and $W_t(X)$ in $\mathbb{F}_q[X]$ be such that:

$$R(X)^t = U_t(X)L(X) + W_t(X), \quad \forall t \in [1, \ell+1), \tag{4.16}$$

where $W_t(X)$ is the remainder of the division of $R(X)^t$ by $L(X)$ and has degree smaller than n. Reverting (4.16) leads to:

$$\begin{aligned}
X^{t(n-1)}R(X^{-1})^t &= \left(X^{n-1}R(X^{-1})\right)^t\\
&= \overline{R}(X)^t = \overline{U}_t(X)\overline{L}(X) + X^{(t-1)(n-1)}\overline{W}_t(X),
\end{aligned} \tag{4.17}$$

where:

$$\overline{U}_t(X) = X^{t(n-1)-n}U_t(X^{-1}), \quad \forall t \in [1, \ell+1), \tag{4.18}$$

$$\overline{W}_t(X) = X^{n-1}W_t(X^{-1}), \quad \forall t \in [1, \ell+1). \tag{4.19}$$

Now let the $\ell+1$ formal power series $S_0^\infty(X), S_1^\infty(X), \dots, S_\ell^\infty(X)$ be defined as:

$$S_t^\infty(X) \overset{\text{def}}{=} \frac{\overline{W}_t(X)}{\overline{L}(X)}, \quad \forall t \in [1, \ell+1), \tag{4.20}$$

$$S_0^\infty(X) \overset{\text{def}}{=} \frac{X^{n-1}}{\overline{L}(X)}. \tag{4.21}$$

Clearly $S_0^\infty(X)$ does not depend on the received word.

The definition of $S_1^\infty(X), S_2^\infty(X), \dots, S_\ell^\infty(X)$ is equivalent to the one of [A-RR00]. In the following, we state the definition of the Sudan syndromes (compared to [A-RR00, Proposition 4.1.]), which we need to solve the full system of n homogeneous linear equations.

Lemma 4.6 (Modified Syndromes for Sudan Reformulation)
Let $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$ be an $[n, k]_q$ GRS code and let $\boldsymbol{\upsilon} = (\upsilon_0\, \upsilon_1\, \dots\, \upsilon_{n-1})$ denote its column multipliers as in Definition 2.28. Let the ℓ power series

$$S_t^{\infty}(X) = \sum_{i=0}^{\infty} S_{t,i} X^i, \quad \forall t \in [1, \ell+1)$$

be defined as in (4.20). Let $\mathbf{r} = (r_0\, r_1\, \dots\, r_{n-1}) = \mathbf{c} + \mathbf{e}$ be the received word in $\mathbb{F}_q^n$, where $\mathbf{c} \in \mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$. Then, the syndrome coefficients are given by:

$$S_{t,i} = \sum_{j=0}^{n-1} r_j^t \frac{\upsilon_j}{\overline{\upsilon}_j^{t-1}} \alpha_j^i, \quad \forall i \in [n + N_t), t \in [1, \ell+1). \tag{4.22}$$

Proof From (4.16) we have

$$R(\alpha_j)^t = r_j^t = W_t(\alpha_j), \quad \forall j \in [n), \forall t \in [1, \ell+1).$$

and for the reciprocal of $W_t(X)$, we obtain similar to (3.9) by standard univariate Lagrange interpolation as in (2.1) the following explicit expression:

$$\overline{W}_t(X) = \sum_{j=0}^{n-1} \left(\frac{r_j}{\overline{\upsilon}_j} \right)^t L_j(\alpha_j)^{-1} \prod_{\substack{i=0 \\ i \neq j}}^{n-1} (1 - \alpha_i X).$$

Thus, the explicit form of the formal power series defined in (4.20) is

$$\begin{aligned} S_t^{\infty}(X) = \sum_{i=0}^{\infty} S_{t,i} X^i &= \frac{\overline{W}_t(X)}{\overline{L}(X)} \\ &= \frac{\sum_{j=0}^{n-1} \left(\frac{r_j}{\overline{\upsilon}_j} \right)^t L_j(\alpha_j)^{-1} \prod_{\substack{i=0 \\ i \neq j}}^{n-1} (1 - \alpha_i X)}{\prod_{j=0}^{n-1} (1 - \alpha_j X)} \\ &= \sum_{i=0}^{\infty} \sum_{j=0}^{n-1} \left(\frac{r_j}{\overline{\upsilon}_j} \right)^t L_j(\alpha_j)^{-1} \alpha_j^i X^i, \end{aligned}$$

and with (2.29), that is

$$S_{t,i} = \sum_{j=0}^{n-1} r_j^t \frac{\upsilon_j}{\overline{\upsilon}_j^{t-1}} \alpha_j^i, \quad \forall t \in [1, \ell+1).$$

■

The syndromes are exactly the same as the ones for the virtually created IGRS code as in (4.6).

Inserting (4.16), (4.20) and (4.21) into (4.15) leads to:

$$\begin{aligned} &\Lambda_0(X) X^{\ell(n-k)-(n-1)} S_0^{\infty}(X) \overline{L}(X) \\ &\quad + \sum_{t=1}^{\ell} \Lambda_t(X) X^{(\ell-t)(n-k)} \Big(\overline{U}_t(X) \overline{L}(X) + X^{(t-1)(n-1)} S_t^{\infty}(X) \overline{L}(X) \Big) \\ &\qquad \equiv \overline{B}(X) \cdot \overline{L}(X) \mod X^{n-\tau+\ell(n-k)}. \end{aligned}$$

Simplifying, we obtain:

$$\Lambda_0(X)X^{\ell(n-k)-(n-1)}S_0^{\infty}(X)+\sum_{t=1}^{\ell}\Lambda_t(X)X^{(\ell-t)(n-k)}X^{(t-1)(n-1)}S_t^{\infty}(X) \\ \equiv \widetilde{\Omega}(X) \mod X^{n-\tau+\ell(n-k)}, \tag{4.23}$$

where

$$\widetilde{\Omega}(X) = \overline{B}(X) - \sum_{t=1}^{\ell}\Lambda_t(X)X^{(\ell-t)(n-k)}\overline{U}_t(X) \tag{4.24}$$

has degree smaller than $N_t + (\ell - t)(n-k) + t(n-1) - n = \ell(n-k) - \tau$. Instead of dividing by $X^{(\ell-1)(n-k)}$ (as in [A-RR00]) we divide (4.23) by $X^{\ell(n-k)-(n-1)}$ and with:

$$(\ell - t)(n-k) + (t-1)(n-1) = \ell(n-k) - (n-1) + t(k-1),$$

we obtain:

$$\Lambda_0(X)S_0(X) + \sum_{t=1}^{\ell}\Lambda_t(X)X^{t(k-1)}S_t(X) \equiv \Omega(X) \mod X^{2n-\tau-1}, \tag{4.25}$$

where $\deg \Omega(X) < n - \tau - 1$.

We omit the infinity indexes for the syndrome polynomials, since we bound their degrees to $N_t + n$ by Definition 4.6. Let us draw the connection to the Extended Key Equation of Roth–Ruckenstein [A-RR00, Equation (24)] at this point. Let us omit the $n - \tau$ highest terms of (4.25) and thus consider the equation modulo X^{n-1}. Then $\Lambda_0(X)S_0(X)$ disappears, because from (4.21) we know that $\Lambda_0(X)S_0(X)$ is a multiple of X^{n-1}.

Both sides of (4.25) are divisible by X^{k-1} and we obtain [A-RR00, Equation (24)]:

$$\sum_{t=1}^{\ell}\Lambda_t(X)X^{(t-1)(k-1)}S_t(X) \equiv \Omega'(X) \mod X^{n-k}, \tag{4.26}$$

with $\deg \Omega'(X) < n - \tau - 1 - (k-1) = n - k - \tau$.

Let us go back to the full system (4.25). We consider the terms of degree higher than $n - \tau$ of (4.25) and we obtain the following n homogeneous linear equations.

$$\sum_{i=0}^{N_0-1}\Lambda_{0,i}\cdot S_{0,-i+j} + \sum_{t=1}^{\ell}\sum_{i=0}^{N_t-1}\Lambda_{t,i}\cdot S_{t,-t(k-1)-i+j} = 0, \quad \forall j \in [n-\tau-1, 2n-\tau-1) \tag{4.27}$$

Reverting back to the originals univariate polynomials $Q_t(X)$, we obtain the following system:

$$\sum_{i=0}^{N_0-1} Q_{0,i}\cdot S_{0,i+j} + \sum_{t=1}^{\ell}\sum_{i=0}^{N_t-1} Q_{t,i}\cdot S_{t,i+j} = 0, \quad \forall j \in [n). \tag{4.28}$$

With $\mathbf{Q}_t \overset{\text{def}}{=} (Q_{t,0}, Q_{t,1}, \dots, Q_{t,N_t-1})^T$ for $t \in [\ell+1)$ and with $\ell+1$ Hankel matrices:

$$\mathbf{S}^{\langle t\rangle} \overset{\text{def}}{=} \left(S_{i,j}^{\langle t\rangle}\right)_{i\in[n)}^{j\in[N_t)} = \left(S_{t,i+j}\right)_{i\in[n)}^{j\in[N_t)}, \quad \forall t \in [\ell+1),$$

KEs for GRS Decoding

we can write (4.28) in the following matrix form:

$$\left(\mathbf{S}^{\langle 0\rangle}\,\mathbf{S}^{\langle 1\rangle}\,\cdots\,\mathbf{S}^{\langle \ell\rangle}\right)\cdot\begin{pmatrix}\mathbf{Q}_0\\ \mathbf{Q}_1\\ \vdots\\ \mathbf{Q}_\ell\end{pmatrix}=\mathbf{0}. \tag{4.29}$$

In the next subsection, we describe how the FIA can be adapted to a horizontal band of $\ell+1$ Hankel matrices as the homogeneous system of equations (4.29).

4.2.2 Adjustment of the Fundamental Iterative Algorithm

The FIA can directly be applied to the $n \times \sum_{t=0}^{\ell} N_t$ matrix $\left(\mathbf{S}^{\langle 0\rangle}\,\mathbf{S}^{\langle 1\rangle}\,\cdots\,\mathbf{S}^{\langle \ell\rangle}\right)$ of (4.29), but if we want to take advantage of the Hankel structure we have to scan the columns of $\left(\mathbf{S}^{\langle 0\rangle}\,\mathbf{S}^{\langle 1\rangle}\,\cdots\,\mathbf{S}^{\langle \ell\rangle}\right)$ in a manner given by the weighted degree requirement of the interpolation problem. Let k be a positive integer and let $\prec_k^H$ denote the ordering for the pairs $\{(\nu,\mu) \mid \nu \in [\ell+1) \text{ and } \mu \in \mathbb{N}\}$ given by:

$$(\nu,\mu) \prec_k^H (\overline{\nu},\overline{\mu}) \iff \begin{cases} \nu+\mu(k-1) < \overline{\nu}+\overline{\mu}(k-1)\\ \text{or}\\ \nu+\mu(k-1) = \overline{\nu}+\overline{\mu}(k-1) \text{ and } \mu < \overline{\mu}.\end{cases} \tag{4.30}$$

The pair that immediately follows (ν,μ) with respect to the order defined by $\prec_k^H$ is denoted by $\mathrm{succ}(\prec_k^H,(\nu,\mu))$. The $n \times \sum_{t=0}^{\ell} N_t$ syndrome matrix of (4.29) is more explicitly:

$$\begin{aligned}&\left(\mathbf{S}^{\langle 0\rangle}\,\mathbf{S}^{\langle 1\rangle}\,\cdots\,\mathbf{S}^{\langle \ell\rangle}\right)=\\ &\left(\mathbf{S}_0^{\langle 0\rangle,T}\,\mathbf{S}_1^{\langle 0\rangle,T}\cdots\,\mathbf{S}_{N_0-1}^{\langle 0\rangle,T}\,\mathbf{S}_0^{\langle 1\rangle,T}\,\mathbf{S}_1^{\langle 1\rangle,T}\cdots\,\mathbf{S}_{N_1-1}^{\langle 1\rangle,T}\cdots\mathbf{S}_0^{\langle \ell\rangle,T}\,\mathbf{S}_1^{\langle \ell\rangle,T}\cdots\,\mathbf{S}_{N_\ell-1}^{\langle \ell\rangle,T}\right),\end{aligned}$$

where $\mathbf{S}_i^{\langle t\rangle}$ is in $\mathbb{F}_q^n$ for all $t \in [\ell+1)$ and $i \in [N_t)$. The columns $\mathbf{S}_i^{\langle t\rangle,T}$ are reordered according to $\prec_{k-1}^H$. The pair (ν,μ) indexes the μ-th column of ν-th sub-matrix $\mathbf{S}_\mu^{\langle \nu\rangle,T}$. More explicitly, we obtain the following reordered matrix:

$$\mathbf{S}' \stackrel{\text{def}}{=} \left(\mathbf{S}_0^{\langle 0\rangle,T}\,\mathbf{S}_1^{\langle 0\rangle,T}\cdots\,\mathbf{S}_{k-1}^{\langle 0\rangle,T}\,\mathbf{S}_0^{\langle 1\rangle,T}\,\mathbf{S}_k^{\langle 0\rangle,T}\,\mathbf{S}_1^{\langle 1\rangle,T}\,\mathbf{S}_{k+1}^{\langle 0\rangle,T}\cdots\,\mathbf{S}_{N_{\ell-1}-1}^{\langle \ell-1\rangle,T}\,\mathbf{S}_{N_\ell-1}^{\langle \ell\rangle,T}\right). \tag{4.31}$$

The corresponding homogeneous system of equations can now be written in terms of the inner product for bivariate polynomials (see (2.5) for the definition of the inner product).

> **Problem 4.7 (Reformulated Sudan Interpolation Problem)**
> Let the $\ell+1$ syndrome polynomials $S_0(X), S_1(X), \dots, S_\ell(X) \in \mathbb{F}_q[X]$ be given as in (4.20) and (4.21). Let
>
> $$S(X,Y) \stackrel{\text{def}}{=} \sum_{t=0}^{\ell} S_t(X)Y^t$$
>
> be the corresponding bivariate syndrome polynomial in $\mathbb{F}_q[X,Y]$. We search a non-zero bivariate polynomial $T(X,Y) \in \mathbb{F}_q[X,Y]$ such that:
>
> $$\langle\, X^\kappa \cdot T(X,Y), S(X,Y)\,\rangle = 0, \quad \forall \kappa \in [n).$$

Hence, the bivariate polynomial $T(X,Y)$ is a valid interpolation polynomial according to Theorem 3.17 for an interpolation multiplicity $m = 1$. Each polynomial $S_t(X)$, as in (4.15), has degree smaller than $N_t + n - 1$. To index the columns of the rearranged $n \times \sum_{t=0}^{\ell} N_t$ matrix $\mathbf{S}'$, let

$$C_{\nu,\mu} \overset{\text{def}}{=} \left| \Big\{ (t,i) \,|\, (t,i) \prec_{k-1}^{H} (\nu,\mu) \Big\} \right|. \tag{4.32}$$

be the first columns of $\mathbf{S}'$ up to the column of $\mathbf{S}_{\mu}^{\langle \nu \rangle, T}$.

Algorithm 4.1 is the modified FIA for solving Problem 4.7. In contrast to the original Roth–Ruckenstein [A-RR00] adaption we consider all n homogeneous linear equations (instead of τ), because we need to consider also the full system of equations for the Guruswami–Sudan case.

Algorithm 4.1: $T(X,Y) = \texttt{HORIZONTAL-HANKEL}(S(X,Y))$

```
Input: Bivariate polynomial $S(X,Y) = \sum_{t=0}^{\ell} S_t(X)Y^t$, with
       $\deg S_t(X) < N_t + n - 1$
Output: Bivariate polynomial $T(X,Y)$
Data structures:
  Bivariate polynomial $T(X,Y) = \sum_{t=0}^{\ell} T_t(X)Y^t \in \mathbb{F}_q[X,Y]$
  $\ell + 1$ column pointers $(\nu,\mu)$, where $\nu \in [\ell+1)$ and $\mu \in [N_\nu)$
  Row pointer $\kappa \in [n)$, Array $R$ of $\ell + 1$ entries in $[n)$
  Array $D$ of $n$ entries in $\mathbb{F}_q$, Array $A$ of $n$ entries in $\mathbb{F}_q[X,Y]$
  Variable $\Delta \in \mathbb{F}_q$, variable compute $\in$ {true, false}
Initialize:
  for every $i \in [n)$: $D[i] \leftarrow 0$, for every $i \in [\ell+1)$: $R[i] \leftarrow 0$
  $(\nu,\mu) \leftarrow (0,0)$, $\kappa \leftarrow 0$, compute $\leftarrow$ false
while $\kappa < n$ do
    if compute then
        $\Delta \leftarrow \langle X^\kappa \cdot T(X,Y), S(X,Y) \rangle$          // Discrepancy calculation
    else
        if $R[\nu] < 1$ then
            $T(X,Y) \leftarrow Y^\nu \cdot X^\mu$; $\Delta \leftarrow S_{\nu,\mu}$; $\kappa \leftarrow 0$
        else
            $T(X,Y) \leftarrow X \cdot A[R[\nu]](X,Y)$; $\Delta \leftarrow D[R[\nu]]$; $\kappa \leftarrow R[\nu] - 1$
        compute $\leftarrow$ true
    if $\Delta = 0$ or $D[\kappa] \neq 0$ then
        if $\Delta \neq 0$ then
            $T(X,Y) \leftarrow T(X,Y) - \frac{\Delta}{D[\kappa]} \cdot A[\kappa](X,Y)$          // Update
        $\kappa \leftarrow \kappa + 1$
    else                                   // Core discrepancy $\Delta \neq 0$ and $D[\kappa] = 0$
    $A[\kappa](X,Y) \leftarrow T(X,Y)$; $D[\kappa] \leftarrow \Delta$; $R[\nu] \leftarrow \kappa$
    $(\nu,\mu) \leftarrow \mathrm{succ}(\prec_k^H, (\nu,\mu))$
    compute $\leftarrow$ false
```

Let the matrix $\left(\mathbf{S}^{\langle 0 \rangle}\ \mathbf{S}^{\langle 1 \rangle}\ \cdots\ \mathbf{S}^{\langle \ell \rangle}\right)$ be the $n \times \sum_{t=0}^{\ell} N_t$ Sudan syndrome matrix, where each entry $S_{i,j}^{\langle t \rangle}$ equals the coefficient $S_{t,i+j}$ of the polynomial $S_t(X)$. The column pointer is given by (ν,μ) and indexes the column $\mathbf{S}_{\mu}^{\langle\nu\rangle,T}$ in $\prec_{k-1}^{H}$-ordering. This is equivalent to scanning the rearranged matrix $\mathbf{S}'$ as in (4.31) column after column (see Line 16 of Algorithm 4.1). The core discrepancy value for row κ is stored in array D as $D[\kappa]$, and the corresponding intermediate bivariate polynomial is stored in array A as $A[\kappa]$. The discrepancy calculation and the update rule (see (3.30) and (3.31) for the basic FIA) are adapted to the bivariate case (see Line 12 of Algorithm 4.1). For each sub-matrix $\mathbf{S}^{\langle\nu\rangle}$, the previous value of the row pointer κ is stored in an array R as $R[\nu]$. We prove the initialization rule for the FIA solving Problem 4.7 when entering a new column of in the following lemma.

Lemma 4.8 (Initialization Rule)
Assume Algorithm 4.1 examined column $(\nu, \mu-1)$ of the $n \times \sum_{t=0}^{\ell} N_t$ input matrix $\mathbf{S} = \left(\mathbf{S}^{\langle 0\rangle}\ \mathbf{S}^{\langle 1\rangle}\ \cdots\ \mathbf{S}^{\langle \ell\rangle}\right)$, i.e., the $(\mu-1)$-th column of the sub-matrix $\mathbf{S}^{\langle\nu\rangle}$, as defined in (4.29) or equivalently the bivariate polynomial $S(X,Y) = \sum_{t=0}^{\ell} S_t(X)Y^t$. Assume that a core discrepancy was obtained in row κ_ν and the row index was stored in the array $R[\nu]$ (see Line 15). The vanishing linear combination was stored in the array $A[\kappa_\nu](X,Y) \leftarrow T(X,Y)$, i.e.:

$$\left\langle\, X^i A[\kappa_\nu](X,Y), S(X,Y) \,\right\rangle = \sum_{t=0}^{\nu}\sum_{j=0}^{\mu-1} A_{t,j}\cdot S_{t,i+j} = 0, \quad \forall i \in [\kappa_\nu),$$

Let Algorithm 4.1 re-enter the same sub-matrix $\mathbf{S}^{\langle\nu\rangle}$.

Then Algorithm 4.1 can examine column (ν,μ) at row $\kappa_\nu - 1$ with the initial value

$$T(X,Y) \leftarrow X \cdot A[R[\nu]](X,Y)$$

instead of starting in row zero.

Proof In terms of the inner product, we have:

$$\begin{aligned}\left\langle\, X^i T(X,Y), S(X,Y) \,\right\rangle &= \left\langle\, X^{i+1} A[R[\nu]](X,Y), S(X,Y) \,\right\rangle \\ &= \sum_{t=0}^{\nu}\sum_{j=0}^{\mu-1} A_{t,j}\cdot S_{t,i+j+1} \\ &= 0, \qquad \forall i \in [\kappa_\nu - 1).\end{aligned}$$

■

Similar to the FIA for one Hankel matrix, we can start examining column μ of the same sub-matrix $\mathbf{S}^{\langle\nu\rangle}$ in row $\kappa_\nu - 1$.

The following theorem summarizes the properties of Algorithm 4.1.

Theorem 4.9 (Correctness and Complexity of Algorithm 4.1)
Let $\mathbf{S} = \left(\mathbf{S}^{\langle 0\rangle}\ \mathbf{S}^{\langle 1\rangle}\ \cdots\ \mathbf{S}^{\langle \ell\rangle}\right)$ be the $n \times \sum_{t=0}^{\ell} N_t$ matrix as defined in (4.29) and $S(X,Y)$ the associated bivariate syndrome polynomial for the reformulated Sudan interpolation problem. Algorithm 4.1 returns a bivariate polynomial $T(X,Y)$ such that:

$$\langle\, X^\kappa T(X,Y), S(X,Y) \,\rangle = 0, \quad \forall \kappa \in [n).$$

The time complexity of Algorithm 4.1 is $\mathcal{O}(\ell n^2)$ operations in $\mathbb{F}_q$.

Proof The correctness of Algorithm 4.1 follows from the correctness of the basic FIA (see Theorem 3.16) and from the correctness of the initialization rule (see Lemma 4.8), because Algorithm 4.1 deals with the column-permuted version $\mathbf{S}'$ of the original matrix $\mathbf{S} = \left(\mathbf{S}^{\langle 0 \rangle}\, \mathbf{S}^{\langle 1 \rangle}\, \cdots\, \mathbf{S}^{\langle \ell \rangle}\right)$. The proof of the complexity of Algorithm 4.1 is similar to the complexity analysis of the FIA adjusted for a Hankel matrix (see proof of Theorem 3.16). We trace the triple:

$$\left((\nu, \mu), (\kappa_0\ \kappa_1\ \ldots\ \kappa_\ell), \delta\right),$$

where (ν, μ) is the current column pointer of Algorithm 4.1 examining the μ-th column of the ν-th sub-matrix $\mathbf{S}^{\langle \nu \rangle}$. The variable $\boldsymbol{\kappa} = (\kappa_0\ \kappa_1\ \ldots\ \kappa_\ell)$ contains the index of the last row κ_ν reached in the sub-matrices $\mathbf{S}^{\langle \nu \rangle}$. These values are stored in the array R in Algorithm 4.1. The value δ is the number of already encountered core discrepancies of Algorithm 4.1. Assume (ν, μ) is the current column pointer of Algorithm 4.1. The two following events in Algorithm 4.1 similar to Algorithm 3.3 can happen:

1. No core discrepancy: Algorithm 4.1 stays in the same column μ of sub-matrix $\mathbf{S}^{\langle \nu \rangle}$ and the row pointer κ_ν is increased by one. The triple becomes:

$$\left((\nu, \mu), \boldsymbol{\kappa}, \delta\right) \leftarrow \left((\nu, \mu), (\kappa_0, \kappa_1, \ldots, \kappa_\nu + 1, \ldots, \kappa_\ell), \delta\right).$$

2. Core discrepancy: Algorithm 4.1 examines column $(\overline{\nu}, \overline{\mu}) = \operatorname{succ}(\prec_k^H, (\nu, \mu))$ and the triple becomes:

$$\left((\nu, \mu), \boldsymbol{\kappa}, \delta\right) \leftarrow \left((\overline{\nu}, \overline{\mu}), (\kappa_0, \kappa_1, \ldots, \kappa_{\overline{\nu}} - 1, \ldots, \kappa_\ell), \delta + 1\right).$$

From (4.32), we have for $(\overline{\nu}, \overline{\mu}) = \operatorname{succ}(\prec_k^H, (\nu, \mu))$ that

$$C_{\overline{\nu}, \overline{\mu}} = C_{\nu, \mu} + 1.$$

and therefore the sum

$$\mathsf{Iter} \overset{\text{def}}{=} C_{\nu,\mu} + \left(\sum_{t=0}^{\ell} \kappa_t\right) + \delta,$$

increases by one in each iteration of Algorithm 4.1. The last value can be bounded by:

$$\mathsf{Iter} < \mathcal{O}(n) + \mathcal{O}(\ell n) + \mathcal{O}(n) \leq \mathcal{O}(\ell n).$$

Each discrepancy computation costs $\mathcal{O}(n)$ and Algorithm 4.1 does not have to examine more than the first $(n+1)$ columns of the $n \times \sum_{t=0}^{\ell} N_t$ matrix $\left(\mathbf{S}^{\langle 0 \rangle}\, \mathbf{S}^{\langle 1 \rangle}\, \cdots\, \mathbf{S}^{\langle \ell \rangle}\right)$. Thus, the total cost of Algorithm 4.1 is $O(\ell n^2)$. ∎

In the following, we illustrate the values of the row pointer κ of Algorithm 4.1, when applied to a syndrome matrix $\mathbf{S} = \left(\mathbf{S}^{\langle 0 \rangle}\, \mathbf{S}^{\langle 1 \rangle}\, \mathbf{S}^{\langle 2 \rangle}\right)$ that consists of three Hankel matrices.

4.2.3 Example: Sudan Decoding of a Generalized Reed–Solomon Code with Adapted FIA

We consider the decoding of an $[16, 4]_{17}$ GRS code as in Subsection 3.3.3. For an interpolation multiplicity $m = 1$, list size $\ell = 2$, we obtain a decoding radius $\tau = 7 = \lfloor (n-k)/2 \rfloor + 1$. The degrees of the three univariate polynomials $Q_0(X)$, $Q_1(X)$ and $Q_2(X)$ in $\mathbb{F}_{17}$ of the Sudan interpolation polynomial $\sum_{t=0}^{2} Q_t(X) Y^t$ are less than $(N_0, N_1, N_2) = (9, 6, 3)$ and we have more unknowns than interpolation constraints $N_0 + N_1 + N_2 > n$.

KEs for GRS Decoding

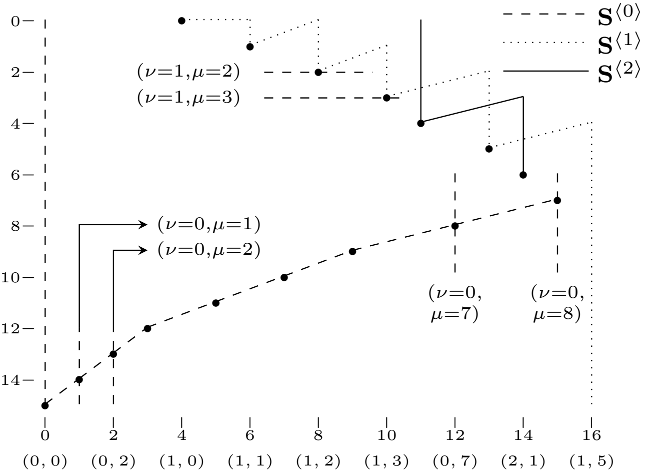


Figure 4.1: Illustration of the row pointers κ_0, κ_1 and κ_2 of Algorithm 4.1 applied to a horizontal band of three Hankel matrices $\mathbf{S}^{\langle 0 \rangle}$, $\mathbf{S}^{\langle 1 \rangle}$ and $\mathbf{S}^{\langle 2 \rangle}$. The columns of the 16×18 matrix $\big(\mathbf{S}^{\langle 0 \rangle}\ \mathbf{S}^{\langle 1 \rangle}\ \mathbf{S}^{\langle 2 \rangle}\big)$ are arranged under $\prec_{4-1}^{H}$-ordering. The three lines trace the row pointers κ_0, κ_1 and κ_2 for each sub-matrix $\mathbf{S}^{\langle 0 \rangle}$, $\mathbf{S}^{\langle 1 \rangle}$ and $\mathbf{S}^{\langle 2 \rangle}$. The second axis of abscissae shows the column pointer (ν, μ) indicating the μ-th column of the sub-matrix $\mathbf{S}^{\langle \nu \rangle}$.

Figure 4.1 illustrates the row pointer of Algorithm 4.1 when the 16×18 syndrome matrix

$$\big(\mathbf{S}^{\langle 0 \rangle}\ \mathbf{S}^{\langle 1 \rangle}\ \mathbf{S}^{\langle 2 \rangle}\big)$$

is examined. The columns of the syndrome matrix are virtually rearranged according to the $\prec_{4-1}^{H}$-ordering and Algorithm 4.1 scans the re-arranged matrix $\mathbf{S}'$ column by column.

The three zig–zag lines in Figure 4.1 trace the value of the row pointer κ_0, κ_1 and κ_2 for the three Hankel sub-matrices $\mathbf{S}^{\langle 0 \rangle}$, $\mathbf{S}^{\langle 1 \rangle}$ and $\mathbf{S}^{\langle 2 \rangle}$. The dots indicate the case, where a core discrepancy occurs. After the 4th column, the columns of the sub-matrices $\mathbf{S}^{\langle 0 \rangle}$ and $\mathbf{S}^{\langle 1 \rangle}$ are interleaved.

After column 10 of the rearranged matrix $\mathbf{S}'$, the columns of $\mathbf{S}^{\langle 0 \rangle}$, $\mathbf{S}^{\langle 1 \rangle}$ and $\mathbf{S}^{\langle 2 \rangle}$ are interleaved. Let us investigate the three marked cases, where a core discrepancy in Algorithm 4.1 occurs.

The first case is from column one to column two of $\mathbf{S}'$, i.e., the column pointer $(0, 1)$ to $(0, 2)$. Entering column $(0, 2)$ allows to set the initial value to 13, because the last core discrepancy occurred in row 14.

The second case is from column eight to ten of $\mathbf{S}'$, i.e., the column pointer $(1, 2)$ to $(1, 3)$. The core discrepancy in column $(1, 2)$ was calculated in row two and we can start examining row one in column $(1, 3)$.

The columns $C_{0,7} = 12$ and $C_{0,8} = 15$ of the re-arranged $\mathbf{S}'$ in Figure 4.1 are the third case. In between column 12 and 15 one column of the sub-matrices $\mathbf{S}^{\langle 1 \rangle}$ and $\mathbf{S}^{\langle 2 \rangle}$ is examined by Algorithm 4.1. In column $(0, 8)$, Algorithm 4.1 starts investigating the row eight, because the core discrepancy in column $(0, 7)$ occurred in row nine.

4.3 Key Equations for the Guruswami–Sudan Principle

4.3.1 Univariate Reformulation

The Guruswami–Sudan interpolation problem for GRS codes, where the multiplicity $m \geq 1$ for the n interpolation points is reformulated. We obtain not one, but a system of m Key Equations. The resulting homogeneous linear system is structured and we show how to adjust the FIA for this case.

We recall that $Q^{[b]}(X,Y)$ denotes the b-th Hasse derivative of a bivariate polynomial $Q(X,Y) \in \mathbb{F}_q[X,Y]$ with respect to the indeterminate Y (see Definition 2.3).

Theorem 4.10 (Univariate Reformulation)
Let $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ be an $[n,k]_q$ GRS code and let the received vector $\mathbf{r} = (r_0\, r_1\, \dots\, r_{n-1}) = \mathbf{c} + \mathbf{e}$, where $\mathbf{c} \in \mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ and $\mathbf{e} \in \mathbb{F}_q^n$, be given. Let $R(X)$ be the Lagrange interpolation polynomial with $\deg R(X) < n$, such that $R(\alpha_i) = r_i/\overline{v}_i$ for all $i \in [n)$ as in (2.3) and let $L(X) = \prod_{i=0}^{n-1}(X - \alpha_i)$ as in (2.1).

An interpolation polynomial $Q(X,Y) \in \mathbb{F}_q[X,Y]$ satisfies Conditions C1 and C2 of Theorem 3.17 for an interpolation multiplicity m, decoding radius τ, and list size ℓ, if and only if there exist polynomials $B_b(X) \in \mathbb{F}_q[X]$, for $b \in [m)$, such that

$$Q^{[b]}(X, R(X)) = B_b(X) \cdot L(X)^{m-b}, \tag{4.33}$$

and $\deg B_b(X) < \ell(n-k) - m\tau + b$.

The following lemma is needed to prove Theorem 4.10.

Lemma 4.11 (Univariate Reformulation of Guruswami–Sudan)
Let $\alpha_i, r_i \in \mathbb{F}_q$ be given, and let $R(X) \in \mathbb{F}_q[X]$ be any polynomial such that $R(\alpha_i) = r_i$. A polynomial $Q(X,Y) \in \mathbb{F}_q[X,Y]$ has multiplicity at least m at (α_i, r_i) if and only if

$$(X - \alpha_i)^{m-b} | Q^{[b]}(X, R(X)), \quad \forall b \in [m).$$

Proof After translation to the origin, we have $(\alpha_i, r_i) = (0,0)$, and therefore $R(0) = 0$, i.e., $X|R(X)$. Let $Q(X,Y) = \sum_b Q_b(X,Y)$, where $Q_b(X,Y)$ is homogeneous of degree b. We first suppose that $Q(X,Y)$ has at least a multiplicity m at $(0,0)$, i.e., $Q_b(X,Y) = 0$, for $b \in [m)$. Hence, we have

$$Q^{[b]}(X, R(X)) = \sum_{i \geq m-b} Q_i^{[b]}(X, R(X)).$$

For $b < m$, the polynomials $Q_i^{[b]}(X,Y)$ have no terms of degree less than $m-b$, and with $X|R(X)$, we have

$$X^{m-b} | Q_i^{[b]}(X, R(X)).$$

It follows, that X^{m-b} divides $Q^{[b]}(X, R(X))$ for all $b \in [m)$ as in Corollary 2.7.

For the converse assume that $X^{m-b} | Q^{[b]}(X, R(X))$, $\forall b \in [m)$. That is, $Q^{[b]}(X, R(X)) = X^{m-b} Z_b(X)$, for some polynomials $Z_0(X), Z_1(X), \dots, Z_{m-1}(X)$. Using Taylor's formula with

KEs for GRS Decoding

the Hasse derivatives, we have:

$$\begin{aligned} Q(X,Y) &= \sum_{b} Q^{[b]}(X,R(X)) \cdot (Y-R(X))^b \\ &= \sum_{b<m} Q^{[b]}(X,R(X)) \cdot (Y-R(X))^b + \sum_{b\geq m} Q^{[b]}(X,R(X)) \cdot (Y-R(X))^b \\ &= \sum_{b<m} X^{m-b} Z_{m-b}(X) \cdot (Y-R(X))^b + \sum_{b\geq m} Q^{[b]}(X,R(X)) \cdot (Y-R(X))^b. \end{aligned}$$

Now, $(Y - R(X))^b$ has only terms of degree at least b, since $X|R(X)$. Thus, we have no terms of degree less than m in $Q(X, Y)$. ■

Proof (of Theorem 4.10) From the previous lemma, we know that $(X-\alpha_i)^{m-b}$ divides $Q^{[b]}(X, R(X))$ for all $b \in [m)$ and $i \in [n)$. Since all polynomials $(X - \alpha_i)$ are distinct, the Chinese Remainder Theorem for univariate polynomials implies that $L(X)^{m-b}|Q^{[b]}(X, R(X))$. The degree condition is:

$$\begin{aligned} \deg B_b(X) &< \deg Q^{[b]}(X,R(X)) - \deg L(X)^{m-b} \\ &= m(n-\tau) + \ell(n-k) - b(n-1) - (m-b)n \\ &= \ell(n-k) - m\tau + b. \end{aligned}$$ ■

We rewrite the m equations of (4.33) more explicitly:

$$\sum_{t=b}^{\ell} \binom{t}{b} Q_t(X) R(X)^{t-b} = B_b(X) \cdot L(X)^{m-b}, \quad \forall b \in [m). \tag{4.34}$$

Recall from (3.38) that

$$N_t \overset{\text{def}}{=} m(n-\tau) - t(k-1), \quad \forall t \in [\ell+1). \tag{4.35}$$

Define the polynomials as:

$$\begin{aligned} \overline{R}(X) &= X^{n-1} \cdot R(X^{-1}), \\ \overline{L}(X) &= X^{n} \cdot L(X^{-1}) = \prod_{i=0}^{n-1} (1-\alpha_i X), \\ \overline{B}_b(X) &= X^{\ell(n-k)-m\tau-b-1} \cdot B_b(X^{-1}), \quad \forall b \in [m), \\ \Lambda_t(X) &= X^{N_t-1} \cdot Q_t(X^{-1}), \quad \forall t \in [\ell+1). \end{aligned}$$

Note that, these polynomials are not necessarily the reciprocal polynomials, because, e.g., for the received polynomial the degree can be smaller than $n-1$.

Reverting both sides of (4.33) and inserting the previously defined reciprocal polynomials leads to:

$$\sum_{t=b}^{\ell} \binom{t}{b} \Lambda_t(X) X^{(\ell-t)(n-k)} \overline{R}(X)^{t-b} = \overline{B}_b(X) \cdot \overline{L}(X)^{m-b}, \quad \forall b \in [m). \tag{4.36}$$

Similar to the Sudan case in (4.16), let:

$$R(X)^{t-b} = U_t^{\langle b \rangle}(X) L(X)^{s-b} + W_t^{\langle b \rangle}(X), \quad \forall b \in [m), t \in [b+1, \ell+1), \tag{4.37}$$

where $W_t^{\langle b \rangle}(X)$ is the remainder of the division of $R(X)^{t-b}$ by $L(X)^{m-b}$ and has degree smaller than $(m-b)n$. Reverting (4.37) leads to:

$$\overline{R}(X)^{t-b} = \overline{U}_t^{\langle b \rangle}(X)\overline{L}(X)^{m-b} + X^{(t-b)(n-1)-(m-b)n+1}\overline{W}_t^{\langle b \rangle}(X), \tag{4.38}$$

where:

$$\overline{U}_t^{\langle b \rangle}(X) = X^{(t-b)(n-1)-(m-b)n}U_t^{\langle b \rangle}(X^{-1}), \quad \forall b \in [m), t \in [b+1, \ell+1), \tag{4.39}$$

$$\overline{W}_t^{\langle b \rangle}(X) = X^{(m-b)n-1}W_t^{\langle b \rangle}(X^{-1}), \quad \forall b \in [m), t \in [b+1, \ell+1). \tag{4.40}$$

Now let the $\sum_{b=0}^{m-1}(\ell+1-b)$ formal power series $S_t^{\langle b \rangle, \infty}(X)$ be defined as:

$$S_t^{\langle b \rangle, \infty}(X) \overset{\text{def}}{=} \frac{\overline{W}_t^{\langle b \rangle}(X)}{\overline{L}(X)^{m-b}}, \quad \forall b \in [1, m), t \in [b+1, \ell+1), \tag{4.41}$$

$$S_b^{\langle b \rangle, \infty}(X) \overset{\text{def}}{=} \frac{X^{(m-b)n-1}}{\overline{L}(X)^{m-b}}, \quad \forall b \in [m). \tag{4.42}$$

Clearly the m power series $S_0^{\langle 0 \rangle, \infty}(X), S_1^{\langle 1 \rangle, \infty}(X), \dots, S_{m-1}^{\langle m-1 \rangle, \infty}(X)$ as in (4.42) do not depend on the received word. Inserting the syndrome polynomials of (4.41) and (4.42) into (4.36) leads to:

$$\begin{aligned}
&\Lambda_b(X)X^{(\ell-b)(n-k)-((m-b)n-1)}S_b^{\langle b \rangle, \infty}(X)\overline{L}(X)^{m-b} \\
&\quad + \sum_{t=b+1}^{\ell} \binom{t}{b} \Lambda_t(X)X^{(\ell-t)(n-k)} \\
&\quad \cdot \left(\overline{U}_t^{\langle b \rangle}(X)\overline{L}(X)^{m-b} + X^{(t-b)(n-1)-(m-b)n+1}S_t^{\langle b \rangle, \infty}(X)\overline{L}(X)^{m-b}\right) \\
&\quad = \overline{B}_b(X) \cdot \overline{L}(X)^{m-b}, \quad \forall b \in [m).
\end{aligned} \tag{4.43}$$

This can be simplified to:

$$\begin{aligned}
&\Lambda_b(X)X^{(\ell-b)(n-k)-((m-b)n-1)}S_b^{\langle b \rangle}(X) \\
&\quad + \sum_{t=b+1}^{\ell} \binom{t}{b} \Lambda_t(X)X^{(\ell-t)(n-k)}X^{(t-b)(n-1)-(m-b)n+1}S_t^{\langle b \rangle}(X) \\
&\qquad \equiv \widetilde{\Omega}_b(X) \mod X^{m(n-\tau)+\ell(n-k)-b(n-1)}, \quad \forall b \in [m),
\end{aligned} \tag{4.44}$$

where

$$\widetilde{\Omega}_b(X) = \overline{B}_b(X) - \sum_{t=b+1}^{\ell} \binom{t}{b} \Lambda_t(X)X^{(\ell-t)(n-k)}\overline{U}_t^{\langle b \rangle}(X),$$

with $\deg \widetilde{\Omega}_b(X) < N_t + (\ell-t)(n-k) + (t-b)(n-1) - (m-b)n = \ell(n-k) - m\tau + b$. The modulo corresponds to the degree of $\deg Q^{[b]}(X, R(X))$, since we limit the degree of the power series of the syndromes and therefore denote them without the infinity sign.

Furthermore with:

$$(\ell-t)(n-k) + (t-b)(n-1) - (m-b)n + 1 = \ell(n-k) - mn + bk + 1(t-b)(k-1),$$

and

$$m(n-\tau)+\ell(n-k)-b(n-1)=\ell(n-k)-mn+bk+1+m(2n-\tau)-bd-1,$$

we can divide both sides of (4.44) by $X^{\ell(n-k)-mn+bk}$ and we obtain:

$$\Lambda_b(X)S_b^{\langle b\rangle}(X)+\sum_{t=b+1}^{\ell}\binom{t}{b}\Lambda_t(X)X^{(t-b)(k-1)}S_t^{\langle b\rangle}(X)$$
$$\equiv \Omega_b(X) \mod X^{m(2n-\tau)-bd-1}, \quad \forall b\in[m), \tag{4.45}$$

where $\deg\Omega_b(X)<\ell(n-k)-m\tau+b-(\ell(n-k)-mn+bk+1)=m(n-\tau)-b(k-1)-1$.

Let us outline a possible reduction of (4.45) similar to the one for the Sudan case from (4.25) to (4.26). We can consider (4.45) modulo $X^{(m-b)n-1}$ and due to (4.42), the term $\Lambda_b(X)S_b^{\langle b\rangle}(X)$ disappears. Furthermore, we can divide (4.45) by X^{k-1} and obtain:

$$\sum_{t=b+1}^{\ell}\binom{t}{b}\Lambda_t(X)X^{(t-b-1)(k-1)}S_t^{\langle b\rangle}(X)\equiv\Omega'_b(X)\mod X^{(m-b)n-k}, \quad \forall b\in[m), \tag{4.46}$$

where $\deg\Omega'_b(X)<m(n-\tau)-b(k-1)-1-(k-1)=m(n-\tau)-k-b(k-1)$.

However, the number of homogeneous linear equations, i.e., the difference between the highest considered term $(m-b)n-k$ and the degree of $\Omega'_b(X)$ is:

$$(m-b)n-k-\big(m(n-\tau)-k-b(k-1)\big)=m\tau-b(n-k+1)=m\tau-bd.$$

The value $m\tau-bd$ can be negative and therefore we do not apply this reduction here (see Subsection 4.3.4 for further details).

Let us go back to the unreduced form as in (4.45). We obtain $(m-b)n$ homogeneous linear equations from (4.45), when considering the coefficients of the terms of degree higher than $m(n-\tau)-b(k-1)$. More explicitly, we have:

$$\sum_{i=0}^{N_b-1}\Lambda_{b,i}\cdot S^{\langle b\rangle}_{b,-i+j}+\sum_{t=b+1}^{\ell}\sum_{i=0}^{N_t-1}\Lambda_{t,i}\cdot S^{\langle t\rangle}_{-(t-b)(k-1)-i+j}$$
$$=0,\quad \forall j\in[m(n-\tau)-b(k-1)-1,m(2n-\tau)-bd-1),\forall b\in[m). \tag{4.47}$$

Reverting back to the original univariate polynomials $Q_0(X)$, $Q_1(X)$, …, $Q_\ell(X)$ we obtain the following system of homogeneous linear equations:

$$\sum_{i=0}^{N_b-1}Q_{b,i}\cdot S^{\langle b\rangle}_{b,i+j}+\sum_{t=b+1}^{\ell}\sum_{i=0}^{N_t-1}Q_{t,i}\cdot S^{\langle b\rangle}_{t,i+j}=0,\quad \forall j\in[(m-b)n),b\in[m). \tag{4.48}$$

With $\mathbf{Q}_t\overset{\text{def}}{=}(Q_{t,0},Q_{t,1},\dots,Q_{t,N_t-1})^T$ for $t\in[\ell+1)$ and with $\sum_{b=0}^{m-1}(\ell+1-b)$ Hankel matrices:

$$\mathbf{S}^{\langle b,t\rangle}\overset{\text{def}}{=}\left(S^{\langle b,t\rangle}_{i,j}\right)_{i\in[(m-b)n)}^{j\in[N_t)}=\left(S^{\langle b\rangle}_{t,i+j}\right)_{i\in[(m-b)n)}^{j\in[N_t)},\quad \forall b\in[m),t\in[b,\ell+1),$$

we can write (4.48) in the following matrix form:

$$\begin{pmatrix} \mathbf{S}^{\langle 0,0\rangle} & \mathbf{S}^{\langle 0,1\rangle} & \dots & \dots & \dots & \mathbf{S}^{\langle 0,\ell\rangle} \\ \mathbf{0} & \mathbf{S}^{\langle 1,1\rangle} & \dots & \dots & \dots & \mathbf{S}^{\langle 1,\ell\rangle} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{S}^{\langle m-1,m-1\rangle} & \dots & \mathbf{S}^{\langle m-1,\ell\rangle} \end{pmatrix} \begin{pmatrix} \mathbf{Q}_0 \\ \mathbf{Q}_1 \\ \vdots \\ \mathbf{Q}_\ell \end{pmatrix} = \mathbf{0}. \tag{4.49}$$

All matrices depend on the received vector $\mathbf{r}$ except the ones on the diagonal: $\mathbf{S}^{\langle b,b\rangle}$, $b \in [m)$ which depend only on the support $\boldsymbol{\alpha}$ of the GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$, the interpolation multiplicity m and the parameter b.

The $\binom{m+1}{2}n \times \sum_{t=0}^{\ell} N_t$ syndrome matrix in (4.49) consists of m bands of b horizontally arranged zero matrices and $\ell+1-b$ Hankel syndrome matrices for $b \in [m)$. This matrix is a called Block-Hankel matrix. The adjustment of the FIA for this Block-Hankel matrix is shown in Subsection 4.3.2.

The explicit expression for the syndromes $S_{t,i}^{\langle b\rangle}$ is difficult to obtain. We compute $S_{t,i}^{\langle b\rangle}$ directly by the power series expansion of

$$S_t^{\langle b\rangle,\infty}(X) = \frac{\overline{W}_t^{\langle b\rangle}(X)}{\overline{L}(X)^{m-b}},$$

as in (4.41) and (4.42). For further discussion on the explicit syndromes see Section 4.4.

4.3.2 The Fundamental Iterative Algorithm for a Block-Hankel Matrix

The extension of the FIA for the case of a Block-Hankel was hinted in Ruckenstein's thesis [O-Ruc01, Section 5.2]. First of all, let us express the m Key Equations of (4.45) in terms of the inner product of bivariate polynomials.

Problem 4.12 (Reformulated Guruswami–Sudan Problem)
Let the m bivariate polynomials $S^{\langle 0\rangle}(X,Y), S^{\langle 1\rangle}(X,Y), \dots, S^{\langle m-1\rangle}(X,Y) \in \mathbb{F}_q[X,Y]$ be defined as:

$$S^{\langle b\rangle}(X,Y) \overset{\text{def}}{=} \sum_{t=b}^{\ell} \sum_{i=0}^{N_t+(m-b)n-1} S_{t,i}^{\langle b\rangle} X^i Y^t, \quad \forall b \in [m), \tag{4.50}$$

where the coefficients $S_{t,i}^{\langle b\rangle}$ are given by the power series of (4.41) and (4.42). We search a non-zero bivariate polynomial $T(X,Y) \in \mathbb{F}_q[X,Y]$ that fulfills:

$$\Big\langle X^\kappa T(X,Y), S^{\langle \vartheta\rangle}(X,Y) \Big\rangle = 0, \quad \forall \vartheta \in [m), \kappa \in [(m-\vartheta)n).$$

We adjust the FIA as an algorithm on a row- and column-interleaved version of the $\binom{m+1}{2}n \times \sum_{t=0}^{\ell} N_t$ Block-Hankel matrix $\mathbf{S}$ of (4.49).

```
Algorithm 4.2: $T(X,Y) = \text{BLOCK-HANKEL}(S^{\langle 0\rangle}(X,Y),\dots,S^{\langle m-1\rangle}(X,Y))$

Input: Bivariate polynomials $S^{\langle b\rangle}(X,Y) = \sum_{t=b}^{\ell} S_t^{\langle b\rangle}(X)Y^t,\ b \in [m)$
Output: Bivariate polynomial $T(X,Y)$

Data structures:
  Bivariate polynomial $T(X,Y) = \sum_{t=0}^{\ell} T_t(X)Y^t$, where $T_t(X) \in \mathbb{F}_q[X]$
  $\ell+1$ column pointers $(0,\mu_0),(1,\mu_1),\dots,(\ell,\mu_\ell)$ where $\mu_t \in [N_t)$
  Row pointer $(\vartheta,\kappa)$, where $\vartheta \in [m)$ and $\kappa \in [(m-\vartheta)n)$
  Array $A[(i,j)]$ of $\binom{m+1}{2}n$ entries in $\mathbb{F}_q[X,Y]$ indexed by the row pointer $(\vartheta,\kappa)$
  Array $D[(i,j)]$ of $\binom{m+1}{2}n$ entries in $\mathbb{F}_q$ indexed by the row pointer $(\vartheta,\kappa)$
  Array $R$ of $\ell+1$ entries to store the row pointer $(\vartheta,\kappa)$
  Variable $\Delta \in \mathbb{F}_q$, variable compute $\in \{\text{true}, \text{false}\}$

Initialize:
  Initialize arrays $A$, $D$ and $C$ to zero
  $(\nu,\mu) \leftarrow (0,0)$ and $(\vartheta,\kappa) \leftarrow (0,0)$
  compute $\leftarrow$ false

while $(\vartheta,\kappa) < (m,0)$ do
    if compute then
        $\Delta \leftarrow \langle X^\kappa \cdot T(X,Y), S^{\langle\vartheta\rangle}(X,Y)\rangle$        // Discrepancy calculation
    else
        if $R[\nu] < 1$ then
            $T(X,Y) \leftarrow Y^\nu \cdot X^\mu$; $\Delta \leftarrow S_{\nu,\mu}^{\langle 0\rangle}$; $(\vartheta,\kappa) \leftarrow (0,0)$
        else
            $T(X,Y) \leftarrow X \cdot A[R[\nu]](X,Y)$; $\Delta \leftarrow D[R[\nu]]$; $(\vartheta,\kappa) \leftarrow R[\nu]$
            if $\kappa = 0$ then
                $(\vartheta,\kappa) \leftarrow (\vartheta-1,n)$
                $\Delta \leftarrow 0$
            $\kappa \leftarrow \kappa - 1$
        compute $\leftarrow$ true
    if $\Delta = 0$ or $D[(\vartheta,\kappa)] \neq 0$ then
        if $\Delta \neq 0$ then
            $T(X,Y) \leftarrow T(X,Y) - \frac{\Delta}{D[(\vartheta,\kappa)]} \cdot A[(\vartheta,\kappa)](X,Y)$        // Update
        $(\vartheta,\kappa) \leftarrow \mathrm{succ}(\prec_n^V, (\vartheta,\kappa))$
    else        // Core discrepancy $\Delta \neq 0$ and $D[\kappa] = 0$
    $A[(\vartheta,\kappa)](X,Y) \leftarrow T(X,Y)$; $D[(\vartheta,\kappa)] \leftarrow \Delta$; $R[\nu] \leftarrow (\vartheta,\kappa)$
    $(\nu,\mu) \leftarrow \mathrm{succ}(\prec_k^H, (\nu,\mu))$
    compute $\leftarrow$ false
```

Let us first define an ordering to describe the vertical rearrangement of the rows of the syndrome matrix $\mathbf{S}$ as in (4.49). Let n be a positive integer and let $\prec_n^V$ denote the ordering indexed by pairs (ϑ,κ), such that:

$$(\vartheta,\kappa) \prec_n^V (\overline{\vartheta},\overline{\kappa}) \iff \begin{cases} \kappa + \vartheta n < \overline{\kappa} + \overline{\vartheta} n, \\ \text{or} \\ \kappa + \vartheta n = \overline{\kappa} + \overline{\vartheta} n \text{ and } \vartheta < \overline{\vartheta}. \end{cases} \tag{4.51}$$

Let $\text{succ}(\prec_n^V, (\vartheta, \kappa))$ denote the pair that immediately follows (ϑ, κ) with respect to order defined by $\prec_n^V$ and let $\text{pred}(\prec_k^V, (\vartheta, \kappa))$ denote the pair that immediately precedes (ϑ, κ) with respect to order defined by $\prec_n^V$. Furthermore, let:

$$R_{\vartheta,\kappa} \overset{\text{def}}{=} \left| \Big\{ (t, i) \mid (t, i) \prec_n^V (\vartheta, \kappa) \Big\} \right|, \tag{4.52}$$

which we use to index the rows of the virtually rearranged matrix (similar to $C_{\nu,\mu}$ as in (4.32) for the horizontal case) and we have

$$R_{\text{pred}(\prec_k^V, (\vartheta, \kappa))} = R_{\vartheta,\kappa} - 1.$$

In the following, $\mathbf{S}'$ denotes the rearranged version of the matrix $\mathbf{S}$ of (4.49), where the columns are ordered under $\prec_{k-1}^H$- and the rows under $\prec_n^V$-ordering. Algorithm 4.2 is the FIA tailored to a Block-Hankel matrix as in (4.49). Similar to the reformulated Sudan interpolation problem, the columns of the Block-Hankel matrix $\mathbf{S}$ are indexed by a pair (ν, μ), where $\nu \in [\ell + 1)$ and $\mu \in [N_\nu)$. Furthermore, the rows are indexed by a couple (ϑ, κ), where $\vartheta \in [m)$ and $\kappa \in [(m - \vartheta) \cdot n)$.

Now, the arrays storing the discrepancies and the intermediate polynomials are still indexed by rows, but the indexes of the rows are two-dimensional, leading to two-dimensional arrays. The two-dimensional array A stores the intermediate bivariate polynomials and the two-dimensional array D, stores the discrepancy values. Both arrays A and D are indexed by the row pointer (ϑ, κ). The discrepancy calculation (see Line 16 of Algorithm 4.2) is adjusted to a Block-Hankel matrix where each sub-horizontal band of Hankel matrices is represented by a bivariate polynomial.

The intermediate bivariate connection polynomial $T^{\langle \vartheta,\kappa \rangle}(X, Y)$ of Algorithm 4.2 examining the κ-th row and the μ-th column of the (ν, ϑ)-th sub-matrix $\mathbf{S}^{\langle \nu,\vartheta \rangle}$, gives us the vanishing linear combination of the sub-matrix consisting of the first $R_{\vartheta,\kappa}$ rows and the first $C_{\nu,\mu}$ columns of the rearranged syndrome matrix $\mathbf{S}'$.

The row pointer of the sub-block $\big(\mathbf{S}^{\langle \nu,0 \rangle}\ \mathbf{S}^{\langle \nu,1 \rangle}\ \cdots\ \mathbf{S}^{\langle \nu,m-1 \rangle}\big)^T$ is stored in the array $R[\nu]$. We need to store $\ell + 1$ row pointers of the form (ϑ, κ).

The adjusted initialization rule of Algorithm 4.2 examining the Block-Hankel syndrome matrix as defined in (4.49) is stated in the following lemma (see Lines 12, 17 and 20 of Algorithm 4.2).

Lemma 4.13 (Initialization Rule)
Assume Algorithm 4.2 examined column $(\nu, \mu - 1)$ of the $\binom{m+1}{2} n \times \sum_{t=0}^{\ell} N_t$ Block-Hankel syndrome matrix $\mathbf{S}$ as defined in (4.49) or equivalently the m bivariate polynomials $S^{\langle 0 \rangle}(X, Y), S^{\langle 1 \rangle}(X, Y), \ldots, S^{\langle m-1 \rangle}(X, Y)$ of Problem 4.12.

Assume that a core discrepancy was obtained in the (ϑ, κ) row of the sub-block $\big(\mathbf{S}^{\langle \nu,0 \rangle}\ \mathbf{S}^{\langle \nu,1 \rangle}\ \cdots\ \mathbf{S}^{\langle \nu,m-1 \rangle}\big)^T$, i.e., the κ-th row of the sub-matrix $\mathbf{S}^{\langle \nu,\vartheta \rangle}$. The row index $(\vartheta, \kappa)_\nu$ was stored in the array $R[\nu] \leftarrow (\vartheta, \kappa)_\nu$. The corresponding vanishing linear combination in form of the bivariate polynomial $T(X, Y)$, was stored in the array $A[(\vartheta, \kappa)_\nu](X, Y) \leftarrow T(X, Y)$ (see Line 19), i.e.:

$$\begin{aligned} \Big\langle\, X^{i_1} A[(\vartheta, \kappa)_\nu](X, Y), S^{\langle i_2 \rangle}(X, Y) \,\Big\rangle &= \sum_{t=0}^{\nu} \sum_{j=1}^{\mu-1} A_{t,j} \cdot S_{t,i_1+j}^{\langle i_2 \rangle} \\ &= 0, \qquad \forall (i_2, i_1) \prec_n^V (\vartheta, \kappa)_\nu. \end{aligned}$$

Let Algorithm 4.2 re-enter the same sub-matrix $\mathbf{S}^{\langle \nu,\vartheta \rangle}$.

We can start examining column (ν, μ) of $\mathbf{S}$ at row $(\vartheta, \kappa - 1)_\nu$ with the initial value $T(X, Y) \leftarrow X \cdot A[(\vartheta, \kappa)_\nu](X, Y)$.

Proof In terms of the inner product, we have:

$$\begin{aligned}\Big\langle\, X^{i_1}T(X,Y), S^{\langle i_2\rangle}(X,Y) \,\Big\rangle &= \Big\langle\, X^{i_1+1}A[R[\nu]](X,Y), S^{\langle i_2\rangle}(X,Y) \,\Big\rangle \\ &= \sum_{t=0}^{\nu}\sum_{j=1}^{\mu-1} A_{t,j}\cdot S^{\langle i_2\rangle}_{t,i_1+j+1} \\ &= 0, \qquad \forall (i_2,i_1) \prec^V_n (\vartheta,\kappa-1).\end{aligned}$$

■

Theorem 4.14 (Correctness and Complexity of Algorithm 4.2)
Let $\mathbf{S}$ be the $\binom{m+1}{2}n \times \sum_{t=0}^{\ell} N_t$ syndrome Block-Hankel matrix of the reformulated Guruswami–Sudan interpolation problem as in (4.49) and let $S^{\langle b\rangle}(X,Y), \forall b \in [m)$ be the corresponding bivariate syndrome polynomials as defined in Problem 4.12. Then Algorithm 4.2 outputs a bivariate polynomial $T(X,Y)$ such that:

$$\Big\langle\, X^{\kappa}T(X,Y), S^{\langle \vartheta\rangle}(X,Y) \,\Big\rangle = 0, \quad \forall \vartheta \in [m), \kappa \in [(m-\vartheta)n).$$

The time complexity of Algorithm 4.2 is $\mathcal{O}(\ell m^4 n^2)$ in $\mathbb{F}_q$.

Proof The correctness follows from the fact that we deal with the row- and column-permuted version $\mathbf{S}'$ of the Block-Hankel matrix $\mathbf{S}$ and that the initialization rule is correct.

In the following, we analyze the complexity of Algorithm 4.2. As in proof of Theorem 4.9, we describe the state of Algorithm 4.2 with the following triple:

$$\big((\nu,\mu), ([\vartheta,\kappa]_0, [\vartheta,\kappa]_1, \dots, [\vartheta,\kappa]_\ell), \delta\big), \tag{4.53}$$

where (ν,μ) is the current column pointer of Algorithm 4.2, when examining the μ-th column of the horizontal band of m vertically arranged Hankel matrices $\big(\mathbf{S}^{\langle \nu,0\rangle}\ \mathbf{S}^{\langle \nu,1\rangle}\ \cdots\ \mathbf{S}^{\langle \nu,m-1\rangle}\big)^T$. The index $[\vartheta,\kappa]_\nu$ is the last considered row in the horizontal band of m sub-matrices $\big(\mathbf{S}^{\langle \nu,0\rangle}\ \mathbf{S}^{\langle \nu,1\rangle}\ \cdots\ \mathbf{S}^{\langle \nu,m-1\rangle}\big)^T$. These values are stored in the array R of Algorithm 4.2. As for Algorithm 4.1, δ denotes the number of already encountered core discrepancies. Assume (ν,μ) is the current column pointer of Algorithm 4.2. The same two cases as for Algorithm 4.1 can happen:

1. No core discrepancy: Algorithm 4.2 remains in the same column (ν,μ) of the sub-matrices $\big(\mathbf{S}^{\langle \nu,0\rangle}\ \mathbf{S}^{\langle \nu,1\rangle}\ \cdots\ \mathbf{S}^{\langle \nu,m-1\rangle}\big)^T$ and the triple becomes:

$$\Big((\nu,\mu), ([\vartheta,\kappa]_0, \dots, [\vartheta,\kappa]_\ell), \delta\Big) \leftarrow \\ \Big((\nu,\mu), ([\vartheta,\kappa]_0, \dots, \mathsf{next}[\prec^V_n, ([\vartheta,\kappa]_\nu)], \dots, [\vartheta,\kappa]_\ell), \delta\Big),$$

2. Core discrepancy: The triple becomes:

$$\Big((\nu,\mu), ([\vartheta,\kappa]_0, \dots, [\vartheta,\kappa]_\ell), \delta\Big) \leftarrow \\ \Big(\mathrm{succ}(\prec^H_k, (\nu,\mu)), ([\vartheta,\kappa]_0, \dots, \mathsf{prev}[\prec^V_n, ([\vartheta,\kappa]_{\overline{\nu}})], \dots, [\vartheta,\kappa]_\ell), \delta+1\Big).$$

In both cases, the sum Iter of the triple is:

$$\mathsf{Iter} = C_{\nu,\mu} + \left(\sum_{t \in [\ell+1)} R_{[\vartheta,\kappa]_t} \right) + \delta,$$

when Algorithm 4.2 examines the (ν, μ)-th column of the Block-Hankel matrix $\mathbf{S}$ of (4.49) and it increases by one in each iteration. The initial value of Iter is zero, and the final value can be bounded by

$$\begin{aligned} \mathsf{Iter} &\leq \binom{m+1}{2} n + \sum_{t=0}^{\ell} \binom{m+1}{2} n + \binom{m+1}{2} n \\ &\leq \mathcal{O}(\ell m^2 n). \end{aligned}$$

The number of iterations of Algorithm 4.2 is bounded by $\mathcal{O}(\ell m^2 n)$.

This gives a total of $\mathcal{O}(\ell m^4 n^2)$, since the discrepancy calculation requires $\mathcal{O}(m^2 n)$. ■

4.3.3 Example: Guruswami–Sudan Decoding of a Generalized Reed–Solomon Code with Adapted FIA

As in Subsection 4.2.3, we consider the decoding of an $[16, 4]_{17}$ GRS code. For interpolation multiplicity $m = 2$, list size $\ell = 4$, the decoding radius is now $\tau = 8$. The degrees of the univariate polynomials $Q_0(X), Q_1(X), \dots, Q_4(X)$ are $(N_0, N_1, N_2, N_3, N_4) = (16, 13, 10, 7, 4)$. The Block-Hankel syndrome matrix:

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}^{\langle 0,0 \rangle} & \mathbf{S}^{\langle 0,1 \rangle} & \mathbf{S}^{\langle 0,2 \rangle} & \mathbf{S}^{\langle 0,3 \rangle} & \mathbf{S}^{\langle 0,4 \rangle} \\ \mathbf{0} & \mathbf{S}^{\langle 1,1 \rangle} & \mathbf{S}^{\langle 1,2 \rangle} & \mathbf{S}^{\langle 1,3 \rangle} & \mathbf{S}^{\langle 1,4 \rangle} \end{pmatrix}$$

is a $(3n = 48) \times (\sum_{t=0}^{4} N_t = 50)$ matrix. It consists of nine non-zero Hankel matrices and one all-zero matrix $\mathbf{S}^{\langle 1,0 \rangle}$ arranged in two horizontal bands of five Hankel matrices.

The values of the row pointer (ϑ, κ) of Algorithm 4.2 for the Block-Hankel matrix are traced in Figure 4.2. The five zig–zag lines in Figure 4.2 trace the row pointer (ϑ, κ), when Algorithm 4.2 examines the five sub-blocks

$$\left(\mathbf{S}^{\langle 0,0 \rangle}\, \mathbf{S}^{\langle 1,0 \rangle}\right)^T, \left(\mathbf{S}^{\langle 1,0 \rangle}\, \mathbf{S}^{\langle 1,1 \rangle}\right)^T, \dots, \left(\mathbf{S}^{\langle 4,0 \rangle}\, \mathbf{S}^{\langle 4,1 \rangle}\right)^T.$$

Additionally to the horizontal ordering $\prec_{k-1}^{H}$ of the columns (as in the Sudan case), now the rows are ordered according to $\prec_{n}^{V}$. Let us consider three cases, where a core discrepancy in Algorithm 4.2 occurred. The first case are the most left two points in Figure 4.2. The value of the column pointer (ν, μ) is $(0, 2)$ and $(0, 3)$. Algorithm 4.2 examines the first band of the two Hankel matrices $\left(\mathbf{S}^{\langle 0,0 \rangle}\, \mathbf{S}^{\langle 1,0 \rangle}\right)^T$. For the first pair no columns were virtually interchanged and the horizontal distance is one.

The second two points with the values of the column pointer $(0, 5)$ and $(0, 6)$ indicate a core discrepancy of Algorithm 4.2, when the second band of the two Hankel matrices $\left(\mathbf{S}^{\langle 0,1 \rangle}\, \mathbf{S}^{\langle 1,1 \rangle}\right)^T$ is examined. The values are traced by the dotted line in Figure 4.2. For the second pair ($(0, 5)$,$(0, 6)$), the columns of the first and second vertical band of Hankel matrices are mixed and therefore the horizontal distance is two. The third considered case, where a core discrepancy occurs, are the most right two points in Figure 4.2 indicated by values $(1, 9)$ and $(1, 10)$ of the row pointer (ϑ, κ). Algorithm 4.2 examined column 42 and a core discrepancy occurred at row $(1, 9)$. In column 43 at row $(1, 10)$ another core discrepancy was examined, and we investigate column 48 (that corresponds to the same sub-matrix as column 43) and run Algorithm 4.2 until the last row.

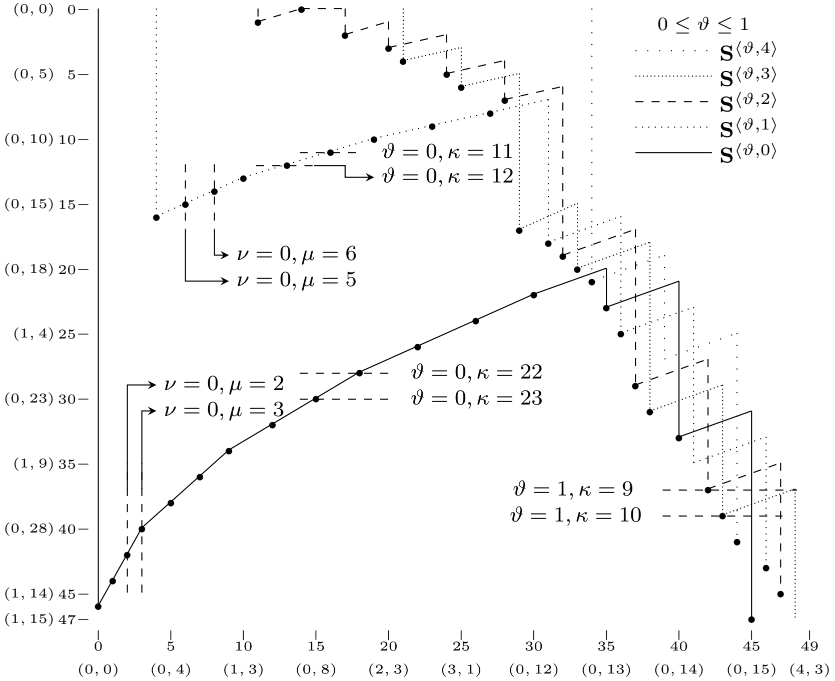

(0, 0) 0
(0, 5) 5
(0, 10) 10
(0, 15) 15
(0, 18) 20
(1, 4) 25
(0, 23) 30
(1, 9) 35
(0, 28) 40
(1, 14) 45
(1, 15) 47
0 5 10 15 20 25 30 35 40 45 49
(0, 0) (0, 4) (1, 3) (0, 8) (2, 3) (3, 1) (0, 12) (0, 13) (0, 14) (0, 15) (4, 3)
$0 \leq \vartheta \leq 1$
$\mathbf{S}^{\langle \vartheta,4 \rangle}$
$\mathbf{S}^{\langle \vartheta,3 \rangle}$
$\mathbf{S}^{\langle \vartheta,2 \rangle}$
$\mathbf{S}^{\langle \vartheta,1 \rangle}$
$\mathbf{S}^{\langle \vartheta,0 \rangle}$
$\vartheta = 0, \kappa = 11$
$\vartheta = 0, \kappa = 12$
$\nu = 0, \mu = 6$
$\nu = 0, \mu = 5$
$\nu = 0, \mu = 2$
$\nu = 0, \mu = 3$
$\vartheta = 0, \kappa = 22$
$\vartheta = 0, \kappa = 23$
$\vartheta = 1, \kappa = 9$
$\vartheta = 1, \kappa = 10$

Figure 4.2: Illustration of the row pointer (ϑ, κ) of Algorithm 4.2 applied to a 48×50 Block-Hankel matrix $\mathbf{S}$. The matrix consists of two vertically arranged bands of five horizontally arranged Hankel matrices. The first band consists of 32 rows and the second one of 16. The plotted matrix $\mathbf{S}'$ consists of the rearranged columns and rows of the matrix $\mathbf{S}$ under $\prec^{H}_{k-1}$- respective $\prec^{V}_{n}$-ordering. The mixture of rows of the two vertical lines starts in line 16 (marked by the dotted horizontal line). The five zig–zag lines trace the row pointer for the five sub-blocks $\big(\mathbf{S}^{\langle 0,0 \rangle} \; \mathbf{S}^{\langle 1,0 \rangle}\big)^T$, $\big(\mathbf{S}^{\langle 1,0 \rangle} \; \mathbf{S}^{\langle 1,1 \rangle}\big)^T, \dots, \big(\mathbf{S}^{\langle 4,0 \rangle} \; \mathbf{S}^{\langle 4,1 \rangle}\big)^T$ of two vertically arranged Hankel matrices. The second axis of abscissae shows the column pointer (ν, μ) indicating the μ-th column of the sub-block $\big(\mathbf{S}^{\langle 0,\nu \rangle} \; \mathbf{S}^{\langle 1,\nu \rangle}\big)^T$.

4.3.4 Reduced Set of Equations

Let us consider the univariate reformulation as in (4.46). The degree of the polynomial $\Omega'_b(X)$ can be greater than $(m-b)n-k$ and it is not clear how to properly truncate this identity, as in [A-RR00; O-Ruc01] for the list decoder with multiplicity $m=1$ or as in the case of the classical Key Equation (see Section 3.2) for BMD decoding.

> **Lemma 4.15 (Reduced Set of Equations for Guruswami–Sudan)**
> Let $d = n-k+1$ be the minimum distance of the considered $[n,k]_q$ GRS code. Let b be such that $m\tau - bd \geq 0$. If $\Lambda_{b+1}(X), \Lambda_{b+2}(X), \dots, \Lambda_{\ell}(X)$ is a solution of (4.46), then there exists $\Lambda_b(X)$ such that $\Lambda_b(X), \Lambda_{b+1}(X), \dots, \Lambda_{\ell}(X)$ is a solution to (4.44).

Proof Let us consider (4.34). We isolate $Q_b(X)$ and get

$$Q_b(X) + \sum_{t=b+1}^{\ell} \binom{t}{b} Q_t(X) \cdot R(X)^{t-b} = B_b(X) \cdot L(X)^{m-b}. \tag{4.54}$$

and thus $Q_b(X)$ is the remainder of the Euclidean division of

$$\sum_{t=b+1}^{\ell} \binom{t}{b} Q_t(X) R(X)^{t-b}$$

by $L(X)^{m-b}$, as long as $\deg Q_b(X) < \deg L(X)^{m-b}$, which gives

$$\begin{aligned} m(n-\tau) - b(k-1) &\leq (m-b)n \\ m\tau - bd &\geq 0. \end{aligned}$$

■

We denote $b_0 = \lfloor (m\tau)/d \rfloor$. Actually, we can consider (4.36) and substitute the $\Lambda_b(X)$, for all $b \in [b_0+1)$, successively. This is possible for the case of the first order system ($m=1$). In the Guruswami–Sudan case ($m>1$), we can obtain a reduced system with $\Lambda_{b_0+1}(X), \Lambda_{b_0+2}(X), \dots, \Lambda_{\ell}(X)$, but it seems that this reduced system lost its Block-Hankel structure.

A future direction is to find a proper reduced polynomial description, i.e., a Key Equation, that leads to a structured set of

$$\sum_{b=0}^{b_0} m\tau - bd + \sum_{b=b_0+1}^{m-1} (m-b)n$$

homogeneous equations and $\sum_{t=b_0+1}^{\ell} N_t$ unknowns.

> **Example 4.16 (Reduced Set of Guruswami–Sudan)**
> Let us calculate the dimension of the reduced set for the previously investigated $[16,4]_{17}$ GRS code with minimum Hamming distance $d=13$. For an interpolation multiplicity $m=2$, list size $\ell=4$ a decoding radius $\tau=8$ is obtained. The full matrix system has $\binom{2+1}{2}16 = 48$ equations and $\sum_{t=0}^{\ell} N_t = 50$ unknowns.
>
> We obtain $b_0 = 1$ and the reduced system consists of $\sum_{b=0}^{1}(s\tau - bd) = 19$ homogeneous equations and with $\sum_{t=2}^{4} N_t = 21$ unknowns. The polynomials $Q_0(X)$ and $Q_1(X)$ of degree smaller than 16 and 13 can be obtained via (4.54).

4.4 Explicit Guruswami–Sudan Syndromes and Hermite Interpolation

4.4.1 Univariate Hermite Interpolation

The basis for the univariate reformulation of the interpolation problem of Guruswami and Sudan considered in the previous sections was the univariate Lagrange interpolation polynomial $R(X)$. The univariate Hermite [A-Her78] interpolation of 1878 generalizes the Lagrange interpolation. An explicit expression of the univariate Hermite interpolation was first given by Spitzbart [A-Spi60]. Let us consider the simplest case, where we take the first derivative into consideration.

Let two vectors $\mathbf{r}, \mathbf{r}_2 \in \mathbb{F}_q^n$ be given. Then, we have a unique polynomial $R(X) \in \mathbb{F}_q[X]$ with deg $R(X)$ smaller than $2n$, such that

$$R(\alpha_j) = r_j \quad \text{and} \quad R^{[1]}(\alpha_j) = r_{2,j}, \quad \forall j \in [n),$$

with:

$$R(X) = \sum_{j=0}^{n-1} r_j A_j(X) + \sum_{j=0}^{n-1} r_{2,j} B_j(X), \tag{4.55}$$

where:

$$\begin{aligned} A_j(X) &= \left(1 - 2(X-\alpha_j)L_j^{[1]}(\alpha_j)\right) \cdot L_j(X)^2, \quad \forall j \in [n), \\ B_j(X) &= (X-\alpha_j)L_j(X)^2, \quad \forall j \in [n), \end{aligned} \tag{4.56}$$

where $L_j(X)$ is as in (2.2).

The polynomials $A_j(X)$ and $B_j(X)$ are such that:

$$\begin{aligned} A_j(\alpha_j) &= 1 \quad \text{and} \quad A_j^{[1]}(\alpha_j) = 0, \\ B_j(\alpha_j) &= 0 \quad \text{and} \quad B_j^{[1]}(\alpha_j) = 1, \quad \forall j \in [n). \end{aligned}$$

The approach can be generalized to an integer $m > 1$ and we explicitly assign values n to each of the first $m-1$ derivatives of the Hermite interpolation polynomial. This polynomial has degree smaller than mn (see [A-Spi60]).

4.4.2 Modified Reformulation for Guruswami–Sudan

Let $R_m(X) \in \mathbb{F}_q[X]$ be the Hermite interpolation polynomial for the n points of degree smaller than or equal to $mn-1$ such that:

$$R_m(\alpha_j) = r_j, \qquad \forall j \in [n) \tag{4.57}$$

$$R_m^{[b]}(\alpha_j) = r_{b,j}, \qquad \forall j \in [n), b \in [1, m). \tag{4.58}$$

In the following we are interested in the degree constraints, if we modify the univariate reformulation of Theorem 4.10 as follows.

Lemma 4.17 (Univariate Reformulation with Hermite)
Let $R_m(X), R_{m-1}(X), \ldots, R_1(X)$ be the m univariate Hermite interpolation polynomials as defined in (4.57) and (4.58). Then a Guruswami–Sudan interpolation polynomial $Q(X, Y)$ (similar to Theorem 4.10) with the parameters n, k, m, ℓ, τ exists if and only if there exist a $B_b(X) \in \mathbb{F}_q[X]$ such that:

$$\deg Q^{[b]}(X, R_{m-b}(X)) = B_b(X) \cdot L(X)^{m-b}, \quad \forall b \in [m), \tag{4.59}$$

where:

$$\begin{aligned}\deg B_b(X) &< \deg Q^{[b]}(X, R_{m-b}(X)) - (m-b)n \\ &< m(n-\tau) + (\ell - b)\big((m-b)n - 1\big) + \ell(1-k) - (m-b)n \\ &< (\ell - b)\big((m-b)n - 1\big) + \ell(1-k) + bn - m\tau.\end{aligned}$$

Proof The proof is analog to the proof of Theorem 4.10 and is independent of the values $R_m^{[b]}(\alpha_i)$ of $R_m(X)$ for all $b \in [1, m)$. The degree conditions follows immediately. ■

Reverting the coefficients of (4.59) is as follows:

$$\begin{aligned}&X^{m(n-\tau)+(\ell-b)\big((m-b)n-1\big)+\ell(1-k)-1} \\ &\cdot \sum_{t=b}^{\ell} \binom{t}{b} Q_t(X^{-1}) R_{m-b}(X^{-1})^{t-b} = \overline{B}_b(X) \cdot \overline{L}(X)^{m-b}.\end{aligned}$$

Substituting the reciprocal polynomials leads to:

$$\begin{aligned}&\Lambda_b(X) X^{(\ell-b)((m-b)n-k)} \\ &\quad + \sum_{t=b+1}^{\ell} \binom{t}{b} \Lambda_t(X) X^{(\ell-t)((m-b)n-k)} \overline{R}_{m-b}(X)^{t-b} = \overline{B}_b(X) \cdot \overline{L}(X)^{m-b}.\end{aligned} \tag{4.60}$$

With:

$$R_{m-b}(X)^{t-b} = U_t^{\langle b \rangle}(X) L(X)^{m-b} + W_t^{\langle b \rangle}(X), \quad \forall b \in [m), t \in [b+1, \ell+1), \tag{4.61}$$

where $W_t^{\langle b \rangle}(X)$ is the remainder of the division of $R_{m-b}(X)^{t-b}$ by $L(X)^{m-b}$ and has degree smaller than $(m-b)n$. Reverting (4.61) leads to:

$$\overline{R}_{m-b}(X)^{t-b} = \overline{U}_t^{\langle b \rangle}(X) \overline{L}(X)^{m-b} + X^{(t-b-1)((m-b)n-1)} \overline{W}_t^{\langle b \rangle}(X), \quad \forall b \in [m),$$

where

$$\begin{aligned}\overline{U}_t^{\langle b \rangle}(X) &= X^{(t-b)((m-b)n-1)-(m-b)n} U_t^{\langle b \rangle}(X^{-1}), \quad \forall b \in [m), t \in [b+1, \ell+1), \\ \overline{W}_t^{\langle b \rangle}(X) &= X^{(m-b)n-1} W_t^{\langle b \rangle}(X^{-1}), \quad \forall b \in [m), t \in [b+1, \ell+1).\end{aligned}$$

Now, we define the syndrome polynomials as in (4.41) and (4.42):

$$S_t^{\langle b \rangle, \infty}(X) \overline{L}(X)^{m-b} = \overline{W}_t^{\langle b \rangle}(X), \qquad \forall b \in [1, m), t \in [b+1, \ell+1), \tag{4.62}$$

$$S_b^{\langle b \rangle, \infty}(X) \overline{L}(X)^{m-b} = X^{(m-b)n-1}, \quad \forall b \in [m). \tag{4.63}$$

Inserting the syndrome definitions of (4.62) and (4.63) into (4.60) and with:

$$\begin{aligned}
&= (\ell - t)\big((m-b)n - k\big) + (t-b-1)\big((m-b)n - 1\big) \\
&\quad - \Big((\ell - b)\big((m-b)n - k\big) - (m-b)n - 1\Big) \\
&= (b-t)\big((m-b)n - k\big) + (t-b)\big((m-b)n - 1\big) \\
&= (t-b)(k-1).
\end{aligned}$$

Dividing the b-th Key Equation by $X^{(\ell-b)\big((m-b)n-k\big)-(m-b)n+1}$ of (4.60) leads to:

$$\begin{aligned}
&\sum_{t=b}^{\ell} \binom{t}{b} \Lambda_t(X) X^{(t-b)(k-1)} S_t^{\langle b \rangle}(X) \\
&\qquad\qquad \equiv \Omega^{(b)}(X) \mod X^{m(2n-\tau)-bd-1}, \quad \forall b \in [m),
\end{aligned} \tag{4.64}$$

where:

$$\begin{aligned}
\deg \Omega_b(X) &< \deg B_b(X) - \Big((\ell - b)\big((m-b)n - k\big) - ((m-b)n - 1\Big) \\
&< m(n-\tau) - b(k-1) - 1.
\end{aligned}$$

4.4.3 Explicit Syndromes

Let us repeat (4.61):

$$R_{m-b}(X)^{t-b} = U_t^{\langle b \rangle}(X) L(X)^{m-b} + W_t^{\langle b \rangle}(X), \quad \forall b \in [m), \tag{4.65}$$

where $\deg W_t^{\langle b \rangle}(X)$ is smaller than $(m-b)n$. The explicit form of $W_t^{\langle b \rangle}(X)$ respectively it reciprocal counterpart are essential for an explicit expression of the syndromes of the modified Guruswami–Sudan reformulation. As for the Sudan case, we can easily get n constraints on $W_t^{\langle b \rangle}(X)$ from (4.65). We have that:

$$R(\alpha_i)^{t-b} = r_i^{t-b} = W_t^{\langle b \rangle}(\alpha_i),$$

and by considering the $b-1$ first Hasse derivatives of (4.65), we obtain the other missing $(m-b-1)n$ constraints to determine $W_t^{\langle b \rangle}(X)$.

It seems to be fruitful to obtain a closed-form expression of explicit syndromes with a reformulation based on polynomials as in (4.57) and (4.58). It is natural to consider a univariate Hermite interpolation polynomial if the multiplicity is greater than one. It is not clear, if a polynomial description of the reduced set as outlined in Subsection 4.4 can be directly obtained with the Hermite-based reformulation. The question remains if the so obtained reduced set is structured. From the complexity point of view it is favorable to adapt an algorithm (as the FIA in Algorithm 4.2) for a larger structured set of equations than to a smaller unstructured one.

We proved the correctness of Algorithm 4.2 and analyzed its complexity based on univariate Key Equations (4.45) for the Guruswami–Sudan interpolation problem for GRS codes.

4.5 Conclusion and Future Work

In Section 4.1 and 4.2, we gave two Key Equations for decoding GRS codes beyond half the minimum distance. The first one is based on the previous work of Schmidt, Sidorenko and Bossert (Section 4.1) and our new contribution is small compared to the original work. We described it for GRS codes and could slightly generalize the bound on the failure probability in Theorem 4.3. We recapitulated the work of Roth and Ruckenstein in Section 4.2 in a slightly different manner to bridge it to the univariate reformulation of the general interpolation problem, where the multiplicity is larger than one (see Lemma 4.5 and Theorem 4.9). In addition, the adaption of the FIA for the obtained Block-Hankel matrix system was described, its correctness proven in Theorem 4.2 and the complexity was analyzed. Furthermore, we showed that a reduction of equations is possible in Lemma 4.15, but it is unclear if this reduced form can be represented in polynomial form, i.e., in terms of a reduced set of Key Equations. In our opinion, the most promising direction is the adaption of the univariate reformulation as outlined in Section 4.4.

Several open research problems exist. A profound comparison between the two approaches of Section 4.1 and Section 4.2 can give new insight and probably some bounds are then transferable. Other algorithmic modifications as e.g., an EEA-like algorithm for the reformulated Sudan and Guruswami–Sudan Key Equation(s), are another future direction. Due to the binomial coefficients, every second Hankel sub-matrix of (4.49) is zero, if we consider GRS codes over the binary extension field. For fields with characteristic two a further reduction of complexity seems possible.

The main challenge is an explicit expression of the Guruswami–Sudan syndromes. This probably allows the formulation of a unique decoding algorithm beyond the radius of the one presented in Section 4.1. The univariate reformulation can also be applied to other related code constructions as folded Reed–Solomon codes [B-Gur07], derivative Reed–Solomon codes [A-GW12] and related code families like Hermitian codes [A-O'S02].

5

"Do not worry about your difficulties in Mathematics. I can assure you mine are still greater."

Albert Einstein (1879–1955) and shown on a sign above the elevator of the Institute of Information Transmission Problems (IITP)

Key Equations for Interpolation-Based Soft-Decision Decoding of Generalized Reed–Solomon Codes

One main challenge for interpolation-based soft-decision decoding of GRS codes à la Kötter–Vardy [O-KV00; A-KV03a], as outlined in Subsection 3.5.3, is the reduction of the complexity of the interpolation step. Kötter *et al.* proposed first in [I-KV03b; I-KMVA03] the re-encoding transformation technique for this purpose. The main idea is to re-encode the k coordinates with highest multiplicity. The obtained codeword is subtracted from the received word. The modified interpolation problem after re-encoding has k zero-coordinates and leads—loosely formulated—to the substitution of the factor n by $n - k$ (which is small for high-rate codes) for the complexity analysis. Kötter *et al.* [I-KV03b; I-KMVA03] obtained a $(1, -1)$-weighted degree rational interpolation problem after the re-encoding transformation. The adaption of the original Kötter algorithm for this rational problem is extensively explained in [A-KMV11] as well as the reduction on the root-finding step.

From a first glance it does not seem very fruitful to reformulate the interpolation-based soft-decision approach of Kötter and Vardy [O-KV00; A-KV03a] in terms of Key Equations as in Section 4.3, but we show that the univariate reformulation after the re-encoding technique of [A-KMV11] leads to a simpler reduced problem than the rational one obtained in [A-KMV11]. Our re-encoding transformation using Key Equations leads to the same reduction of linear equations as in [A-KMV11], but we do not pass to a rational interpolation problem due to the univariate reformulation.

In Section 5.1, we derived the non-re-encoded univariate reformulation of the Kötter–Vardy approach as in Theorem 3.21. In addition, the possible adaption of the FIA is outlined. In Section 5.2, we cover the reformulation of the re-encoded problem in terms of Key Equations. Both approaches of Section 5.1 and 5.2 were not published yet. A short conclusion and future work is given in Section 5.3.

Recall from Theorem 3.21 that for a given $[n, k]_q$ GRS code $\mathcal{GRS}(\boldsymbol{\overline{\upsilon}}, \boldsymbol{\alpha}, k)$ and $q \times n$ multiplicity matrix $\mathbf{m}$, we search a bivariate interpolation polynomial $Q(X, Y) \in \mathbb{F}_q[X, Y]$ with $(1, k-1)$-weighted degree smaller than $\delta + 1$ and Y-degree ℓ, such that:

$$Q^{[a,b]}(\alpha_j, \beta_i/\overline{\upsilon}_j) = 0, \quad \forall a, b \text{ with } a + b < m_{i,j}, i \in [q), j \in [n),$$

where $\beta_0, \beta_1, \dots, \beta_{q-1}$ denotes all distinct elements of $\mathbb{F}_q$.

5.1 Key Equations for the Kötter–Vardy Algorithm

5.1.1 From the Multiplicity Matrix to Univariate Polynomials

Let $\mathbf{c}$ be a codeword of an $[n,k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$ and let, as mentioned in Subsection 3.5.3, the multiplicity matrix $\mathbf{m} = (m_{i,j})_{i \in [q]}^{j \in [n]}$ with nonnegative entries $m_{i,j}$ be given. The support of $\mathcal{GRS}(\overline{\boldsymbol{\upsilon}}, \boldsymbol{\alpha}, k)$ consists of n distinct elements $\boldsymbol{\alpha} = (\alpha_0 \, \alpha_1 \, \dots \, \alpha_{n-1})$ and the q elements of $\mathbb{F}_q$ are denoted by $\beta_0, \beta_1, \dots, \beta_{q-1}$, where $\beta_0 = 0$.

Let the received vector be $\mathbf{r} = \mathbf{c} + \mathbf{e}$, where $\mathbf{e} \in \mathbb{F}_q^n$. The multiplicity $m_{i,j}$ is related to the probability that the j-th symbol r_j is $\beta_i \in \mathbb{F}_q$.

In practical scenarios, the $q \times n$ multiplicity matrix $\mathbf{m}$ is sparse and not all qn points $(\alpha_j, \beta_i) \in \{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\} \times \mathbb{F}_q$ need to be considered for the interpolation. We can describe our univariate reformulation for all qn points and set where appropriate the multiplicity to zero.

Let the q maps be given:

$$\begin{array}{rrcll} p_i : & \{0,1,\dots,n-1\} & \to & \{0,1,\dots,q-1\} & \\ & j & \mapsto & p_i(j), & \forall i \in [q), \end{array}$$

with $p_{i_1}(j) \neq p_{i_2}(j)$ for all $j \in [n)$ and $i_1, i_2 \in [q)$ with $i_1 \neq i_2$. Let the q sets $\mathcal{P}_0, \mathcal{P}_1, \dots, \mathcal{P}_{q-1}$ of n points be defined as follows:

$$\mathcal{P}_i \overset{\text{def}}{=} \Big\{ (\alpha_0, \beta_{p_i(0)}), (\alpha_1, \beta_{p_i(1)}), \dots, (\alpha_{n-1}, \beta_{p_i(n-1)}) \Big\}, \quad \forall i \in [q), \tag{5.1}$$

where $(\alpha_j, \beta_{p_i(j)}) \in \{\alpha_0, \alpha_1, \dots, \alpha_{n-1}\} \times \mathbb{F}_q$ and the set $\mathcal{P}$ of qn points is defined as:

$$\mathcal{P} \overset{\text{def}}{=} \mathcal{P}_0 \cup \mathcal{P}_1 \cup \dots \cup \mathcal{P}_{q-1}. \tag{5.2}$$

Due to the definition the maps $p_0, p_1, \dots, p_{q-1}$, the sets $\mathcal{P}_0, \mathcal{P}_1, \dots, \mathcal{P}_{q-1}$ are disjoint. Now, we reformulate the interpolation problem of Theorem 3.21 in a univariate way, as we did it for the classical Guruswami–Sudan problem in Section 4.3.

Recall that $[a]^+$ denotes $\max\{a, 0\}$. Let the maximal multiplicity out of the n points in $\mathcal{P}_i$ of a given multiplicity matrix $\mathbf{m}$ be:

$$m_i \overset{\text{def}}{=} \max_{j \in [n)} m_{p_i(j),j}, \quad \forall i \in [q). \tag{5.3}$$

Define univariate polynomials in $\mathbb{F}_q[X]$ as:

$$L^{\langle i,b \rangle}(X) \overset{\text{def}}{=} \prod_{j=0}^{n-1} (X - \alpha_j)^{[m_{p_i(j),j} - b]^+}, \quad \forall b \in [m_i), i \in [q), \tag{5.4}$$

with degree:

$$d_{i,b} \overset{\text{def}}{=} \deg L^{\langle i,b \rangle}(X) = \sum_{j=0}^{n-1} [m_{p_i(j),j} - b]^+, \quad \forall b \in [m_i), i \in [q). \tag{5.5}$$

The q unique polynomials $R_0(X), R_1(X), \dots, R_{q-1}(X) \in \mathbb{F}_q[X]$ denote the Lagrange interpolation polynomials of the n points in $\mathcal{P}_0, \mathcal{P}_1, \dots, \mathcal{P}_{q-1}$ such that:

$$R_i(\alpha_j) = \frac{\beta_{p_i(j)}}{\overline{\upsilon}_j}, \quad \forall j \in [n), i \in [q), \tag{5.6}$$

as given in (2.3).

We illustrate the univariate interpolation polynomials of (5.6) and the multiplicity matrix $\mathbf{m}$ in Figure 5.1.

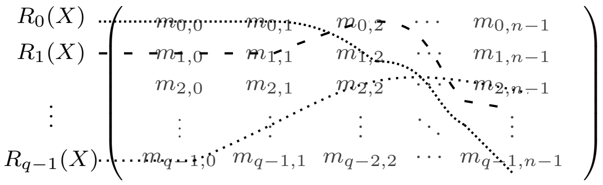

$R_0(X)$
$R_1(X)$
$\vdots$
$R_{q-1}(X)$
$m_{0,0}$ $m_{0,1}$ $m_{0,2}$ $\cdots$ $m_{0,n-1}$
$m_{1,0}$ $m_{1,1}$ $m_{1,2}$ $\cdots$ $m_{1,n-1}$
$m_{2,0}$ $m_{2,1}$ $m_{2,2}$ $\cdots$ $m_{2,n-1}$
$\vdots$ $\vdots$ $\vdots$ $\ddots$ $\vdots$
$m_{q-1,0}$ $m_{q-1,1}$ $m_{q-2,2}$ $\cdots$ $m_{q-1,n-1}$

Figure 5.1: The $q \times n$ multiplicity matrix $\mathbf{m} = (m_{i,j})_{i\in[q)}^{j\in[n)}$ of the Kötter–Vardy [A-KV03a] algorithm and the q univariate polynomials $R_0(X), R_1(X), \dots, R_{q-1}(X) \in \mathbb{F}_q[X]$.

Lemma 5.1 (Univariate Reformulation of Kötter–Vardy)
Let the parameters of an $[n,k]_q$ GRS code, the multiplicity matrix $\mathbf{m}$, the point sets $\mathcal{P}_0, \mathcal{P}_1, \dots, \mathcal{P}_{q-1}$ as in (5.1) and the maximal multiplicities $m_0, m_1, \dots, m_{q-1}$ as in (5.3) be given. Let the q Lagrange interpolation polynomials $R_0(X), R_1(X), \dots, R_{q-1}(X) \in \mathbb{F}_q[X]$ as in (5.6) be given.

The bivariate polynomial $Q(X,Y)$ is a solution to Theorem 3.21 for given parameters n, k, list size ℓ, multiplicity matrix $\mathbf{m}$ and of $\mathrm{wdeg}_{1,\text{k-1}}\, Q(X,Y) < \delta + 1$ if and only if there exist $\sum_{i=0}^{q-1} m_i$ polynomials $B_b^{\langle i \rangle}(X) \in \mathbb{F}_q[X]$ such that:

$$Q^{[b]}(X, R_i(X)) = B_b^{\langle i \rangle}(X) \cdot L^{\langle i,b \rangle}(X), \quad \forall b \in [m_i), i \in [q), \tag{5.7}$$

where $L^{\langle i,b \rangle}(X)$ with $d_{i,b} = \deg L^{\langle i,b \rangle}(X)$ is defined as in (5.4) and with

$$\deg B_b^{\langle i \rangle}(X) < \delta + \ell(n-k) - b(n-1) - d_{i,b}, \quad \forall b \in [m_i), i \in [q). \tag{5.8}$$

Proof From Lemma 4.11, we know that both directions hold for one point (α_i, r_i) with multiplicity m.

We conclude that

$$(X - \alpha_j)^{[m_{i,j}-b]^+} \mid Q^{[b]}(X, R_i(X)), \quad \forall j \in [n), b \in [m_i), i \in [q).$$

Then, the Chinese Remainder Theorem implies that

$$L^{\langle i,b \rangle}(X) \mid Q^{[b]}(X, R_i(X)), \quad \forall b \in [m_i), i \in [q).$$

The degree condition of (5.8) follows directly. ■

From Lemma 5.1, we obtain q Guruswami–Sudan-like Key Equations as in Section 4.3, which provide:

$$\sum_{i=0}^{q-1} \sum_{b=0}^{m_i-1} d_{i,b} = \frac{1}{2} \sum_{i=0}^{q-1} \sum_{j=0}^{n-1} m_{i,j}(m_{i,j}+1)$$

homogeneous linear equations.

5.1.2 Block–Hankel Structure and Fundamental Iterative Algorithm

With Lemma 5.1 we obtain $1/2 \sum_{i=0}^{q-1} \sum_{j=0}^{n-1} m_{i,j}(m_{i,j}+1)$ homogeneous linear equations on the interpolation polynomial $Q(X,Y)$ and we can use a generalization of the FIA to solve it efficiently as in Section 4.3. Let $Q(X,Y) = \sum_{t=0}^{\ell} \sum_{i=0}^{N_t-1} Q_{t,i} X^i Y^t$ be a solution as in Theorem 3.21. Then, we obtain q Guruswami–Sudan-like Key Equations and consider only the terms of highest degree, i.e.:

$$\sum_{t=b}^{\ell} \sum_{j=0}^{N_t-1} \binom{t}{b} Q_{t,j} \cdot S_{t,j+u}^{\langle i,b \rangle} = 0, \quad \forall b \in [m_i), u \in [d_{i,b}), i \in [q), \tag{5.9}$$

where $S_t^{\langle i,b \rangle}(X) \in \mathbb{F}_q[X]$ are the syndrome polynomials as in (4.41) and (4.42), which depend on the polynomials

$$\begin{aligned} R_0(X), R_1(X), \dots, R_{q-1}(X) \quad & \text{as in (3.2) and} \\ L^{\langle 0,0 \rangle}(X), L^{\langle 1,0 \rangle}(X), \dots, L^{\langle q-1, m_{q-1} \rangle}(X) \quad & \text{as in (5.4).} \end{aligned}$$

Let $\mathbf{Q}_t = (Q_{t,0}, Q_{t,1}, \dots, Q_{t,N_t-1})^T$ be the vector that contains the coefficients of the polynomial $Q_t(X) \in \mathbb{F}_q[X]$ for all $t \in [\ell+1)$. The homogeneous linear equations of (5.9) can be written in matrix form as follows:

$$\begin{pmatrix} \mathbf{S}^{\langle 0 \rangle} \\ \mathbf{S}^{\langle 1 \rangle} \\ \vdots \\ \mathbf{S}^{\langle q-1 \rangle} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{Q}_0 \\ \mathbf{Q}_1 \\ \vdots \\ \mathbf{Q}_\ell \end{pmatrix} = \mathbf{0}, \tag{5.10}$$

where each sub-matrix

$$\mathbf{S}^{\langle i \rangle} = \begin{pmatrix} \mathbf{S}^{\langle i,0,0 \rangle} & \mathbf{S}^{\langle i,0,1 \rangle} & \dots & \dots & \dots & \mathbf{S}^{\langle i,0,\ell \rangle} \\ \mathbf{0} & \mathbf{S}^{\langle i,1,1 \rangle} & \dots & \dots & \dots & \mathbf{S}^{\langle i,1,\ell \rangle} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{S}^{\langle i,m_i-1,m_i-1 \rangle} & \dots & \mathbf{S}^{\langle i,m_i-1,\ell \rangle} \end{pmatrix}$$

is a $1/2 \sum_{j=0}^{n-1} m_{i,j}(m_{i,j}+1) \times \sum_{t=0}^{\ell} N_t$ Block-Hankel matrix for all $i \in [q)$. The coefficients of the Hankel sub-matrix $\mathbf{S}^{\langle i,b,t \rangle}$ are given by:

$$S_{j,u}^{\langle i,b,t \rangle} = S_{t,j+u}^{\langle i,b \rangle}, \quad \forall b \in [m_i), t \in [\ell), i \in [q), j \in [d_{i,b}), u \in [N_t).$$

Let us shortly describe how the FIA, as explained in Subsection 4.3.2 for one Block-Hankel matrix, can be adapted to a matrix, which consists of q vertically arranged Block-Hankel matrices. We need q column and row pointers as in Algorithm 4.2 to index the columns and rows of the different q sub-Block-Hankel matrices. Equivalently the arrays that store the discrepancies and the intermediate polynomials need to be suited for q Block-Hankel matrices. Probably, if the rows of the matrix

$$\left(\mathbf{S}^{\langle 0 \rangle} \mathbf{S}^{\langle 1 \rangle} \cdots \mathbf{S}^{\langle q-1 \rangle} \right)^T$$

are interleaved in a similar manner as the rows of each sub-matrix $\mathbf{S}^{\langle i \rangle}$, a complexity-reducing initialization of the FIA is possible. The time complexity is probably increased by a factor of q compared to a single Block-Hankel matrix. If the FIA is not adapted than the increase of the time complexity is by factor q^2 instead of q. The space complexity is clearly increased by factor q. The main advantage of the FIA is its applicability for the reduced set of homogeneous equations after the re-encoding transformation as explained in next section.

5.2 Re-Encoding Transformation with Key Equations

5.2.1 Ordering of Multiplicities

We show that the univariate reformulation after the re-encoding transformation leads to a reduced system of homogeneous linear equations. The matrix consists of q vertically aligned Block-Hankel matrices as in (5.10), where the reduction through re-encoding concerns only one out of the q Block-Hankel matrices. Algorithm 4.2 can be applied and does not need to be changed fundamentally as, e.g., the Kötter interpolation algorithm [O-Kö96b; A-Kö96a] has to be adapted for the $(1, -1)$-weighted degree interpolation problem, which was obtained through the re-encoding transformation as shown in [A-KMV11, Section IV.C].

Let $\boldsymbol{\alpha} = (\alpha_0\, \alpha_1\, \ldots\, \alpha_{n-1})$ be the support of a given $[n, k]_q$ GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$ and let $\mathbb{F}_q = \{\beta_0, \beta_1, \ldots, \beta_{q-1}\}$, where $\beta_0 = 0$.

> **Definition 5.2 (Column Leader)**
> Let $\mathbf{m}_0, \mathbf{m}_1, \ldots, \mathbf{m}_{n-1} \in \mathbb{F}_q^q$ denote the n columns of a given $q \times n$ multiplicity matrix
>
> $$\mathbf{m} = \left(\mathbf{m}_0^T\, \mathbf{m}_1^T\, \ldots\, \mathbf{m}_{n-1}^T\right).$$
>
> Let the point (α_j, β_i) with maximal multiplicity $\max_{i \in [q)} m_{i,j}$ in one column $\mathbf{m}_j^T$ be the column leader.

We assume that the first k columns $\mathbf{m}_0^T, \mathbf{m}_1^T, \ldots, \mathbf{m}_{k-1}^T$ are the columns that contain the k column leaders with the greatest multiplicity among the n column leaders of $\mathbf{m}$.

Furthermore, let the map p_0 be such that the first k points

$$(\alpha_0, \beta_{p_0(0)}), (\alpha_1, \beta_{p_0(1)}), \ldots, (\alpha_{k-1}, \beta_{p_0(k-1)})$$

are the first k column leaders, i.e.:

$$m_{p_0(j),j} = \max_{i \in [q)} m_{i,j}, \quad \forall j \in [k).$$

Let, as in Section 5.1, the q Lagrange interpolation polynomials $R_0(X)$, $R_1(X)$, $\ldots$, $R_{q-1}(X)$ be such that $R_i(\alpha_j) = \beta_{p_i(j)}/\overline{v}_j$, for all $j \in [n)$, $i \in [q)$.

The first k column leaders as in Definition 5.2 are the basis for the re-encoding transformation.

Let $h(X) \in \mathbb{F}_q[X]$ be the unique Lagrange polynomial of degree smaller than k such that the first k column leaders are interpolated, i.e.:

$$h(\alpha_i) = \frac{\beta_{p_0(i)}}{\overline{v}_i}, \quad \forall i \in [k). \tag{5.11}$$

Clearly, $\mathrm{eval}(h(X), \overline{\boldsymbol{v}}, \boldsymbol{\alpha})$ is a codeword of the given GRS code $\mathcal{GRS}(\overline{\boldsymbol{v}}, \boldsymbol{\alpha}, k)$.

Then, the q sets $\widehat{\mathcal{P}}_0, \widehat{\mathcal{P}}_1, \ldots, \widehat{\mathcal{P}}_{q-1}$ after the re-encoding transformation, where each set contains n distinct interpolation points, are:

$$\widehat{\mathcal{P}}_0 \overset{\text{def}}{=} \Big\{\big(\alpha_0, 0\big), \ldots, \big(\alpha_{k-1}, 0\big), \big(\alpha_k, \beta_{p_0(k)} - h(\alpha_k)\big), \ldots, \big(\alpha_{n-1}, \beta_{p_0(n-1)} - h(\alpha_{n-1})\big)\Big\},$$

$$\widehat{\mathcal{P}}_j \overset{\text{def}}{=} \Big\{\big(\alpha_0, \beta_{p_j(0)} - h(\alpha_0)\big), \ldots, \big(\alpha_{n-1}, \beta_{p_j(n-1)} - h(\alpha_{n-1})\big)\Big\}, \ \forall j \in [1, q).$$

The q Lagrange interpolation polynomials are now $\widehat{R}_0(X), \widehat{R}_1(X), \dots, \widehat{R}_{q-1}(X) \in \mathbb{F}_q[X]$ for the n points in $\widehat{\mathcal{P}}_0, \widehat{\mathcal{P}}_1, \dots, \widehat{\mathcal{P}}_{q-1}$ are:

$$\widehat{R}_j(X) = R_j(X) - h(X). \tag{5.12}$$

Clearly, there exists a polynomial $K_0(X) \in \mathbb{F}_q[X]$ with degree smaller than $n-k$, such that:

$$\widehat{R}_0(X) = \prod_{i=0}^{k-1} (X - \alpha_i) \cdot K_0(X). \tag{5.13}$$

Let the q maps

$$\begin{array}{rccll} \widehat{p}_i : & \{0,1,\dots,n-1\} & \to & \{0,1,\dots,q-1\} & \\ & j & \mapsto & \widehat{p}_i(j), & \forall i \in [q), \end{array}$$

be defined such that:

$$\beta_{\widehat{p}_i(j)} = \beta_{p_i(j)} - h(\alpha_j), \quad \forall j \in [n), i \in [q).$$

Then, the re-encoded multiplicity matrix is:

$$\widehat{\mathbf{m}} = \left(\widehat{m}_{i,j}\right)_{i\in[q)}^{j\in[n)} = \left(m_{\widehat{p}_i(j),j}\right)_{i\in[q)}^{j\in[n)}. \tag{5.14}$$

Equivalently to (5.3), define

$$\widehat{m}_i \overset{\text{def}}{=} \max_{j\in[n)} m_{\widehat{p}_i(j),j}, \quad \forall i \in [q). \tag{5.15}$$

The first k column leaders have multiplicity $m_{p_0(0),0}, m_{p_0(1),1}, \dots, m_{p_0(k-1),k-1}$ or equivalently after the re-encoding transformation $\widehat{m}_{0,0}, \widehat{m}_{0,1}, \dots, \widehat{m}_{0,k-1}$. For ease of notation let as [A-KMV11]:

$$\nu_i \overset{\text{def}}{=} \widehat{m}_{0,i}, \quad \forall i \in [k).$$

Let us recall [A-KMV11, Theorem 3]. For the proof see [A-KMV11, Theorem 3].

> **Theorem 5.3 (Re-Encoding Transformation [A-KMV11, Theorem 3])**
> A polynomial $Q(X,Y)$ is a solution to Theorem 3.21 for a given $q \times n$ multiplicity matrix $\mathbf{m}$ if and only if the polynomial
> $$P(X,Y) = Q(X, Y + h(X))$$
> is a solution according to Theorem 3.21 with multiplicity matrix $\widehat{\mathbf{m}}$ as in (5.14).

Clearly, the Y-degree ℓ and the weighted degree are identical for both polynomials $Q(X,Y)$ and $P(X,Y)$.

From Corollary 2.9, we know that $P(X,Y) = \sum_{t=0}^{\ell} P_t(X)Y^t$ that interpolates the k re-encoded points of $\widehat{\mathcal{P}}_0$ with multiplicity $\nu_0, \nu_1, \dots, \nu_{k-1}$ holds if and only if $P_t(X)$ are divisible by $\prod_{i=0}^{k-1}(X - \alpha_i)^{[\nu_i - t]^+}$. Let the $\ell+1$ polynomials $\overline{P}_0(X), \overline{P}_1(X), \dots, \overline{P}_\ell(X)$ such that

$$P_t(X) = \prod_{i=0}^{k-1}(X-\alpha_i)^{[\nu_i - t]^+}\,\overline{P}_t(X).$$

We have:

$$\begin{aligned}\deg \overline{P}_t(X) &< \deg Q_t(X) - \sum_{i=0}^{k-1} [\nu_i - t]^+ \\ &< \delta + 1 - t(k-1) - \sum_{i=0}^{k-1} [\nu_i - t]^+, \quad t \in [\ell+1),\end{aligned}$$

and $Q(X,Y) = \sum_{t=0}^{\ell} Q_t(X)Y^t$ is a solution of the original interpolation problem with multiplicity matrix $\mathbf{m}$.

5.2.2 Reduced Set of Univariate Equations

Let

$$\widehat{L}^{\langle 0,b\rangle}(X) \overset{\text{def}}{=} \prod_{j=k}^{n-1} (X - \alpha_j)^{[\widehat{m}_{\widehat{p}_0(j),j} - b]^+}, \quad \forall b \in [\widehat{m}_0), \tag{5.16}$$

$$\widehat{L}^{\langle i,b\rangle}(X) \overset{\text{def}}{=} \prod_{j=0}^{n-1} (X - \alpha_j)^{[\widehat{m}_{\widehat{p}_i(j),j} - b]^+}, \quad \forall b \in [\widehat{m}_i), i \in [1,q), \tag{5.17}$$

with degree:

$$\widehat{d}_{0,b} \overset{\text{def}}{=} \deg \widehat{L}^{\langle 0,b\rangle}(X) = \sum_{j=k}^{n-1} [\widehat{m}_{\widehat{p}_0(j),j} - b]^+, \quad \forall b \in [\widehat{m}_0), \tag{5.18}$$

$$\widehat{d}_{i,b} \overset{\text{def}}{=} \deg \widehat{L}^{\langle i,b\rangle}(X) = \sum_{j=0}^{n-1} [\widehat{m}_{\widehat{p}_i(j),j} - b]^+, \quad \forall b \in [\widehat{m}_i), i \in [1,q). \tag{5.19}$$

Lemma 5.4 (Univariate Reformulation after Re-Encoding)
Let the $q \times n$ multiplicity matrix $\widehat{\mathbf{m}}$, the multiplicities $\widehat{m}_0, \widehat{m}_1, \dots, \widehat{m}_{q-1}$ as in (5.15), the q Lagrange interpolation polynomials

$$\widehat{R}_0(X) = \prod_{i=0}^{k-1} (X - \alpha_i) K_0(X),$$

$\widehat{R}_1(X), \dots, \widehat{R}_{q-1}(X)$ as in (5.12) and the polynomials

$$\widehat{L}^{\langle 0,0\rangle}(X), \widehat{L}^{\langle 0,1\rangle}(X), \dots, \widehat{L}^{\langle q-1,\widehat{m}_{q-1}\rangle}(X)$$

as in (5.16) and (5.17) be given after the re-encoding transformation. Then, the interpolation polynomial in $\mathbb{F}_q[X,Y]$

$$P(X,Y) = \sum_{t=0}^{\ell} \left(\overline{P}_t(X) \prod_{i=0}^{k-1} (X - \alpha_i)^{[\nu_i - t]^+} \right) Y^t \tag{5.20}$$

Soft-Decision

is a solution to Theorem 3.21 for the multiplicity matrix $\widehat{\mathbf{m}}$ and parameters ℓ and δ if and only if there exist $\sum_{i=0}^{q-1} \widehat{m}_i$ polynomials $\widehat{B}_b^{\langle i\rangle}(X) \in \mathbb{F}_q[X]$ such that:

$$\sum_{t=b}^{\ell}\left(\binom{t}{b}\overline{P}_t(X)\prod_{i=0}^{k-1}(X-\alpha_i)^{[t-\nu_i]^+}\right)K_0(X)^{t-b} = \widehat{B}_b^{\langle 0\rangle}(X)\widehat{L}^{\langle 0,b\rangle}(X), \forall b \in [\widehat{m}_0) \tag{5.21}$$

$$\sum_{t=b}^{\ell}\left(\binom{t}{b}\overline{P}_t(X)\prod_{i=0}^{k-1}(X-\alpha_i)^{[\nu_i-t]^+}\right)\widehat{R}_j(X)^{t-b} = \widehat{B}_b^{\langle j\rangle}(X)\widehat{L}^{\langle j,b\rangle}(X),\ \forall b \in [\widehat{m}_j), j \in [1,q), \tag{5.22}$$

with

$$\deg \widehat{B}_b^{\langle j\rangle}(X) < \delta + \ell(n-k) - b(n-1) - \widehat{d}_{i,b}, \quad \forall j \in [q), b \in [\widehat{m}_j).$$

Proof Through the re-encoding transformation, we have k points with multiplicities $\nu_0, \nu_1, \dots, \nu_{k-1}$ and with a zero Y-coordinate. From Corollary 2.9, we know that this holds if and only if the univariate polynomials $P_t(X)$ are divisible by $\prod_{i=0}^{k-1}(X-\alpha_i)^{[\nu_i-t]^+}$. Therefore, the statement for the $q-1$ Guruswami–Sudan-like reformulations (5.22) directly follows from Corollary 2.9 and Lemma 5.1.

Let us investigated the reduced reformulation (5.21). From Lemma 5.1, we know that $P(X,Y)$ is a solution if and only if there exist $\widehat{m}_0$ polynomials $\widehat{B}_0^{\langle 0\rangle}(X)$, $\widehat{B}_1^{\langle 0\rangle}(X), \dots, \widehat{B}_{\widehat{m}_0-1}^{\langle 0\rangle}(X) \in \mathbb{F}_q[X]$ such that:

$$P^{[b]}(X, \widehat{R}_0(X)) = \widehat{B}_b^{\langle 0\rangle}(X)\cdot \widehat{L}^{\langle 0,b\rangle}(X), \quad \forall b \in [\widehat{m}_0).$$

We substitute (5.13) into (5.20) and obtain:

$$\begin{aligned} P^{[b]}(X,\widehat{R}_0(X)) &= \sum_{t=b}^{\ell}\binom{t}{b}\left(\overline{P}_t(X)\prod_{i=0}^{k-1}(X-\alpha_i)^{[\nu_i-t]^+}\right)\widehat{R}_0(X)^{t-b} \\ &= \sum_{t=b}^{\ell}\binom{t}{b}\left(\overline{P}_t(X)\prod_{i=0}^{k-1}(X-\alpha_i)^{[\nu_i-t]^+}\prod_{i=0}^{k-1}(X-\alpha_i)^{t-b}\right)K_0(X)^{t-b}. \end{aligned}$$

With $[\nu_i - t]^+ + t - b = \nu_i - b + [t-\nu_i]^+$ (note that $t \geq b$), we obtain:

$$\begin{aligned} P^{[b]}(X,\overline{R}(X)) = &\left(\prod_{i=0}^{k-1}(X-\alpha_i)^{[\nu_i-b]^+}\right) \\ &\cdot\left(\sum_{t=b}^{\ell}\binom{t}{b}\left(\overline{P}_t(X)\prod_{i=0}^{k-1}(X-\alpha_i)^{[t-\nu_i]^+}\right)K_0(X)^{t-b}\right), \end{aligned} \tag{5.23}$$

for all $b \in [\widehat{m}_0)$. ■

We find the univariate polynomials $\overline{P}_0(X), \overline{P}_1(X), \dots, \overline{P}_\ell(X)$ with reduced degree $\deg \overline{P}_t(X) < \delta + 1 - t(k-1) - \sum_{i=0}^{k-1}[\nu_i - t]^+$ after the re-encoding transformation by solving (5.21) and (5.22). The solution of the reduced interpolation problem is obtained by summing up:

$$\overline{P}_t(X)\cdot\prod_{i=0}^{k-1}(X-\alpha_i)^{[\nu_i-t]^+}\cdot Y^t$$

for all $t \in [\ell+1)$.

5.2.3 Reduced Set of Homogeneous Equations in Block-Hankel Form

In this subsection, we outline the basic step to get from (5.21) and (5.22) to a reduced set of homogeneous linear equations. Define:

$$\widehat{N}_t = N_t - \sum_{i=0}^{k-1} [\nu_i - t]^+, \quad \forall t \in [\ell). \tag{5.24}$$

From (5.21) and (5.22), we obtain the following linear homogeneous set of equations:

$$\sum_{t=b}^{\ell} \sum_{j=0}^{\widehat{N}_t - 1} \binom{t}{b} \overline{P}_{t,j} \cdot \widetilde{S}^{\langle 0,b \rangle}_{t,j+u} = 0, \quad \forall b \in [\widehat{m}_0), u \in [d_{0,b}),$$

$$\sum_{t=b}^{\ell} \sum_{j=0}^{\widehat{N}_t - 1} \binom{t}{b} \overline{P}_{t,j} \cdot \widehat{S}^{\langle i,b \rangle}_{t,j+u} = 0, \quad \forall b \in [\widehat{m}_i), u \in [d_{i,b}), i \in [1,q),$$

where $\widetilde{S}^{\langle 0,b \rangle}_{t,j}$ are the coefficients of the power series of

$$\frac{K_0(X)^{t-b} \prod_{i=0}^{k-1} (X - \alpha_i)^{[t-\nu_i]^+}}{\widehat{L}^{\langle 0,b \rangle}(X)}$$

and $\widehat{S}^{\langle i,b \rangle}_{t,j}$ are the coefficients of the fraction

$$\frac{\widehat{R}_i(X)^{t-b}}{\widehat{L}^{\langle i,b \rangle}(X)}$$

for all $i \in [1,q)$.

Let $\overline{\mathbf{P}}_t = (\overline{P}_{t,0}, \overline{P}_{t,1}, \dots, \overline{P}_{t,\widehat{N}_t - 1})^T$ be the vector that contains the coefficients of the polynomial $\overline{P}_t(X) \in \mathbb{F}_q[X]$ for all $t \in [\ell+1)$.

$$\begin{pmatrix} \widetilde{\mathbf{S}}^{\langle 0 \rangle} \\ \widehat{\mathbf{S}}^{\langle 1 \rangle} \\ \vdots \\ \widehat{\mathbf{S}}^{\langle q-1 \rangle} \end{pmatrix} \cdot \begin{pmatrix} \overline{\mathbf{P}}_0 \\ \overline{\mathbf{P}}_1 \\ \vdots \\ \overline{\mathbf{P}}_\ell \end{pmatrix} = \mathbf{0}. \tag{5.25}$$

Each matrix $\widehat{\mathbf{S}}^{\langle i \rangle}$ is a $1/2 \sum_{j=0}^{n-1} \widehat{m}_{i,j}(\widehat{m}_{i,j}+1) \times \sum_{t=0}^{\ell} \widehat{N}_t$ for all $i \in [1,q)$ Block-Hankel matrix. The Block-Hankel matrix $\widetilde{\mathbf{S}}^{\langle 0 \rangle}$ is a $1/2 \sum_{j=k}^{n-1} \widehat{m}_{0,j}(\widehat{m}_{0,j}+1) \times \sum_{t=0}^{\ell} \widehat{N}_t$ matrix.

The difference of the number of columns of the $(\widetilde{\mathbf{S}}^{\langle 0 \rangle}\ \widehat{\mathbf{S}}^{\langle 1 \rangle}\ \dots \widehat{\mathbf{S}}^{\langle q-1 \rangle})^T$ compared to the matrix $(\mathbf{S}^{\langle 0 \rangle}\ \mathbf{S}^{\langle 1 \rangle}\ \dots \mathbf{S}^{\langle q-1 \rangle})^T$ of (5.25) is

$$\sum_{t=0}^{\ell} \sum_{i=0}^{k-1} [\nu_i - t]^+,$$

and equals the reduction of homogeneous linear equations.

5.3 Conclusion and Future Work

We proposed the univariate reformulation of the bivariate interpolation problem of Kötter–Vardy for soft-decision decoding GRS codes in Lemma 5.1. The obtain polynomial expression are q Guruswami–Sudan like Key Equations in Section 5.1. The univariate reformulation after the re-encoding transformation was stated in Lemma 5.4 and described in Section 5.2.

We gave the complete algebraic description for both univariate reformulations and proved the main theorems. We shortly outlined the adaption of the FIA for the obtained set of homogeneous equations and roughly estimates its space and time complexity. The adaption of the FIA (or similar algorithm) for the vertically arranged Block-Hankel matrices is an open issue. Furthermore, the re-encoding transformation in general can be applied to related code families as Chinese-Remainder-Theorem or Algebraic-Geometry codes.

6

"We must not forget that when radium was discovered no one knew that it would prove useful in hospitals. The work was one of pure science. And this is a proof that scientific work must not be considered from the point of view of the direct usefulness of it. It must be done for itself, for the beauty of science, and then there is always the chance that a scientific discovery may become like the radium a benefit for humanity."

Marie Curie (1867–1934)

Bounding the Minimum Distance of Cyclic Codes

Although cyclic codes were developed at the end of the 1950s by Prange [O-Pra57], they still play a central role in (distributed) storage and communication systems. However, determining their minimum distance from a given defining set is an open research problem. Vardy [A-Var97] showed that determining the minimum distance of binary linear codes is NP hard (and probably this holds for linear codes over any alphabet size). Dumer *et al.* [A-DMS03] showed the hardness of approximating the minimum distance of linear codes. Therefore, several lower bounds on the minimum distance of linear (cyclic) codes and efficient decoding algorithms up to these bounds exist. This chapter deals with lower bounds on the minimum distance of linear cyclic codes over $\mathbb{F}_q$.

In the 1970s, Hartmann and Tzeng [A-Har72; A-HT72; A-HTC72; A-HT74] generalized the well-known bound by Bose, Ray-Chaudhuri [A-BRC60] and Hocquenghem [A-Hoc59], abbreviated BCH. Feng and Tzeng [A-FT89; A-FT91a] extended the BCH decoding algorithms of Berlekamp–Massey [B-Ber68; A-Mas69] and Sugiyama *et al.* [A-SKHN75; A-SKHN76] to decode in quadratic-time up to the Hartmann–Tzeng bound. Further extensions of the BCH bound were *inter alia* developed by Roos [A-Roo82; A-Roo83], van Lint and Wilson [A-LW86] (denoted as AB or shifting method), Schaub and Massey [O-MS88b; O-Sch88], Duursma and Kötter [A-DK94; O-Kö96b; O-Duu93], Shen [A-SWTS96], Augot and Levy-dit-Vehel [A-AL96], Boston [A-Bos01], Duursma and Pellikaan [A-DP06] as well as Betti and Sala [A-BS06].

An extensive discussion can be found in van Lint's book [B-Lin99, Chapter 6], in the preliminary version of Pellikaan *et. al.* [B-PWBJ12, Chapter 7], Charpin's chapter [O-Cha98] in the Handbook of Coding Theory, Blahut's book [B-PHB98b, Chapter 19], Peterson and Weldon [B-PW72, Chapter 8] and in the book of MacWilliams and Sloane [B-MS88a, Chapter 7]. The survey paper of Augot *et al.* [O-ACS91] also gives an overview on the existing bounds.

Although these improved bounds show that for many codes the actual distance is higher than the BCH bound, there is no general decoding algorithm up the actual distance of cyclic codes.

Hartmann and Tzeng [A-Har72; A-HT74] proposed two variants of an iterative decoding algorithm up to their bound. However, these algorithms require the calculation of missing syndromes and the solution of non-linear equations. An approach for decoding all binary cyclic codes up to their actual minimum distance of length less than 63 was given by Feng and Tzeng [A-TF94]. They use a generalized syndrome matrix and fit the known syndrome coefficients manually for each code into the structure of the matrix. Various other decoding variants exist.

This chapter covers parts of the published work [I-ZWB11; I-ZB12c; A-ZWB12b; A-ZB12a; I-ZB12b; I-ZWZGB13] and unpublished one (especially Section 6.4).

We provide a homepage [O-ZJ12] with numeric results for cyclic code over $\mathbb{F}_2, \mathbb{F}_3, \mathbb{F}_5$ and $\mathbb{F}_7$.

In Section 6.1, we use rational functions to bound the minimum distance of cyclic codes. For this approach, we give an error-only syndrome-based decoding algorithm and derive a generalized Forney formula for the error-evaluation. The bound is denoted by $d^*_{\text{I-a}}$ and is also considered in [I-ZWB11; A-ZWB12b]. It is very close to the one proposed in Section 6.2, which is therefore denoted by $d^*_{\text{I-b}}$. It is based on the embedding of a given cyclic code into a cyclic product code (see Subsection 2.3.2) and is discussed in [I-ZB12c; A-ZB12a; I-ZB12b]. Furthermore, the idea is extended to two other bounds, which are denoted by d^*_{II} and d^*_{III}. While the bound d^*_{II} is straight forward and in the sense of embedding a cyclic code into a cyclic product code, the theorem on the bound d^*_{III} shows a new direction (see also [I-ZWZGB13]).

Good candidates, which can be associated to a given linear cyclic code to bound its minimum distance, are identified in Section 6.3. We give necessary and sufficient conditions for lowest-rate non-primitive binary cyclic codes of minimum distance two. Furthermore, a sufficient condition for non-primitive binary cyclic codes of minimum distance three is derived.

In Section 6.4, we define a cyclic generalized product code and give the basic properties. The possible embedding of a given code into a cyclic generalized product code is outlined.

We conclude and give further research directions in Section 6.5.

6.1 Bounding the Minimum Distance by Rational Functions

6.1.1 Overview

This approach originates from decoding Goppa codes [A-Gop70; A-Gop71] and their generalizations [I-BS97; A-SM81]. We match the roots of an $[n, k, d]_q$ cyclic code to non-zeros of the power series expansion of a rational function. This allows to formulate a new lower bound on the minimum distance of cyclic codes. We identify some classes of cyclic codes and refine the bound on their distance. Our approach covers the class of reversible codes [A-Mas64]. The proposed new lower bound is better than the BCH bound and for most codes also better than the Hartmann–Tzeng bound (see Theorem 2.18). Moreover, we generalized some Boston [A-Bos01] bounds.

In addition, we give an efficient decoding algorithm up to our new bound. This decoding algorithm is based on a generalized Key Equation, a modified Chien search and a generalized Forney's formula [A-For65] for the error evaluation. The time complexity of the whole decoding procedure is quadratic with the length of the cyclic code.

6.1.2 More Preliminaries on Cyclic Codes

> **Lemma 6.1 (Cardinality of Coset)**
> Let s be the smallest integer such that the length n divides $(q^s - 1)$, then the cardinality of the cyclotomic coset $M_{r,q}^{\langle n \rangle}$ as in Definition 2.17 is $|M_{r,q}^{\langle n \rangle}| = s$ if $\gcd(n, r) = 1$.

Proof The cyclotomic coset $M_{r,q}^{\langle n \rangle}$ has cardinality $|M_{r,q}^{\langle n \rangle}| = j$ if and only if j is the smallest integer such that

$$\begin{aligned} r \cdot q^j &\equiv r \mod n \\ \iff r \cdot (q^j - 1) &\equiv 0 \mod n. \end{aligned}$$

Since $\gcd(n, r) = 1$, this is equivalent to $n \mid (q^j - 1)$. Since s is the smallest integer such that the length n divides $(q^s - 1)$, j equals s and hence, $|M_{r,q}^{\langle n \rangle}| = s$. ■

Let us state some preliminaries on rational functions.

Definition 6.2 (Period of a Power Series)
Let a formal power series $a(X) = \sum_{j=0}^{\infty} a_j X^j$ with $a_j \in \mathbb{F}_q$ be given. The period $p(a(X))$ of the infinite sequence $a(X)$ is the smallest p, such that

$$a(X) = \frac{\sum_{j=0}^{p-1} a_j X^j}{-X^p + 1}.$$

Throughout this section, we use the power series expansion of the fraction of two polynomials $h(X)$ and $f(X)$ in $\mathbb{F}_q[X]$ with

$$v \overset{\text{def}}{=} \deg h(X) < u \overset{\text{def}}{=} \deg f(X). \tag{6.1}$$

Let α be an n-th root of unity in some extension field $\mathbb{F}_{q^l}$. We require that:

C1) $\deg \gcd(h(X), f(X)) = 0$, and

C2) $\deg \gcd(f(X\alpha^i), f(X\alpha^j)) = 0, \quad \forall i, j \in [n)$ with $i \neq j$,

to prove our main theorem on the minimum distance.

The following lemma establishes a connection between the length n of the code and the period of the power series of $h(X)/f(X)$, such that C2) is fulfilled.

Lemma 6.3 (Code Length, Period of a Power Series)
Let α be an n-th root of unity of $\mathbb{F}_{q^l}$, where $n|(q^l - 1)$. Let $h(X), f(X) \in \mathbb{F}_q[X]$ with $\deg \gcd(h(X), f(X)) = 0$ and degree as in (6.1) be given. The formal power series over $\mathbb{F}_q$ is defined as:

$$\sum_{j=0}^{\infty} a_j X^j \overset{\text{def}}{=} \frac{h(X)}{f(X)}, \tag{6.2}$$

with period $p = p\big(h(X)/f(X)\big)$ as in Definition 6.2.

If $\gcd(n, p) = 1$, then

$$\deg \gcd\left(f(X\alpha^i), f(X\alpha^j)\right) = 0, \quad \forall i, j \in [n) \text{ with } i \neq j.$$

Proof From Definition 6.2, we have

$$h(X)(-X^p + 1) = f(X)(a_0 + a_1 X + \ldots + a_{p-1} X^{p-1}),$$

and from $\deg \gcd(f(X), h(X)) = 0$, it follows that $-X^p + 1 \equiv 0 \mod f(X)$. Hence, for two different polynomials $f(X\alpha^i)$ and $f(X\alpha^j)$, for any $i, j \in [n)$ with $i \neq j$:

$$X^p \alpha^{ip} - 1 \equiv 0 \mod f(X\alpha^i) \quad \text{and} \tag{6.3}$$

$$X^p \alpha^{jp} - 1 \equiv 0 \mod f(X\alpha^j). \tag{6.4}$$

Assume there is some element $\beta \in \mathbb{F}_{q^{us}}^*$, such that

$$f(\beta\alpha^i) = f(\beta\alpha^j) = 0,$$

$$\text{i.e., } \gcd\big(f(X\alpha^i), f(X\alpha^j)\big) \equiv 0 \mod (X - \beta).$$

Equation (6.3) and (6.4) give the following:

$$\beta^p \alpha^{ip} - 1 = 0 \quad \text{and} \quad \beta^p \alpha^{jp} - 1 = 0 \,.$$

Therefore, $\beta^p \alpha^{ip} = \beta^p \alpha^{jp}$, and $\alpha^{ip} = \alpha^{jp}$, hence, $\alpha^{(i-j)p} = 1$. For any $i \neq j, i, j \in [n)$, this can be true only if $\gcd(p, n) > 1$. ■

6.1.3 Bound I-a: Rational Functions

We directly state the bound on the minimum distance of an $[n, k, d]_q$ cyclic code.

> **Theorem 6.4 (Bound I-a)**
> Let $\mathcal{C}$ be an $[n, k, d]_q$ cyclic code and let α denote an n-th root of unity in some extension of $\mathbb{F}_q$. Let two co-prime polynomials $h(X)$ and $f(X)$ in $\mathbb{F}_q[X]$ with degrees v and u and with
>
> $$\gcd\left(n, p\left(\frac{h(X)}{f(X)}\right)\right) = 1 \qquad \text{and} \qquad \sum_{j=0}^{\infty} a_j X^j = \frac{h(X)}{f(X)}$$
>
> be given. Let a non-zero integer m with $\gcd(m, n) = 1$ be given and let the power series be defined as:
>
> $$a(f, \alpha^{im} X) \overset{\text{def}}{=} \alpha^{if} \sum_{j=0}^{\infty} a_j (\alpha^{im} X)^j = \frac{\alpha^{fi} h(\alpha^{im} X)}{f(\alpha^{im} X)}.$$
>
> Let the integers f, δ with $\delta \geq 2$ be given, such that for all $c(X) \in \mathcal{C}$:
>
> $$\sum_{j=0}^{\infty} a_j c(\alpha^{f+jm}) X^j \equiv 0 \mod X^{\delta-1}. \tag{6.5}$$
>
> Then, the minimum distance d of $\mathcal{C}$ satisfies the following inequality:
>
> $$d \geq d^*_{\text{I-a}} \overset{\text{def}}{=} \left\lceil \frac{\delta - 1 - v}{u} + 1 \right\rceil. \tag{6.6}$$

Proof With $c(X) = \sum_{i \in Y} c_i X^i$, we can rewrite (6.5):

$$\sum_{j=0}^{\infty} a_j c(\alpha^{jm+f}) X^j = \sum_{j=0}^{\infty} \sum_{i \in Y} a_j c_i \alpha^{i(jm+f)} X^j.$$

Interchanging the summation gives us:

$$\begin{aligned}
\sum_{j=0}^{\infty} \sum_{i \in Y} a_j c_i \alpha^{i(jm+f)} X^j &= \sum_{i \in Y} c_i \left(\sum_{j=0}^{\infty} a_j \alpha^{i(jm+f)} X^j \right) \\
&= \sum_{i \in Y} c_i \left(\sum_{j=0}^{\infty} a_j \alpha^{if} (\alpha^{im} X)^j \right).
\end{aligned}$$

We write the power-series as fraction and obtain:

$$\begin{aligned}\sum_{i\in Y} c_i \left(\sum_{j=0}^{\infty} a_j \alpha^{if} (\alpha^{im}X)^j \right) &= \sum_{i\in Y} c_i \frac{\alpha^{if} h(\alpha^{im}X)}{f(\alpha^{im}X)} \\ &\equiv 0 \mod X^{\delta-1}. \end{aligned} \tag{6.7}$$

From Lemma 6.3 and with $\gcd(m,n)=1$, we know that $\deg\gcd\big(f(\alpha^{im}X), f(\alpha^{jm}X)\big)=0,\ \forall i \neq j$. We obtain from (6.7):

$$\frac{\sum\limits_{i\in Y} \Big(c_i \cdot \alpha^{if} \cdot h(\alpha^{im}X) \cdot \prod_{\substack{j\in Y\\ j\neq i}} f(\alpha^{jm}X) \Big)}{\prod_{i\in Y} f(\alpha^{im}X)} \equiv 0 \mod X^{\delta-1}. \tag{6.8}$$

Let $|Y|=d$. The degree of the denominator in (6.8) is ud and the degree of the numerator in (6.8) is at most $(d-1)u+v$ and has to be greater than or equal to $\delta-1$, i.e.,

$$\begin{aligned}(d-1)u+v &\geq \delta-1 \\ d &\geq \left\lceil \frac{\delta-1-v}{u} + 1 \right\rceil. \end{aligned}$$

■

Let us describe Theorem 6.4. According to (6.5), we search the longest "sequence"

$$a_0 c(\alpha^f), a_1 c(\alpha^{f+m}), \ldots, a_{\delta-2} c(\alpha^{f+(\delta-2)m}),$$

that is a zero-sequence, i.e., the product of the coefficient a_j and the evaluated codeword $c(\alpha^{f+jm})$ gives zero for all $j \in [\delta-1)$. We require a root α^{jm} of the code $\mathcal{C}$, if the coefficient a_{j-f} of the power series $a(f, \alpha^{jm}X)$ is non-zero.

Example 6.5 (Binary Cyclic Code)
Consider the $[17,9]_2$ cyclic code $\mathcal{C}$ with defining set

$$\begin{aligned} D = M_{1,2}^{\langle 17 \rangle} &= \{1,2,4,8,16,15,13,9\} \\ &\equiv \{1,2,4,8,-1,-2,-4,-8\} \mod 17. \end{aligned}$$

Let $f=-4$, $m=1$, $h(X)=X+1$ and $f(X)=X^2+X+1 \in \mathbb{F}_2[X]$ be given. Then, $a(-4,\alpha^i X)$ has period three according to Definition 6.2. We have $(a_0\ a_1\ a_2) = (1\ 0\ 1)$ for the first three coefficients, which are repeated periodically.

The following table illustrates how we match the roots of the generator polynomial of $\mathcal{C}$ to the zeros of the power series expansion $a(-4,\alpha^i X)$. In the first row, the defining set D is shown. The □ marks elements that are not necessarily roots of the code. In the second row of the table, the power series expansion $\mathbf{a} = (a_0\ a_1\ a_2\ a_0\ a_1\ \ldots)$ is shown for the considered interval:

D	-4	□	-2	-1	□	1	2	□	4
a	1	0	1	1	0	1	1	0	1

We have $a_j \cdot c(\alpha^{j-4}) = 0,\ \forall j \in [9)$, and for all $c(X) \in \mathcal{C}$. The zero-sequence is of length $\delta-1=9$ and therefore with Theorem 6.4 that $d \geq d^*_{\text{I-a}} = 5$. This is the actual distance d of $\mathcal{C}$.

In the next section, we see that $\mathcal{C}$ of Example 6.5 belongs to the class of reversible codes and we can associate this rational function to the whole class of these codes.

6.1.4 Some Classes of Cyclic Codes

We classify $[n, k, d]_q$ cyclic codes by subsets of their defining set D and their length n. We specify our new lower bound (Theorem 6.4) on the minimum distance for some classes of codes. Additionally, we compare it to the BCH [A-Hoc59; A-BRC60] and the Hartmann–Tzeng [A-HT72] bound as in Theorem 2.18, which we denote by d^*_{BCH} and d^*_{HT}, respectively. We use the following power series expansions $1/f(X)$ over $\mathbb{F}_q$ with period p, where $\mathbf{a} = (a_0\ a_1\ \dots\ a_{p-1})$ denotes the coefficients:

- $1/(X^2 + X + 1)$ over $\mathbb{F}_q$ with $\mathbf{a} = $ (1 -1 0) and $p = 3$,
- $1/(X^3 + X^2 + X + 1)$ over $\mathbb{F}_q$ with $\mathbf{a} = $ (1 -1 0 0) and $p = 4$,
- $1/(X^3 + X + 1)$ over $\mathbb{F}_2$ with $\mathbf{a} = $ (1 1 1 0 1 0 0) and $p = 7$,
- $1/(X^4 + X + 1)$ over $\mathbb{F}_2$ with $\mathbf{a} = $ (1 1 1 1 0 1 0 1 1 0 0 1 0 0 0) and $p = 15$.

We match a power series expansion $a(f, \alpha^{im}X)$ to the roots of the generator polynomial, such that the bound of Theorem 6.4 is maximized. Throughout this section, we assume due to Lemma 6.3 that $\gcd(n, p) = 1$ and we use Theorem 6.4 to state the lower bound $d^*_{\text{I-a}}$ on the distance of the codes.

Table 6.1 shows several power series expansions and their denominator $h(X)$ and numerator $f(X)$. First, we apply our approach to the wide class of reversible codes. Afterwards, we show how our principle can equivalently be used for non-reversible codes.

$(a_0\ a_1\ \dots\ a_{p-1})$	$f(X)$	$h(X)$
(1 -1 0)	$1 + X + X^2$	1
(-1 0 1)	$1 + X + X^2$	$-1 - X$
(0 1 -1)	$1 + X + X^2$	X
(1 -1 0 0)	$1 + X + X^2 + X^3$	1
(0 1 -1 0)	$1 + X + X^2 + X^3$	X
(0 0 1 -1)	$1 + X + X^2 + X^3$	X^2
(-1 0 0 1)	$1 + X + X^2 + X^3$	$-1 - X - X^2$

Table 6.1: Power series $(a_0\ a_1\ \dots\ a_{p-1})$ for some rational functions $h(X)/f(X)$ over $\mathbb{F}_2$.

6.1.5 Reversible Codes

In this subsection, we show how our approach can be applied for a large class of cyclic codes, the class of reversible codes [A-Mas64; B-MS88a]. An $[n, k, d]_q$ code $\mathcal{C}$ with defining set D is reversible if for any codeword

$$(c_0\, c_1\, \dots\, c_{n-1}) \in \mathcal{C} \quad \Longrightarrow \quad (c_{n-1}\, c_{n-2}\, \dots\, c_0) \in \mathcal{C}$$

holds. A cyclic code is reversible if and only if the reciprocal of every zero of the generator polynomial $g(X)$ is also a zero of $g(X)$, i.e.,

$$D = \{i_1, i_2, \dots, i_\ell, -i_1, -i_2, \dots, -i_\ell\}. \tag{6.9}$$

A special class of reversible codes, which we call symmetric reversible codes is given based on the following lemma.

Lemma 6.6 (Symmetric Reversible Codes)
Let n and q with $\gcd(n,q)=1$ be given. Any union of cyclotomic cosets $M_{i,q}^{\langle n\rangle}$ is a defining set of a reversible code if and only if $n \mid (q^m+1)$, for some $m \in \mathbb{N}$.

Proof Any union of cyclotomic cosets defines a reversible code if and only if any coset is reversible, i.e., if for all r and some integer m:

$$M_{r,q}^{\langle n\rangle} = \{r, r\cdot q, \ldots, r\cdot q^{m-1}, -r, -r\cdot q, \ldots, -r\cdot q^{m-1}\}.$$

Therefore for all r, the following has to hold:

$$\begin{aligned} r\cdot q^m &\equiv -r \mod n \\ \iff r\cdot(q^m+1) &\equiv 0 \mod n. \end{aligned}$$

Since $r=1$ always defines a cyclotomic coset, $(q^m+1)\equiv 0 \mod n$ has to hold. This is fulfilled if and only if $n \mid (q^m+1)$ and in this case also $r\cdot(q^m+1)\equiv 0 \mod n$ holds for any r. ■

Moreover, the following lemma provides the cardinality of all cyclotomic cosets if $n \mid (q^m+1)$.

Lemma 6.7 (Cardinality of Symmetric Reversible Codes)
Let m be the smallest integer such that n divides (q^m+1), then the cardinality of the cyclotomic coset $M_{r,q}^{\langle n\rangle}$ is

$$|M_{r,q}^{\langle n\rangle}| = 2m$$

if $\gcd(n,r)=1$.

Proof Since $n \mid (q^m+1)$, it follows also that $n \mid (q^m+1)(q^m-1) = (q^{2m}-1)$. Since m is the smallest integer such that n divides (q^m+1), also $s=2m$ is the smallest integer such that $n \mid (q^s-1)$. With Lemma 6.1, we obtain $|M_{r,q}^{\langle n\rangle}| = s$ if $\gcd(n,r)=1$. Therefore, $|M_{r,q}^{\langle n\rangle}| = s = 2m$. ■

In order to illustrate our bound, we first restrict ourselves to binary codes. To give a new bound on the minimum distance, we first use the rational function $a(X)=h(X)/f(X)$ with $f(X)=X^2+X+1$, where $p(a(X))=3$. For a binary symmetric reversible code $\mathcal{C}$, we know from (6.9) that each cyclotomic coset is symmetric. Therefore, if $\{1\}\subseteq D$, we know that $\{-4,-2,-1,1,2,4\}$ is a subset of the defining set D. Let us use the (cyclically shifted) power series expansion $\mathbf{a}=(-1\ 0\ 1\ \ldots)$. According to Table 6.1, we have $h(X)=-1-X$. We match the roots of $\mathcal{C}$ for $f=-4$ and $m=1$, to a zero-sequence of length $\delta-1=9$. Therefore our bound provides $d\geq d^*_{\text{I-a}}=5$.

Let the defining set D of the binary symmetric reversible code $\mathcal{C}$ additionally include 5. Then we obtain for $f=-6$ and $m=1$ a sequence of length $\delta-1=13$, which results in $d^*_{\text{I-a}}=7$.

In the same way, if $\{1,5,7\}\subseteq D$, we obtain $\delta-1=21$ with $f=-10$ and $m=1$ and thus, $d^*_{\text{I-a}}=11$. These parameters are shown in Table 6.2 and compared with the BCH and Hartmann–Tzeng bound.

As mentioned before, reversible codes are defined such that the reciprocal of each root of the generator polynomial is also a root. Therefore, a defining set where $r\subseteq D$, and also $-r\subseteq D$ defines a reversible code if $\gcd(r,n)=1$ and $\gcd(-r,n)=1$. The conditions are necessary to guarantee that both cyclotomic cosets have the same cardinality (compare Lemma 6.1) and hence each reciprocal root is also in the defining set. The second row of Table 6.2 shows the required subsets of the defining set in order to

Binary Symmetric Reversible	$\{1\} \subseteq D$ $k \geq n - \ell$	$\{1, 5\} \subseteq D$ $k \geq n - 2\ell$	$\{1, 5, 7\} \subseteq D$ $k \geq n - 3\ell$
Binary Reversible	{-1, 1} $\subseteq D$ $k \geq n - 2\ell$	{-5,-1, 1, 5} $\subseteq D$ $k \geq n - 4\ell$	{-7,-5,-1, 1, 5, 7} $\subseteq D$ $k \geq n - 6\ell$
General q-ary	{-4,-2,-1, 1, 2, 4} $\subseteq D$	{-5,-4,-2,-1, 1, 2, 4, 5} $\subseteq D$	{-10,-7,-5,-4,-2,-1, 1, 2, 4, 5, 7, 10} $\subseteq D$
BCH $c(\alpha^{f_1}), \dots,$ $c(\alpha^{f_1+(\delta-2)m_1})$	$d^*_{\text{BCH}} = 4$ $f_1 = -4$ $m_1 = 3$	$d^*_{\text{BCH}} = 5$ $f_1 = -5$ $m_1 = 3$	$d^*_{\text{BCH}} = 8$ $f_1 = -10$ $m_1 = 3$
Hartmann –Tzeng $c(\alpha^{f_1}), \dots$ $c(\alpha^{f_1+(\delta-2)m_1+\nu m_2})$	$d^*_{\text{HT}} = 5$ $f_1 = -4$ $m_1 = 3$ $m_2 = 2$ $\delta = 4$ $\nu = 1$	$d^*_{\text{HT}} = 6$ $f_1 = -5$ $m_1 = 3$ $m_2 = 1$ $\delta = 5$ $\nu = 1$	$d^*_{\text{HT}} = 9$ $f_1 = -10$ $m_1 = 3$ $m_2 = 2$ $\delta = 8$ $\nu = 1$
Fractions $a_0 c(\alpha^f), \dots$ $a_{\delta-2} c(\alpha^{f+m(\delta-2)})$	$d^*_{\text{I-a}} = 5$ $f_1 = -4$ $m = 1$ $\delta = 10$ $\mathbf{a}$ = (-1 0 1)	$d^*_{\text{I-a}} = 7$ $f_1 = -6$ $m = 1$ $\delta = 14$ $\mathbf{a}$ = (0 0 -1)	$d^*_{\text{I-a}} = 11$ $f_1 = -10$ $m = 1$ $\delta = 22$ $\mathbf{a}$ = (-1 0 1)

Table 6.2: Comparison of the BCH and the Hartmann–Tzeng bounds on the minimum distance of q-ary cyclic codes of length n with $\gcd(n, 3) = 1$. The denominator of the rational fraction is $f(X) = X^2 + X + 1$.

obtain the same parameters as for binary symmetric reversible codes. Note that l is the smallest integer such that the length n divides $q^l - 1$.

The third row of Table 6.2 gives these results in general. In Table 6.2, $\gcd(n, p = 3) = 1$ has due to Lemma 6.3.

> **Example 6.8 (Binary Symmetric Reversible Code)**
> The binary $[17, 9, 5]_2$ cyclic code $\mathcal{C}$ from Example 6.5 is a symmetric reversible code since Lemma 6.6 is fulfilled. If $\{1\} \subseteq D$, then
>
> $$\begin{aligned} D &= \{1, 2, 4, 8, 16, 15, 13, 9\} \\ &\equiv \{1, 2, 4, 8, -1, -2, -4, -8\} \mod 17 \end{aligned}$$
>
> and we obtain $d^*_{\text{I-a}} = 5$.

For this class of binary cyclic codes, the bound $d \geq 5$ on the minimum distance can be also obtained by another way. With $f = -4$ and $m = 3$ we know from the BCH bound that the minimum distance is at least four. A binary cyclic code of even weight codewords has the zero in the defining set and we would obtain five consecutive zeros (resulting in a minimum distance of at least six). This implies that a codeword of weight four can not exists and therefore a binary cyclic code $\mathcal{C}(D)$, where

$\{-4,-2,-1,1,2,4\} \subseteq D$, has at least minimum distance five.

In Table 6.3, we list some classes of cyclic codes where the denominator $f(X)$ of the rational function $\alpha^{if}h(\alpha^{im}X)/f(\alpha^{im}X)$ has degree three and the period is $p(1/(X^3+X^2+X+1)) = 4$. The power series expansion is $1/(X^3+X^2+X+1) = (1-X)/(-X^4+1)$. Let us consider the second class, where in the case of a binary symmetric reversible code the set $\{3,5,11\}$ must be in the defining set of the code. The Hartmann–Tzeng bound gives the same lower bound on the minimum distance as our approach $d^*_{\text{HT}} = 5$.

Binary Symmetric Reversible	$\{3,5\} \subseteq D$ $k \geq n-2\ell$	$\{3,5,11\} \subseteq D$ $k \geq n-3\ell$	$\{3,5,11,13\} \subseteq D$ $k \geq n-4\ell$
Binary Reversible	{-5,-3, $3,5\} \subseteq D$ $k \geq n-4\ell$	{-11,-5,-3, $3,$ $5,11\} \subseteq D$ $k \geq n-6\ell$	{-13,-11,-5,-3, $3,5,11,13\} \subseteq D$ $k \geq n-8\ell$
General q-ary	{-6,-5,-3, $3,5,6\} \subseteq D$	{-11,-6,-5,-3, $3,5,6,11\} \subseteq D$	{-13,-11,-6,-5,-3, $3,5,6,11,13\} \subseteq D$
BCH $c(\alpha^{f_1}),\ldots,$ $c(\alpha^{f_1+(\delta-2)m_1})$	$d^*_{\text{BCH}} = 3$ $f_1 = -6$ $m_1 = 1$	$d^*_{\text{BCH}} = 3$ $f_1 = -6$ $m_1 = 1$	$d^*_{\text{BCH}} = 4$ $f_1 = -13$ $m_1 = 1$
Hartmann –Tzeng $c(\alpha^{f_1}),\ldots$ $c(\alpha^{f_1+(\delta-2)m_1+\nu m_2})$	$d^*_{\text{HT}} = 3$ $f_1 = -6$ $m_1 = 1$ $m_2 = 0$ $\delta = 3$	$d^*_{\text{HT}} = 5$ $f_1 = -11$ $m_1 = 8$ $m_2 = 6$ $\delta = 4$ $\nu = 1$	$d^*_{\text{HT}} = 6$ $f_1 = -13$ $m_1 = 8$ $m_2 = 2$ $\delta = 5$ $\nu = 1$
Fractions $a_0 c(\alpha^f),\ldots$ $a_{\delta-2}c(\alpha^{f+m(\delta-2)})$	$d^*_{\text{I-a}} = 4$ $f_1 = -9$ $m = 2$ $\delta = 11$ $\mathbf{a} = (0\ 0\ 1\ \text{-}1)$	$d^*_{\text{I-a}} = 5$ $f_1 = -11$ $m = 2$ $\delta = 13$ $\mathbf{a} = (0\ 0\ 1\ \text{-}1)$	$d^*_{\text{I-a}} = 7$ $f_1 = -17$ $m = 2$ $\delta = 19$ $\mathbf{a} = (0\ 0\ 1\ \text{-}1)$

Table 6.3: Comparison of the BCH and the Hartmann–Tzeng bounds on the minimum distance of q-ary cyclic codes of length n with $\gcd(n,4) = 1$. The denominator of the rational fraction is $f(X) = X^3+X^2+X+1$.

Example 6.9 (Binary Cyclic Code)
The binary $[45,31,4]_2$ cyclic code $\mathcal{C}(D)$ with $\{-5,-3,3,5\} \subseteq D$ has the following defining set

$$D = \{3,5,6,10,12,20,21,24,25,33,35,39,40,42\}.$$

The code $\mathcal{C}(D)$ belong to the class of codes in the first column of Table 6.3. We obtain $d^*_{\text{I-a}} = 4$, which is the actual distance of the code. Note that 3 divides 45 and therefore we cannot use Table 6.2.

6.1.6 Non-Reversible Codes

In this subsection, we use our principle equivalently for non-reversible codes. Some classes of binary cyclic codes are given. The power series expansion of the polynomial $f(X) = X^3 + X + 1$ over $\mathbb{F}_2[X]$ has period $p = 7$. To obtain a bound on the minimum distance, we consider the case of binary cyclic codes, where the defining set D contains the 0. Assume that $\{-3, 0, 1, 7\} \subseteq D$. The sequence of zeros of the binary code can be matched to the rational function for $f = -4$ and $m = 1$. The corresponding distance is then $d^*_{\text{I-a}} = 5$. This and some other combinations of subsets of D are shown in Table 6.4. Another class of binary cyclic codes can be identified using the polynomial $f(X) = X^4 + X + 1$ with $p(1/f(X)) = 15$. We use the shifted power series expansion such that $\mathbf{a} = (1\,0\,0\,1\,0\,0\,0\,1\,1\,1\,1\,0\,1\,0\,1)$.

Binary **Codes**	$\{-3, 0, 1, 7\}$ $\subseteq D$ $k \geq n - 4\ell$	$\{-3, 0, 1, 7, 9\}$ $\subseteq D$ $k \geq n - 5\ell$	$\{-3, 0, 1, 7, 9, 11\}$ $\subseteq D$ $k \geq n - 6\ell$
BCH $c(\alpha^{f_1}), \ldots,$ $c(\alpha^{f_1+(\delta-2)m_1})$	$d^*_{\text{BCH}} = 4$ $f_1 =$ -3 $m_1 = 5$	$d^*_{\text{BCH}} = 4$ $f_1 =$ -3 $m_1 = 5$	$d^*_{\text{BCH}} = 4$ $f_1 =$ -3 $m_1 = 5$
Hartmann **–Tzeng** $c(\alpha^{f_1}), \ldots$ $c(\alpha^{f_1+(\delta-2)m_1+\nu m_2})$	$d^*_{\text{HT}} = 4$ $f_1 =$ -3 $m_1 = 5$ $m_2 = 0$ $\delta = 4$	$d^*_{\text{HT}} = 4$ $f_1 =$ -3 $m_1 = 5$ $m_2 = 0$ $\delta = 4$	$d^*_{\text{HT}} = 4$ $f_1 =$ -3 $m_1 = 5$ $m_2 = 0$ $\delta = 4$
Fractions $a_0 c(\alpha^f), \ldots$ $a_{\delta-2} c(\alpha^{f+m(\delta-2)})$	$d^*_{\text{I-a}} = 5$ $f_1 =$ -4 $m = 1$ $\delta = 14$ $\mathbf{a} = (100110)$	$d^*_{\text{I-a}} = 6$ $f_1 =$ -4 $m = 1$ $\delta = 16$ $\mathbf{a} = (100110)$	$d^*_{\text{I-a}} = 7$ $f_1 =$ -4 $m = 1$ $\delta = 19$ $\mathbf{a} = (100110)$

Table 6.4: Comparison of the BCH and the Hartmann–Tzeng bounds on the minimum distance of q-ary cyclic codes of length n with $\gcd(n, 7) = 1$. The denominator of the rational fraction is $f(X) = X^3 + X + 1$.

As required by Lemma 6.3, we only consider lengths n, such that $\gcd(n, p = 15) = 1$. We can match a concatenation of $\mathbf{a}$ to the roots of the generator polynomial for $f = -6$ and $m = 1$ if $\{1, 3, 9, -3\} \subseteq D$. Our bound on the distance yields $d^*_{\text{I-a}} = 6$, since $\deg f(X) = 4$, whereas the BCH and the Hartmann–Tzeng bound give $d^*_{\text{BCH}} = d^*_{\text{HT}} = 5$.

6.1.7 Generalizing Boston's Bounds

Boston gave ten bounds, denoted by d^*_{BO}, on the minimum distance of an $[n, k, d]_q$ cyclic code in [A-Bos01]. He uses algebraic geometry for the proof. These bounds are each for a specific subset of the defining set and do not consider whole classes of codes. In this section, we show how our approach generalizes some of these bounds.

Six of Boston's ten bounds are given as follows.

Theorem 6.10 (Boston Bounds, [A-Bos01])
The following bounds on the minimum distance of an $[n, k, d]_q$ cyclic code $\mathcal{C}$ with defining set D hold:

B1) If $3 \nmid n$ and $\{0, 1, 3, 4\} \subseteq D$, then $d^*_{\mathsf{BO}} = 4$,
B2) If $\{0, 1, 3, 5\} \subseteq D$, then $d^*_{\mathsf{BO}} = 4$,
B5) If $3 \nmid n$ and $\{0, 1, 3, 4, 6\} \subseteq D$, then $d^*_{\mathsf{BO}} = 5$,
B6) If $4 \nmid n$ and $\{0, 1, 2, 4, 5, 6, 8\} \subseteq D$, then $d^*_{\mathsf{BO}} = 6$,
B7) If $3 \nmid n$ and $\{0, 1, 3, 4, 6, 7\} \subseteq D$, then $d^*_{\mathsf{BO}} = 6$,
B10) If $3 \nmid n$ and $\{0, 1, 3, 4, 6, 7, 9\} \subseteq D$, then $d^*_{\mathsf{BO}} = 7$.

We use again two power series expansions $1/f(X)$. The first power series expansion is $1/(X^2 + X + 1)$ of period $p = 3$ with $(a_0\ a_1\ a_2) = (1\ \text{-}1\ 0)$. The second considered power series expansion $1/(X^2 + 1)$ has period $p = 4$ with $(a_0\ a_1\ a_2\ a_3) = (1\ 0\ \text{-}1\ 0)$. Note that the latter is actually a special case of the BCH bound. Table 6.5 shows the six Boston bounds. Boston's bounds B1 B2, B5, B6 and B7 are special cases of our bound. However, for Boston's bound B10, our approach gives a worse result. Moreover, Boston raised the following question [A-Bos01]:

No	$\mathcal{I} =$	$\mathbf{f(X)}$	$\mathbf{a}$	$\mathbf{d}^*_{\mathsf{I\text{-}a}}$	Conditions
1	[-1, 5]	$X^2 + X + 1$	(0 1 -1 . . .)	4	$\gcd(n, 3) = 1$
2	[0, 6]	$X^2 + 1$	(0 1 0 -1 . . .)	4	$\gcd(n, 2) = 1$
5	[-1, 6]	$X^2 + X + 1$	(0 1 -1 . . .)	5	$\gcd(n, 3) = 1$
6	[-1, 8]	$X^2 + 1$	(0 1 0 -1 . . .)	6	$\gcd(n, 2) = 1$
7	[-1, 8]	$X^2 + X + 1$	(0 1 -1 . . .)	6	$\gcd(n, 3) = 1$
10	[-1, 9]	$X^2 + X + 1$	(0 1 -1 . . .)	6	$\gcd(n, 3) = 1$

Table 6.5: Some of Boston's bounds [A-Bos01] on the minimum distance compared to our approach.

Question 6.11 (Boston's Question, [A-Bos01])
Let $3 \nmid n$ and the set $T = \{0, 1, 3, 4, 6, 7, 9, 10, \dots, r\} \subseteq D$. Is the minimum distance d then $d \geq d^*_{\mathsf{BO}} = |T|$?

Counter-examples show that Boston's conjecture is not true (see Example 6.12), since the actual distance of such codes is not always $d^*_{\mathsf{BO}} = r + 1$. However, using the power series expansion of $1/(X^2 + X + 1)$ with $\mathbf{a} = (0\ 1\ \text{-}1 \dots)$ we obtain $\delta - 1 = r + 2$. The minimum distance of such codes can be bounded by $d^*_{\mathsf{I\text{-}a}} = \lceil (r + 1)/2 + 1 \rceil$ with $u = \deg f(X) = 2$ and $v = h(X) = 1$.

Example 6.12 (Ternary Cyclic Code of Length 20)
Let

$$D = \{0, 1, 2, 3, 4, 6, 7, 8, 9, 10, 12, 14, 16, 18\}$$

be the defining set of a $[20, 6]_3$ cyclic code. For Boston's scheme, we can use $T = \{0, 1, 3, 4, 6, 7, 9, 10, 12\}$ with $|T| = 9$. The actual distance is $d = 8$ and therefore, Boston's conjecture is not true. The BCH bound yields $d^*_{\mathsf{BCH}} = 6$. Our new bound is tight and with $r = 12$, we obtain $d^*_{\mathsf{I\text{-}a}} = \lceil (r + 1)/2 + 1 \rceil = 8$.

6.1.8 Generalized Key Equation and Decoding Algorithm

An efficient error-only decoding algorithm up to the bound $d^*_{\text{I-a}}$ of Theorem 6.4 based on a generalized Key Equation is considered in this subsection.

Let $c(X)$ be a codeword of a given $[n, k, d]_q$ code $\mathcal{C}$. Let $r(X) = c(X) + e(X)$ be the received polynomial, where $e(X) = \sum_{i \in E} e_i X^i \in \mathbb{F}_q[X]$ is the error word and $E = \{j_0, j_1, \dots, j_{\varepsilon-1}\} \subseteq \{0, 1, \dots, n-1\}$ is the set of error positions of cardinality $|E| = \varepsilon$. Let the integers f, m with $\gcd(m, n) = 1$, $\delta \geq 2$ and the two polynomials $h(X), f(X) \in \mathbb{F}_q[X]$ with $\deg f(X) = u < \deg h(X) = v$ be given as in Theorem 6.4 for $d^*_{\text{I-a}}$. We define the syndrome polynomial $S(X)$:

$$\begin{aligned} S(X) &\equiv \sum_{i=0}^{n-1} r_i \frac{\alpha^{if} h(\alpha^{im} X)}{f(\alpha^{im} X)} \\ &= \sum_{i \in E} e_i \frac{\alpha^{if} h(\alpha^{im} X)}{f(\alpha^{im} X)} \mod X^{\delta-1}. \end{aligned} \tag{6.10}$$

Thus, with $\sum_{j=0}^{\infty} a_j X^j = h(X)/f(X)$ the explicit form of the syndrome polynomial can be written as:

$$S(X) = \sum_{j=0}^{\delta-2} a_j r(\alpha^{f+jm}) X^j = \sum_{j=0}^{\delta-2} a_j e(\alpha^{f+jm}) X^j. \tag{6.11}$$

We introduce a generalized error-locator polynomial $\Lambda(X)$ and error-evaluator polynomial $\Omega(X)$ and relate it to the syndrome definition of (6.10). Let E denote the set of error positions and let $\varepsilon = |E|$. We define $\Lambda(X)$ as:

$$\Lambda(X) \stackrel{\text{def}}{=} \prod_{i \in E} f(\alpha^{im} X). \tag{6.12}$$

Let

$$\Omega(X) \stackrel{\text{def}}{=} \sum_{i \in E} \Big(e_i \cdot \alpha^{if} \cdot h(\alpha^{im} X) \cdot \prod_{\substack{j \in E \\ j \neq i}} f(\alpha^{jm} X) \Big), \tag{6.13}$$

and we obtain with (6.11) a generalized Key Equation:

$$\begin{gathered} \Lambda(X) \cdot S(X) \equiv \Omega(X) \mod X^{\delta-1} \quad \text{with} \\ \deg \Omega(X) \leq (\varepsilon - 1)u + v < \deg \Lambda(x) = \varepsilon u, \end{gathered} \tag{6.14}$$

since $v < u$.

The main step of our decoding algorithm is to determine $\Lambda(X)$ and $\Omega(X)$ if $S(X)$ is given. The following lemma shows that there is a unique solution for $\Lambda(X)$ if the number of errors is not too big.

Lemma 6.13 (Solving the Key Equation)
Let $S(X)$ with $\deg S(X) = \delta - 2$ be given as in (6.11). If

$$\varepsilon = |E| \leq \left\lfloor \frac{d^*_{\text{I-a}} - 1}{2} \right\rfloor, \tag{6.15}$$

there is a unique solution (up to a scalar factor) of the Key Equation (6.14) with $\deg \Omega(X) \leq (\varepsilon - 1)u + v < \deg \Lambda(X) = \varepsilon u$. We can find this solution by the EEA, i.e., Algorithm 3.1 with input polynomials $X^{\delta-1}$ and $S(X)$ and stopping criteria $\mathsf{crit} = \{u_i < \delta/2 - 1\}$.

Proof For the explicit proof we refer to [B-MS88a, Theorem 16, p. 367], where it is shown that there is a unique solution of the generalized Key Equation (6.14) and that the EEA finds it if

$$\deg \Lambda(X) = \varepsilon u \leq \left\lfloor \frac{\delta - 1}{2} \right\rfloor ,$$

and therefore

$$\varepsilon \leq \left\lfloor \frac{\delta - 1}{2u} \right\rfloor = \left\lfloor \frac{(d^*_{\text{I-a}} - 1)u + v}{2u} \right\rfloor = \left\lfloor \frac{(d^*_{\text{I-a}} - 1)}{2} \right\rfloor , \tag{6.16}$$

since $v/2u < 1/2$. ■

Then, we obtain the unique (except for a scalar factor) solution for $\Lambda(X)$ and $\Omega(X)$ of (6.14), if (6.15) holds.

The Key Equation (6.14) can be written as a linear system of equations, with $\varepsilon u + 1$ coefficients of $\Lambda(X)$. If we consider only the equations which do not depend on $\Omega(X)$, we obtain:

$$\begin{pmatrix} S_0 & S_1 & \dots & S_{\varepsilon u} \\ S_1 & S_2 & \dots & S_{\varepsilon u+1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{\delta-\varepsilon u-2} & S_{\delta-\varepsilon u-1} & \dots & S_{\delta-2} \end{pmatrix} \cdot \begin{pmatrix} \Lambda_{\varepsilon u} \\ \Lambda_{\varepsilon u-1} \\ \vdots \\ \Lambda_0 \end{pmatrix} = \mathbf{0}. \tag{6.17}$$

There is a unique solution up to a scalar factor if and only if the rank of the syndrome matrix is εu. One coefficient of $\Lambda(X)$ can be chosen arbitrarily (here $\Lambda_0 = 1$), since a scalar factor does not change the roots. From this we obtain the same condition on the decoding radius as in Lemma 6.13.

If we have found $\Lambda(X)$, we can determine its factors $f(\alpha^{im}X)$, where $i \in E$. These factors are disjoint since

$$\deg(\gcd(f(\alpha^{im}X), f(\alpha^{jm}X))) = 0, \quad \forall i \neq j$$

and therefore these factors provide the error positions. We calculate only one root β_i of each $f(\alpha^{im}X)$ in a preprocessing step. To find the error positions if $\Lambda(X)$ is given, we perform a Chien search with $\beta_0, \beta_1, \dots, \beta_{n-1}$. This is shown in Algorithm 6.1 and Theorem 6.15 proves that each β_i uniquely determines $f(\alpha^{im}X)$.

For the non-binary case, we have to calculate the error values at the error positions. This can be done by a generalized Forney's formula [A-For65]. In order to obtain this error evaluation formula, we use the explicit expression for $\Omega(X)$ from (6.13). As mentioned before, the preprocessing step calculates n values $\beta_0, \beta_1, \dots, \beta_{n-1}$ such that

$$\begin{aligned} f(\alpha^i \beta_i) &= 0, \quad \forall i \in [n) \quad \text{and} \\ f(\alpha^j \beta_i) &\neq 0, \quad \forall j \neq i. \end{aligned}$$

The evaluation of $\Omega(X)$ at β_ℓ, $\ell \in E$, yields:

$$\Omega(\beta_\ell) = \sum_{i \in E} \Big(e_i \cdot \alpha^{if} \cdot h(\alpha^{im}\beta_\ell) \cdot \prod_{\substack{j \in E \\ j \neq i}} f(\alpha^{jm}\beta_\ell) \Big).$$

With $f(\alpha^\ell \beta_\ell) = 0$, the product $\prod_{\substack{j \in E \\ j \neq i}} f(\alpha^j \beta_\ell)$ is zero if $\ell \in E \backslash \{i\}$ and non-zero only if $\ell = i$. Hence, we obtain

$$\Omega(\beta_\ell) = e_\ell \cdot \alpha^{\ell f} \cdot h(\alpha^{\ell m}\beta_\ell) \cdot \prod_{\substack{j \in E \\ j \neq \ell}} f(\alpha^{jm}\beta_\ell). \tag{6.18}$$

This derivation provides the following lemma.

Lemma 6.14 (Generalized Error Evaluation)
Let α be an n-th root of unity. Let the integers m, f and the polynomials $h(\alpha^i X)$, $f(\alpha^i X)$, $\Lambda(X) = \prod_{i \in E} f(\alpha^{im} X)$ and $\Omega(X)$ from (6.13), for all $i \in [n)$ with $\deg(\gcd(f(\alpha^i X), f(\alpha^j X))) = 0$ be given. Then, the error values e_ℓ for all $\ell \in \mathcal{E}$ are given by

$$\begin{aligned} e_\ell &= \frac{\Omega(\beta_\ell)}{\alpha^{\ell f} \cdot h(\alpha^{\ell m} \beta_\ell) \prod_{\substack{j \in E \\ j \neq \ell}} f(\alpha^{jm} \beta_\ell)} \\ &= \frac{\Omega(\beta_\ell) \cdot f'(\alpha^{\ell m} \beta_\ell)}{\alpha^{\ell f} \cdot h(\alpha^{\ell m} \beta_\ell) \cdot \Lambda'(\beta_\ell)}, \end{aligned} \tag{6.19}$$

where $f'(\alpha^i X)$ and $\Lambda'(X)$ denote the first derivatives of $f(\alpha^i X)$ and $\Lambda(X)$ respectively.

Proof The lemma follows from (6.18) and the fact that

$$\Lambda'(X) = \sum_{i \in E} f'(\alpha^{im} X) \prod_{\substack{j \in E \\ j \neq i}} f(\alpha^{jm} X)$$

and therefore

$$\Lambda'(\beta_\ell) = f'(\alpha^{\ell m} \beta_\ell) \prod_{\substack{j \in E \\ j \neq \ell}} f(\alpha^{jm} \beta_{\ell m}). \qquad \blacksquare$$

The classical Forney formula [A-For65], is obtained from (6.19) for

$$a(f, \alpha^{im} X) = \frac{\alpha^{if}}{1 - \alpha^{im} X}.$$

The decoding approach is summarized in Algorithm 6.1 and its correctness is proved in Theorem 6.15.

Algorithm 6.1: $c(X) = \text{DECODEFRACTION}(r(X), f(X), h(X), f, m, \delta)$

```
Input: Received word $r(X)$, $f(X)$, $h(X)$, Parameters $f$, $m$ and $\delta$
Output: Estimated codeword $c(X)$ or DECODING FAILURE
Preprocessing:
   for all $i \in [n]$: calculate one root $\beta_i$ of $f(\alpha^i X)$
1  Calculate $S(X)$ by (6.11)                                    // Syndrome Calculation
2  Set $\mathsf{crit} = \{\deg u_i < \delta/2 - 1\}$
3  $\_, \Lambda(X), \Omega(X) = \text{EEA}\big(X^{\delta-1}, S(X), \mathsf{crit}\big)$    // Modified Euclidean Algorithm
4  Find all $i$, where $\Lambda(\beta_i) = 0 \Rightarrow E = \{i_0, i_1, \dots, i_{\varepsilon-1}\}$    // Chien-like search
5  if $\varepsilon u < \deg \Lambda(X)$ then
6      Declare DECODING FAILURE
7  else
8      Determine error values $e_{i_0}, e_{i_1}, \dots, e_{i_{\varepsilon-1}}$ by (6.19)
9      $e(X) \leftarrow \sum_{\ell \in E} e_\ell X^\ell$
10     $c(X) \leftarrow r(X) - e(X)$
```

The sign ␣ in Line 3 of Algorithm 6.1 indicates that the returned polynomial is not needed for further calculations.

> **Theorem 6.15 (Correctness and Complexity of Algorithm 6.1)**
> Let $r(X)$ be the received word and let
>
> $$d(r(X), c(X)) \leq \lfloor (d^*_{\text{I-a}} - 1)/2 \rfloor$$
>
> for some codeword $c(X) \in \mathcal{C}$, then Algorithm 6.1 returns $c(X)$ with complexity $\mathcal{O}((\deg f(X) \cdot n)^2)$ operations in $\mathbb{F}_{q^l}$.

PROOF Let the syndrome polynomial $S(X)$ be defined by (6.11). As shown in Lemma 6.13, we can then solve the Key Equation uniquely for $\Lambda(X)$ if $\varepsilon \leq \big\lfloor (d^*_{\text{I-a}} - 1)/2 \big\rfloor$. Therefore, we obtain $\Lambda(X) = \prod_{i \in E} f(\alpha^i X)$ with $\deg \Lambda(X) = \varepsilon u$ as in Line 3 of Algorithm 6.1 and also the error-evaluator polynomial $\Omega(X)$ from Algorithm 3.2 with stopping criteria $\mathsf{crit} = \{u_i < \delta/2 - 1\}$.

To explain the preprocessing and the Chien search, we recall that for each polynomial $a(X)$ of degree u defined over $\mathbb{F}_{q^l}$ there exists a splitting field, i.e., an extension field $\mathbb{F}_{q^{us}}$ of $\mathbb{F}_{q^s}$, in which $a(X)$ has u roots. Therefore, each $f(\alpha^i X)$ can be decomposed into $u = \deg f(\alpha^i X)$ linear factors over a field $\mathbb{F}_{q^{us}}$. These factors are disjoint since $\deg(\gcd(f(\alpha^i X), f(\alpha^j X))) = 0$ and hence, one root of $f(\alpha^i X)$ uniquely defines $f(\alpha^i X)$ and i. Hence, $\Lambda(\beta_j) = 0$ if and only if $j \in E$ and in Line 4 of Algorithm 6.1 the error positions are correctly identified.

Lemma 6.14 proves the generalized error evaluation and therefore, if

$$d(r(X), c(X)) \leq \lfloor (d^*_{\text{I-a}} - 1)/2 \rfloor$$

for some codeword $c(X) \in \mathcal{C}$, Algorithm 6.1 returns $c(X)$, otherwise a decoding failure.

To prove the complexity, we note that the input polynomials $S(X)$ and $X^{\delta-1}$ of the EEA have degrees at most $\delta - 2$ and $\delta - 1$, respectively. Therefore, the complexity of the EEA is quadratic in δ, i.e., $\mathcal{O}(\delta^2) \approx \mathcal{O}((u \cdot d^*_{\text{I-a}})^2)$. The Chien search and the generalized error evaluation require the same complexity as for the classical case, which is $\mathcal{O}(n^2)$. Therefore, we can upper bound the complexity of Algorithm 6.1 by

$$\mathcal{O}((u \cdot n)^2) = \mathcal{O}((\deg f(X) \cdot n)^2). \qquad \blacksquare$$

We consider the $[17, 9, 5]_2$ code from Example 6.5 to illustrate Algorithm 6.1 in the following.

Example 6.16 (Decoding Binary Cyclic Code of Length 17)
We consider again the $[17, 9, 5]_2$ cyclic code as in Example 6.5. The associated power series $a(-4, \alpha^i X)$ up to the $\delta - 2$ coefficient is:

$$\begin{aligned} a(-4, \alpha^i X) &= \frac{\alpha^{i13} \cdot h(\alpha^i X)}{f(\alpha^i X)} \\ &= \frac{\alpha^{13i} + \alpha^{14i} X}{1 + \alpha^i X + \alpha^{2i} X^2} \\ &= \alpha^{13i} + \alpha^{15i} X^2 + \alpha^{16i} X^3 + \alpha^i X^5 + \alpha^{2i} X^6 + \alpha^{4i} X^8 + \dots . \end{aligned} \tag{6.20}$$

For the syndrome polynomial, we obtain with $\delta - 1 = 9$ and (6.10), (6.11) and (6.20):

$$\begin{aligned} S(X) &= \sum_{i=0}^{n-1} e_i \cdot (\alpha^{13i} + \alpha^{15i} X^2 + \dots + \alpha^{4i} X^8) \\ &= \sum_{i \in \mathcal{E}} (\alpha^{13i} + \alpha^{15i} X^2 + \dots + \alpha^{4i} X^8) \\ &= r(\alpha^{13}) + r(\alpha^{15}) X^2 + \dots + r(\alpha^4) X^8 \\ &= S_0 + S_2 X^2 + S_3 X^3 + S_5 X^5 + S_6 X^6 + S_8 X^8. \end{aligned}$$

As in Algorithm 6.1, we call the EEA with the above syndrome polynomial as follows:

$$\text{EEA}\big(X^9, S(X), \{u_i < 4\}\big).$$

Assume, two errors occurred, then we obtain $\Lambda(X)$ with $\deg \Lambda(X) = \varepsilon u = 2 \cdot 2 = 4$.

Using the EEA is equivalent to solving the following system of equations for $\Lambda(X)$

$$\begin{pmatrix} S_0 & 0 & S_2 & S_3 & 0 \\ 0 & S_2 & S_3 & 0 & S_5 \\ S_2 & S_3 & 0 & S_5 & S_6 \\ S_3 & 0 & S_5 & S_6 & 0 \end{pmatrix} \cdot \begin{pmatrix} \Lambda_4 \\ \Lambda_3 \\ \vdots \\ \Lambda_0 \end{pmatrix} = \mathbf{0},$$

and with both approaches, $\Lambda(X)$ has the roots $f(\alpha^i X) = 1 + \alpha^i X + (\alpha^i X)^2, \forall i \in E$. We know that each $f(\alpha^i X) = 1 + \alpha^i X + (\alpha^i X)^2$ has two roots in $\mathbb{F}_{2^8}$ which are unique. We have a look-up-table with one root β_i of each $f(\alpha_i X)$ and we perform a Chien search for $\Lambda(X)$ with $\beta_0, \beta_1, \dots, \beta_{n-1}$. Since this is a binary code, we do not need an error evaluation and can reconstruct the error.

6.2 Bounding Minimum Distance by Embedding a Cyclic Code Into a Cyclic Product Codes

6.2.1 Bound I-b: Basic Idea

We embed a given cyclic code into a cyclic product code, as defined in Section 2.3.2, to bound its minimum distance. The results are similar to those obtained in Section 6.1, but we think it is more elegant and gives new insights. The bound derived in this section coincides in several cases with $d^*_{\text{I-a}}$. Therefore, it is denoted by $d^*_{\text{I-b}}$.

To see the connection, let us prove the following lemma.

Lemma 6.17 (Rational Function and Evaluated Codeword Sequence)
Let $c(X) = \sum_{i \in Y} c_i X^i$ be a codeword of a given $[n, k, d]_q$ cyclic code $\mathcal{C}$. Let α denote an n-th root of unity in some extension field of $\mathbb{F}_q$ and let f and m be two integers with $\gcd(m, n) = 1$. Then, the power series

$$\sum_{i=0}^{\infty} c(\alpha^{f+im}) X^i \tag{6.21}$$

equals the one of Theorem 6.4 with

$$a(f, \alpha^{jm} X) = \frac{\alpha^{jf} h(\alpha^{jm} X)}{f(\alpha^{jm} X)}, \tag{6.22}$$

where:

$$h(\alpha^{im} X) = \sum_{j \in Y} c_j \alpha^{fj} \prod_{\substack{\ell \in Y \\ \ell \neq j}} (1 - \alpha^{m\ell X}),$$

$$f(\alpha^{im} X) = \prod_{j \in Y} (1 - \alpha^{mj} X).$$

Proof Let us write (6.21) more explicitly:

$$\sum_{i=0}^{\infty} c(\alpha^{f+im}) X^i = \sum_{i=0}^{\infty} \sum_{j \in Y} c_j \alpha^{(f+im)j} X^i = \sum_{j \in Y} \sum_{i=0}^{\infty} c_j \alpha^{(f+im)j} X^i$$
$$= \sum_{j \in Y} \sum_{i=0}^{\infty} c_j \alpha^{fj} \alpha^{mji} X^i = \sum_{j \in Y} \sum_{i=0}^{\infty} c_j \alpha^{fj} (\alpha^{mj} X)^i.$$

Using the geometric series, we obtain:

$$\sum_{j \in Y} \sum_{i=0}^{\infty} c_j \alpha^{fj} (\alpha^{mj} X)^i = \sum_{j \in Y} \frac{c_j \alpha^{fj}}{1 - \alpha^{mj} X}$$
$$= \frac{\sum_{j \in Y} c_j \alpha^{fj} \prod_{\ell \neq j} (1 - \alpha^{m\ell} X)}{\prod_{j \in Y} (1 - \alpha^{mj} X)}. \qquad \blacksquare$$

Let us restate [A-ZB12a, Theorem 2] on the minimum distance of cyclic codes using cyclic product codes.

Theorem 6.18 (BCH Bound Generalization — Bound I-b)
Let an $[n_a, k_a, d_a]_q$ cyclic code $\mathcal{A}$ and a second $[n_b, k_b, d_b]_q$ cyclic code $\mathcal{B}$ with $\gcd(n_a, n_b) = 1$ be given. Let α be an element of order n_a in $\mathbb{F}_{q^{l_a}}$ and β of order n_b in $\mathbb{F}_{q^{l_b}}$ respectively. Let five integers $f_1, f_2, m_1, m_2, \delta$ with $m_1 \neq 0, m_1 \neq 0, \gcd(n_a, m_1) = \gcd(n_b, m_2) = 1$ and $\delta \geq 2$ be given, such that:

$$\sum_{i=0}^{\infty} a(\alpha^{f_1+im_1}) \cdot b(\beta^{f_2+im_2})X^i \equiv 0 \mod X^{\delta-1} \tag{6.23}$$

holds for all codewords $a(X) \in \mathcal{A}$ and $b(X) \in \mathcal{B}$. Then, we have:

$$d_a \geq d^*_{\text{I-b}} = \left\lceil \frac{\delta}{d_b} \right\rceil.$$

The polynomial of (6.23) has coefficients in $\mathbb{F}_{q^l}[X]$, where $l = \text{lcm}(l_a, l_b)$.

Proof From Theorem 2.21 we know that (6.23) corresponds to $\delta - 1$ consecutive zeros in the defining set D of $\mathcal{C} = \mathcal{A} \otimes \mathcal{B}$ and therefore its distance $d = d_a d_b$ is greater than or equal to δ according to the BCH bound. ■

Moreover, this yields the following explicit relation.

Lemma 6.19 (Explicit Relation for Bound I-b)
Let the integers $f_1, f_2, m_1, m_2, \delta$ with $m_1 \neq 0, m_2 \neq 0, \gcd(n_a, m_1) = \gcd(n_b, m_2) = 1, \delta \geq 2$ and the two cyclic codes $\mathcal{A}$ and $\mathcal{B}$ be given as in Theorem 6.18. Furthermore, let two integers u and v be given, such that $un_a + vn_b = 1$. Then, the two integers:

$$\begin{aligned} f &= f_1 \cdot v^2 \cdot n_b + f_2 \cdot u^2 \cdot n_a, \qquad \text{and} \\ m &= m_1 \cdot v^2 \cdot n_b + m_2 \cdot u^2 \cdot n_a, \end{aligned}$$

denote the parameters such that:

$$\sum_{i=0}^{\infty} c(\gamma^{f+im})X^i \equiv 0 \mod X^{\delta-1} \tag{6.24}$$

holds for all $c(X) \in \mathcal{A} \otimes \mathcal{B}$, where γ is an element of order $n_a n_b$ in $\mathbb{F}_{q^l}$.

Proof Let $g_a(X)$ be the generator polynomial of $\mathcal{A}$ and $g_b(X)$ that of $\mathcal{B}$. From Theorem 2.21 we know that if α^i is a root of $g_a(X)$, then γ^{vi} is a root of $g(X)$ as in (2.22) and γ^{ui} is a root of $g(X)$ if β^i is a root of $g_b(X)$. Therefore we want $f + im \equiv v(f_1 + im_1) \mod n_a$ and $f + im \equiv u(f_2 + im_2) \mod n_b$ and the Chinese Remainder Theorem gives the result. ■

Example 6.20 (BCH Bound of the Cyclic Product Code)
Let $\mathcal{A}$ be the binary reversible $[17, 9, 5]_2$ code as in Example 6.5 with defining set:

$$D_{\mathcal{A}} = \{1, 2, 4, 8, -8, -4, -2, -1\},$$

and let $\mathcal{B}$ denote the binary $[3, 2, 2]_2$ single parity check code with $D_{\mathcal{B}} = \{0\}$. Let $\alpha \in \mathbb{F}_{2^8}$ and $\beta \in \mathbb{F}_{2^4}$ denote elements of order 17 and 3, respectively. Then, we know that for $f_1 = -4$, $f_2 = -1$ and $m_1 = m_2 = 1$ Theorem 6.18 holds for $\delta = 10$ and therefore $d_a \geq 5$, which is the true minimum distance of $\mathcal{A}$.

Since $-1 \cdot 17 + 6 \cdot 3 = 1$, according to Theorem 2.22 the defining set of the cyclic product code $\mathcal{A} \otimes \mathcal{B}$ is:

$$\begin{aligned} D_{\mathcal{A}\otimes\mathcal{B}} &= \Big\{\{3,5,6,7,10,11,12,14\} \cup \{20,22,23,24,27,28,29,31\} \cup \\ &\qquad \{37,39,40,41,44,45,46,48\} \cup \{0\} \cup \{3\} \cup \cdots \cup \{48\}\Big\} \\ &= \Big\{0,3,5,6,7,9,10,11,12,14,15,18,20,21,22,23,24,27,28,29,30,31,33, \\ &\qquad 36,37,39,40,41,42,44,45,46,48\Big\}, \end{aligned}$$

and Lemma 6.19 gives $f = 10$ and $m = 23$.

6.2.2 Syndrome-Based Error/Erasure Decoding Approach up to Bound I-b

Let the set $E = \{i_0, i_1, \ldots, i_{\varepsilon-1}\}$ with cardinality $|E| = \varepsilon$ be the set of erroneous positions. The corresponding error polynomial is denoted by $e(X) = \sum_{i \in E} e_i X^i$. Let ? mark an erasure and let the set $Z = \{j_0, j_1, \ldots, j_{\zeta-1}\}$ with cardinality $|Z| = \zeta$ be the set of erased positions. Let the received polynomial

$$\widetilde{r}(X) = \sum_{i=0}^{n-1} \widetilde{r}_i X^i$$

with $\widetilde{r}_i \in \mathbb{F}_q \cup \{?\}$ be given.

In the first step of the decoding process, the erasures in $\widetilde{r}(X)$ are substituted by an arbitrary element from $\mathbb{F}_q$. For simplicity, it is common to choose the zero-element. Thus, the corresponding erasure polynomial in $\mathbb{F}_q[X]$ is denoted by

$$z(X) = \sum_{i \in Z} z_i X^i,$$

where $\widetilde{r}_i + z_i = a_i + z_i = 0, \ \forall i \in Z$. Let the modified received polynomial $r(X) \in \mathbb{F}_q[X]$ be

$$r(X) = \sum_{i=0}^{n-1} r_i X^i = a(X) + e(X) + z(X), \tag{6.25}$$

where $a(X) \in \mathcal{A}$.

Definition 6.21 (Syndromes for Bound I-b)
Let an $[n_a, k_a, d_a]_q$ cyclic code $\mathcal{A}$ and a second $[n_b, k_b, d_b]_q$ code $\mathcal{B}$ with $\gcd(n_a, n_b) = 1$ be given. Furthermore, let the five integers f_1, f_2, m_1, m_2, δ and the second code $\mathcal{B}$ be given, such that Theorem 6.18 holds. Let the modified received polynomial $r(X) \in \mathbb{F}_q[X]$ as in (3.18) be given. Let $b(X) \in \mathcal{B}$ denote a codeword of weight d_b.

Then, we define a syndrome polynomial $S(X) \in \mathbb{F}_{q^l}[X]$ as follows:

$$S(X) \overset{\text{def}}{\equiv} \sum_{i=0}^{\infty} r(\alpha^{f_1+im_1}) \cdot b(\beta^{f_2+im_2}) X^i \mod X^{\delta-1} \tag{6.26}$$

$$= \sum_{i=0}^{\delta-2} \Big(e(\alpha^{f_1+im_1}) + z(\alpha^{f_1+im_1}) \Big) \cdot b(\beta^{f_2+im_2}) X^i. \tag{6.27}$$

Since we know the positions of the erasures, we can compute an erasure-locator polynomial similar to the error/erasure decoding of GRS codes as discussed in Subsection 3.2.2.

Definition 6.22 (Erasure-Locator Polynomial)
Let the set Z with $|Z| = \zeta$ and a codeword $b(X) = \sum_{i \in W} b_i X^i \in \mathcal{B}$ with weight d_b be given. Then we define an erasure-locator polynomial $\Psi(X) \in \mathbb{F}_{q^l}[X]$ as follows:

$$\Psi(X) \overset{\text{def}}{=} \prod_{i \in Z} \left(\prod_{j \in W} \left(1 - X\alpha^i \beta^j\right) \right). \tag{6.28}$$

Note that $\Psi(X)$ has degree $\zeta \cdot d_b$. As for the GRS approach in Lemma 3.8, we define a modified syndrome polynomial $\widetilde{S}(X)$ and point out (in the following lemma), which coefficients of $\widetilde{S}(X)$ depend only on the error $e_{i_0}, e_{i_1}, \dots, e_{i_{\varepsilon-1}}$.

Lemma 6.23 (Modified Syndrome Polynomial)
Let the erasure-locator polynomial $\Psi(X)$ of Definition 6.22 and the syndrome polynomial $S(X)$ of Definition 6.21 be given. Then the highest $\delta - 1 - \zeta \cdot d_b$ coefficients of

$$\widetilde{S}(X) \overset{\text{def}}{\equiv} \Psi(X) \cdot S(X) \mod X^{\delta-1} \tag{6.29}$$

depend only on the error polynomial $e(X)$.

Proof From (6.26), we have:

$$\begin{aligned} &\sum_{i=0}^{\infty} r(\alpha^{f_1+im_1}) \cdot b(\beta^{f_2+im_2}) X^i \\ &\quad \equiv \sum_{i=0}^{\infty} \Big(e(\alpha^{f_1+im_1}) + z(\alpha^{f_1+im_1}) \Big) b(\beta^{f_2+im_2}) X^i \mod X^{\delta-1} \\ &\quad \equiv \sum_{i=0}^{\infty} \left(\sum_{j \in E} e_j \alpha^{j(f_1+im_1)} + \sum_{j \in Z} z_j \alpha^{j(f_1+im_1)} \right) b(\beta^{f_2+im_2}) X^i \mod X^{\delta-1}, \end{aligned}$$

and with (6.22) for $b(X) = \sum_{i \in W} b_i X^i$, we can write:

$$\begin{aligned} S(X) \equiv & \sum_{i \in E} e_i \alpha^{f_1 + i m_1} \sum_{j \in W} \frac{b_j}{1 - X\alpha^i \beta^j} + \\ & \sum_{i \in Z} z_i \alpha^{f_1 + i m_1} \sum_{j \in W} \frac{b_j}{1 - X\alpha^i \beta^j} \mod X^{\delta-1} \\ \equiv & \sum_{i \in E} e_i \alpha^{f_1 + i m_1} \frac{\sum\limits_{j \in Z} \Big(b_j \prod\limits_{\substack{\ell \in Z \\ \ell \neq j}} (1 - X\alpha^i \beta^\ell) \Big)}{\prod\limits_{j \in Z} \big(1 - X\alpha^i \beta^j\big)} + \\ & \sum_{i \in Z} z_i \alpha^{f_1 + i m_1} \frac{\sum\limits_{j \in Z} \Big(b_j \prod\limits_{\substack{\ell \in Z \\ \ell \neq j}} (1 - X\alpha^i \beta^\ell) \Big)}{\prod\limits_{j \in Z} \big(1 - X\alpha^i \beta^j\big)} \mod X^{\delta-1}, \end{aligned}$$

and finally, we obtain:

$$\begin{aligned} S(X) \equiv & \frac{\overbrace{\sum\limits_{i \in E} \Big(e_i \alpha^{f_1 + i m_1} \sum\limits_{j \in W} \Big(b_j \prod\limits_{\substack{\ell \in W \\ \ell \neq j}} (1 - X\alpha^i \beta^\ell) \Big) \prod\limits_{\substack{m \in E \\ m \neq i}} \prod\limits_{s \in W} (1 - X\alpha^m \beta^s) \Big)}^{\overset{\text{def}}{=} \Omega(X)}}{\prod\limits_{i \in E} \Big(\prod\limits_{j \in W} \big(1 - X\alpha^i \beta^j\big) \Big)} + \\ & \frac{\overbrace{\sum\limits_{i \in Z} \Big(z_i \alpha^{f_1 + i m_1} \sum\limits_{j \in W} \Big(b_j \prod\limits_{\substack{\ell \in W \\ \ell \neq j}} (1 - X\alpha^i \beta^\ell) \Big) \prod\limits_{\substack{m \in Z \\ m \neq i}} \prod\limits_{s \in W} (1 - X\alpha^m \beta^s) \Big)}^{\overset{\text{def}}{=} A(X)}}{\prod\limits_{i \in Z} \Big(\prod\limits_{j \in W} \big(1 - X\alpha^i \beta^j\big) \Big)} \mod X^{\delta-1}, \end{aligned}$$

where $A(X)$ has degree at most $d_b \cdot (\zeta - 1) + d_b - 1 = d_b \cdot \zeta - 1$. ■

Similar to the erasure-locator polynomial, we define an error-locator polynomial as follows:

$$\Lambda(X) \overset{\text{def}}{=} \prod_{i \in E} \Big(\prod_{j \in W} \big(1 - X\alpha^i \beta^j\big) \Big). \tag{6.30}$$

Let $\widetilde{\Omega}(X) \overset{\text{def}}{=} \Omega(X) \cdot \Psi(X) + A(X) \cdot \Lambda(X)$ and with (6.29) and (6.30), we obtain the following Key Equation:

$$\widetilde{S}(X) \equiv \frac{\widetilde{\Omega}(X)}{\Lambda(X)} \mod X^{\delta-1}, \text{ with } \begin{array}{ll} \deg \Lambda(X) & = \varepsilon \cdot d_b \\ \deg \widetilde{\Omega}(X) & \leq (\varepsilon + \zeta) \cdot d_b - 1. \end{array} \tag{6.31}$$

Note that in the erasure-free case $\Omega(X)$ is the error-evaluator polynomial with $\deg \Omega(X) \leq \varepsilon \cdot d_b - 1$. The following lemma is similar to Theorem 3.9 and is stated without proof.

Lemma 6.24 (Solving the Key Equation for Error/Erasure Decoding)
Assume $\zeta < d^*_{\text{I-b}} - 1$ erasures occurred. Let $\widetilde{S}(X)$ with $\deg \widetilde{S}(X) \leq \delta - 2$ as in (6.29) be given. If

$$\varepsilon = |E| \leq \left\lfloor \frac{d^*_{\text{I-b}} - 1 - \zeta}{2} \right\rfloor,$$

then there exists a unique solution of (6.31). Recall that Algorithm 3.1, that calculates u_{i+1}, s_{i+1} and t_{i+1}, such that:

$$u_{i+1} = s_{i+1}a + t_{i+1}b$$

holds in every step for the input a and b. We use Algorithm 3.1 with the input polynomials $X^{\delta-1}$ and $\widetilde{S}(X)$ to determine the error-locator polynomial of (6.31). Furthermore, we have the following stopping rule crit of Algorithm 3.1. We stop, if the remainder polynomial $u_i(X)$ in the i-th step of the EEA, i.e., Algorithm 3.1, fulfills:

$$\deg u_{i-1}(X) \geq \frac{\delta - 1 + \zeta \cdot d_b}{2} \quad \text{and} \quad \deg u_i(X) \leq \frac{\delta - 1 + \zeta \cdot d_b}{2} - 1. \tag{6.32}$$

Then, Algorithm 3.1 returns the error-locator polynomial $\Lambda(X)$ as in (6.30) and the error/erasure-evaluation polynomial $\widetilde{\Omega}(X) = \Omega(X) \cdot \Psi(X) + A(X) \cdot \Lambda(X)$ as in (6.31).

Furthermore, we know from the EEA that for $\varepsilon \leq \lfloor (d^*_{\text{I-b}} - 1 - \zeta)/2 \rfloor$ a unique solution $\Lambda(X)$ exists.

We can use the error-evaluation of Lemma 6.14. Therefore, let the two polynomials $f(X), h(X) \in \mathbb{F}_{q^l}[X]$ be defined as follows:

$$f(X) \overset{\text{def}}{=} \prod_{j \in W} \big(1 - X\beta^j\big), \tag{6.33}$$

$$h(X) \overset{\text{def}}{=} \sum_{j \in W} \big(b_j \prod_{\substack{\ell \in W \\ \ell \neq j}} (1 - X\beta^\ell)\big). \tag{6.34}$$

Due to $\gcd(n_a, n_b) = 1$, we have $\gcd(f(X\alpha^i), f(X\alpha^j)) = 1$, $\forall i \neq j$ and therefore each of the n_a polynomials $f(X\alpha^0)$, $f(X\alpha^1), \dots, f(X\alpha^{n_a - 1})$ can be identified by one root similar to the rational approach presented in Section 6.1. Let $\kappa \in W$. Then, we have $f(\beta^{-\kappa}) = 0$. Furthermore, let n_a distinct roots $\gamma_0, \gamma_1, \dots, \gamma_{n_a - 1}$ be defined as:

$$\gamma_i \overset{\text{def}}{=} \beta^{-\kappa}\alpha^{-i}, \quad i \in [n_a). \tag{6.35}$$

Then, each γ_i is a root of $f(X\alpha^i)$. Note that each polynomial $f(X\alpha^i)$ has $|W| = d_b$ roots, but we need only one of them.

6.2.3 Bound II: Generalized Hartmann–Tzeng Bound Using Cyclic Product Code

In this section, we consider the first generalization of Theorem 6.18, where the bound $d^*_{\text{I-b}}$ was proven.

Theorem 6.25 (Bound II: Generalized Hartmann–Tzeng Bound)
Let $\mathcal{A}$ be an $[n_a, k_a, d_a]_q$ cyclic code and $\mathcal{B}$ an $[n_b, k_b, d_b]_q$ with $\gcd(n_a, n_b) = 1$. Let α be an element of order n_a in $\mathbb{F}_{q^{l_a}}$, β of order n_b in $\mathbb{F}_{q^{l_b}}$, respectively. Let six integers $f_1, f_2, m_1, m_2, \delta, \nu$ with $m_1 \neq 0$, $m_2 \neq 0$, $\gcd(n_a, m_1) = \gcd(n_b, m_2) = 1$, $\delta \geq 2$ and $\nu \geq 1$ be given, such that:

$$\sum_{i=0}^{\infty} a(\alpha^{f_1+im_1+j}) \cdot b(\beta^{f_2+im_2+j}) X^i \equiv 0 \mod X^{\delta-1}, \quad \forall j \in [\nu+1) \tag{6.36}$$

holds for all codewords $a(X) \in \mathcal{A}$ and $b(X) \in \mathcal{B}$. Then, the minimum distance d_a of $\mathcal{A}$ is lower bounded by:

$$d_a \geq d^*_{\text{II}} \overset{\text{def}}{=} \left\lceil \frac{\delta+\nu}{d_b} \right\rceil. \tag{6.37}$$

Proof From the generator polynomial of the cyclic product code $\mathcal{A} \otimes \mathcal{B}$ (see Theorem 2.22) we know that whenever $a(X) \in \mathcal{A}$ or $b(X) \in \mathcal{B}$ have a zero, then a codeword of the cyclic product code $\mathcal{A} \otimes \mathcal{B}$ is also zero at the evaluated point (as stated in Lemma 6.19). Therefore, $\delta + \nu$ is the Hartmann–Tzeng bound (see Theorem 2.18) of $\mathcal{A} \otimes \mathcal{B}$ and therefore $d_a d_b \geq \delta + \nu$. ■

6.2.4 Bound III: Using a Second Cyclic Code

In this section, we consider the second generalization of Theorem 6.18, where the bound $d^*_{\text{I-b}}$ was proposed. The proof of the statement is more involved.

Theorem 6.26 (Bound III)
Let $\mathcal{A}$ be an $[n_a, k_a, d_a]_q$ cyclic code and $\mathcal{B}$ a second $[n_b, k_b, d_b]_q$ cyclic code with $\gcd(n_a, n_b) = 1$. Let α be an element of order n_a in $\mathbb{F}_{q^{l_a}}$, β of order n_b in $\mathbb{F}_{q^{l_b}}$ respectively. Let six integers f_1, f_2, m_1, m_2, δ, ν with $m_1 \neq 0$, $m_2 \neq 0$, $\gcd(n_a, m_1) = \gcd(n_b, m_2) = 1$, $\delta \geq 2$ and $\nu \geq 1$ be given, such that:

$$\sum_{i=0}^{\infty} a(\alpha^{f_1+im_1+j}) \cdot b(\beta^{f_2+im_2}) X^i \equiv 0 \mod X^{\delta-1}, \quad \forall j \in [\nu+1) \tag{6.38}$$

holds for for all codewords $a(X) \in \mathcal{A}$ and $b(X) \in \mathcal{B}$.

Then, the minimum distance d_a of $\mathcal{A}$ is lower bounded by:

$$d_a \geq d^*_{\text{III}} \overset{\text{def}}{=} \left\lceil \frac{\delta}{d_b} + \nu \right\rceil. \tag{6.39}$$

Proof Let

$$a(X) = \sum_{i \in Y} a_i X^i \quad \text{with} \quad Y = \{i_0, i_1, \dots, i_{y-1}\} \quad \text{and}$$
$$b(X) = \sum_{i \in Z} b_i X^i \quad \text{with} \quad Z = \{j_0, j_1, \dots, j_{z-1}\}.$$

We combine the $\nu + 1$ equations of (6.38), i.e., multiplying each of it by $\lambda_i \in \mathbb{F}_{q^{l_a}}$. This is similar to the explicit proof of the Hartmann–Tzeng bound of Theorem 2.18. We obtain:

$$\begin{aligned}
\sum_{i=0}^{\infty} \Bigg(& \lambda_0 \sum_{\ell \in Z} b_\ell \beta^{\ell(f_2+im_2)} (a_{i_1}\alpha^{i_1(f_1+im_1)} + \cdots + a_{i_y}\alpha^{i_y(f_1+im_1)}) + \\
& \lambda_1 \sum_{\ell \in Z} b_\ell \beta^{\ell(f_2+im_2)} (a_{i_1}\alpha^{i_1(f_1+im_1+1)} + \cdots + a_{i_y}\alpha^{i_y(f_1+im_1+1)}) + \cdots + \\
& \lambda_\nu \sum_{\ell \in Z} b_\ell \beta^{\ell(f_2+im_2)} (a_{i_1}\alpha^{i_1(f_1+im_1+\nu)} + \cdots + a_{i_y}\alpha^{i_y(f_1+im_1+\nu)}) \Bigg) X^i \\
& \equiv 0 \mod X^{\delta-1}.
\end{aligned}$$

Simplified, this results in:

$$\begin{aligned}
\sum_{i=0}^{\infty} b(\beta^{f_2+im_2}) \Bigg(\sum_{\ell \in Y} a_\ell \alpha^{\ell(f_1+im_1)} (\lambda_0 + \alpha^\ell \lambda_1 + \ldots + \alpha^{\ell\nu}\lambda_\nu) \Bigg) X^i & \\
\equiv 0 \mod X^{\delta-1}. & \qquad (6.40)
\end{aligned}$$

We want to annihilate the first ν terms and guarantee that the linear combination is nonzero. The corresponding heterogeneous system of $\nu + 1$ equations is:

$$\begin{pmatrix} 1 & \alpha^{i_0} & \alpha^{i_0 2} & \cdots & \alpha^{i_0 \nu} \\ 1 & \alpha^{i_1} & \alpha^{i_1 2} & \cdots & \alpha^{i_1 \nu} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{i_\nu} & \alpha^{i_\nu 2} & \cdots & \alpha^{i_\nu \nu} \end{pmatrix} \begin{pmatrix} \lambda_0 \\ \lambda_1 \\ \vdots \\ \lambda_\nu \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \qquad (6.41)$$

and has a unique nonzero solution due to the full rank of the square Vandermonde matrix of order $\nu + 1$ generated by the distinct elements $\alpha^{i_0}, \alpha^{i_1}, \ldots, \alpha^{i_\nu}$.

Let $\widetilde{Y} \overset{\text{def}}{=} Y \setminus \{i_0, i_1, \ldots, i_{\nu-1}\}$ and (6.40) leads to:

$$\sum_{i=0}^{\infty} b(\beta^{f_2+im_2}) \Big(\sum_{\ell \in \widetilde{Y}} a_i \alpha^{\ell(f_1+im_1)} (\lambda_0 + \alpha^\ell \lambda_1 + \cdots + \alpha^{\ell\nu}\lambda_\nu) \Big) X^i \equiv 0 \mod X^{\delta-1}.$$

This leads to (for the ease of notation, we let $m_1 = m_2 = 1$):

$$\frac{\sum\limits_{i \in \widetilde{Y}} \Big(a_i \alpha^{if_1} \sum\limits_{j \in Z} \Big(b_j \beta^{jf_2} \prod\limits_{\substack{\ell \in Z \\ \ell \neq j}} (1 - X\alpha^i \beta^\ell) \Big) \prod\limits_{\substack{h \in \widetilde{Y} \\ h \neq i}} \prod\limits_{p \in Z} (1 - X\alpha^h \beta^p) \Big)}{\prod\limits_{i \in \widetilde{Y}} \big(\prod\limits_{j \in Z} (1 - X\alpha^i \beta^j) \big)} \equiv 0 \mod X^{\delta-1},$$

where the numerator is a nonzero linear combination of the polynomials $\prod_{(h,l)\neq(i,j)}(1 - X\alpha^h\beta^l)$. It is easily shown that all of those polynomials are distinct and linearly independent if and only if $\gcd(n_a, n_b) = \gcd(n_a, m_1) = \gcd(n_b, m_2) = 1$. Hence, the numerator is a nonzero polynomial and its degree is smaller than or equal to $z-1+z(y-\nu)-1=-1+zy-\nu z=z(y-\nu)-1$ and with $d_a \geq y$ and $d_b \geq z$, we obtain:

$$d_b \cdot (d_a-\nu)-1 \geq \delta - 1$$
$$d_a \geq \left\lceil \frac{\delta}{d_b}+\nu \right\rceil . \qquad \blacksquare$$

6.2.5 Decoding up to Bound II

Let $r(X)=a(X)+e(X)$ be the received polynomial, where $e(X)=\sum_{i\in E} e_i X^i \in \mathbb{F}_{q^{l_a}}[X]$ is the error word and $E=\{j_0,j_1,\dots,j_{\varepsilon-1}\} \subseteq \{0,\dots,n_a-1\}$ is the set of error positions of cardinality $|E|=\varepsilon$ and $a(X)$ is a codeword of a given $[n_a,k_a,d_a]_q$ code $\mathcal{A}$.

We describe how to decode up to the generalized bound from Theorem 6.25. Therefore, we want to decode $\varepsilon \leq \tau$ errors, where

$$\tau \leq \frac{d^*_{\text{II}}-1}{2}=\frac{\delta+\nu-1}{2d_b}. \tag{6.42}$$

Let $b(X)\in\mathcal{B}$ be of weight d_b and $\alpha \in \mathbb{F}_{q^{l_a}}, \beta \in \mathbb{F}_{q^{l_b}}$ and the integers $f_1,f_2,m_1\neq 0, m_2 \neq 0$ be given such that Theorem 6.25 for δ and ν holds. Denote $l=\operatorname{lcm}(l_a,l_b)$. We define $\nu+1$ syndrome polynomials:

$$\begin{aligned} S_j(X) &\overset{\text{def}}{\equiv} \sum_{i=0}^{\infty} r(\alpha^{f_1+im_1+j})\cdot b(\beta^{f_2+im_2+j})X^i \mod X^{\delta-1} \\ &= \sum_{i=0}^{\delta-2} e(\alpha^{f_1+im_1+j})\cdot b(\beta^{f_2+im_2+j})X^i, \quad \forall j \in [\nu+1). \end{aligned} \tag{6.43}$$

This generalizes our previous approach of Section 6.2 to $\nu+1$ syndrome sequences of length $\delta-1$. Hence, we obtain $\nu+1$ Key Equations with a common error-locator polynomial $\Lambda(X)\in\mathbb{F}_{q^l}[X]$ of degree $d_b\varepsilon$ (compare also [A-ZWB12b, Equation (20)]):

$$\Lambda(X)\cdot S_j(X) \equiv \Omega_j(X) \mod X^{\delta-1}, \quad j \in [\nu+1),$$

where the degree of all $\Omega_0(X),\Omega_1(X),\dots,\Omega_\nu(X)$ is less than $d_b\varepsilon$. This is similar to the collaborative decoding of Interleaved GRS codes as discussed in Section 3.4.

The syndrome calculation results in $\nu+1$ syndrome sequences of the same length $\delta-1$ to determine one common $\Lambda(X)\in\mathbb{F}_{q^l}[X]$. Solving these $\nu+1$ Key Equations jointly is a multi-sequence shift-register synthesis problem for sequences of equal length; for efficient algorithms see e.g., [A-FT89; A-FT91a; A-SS11; A-ZW11].

The basic task is to solve the following heterogeneous linear system of equations for $\Lambda(X)=\Lambda_0+\Lambda_1X+\dots+\Lambda_{d_b\varepsilon}X^{d_b\varepsilon}$, which we normalize such that $\Lambda_0=1$:

$$\begin{pmatrix} \mathbf{S}^{\langle 0\rangle} \\ \mathbf{S}^{\langle 1\rangle} \\ \vdots \\ \mathbf{S}^{\langle \nu\rangle} \end{pmatrix} \cdot \begin{pmatrix} \Lambda_{d_b\varepsilon} \\ \vdots \\ \Lambda_2 \\ \Lambda_1 \end{pmatrix} = \begin{pmatrix} \mathbf{T}^{\langle 0\rangle} \\ \mathbf{T}^{\langle 1\rangle} \\ \vdots \\ \mathbf{T}^{\langle \nu\rangle} \end{pmatrix}, \tag{6.44}$$

where each sub-matrix $\mathbf{S}^{\langle j \rangle}$ is a $(\delta - 1 - d_b\varepsilon) \times (d_b\varepsilon)$ matrix and $\mathbf{T}^{\langle j \rangle}$ is a column vector of length $\delta - 1 - d_b t$ as follows:

$$\mathbf{S}^{\langle j \rangle} = \begin{pmatrix} S_0^{\langle j \rangle} & S_1^{\langle j \rangle} & \dots & S_{d_b\varepsilon-1}^{\langle j \rangle} \\ S_1^{\langle j \rangle} & S_2^{\langle j \rangle} & \dots & S_{d_b\varepsilon}^{\langle j \rangle} \\ \vdots & \vdots & \ddots & \vdots \\ S_{\delta-2-d_b\varepsilon}^{\langle j \rangle} & S_{\delta-1-d_b\varepsilon}^{\langle j \rangle} & \dots & S_{\delta-3}^{\langle j \rangle} \end{pmatrix} \tag{6.45}$$

and $\mathbf{T}^{\langle j \rangle} = (S_{d_b\varepsilon}^{\langle j \rangle} \, S_{d_b\varepsilon+1}^{\langle j \rangle} \, \dots \, S_{\delta-2}^{\langle j \rangle})^T$. In the following, denote

$$\mathbf{S} \overset{\text{def}}{=} (\mathbf{S}^{\langle 0 \rangle,T} \, \mathbf{S}^{\langle 1 \rangle,T} \, \dots \, \mathbf{S}^{\langle \nu \rangle,T})^T.$$

We consider again the heterogeneous system. In order to guarantee unique decoding, we have to prove that the syndrome matrix $\mathbf{S}$ from (6.44) has full rank if (6.42) is fulfilled. For simplicity, we consider only $d_b = 2$, where $\mathcal{B}$ is a single parity check code. In the following, we analyze the rank of this syndrome matrix if the condition on the decoding radius (6.42) is fulfilled.

> **Theorem 6.27 (Decoding up to Bound II for $d_b = 2$)**
> Let $\mathcal{B}$ be an $[n_b, n_b - 1, 2]_q$ single parity check code with $d_b = 2$ and let $\gcd(n_a, n_b) = \gcd(n_a, m_1) = \gcd(n_b, m_2) = 1$ hold. Moreover, let (6.42) be fulfilled and let $\nu + 1$ syndrome sequences of length $\delta - 1$ be defined as in (6.43). Then, the syndrome matrix $\mathbf{S}$ with the sub-matrices from (6.45) has $\operatorname{rank}(\mathbf{S}) = 2\varepsilon$.

Proof Let us w.l.o.g. assume that $b(X) = 1 + X$ and $f_1 = f_2 = 0$. Then, the $\nu + 1$ syndrome polynomials in $\mathbb{F}_{q^l}[X]$ are

$$S_j(X) = \sum_{i=0}^{\delta-2} e(\alpha^{im_1+j})(1 + \beta^{im_2+j})X^i, \quad \forall j \in [\nu + 1).$$

Similar to [A-FT91a, Section VI], we can decompose the syndrome matrix into three matrices as follows.

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}^{\langle 0 \rangle} \\ \mathbf{S}^{\langle 1 \rangle} \\ \vdots \\ \mathbf{S}^{\langle \nu \rangle} \end{pmatrix} = \mathbf{X} \cdot \mathbf{Y} \cdot \overline{\mathbf{X}} = \begin{pmatrix} \mathbf{X}^{\langle 0 \rangle} \\ \mathbf{X}^{\langle 1 \rangle} \\ \vdots \\ \mathbf{X}^{\langle \nu \rangle} \end{pmatrix} \cdot \mathbf{Y} \cdot \overline{\mathbf{X}},$$

where $\mathbf{X}$ is a $(\nu + 1)(\delta - 1 - 2\varepsilon) \times 2\varepsilon$ matrix over $\mathbb{F}_{q^l}$ and $\mathbf{Y}$ and $\overline{\mathbf{X}}$ are $2\varepsilon \times 2\varepsilon$ matrices over $\mathbb{F}_q$

and $\mathbb{F}_{q^l}$, respectively. The decomposition provides the following matrices with $\kappa = \delta - 2 - 2\varepsilon$:

$$\mathbf{X}^{\langle j \rangle} =$$
$$\begin{pmatrix}
\alpha^{j_0(j)} & \alpha^{j_1(j)} & \dots & \alpha^{j_{\varepsilon-1}(j)} \\
\alpha^{j_0(j+m_1)} & \alpha^{j_1(j+m_1)} & \dots & \alpha^{j_{\varepsilon-1}(j+m_1)} \\
\vdots & \vdots & \ddots & \vdots \\
\alpha^{j_0(j+m_1\kappa)} & \alpha^{j_1(j+m_1\kappa)} & \dots & \alpha^{j_{\varepsilon-1}(j+m_1\kappa)} \\
\beta^{j}\alpha^{j_0(j)} & \beta^{j}\alpha^{j_1(j)} & \dots & \beta^{j}\alpha^{j_{\varepsilon-1}(j)} \\
\beta^{j+m_2}\alpha^{j_0(j+m_1)} & \beta^{j+m_2}\alpha^{j_1(j+m_1)} & \dots & \beta^{j+m_2}\alpha^{j_{\varepsilon-1}(j+m_1)} \\
\vdots & \vdots & \ddots & \vdots \\
\beta^{j+m_2\kappa\varepsilon}\alpha^{j_0(j+m_1\kappa)} & \beta^{j+m_2\kappa\varepsilon}\alpha^{j_1(j+m_1\kappa)} & \dots & \beta^{j+m_2\kappa\varepsilon}\alpha^{j_{\varepsilon-1}(j+m_1\kappa)}
\end{pmatrix},$$

and $\mathbf{Y} = \operatorname{diag}(e_{j_0}, e_{j_1}, \dots, e_{j_{\varepsilon-1}}, e_{j_0}, e_{j_1}, \dots, e_{j_{\varepsilon-1}})$ and

$$\overline{\mathbf{X}} = \begin{pmatrix}
1 & \alpha^{j_0 m_1} & \dots & \alpha^{j_0 m_1(2\varepsilon-1)} \\
1 & \alpha^{j_1 m_1} & \dots & \alpha^{j_1 m_1(2\varepsilon-1)} \\
\vdots & \vdots & \ddots & \vdots \\
1 & \alpha^{j_{\varepsilon-1} m_1} & \dots & \alpha^{j_{\varepsilon-1} m_1(2\varepsilon-1)} \\
1 & \beta^{m_2}\alpha^{j_0 m_1} & \dots & (\beta^{m_2}\alpha^{j_0 m_1})^{(2\varepsilon-1)} \\
1 & \beta^{m_2}\alpha^{j_1 m_1} & \dots & (\beta^{m_2}\alpha^{j_1 m_1})^{(2\varepsilon-1)} \\
\vdots & \vdots & \ddots & \vdots \\
1 & \beta^{m_2}\alpha^{j_{\varepsilon-1} m_1} & \dots & (\beta^{m_2}\alpha^{j_{\varepsilon-1} m_1})^{(2\varepsilon-1)}
\end{pmatrix}.$$

Since $\mathbf{Y}$ is a diagonal matrix, it is non-singular. From

$$\gcd(n_a, n_b) = \gcd(n_a, m_1) = \gcd(n_b, m_2) = 1$$

we know that $\overline{\mathbf{X}}$ is a Vandermonde matrix and has full rank. Hence, $\mathbf{Y} \cdot \overline{\mathbf{X}}$ is a non-singular $2\varepsilon \times 2\varepsilon$ matrix and therefore $\operatorname{rank}(\mathbf{S}) = \operatorname{rank}(\mathbf{X})$. In order to analyze the rank of $\mathbf{X}$, we proceed similar as in [A-FT91a, Section VI] and use the following corollary, which follows directly from [A-LW86, Theorem 4].

Corollary 6.28 (LW-Matrix Product and Rank)
Let the following matrix operation be defined as in [A-LW86]:

$$\mathbf{X} = \mathbf{A} * \mathbf{B} = \begin{pmatrix}
a_{0,0}\mathbf{b}_0 & a_{0,1}\mathbf{b}_1 & \dots & a_{0,2\varepsilon-1}\mathbf{b}_{2\varepsilon-1} \\
a_{1,0}\mathbf{b}_0 & a_{1,2}\mathbf{b}_1 & \dots & a_{1,2\varepsilon-1}\mathbf{b}_{2\varepsilon-1} \\
\vdots & \vdots & \ddots & \vdots \\
a_{\nu,0}\mathbf{b}_0 & a_{\nu,2}\mathbf{b}_1 & \dots & a_{\nu,2\varepsilon-1}\mathbf{b}_{2\varepsilon-1}
\end{pmatrix},$$

where $\mathbf{A}$ is a $(\nu+1) \times 2\varepsilon$ matrix, $\mathbf{B}$ is a $(\delta - 1 - 2\varepsilon) \times 2\varepsilon$ matrix and $\mathbf{b}_i$ denotes the ith column of $\mathbf{B}$, and $\mathbf{X}$ has 2ε columns. If $\operatorname{rank}(\mathbf{A}) + \operatorname{rank}(\mathbf{B}) > 2\varepsilon$, then $\operatorname{rank}(\mathbf{X}) = 2\varepsilon$.

We use the matrix operation from Corollary 6.28 to rewrite $\mathbf{X} = \mathbf{A} * \mathbf{B}$, where

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & \dots & 1 & 1 & 1 & \dots & 1 \\ \alpha^{j_0} & \alpha^{j_1} & \dots & \alpha^{j_{\varepsilon-1}} & \beta\alpha^{j_0} & \beta\alpha^{j_1} & \dots & \beta\alpha^{j_{\varepsilon-0}} \\ \alpha^{j_0 2} & \alpha^{j_1 2} & \dots & \alpha^{j_{\varepsilon-1} 2} & (\beta\alpha^{j_0})^2 & (\beta\alpha^{j_1})^2 & \dots & (\beta\alpha^{j_{\varepsilon-1}})^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \alpha^{j_0 \nu} & \alpha^{j_1 \nu} & \dots & \alpha^{j_{\varepsilon-1} \nu} & (\beta\alpha^{j_0})^\nu & (\beta\alpha^{j_1})^\nu & \dots & (\beta\alpha^{j_{\varepsilon-1}})^\nu \end{pmatrix},$$

and $\mathbf{B} = \mathbf{X}^{\langle 0 \rangle}$.

Since $\gcd(n_a, n_b) = \gcd(n_a, m_1) = \gcd(n_b, m_2) = 1$, both matrices $\mathbf{A}$ and $\mathbf{B}$ are Vandermonde matrices of ranks:

$$\operatorname{rank}(\mathbf{A}) = \min\{\nu + 1, 2t\}, \qquad \operatorname{rank}(\mathbf{B}) = \min\{\delta - 1 - 2t, 2t\}.$$

Note that w.l.o.g. we can always define m_1, m_2, δ and ν such that $\nu + 1 \leq \delta - 1$. Therefore, from (6.42) we obtain:

$$t \leq \frac{d^*_{\mathrm{II}} - 1}{2} = \frac{\delta + \nu - 1}{2d_b} \leq \frac{2(\delta - 1) - 1}{2d_b} < \frac{\delta - 1}{d_b}. \tag{6.46}$$

Hence, investigating all possible four cases of $\operatorname{rank}(\mathbf{A}) + \operatorname{rank}(\mathbf{B})$ gives:

$$\begin{aligned} 2t + 2t = 4t &> 2t, \\ 2t + \nu + 1 &> 2t, \\ \delta - 1 - 2t + 2t = \delta - 1 &> 2t, \\ \delta - 1 - 2t + \nu + 1 \geq 2d_b t - 2t + 1 = 2t + 1 &> 2t, \end{aligned}$$

where the last two above inequalities used (6.46) and $d_b = 2$. Thus, $\operatorname{rank}(\mathbf{A}) + \operatorname{rank}(\mathbf{B}) > 2t$. With Corollary 6.28, we proved the statement. ■

Therefore, the Key Equation (6.44) has a unique solution, which can be found by any multi-sequence shift-register synthesis algorithm with $\mathcal{O}(sn^2)$ operations over $\mathbb{F}_{q^l}$ [A-FT89; A-FT91a; A-ZW11]. The extension of the proof for decoding up to $\varepsilon \leq \tau$ errors as in (6.42) to other associated codes $\mathcal{B}$ with $d_b \geq 2$ is straight-forward. The decomposition of the syndrome matrices can be done similarly and we can prove that the syndrome matrix $\mathbf{S}$ has rank $d_b \varepsilon$.

6.3 Lowest-Code-Rate Binary Cyclic Codes with Minimum Distance Two and Three

6.3.1 Motivation and Previous Work

To obtain a huge family of cyclic codes for the bounds I-b (Theorem 6.18), II (Theorem 6.25), III (Theorem 6.26), the cardinality of the required subset of their defining set should be small. This implies a high cardinality of the defining set $D_{\mathcal{B}}$ of the associated second code $\mathcal{B}$. On the one hand, we need a low-rate k_b/n_b which implies a high $|D_{\mathcal{B}}|$. On the other hand, the minimum distance d_b should be small to obtain a good bound for all three cases I-b, II and III. This motivates the investigation of small-minimum-distance cyclic codes with lowest code-rate.

Primitive binary cyclic codes with minimum distance three were investigated by Charpin, Tietäväinen and Zinoviev in [A-CTZ97; A-CTZ99]. We generalize the results of [A-CTZ97] to binary cyclic codes of arbitrary length and show afterwards the implications, when we want to use them to bound the minimum distance of a given cyclic code as in Theorem 6.18. We derive necessary and sufficient conditions for binary non-primitive cyclic codes with minimum distance two and a sufficient condition for minimum distance three. For the case of minimum distance two, we show the defining set of codes of lowest code-rate. Parts of these result were published in [A-ZB12a, Section 5].

Lemma 6.29 (Primitive Binary Cyclic Codes with $d = 2$ [A-CTZ97, Lemma 1])
Let i, j with $0 \leq i < j \leq n-1$ be two arbitrary integers that do not belong to the same cyclotomic coset modulo n. Then the binary $[n,k]_2$ cyclic code $\mathcal{C}$ with generator polynomial

$$g(X) = M_{i,2}^{\langle n \rangle}(X) \cdot M_{j,2}^{\langle n \rangle}(X)$$

has minimum distance two if and only if $\gcd(n, i, j) > 1$.

Proof Let α be an n-th root of unity. A binary cyclic code $\mathcal{C}$ with generator polynomial

$$g(X) = M_{i,2}^{\langle n \rangle}(X) \cdot M_{j,2}^{\langle n \rangle}(X)$$

of length n has minimum distance two if there exist a binomial $c(X) = X^k + X^\ell$ that fulfills

$$c(\alpha^i) = c(\alpha^j) = 0.$$

This holds if and only if

$$\alpha^{ki} = \alpha^{\ell i} \quad \text{and} \quad \alpha^{kj} = \alpha^{\ell j}$$

or, equivalently,

$$(k-\ell)i \equiv (k-\ell)j \equiv 0 \mod n.$$

Both congruences are valid if and only if $n/\gcd(n,i,j)$ divides $k-\ell$. Therefore, such k and ℓ exist if and only if $\gcd(n,i,j) > 1$. ■

Theorem 6.30 (Primitive Binary Cyclic Codes with $d = 2$ [A-CTZ97])
Let $i_1, i_2, \dots, i_s$ with $0 \leq i_1 < \dots < i_s \leq n-1$ be s arbitrary integers that do not belong to the same cyclotomic coset modulo n. Then the binary $[n,k]_2$ cyclic code $\mathcal{C}$ with generator polynomial

$$g(X) = \prod_{j=1}^{s} M_{i_j,q}^{\langle n \rangle}(X)$$

has minimum distance two if and only if $\gcd(n, i_1, i_2, \dots, i_s) > 1$.

We skip the proof of Theorem 6.30, because it is straightforward to the proof of Lemma 6.29.

The following lemma is a generalization of [A-CTZ97, Theorem 1] to binary cyclic codes of arbitrary length with minimum distance three.

Lemma 6.31 (Binary Cyclic Codes with $d = 3$)
Let i, j with $0 \leq i < j \leq n-1$ be arbitrary integers that do not belong to the same cyclotomic coset modulo n. Let g be such that $2^g - 1$ divides n. If there exists an integer r with $0 < r < 2^g - 1$,

where $\gcd(r, 2^g - 1) = 1$, such that both i and j are in $M_{r,2}^{\langle 2^g-1\rangle}$, then the binary $[n, k]_2$ cyclic code $\mathcal{C}$ with generator polynomial

$$g(X) = M_{i,2}^{\langle n\rangle}(X) \cdot M_{j,2}^{\langle n\rangle}(X)$$

has minimum distance $d \leq 3$. If, moreover, $\gcd(n, i, j) = 1$, then $d = 3$.

Proof Let γ be a primitive element of $\mathbb{F}_{2^s}$, let $z = (2^s - 1)/n$ and let $\alpha = \gamma^z$. Let $u = n/(2^g - 1)$, then $\beta = \alpha^u = \gamma^{(2^s-1)/(2^g-1)}$, is a primitive element of $\mathbb{F}_{2^g}$. Let b be an integer in the interval $[1, 2^g - 2]$ such that:

$$1 + \beta + \beta^b = 0.$$

Define

$$c(X) = 1 + X^{u(1/r)} + X^{u(b/r)},$$

where the quotients $1/r$ and b/r are calculated in the ring $\mathbb{Z}_{2^g-1}$ of integers modulo $2^g - 1$. For $i \in M_{r,2}^{\langle 2^g-1\rangle}$, two non-negative integers k and ℓ exist such that

$$i = \ell(2^g - 1) + 2^k r.$$

Thus,

$$\begin{aligned} c(\alpha^i) &= 1 + \alpha^{ui(1/r)} + \alpha^{ui(b/r)} \\ &= 1 + \beta^{i(1/r)} + \beta^{i(b/r)} \\ &= 1 + \beta^{2^k r(1/r)} + \beta^{2^k r(b/r)} \\ &= 1 + \beta^{2^k} + \beta^{b2^k} \\ &= (1 + \beta + \beta^b)^{2^k} \\ &= 0. \end{aligned}$$

Therefore, the code $\mathcal{C}$ has minimum distance $d \leq 3$. If $\gcd(n, i, j) = 1$ as in Lemma 6.29, then the minimum distance is unequal two and therefore three. ■

Note that in [A-CTZ97] the length of the cyclic code was $n = 2^s - 1$ and $u = (2^s - 1)/(2^g - 1)$. Lemma 6.29 and Theorem 6.32 can be generalized to cyclic codes, where the generator polynomial $g(X)$ is a product of several minimal polynomials.

Theorem 6.32 (Binary Cyclic Codes with $d = 3$)
Let $i_1, i_2, \ldots, i_s$ with $0 \leq i_1 < \cdots < i_s \leq n - 1$ be s arbitrary integers that do not belong to the same cyclotomic coset modulo n. Let g be such that $2^g - 1$ divides n. If there exists an integer r with $0 < r < 2^g - 1$, where $\gcd(r, 2^g - 1) = 1$, such that all s integers $i_1, i_2, \ldots, i_s$ are in $M_{r,2}^{\langle 2^g-1\rangle}$, then the binary $[n, k]_2$ cyclic code $\mathcal{C}$ with generator polynomial

$$g(X) = \prod_{j=1}^{s} M_{i_j,2}^{\langle n\rangle}(X)$$

has minimum distance $d \leq 3$. If, moreover, $\gcd(n, i_1, \ldots, i_s) = 1$, then $d = 3$.

We skip the proof of Theorem 6.32, because it is straightforward to the proof of Lemma 6.31. Let us consider an example of a non-primitive binary cyclic code with minimum distance three.

> **Example 6.33 (Non-primitive Binary Cyclic Code with $d = 3$)**
> Let $n = 119 = (2^3 - 1) \cdot 17$. In this case $g = 3$ (see Theorem 6.32). Then $\{1, 11, 51\}$ belong to $M_{1,2}^{\langle 7 \rangle}$ and we have $\gcd(1, 11, 51) = 1$. Therefore, the binary cyclic code of length $n = 119$ with generator polynomial
>
> $$g(X) = M_{1,2}^{\langle 119 \rangle}(X) \cdot M_{11,2}^{\langle 119 \rangle}(X) \cdot M_{51,2}^{\langle 119 \rangle}(X),$$
>
> has dimension $k = 68$ and minimum distance $d = 3$.

6.3.2 Implications for Bounding the Minimum Distance

We consider lowest-code-rate binary cyclic codes of minimum distance two and three. They are good candidates for bounding the minimum distance as discussed in the previous section.

We first consider lowest-code-rate binary cyclic codes of minimum distance two. As previously, the sign $\Box$ marks a non-zero in the defining set.

> **Proposition 6.34 (Lowest-Code-Rate Binary Cyclic Codes with $d = 2$)**
> Let $a > 1$, $g > 1$ and n be three integers, such that $n = ag$. Let g be in the defining set D. Then the binary $[n, k]_2$ cyclic code $\mathcal{C}$ with defining set:
>
> $$D = \{0, \Box, \dots, \Box, g, \Box, \dots, \Box, 2g, \Box, \dots, \Box, (a-1)g, \Box, \dots, \Box\}$$
>
> is the binary cyclic code of smallest dimension $k = a(g-1)$, lowest code-rate $R = (g-1)/g$ and minimum distance two.

Proof We want to maximize $|D|$ while keeping the minimum distance d of $\mathcal{C}$ at two. Therefore, we select for a given g every cyclotomic coset $M_{i,2}^{\langle n \rangle}$ with $\gcd(i, g) > 1$ for all $i \in [n)$ to be in D with aimed minimum distance two. One the one hand, this guarantees the maximization of $|D|$ and therefore the minimization of the code-rate. On the other hand, due to the condition $\gcd(i, g) > 1$ (Theorem 6.30) the minimum distance of $\mathcal{C}$ remains two. ■

The defining set of Proposition 6.34 is equal to the defining set of a cyclic product code $\mathcal{A} \otimes \mathcal{B}$ as in Theorem 2.21, where $\mathcal{A}$ is a $[g, g-1, 2]_2$ cyclic single-parity check code with defining set $\{0\}$ and $\mathcal{B}$ is a trivial $[a, a, 1]_2$ code.

A direct consequence of Proposition 6.34 is that we do not need to investigate these binary cyclic codes of minimum distance two any more. We obtain the same result when we select a $[g, g-1, 2]_2$ single-parity check code as associated code $\mathcal{B}$.

> **Conjecture 6.35 (Lowest-Code-Rate Binary Cyclic Codes with $d = 3$)**
> Let $a > 1$, $g > 1$ and n be three integers, such that $n = a(2^g - 1)$. Let r be an integer with $0 < r < 2^g - 1$, where $\gcd(r, 2^g - 1) = 1$. Let r be in the defining set D. Then the binary cyclic code $\mathcal{C}$ with distance three of length n with defining set:
>
> $$\begin{aligned} D = \{r \cdot i \mod n \mid i = & j(2^g-1)+1, j(2^g-1)+2, j(2^g-1)+4, \dots, \\ & j(2^g-1)+2^{g-1} \quad \forall j \in [a)\} \end{aligned}$$

is the binary cyclic code with the smallest dimension $k = a(2^g - 1 - g)$, lowest code-rate $R = (2^g - 1 - g)/(2^g - 1)$ and minimum distance three.

Similar to the proof of Proposition 6.34, we can reasoning Conjecture 6.35. We want to maximize $|D|$ while keeping d of $\mathcal{C}$ at three. For a given r and for $(2^g - 1)|n$, we select every cyclotomic coset $M_{i,2}^{\langle n \rangle}$ for all $i \in [n)$ to be in the D of $\mathcal{C}$ with aimed minimum distance three, such that $i \in M_{r,2}^{\langle 2^g-1 \rangle}$. One the one hand, this guarantees the maximization of $|D|$ and therefore the minimization of the code-rate. On the other hand, due to the condition that $M_{i,2}^{\langle n \rangle}$ should be selected such that $i \in M_{r,2}^{\langle 2^g-1 \rangle}$ (Theorem 6.32) the minimum distance of $\mathcal{C}$ remains three.

Lemma 6.31 gives only a sufficient and not as Theorem 6.30 for distance two a necessary and sufficient condition. It is an open problem to prove Conjecture 6.35.

Note 6.36 (Connection to Binary Hamming Code)
Let $r = 1$ in Proposition 6.35. Then

$$M_{1,2}^{\langle 2^g-1 \rangle} = \{1, 2, 4, \dots, 2^{g-1}\}$$

is the cyclotomic coset of a binary Hamming code of length $2^g - 1$. The defining set of the corresponding lowest-code-rate binary cyclic code is a “repetition” of the defining set of the Hamming code of length $2^g - 1$.

Example 6.37 (Non-primitive Binary Cyclic Code with $d = 3$ and Lowest Code-Rate)
Let us again consider Example 6.33 with $n = 119 = (2^3 - 1) \cdot 17$ and $k = 68$. The binary cyclic code of length $n = 119$ with generator polynomial

$$g(X) = M_{1,2}^{\langle 119 \rangle}(X) \cdot M_{11,2}^{\langle 119 \rangle}(X) \cdot M_{51,2}^{\langle 119 \rangle}(X)$$

and with minimum distance three has lowest code rate $R = (2^3 - 1 - 3)/(2^3 - 1) = 68/119$. Its defining set D is:

$$D = \{\square, 1, 2, \square, 4, \square, \square, \square, 8, 9, \square, 11, \square, \square, \square, 15, 16, \square, 18, \square, \square, \square, 22, \dots, 116, \square, \square\}.$$

The defining set of Proposition 6.35 is equal to the defining set of a cyclic product code $\mathcal{A} \otimes \mathcal{B}$ as in Theorem 2.21, where $\mathcal{A}$ is for $r = 1$ a $[2^g - 1, 2^g - 1 - g, 3]_2$ cyclic Hamming code with defining set $\{1, 2, \dots, 2^{g-1}\}$ and $\mathcal{B}$ is a trivial $[a, a, 1]_2$ code.

6.4 Cyclic Generalized Product Codes

6.4.1 Related Work and Basic Idea

A linear generalized concatenated code as in Definition 2.26, where the s outer (or row) codes $\mathcal{A}_0$, $\mathcal{A}_1$, $\dots$, $\mathcal{A}_{s-1}$ and the inner (or column) codes $\mathcal{B}_0$, $\mathcal{B}_1$, $\dots$, $\mathcal{B}_{s-1}$ are defined over the same alphabet, is called generalized product code. This class of linear block codes was—independently of Blokh and Zyablov’s work [A-BZ74]—considered before by Marchukov [A-Mar68] and Gore [A-Gor70]. We consider the cyclic variant of generalized product codes and give explicitly the defining set, respectively the generator polynomial, which was not done so far.

In contrast to this, many publications cover the cyclic variant of generalized concatenated codes (see Berlekamp and Jensen [A-BJ74], Jensen [A-Jen85] and de Rooij–van Lint [A-RL91]). It is possible to construct cyclic generalized concatenated codes by a quasi-cyclic outer code and a inner cyclic code (see Jensen [A-Jen92]). After the basic properties of generalized product codes, we outline how cyclic generalized product codes can be used—similar to cyclic product codes—to bound the minimum distance of a cyclic code.

6.4.2 Definition and Defining Set

Let us first determine the generator polynomial of a cyclic code that is the direct sum as in Definition 2.13 of several cyclic codes.

Theorem 6.38 (Generator Polynomial of a Cyclic Direct Sum Code)
Let s $[n, k_i]_q$ cyclic codes $\mathcal{C}_i$ for all $i \in [s)$ with $\sum_{i=0}^{s-1} k_i < n$ and with generator polynomials $g_i(X) \in \mathbb{F}_q[X], \forall i \in [s)$ be given. Then, the polynomial:

$$g(X) = \gcd\Big(g_0(X), g_1(X), \dots, g_{s-1}(X)\Big) \tag{6.47}$$

is the generator polynomial of the cyclic direct sum code $\bigoplus_{i=0}^{s-1} \mathcal{C}_i$.

Proof For every $c(X) \in \bigoplus_{i=0}^{s-1} \mathcal{C}_i$, the greatest common divisor $\gcd\big(g_0(X), g_1(X), \dots, g_{s-1}(X)\big)$ divides $c(X)$. For the converse, we know that the EEA returns a relation, such that:

$$s_0(X)g_0(X) + s_1(X)g_1(X) + \dots + s_{s-1}(X)g_{s-1}(X) = \gcd\big(g_0(X), g_1(X), \dots, g_{s-1}(X)\big)$$

and therefore $\gcd\big(g_0(X), g_1(X), \dots, g_{s-1}(X)\big) \in \bigoplus_{i=0}^{s-1} \mathcal{C}_i$ and thus is the generator polynomial of the cyclic direct sum code $\bigoplus_{i=0}^{s-1} \mathcal{C}_i$. ■

Corollary 6.39 (Defining Set of a Cyclic Direct Sum Code)
Let s $[n, k_i]_q$ cyclic codes $\mathcal{C}_i$ for all $i \in [s)$ with $\sum_{i=0}^{s-1} k_i < n$ and with defining sets $D_i, \forall i \in [s)$ be given. The defining set of $\mathcal{C} = \bigoplus_{i=0}^{s-1} \mathcal{C}_i$ is:

$$D_{\mathcal{C}} = \bigcap_{i=0}^{s-1} D_i.$$

The following lemma is essential for the construction of cyclic generalized product codes and is the cyclic pendant to Corollary 2.15.

Lemma 6.40 (Generator Polynomials for a Partition Chain)
Let s $[n, k_i]_q$ cyclic codes $\mathcal{C}_i$ for all $i \in [s)$ with $\sum_{i=0}^{s-1} k_i < n$ and with generator polynomial $g_i(X) \in \mathbb{F}_q[X], \forall i \in [s)$ be given. Then,

$$\mathcal{C}_0 \supset \mathcal{C}_1 \supset \dots \supset \mathcal{C}_{s-1} \tag{6.48}$$

holds if and only if

$$g_i(X) \mid g_{i+1}(X), \quad \forall i \in [s-1).$$

Bounding Distance

Proof Then, all codewords of $\mathcal{C}_{i+1}$, which are a multiple of $g_{i+1}(X)$, are then a multiple of $g_i(X)$ and therefore codewords of $\mathcal{C}_i$. For the converse, the subset of the codewords of $\mathcal{C}_i$ which are multiples of $g_{i+1}(X)$ are codewords of $\mathcal{C}_{i+1}$. ■

A generalized product code is a generalized concatenated code as in Definition 2.26, where the outer (or row) code is over the same alphabet as the inner (or column) code. Let us prove the equality for the minimum distance in the following.

Lemma 6.41 (Distance of Generalized Product Code)
Let s outer (or inner) $[n_a, k_{a,i}, d_{a,i}]_q$ codes $\mathcal{A}_i$ for all $i \in [s)$ with $\sum_{i=0}^{s-1} k_i < n$ be given. Let $\mathcal{B}_i$ denote $[n_b, k_{b,i}, d_{b,i}]_q$ codes for all $i \in [s)$. Furthermore, let

$$\mathcal{B}_0 \supset \mathcal{B}_1 \supset \cdots \supset \mathcal{B}_{s-1}$$

as in Corollary 2.15 hold. Let

$$k = \sum_{i=0}^{s-2} \big(k_{a,i}(k_{b,i} - k_{b,i+1})\big) + k_{a,s-1}k_{b,s-1}.$$

Then, the $[n_a n_b, k, d]_q$ generalized product code $\left(\bigoplus_{i=0}^{s-2} (\mathcal{A}_i \otimes \mathcal{B}_i \backslash \mathcal{B}_{i+1})\right) \oplus (\mathcal{A}_{s-1} \otimes \mathcal{B}_{s-1})$ has minimum distance

$$d = \min_{i \in [s)} \big(d_{a,i} \cdot d_{b,i}\big).$$

Proof Similar to the proof of Theorem 2.27. A codeword $\mathbf{a}_i$ of $\mathcal{A}_i$ with minimal Hamming weight $d_{a,i}$ affects a sub-code $\mathcal{B}_{i+1}$ of $\mathcal{B}_i$ having at least weight $d_{b,i}$. For each sub-product code such a codeword exist and therefore the equality holds. ■

In contrast to generalized concatenated codes as in Definition 2.26, the minimum distance of generalized product codes equals the minimum of the product of all minimum distance of the sub-codes. This is similar to the fact, that the distance of a product code equals the product of the minimum distances of its sub-codes, whereas the minimum distance of a generalized concatenated code as in Definition 2.24 can be greater than the product of the minimum distances of the sub-codes.

Let us refine the conditions such that the generalized product code is cyclic.

Definition 6.42 (Cyclic Generalized Product Code)
Let s outer (or row) $[n_a, k_{a,i}, d_{a,i}]_q$ cyclic codes $\mathcal{A}_i$ for all $i \in [s)$ with defining sets $D_{\mathcal{A}_i}$ be given. Let $\mathcal{B}_i$ denote $[n_b, k_{b,i}, d_{b,i}]_q$ cyclic codes with defining set $D_{\mathcal{B}_i}$ for all $i \in [s)$. Let

$$\mathcal{B}_0 \supset \mathcal{B}_1 \supset \cdots \supset \mathcal{B}_{s-1}.$$

Furthermore, let $un_a + vn_b = 1$ for some integers u and v.

Define the sets:

$$\begin{aligned} B_{\mathcal{A}_i} &\overset{\text{def}}{=} (D_{\mathcal{A}_i} \cdot v)_{n_a}, \quad \forall i \in [s), \\ A_{\mathcal{B}_i \backslash \mathcal{B}_{i+1}} &\overset{\text{def}}{=} \Big((D_{\mathcal{B}_i} \backslash D_{\mathcal{B}_{i+1}}) \cdot u\Big)_{n_b}, \quad \forall i \in [s-1) \\ A_{\mathcal{B}_{s-1}} &\overset{\text{def}}{=} (D_{\mathcal{B}_{s-1}} \cdot u)_{n_b}, \end{aligned}$$

where the operations on the set are as defined in (2.14), (2.15) and (2.16).

The defining set of the i-th cyclic product sub-code $\mathcal{A}_i \otimes (\mathcal{B}_i \backslash \mathcal{B}_{i+1})$ is:

$$D_{\mathcal{A}_i \otimes (\mathcal{B}_i \backslash \mathcal{B}_{i+1})} = \left\{ \bigcup_{j=0}^{n_b-1} (B_{\mathcal{A}_i} + jn_a) \right\} \cup \left\{ \bigcup_{j=0}^{n_a-1} (A_{\mathcal{B}_i \backslash \mathcal{B}_{i+1}} + jn_b) \right\}, \quad \forall i \in [s-1).$$

The defining set of the s-th cyclic product sub-code $\mathcal{A}_{s-1} \otimes \mathcal{B}_{s-1}$ is:

$$D_{\mathcal{A}_{s-1} \otimes \mathcal{B}_{s-1}} = \left\{ \bigcup_{j=0}^{n_b-1} (B_{\mathcal{A}_{s-1}} + jn_a) \right\} \cup \left\{ \bigcup_{j=0}^{n_a-1} (A_{\mathcal{B}_{s-1}} + jn_b) \right\}.$$

The set

$$D_{\mathcal{C}} = \left(\bigcap_{i=0}^{s-2} D_{\mathcal{A}_i \otimes (\mathcal{B}_i \backslash \mathcal{B}_{i+1})} \right) \cap D_{\mathcal{A}_{s-1} \otimes \mathcal{B}_{s-1}}$$

is the defining set of a cyclic generalized product code $\mathcal{C} = \left(\bigoplus_{i=0}^{s-2} (\mathcal{A}_i \otimes \mathcal{B}_i \backslash \mathcal{B}_{i+1}) \right) \oplus (\mathcal{A}_{s-1} \otimes \mathcal{B}_{s-1})$ of order s. The generator polynomial is:

$$\begin{aligned} g(X) = \gcd \Big(& X^{n_a n_b} - 1, g_{\mathcal{A}_0}(X^{bn_b}) g_{\mathcal{B}_0 \backslash \mathcal{B}_1}(X^{an_a}), g_{\mathcal{A}_1}(X^{bn_b}) g_{\mathcal{B}_1 \backslash \mathcal{B}_2}(X^{an_a}), \\ & \ldots, g_{\mathcal{A}_{s-1}}(X^{bn_b}) g_{\mathcal{B}_{s-1}}(X^{an_a}) \Big). \end{aligned}$$

6.4.3 Example of a Cyclic Generalized Product Code

Let $n_a = 5$ and $n_b = 7$. We consider a cyclic generalized product code of order $s = 2$ and length $n_a n_b = 35$. Let $u = 3$ and $v = -2$ be the coefficients of a Bézout relation.

The two outer (row) codes are the $[5, 1, 5]_2$ cyclic repetition code $\mathcal{A}_0$ and the $[5, 4, 2]_2$ cyclic single-parity check code $\mathcal{A}_1$ with defining sets:

$$D_{\mathcal{A}_0} = \{1, 2, 3, 4\} \qquad \text{and} \qquad D_{\mathcal{A}_1} = \{0\}.$$

Let $\mathcal{B}_0$ be the single codeword $[7, 0, \infty]_2$ with defining set $D_{\mathcal{B}_0} = \{0, 1, 2, 3, 4, 5, 6\}$. Let $\mathcal{B}_1$ be the $[7, 4, 3]_2$ cyclic Hamming code $\mathcal{B}_1$ with defining set $D_{\mathcal{B}_1} = \{3, 5, 6\}$. Then, the first inner code $\mathcal{B}_0 \backslash \mathcal{B}_1$ is the $[7, 3, 4]_2$ cyclic code with defining set

$$D_{\mathcal{B}_0 \backslash \mathcal{B}_1} = \{0, 1, 2, 4\}.$$

According to Definition 6.42, we obtain the following shifted defining set:

$$\begin{aligned} B_{\mathcal{A}_0} &= (D_{\mathcal{A}_0} \cdot -2)_5 = \{1, 2, 3, 4\}, \\ B_{\mathcal{A}_1} &= (D_{\mathcal{A}_1} \cdot -2)_5 = \{0\}, \\ A_{\mathcal{B}_0 \backslash \mathcal{B}_1} &= (D_{\mathcal{B}_0 \backslash \mathcal{B}_1} \cdot 3)_7 = \{0, 3, 5, 6\}, \\ A_{\mathcal{B}_1} &= (D_{\mathcal{B}_1} \cdot 3)_7 = \{1, 2, 4\}. \end{aligned}$$

The shifted sets (according to Definition 6.42) are shown in row one, two, six and seven in the Table 6.6. The two rows with the symbol $\otimes$ in the first column are the defining sets of the corresponding cyclic product sub-codes $\mathcal{A}_0 \otimes (\mathcal{B}_0 \backslash \mathcal{B}_1)$ and $\mathcal{A}_1 \otimes \mathcal{B}_1$. The row in the middle, that has a $\oplus$ in the first column, is the defining set of the cyclic generalized product code $(\mathcal{A}_0 \otimes (\mathcal{B}_0 \backslash \mathcal{B}_1)) \oplus (\mathcal{A}_1 \otimes \mathcal{B}_1)$.

$B_{\mathcal{A}_0}$	□	1	2	3	4	□	1	2	3	4	□	1	2	3	4	□	1	2	3	4	□	1	2	3	4	□	1	2	3	4	□	1	2	3	4
$A_{\mathcal{B}_0 \setminus \mathcal{B}_1}$	0	□	□	3	□	5	6	0	□	□	3	□	5	6	0	□	□	3	□	5	6	0	□	□	3	□	5	6	0	□	□	3	□	5	6
$\otimes$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	□	16	17	18	19	20	21	22	23	24	□	26	27	28	29	□	31	32	33	34
$\oplus$	0	1	2	□	4	5	□	□	8	9	10	11	□	□	□	□	16	□	18	□	20	□	22	23	□	□	□	□	□	29	□	□	32	□	□
$\otimes$	0	1	2	□	4	5	□	□	8	9	10	11	□	□	□	15	16	□	18	□	20	□	22	23	□	25	□	□	□	29	30	□	32	□	□
$B_{\mathcal{A}_1}$	0	□	□	□	□	0	□	□	□	□	0	□	□	□	□	0	□	□	□	□	0	□	□	□	□	0	□	□	□	□	0	□	□	□	□
$A_{\mathcal{B}_1}$	□	1	2	□	4	□	□	□	1	2	□	4	□	□	□	1	2	□	4	□	□	□	1	2	□	4	□	□	□	1	2	□	4	□	□

Table 6.6: Illustration of the defining set of a cyclic generalized product code $\big(\mathcal{A}_0 \otimes (\mathcal{B}_0 \backslash \mathcal{B}_1)\big) \oplus (\mathcal{A}_1 \otimes \mathcal{B}_1)$ of order two. The first three rows give the summation of the first cyclic product sub-code $\mathcal{A}_0 \otimes (\mathcal{B}_0 \backslash \mathcal{B}_1)$. The last three rows give the summation of the second cyclic product sub-code $\mathcal{A}_1 \otimes \mathcal{B}_1$. In row four the defining set of the cyclic generalized product code is formed by the direct sum of row three and five.

The cyclic generalized product code is an $[35, 19, d]_2$ code with distance (according to Theorem 2.27):

$$d = \min\big(d_{a,0} \cdot d_{b,0}, d_{a,1} \cdot d_{b,1}\big) = \min\Big(5 \cdot \infty, 2 \cdot 3\Big) = 6.$$

The defining set of the cyclic generalized product code $(\mathcal{A}_0 \otimes (\mathcal{B}_0 \backslash \mathcal{B}_1)) \oplus (\mathcal{A}_1 \otimes \mathcal{B}_1)$ is the union of $M_{0,2}^{\langle 35 \rangle} \cup M_{1,2}^{\langle 35 \rangle} \cup M_{5,2}^{\langle 35 \rangle}$.

6.4.4 Using Cyclic Generalized Product Codes for Bounding the Minimum Distance

Similar to the approach in Section 6.2, we think it is possible to embed a given $[n, k, d]_q$ cyclic code $\mathcal{C}$ into a cyclic generalized product code to give a lower bound on its minimum distance d. In contrast to (cyclic) generalized concatenated codes, the minimum distance of (cyclic) generalized product codes is given by equality (see Lemma 6.41) and therefore a similar approach as in Section 6.2 seems possible. Let us assume the first product sub-code is used as the approach in Section 6.2. Then the other $s - 1$ product sub-codes of the cyclic generalized product code of order s would add "non-zeros" and therefore the obtained bound on the distance of the cyclic generalized product would hold for a wider class of cyclic codes than the one that uses only the product code as in Theorem 6.18.

6.5 Conclusion and Future Work

In this chapter, we presented two new approaches for bounding the minimum distance of linear cyclic codes. The first bound I-a is based on the association of a rational function to the sequence of zeros of a given cyclic code and is proven in Theorem 6.4. We gave the proof and a syndrome-based error decoding algorithm based on the Extended Euclidean Algorithm in Theorem 6.15. An error-evaluation strategy based on a generalization of Forney's formula was developed in Section 6.1.

The other three bounds I-b (Theorem 6.18), II (Theorem 6.25) and III (Theorem 6.26) are based on the association of another cyclic code. We propose a syndrome-based error/erasure decoding algorithm and an error-evaluation for bound I-b. The correctness of the error-only decoding approach for bound II was also proven in Section 6.2. A decoding method for bound III is an open task.

The defining sets of non-primitive binary lowest-code-rate cyclic codes of minimum distance two and of low-rate cyclic codes with minimum distance three are given in Proposition 6.34 and 6.35. We conclude that it corresponds to the defining sets of cyclic product codes in Section 6.3. The relevance of these codes for our approach of embedding a cyclic code into another cyclic product code was demonstrated.

In Section 6.4, we defined linear cyclic generalized product codes and proved their main properties. We outline how they can be used in a similar way as linear cyclic product codes to bound the minimum distance of a given linear cyclic code.

Besides this, several future research directions are possible. One issue for the approach of Section 6.2 is the complexity of the decoding algorithm when associating a second code. The order of the common extension field influences the decoding complexity directly, since all operations are done in this extension field.

The concept of Section 6.2 can be extended to several associated codes, that form then a linear cyclic product code of order $s \geq 2$. A deeper comparison to existing bounds should be carried out. Furthermore, conditions for non-binary lowest-rate cyclic code with distance two and three can be worked out. Similar results as the one for the binary cyclic codes of Section 6.3 are expected.

The approach as for the three bounds I-b, II and III can be extended to cyclic generalized product codes (see Section 6.4). Probably, also cyclic generalized concatenated codes can be used in a similar manner. Is it not excluded that these approaches can be used for linear (non-cyclic) codes, too. In principal every existing lower bound on the minimum distance of cyclic codes can be generalized as the BCH bound in Theorem 6.18 and the Hartmann–Tzeng bound in Theorem 6.25.

We provide a homepage [O-ZJ12] with numeric results for cyclic code over $\mathbb{F}_2, \mathbb{F}_3, \mathbb{F}_5$ and $\mathbb{F}_7$.

7

"Thanks for showing interest in the preliminary version of my book. It was halted for a year, since I had to do research and publish papers. A book counts for only one item!"

Ruud Pellikaan (born 1953) via email

Concluding Remarks

Within this dissertation, new algebraic soft- and hard-decision decoding approaches for Generalized Reed–Solomon and linear cyclic codes, both defined over finite fields and in Hamming metric, were developed.

We reformulated the bivariate interpolation problem of Guruswami–Sudan and Kötter–Vardy (see Chapter 4 and Chapter 5 respectively) and obtained a generalization of the classical Key Equation, which is the basis for the established syndrome-based unique Bounded Minimum Distance decoding approaches as the Berlekamp–Massey and the Sugiyama–Kasahara–Hirasawa–Namekawa algorithm. Based on the previous work of Roth and Ruckenstein for the Sudan algorithm (multiplicity one for all n points), a set of Key Equations for both cases, the Guruswami–Sudan approach, where all n points are interpolated with the same multiplicity (hard-decision) and the Kötter–Vardy extension, where the interpolation is based on a $q \times n$ multiplicity matrix (soft-decision), was derived. We obtained two systems of homogeneous linear equations, where the matrices are structured, i.e., a Block-Hankel matrix and a vertical band of Block-Hankel matrices respectively.

Both systems can be solved efficiently. We adapted the Fundamental Iterative Algorithm, that goes back to the work of Feng and Tzeng, for both cases. For the case of the hard-decision variant of Guruswami–Sudan (same multiplicity for all n points), we proposed the complexity-reducing initialization rule and proved the correctness of the Fundamental Iterative Algorithm. In addition, we analyzed its complexity. The reduction of equations and an explicit syndrome expression remain an open task. We showed that in the case of different multiplicities (Kötter–Vardy) and after re-encoding transformation, the univariate reformulation leads to a set of Key Equations over the polynomial ring. In addition, the reduced set of linear homogeneous equations consists still of vertically arranged Block-Hankel matrices, but with reduced dimensions.

In Chapter 6, we proposed two new techniques for bounding the minimum Hamming distance of a linear cyclic code. The first one used rational functions to fill missing zeros in the defining set of a given cyclic code. We identified several classes of codes and showed the connection to some existing bounds. Based on a new syndrome definition, a Key Equation with a generalized error-locator polynomial was derived. We adapted the Extended Euclidean Algorithm—similar to the approach of Sugiyama–Kasahara–Hirasawa–Namekawa—and proved a generalization of the Forney formula. The second technique embeds a given linear cyclic code into a linear cyclic product code. We prove the main theorems on the minimum distance that generalizes the Bose–Ray-Chaudhuri–Hocquenghem and the Hartmann–Tzeng bound. Probably, several other bounds on the minimum distance of linear cyclic codes can be extended in the same way. Similar to the rational function approach, a Key Equation for syndrome-based error/erasure decoding up the generalization of the Bose–Ray-Chaudhuri–Hocquenghem bound was given. Further, we proved the error-only syndrome-based decoding up to the generalized Hartmann–Tzeng bound.

Necessary and sufficient conditions for lowest-code-rate non-primitive binary cyclic codes of minimum distance two and a sufficient condition for binary cyclic codes of minimum distance three were given. We shown their relevance for the embedding technique. A further extension of the embedding-technique to cyclic generalized product codes was discussed and their basic properties were outlined.

Several open research directions were identified at the end of each chapter.

Bibliography

Articles

[A-Abr68] N. Abramson. “Cascade Decoding of Cyclic Product Codes”. *IEEE Trans. Commun.*, vol. 16.3 (1968), pp. 398–402. DOI: 10.1109/TCOM.1968.1089859 (cit. on p. 29).

[A-AK11] M. Ali and M. Kuijper. “A Parametric Approach to List Decoding of Reed–Solomon Codes Using Interpolation”. *IEEE Trans. Inform. Theory*, vol. 57.10 (2011), pp. 6718–6728. DOI: 10.1109/TIT.2011.2165803 (cit. on p. 66).

[A-AL96] D. Augot and F. Levy-dit-Vehel. “Bounds on the Minimum Distance of the Duals of BCH Codes”. *IEEE Trans. Inform. Theory*, vol. 42.4 (1996), pp. 1257–1260. DOI: 10.1109/18.508853 (cit. on p. 107).

[A-Ale05] M. Alekhnovich. “Linear Diophantine Equations over Polynomials and Soft Decoding of Reed–Solomon Codes”. *IEEE Trans. Inform. Theory*, vol. 51.7 (2005), pp. 2257–2265. DOI: 10.1109/TIT.2005.850097 (cit. on pp. 65–67).

[A-Arm10] M. A. Armand. “A Note on Interleaved Reed-Solomon Codes Over Galois Rings”. *IEEE Trans. Inform. Theory*, vol. 56.4 (2010), pp. 1574–1581. DOI: 10.1109/TIT.2010.2040943 (cit. on p. 15).

[A-Aug04] D. Augot. “Madhu Sudan’s Work on Error-Correcting Codes”. *Eur. Math. Soc. Newsl.*, vol. 51 (2004), pp. 8–10 (cit. on p. 60).

[A-Bas65] L. A. Bassalygo. “New Upper Bounds for Error-Correcting Codes”. *Probl. Inf. Transm.*, vol. 1.4 (1965), pp. 41–44 (cit. on pp. 16, 61).

[A-BB10a] P. Beelen and K. Brander. “Efficient List Decoding of a Class of Algebraic-Geometry Codes”. *Adv. Math. Commun*, vol. 4.4 (2010), pp. 485–518. DOI: 10.3934/amc.2010.4.485 (cit. on p. 67).

[A-BB10b] P. Beelen and K. Brander. “Key Equations for List Decoding of Reed–Solomon Codes and How to Solve Them”. *J. Symbolic Comput.*, vol. 45.7 (2010), pp. 773–786. DOI: 10.1016/j.jsc.2010.03.010 (cit. on pp. 65–67).

[A-Ber72] E. R. Berlekamp. “A Survey of Coding Theory”. *J. R. Stat. Soc. Ser. A*, vol. 135.1 (1972), pp. 44–73 (cit. on p. 15).

[A-BHH13] M. Blaum, J. Hafner, and S. Hetzler. “Partial-MDS Codes and their Application to RAID Type of Architectures”. *IEEE Trans. Inform. Theory*, vol. Early Access Online (2013). DOI: 10.1109/TIT.2013.2252395 (cit. on p. 15).

[A-BHHW98] I. Blake, C. Heegard, T. Hoholdt, and V. Wei. “Algebraic-Geometry Codes”. *IEEE Trans. Inform. Theory*, vol. 44.6 (1998), pp. 2596–2618. DOI: 10.1109/18.720550 (cit. on p. 15).

[A-BHNW13] P. Beelen, T. Høholdt, J. S. R. Nielsen, and Y. Wu. “On Rational Interpolation-Based List-Decoding and List-Decoding Binary Goppa Codes”. *IEEE Trans. Inform. Theory*, vol. 59.6 (2013), pp. 3269–3281. DOI: 10.1109/TIT.2013.2243800 (cit. on p. 65).

[A-BJ74] E. R. Berlekamp and J. Justesen. "Some Long Cyclic Linear Binary Codes Are Not So Bad". *IEEE Trans. Inform. Theory*, vol. 20.3 (1974), pp. 351–356. DOI: 10.1109/TIT.1974.1055222 (cit. on p. 139).

[A-BK10] J. Bellorado and A. Kavcic. "Low-Complexity Soft-Decoding Algorithms for Reed–Solomon Codes—Part I: An Algebraic Soft-In Hard-Out Chase Decoder". *IEEE Trans. Inform. Theory*, vol. 56.3 (2010), pp. 945–959. DOI: 10.1109/TIT.2009.2039073 (cit. on p. 42).

[A-BKMP10] J. Bellorado, A. Kavcic, M. Marrow, and L. Ping. "Low-Complexity Soft-Decoding Algorithms for Reed-Solomon Codes—Part II: Soft-Input Soft-Output Iterative Decoding". *IEEE Trans. Inform. Theory*, vol. 56.3 (2010), pp. 960–967. DOI: 10.1109/TIT.2009.2039091 (cit. on p. 42).

[A-BKY07] D. Bleichenbacher, A. Kiayias, and M. Yung. "Decoding Interleaved Reed–Solomon Codes over Noisy Channels". *Theor. Comput. Sci.*, vol. 379.3 (2007), pp. 348–360. DOI: 10.1016/j.tcs.2007.02.043 (cit. on p. 38).

[A-BMVT78] E. R. Berlekamp, R. McEliece, and H. C. A. Van Tilborg. "On the Inherent Intractability of Certain Coding Problems". *IEEE Trans. Inform. Theory*, vol. 24.3 (1978), pp. 384–386. DOI: 10.1109/TIT.1978.1055873 (cit. on p. 41).

[A-Bos01] N. Boston. "Bounding Minimum Distances of Cyclic Codes Using Algebraic Geometry". *Electron. Notes Discrete Math.*, vol. 6 (2001), pp. 385–394. DOI: 10.1016/S1571-0653(04)00190-8 (cit. on pp. 107, 108, 116, 117).

[A-BRC60] R. C. Bose and D. K. Ray-Chaudhuri. "On A Class of Error Correcting Binary Group Codes". *Inf. Control*, vol. 3.1 (1960), pp. 68–79. DOI: 10.1016/S0019-9958(60)90287-4 (cit. on pp. 15, 19, 27, 107, 112).

[A-BS06] E. Betti and M. Sala. "A New Bound for the Minimum Distance of a Cyclic Code From Its Defining Set". *IEEE Trans. Inform. Theory*, vol. 52.8 (2006), pp. 3700–3706. DOI: 10.1109/TIT.2006.876240 (cit. on p. 107).

[A-BW65] H. Burton and E. Weldon. "Cyclic Product Codes". *IEEE Trans. Inform. Theory*, vol. 11.3 (1965), pp. 433–439. DOI: 10.1109/TIT.1965.1053802 (cit. on p. 29).

[A-BZ74] E. L. Blokh and V. V. Zyablov. "Coding of Generalized Concatenated Codes". *Probl. Inf. Transm.*, vol. 10.3 (1974), pp. 45–50 (cit. on pp. 16, 32, 138).

[A-CF07] D. J. Costello and G. D. Forney. "Channel Coding: The Road to Channel Capacity". *Proc. IEEE*, vol. 95.6 (2007), pp. 1150–1177. DOI: 10.1109/JPROC.2007.895188 (cit. on p. 15).

[A-Cha72] D. Chase. "Class of Algorithms for Decoding Block Codes With Channel Measurement Information". *IEEE Trans. Inform. Theory*, vol. 18.1 (1972), pp. 170–182. DOI: 10.1109/TIT.1972.1054746 (cit. on p. 42).

[A-Chi64] R. Chien. "Cyclic Decoding Procedures for Bose-Chaudhuri-Hocquenghem Codes". *IEEE Trans. Inform. Theory*, vol. 10.4 (1964), pp. 357–363. DOI: 10.1109/TIT.1964.1053699 (cit. on p. 40).

[A-CTZ97] P. Charpin, A. Tietäväinen, and V. A. Zinoviev. "On Binary Cyclic Codes with Minimum Distance $d = 3$". *Probl. Inf. Transm.*, vol. 33.4 (1997), pp. 287–296 (cit. on pp. 17, 134–136).

[A-CTZ99] P. Charpin, A. Tietäväinen, and V. A. Zinoviev. "On the Minimum Distances of Non-Binary Cyclic Codes". *Des. Codes Cryptogr.*, vol. 17 (1999), pp. 81–85. DOI: 10.1023/A:1008354504832 (cit. on pp. 17, 134).

[A-DB95] D. Dabiri and I. F. Blake. “Fast Parallel Algorithms for Decoding Reed–Solomon Codes Based on Remainder Polynomials”. *IEEE Trans. Inform. Theory*, vol. 41.4 (1995), pp. 873–885. DOI: 10.1109/18.391235 (cit. on p. 43).

[A-Del75] P. Delsarte. “On Subfield Subcodes of Modified Reed–Solomon Codes”. *IEEE Trans. Inform. Theory*, vol. 21.5 (1975), pp. 575–576. DOI: 10.1109/TIT.1975.1055435 (cit. on pp. 16, 19, 34).

[A-Del78] P. Delsarte. “Bilinear Forms over a Finite Field, with Applications to Coding Theory”. *J. Combin. Theory Ser. A*, vol. 25.3 (1978), pp. 226–241. DOI: 10.1016/0097-3165(78)90015-8 (cit. on p. 15).

[A-DK94] I. M. Duursma and R. Kötter. “Error-Locating Pairs for Cyclic Codes”. *IEEE Trans. Inform. Theory*, vol. 40.4 (1994), pp. 1108–1121. DOI: 10.1109/18.335964 (cit. on p. 107).

[A-DKT07] I. I. Dumer, G. A. Kabatiansky, and C. Tavernier. “List Decoding of the First-Order Binary Reed–Muller Codes”. *Probl. Inf. Transm.*, vol. 43.3 (2007), pp. 225–232. DOI: 10.1134/S0032946007030052 (cit. on p. 16).

[A-DMS03] I. I. Dumer, D. Micciancio, and M. Sudan. “Hardness of Approximating the Minimum Distance of a Linear Code”. *IEEE Trans. Inform. Theory*, vol. 49.1 (2003), pp. 22–37. DOI: 10.1109/TIT.2002.806118 (cit. on p. 107).

[A-Dor87] J. Dornstetter. “On the Equivalence Between Berlekamp’s and Euclid’s Algorithms”. *IEEE Trans. Inform. Theory*, vol. 33.3 (1987), pp. 428–431. DOI: 10.1109/TIT.1987.1057299 (cit. on p. 50).

[A-DP06] I. M. Duursma and R. Pellikaan. “A Symmetric Roos Bound for Linear Codes”. *Journal of Combinatorial Theory Series A - Special Issue in Honor of J. H. van Lint*, vol. 113 (2006), pp. 1677–1688. DOI: 10.1016/j.jcta.2006.03.020 (cit. on p. 107).

[A-EKM06] M. E. El-Khamy and R. J. Mceliece. “Iterative Algebraic Soft-Decision List Decoding of Reed–Solomon Codes”. *IEEE J. Sel. Areas Commun.*, vol. 24.3 (2006), 481–490 (cit. on p. 42).

[A-Eli54] P. Elias. “Error-Free Coding”. *IEEE Trans. Inform. Theory*, vol. 4.4 (1954), pp. 29–37. DOI: 10.1109/TIT.1954.1057464 (cit. on p. 28).

[A-For65] G. D. Forney. “On Decoding BCH Codes”. *IEEE Trans. Inform. Theory*, vol. 11.4 (1965), pp. 549–557. DOI: 10.1109/TIT.1965.1053825 (cit. on pp. 47, 51, 108, 119, 120).

[A-For66b] G. D. Forney. “Generalized Minimum Distance Decoding”. *IEEE Trans. Inform. Theory*, vol. 12.2 (1966), pp. 125–131. DOI: 10.1109/TIT.1966.1053873 (cit. on p. 42).

[A-FT08] R. Fourquet and C. Tavernier. “An Improved List Decoding Algorithm for the Second Order Reed–Muller Codes and its Applications”. *Des. Codes Cryptogr.*, vol. 49 (2008), pp. 323–340. DOI: 10.1007/s10623-008-9184-8 (cit. on p. 16).

[A-FT85] G.-L. Feng and K. Tzeng. “An Iterative Algorithm of Shift-Register Synthesis for Multiple Sequences”. *Scientia Sinica (Series A)*, vol. 28.11 (1985), pp. 1222–1232 (cit. on pp. 39, 50, 52).

[A-FT89] G.-L. Feng and K. Tzeng. “A Generalized Euclidean Algorithm for Multisequence Shift-Register Synthesis”. *IEEE Trans. Inform. Theory*, vol. 35.3 (1989), pp. 584–594. DOI: 10.1109/18.30981 (cit. on pp. 16, 60, 107, 131, 134).

[A-FT91a] G.-L. Feng and K. Tzeng. "A Generalization of the Berlekamp–Massey Algorithm for Multisequence Shift-Register Synthesis with Applications to Decoding Cyclic Codes". *IEEE Trans. Inform. Theory*, vol. 37.5 (1991), pp. 1274–1287. DOI: 10.1109/18.133246 (cit. on pp. 39, 50, 52–54, 107, 131–134).

[A-FT91b] G.-L. Feng and K. Tzeng. "Decoding Cyclic and BCH Codes up to Actual Minimum Distance Using Nonrecurrent Syndrome Dependence Relations". *IEEE Trans. Inform. Theory*, vol. 37.6 (1991), pp. 1716–1723. DOI: 10.1109/18.104340 (cit. on pp. 16, 53).

[A-Gab85] E. M. Gabidulin. "Theory of Codes with Maximum Rank Distance". *Probl. Peredachi Inf.*, vol. 21.1 (1985), pp. 3–16 (cit. on p. 15).

[A-GKKG06] W. Gross, F. Kschischang, R. Kötter, and P. Gulak. "Applications of Algebraic Soft-Decision Decoding of Reed–Solomon Codes". *IEEE Trans. Commun.*, vol. 54.7 (2006), pp. 1224–1234. DOI: 10.1109/TCOMM.2006.877972 (cit. on p. 42).

[A-Gop70] V. D. Goppa. "A New Class of Linear Error Correcting Codes". *Probl. Peredachi. Inf.*, vol. 6 (1970), pp. 24–30 (cit. on p. 108).

[A-Gop71] V. D. Goppa. "Rational Representation of Codes and (L,g) Codes". *Probl. Peredachi. Inf.*, vol. 7 (1971), pp. 41–49 (cit. on p. 108).

[A-Gor70] W. Gore. "Further Results on Product Codes". *IEEE Trans. Inform. Theory*, vol. 16.4 (1970), pp. 446–451. DOI: 10.1109/TIT.1970.1054477 (cit. on p. 138).

[A-GR08] V. Guruswami and A. Rudra. "Explicit Codes Achieving List Decoding Capacity: Error-Correction With Optimal Redundancy". *IEEE Trans. Inform. Theory*, vol. 54.1 (2008), pp. 135–150. DOI: 10.1109/TIT.2007.911222 (cit. on p. 15).

[A-GRS00a] O. Goldreich, D. Ron, and M. Sudan. "Chinese Remaindering with Errors". *IEEE Trans. Inform. Theory*, vol. 46.4 (2000), pp. 1330–1338. DOI: 10.1109/18.850672 (cit. on p. 16).

[A-GRS00b] O. Goldreich, R. Rubinfeld, and M. Sudan. "Learning Polynomials with Queries: The Highly Noisy Case". *SIAM J. Discrete Math.*, vol. 13.4 (2000), pp. 535–570. DOI: 10.1137/S0895480198344540 (cit. on p. 16).

[A-GS92] P. Gemmell and M. Sudan. "Highly Resilient Correctors for Polynomials". *Inform. Process. Lett.*, vol. 43.4 (1992), pp. 169–174. DOI: 10.1016/0020-0190(92)90195-2 (cit. on p. 43).

[A-GS99] V. Guruswami and M. Sudan. "Improved Decoding of Reed–Solomon and Algebraic-Geometry Codes". *IEEE Trans. Inform. Theory*, vol. 45.6 (1999), pp. 1757–1767. DOI: 10.1109/18.782097 (cit. on pp. 16, 39, 41, 42, 60–62, 65, 66).

[A-GV05] V. Guruswami and A. Vardy. "Maximum-Likelihood Decoding of Reed–Solomon Codes is NP-Hard". *IEEE Trans. Inform. Theory*, vol. 51.7 (2005), pp. 2249–2256. DOI: 10.1109/TIT.2005.850102 (cit. on p. 41).

[A-GW12] V. Guruswami and C. Wang. "Linear-Algebraic List Decoding for Variants of Reed–Solomon Codes". *Electronic Colloquium on Computational Complexity (ECCC)*, vol. 19.73 (2012), pp. 1–24 (cit. on p. 95).

[A-GZ61] D. Gorenstein and N. Zierler. "A Class of Error-Correcting Codes in p^m". *J. Soc. Ind. Appl. Math.*, vol. 9.2 (1961), pp. 207–214 (cit. on p. 40).

[A-Ham50] R. W. Hamming. "Error Detecting and Error Correcting Codes". *Bell Syst. Tech. J.*, vol. 29.2 (1950), pp. 147–160 (cit. on p. 15).

[A-Har72] C. Hartmann. "Decoding Beyond the BCH Bound". *IEEE Trans. Inform. Theory*, vol. 18.3 (1972), pp. 441–444. DOI: 10.1109/TIT.1972.1054824 (cit. on pp. 19, 26, 107).

[A-Has36] H. Hasse. “Theorie der höheren Differentiale in einem algebraischen Funktionenkörper mit vollkommenem Konstantenkörper bei beliebiger Charakteristik”. *J. Reine Angew. Math.*, vol. 1936.175 (1936), pp. 50–54 (cit. on p. 21).

[A-Her78] C. Hermite. “Sur la Formule d’Interpolation de Lagrange”. *J. Reine Angew. Math.*, vol. 84.1 (1878), pp. 70–79 (cit. on p. 92).

[A-Hoc59] A. Hocquenghem. “Codes Correcteurs d’Erreurs”. *Chiffres (Paris)*, vol. 2 (1959), pp. 147–156 (cit. on pp. 15, 19, 27, 107, 112).

[A-HT72] C. Hartmann and K. Tzeng. “Generalizations of the BCH Bound”. *Inf. Control*, vol. 20.5 (1972), pp. 489–498. DOI: 10.1016/S0019-9958(72)90887-X (cit. on pp. 16, 19, 26, 107, 112).

[A-HT74] C. Hartmann and K. Tzeng. “Decoding Beyond the BCH Bound Using Multiple Sets of Syndrome Sequences”. *IEEE Trans. Inform. Theory*, vol. 20.2 (1974), pp. 292–295. DOI: 10.1109/TIT.1974.1055178 (cit. on pp. 19, 107).

[A-HTC72] C. Hartmann, K. Tzeng, and R. Chien. “Some Results on the Minimum Distance Structure of Cyclic Codes”. *IEEE Trans. Inform. Theory*, vol. 18.3 (1972), pp. 402–409. DOI: 10.1109/TIT.1972.1054816 (cit. on pp. 19, 107).

[A-Jen85] J. Jensen. “The Concatenated Structure of Cyclic and Abelian Codes”. *IEEE Trans. Inform. Theory*, vol. 31.6 (1985), pp. 788–793. DOI: 10.1109/TIT.1985.1057109 (cit. on p. 139).

[A-Jen92] J. Jensen. “Cyclic Concatenated Codes with Constacyclic Outer Codes”. *IEEE Trans. Inform. Theory*, vol. 38.3 (1992), pp. 950 –959. DOI: 10.1109/18.135637 (cit. on p. 139).

[A-JH00] J. Jensen and A. Heydtmann. “On the Equivalence of the Berlekamp–Massey and the Euclidean Algorithms for Decoding”. *IEEE Trans. Inform. Theory*, vol. 46.7 (2000), pp. 2614–2624. DOI: 10.1109/18.887869 (cit. on p. 50).

[A-JN04] J. Jiang and K. Narayanan. “Iterative Soft Decoding of Reed–Solomon Codes”. *IEEE Commun. Lett.*, vol. 8.4 (2004), pp. 244–246. DOI: 10.1109/LCOMM.2004.827977 (cit. on p. 42).

[A-JN06] J. Jiang and K. Narayanan. “Iterative Soft-Input Soft-Output Decoding of Reed–Solomon Codes by Adapting the Parity-Check Matrix”. *IEEE Trans. Inform. Theory*, vol. 52.8 (2006), pp. 3746–3756. DOI: 10.1109/TIT.2006.878176 (cit. on p. 42).

[A-Joh62] S. Johnson. “A New Upper Bound for Error-Correcting Codes”. *IRE Trans. Inf. Theor.*, vol. 8.3 (1962), pp. 203–207. DOI: 10.1109/TIT.1962.1057714 (cit. on pp. 16, 61).

[A-Kö96a] R. Kötter. “Fast Generalized Minimum-Distance Decoding of Algebraic-Geometry and Reed–Solomon Codes”. *IEEE Trans. Inform. Theory*, vol. 42.3 (1996), pp. 721–737. DOI: 10.1109/18.490540 (cit. on pp. 66, 101).

[A-KK08] R. Kötter and F. Kschischang. “Coding for Errors and Erasures in Random Network Coding”. *IEEE Trans. Inform. Theory*, vol. 54.8 (2008), pp. 3579–3591. DOI: 10.1109/TIT.2008.926449 (cit. on p. 15).

[A-KMV11] R. Kötter, J. Ma, and A. Vardy. “The Re-Encoding Transformation in Algebraic List-Decoding of Reed–Solomon Codes”. *IEEE Trans. Inform. Theory*, vol. 57.2 (2011), pp. 633–647. DOI: 10.1109/TIT.2010.2096034 (cit. on pp. 16, 17, 60, 66, 97, 101, 102).

[A-KP04] M. Kuijper and J. Polderman. “Reed–Solomon List Decoding From a System-Theoretic Perspective”. *IEEE Trans. Inform. Theory*, vol. 50.2 (2004), pp. 259–271. DOI: 10.1109/TIT.2003.822593 (cit. on p. 66).

[A-Kra03] V. Krachkovsky. “Reed–Solomon Codes for Correcting Phased Error Bursts”. *IEEE Trans. Inform. Theory*, vol. 49.11 (2003), pp. 2975–2984. DOI: 10.1109/TIT.2003.819333 (cit. on pp. 38, 60).

[A-Kra97] V. Krachkovsky. “Decoding for Iterative Reed–Solomon Coding Schemes”. *IEEE Trans. Magn.*, vol. 33.5 (1997), pp. 2740–2742. DOI: 10.1109/20.617715 (cit. on pp. 15, 38).

[A-KV03a] R. Kötter and A. Vardy. “Algebraic Soft-Decision Decoding of Reed–Solomon Codes”. *IEEE Trans. Inform. Theory*, vol. 49.11 (2003), pp. 2809–2825. DOI: 10.1109/TIT.2003.819332 (cit. on pp. 16, 39, 60, 62–64, 97, 99).

[A-LO09] K. Lee and M. E. O’Sullivan. “List Decoding of Hermitian Codes using Gröbner Bases”. *J. Symbolic Comput.*, vol. 44.12 (2009), pp. 1662–1675. DOI: 10.1016/j.jsc.2007.12.004 (cit. on pp. 65, 66).

[A-LW70] S. Lin and E. J. Weldon. “Further Results on Cyclic Product Codes”. *IEEE Trans. Inform. Theory*, vol. 16.4 (1970), pp. 452–459. DOI: 10.1109/TIT.1970.1054491 (cit. on pp. 29, 30).

[A-LW86] J. van Lint and R. Wilson. “On The Minimum Distance of Cyclic Codes”. *IEEE Trans. Inform. Theory*, vol. 32.1 (1986), pp. 23–40. DOI: 10.1109/TIT.1986.1057134 (cit. on pp. 107, 133).

[A-Man76] D. Mandelbaum. “On a Class of Arithmetic Codes and a Decoding Algorithm”. *IEEE Trans. Inform. Theory*, vol. 22.1 (1976), pp. 85–88. DOI: 10.1109/TIT.1976.1055504 (cit. on p. 15).

[A-Mar68] A. Marchukov. “Summation of the Products of Codes”. *Probl. Inf. Transm.*, vol. 4.2 (1968), pp. 8–15 (cit. on p. 138).

[A-Mas64] J. L. Massey. “Reversible Codes”. *Inf. Control*, vol. 7.3 (1964), pp. 369–380. DOI: 10.1016/S0019-9958(64)90438-3 (cit. on pp. 108, 112).

[A-Mas69] J. L. Massey. “Shift-Register Synthesis and BCH Decoding”. *IEEE Trans. Inform. Theory*, vol. 15.1 (1969), pp. 122–127. DOI: 10.1109/TIT.1969.1054260 (cit. on pp. 16, 50, 107).

[A-Mul54] D. Muller. “Application of Boolean Algebra to Switching Circuit Design and to Error Detection”. *IRE Trans. Inf. Theor.*, vol. EC-3.3 (1954), pp. 6–12. DOI: 10.1109/IREPGELC.1954.6499441 (cit. on p. 15).

[A-Pet60] W. Peterson. “Encoding and Error-Correction Procedures for the Bose-Chaudhuri Codes”. *IEEE Trans. Inform. Theory*, vol. 6.4 (1960), pp. 459–470. DOI: 10.1109/TIT.1960.1057586 (cit. on pp. 15, 40).

[A-Pla97] J. S. Plank. “A Tutorial on Reed–Solomon Coding for Fault-Tolerance in RAID-Like Systems”. *Softw. Pract. Exper.*, vol. 27.9 (1997), pp. 995–1012. DOI: 10.1002/(SICI)1097-024X(199709)27:9¡995::AID-SPE111¿3.3.CO;2-Y (cit. on p. 15).

[A-Ree54] I. Reed. “A Class of Multiple-Error-Correcting Codes and the Decoding Scheme”. *IEEE Trans. Inform. Theory*, vol. 4.4 (1954), pp. 38–49. DOI: 10.1109/TIT.1954.1057465 (cit. on p. 15).

[A-RL91] P. de Rooij and J. van Lint. “More on the Minimum Distance of Cyclic Codes”. *IEEE Trans. Inform. Theory*, vol. 37.1 (1991), pp. 187–189. DOI: 10.1109/18.61137 (cit. on p. 139).

[A-Roo82] C. Roos. “A Generalization of the BCH Bound for Cyclic Codes, Including the Hartmann–Tzeng Bound”. *J. Combin. Theory Ser. A*, vol. 33.2 (1982), pp. 229–232. DOI: 10.1016/0097-3165(82)90014-0 (cit. on p. 107).

[A-Roo83] C. Roos. "A New Lower Bound for the Minimum Distance of a Cyclic Code". *IEEE Trans. Inform. Theory*, vol. 29.3 (1983), pp. 330–332. DOI: 10.1109/TIT.1983.1056672 (cit. on p. 107).

[A-Rot91] R. Roth. "Maximum-Rank Array Codes and Their Application to Crisscross Error Correction". *IEEE Trans. Inform. Theory*, vol. 37.2 (1991), pp. 328–336. DOI: 10.1109/18.75248 (cit. on p. 15).

[A-RR00] R. Roth and G. Ruckenstein. "Efficient Decoding of Reed–Solomon Codes Beyond Half the Minimum Distance". *IEEE Trans. Inform. Theory*, vol. 46.1 (2000), pp. 246–257. DOI: 10.1109/18.817522 (cit. on pp. 40, 66, 67, 72, 73, 75, 77, 91).

[A-RS60] I. S. Reed and G. Solomon. "Polynomial Codes Over Certain Finite Fields". *J. Soc. Ind. Appl. Math.*, vol. 8.2 (1960), pp. 300–304. DOI: 10.1137/0108018 (cit. on pp. 15, 19, 34).

[A-Sha48] C. E. Shannon. "A Mathematical Theory of Communication". *Bell Syst. Tech. J.*, vol. 27 (1948), pp. 370–423, 623–656 (cit. on p. 15).

[A-Sho06] A. Shokrollahi. "Raptor Codes". *IEEE Trans. Inform. Theory*, vol. 52.6 (2006), pp. 2551–2567. DOI: 10.1109/TIT.2006.874390 (cit. on p. 15).

[A-SKHN75] Y. Sugiyama, M. Kasahara, S. Hirasawa, and T. Namekawa. "A Method for Solving Key Equation for Decoding Goppa Codes". *Inf. Control*, vol. 27.1 (1975), pp. 87–99. DOI: 10.1016/S0019-9958(75)90090-X (cit. on pp. 16, 39, 50, 51, 107).

[A-SKHN76] Y. Sugiyama, M. Kasahara, S. Hirasawa, and T. Namekawa. "An Erasures-and-Errors Decoding Algorithm for Goppa Codes". *IEEE Trans. Inform. Theory*, vol. 22.2 (1976), pp. 238–241. DOI: 10.1109/TIT.1976.1055517 (cit. on pp. 39, 47, 51, 107).

[A-SKK08] D. Silva, F. Kschischang, and R. Kötter. "A Rank-Metric Approach to Error Control in Random Network Coding". *IEEE Trans. Inform. Theory*, vol. 54.9 (2008), pp. 3951–3967. DOI: 10.1109/TIT.2008.928291 (cit. on p. 15).

[A-Sle60] D. Slepian. "Some Further Theory of Group Codes". *Bell Syst. Tech. J.*, vol. 39.5 (1960), pp. 1219–1252 (cit. on pp. 16, 28).

[A-SLX12] T. Siyun, C. Li, and M. Xiao. "Progressive List-Enlarged Algebraic Soft Decoding of Reed–Solomon Codes". *IEEE Commun. Lett.*, vol. 16.6 (2012), pp. 901–904. DOI: 10.1109/LCOMM.2012.042512.112511 (cit. on p. 42).

[A-SM81] N. A. Shekhunova and E. T. Mironchikov. "Cyclic (L,G) Codes". *Probl. Peredachi. Inf.*, vol. 17.2 (1981), pp. 3–10 (cit. on p. 108).

[A-Spi60] A. Spitzbart. "A Generalization of Hermite's Interpolation Formula". *Amer. Math. Monthly*, vol. 67.1 (1960), pp. 42–46. DOI: 10.2307/2308924 (cit. on p. 92).

[A-SS11] V. R. Sidorenko and G. Schmidt. "A Linear Algebraic Approach to Multisequence Shift-Register Synthesis". *Probl. Inf. Transm.*, vol. 47.2 (2011), pp. 149–165. DOI: 10.1134/S0032946011020062 (cit. on pp. 54, 131).

[A-SSB09] G. Schmidt, V. R. Sidorenko, and M. Bossert. "Collaborative Decoding of Interleaved Reed–Solomon Codes and Concatenated Code Designs". *IEEE Trans. Inform. Theory*, vol. 55.7 (2009), pp. 2991–3012. DOI: 10.1109/TIT.2009.2021308 (cit. on p. 38).

[A-SSB10] G. Schmidt, V. R. Sidorenko, and M. Bossert. "Syndrome Decoding of Reed–Solomon Codes Beyond Half the Minimum Distance Based on Shift-Register Synthesis". *IEEE Trans. Inform. Theory*, vol. 56.10 (2010), pp. 5245–5252. DOI: 10.1109/TIT.2010.2060130 (cit. on pp. 40, 67, 68, 70–72).

[A-Sud97] M. Sudan. "Decoding of Reed–Solomon Codes Beyond the Error-Correction Bound". *J. Complexity*, vol. 13.1 (1997), pp. 180–193. DOI: 10.1006/jcom.1997.0439 (cit. on pp. 16, 39, 41, 60, 65–67, 72).

[A-SW99] M. Shokrollahi and H. Wasserman. "List Decoding of Algebraic-Geometric Codes". *IEEE Trans. Inform. Theory*, vol. 45.2 (1999), pp. 432–437. DOI: 10.1109/18.748993 (cit. on p. 41).

[A-SWTS96] K. Shen, C. Wang, K. Tzeng, and B.-Z. Shen. "Generation of Matrices for Determining Minimum Distance and Decoding of Cyclic Codes". *IEEE Trans. Inform. Theory*, vol. 42.2 (1996), pp. 653–657. DOI: 10.1109/18.485738 (cit. on p. 107).

[A-TF94] K. Tzeng and G.-L. Feng. "A New Procedure for Decoding Cyclic and BCH Codes up to Actual Minimum Distance". *IEEE Trans. Inform. Theory*, vol. 40.5 (1994), 1364–1374. DOI: 10.1109/18.333854 (cit. on p. 107).

[A-Tri07] P. Trifonov. "Interpolation in List Decoding of Reed–Solomon Codes". *Probl. Inf. Transm.*, vol. 43.3 (2007), pp. 190–198. DOI: 10.1134/S0032946007030027 (cit. on pp. 65, 66).

[A-Tri10] P. Trifonov. "Efficient Interpolation in the Guruswami–Sudan Algorithm". *IEEE Trans. Inform. Theory*, vol. 56.9 (2010), pp. 4341–4349. DOI: 10.1109/TIT.2010.2053901 (cit. on pp. 65, 66).

[A-Var97] A. Vardy. "The Intractability of Computing the Minimum Distance of a Code". *IEEE Trans. Inform. Theory*, vol. 43.6 (1997), pp. 1757–1766. DOI: 10.1109/18.641542 (cit. on p. 107).

[A-Wu08] Y. Wu. "New List Decoding Algorithms for Reed–Solomon and BCH Codes". *IEEE Trans. Inform. Theory*, vol. 54.8 (2008), pp. 3611–3630. DOI: 10.1109/TIT.2008.926355 (cit. on pp. 65, 66).

[A-YB94] T. Yaghoobian and I. F. Blake. "Two New Decoding Algorithms for Reed–Solomon Codes". *Appl. Algebra Engrg. Comm. Comput.*, vol. 5.1 (1994), pp. 23–43. DOI: 10.1007/BF01196623 (cit. on p. 43).

[A-ZB12a] A. Zeh and S. Bezzateev. "A New Bound on the Minimum Distance of Cyclic Codes Using Small-Minimum-Distance Cyclic Codes". *Des. Codes Cryptogr.*, (2012), pp. 1–18. DOI: 10.1007/s10623-012-9721-3 (cit. on pp. 107, 108, 124, 135).

[A-ZGA11] A. Zeh, C. Gentner, and D. Augot. "An Interpolation Procedure for List Decoding Reed–Solomon Codes Based on Generalized Key Equations". *IEEE Trans. Inform. Theory*, vol. 57.9 (2011), pp. 5946–5959. DOI: 10.1109/TIT.2011.2162160 (cit. on p. 67).

[A-Zin76] V. A. Zinoviev. "Generalized Cascade Codes". *Probl. Inf. Transm.*, vol. 12.1 (1976), pp. 5–15 (cit. on pp. 32, 33).

[A-ZSB99] V. V. Zyablov, S. Shavgulidze, and M. Bossert. "An Introduction to Generalized Concatenated Codes". *Eur. Trans. Telecommun.*, vol. 10.6 (1999), pp. 609–622. DOI: 10.1002/ett.4460100606 (cit. on pp. 32, 34).

[A-ZW11] A. Zeh and A. Wachter. "Fast Multi-Sequence Shift-Register Synthesis with the Euclidean Algorithm". *Adv. Math. Commun*, vol. 5.4 (2011), pp. 667–680. DOI: 10.3934/amc.2011.5.667 (cit. on pp. 60, 131, 134).

[A-ZWB12b] A. Zeh, A. Wachter-Zeh, and S. V. Bezzateev. "Decoding Cyclic Codes up to a New Bound on the Minimum Distance". *IEEE Trans. Inform. Theory*, vol. 58.6 (2012), pp. 3951–3960. DOI: 10.1109/TIT.2012.2185924 (cit. on pp. 107, 108, 131).

[A-O'S02] M. E. O'Sullivan. "The Key Equation for One-Point Codes and Efficient Error Evaluation". *J. Pure Appl. Algebra*, vol. 169.2-3 (2002), 295–320. DOI: 10.1016/S0022-4049(01)00074-3 (cit. on p. 95).

[A-Wac13] A. Wachter-Zeh. "Bounds on List Decoding of Rank Metric Codes". *accepted for IEEE Transactions on Information Theory*, (2013) (cit. on p. 16).

Books

[B-Ber68] E. R. Berlekamp. *Algebraic Coding Theory.* McGraw-Hill, 1968 (cit. on pp. 16, 50, 107).

[B-Bla83] R. E. Blahut. *Theory and Practice of Error Control Codes.* Addison-Wesley, 1983 (cit. on p. 16).

[B-Bos13] M. Bossert. *Kanalcodierung.* überarbeitete Auflage. Oldenbourg Wissenschaftsverlag, 2013 (cit. on pp. 16, 25, 33).

[B-Bos98] M. Bossert. *Kanalcodierung.* 2nd ed. Teubner Verlag, 1998 (cit. on p. 34).

[B-Bos99] M. Bossert. *Channel Coding for Telecommunications.* 1st ed. Wiley, 1999 (cit. on p. 34).

[B-Che09] L. Chen. *List Decoding of Reed–Solomon Codes and Algebraic–Geometric Codes: Decoding Principles, Complexity Reduction and an Engineering Insight into the System.* LAP Lambert Academic Publishing, 2009 (cit. on p. 42).

[B-GG03] J. Gathen and J. Gerhard. *Modern Computer Algebra.* Cambridge University Press, 2003 (cit. on p. 50).

[B-Gur04] V. Guruswami. *List Decoding of Error-Correcting Codes.* Vol. 3282. Lecture Notes in Computer Science. Springer, 2004 (cit. on pp. 62, 64).

[B-Gur07] V. Guruswami. *Algorithmic Results in List Decoding.* Now Publishers Inc, 2007 (cit. on p. 95).

[B-JH04] J. Justesen and T. Høholdt. *A Course in Error-Correcting Codes.* European Mathematical Society, 2004 (cit. on pp. 43, 60).

[B-KKS05] G. Kabatiansky, E. Krouk, and S. Semenov. *Error Correcting Coding and Security for Data Networks: Analysis of the Superchannel Concept.* 1st ed. Wiley, 2005 (cit. on p. 60).

[B-LC04] S. Lin and D. J. Costello. *Error Control Coding.* 2nd ed. Prentice Hall, 2004 (cit. on pp. 25, 33).

[B-Lin99] J. van Lint. *Introduction to Coding Theory.* Springer, 1999 (cit. on p. 107).

[B-Lip81] J. D. Lipson. *Elements of Algebra and Algebraic Computing.* Addison-Wesley Educational Publishers Inc, 1981 (cit. on pp. 50, 51).

[B-Moo05] T. K. Moon. *Error Correction Coding: Mathematical Methods and Algorithms.* Wiley-Interscience, 2005 (cit. on p. 60).

[B-MS88a] F. MacWilliams and N. Sloane. *The Theory of Error-Correcting Codes.* North Holland Publishing Co., 1988 (cit. on pp. 25, 28, 34, 50, 51, 107, 112, 119).

[B-PHB98b] V. S. Pless, W. C. Huffman, and R. A. Brualdi. *Handbook of Coding Theory, Volume 1 and 2.* 1st ed. North Holland, 1998 (cit. on p. 107).

[B-PW72] W. W. Peterson and E. J. Weldon. *Error-Correcting Codes.* 2nd ed. The MIT Press, 1972 (cit. on pp. 25, 107).

[B-PWBJ12] R. Pellikaan, X.-W. Wu, S. Bulygin, and R. Jurrius. *Error-Correcting Codes and Cryptology (Preliminary Version: 23 January 2012)*. 2012 (cit. on pp. 25, 107).

[B-Rot06] R. Roth. *Introduction to Coding Theory*. Cambridge University Press, 2006 (cit. on pp. 16, 34, 38, 46, 60).

[B-Wic99] S. B. Wicker. *Reed–Solomon Codes and Their Applications*. Wiley-IEEE Press, 1999 (cit. on p. 15).

InProceedings

[I-Ale02] M. Alekhnovich. "Linear Diophantine Equations over Polynomials and Soft Decoding of Reed–Solomon Codes". *IEEE Symposium on Foundations of Computer Science (FOCS)*. 2002, pp. 439–448 (cit. on pp. 65–67).

[I-AZ08] D. Augot and A. Zeh. "On the Roth and Ruckenstein Equations for the Guruswami–Sudan Algorithm". *IEEE International Symposium on Information Theory (ISIT)*. Toronto, Canada, 2008, pp. 2620–2624. DOI: 10.1109/ISIT.2008.4595466 (cit. on pp. 65, 67).

[I-BS97] S. Bezzateev and N. Shekhunova. "One Generalization of Goppa Codes". *IEEE International Symposium on Information Theory (ISIT)*. Ulm, Germany, 1997. DOI: 10.1109/ISIT.1997.613221 (cit. on p. 108).

[I-GSS00] V. Guruswami, A. Sahai, and M. Sudan. "Soft-decision Decoding of Chinese Remainder Codes". *IEEE Symposium on Foundations of Computer Science (FOCS)*. Redondo Beach, USA, 2000, pp. 159–168. DOI: 10.1109/SFCS.2000.892076 (cit. on p. 16).

[I-HB08] T. Høholdt and P. Beelen. "List Decoding Using Syndromes". Ed. by J. Hirschfeld, J. Chaumine, and R. Rolland. Vol. 5. Number Theory and Its Applications. World Scientific, 2008, pp. 315–331. DOI: 10.1142/9789812793430˙0016 (cit. on p. 67).

[I-Kö06] R. Kötter. "On Optimal Weight Assignmnents for Multivariate Interpolation List-Decoding". *IEEE Information Theory Workshop (ITW)*. Punta del Este, Uruguay, 2006, pp. 37–41. DOI: 10.1109/ITW.2006.1633777 (cit. on p. 42).

[I-KMVA03] R. Kötter, J. Ma, A. Vardy, and A. Ahmed. "Efficient Interpolation and Factorization in Algebraic Soft-Decision Decoding of Reed–Solomon Codes". *IEEE International Symposium on Information Theory (ISIT)*. Yokohama, Japan, 2003, pp. 365–365. DOI: 10.1109/ISIT.2003.1228381 (cit. on pp. 66, 97).

[I-Kra98] V. Krachkovsky. "Decoding of Parallel Reed–Solomon Codes with Applications to Product and Concatenated Codes". *IEEE International Symposium on Information Theory (ISIT)*. Cambridge, USA, 1998, p. 55. DOI: 10.1109/ISIT.1998.708636 (cit. on p. 38).

[I-KV03b] R. Kötter and A. Vardy. "A Complexity Reducing Transformation in Algebraic List Decoding of Reed–Solomon Codes". *IEEE Information Theory Workshop (ITW)*. Paris, France, 2003, pp. 10–13. DOI: 10.1109/ITW.2003.1216682 (cit. on pp. 42, 66, 97).

[I-LO06] K. Lee and M. O'Sullivan. "An Interpolation Algorithm using Gröbner Bases for Soft-Decision Decoding of Reed-Solomon Codes". *IEEE International Symposium on Information Theory (ISIT)*. Seattle, USA, 2006, pp. 2032–2036. DOI: 10.1109/ISIT.2006.261906 (cit. on pp. 65, 66).

[I-LS12] W. Li and V. R. Sidorenko. "On the Error-Erasure-Decoder of the Chinese Remainder Codes". *International Symposium on Problems of Redundancy in Information and Control Systems (RED)*. 2012, pp. 37–40. DOI: 10.1109/RED.2012.6338403 (cit. on p. 16).

[I-NZ13] J. S. R. Nielsen and A. Zeh. “Multi-Trial Guruswami–Sudan Decoding for Generalised Reed–Solomon Codes”. *International Workshop on Coding and Cryptography (WCC).* 2013 (cit. on p. 42).

[I-RR98] R. Roth and G. Ruckenstein. “Efficient Decoding of Reed–Solomon Codes Beyond Half the Minimum Distance”. *IEEE International Symposium on Information Theory (ISIT).* Cambridge, USA, 1998, p. 56. DOI: 10.1109/ISIT.1998.708637 (cit. on pp. 40, 67, 72).

[I-SS06] G. Schmidt and V. R. Sidorenko. “Multi-Sequence Linear Shift-Register Synthesis: The Varying Length Case”. *IEEE International Symposium on Information Theory (ISIT).* 2006, pp. 1738–1742. DOI: 10.1109/ISIT.2006.261652 (cit. on p. 60).

[I-SSB06] G. Schmidt, V. R. Sidorenko, and M. Bossert. “Decoding Reed–Solomon Codes Beyond Half the Minimum Distance using Shift-Register Synthesis”. *IEEE International Symposium on Information Theory (ISIT).* Seattle, USA, 2006, pp. 459–463. DOI: 10.1109/ISIT.2006.261711 (cit. on pp. 40, 67, 68, 70).

[I-Sud01] M. Sudan. “Coding Theory: Tutorial and Survey”. *IEEE Symposium on Foundations of Computer Science (FOCS).* Washington, USA, 2001, pp. 36–51. DOI: 10.1109/SFCS.2001.959879 (cit. on p. 15).

[I-ZB12b] A. Zeh and S. Bezzateev. “A New Error and Erasure Decoding Approach for Cyclic Codes”. *International Workshop on Algebraic and Combinatorial Coding Theory (ACCT).* Pomorie, Bulgaria, 2012 (cit. on pp. 107, 108).

[I-ZB12c] A. Zeh and S. Bezzateev. “Describing a Cyclic Code by Another Cyclic Code”. *IEEE International Symposium on Information Theory (ISIT).* Boston, USA, 2012, pp. 2896–2900. DOI: 10.1109/ISIT.2012.6284054 (cit. on pp. 107, 108).

[I-ZGB09] A. Zeh, C. Gentner, and M. Bossert. “Efficient List-Decoding of Reed–Solomon Codes with the Fundamental Iterative Algorithm”. *IEEE Information Theory Workshop (ITW).* Taormina, Italy, 2009, pp. 130–134. DOI: 10.1109/ITW.2009.5351241 (cit. on p. 67).

[I-ZWB11] A. Zeh, A. Wachter, and S. Bezzateev. “Efficient Decoding of Some Classes of Binary Cyclic Codes Beyond the Hartmann–Tzeng Bound”. *IEEE International Symposium on Information Theory (ISIT).* St. Petersburg, Russia, 2011, pp. 1017–1021. DOI: 10.1109/ISIT.2011.6033683 (cit. on pp. 107, 108).

[I-ZWB12a] A. Zeh, A. Wachter, and M. Bossert. “Unambiguous Decoding of Generalized Reed–Solomon Codes Beyond Half the Minimum Distance”. *International Zurich Seminar (IZS).* Zurich, Switzerland, 2012, pp. 63–66 (cit. on p. 67).

[I-ZWZGB13] A. Zeh, A. Wachter-Zeh, M. Gadouleau, and S. Bezzateev. “Generalizing Bounds on the Minimum Distance of Cyclic Codes Using Cyclic Product Codes”. *IEEE International Symposium on Information Theory (ISIT).* Istanbul, Turkey, 2013 (cit. on pp. 107, 108).

[I-McE03] R. J. McEliece. “On the Average List Size for the Guruswami–Sudan Decoder”. *International Symposium on Communications Theory and Applications (ISCTA).* Ambleside, UK, 2003 (cit. on p. 60).

Others

[O-ACS91] D. Augot, P. Charpin, and N. Sendrier. “The Minimum Distance of Some Binary Codes via the Newton’s Identities”. *EUROCODE ’90.* Ed. by G. Cohen and P. Charpin. Lecture Notes in Computer Science 514. Springer Berlin Heidelberg, 1991, pp. 65–73 (cit. on p. 107).

[O-AO09] M. B. Amorós and M. E. O'Sullivan. *The Berlekamp–Massey Algorithm and the Euclidean Algorithm: a Closer Link.* Tech. rep. 2009, pp. 1–12 (cit. on p. 50).

[O-Bar11] M. Barbier. "Décodage en liste et application à la sécurité de l'information". PhD thesis. Paris, France: Ecole Polytechnique ParisTech, 2011 (cit. on p. 16).

[O-BGMZ99] M. Bossert, H. Griesser, J. Maucher, and v. v. Zyablov. "Some Results on Generalized Concatenation of Block Codes". *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes.* Ed. by M. Fossorier, H. Imai, S. Lin, and A. Poli. Lecture Notes in Computer Science 1719. Springer Berlin Heidelberg, 1999, pp. 181–190 (cit. on p. 34).

[O-BH08a] P. Beelen and T. Høholdt. "A Syndrome Formulation of the Interpolation Step in the Guruswami–Sudan Algorithm". *Coding Theory and Applications.* Ed. by A. Barbero. Vol. 5228. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2008, pp. 20–32 (cit. on pp. 65, 67).

[O-BH08b] P. Beelen and T. Høholdt. "The Decoding of Algebraic Geometry Codes". World Scientific, 2008, pp. 49–98 (cit. on pp. 65, 66).

[O-BKY03] D. Bleichenbacher, A. Kiayias, and M. Yung. "Decoding of Interleaved Reed–Solomon Codes over Noisy Data". Ed. by J. C. M. Baeten, J. K. Lenstra, J. Parrow, and G. J. Woeginger. Vol. 2719. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2003, pp. 97–108 (cit. on p. 38).

[O-Bra10] K. Brander. "Interpolation and List Decoding of Algebraic Codes". PhD thesis. Kopenhagen, Denmark: Technical University of Denmark (DTU), 2010 (cit. on p. 67).

[O-Cha98] P. Charpin. "Open Problems on Cyclic Codes". *Handbook of Coding Theory.* Ed. by V. S. Pless and W. Huffman. Amsterdam: Elsevier, 1998, pp. 963–1064 (cit. on pp. 25, 107).

[O-Dum98] I. I. Dumer. "Concatenated Codes and Their Multilevel Generalizations". *Handbook of Coding Theory.* Ed. by V. S. Pless and W. Huffman. Amsterdam: Elsevier, 1998, pp. 1911–1988 (cit. on p. 33).

[O-Duu93] I. M. Duursma. "Decoding Codes from Curves and Cyclic Codes". PhD thesis. Eindhoven, Netherlands: Technische Universiteit Eindhoven, 1993 (cit. on p. 107).

[O-Eli57] P. Elias. *List Decoding for Noisy Channels.* Tech. rep. 335. Research Laboratory of Electronics Massachusetts Institute of Technology Cambridge, 1957 (cit. on p. 41).

[O-For66a] G. D. Forney. "Concatenated Codes". PhD thesis. Cambridge, USA: Massachusetts Institute of Technology (MIT), 1966 (cit. on pp. 16, 31, 32).

[O-Gao03] S. Gao. "A New Algorithm for Decoding Reed–Solomon Codes". *Communications, Information and Network Security.* Ed. by V. K. Bhargava, H. V. Poor, V. Tarokh, and S. Yoon. The Springer International Series in Engineering and Computer Science 712. Springer US, 2003, pp. 55–68 (cit. on pp. 40, 43).

[O-Gri02] H. Griesser. "On Soft Concatenated Decoding of Block Codes". PhD thesis. Ulm, Germany: University of Ulm, 2002 (cit. on p. 34).

[O-Hal12] J. I. Hall. "Notes in Coding Theory". 2012 (cit. on p. 35).

[O-Hey01] A. Heydtmann. "Decoding Algebraic Codes". PhD thesis. Kopenhagen, Denmark: Technical University of Denmark (DTU), 2001 (cit. on p. 50).

[O-Huf98] C. W. Huffman. "Codes and Groups". *Handbook of Coding Theory.* Ed. by V. S. Pless and W. Huffman. Amsterdam: Elsevier, 1998, pp. 1345–1440 (cit. on pp. 34, 35).

[O-Kö96b] R. Kötter. "On Algebraic Decoding of Algebraic-Geometric and Cyclic Codes". PhD thesis. Linköping, Sweden: Linköping University, 1996 (cit. on pp. 52, 66, 101, 107).

[O-Ksc03] F. R. Kschischang. "Product Codes". *Wiley Encyclopedia of Telecommunications.* John Wiley & Sons, Inc., 2003 (cit. on pp. 28, 29).

[O-KV00] R. Kötter and A. Vardy. "Algebraic Soft-Decision of Reed-Solomon Codes". 2000 (cit. on pp. 62, 97).

[O-MS88b] J. L. Massey and T. Schaub. "Linear Complexity in Coding Theory". *Coding Theory and Applications.* Ed. by G. Cohen and P. Godlewski. Vol. 311. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1988, pp. 19–32 (cit. on p. 107).

[O-Nie13] J. S. R. Nielsen. "List Decoding of Algebraic Codes". PhD thesis. Kopenhagen, Denmark: Technical University of Denmark (DTU), 2013 (cit. on p. 65).

[O-PHB98a] V. S. Pless, C. W. Huffman, and R. A. Brualdi. "An Introduction to Algebraic Codes". *Handbook of Coding Theory.* Ed. by V. S. Pless and W. Huffman. Amsterdam: Elsevier, 1998, pp. 3–139 (cit. on p. 34).

[O-Pra57] E. Prange. *Cyclic Error-Correcting Codes in Two Symbols.* Tech. rep. Air Force Cambridge Research Center, 1957 (cit. on pp. 16, 107).

[O-Qui12] G. Quintin. "On Some Guruswami–Sudan List Decoding Algorithms over Finite Rings". PhD thesis. Ecole Polytechnique ParisTech, Paris, France, 2012 (cit. on p. 15).

[O-Ruc01] G. Ruckenstein. "Error Decoding Strategies for Algebraic Codes". PhD thesis. Haifa, Israel: Technion, Israel Institute of Technology, 2001 (cit. on pp. 66, 67, 72, 85, 91).

[O-Sch07] G. Schmidt. "Algebraic Decoding Beyond Half the Minimum Distance Based on Shift-Register Synthesis". PhD thesis. Ulm, Germany: University of Ulm, 2007 (cit. on pp. 40, 54, 67).

[O-Sch88] T. Schaub. "A Linear Complexity Approach to Cyclic Codes". PhD thesis. Zürich, Switzerland: Swiss Federal Institute of Technology in Zurich, 1988 (cit. on p. 107).

[O-Sen11] C. Senger. "Generalized Minimum Distance Decoding with Arbitrary Error/Erasure Tradeoff". PhD thesis. Ulm, Germany: University of Ulm, 2011 (cit. on p. 42).

[O-Sud00] M. Sudan. "List Decoding: Algorithms and Applications". *Theoretical Computer Science: Exploring New Frontiers of Theoretical Informatics.* Ed. by J. v. Leeuwen, O. Watanabe, M. Hagiya, P. D. Mosses, and T. Ito. Lecture Notes in Computer Science 1872. Springer Berlin Heidelberg, 2000, pp. 25–41 (cit. on p. 15).

[O-WB86] L. R. Welch and E. R. Berlekamp. "Error Correction for Algebraic Block Codes". Pat. 4633470. 1986 (cit. on pp. 16, 40, 43).

[O-Woz58] J. M. Wozencraft. *List Decoding.* Technical Report. 1958, pp. 90–95 (cit. on p. 41).

[O-ZJ12] A. Zeh and T. Jerkovits. *www.boundtables.org.* 2012 (cit. on pp. 107, 143).

[O-McE78] R. J. McEliece. *A Public-Key Cryptosystem Based on Algebraic Coding Theory.* Technical report. 1978, pp. 114–116 (cit. on p. 15).