

Symmetric Models for Syndrome Decoding

Elisa Gorla¹ and Simone Trebbiani¹

¹Institut de mathématiques, Université de Neuchâtel

Abstract. This paper introduces a new polynomial model for the exact variant of the Syndrome Decoding Problem (SDP) in the binary case. The model is based on elementary symmetric polynomials. We estimate the computational complexity of solving the corresponding polynomial system by establishing bounds on the degree of regularity and on the solving degree of the ideal associated to the model. The complexity estimate is lower than for previous polynomial models. We also provide a variant of the model whose complexity depends directly on the specific instance of the SDP and is lower than for the first model. Finally, we discuss how to apply our approach to solve other variants of the SDP.

Keywords. Syndrome Decoding Problem · Code-based cryptography · Solving degree · Elementary symmetric polynomials · Degree of regularity

Introduction

Over the past few years, there has been growing interest in new branches of cryptography, driven by the search for problems capable of resisting attacks from the emerging quantum computing. Several proposed schemes are based on problems from coding theory, thus giving rise to *code-based cryptography*. For example, three of them (BIKE, Classic McEliece, and HQC) were included by the *National Institute of Standards and Technology* in the list of candidates in the fourth and final round of the Post-Quantum Cryptography Standardization Process¹. HQC was one of two key-encapsulation mechanisms selected for standardization². Two more code-based schemes

¹<https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-4-submissions>

²<https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization/selected-algorithms>

(CROSS and LESS) were included in the list of second round candidates in the Additional Digital Signature Schemes PQC Standardization Process³, although they were not selected for the third round [Ala+26].

For this reason, assessing the hardness of the underlying mathematical problems becomes of central importance. One can do so by estimating the computational complexity of the best known algorithms or by exploring new attack strategies. One such strategy is to model the problem as a polynomial system and then to use Gröbner bases techniques to solve it. For the *Syndrome Decoding Problem* (SDP), a first step in this direction was taken in [MPS23], where a polynomial model for an instance of the SDP was proposed. A further step was taken in [Cam+25], where the original model was improved and an estimate of its computational complexity was given. In the same work, the authors also provided a polynomial model for the SDP over larger fields.

Our contributions & Structure of the paper. In Section 1 we recall some preliminaries on the Syndrome Decoding Problem and the Polynomial System Solving Problem and give a brief overview of the previous polynomial models for the SDP. In Section 2.1 we propose a new polynomial model based on elementary symmetric polynomials for the Exact Syndrome Decoding Problem over the binary field, and we investigate its computational complexity, obtaining a bound that is lower than that of [Cam+25] (Theorem 2.7). In Section 2.2 we then propose a variant of the model, for which the complexity of solving the polynomial system depends more clearly on the specific instance of the SDP and is even lower than the previous result (Corollary 2.13). In Section 2.3, we discuss a linear algebra problem that arises from the variant model. Finally, in the last section we discuss how to use the ideas of the previous sections to solve related problems, such as the Bounded SDP (Section 3.1), the non-binary version of the SDP (Section 3.2), and the Regular SDP (Section 3.3).

1 Preliminaries

1.1 Notation

Let $\mathbb{F}_q$ be a finite field with $q \geq 2$ elements and R_n the polynomial ring $\mathbb{F}_q[x_1, \dots, x_n]$ with $n \geq 1$ variables and coefficients in $\mathbb{F}_q$. $R_{n,d}$ denotes the vector space of homogeneous polynomials of R_n of degree d . $(f_1, \dots, f_j)$

³<https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures>

denotes the ideal generated by the polynomials $f_1, \dots, f_j$.

Let $f \in R_n$ be a polynomial and let f^{top} denote the homogeneous component of highest degree of f . If $S = \{f_1, \dots, f_j\}$ is a set of polynomials, $S^{\text{top}} = \{f_1^{\text{top}}, \dots, f_j^{\text{top}}\}$ is the set of the homogeneous components of highest degree of the elements of S .

Let $[n]$ denote the set $\{1, \dots, n\}$ and S_n its group of permutations.

For $\alpha \subseteq [n]$, let $x_\alpha = x_{\alpha_1} \cdots x_{\alpha_i}$ be the squarefree monic monomial in R_n that is the product of the variables indexed by the elements of α .

Given a set A , the cardinality of A is denoted by $|A|$. So, for any given subset $\alpha \subseteq [n]$ as above, we have $\deg(x_\alpha) = \deg(x_{\alpha_1} \cdots x_{\alpha_i}) = |\alpha|$.

Given $n, m \geq 1$, let $\mathbb{F}_q^{m \times n}$ be the set of matrices with m rows, n columns and coefficients in $\mathbb{F}_q$. We denote the determinant and the rank of a matrix A by $\det(A)$ and $\text{rk}(A)$, respectively.

1.2 The Syndrome Decoding Problem

Before defining the Syndrome Decoding Problem, we recall the usual notation for linear codes. Given integers $n \geq 2$ and $k \in [n]$, a $[n, k]$ -linear code $\mathbf{C}$ is a k -dimensional subspace of $\mathbb{F}_q^n$, where n is its *length* and k its *dimension*. An element of $\mathbf{C}$ is called a *codeword*, and we define the *Hamming weight* of a codeword $c \in \mathbf{C}$, denoted by $\text{wt}(c)$, as the number of its non-zero entries. The *minimum distance* of $\mathbf{C}$, denoted by $d(\mathbf{C})$, is the Hamming weight of the smallest non-zero codeword of $\mathbf{C}$.

Given a $[n, k]$ -linear code $\mathbf{C}$, a full rank matrix $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ is a *parity check matrix* of $\mathbf{C}$ if $\mathbf{H} \cdot c^\top = 0$ holds for every codeword c of $\mathbf{C}$. Given a parity check matrix $\mathbf{H}$ of $\mathbf{C}$, the *syndrome* of a vector $e \in \mathbb{F}_q^n$ is $s = \mathbf{H} \cdot e^\top \in \mathbb{F}_q^{n-k}$.

We can now describe the Syndrome Decoding Problem (SDP) in two variants: Given $n \geq 2$ and $k \in [n]$, a parity check matrix $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ of a $[n, k]$ -linear code $\mathbf{C}$, let s be a vector in $\mathbb{F}_q^{n-k}$ and t be an integer such that $0 \leq t \leq n$.

Problem 1 ((B)SDP: (Bounded) Syndrome Decoding Problem). Find a vector $e \in \mathbb{F}_q^n$ such that $\mathbf{H} \cdot e^\top = s^\top$ and $\text{wt}(e) \leq t$.

Problem 2 (ESDP: Exact weight Syndrome Decoding Problem). Find a vector $e \in \mathbb{F}_q^n$ such that $\mathbf{H} \cdot e^\top = s^\top$ and $\text{wt}(e) = t$.

A triple $(\mathbf{H}, s, t)$ as above is an *instance* of the SDP.

The SDP is known to be NP-complete [BMT78] and believed to be hard on average. For this reason, some code-based cryptosystems base their se-

curity on the hardness of the SDP, most notably the McEliece cryptosystem [McE78] and its later variants, such as the one [Cla22] proposed at the NIST competition [Ala+25]. In the McEliece settings, the public key is a seemingly random code $\mathbf{C}'$, which is a scrambled version of a t -error correcting code $\mathbf{C}$, which is the private key. The encryption of a message m is its encoding through $\mathbf{C}'$, with the addition of some noise e of weight t . The original message can be recovered thanks to the error correction properties of $\mathbf{C}$. An equivalent version based directly on the parity check matrix, known as the Niederreiter scheme [Nie86], can be interpreted by an attacker as an instance of the SDP.

The maximum weight of a uniquely correctable error e depends on the properties of $\mathbf{C}$. More precisely, if the target weight t is chosen such that $t \leq \left\lfloor \frac{d(\mathbf{C})-1}{2} \right\rfloor$, the uniqueness of the solution and, therefore, the recovery of the message is guaranteed. An SDP-based cryptographic scheme is generally constructed so that the weight t of the inserted error is maximized, while still guaranteeing unique decoding.

1.3 The Polynomial System Solving Problem

In this paper, we devise a polynomial system whose solutions coincide with the solutions of the Syndrome Decoding Problem. By doing so, we reduce the SDP to the problem of solving a polynomial system. The related problem is therefore:

Problem 3 (PoSSo: Polynomial System Solving). Given a set of r polynomials $F = \{f_1, \dots, f_r\} \subseteq R_n$, find a solution of the system $F = 0$, that is, find a vector $y = (y_1, \dots, y_n) \in \mathbb{F}_q^n$ such that $f_1(y) = \dots = f_r(y) = 0$.

PoSSo was proven to be NP-complete in the case $q = 2$ in [FY79, p. 28]. In addition, it is widely believed to be an average-case difficult problem. This is witnessed by the fact that a whole branch of post-quantum cryptography, called *multivariate cryptography*, relies on its hardness.

The generic approach for solving the PoSSo problem is via *Gröbner bases*. Gröbner bases were introduced by Buchberger in [Buc06] and may be regarded a “nice” system of generators of the ideal (F) . Over a finite field and for a system that has finitely many solutions over any field extension, the solutions can be computed in polynomial time from the reduced lexicographic Gröbner basis of the system. We refer the interested reader to [KR00] for a mathematical treatment of Gröbner bases and to [CG21] for a discussion on how to use them in cryptography. In [CG21] it is argued that, for systems

that are currently relevant for cryptography, the complexity of computing their solutions using Gröbner bases is dominated by the complexity of computing a degree-reverse lexicographic (**degrevlex**) Gröbner basis. This will therefore be the focus of the rest of our discussion.

There are many algorithms for the computation of Gröbner bases, such as F4 [Fau99], F5 [Fau02], and XL [Cou+00] and its variants [Buc+09; Moh+08], which all rely on Gaussian elimination on the *Macaulay matrices* associated with the polynomial system $F = \{f_1, \dots, f_r\}$. The Macaulay matrix $M_{\leq d}$ of degree d of F is defined as follows: It has columns indexed by the monic monomials x_β of degree $\leq d$ (ordered according to a specific term order), rows indexed by the polynomials $x_\alpha f_i$, where x_α is a monic monomial such that $\deg(x_\alpha f_i) \leq d$, and its entry $((\alpha, i), \beta)$ is the coefficient of the monomial x_β in the polynomial $x_\alpha f_i$. The least d such that Gaussian elimination on the Macaulay matrix $M_{\leq d}$ produces a **degrevlex** Gröbner basis of F is the *solving degree* of F , denoted by $d_{\text{sol}}(F)$. Therefore, up to constant factors, the complexity of computing a **degrevlex** Gröbner basis is asymptotically dominated by the complexity of performing Gaussian elimination [GV13, pp. 111–118] on the Macaulay matrix of degree $d_{\text{sol}}(F)$. Since the matrix has $\sum_{i=1}^r \binom{n+d_{\text{sol}}(F)-\deg(f_i)}{d_{\text{sol}}(F)-\deg(f_i)}$ rows and $\binom{n+d_{\text{sol}}(F)}{d_{\text{sol}}(F)}$ columns, the complexity is bounded from above by

$$O\left(\binom{n+d_{\text{sol}}(F)}{d_{\text{sol}}(F)}^\omega\right)$$

for sufficiently large $n, d_{\text{sol}}(F)$, where $2 \leq \omega < 3$ is the *matrix multiplication exponent* [GG13, pp. 313–316].

The algorithms mentioned above employ their own strategies to reduce the number of operations required to compute a Gröbner basis, but we consider this rough estimate to keep the discussion as general as possible.

The issue is that the solving degree of a system F may be difficult to compute or estimate: for this reason, it has been related to several algebraic invariants (see e.g. [CG23]), such as the *degree of regularity*, introduced by Bardet, Faugère, and Salvy in [FBS04]. The degree of regularity of F is defined as

$$d_{\text{reg}}(F) = \min\{d \in \mathbb{N} \mid (F^{\text{top}}) \cap R_{n,d} = R_{n,d}\}.$$

This invariant is often easier to compute than the solving degree. The next result by Salizzoni relates the degree of regularity and the solving degree. It allows us to focus on the degree of regularity in the sequel.

Theorem 1.1 ([Sal25, Theorem 1]). *Let $F = \{f_1, \dots, f_r\}$ be a family of polynomials and let σ be a degree-compatible term order. Then,*

$$d_{\text{sol}}(F) \leq \max\{d_{\text{reg}}(F) + 1, \deg(f_1), \dots, \deg(f_r)\}.$$

Since `degrevlex` is degree-compatible, we can use this result to estimate the solving degree, and thus the complexity of solving polynomial systems. This will allow us to compare different polynomial models.

1.4 Previous polynomial models for SDP

There are several approaches to solve the SDP, one of the most prominent being *Information Set Decoding*. The first algorithm was proposed by Prange [Pra62], followed by several subsequent improvements and variants (for more details, see e.g. [WGR24]).

Since the focus of this work is to provide a polynomial model to solve the SDP, we focus on known polynomial models. The first version by Meneghetti, Pellegrini, and Sala in [MPS23] was later improved by Caminata, Cartor, Meneghetti, Mora, and Pellegrini in [Cam+25]. Given an instance $(\mathbf{H}, s, t)$ of the SDP over $\mathbb{F}_2$, [MPS23] uses two different types of encoding variables: the variables x_i are the entries of a solution vector x of the SDP, while the variables $y_{i,j}$ are the digits of the binary expansion of the Hamming weight of x . The model involves the following polynomial equations:

- the *parity check encoding*, given by the linear equations $\mathbf{H} \cdot x^\top = s$,
- the *Hamming weight computation encoding*, which forces the entries of the vector $y_i = (y_{i,1}, \dots, y_{i, \lfloor \log_2 n \rfloor + 1})$ to be the digits of the binary expansion of the Hamming weight of the truncated vector $(x_1, \dots, x_i)$, for $i = 1, \dots, n$,
- the *weight constraint encoding*, which enforces the constraint $\text{wt}(x) \leq t$ or $\text{wt}(x) = t$;
- the *field equations* $x_i^2 = x_i$ and $y_{i,j}^2 = y_{i,j}$.

The authors of [Cam+25] improve the model by lowering the degrees of the equations involved in the Hamming weight computation encoding and in the weight constraint encoding. They are able to produce a system whose equations have degree at most two. Moreover, after some linearization steps, the resulting system involves fewer variables and fewer equations than the original.

They also study the complexity of their model applied to the ESDP over $\mathbb{F}_2$. Combining the result of [Cam+25, Remark 8] on the degree of regularity of the polynomial system $S_{\text{QM}} = 0$ and Theorem 1.1, one obtains the following result:

Theorem 1.2. *The solving degree of S_{QM} is bounded from above by*

$$d_{\text{sol}}(S_{\text{QM}}) \leq \left\lceil \frac{n-1}{2} \right\rceil + \lfloor \log_2 t \rfloor \left\lceil \frac{n-2}{2} \right\rceil + 2.$$

Moreover, the authors of [Cam+25] provide a variant of their system for the SDP that is valid for any $q \geq 2$. We briefly discuss it in Section 3.2. The next section focuses on the binary case, which is the most relevant from a computational and cryptographic point of view. For example, it is the parameter choice in Classic McEliece [Cla22].

2 New models for the ESDP via elementary symmetric polynomials

In this section, we propose two models for the ESDP that use elementary symmetric polynomials. Our goal is to produce a polynomial system that can be solved with lower complexity than the one in Theorem 1.2. We focus on the binary case and let $q = 2$ and $R_n = \mathbb{F}_2[x_1, \dots, x_n]$.

2.1 A first model for the ESDP

We start by recalling the definition of elementary symmetric polynomial. A polynomial in R_n is *symmetric* if it is invariant under any permutation of the variables $x_1, \dots, x_n$, while an ideal is *symmetric* if it contains the polynomials obtained from any polynomial in the ideal by permuting the variables.

Definition 2.1. Let $n \geq 1$ and $j \in [n]$. The *elementary symmetric polynomial* of degree j in the variables $x_1, \dots, x_n$ is

$$e_{n,j}(x) = \sum_{\alpha \subseteq [n], |\alpha|=j} x_\alpha = \sum_{1 \leq \alpha_1 < \dots < \alpha_j \leq n} x_{\alpha_1} \cdots x_{\alpha_j} \in R_{n,j}.$$

The same definition applies over any field. It is useful to also define $e_{n,0} = 1$ and $e_{n,j} = 0$ for every $j > n$.

We start by recalling Lucas' Theorem [Luc91, pp. 417–420], which allows one to compute binomial coefficients modulo a prime number.

Theorem 2.1 (Lucas' Theorem). *Let p be a prime, and let*

$$\begin{aligned} m &= m_0 + m_1p + m_2p^2 + \cdots + m_rp^r, \\ n &= n_0 + n_1p + n_2p^2 + \cdots + n_rp^r \end{aligned}$$

such that $0 \leq m_i, n_i < p$ for every $i = 0, \dots, r$ (named p -adic expansion of m and n , or just binary expansion if $p = 2$). Then

$$\binom{m}{n} = \prod_{i=0}^r \binom{m_i}{n_i} \pmod{p}.$$

Next, we show how to express the weight of a vector via the values of the elementary symmetric polynomials at its entries.

Lemma 2.2. *Let $n \geq 1$ and $r = \lfloor \log_2 n \rfloor$. For $0 \leq t \leq n$, let $t = t_0 + t_12 + \cdots + t_r2^r$ be the binary expansion of t . For a vector $y \in \mathbb{F}_2^n$, the following conditions are equivalent:*

- 1) $\text{wt}(y) = t$,
- 2) $e_{n,j}(y) = \binom{t}{j}$ for all $j = 1, \dots, n$,
- 3) $e_{n,2^i}(y) = t_i$ for all $i = 0, \dots, r$.

Proof. "1) $\Rightarrow$ 2)": Let $y = (y_1, \dots, y_n) \in \mathbb{F}_2^n$ be a vector of weight t and $T \subseteq [n]$ be the set of the indexes corresponding to the non-zero entries of y . Then

$$\begin{aligned} e_{n,j}(y) &= \sum_{1 \leq \alpha_1 < \dots < \alpha_j \leq n} y_{\alpha_1} \cdots y_{\alpha_j} = 0 + \sum_{\substack{1 \leq \alpha_1 < \dots < \alpha_j \leq n, \\ \alpha_1, \dots, \alpha_j \in T}} 1 = \\ &= \sum_{\alpha \subseteq T, |\alpha|=j} 1 = |\{\alpha \subseteq T \mid |\alpha| = j\}| = \binom{t}{j} \pmod{2}. \end{aligned}$$

"2) $\Rightarrow$ 3)": By Theorem 2.1, for every $i = 0, \dots, r$ we have

$$e_{n,2^i}(y) = \binom{t}{2^i} = \binom{t_i}{1} \cdot \prod_{\ell=0, \ell \neq i}^r \binom{t_\ell}{0} = t_i,$$

and thus the thesis.

"3) $\Rightarrow$ 1)": Let $y \in \mathbb{F}_2^n$ be such that $e_{n,2^i}(y) = t_i$ for all $i = 0, \dots, r$ and suppose that $\text{wt}(y) = t'$. Thanks to "1) $\Rightarrow$ 2)", we have that $e_{n,j}(y) = \binom{t'}{j}$ for all $j = 1, \dots, n$. Therefore, for every $i \in \{0, \dots, r\}$, one has

$$t_i = e_{n,2^i}(y) = \binom{t'}{2^i} = t'_i$$

where the last equality follows from the proof of "2) $\Rightarrow$ 3)". We conclude that $t = t'$. □

This result allows us to express the weight of a vector via the evaluations of the elementary symmetric polynomials at its entries, and therefore, to produce a new model for the ESDP over $\mathbb{F}_2$.

Given a parity check matrix $\mathbf{H} = (h_{i,j})_{i=1, \dots, n-k}^{j=1, \dots, n}$, a syndrome $s = (s_i)_{i=1, \dots, n-k}$ and a target weight $0 \leq t \leq n$ with binary expansion $t = \sum_{\ell=0}^r t_\ell 2^\ell$, where $r = \lfloor \log_2 n \rfloor$, the corresponding instance $(\mathbf{H}, s, t)$ of the ESDP can be modelled via the following sets of polynomials in R_n :

– **parity check encoding:**

$$S_1 = \left\{ \sum_{j=1}^n h_{i,j} x_j + s_i \mid i = 1, \dots, n-k \right\},$$

– **finite field equations:**

$$S_2 = \{x_j^2 + x_j \mid j = 1, \dots, n\},$$

– **weight constraint encoding:**

$$S_3 = \{e_{n,2^i}(x) + t_i \mid i = 0, \dots, r\}, \quad r = \lfloor \log_2 n \rfloor.$$

Let $S = S_1 \cup S_2 \cup S_3$. The solutions of the polynomial system $S = 0$ coincide with the solutions of the instance $(\mathbf{H}, s, t)$ of the ESDP.

We now estimate the computational complexity of this model. As discussed in Section 1.3, it suffices to determine the degree of regularity of S .

For that, we consider $S^{\text{top}} = S_1^{\text{top}} \cup S_2^{\text{top}} \cup S_3^{\text{top}}$, where

$$S_1^{\text{top}} = \left\{ \sum_{j=1}^n h_{i,j} x_j \mid i = 1, \dots, n-k \right\},$$

$$S_2^{\text{top}} = \{x_j^2 \mid j = 1, \dots, n\}, \quad S_3^{\text{top}} = \{e_{n,2^i}(x) \mid i = 0, \dots, r\}.$$

Notice that S^{top} is independent of the target weight t . Moreover, S_2^{top} and S_3^{top} do not depend on the specific instance $(\mathbf{H}, s, t)$ of the ESDP, but only depend on the parameter n . For this reason, we first focus on the degree of regularity of $S_2^{\text{top}} \cup S_3^{\text{top}}$.

Notation 2.1. Let $Q_n = (x_1^2, \dots, x_n^2) \subseteq R_n$ be the ideal generated by the squares of the variables. Let $r = \lfloor \log_2 n \rfloor$ and let $I_n = (e_{n,2^i} \mid i = 0, \dots, r) + Q_n \subseteq R_n$ be generated by the polynomials in S_2^{top} and S_3^{top} .

Next we recall some properties of the elementary symmetric polynomials $e_{n,j} \in R_n$. It is easy to check that they can be computed recursively for $n > 1$ using the formulas

$$e_{n,j} = \begin{cases} e_{n-1,1} + x_n & \text{if } j = 1, \\ e_{n-1,j} + x_n \cdot e_{n-1,j-1} & \text{if } 2 \leq j \leq n-1, \\ e_{n-1,n-1} \cdot x_n & \text{if } j = n, \end{cases} \quad (1)$$

where $e_{n-1,j} = \sum_{\alpha \subseteq [n-1], |\alpha|=j} x_\alpha$ is the elementary symmetric polynomial of degree j in the first $n-1$ variables $x_1, \dots, x_{n-1}$ and $e_{1,1}$ is equal to x_1 .

We also have the following

Lemma 2.3. Let $j, j' \in [n]$, with binary expansions $j = j_0 + j_1 2 + \dots + j_r 2^r$ and $j' = j'_0 + j'_1 2 + \dots + j'_r 2^r$. Then

$$e_{n,j} \cdot e_{n,j'} = \begin{cases} 0 & \text{if } j + j' > n \text{ or there exists } \ell \text{ such that } j_\ell = 1 = j'_\ell, \\ e_{n,j+j'} & \text{else,} \end{cases}$$

modulo Q_n .

Proof. By definition

$$e_{n,j} \cdot e_{n,j'} = \left(\sum_{\alpha \subseteq [n], |\alpha|=j} x_\alpha \right) \left(\sum_{\beta \subseteq [n], |\beta|=j'} x_\beta \right) = \sum_{\substack{\alpha, \beta \subseteq [n], \\ |\alpha|=j, |\beta|=j'}} x_\alpha x_\beta.$$

If $j + j' > n$, then x_α and x_β share a variable for every $\alpha, \beta \subseteq [n]$.
Therefore $x_\alpha x_\beta \in Q_n$ and $e_{n,j} \cdot e_{n,j'} = 0$ modulo Q_n .

If $j + j' \leq n$, then modulo Q_n

$$\begin{aligned}
e_{n,j} \cdot e_{n,j'} &= \sum_{\substack{\alpha, \beta \subseteq [n], \\ |\alpha|=j, |\beta|=j'}} x_\alpha x_\beta = \sum_{\substack{\alpha, \beta \subseteq [n], \alpha \cap \beta = \emptyset, \\ |\alpha|=j, |\beta|=j'}} x_\alpha x_\beta = \\
&= \sum_{\gamma \subseteq [n], |\gamma|=j+j'} \left(\sum_{\substack{\alpha, \beta \subseteq [n], |\alpha|=j, |\beta|=j', \\ \alpha \cap \beta = \emptyset, \alpha \cup \beta = \gamma}} x_\alpha x_\beta \right) = \\
&= \sum_{\gamma \subseteq [n], |\gamma|=j+j'} \left(|\{(\alpha, \beta) \mid \alpha, \beta \subseteq [n], |\alpha|=j, |\beta|=j', \alpha \cap \beta = \emptyset, \alpha \cup \beta = \gamma\}| \cdot x_\gamma \right) = \\
&= \sum_{\gamma \subseteq [n], |\gamma|=j+j'} (|\{\alpha \subseteq [n] \mid |\alpha|=j, \alpha \subseteq \gamma\}| \cdot x_\gamma) = \\
&= \sum_{\gamma \subseteq [n], |\gamma|=j+j'} \binom{j+j'}{j} x_\gamma = \binom{j+j'}{j} e_{n,j+j'}.
\end{aligned}$$

By Lucas' Theorem (Theorem 2.1)

$$\binom{j+j'}{j} = \prod_{i=0}^r \binom{j_i+j'_i}{j_i}.$$

Therefore, if there exists ℓ such that $j_\ell = 1 = j'_\ell$, then $\binom{j_\ell+j'_\ell}{j_\ell} = \binom{0}{1} = 0$, so $\binom{j+j'}{j} = 0$ and $e_{n,j} \cdot e_{n,j'} = 0$. If no such ℓ exists, then $\binom{j_i+j'_i}{j_i} = 1$ for all i , hence $\binom{j+j'}{j} = 1$ and $e_{n,j} \cdot e_{n,j'} = e_{n,j+j'}$. □

The lemma implies that

$$e_{n,j} = e_{n,j_0+j_1 2+\dots+j_r 2^r} = e_{n, \sum_{i:j_i=1} 2^i} = \prod_{i:j_i=1} e_{n, 2^i}, \quad (2)$$

so $e_{n,j} \in I_n$ for all $j \in [n]$. It follows that

$$I_n = (e_{n,j} \mid j = 1, \dots, n) + Q_n.$$

We now want to compute the degree of regularity of I_n . Since I_n is generated by homogeneous polynomials, it is a homogeneous ideal, and we

can write it as $I_n = I_{n,0} \oplus I_{n,1} \oplus \cdots \oplus I_{n,j} \oplus \cdots$, where $I_{n,j} = I_n \cap R_{n,j}$ is the $\mathbb{F}_2$ -vector space of homogeneous polynomials of I_n of degree j . Therefore, the quotient $A_n = R_n/I_n$ is graded via $A_n = A_{n,0} \oplus A_{n,1} \oplus \cdots \oplus A_{n,j} \oplus \cdots$, where $A_{n,j} = R_{n,j}/I_{n,j}$ is an $\mathbb{F}_2$ -vector space. Let $a_{n,j} = \dim_{\mathbb{F}_2} A_{n,j}$.

In order to compute $d_{\text{reg}}(I_n)$, we need to determine the largest j such that $a_{n,j} \neq 0$. We prove a stronger result, some of which may be useful in the sequel. We start with a preliminary lemma.

Lemma 2.4. *Let $n \geq 3$ and $R_n = \mathbb{F}[x_1, \dots, x_n]$ be the polynomial ring in n variables over a field $\mathbb{F}$. Let $I \subseteq R_n$ be an ideal such that $Q_n = (x_1^2, \dots, x_n^2) \subseteq I$. For every $3 \leq m \leq n$ and $2 \leq j \leq m-1$,*

$$\text{if } e_{m,j-1} \text{ and } e_{m,j} \in I, \text{ then } e_{m-1,j} \in I.$$

Proof. Using (1) we have

$$e_{m,j} = e_{m-1,j} + x_m e_{m-1,j-1} \quad \text{and} \quad e_{m,j-1} = e_{m-1,j-1} + x_m e_{m-1,j-2}.$$

Since $e_{m-1,0} = 1$, we have

$$\begin{aligned} e_{m-1,j} &= e_{m,j} - x_m e_{m-1,j-1} = e_{m,j} - x_m(e_{m,j-1} - x_m e_{m-1,j-2}) = \\ &= \underbrace{e_{m,j}}_{\in I} - x_m \underbrace{e_{m,j-1}}_{\in I} + \underbrace{x_m^2}_{\in Q_n \subseteq I} e_{m-1,j-2} \in I. \end{aligned}$$

□

The next proposition follows from repeatedly applying Lemma 2.4.

Proposition 2.5. *Let $n \geq 1$ and let $R_n = \mathbb{F}[x_1, \dots, x_n]$ be the polynomial ring in n variables over a field $\mathbb{F}$. Let $I \subseteq R_n$ be an ideal such that $Q_n = (x_1^2, \dots, x_n^2) \subseteq I$. For every $0 \leq u \leq n-1$,*

$$\text{if } e_{n,u+1}, \dots, e_{n,n} \in I, \text{ then } I \cap R_{n,j} = R_{n,j} \text{ for } j \geq \left\lfloor \frac{n+u}{2} \right\rfloor + 1.$$

In particular,

$$d_{\text{reg}}(I) \leq \left\lfloor \frac{n+u}{2} \right\rfloor + 1.$$

Proof. The cases $n = 1$ and $n = 2$ can be checked directly. If $n \geq 3$, then we can apply the previous lemma recursively to obtain

Step 0	$e_{n,u+1}$	$e_{n,u+2}$	$e_{n,u+3}$	$\dots$	$e_{n,n-2}$	$e_{n,n-1}$	$e_{n,n}$	$\in I$
Step 1	$/$	$e_{n-1,u+2}$	$e_{n-1,u+3}$	$\dots$	$e_{n-1,n-2}$	$e_{n-1,n-1}$	$/$	$\in I$
Step 2	$/$	$/$	$e_{n-2,u+3}$	$\dots$	$e_{n-2,n-2}$	$/$	$/$	$\in I$
$\vdots$	$/$	$/$	$/$	$\vdots$	$/$	$/$	$/$	$\in I$

For $u + s + 1 \leq n - s$, that is for $2s \leq n - u - 1$, at Step s we have $e_{n-s, u+s+1}, \dots, e_{n-s, n-s} \in I$. Choosing the largest possible value for s , i.e. for $s = \lfloor \frac{n-u-1}{2} \rfloor$, we get that $e_{\lfloor \frac{n+u}{2} \rfloor + 1, \lfloor \frac{n+u}{2} \rfloor + 1} \in (e_{n, u+1}, \dots, e_{n, n}) + Q_n \subseteq I$. Since $(e_{n, u+1}, \dots, e_{n, n}) + Q_n$ is symmetric, any monomial obtained from $e_{\lfloor \frac{n+u}{2} \rfloor + 1, \lfloor \frac{n+u}{2} \rfloor + 1} = x_1 \cdots x_{\lfloor \frac{n+u}{2} \rfloor + 1}$ by permuting the variables still belongs to it, and thus to I . It follows that all squarefree monomials of degree $\lfloor \frac{n+u}{2} \rfloor + 1$ belong to I . Since $Q_n \subseteq I$ contains all non-squarefree monomials, every monomial of degree $\geq \lfloor \frac{n+u}{2} \rfloor + 1$ belongs to I , from which the thesis follows. $\square$

We are now ready to compute the degree of regularity of I_n .

Proposition 2.6. *For $n \geq 1$, let $I_n = (e_{n,1}, \dots, e_{n,n}) + (x_1^2, \dots, x_n^2) \subseteq R_n$ and $A_n = R_n/I_n$. The dimension $a_{n,j}$ of $A_{n,j}$ as an $\mathbb{F}_2$ -vector space is*

$$a_{n,j} = \begin{cases} \binom{n}{j} - \binom{n}{j-1} & \text{if } 0 \leq j \leq \lfloor \frac{n}{2} \rfloor, \\ 0 & \text{if } j \geq \lfloor \frac{n}{2} \rfloor + 1, \end{cases}$$

where $\binom{n}{-1} = 0$. Since $\binom{n}{j} - \binom{n}{j-1} \neq 0$ for all $j \leq \lfloor \frac{n}{2} \rfloor$, this implies that

$$d_{\text{reg}}(I_n) = \left\lfloor \frac{n}{2} \right\rfloor + 1.$$

Proof. For $j \geq 0$ one has

$$A_{n,j} = \frac{R_{n,j}}{I_{n,j}} \cong \frac{R_{n,j}/Q_{n,j}}{I_{n,j}/Q_{n,j}}$$

hence

$$a_{n,j} = \dim_{\mathbb{F}_2} \frac{R_{n,j}}{Q_{n,j}} - \dim_{\mathbb{F}_2} \frac{I_{n,j}}{Q_{n,j}},$$

where $Q_{n,j} = Q_n \cap R_{n,j}$. Since $R_{n,j}/Q_{n,j}$ is generated by all squarefree monomials of degree j ,

$$\dim_{\mathbb{F}_2} \frac{R_{n,j}}{Q_{n,j}} = \binom{n}{j}.$$

The quotient $I_{n,j}/Q_{n,j}$ is generated by all products of the form $x_\alpha \cdot e_{n,m}$, where $1 \leq m \leq j$ and x_α is a squarefree monomial of degree $j - m$. However, in general, these generators are not linearly independent. In order to compute the dimension of $I_{n,j}/Q_{n,j}$, we consider an *incidence matrix* $M_{n,j}$ where:

- the rows are indexed by the generators $x_\alpha \cdot e_{n,m}$ described before,
- the columns are indexed by the squarefree monomials of degree j in n variables,
- the entry in row $x_\alpha \cdot e_{n,m}$ and column x_β is the coefficient of the monomial x_β in the polynomial $x_\alpha \cdot e_{n,m}$.

By construction, the rank of $M_{n,j}$ is the dimension of $I_{n,j}/Q_{n,j}$ as an $\mathbb{F}_2$ -vector space. Since $\deg x_\alpha + m = j$ and modulo Q_n , we have

$$x_\alpha \cdot e_{n,m}(x) = \sum_{\gamma \subseteq [n], |\gamma|=m} x_\alpha x_\gamma = \sum_{\substack{\gamma \subseteq [n], |\gamma|=m, \\ \gamma \cap \alpha = \emptyset}} x_\alpha x_\gamma = \sum_{\substack{\beta \subseteq [n], |\beta|=j, \\ \alpha \subseteq \beta}} x_\beta.$$

In other words, the coefficient of x_β in $x_\alpha \cdot e_{n,m}$ is 1 if and only if $\alpha \subseteq \beta$. Therefore, the matrix $M_{n,j}$ is the incidence matrix relative to the inclusion of the subsets $\alpha \subseteq [n]$ of cardinality $0 \leq |\alpha| \leq j-1$ in the subsets $\beta \subseteq [n]$ of cardinality j . For example, if $n = 4$ and $j = 3$, we get the incidence matrix

$M_{4,3}$ via $\cdot$	$x_1x_2x_3$	$x_1x_2x_4$	$x_1x_3x_4$	$x_2x_3x_4$	
$1 \cdot e_{4,3}$	1	1	1	1	$\emptyset$
$x_1 \cdot e_{4,2}$	1	1	1	0	$\{1\}$
$x_2 \cdot e_{4,2}$	1	1	0	1	$\{2\}$
$x_3 \cdot e_{4,2}$	1	0	1	1	$\{3\}$
$x_1x_2 \cdot e_{4,1}$	1	1	0	0	$\{1, 2\}$
$x_1x_3 \cdot e_{4,1}$	1	0	1	0	$\{1, 3\}$
$x_2x_3 \cdot e_{4,1}$	1	0	0	1	$\{2, 3\}$
$x_4 \cdot e_{4,2}$	0	1	1	1	$\{4\}$
$x_1x_4 \cdot e_{4,1}$	0	1	1	0	$\{1, 4\}$
$x_2x_4 \cdot e_{4,1}$	0	1	0	1	$\{2, 4\}$
$x_3x_4 \cdot e_{4,1}$	0	0	1	1	$\{3, 4\}$
	$\{1, 2, 3\}$	$\{1, 2, 4\}$	$\{1, 3, 4\}$	$\{2, 3, 4\}$	$M_{4,3}$ via $\subseteq$

Wilson [Wil90, Lemma 1] proved that

$$\text{rk}(M_{n,j}) = \binom{n}{j-1} \quad \text{for every } j \leq \left\lfloor \frac{n}{2} \right\rfloor.$$

This implies that $a_{n,j} = \binom{n}{j} - \binom{n}{j-1}$ for every $j \leq \left\lfloor \frac{n}{2} \right\rfloor$, as desired.

To conclude the proof of Proposition 2.6, it remains to prove that $a_{n,j} = 0$ for every $j \geq \left\lfloor \frac{n}{2} \right\rfloor + 1$, that is, to prove that $I_{n,j} = R_{n,j}$ for every $j \geq \left\lfloor \frac{n}{2} \right\rfloor + 1$.

Finally, the ideal I_n satisfies the hypothesis of Proposition 2.5 with $\mathbb{F} = \mathbb{F}_2$ and $u = 0$, and therefore $I_{n,j} = R_{n,j}$ for every $j \geq \lfloor \frac{n}{2} \rfloor + 1$, which concludes the proof. $\square$

The previous results allow us to give a first estimate of $d_{\text{sol}}(S)$, where, we recall, the solutions of $S = 0$ correspond to the solutions of the related instance of the ESDP.

Theorem 2.7. *For $n \geq 2$, the solving degree of S is bounded from above by*

$$d_{\text{sol}}(S) \leq \begin{cases} 2^r + 1 & \text{if } n = 2^{r+1} - 2 \text{ or } n = 2^{r+1} - 1, \\ 2^r & \text{if } r \geq 2 \text{ and } 2^r \leq n \leq 2^{r+1} - 3, \end{cases} \quad (3)$$

where $r = \lfloor \log_2 n \rfloor \geq 1$.

Proof. By Theorem 1.1, we have that $d_{\text{sol}}(S) \leq \max\{d_{\text{reg}}(S)+1, \max_{f \in S} \deg(f)\}$, where

$$d_{\text{reg}}(S) + 1 \leq d_{\text{reg}}(S_2 \cup S_3) + 1 = d_{\text{reg}}(I_n) + 1 = \left\lfloor \frac{n}{2} \right\rfloor + 2$$

by Proposition 2.6 and

$$\max_{f \in S} \deg(f) = \max_{f \in S_1 \cup S_2 \cup S_3} \deg(f) = \max\{1, 2, 2^0, \dots, 2^r\} = 2^r$$

since $r \geq 1$. We distinguish two cases:

- If $r \geq 2$ and $2^r \leq n \leq 2^{r+1} - 3$, then

$$\left\lfloor \frac{n}{2} \right\rfloor + 2 \leq \left\lfloor 2^r - \frac{3}{2} \right\rfloor + 2 = 2^r - 2 + 2 = 2^r.$$

- If $n \in \{2^{r+1} - 2, 2^{r+1} - 1\}$, then

$$\left\lfloor \frac{n}{2} \right\rfloor + 2 = 2^r - 1 + 2 = 2^r + 1 > 2^r.$$

Notice that this includes the case $r = 1$, since in that case $n \in \{2, 3\}$.

This completes the proof of the statement. $\square$

Comparing this result with Theorem 1.2 we see that, for non-trivial values of t , the complexity of solving our polynomial model S for the ESDP is smaller than the complexity of solving the model from [Cam+25].

Notice that the bound of Theorem 2.7 is obtained by considering only the system consisting of the field equations and the equations corresponding to the weight constraint. Since the dependence on the weight t is eliminated when passing to the top-dimensional part S^{top} of the system, the bound depends only on n . In particular, it is independent of the underlying code and of the instance $(\mathbf{H}, s, t)$ of the ESDP. This suggests that one may further improve our model using the parity check encoding. In the next section, we propose a variant of the model that depends on the instance of the ESDP.

2.2 An improved model and its computational complexity

To improve the model described in the previous section, notice that the bound given in Theorem 2.7 is often controlled by the degrees of the equations. Therefore, we propose to remove the highest-degree polynomial from the system $S = 0$.

We now analyze the resulting system. The next lemma is a variation of Lemma 2.2 and is provided for completeness.

Lemma 2.8. *Given $n \geq 1$, $r = \lfloor \log_2 n \rfloor$ and $0 \leq t \leq n$ with binary expansion $t = \sum_{\ell=0}^r t_\ell 2^\ell$, let*

$$S'_3 = \{e_{n,2^i}(x) + t_i \mid i = 0, \dots, r-1\}.$$

The vectors of $\mathbb{F}_2^n$ that satisfy the polynomial system $S'_3 = 0$ are exactly

$$\begin{cases} \text{the vectors of weight } t \text{ or } t + 2^r & \text{if } 0 \leq t \leq n - 2^r \\ \text{the vectors of weight } t & \text{if } n - 2^r + 1 \leq t \leq 2^r - 1 \\ \text{the vectors of weight } t \text{ or } t - 2^r & \text{if } 2^r \leq t \leq n. \end{cases}$$

Proof. This easily follows from Lemma 2.2:

- If $0 \leq t \leq n - 2^r$, then $t_r = 0$. Let $t' = t + 2^r$, then $2^r \leq t' \leq n$. Therefore, a solution $y \in \mathbb{F}_2^n$ of $S'_3 = 0$, can have either $\text{wt}(y) = t$, if $e_{n,2^r}(y) = 0$, or $\text{wt}(y) = t' = t + 2^r$, if $e_{n,2^r}(y) = 1$.
- If $n - 2^r + 1 \leq t \leq 2^r - 1$, we have $t_r = 0$. Let y be a solution of $S'_3 = 0$. Then $e_{n,2^r}(y)$ can be either 0 or 1. If $e_{n,2^r}(y) = 0$, then

by Lemma 2.2, its weight is exactly t . If $e_{n,2^r}(y) = 1$, y has weight $t + 2^r \geq n + 1$, which is a contradiction. Therefore, all solutions of $S'_3 = 0$ must have weight t .

- If $2^r \leq t \leq n$, we can make similar considerations as in the first case with $t'' = t - 2^r$, which is such that $0 \leq t'' \leq n - 2^r$.

□

We can now consider the improved model $S' = S_1 \cup S_2 \cup S'_3$. Notice that the solutions of the instance $(\mathbf{H}, s, t)$ of the ESDP are also solutions of the polynomial system $S' = 0$. We recall that the ESDP has a unique solution if $t \leq \left\lfloor \frac{d(\mathbf{C})-1}{2} \right\rfloor \leq \left\lfloor \frac{n-1}{2} \right\rfloor \leq 2^r - 1$. Notice that this falls in the two first cases of Lemma 2.8. Under the cryptographic assumption that the solution of an instance $(\mathbf{H}, s, t)$ of the ESDP is unique, there are two possible scenarios:

- The solution of $S' = 0$ is unique, hence it is the solution of the ESDP. This occurs if the *wrong* instance $(\mathbf{H}, s, t \pm 2^r)$ happens to have no solution, or if the target weight t is such that $n - 2^r + 1 \leq t \leq 2^r - 1$. For example, if n is a power of 2, such as in the Parameter set 9 and 10 of [Cla22], all the non-trivial choices for t are in this interval.
- The system $S' = 0$ has more than one solution, but it contains exactly one vector of weight t . In this situation, an attacker can efficiently recover the desired vector by computing the Hamming weight of the solutions of $S' = 0$ and discarding the ones with the wrong weight. This final selection step is computationally negligible and does not contribute to the overall complexity of the model.

In order to compute the complexity of solving the ESDP using this new polynomial model, we want to estimate the solving degree of the polynomial system $S' = 0$. As we have done for the previous model, we start by bounding the degree of regularity $d_{\text{reg}}(S')$. We start by studying the ideal

$$J_n = (e_{n,2^0}, \dots, e_{n,2^{r-1}}) + Q_n,$$

which is generated by the polynomials in S_2^{top} and $(S'_3)^{\text{top}}$. As for I_n , by Lemma 2.3

$$J_n = (e_{n,j} \mid j = 1, \dots, 2^r - 1) + Q_n.$$

The next lemma will be useful in the sequel.

Lemma 2.9. *Let $k \leq n$. For every $j = 1, \dots, n$, one has*

$$e_{n,j}(x_1, \dots, x_n) = \sum_{i=0}^j e_{k,i}(x_1, \dots, x_k) \cdot e_{n-k,j-i}(x_{k+1}, \dots, x_n),$$

where $e_{m,0} = 1$ and $e_{m,l} = 0$ for $l > m$.

Proof. If $i > k$ or $j-i > n-k$, the product $e_{k,i}(x_1, \dots, x_k) \cdot e_{n-k,j-i}(x_{k+1}, \dots, x_n)$ is equal to 0. If i, j are such that $i \leq k$ and $j-i \leq n-k$, then

$$\begin{aligned} e_{k,i}(x_1, \dots, x_k) \cdot e_{n-k,j-i}(x_{k+1}, \dots, x_n) &= \\ &= \left(\sum_{\alpha \subseteq [k], |\alpha|=i} x_\alpha \right) \left(\sum_{\beta \subseteq \{k+1, \dots, n\}, |\beta|=j-i} x_\beta \right) = \\ &= \sum_{\substack{\alpha \subseteq [k], |\alpha|=i, \\ \beta \subseteq \{k+1, \dots, n\}, |\beta|=j-i}} x_\alpha x_\beta = \sum_{\substack{\gamma \subseteq [n], |\gamma \cap [k]|=i, \\ |\gamma \cap \{k+1, \dots, n\}|=j-i}} x_\gamma, \end{aligned}$$

hence

$$\begin{aligned} &\sum_{i=0}^j e_{k,i}(x_1, \dots, x_k) \cdot e_{n-k,j-i}(x_{k+1}, \dots, x_n) = \\ &= \sum_{i=0}^j \sum_{\substack{\gamma \subseteq [n], |\gamma \cap [k]|=i, \\ |\gamma \cap \{k+1, \dots, n\}|=j-i}} x_\gamma = \sum_{\gamma \subseteq [n], |\gamma|=j} x_\gamma = e_{n,j}(x_1, \dots, x_n). \end{aligned}$$

□

The lemma allows us to express the ideal J_n in different ways.

Lemma 2.10. *For any $1 \leq k \leq n-1$, the following ideals are equal:*

$$\begin{aligned} J_n &= (e_{n,2^i}(x) \mid i = 0, \dots, r-1) + Q_n \\ &= (e_{n,j}(x) \mid j = 1, \dots, 2^r - 1) + Q_n \\ &= (e_{k,2^i}(x_1, \dots, x_k) + e_{n-k,2^i}(x_{k+1}, \dots, x_n) \mid i = 0, \dots, r-1) + Q_n \\ &= (e_{k,j}(x_1, \dots, x_k) + e_{n-k,j}(x_{k+1}, \dots, x_n) \mid j = 1, \dots, 2^r - 1) + Q_n. \end{aligned}$$

Proof. Throughout the proof, we denote

$$K_n = (e_{k,2^i}(x_1, \dots, x_k) + e_{n-k,2^i}(x_{k+1}, \dots, x_n) \mid i = 0, \dots, r-1) + Q_n,$$

$$K'_n = (e_{k,j}(x_1, \dots, x_k) + e_{n-k,j}(x_{k+1}, \dots, x_n) \mid j = 1, \dots, 2^r - 1) + Q_n.$$

We also write $e_{k,j}$ for $e_{k,j}(x_1, \dots, x_k)$, and similarly for $e_{n,j}$ and $e_{n-k,j}$.

The second equality follows from Lemma 2.3. It is clear that $K_n \subseteq K'_n$. Hence, to prove the chain of equalities in the statement, it suffices to prove the inclusions $K'_n \subseteq J_n$ and $J_n \subseteq K_n$.

$K'_n \subseteq J_n$: We prove that $e_{k,j} + e_{n-k,j} \in J_n$ for all $j = 1, \dots, 2^r - 1$ by induction on j . Since $e_{k,j} + e_{n-k,j} = 0$ for $j > \max\{k, n-k\}$, we may restrict to $j \leq \max\{k, n-k\}$. The case $j = 1$ is trivial, since $e_{k,1} + e_{n-k,1} = e_{n,1} \in J_n$. For $j > 1$, we consider two cases.

– If $j \leq \min\{k, n-k\}$, by Lemma 2.9 we have

$$e_{n,j} = \sum_{i=0}^j e_{k,i} \cdot e_{n-k,j-i} = e_{k,j} + \sum_{i=1}^{j-1} e_{k,i} \cdot e_{n-k,j-i} + e_{n-k,j}.$$

Since $e_{n,j} \in J_n$, it suffices to show that $q_j = \sum_{i=1}^{j-1} e_{k,i} \cdot e_{n-k,j-i} \in J_n$.

Let $p_i = e_{k,i} + e_{n-k,i}$. One has

$$q_j = \sum_{i=1}^{j-1} e_{k,i}(p_{j-i} + e_{k,j-i}) = \sum_{i=1}^{j-1} e_{k,i} \cdot p_{j-i} + \sum_{i=1}^{j-1} e_{k,i} \cdot e_{k,j-i}.$$

The first sum, that we denote by p'_j , belongs to J_n , since $p_i \in J_n$ for all $i = 1, \dots, j-1$ by the induction hypothesis. By Lemma 2.3, the second sum equals $\alpha e_{k,j}$, for some $\alpha \in \mathbb{F}_2$. If $\alpha = 0$, then $q_j = p'_j \in J_n$. If $\alpha = 1$, then $q_j = p'_j + e_{k,j}$, and therefore $e_{n,j} = e_{k,j} + q_j + e_{n-k,j} = p'_j + e_{n-k,j}$, which implies that $e_{n-k,j} \in J_n$. In addition,

$$q_j = \sum_{i=1}^{j-1} (p_i + e_{n-k,i})e_{n-k,j-i} = \sum_{i=1}^{j-1} p_i \cdot e_{n-k,j-i} + \sum_{i=1}^{j-1} e_{n-k,i} \cdot e_{n-k,j-i}.$$

Let p''_j denote the first sum, then $p''_j \in J_n$ as before and the second sum is $\alpha e_{n-k,j}$, for the same $\alpha \in \mathbb{F}_2$ as above, since the bit check of Lemma 2.3 involves the same pairs $(i, j-i)$. Since $\alpha = 1$, we obtain $q_j = p''_j + e_{n-k,j} \in J_n$. This concludes the proof.

– If $j > \min\{k, n-k\}$, up to exchanging k and $n-k$ we may assume without loss of generality that $k < j \leq n-k$. We want to prove that $e_{k,j} + e_{n-k,j} = e_{n-k,j} \in J_n$. We proceed as before:

$$e_{n,j} = \sum_{i=0}^j e_{k,i} \cdot e_{n-k,j-i} \stackrel{j > k}{=} \sum_{i=1}^k e_{k,i} \cdot e_{n-k,j-i} + e_{n-k,j}.$$

Again, letting $p_i = e_{k,i} + e_{n-k,i}$, we have

$$\sum_{i=1}^k e_{k,i} \cdot e_{n-k,j-i} = \sum_{i=1}^k e_{k,i} (p_{j-i} + e_{k,j-i}) = \sum_{i=1}^k e_{k,i} \cdot p_{j-i} + \sum_{i=1}^k e_{k,i} \cdot e_{k,j-i},$$

where the first summand belongs to J_n , since by induction $p_i \in J_n$ if $i < j$, and the second summand is 0 by Lemma 2.3. Therefore, $e_{n-k,j} \in J_n$ as desired.

$J_n \subseteq K_n$: We want to prove that $e_{n,2^i} \in K_n$ for $i = 0, \dots, r-1$. We proceed by induction on i . The case $i = 0$ is clear. Hence, suppose $i > 0$ and assume that $e_{n,2^j} \in K_n$ for $0 \leq j \leq i-1$. By Lemma 2.3, this implies $e_{n,j} \in K_n$ for $1 \leq j \leq 2^i - 1$. By Lemma 2.9, moreover,

$$e_{n,2^i} = \sum_{j=0}^{2^i} e_{k,j} \cdot e_{n-k,2^i-j} = e_{k,2^i} + \sum_{j=1}^{2^i-1} e_{k,j} \cdot e_{n-k,2^i-j} + e_{n-k,2^i}.$$

So it suffices to prove that $\sum_{j=1}^{2^i-1} e_{k,j} \cdot e_{n-k,2^i-j} \in K_n$.

Let $1 \leq u \leq \min\{k, 2^i - 1\}$, then by Lemma 2.9

$$\begin{aligned} K_n \ni e_{k,u} \cdot e_{n,2^i-u} &= e_{k,u} \sum_{j=0}^{2^i-u} e_{k,j} \cdot e_{n-k,2^i-u-j} = \\ &= \sum_{j'=u}^{2^i-1} e_{k,u} \cdot e_{k,j'-u} \cdot e_{n-k,2^i-j'} \end{aligned}$$

The last equality follows from setting $j' = j + u$ and by observing that $e_{k,u} \cdot e_{k,2^i-u} = 0$ for every u by Lemma 2.3, since u and $2^i - u$ must share at least one non-zero bit in their binary expansion. Therefore

$$\begin{aligned} K_n \ni \sum_{u=1}^{\min\{k, 2^i-1\}} e_{k,u} \cdot e_{n,2^i-u} &= \sum_{u=1}^{\min\{k, 2^i-1\}} \sum_{j'=u}^{2^i-1} e_{k,u} \cdot e_{k,j'-u} \cdot e_{n-k,2^i-j'} = \\ &= \sum_{j'=1}^{2^i-1} \sum_{u=1}^{j'} e_{k,u} \cdot e_{k,j'-u} \cdot e_{n-k,2^i-j'} = \\ &= \sum_{j'=1}^{2^i-1} |T_{j'}| \cdot e_{k,j'} \cdot e_{n-k,2^i-j'}, \end{aligned}$$

where $T_{j'}$ is defined, following Lemma 2.3, as the set

$$\begin{aligned} T_{j'} &= \{u \in [1, j'] \mid \text{there is no } \ell \text{ such that } u_\ell = 1 = (j' - u)_\ell\} = \\ &= \{u \in [1, j'] \mid \text{if } j'_\ell = 0 \text{ for some } \ell, \text{ then } t_\ell = 0\}. \end{aligned}$$

Notice that if $j' > k$, then $e_{k, j'} = 0$ and $T_{j'}$ plays no role. In addition, for every j'

$$|T_{j'}| = 2^{|\{\ell \in [0, \lceil \log_2(j') \rceil] \mid j'_\ell = 1\}|} - 1,$$

since, if $j'_\ell = 1$, then u_ℓ can be either 0 or 1, however $0 \notin T_{j'}$. In particular, $|T_{j'}|$ is odd for all $j' \leq k$. It follows that

$$\sum_{j=1}^{2^i-1} e_{k, j} \cdot e_{n-k, 2^i-j} = \sum_{u=1}^{\min\{k, 2^i-1\}} e_{k, u} \cdot e_{n, 2^i-u} \in K_n,$$

which concludes the proof. $\square$

Before continuing with the estimate of $d_{\text{reg}}(S')$, notice that one could use the parity check encoding to eliminate some variables. In fact, since $\mathbf{H} \in \mathbb{F}_2^{(n-k) \times n}$ is a full rank matrix, it can be written in the form

$$\mathbf{H} = (\mathbf{L} \mid \mathbf{I}_{n-k}) = \left(\begin{array}{cccc|c} \cdots & \ell_1 & \cdots & & \vdots \\ & \vdots & & & \vdots \\ \cdots & \ell_{n-k} & \cdots & & \vdots \end{array} \right),$$

where $\mathbf{I}_{n-k}$ is the identity matrix of size $n-k$ and $\mathbf{L}$ is a matrix in $\mathbb{F}_2^{(n-k) \times k}$, by performing a Gaussian elimination on the rows and, if necessary, swapping columns. Notice that these operations do not change the underlying problem, since the matrix obtained is the parity check matrix of an equivalent code. We can therefore assume that $\mathbf{H}$ is of the form above. With this assumption, the equations corresponding to S_1 are of the form

$$x_{k+1} = \ell_1(x_1, \dots, x_k) + s_1, \dots, x_n = \ell_{n-k}(x_1, \dots, x_k) + s_{n-k}.$$

Hence, S_1^{top} consists of the linear polynomials $x_j + \ell_{j-k}(x_1, \dots, x_k)$ for $j = k+1, \dots, n$. Notice that

$$\frac{R_n}{J_n + (S_1^{\text{top}})} = \frac{R_n}{J_n + (x_{k+1} + \ell_1(x_1, \dots, x_k), \dots, x_n + \ell_{n-k}(x_1, \dots, x_k))} \cong \frac{R_k}{J'_k},$$

where

$$J'_k = (e_{k,2^i}(x_1, \dots, x_k) + e_{n-k,2^i}(\ell_1, \dots, \ell_{n-k}) \mid i = 0, \dots, r-1) + Q_k$$

and $\ell_j = \ell_j(x_1, \dots, x_k)$. Therefore, since the ideal generated by $(S')^{\text{top}}$ is precisely $J_n + (S_1^{\text{top}})$, we have that $d_{\text{reg}}(S') = d_{\text{reg}}(J'_k)$, and for this reason we can now focus on the ideal J'_k .

Notation 2.2. For $j = 1, \dots, n-k$, we write ℓ_j for $\ell_j(x_1, \dots, x_k)$. For every $\alpha = \{\alpha_1, \dots, \alpha_i\} \subseteq [n-k]$, ℓ_α denotes the product $\ell_{\alpha_1} \cdots \ell_{\alpha_i} \in R_{k,i}$. Finally, for $g \in R_{n-k}$, we abbreviate $g(\ell_1, \dots, \ell_{n-k})$ as $g(\ell)$.

The substitution reduces the degrees of the involved polynomials, since $e_{k,2^i}(x)$ and $e_{n-k,2^i}(\ell)$ are 0 if 2^i is greater than k and $n-k$, respectively. Moreover, it highlights and isolates the dependence on the matrix $\mathbf{L}$, and so on the specific instance of the ESDP. The next lemma clarifies that dependence.

Lemma 2.11. Let $k \leq n$ and $1 \leq i \leq n-k$. For any $\beta \subseteq [k]$ of cardinality i , the coefficient of the squarefree monomial x_β in the polynomial

$$e_{n-k,i}(\ell) = \sum_{\alpha \subseteq [n-k], |\alpha|=i} \ell_\alpha \in R_{k,i}$$

is the sum of all i -minors of $\mathbf{L}$ whose columns are indexed by β .

Proof. Let $\mathbf{L} = (\ell_{i,j})_{i=1, \dots, n-k}^{j=1, \dots, k}$. For any $\alpha = \{\alpha_1, \dots, \alpha_i\} \subseteq [n-k]$ and $\beta = \{\beta_1, \dots, \beta_i\} \subseteq [k]$, the determinant of the submatrix of $\mathbf{L}$ consisting of rows indexed by α and columns indexed by β is

$$\det(\mathbf{L}[\alpha, \beta]) = \sum_{\sigma \in S_i} \prod_{j=1}^i \ell_{\alpha_j, \beta_{\sigma(j)}}.$$

We want to prove that the coefficient of x_β in $e_{n-k,i}(\ell)$ is the sum of all $\det(\mathbf{L}[\alpha, \beta])$, with $\alpha \subseteq [n-k]$ and $|\alpha| = i$.

Consider a summand ℓ_α of $e_{n-k,i}(\ell)$. We have

$$\begin{aligned} \ell_\alpha &= \ell_{\alpha_1}(x_1, \dots, x_k) \cdots \ell_{\alpha_i}(x_1, \dots, x_k) = \\ &= (\ell_{\alpha_1,1}x_1 + \cdots + \ell_{\alpha_1,k}x_k) \cdots (\ell_{\alpha_i,1}x_1 + \cdots + \ell_{\alpha_i,k}x_k). \end{aligned}$$

The monomial x_β is equal to the product $x_{\beta_{\sigma(1)}} \cdots x_{\beta_{\sigma(i)}}$ for every permutation $\sigma \in S_i$. For a given permutation σ , the coefficient of the product $x_{\beta_{\sigma(1)}} \cdots x_{\beta_{\sigma(i)}}$ is $\prod_{j=1}^i \ell_{\alpha_j, \beta_{\sigma(j)}}$. Therefore, the coefficient of x_β in ℓ_α

is $\sum_{\sigma \in S_i} \prod_{j=1}^i \ell_{\alpha_j, \beta_{\sigma(j)}} = \det(\mathbf{L}[\alpha, \beta])$. The thesis follows from observing that $e_{n-k, i}(\ell) = \sum_{\alpha \subseteq [n-k], |\alpha|=i} \ell_\alpha$.

□

This lemma gives us a way to compute $e_{n-k, 2^i}(\ell)$, but more importantly, it gives us a way to quickly see if some of the $e_{n-k, 2^i}(\ell)$ can be discarded. In fact, if $2^i > \text{rk}(\mathbf{L})$, then every minor of order 2^i of the matrix $\mathbf{L}$ vanishes, hence $e_{n-k, 2^i}(\ell) \in Q_k$. Therefore

$$\begin{aligned} J'_k &= (e_{k, 2^i}(x) + e_{n-k, 2^i}(\ell) \mid i = 0, \dots, r-1) + Q_k = \\ &= (e_{k, 2^i}(x) + e_{n-k, 2^i}(\ell) \mid 1 \leq 2^i \leq \text{rk}(\mathbf{L})) + (e_{k, 2^i}(x) \mid \text{rk}(\mathbf{L}) < 2^i \leq 2^{r-1}) + Q_k. \end{aligned}$$

Since $\mathbf{L} \in \mathbb{F}_q^{(n-k) \times k}$, we have

$$\text{rk}(\mathbf{L}) \leq \min\{k, n-k\} \leq \left\lfloor \frac{n}{2} \right\rfloor \quad \text{and} \quad 2^{r-1} \leq \left\lfloor \frac{n}{2} \right\rfloor < 2^r,$$

so $\text{rk}(\mathbf{L}) \leq 2^r - 1$. As we are not aware of any relation between $\text{rk}(\mathbf{L})$ and 2^{r-1} , we rewrite J'_k as

$$J'_k = (e_{k, j}(x) + e_{n-k, j}(\ell) \mid 1 \leq j \leq \text{rk}(\mathbf{L})) + (e_{k, j}(x) \mid \text{rk}(\mathbf{L}) < j \leq 2^r - 1) + Q_k.$$

Equality holds, since $e_{k, j}(x_1, \dots, x_k) + e_{n-k, j}(x_{k+1}, \dots, x_n) \in J_n$ for every $j = 1, \dots, 2^r - 1$ by Lemma 2.10, which implies that $e_{k, j}(x) + e_{n-k, j}(\ell) \in J'_k$.

We are now ready to estimate the degree of regularity of J'_k , hence the complexity of solving the polynomial system $S' = 0$ to solve the ESDP. We proceed similarly to what we have done for $d_{\text{reg}}(I_n)$, even if this time we will be able to give just a bound, and not to get the exact value of the degree of regularity. We restrict to the case $n \geq 4$, since the other cases are not cryptographically relevant.

Theorem 2.12. *Suppose that $n \geq 4$. The degree of regularity of the ideal J'_k is bounded by*

$$d_{\text{reg}}(J'_k) \leq \begin{cases} \left\lfloor \frac{k + \text{rk}(\mathbf{L})}{2} \right\rfloor + 2 = 2^r + 1 & \text{if } n = 2^{r+1} - 1, k \geq 2^r, \\ & \text{and } \text{rk}(\mathbf{L}) = n - k \text{ (}\mathbf{L} \text{ has maximal rank)}, \\ \left\lfloor \frac{k + \text{rk}(\mathbf{L})}{2} \right\rfloor + 1 & \text{otherwise.} \end{cases}$$

Proof. We divide the proof in three cases, one of which is divided in three subcases.

- If $k \leq 2^r - 1$, then $e_{k,j}(x) = e_{k,j}$ vanishes for every $j > k$, so $J'_k = (e_{k,j}(x) + e_{n-k,j}(\ell) \mid 1 \leq j \leq \text{rk}(\mathbf{L})) + (e_{k,j}(x) \mid \text{rk}(\mathbf{L}) < j \leq k) + Q_k$. If $\text{rk}(\mathbf{L}) < k$, by applying Proposition 2.5 with $u = \text{rk}(\mathbf{L})$ to J'_k , we obtain

$$d_{\text{reg}}(J'_k) \leq \left\lfloor \frac{k + \text{rk}(\mathbf{L})}{2} \right\rfloor + 1. \quad (4)$$

If instead $\text{rk}(\mathbf{L}) = k$, (4) still holds, since $d_{\text{reg}}(J'_k) \leq d_{\text{reg}}(Q_k) = k + 1$.

- If $2^r \leq k$ and $\text{rk}(\mathbf{L}) \leq 2^r - 2$, using Lemma 2.4 as in the proof of Proposition 2.5, but starting from the last polynomials of the list, we obtain:

$$\begin{array}{l|l} \text{Step 0} & e_{k, \text{rk}(\mathbf{L})+1} \quad e_{k, \text{rk}(\mathbf{L})+2} \quad \cdots \quad e_{k, 2^r-2} \quad e_{k, 2^r-1} \in J'_k \\ \text{Step 1} & / \quad e_{k-1, \text{rk}(\mathbf{L})+2} \quad \cdots \quad e_{k-1, 2^r-2} \quad e_{k-1, 2^r-1} \in J'_k \\ \vdots & / \quad / \quad \vdots \quad \vdots \quad \vdots \in J'_k. \end{array}$$

If $\text{rk}(\mathbf{L}) + 1 + s \leq 2^r - 1 \leq k - s$, that is, for

$$s \leq \min\{2^r - 2 - \text{rk}(\mathbf{L}), k - 2^r + 1\},$$

in Step s we have $e_{k-s, \text{rk}(\mathbf{L})+1+s}, \dots, e_{k-s, 2^r-1} \in J'_k$. We analyze three different cases, depending on the value of $k + \text{rk}(\mathbf{L})$. Notice that $k + \text{rk}(\mathbf{L}) \leq n \leq 2^{r+1} - 1$.

- If $k + \text{rk}(\mathbf{L}) \leq 2^{r+1} - 3$, then the last meaningful step happens for $\bar{s} = k - 2^r + 1$. In this case, $e_{2^r-1, k+\text{rk}(\mathbf{L})-2^r+2}, \dots, e_{2^r-1, 2^r-1} \in J'_k$. Applying Proposition 2.5 with $2^r - 1$ variables and $u = k + \text{rk}(\mathbf{L}) - 2^r + 1$, we get that

$$d_{\text{reg}}(J'_k) \leq \left\lfloor \frac{k + \text{rk}(\mathbf{L})}{2} \right\rfloor + 1.$$

- If $k + \text{rk}(\mathbf{L}) = 2^{r+1} - 2$, then the last meaningful step happens for $\bar{s} = 2^r - 2 - \text{rk}(\mathbf{L}) = k - 2^r$. In this case, $e_{2^r, 2^r-1} \in J'_k$, therefore $x_{2^r} \cdot e_{2^r, 2^r-1} = x_{2^r} \cdot e_{2^r-1, 2^r-1} + x_{2^r}^2 \cdot e_{2^r-1, 2^r-2} \in J'_k$, hence $e_{2^r, 2^r} \in J'_k$. The same can be done analogously with the other variables, and since $\left\lfloor \frac{k+\text{rk}(\mathbf{L})}{2} \right\rfloor + 1 = \left\lfloor \frac{2^{r+1}-2}{2} \right\rfloor + 1 = 2^r$, bound (4) holds also in this case, as in the proof of Proposition 2.5.

- If $k + \text{rk}(\mathbf{L}) = 2^{r+1} - 1$, then the last meaningful step happens for $\bar{s} = 2^r - 2 - \text{rk}(\mathbf{L}) = k - 2^r - 1$. In this case, $e_{2^r+1, 2^r-1} \in J'_k$, hence

$$\begin{aligned}
x_{2^r+1}x_{2^r} \cdot e_{2^r+1, 2^r-1} &= x_{2^r+1}x_{2^r}(e_{2^r, 2^r-1} + x_{2^r+1}e_{2^r, 2^r-2}) = \\
&= x_{2^r+1}x_{2^r}(e_{2^r-1, 2^r-1} + x_{2^r}e_{2^r-1, 2^r-2} + x_{2^r+1}e_{2^r, 2^r-2}) = \\
&= x_{2^r+1}x_{2^r}e_{2^r-1, 2^r-1} + x_{2^r+1}x_{2^r}^2e_{2^r-1, 2^r-2} + x_{2^r+1}^2x_{2^r}e_{2^r, 2^r-2} = \\
&= e_{2^r+1, 2^r+1} + x_{2^r+1}x_{2^r}^2e_{2^r-1, 2^r-2} + x_{2^r+1}^2x_{2^r}e_{2^r, 2^r-2}
\end{aligned}$$

belongs to J'_k , and therefore $e_{2^r+1, 2^r+1} \in J'_k$. We obtain the bound

$$d_{\text{reg}}(J'_k) \leq \left\lfloor \frac{k + \text{rk}(\mathbf{L})}{2} \right\rfloor + 2 = 2^r + 1. \quad (5)$$

Notice that this case only occurs when $n = 2^{r+1} - 1$ and $\text{rk}(\mathbf{L}) = n - k \leq 2^r - 2 \leq k - 2$, which implies $k \geq 2^r + 1$.

- If $2^r \leq k$ and $\text{rk}(\mathbf{L}) \geq 2^r - 1$, then $n \geq k + \text{rk}(\mathbf{L}) \geq 2^{r+1} - 1$. This forces $n = 2^{r+1} - 1$, $k = 2^r$ and $\text{rk}(\mathbf{L}) = n - k = 2^r - 1$. Then (5) holds, since $d_{\text{reg}}(J'_k) \leq k + 1 = \left\lfloor \frac{k + \text{rk}(\mathbf{L})}{2} \right\rfloor + 2$.

□

Since the polynomials in S' have degree smaller than or equal to 2^{r-1} , by combining Theorem 2.12 and Theorem 1.1 we obtain

Corollary 2.13. *suppose that $n \geq 4$. The solving degree of the polynomial system $S' = 0$ is bounded by*

$$d_{\text{sol}}(S') \leq \begin{cases} 2^r + 2 & \text{if } n = 2^{r+1} - 1, k \geq 2^r, \\ & \text{and } \text{rk}(\mathbf{L}) = n - k, \\ \max \left\{ \left\lfloor \frac{k + \text{rk}(\mathbf{L})}{2} \right\rfloor + 2, 2^{r-1} \right\} & \text{otherwise.} \end{cases}$$

While the model S' solves a relaxed version of ESDP, the total complexity of solving the ESDP using the model S' is dominated by the complexity of solving $S' = 0$. Moreover, the solving degree of S' is almost always smaller than the solving degree of S , which we estimated in Theorem 2.7. Therefore, it is more efficient to solve the ESDP using the model S' . In contrast with the case of S , the bound on the solving degree of S' depends on the dimension

k of the code and the rank of the matrix $\mathbf{L}$, and therefore on the specific instance of the ESDP.

In the next section, we briefly discuss how to refine the bounds of Theorem 2.12 and Corollary 2.13 by relating the rank of the matrix $\mathbf{L}$ to invariants of the code $\mathbf{C}$.

2.3 Bounds on the rank of the matrix $\mathbf{L}$

Given a parity check matrix $\mathbf{H} \in \mathbb{F}_2^{(n-k) \times n}$ of the code $\mathbf{C}$, we consider a matrix $\mathbf{L} \in \mathbb{F}_2^{(n-k) \times k}$ such that $(\mathbf{L} \mid \mathbf{I}_{n-k})$ is the parity check matrix of a code equivalent to $\mathbf{C}$. A matrix $\mathbf{L}$ exists, since $\mathbf{H}$ is full rank, however it is not unique. In fact, different choices of $\mathbf{L}$ may even have different ranks, as the next example shows.

Example 2.1. Let

$$\mathbf{H} = \left(\begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{array} \right) \quad \text{and} \quad \mathbf{H}' = \left(\begin{array}{cc|cc} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{array} \right).$$

It is easy to check that $\mathbf{H}$ and $\mathbf{H}'$ are parity check matrix of two equivalent codes. However, the corresponding $\mathbf{L}$ and $\mathbf{L}'$ have rank 2 and 1, respectively.

By Theorem 2.12, the matrix $\mathbf{L}$ of least rank corresponds to the system S' with predicted smallest solving degree. It is not always clear, however, how low $\text{rk}(\mathbf{L})$ can be and how one can produce an $\text{rk}(\mathbf{L})$ of small rank, starting from a given parity check matrix $\mathbf{H}$. The rest of the section is devoted to the study of these questions.

We start with a lemma that computes the rank of $\mathbf{L}$ for codes with large minimum distance.

Lemma 2.14. *If the minimum distance of the code is $d(\mathbf{C}) \geq k$, then any matrix $\mathbf{L}$ as above has full rank.*

Proof. Let $a_1, \dots, a_k \in \mathbb{F}_2$ be such that $a_1 \ell_1^c + \dots + a_k \ell_k^c = 0$, where ℓ_i^c denotes the i -th column of $\mathbf{L}$, $i = 1, \dots, k$. Therefore, the vector $a = (a_1, \dots, a_k, 0, \dots, 0) \in \mathbb{F}_2^n$ is such that $\mathbf{H} \cdot a^\top = (\mathbf{L} \mid \mathbf{I}_{n-k}) \cdot a^\top = 0$. Since $\mathbf{H}$ is a parity check matrix of $\mathbf{C}$, this implies that a is a codeword of $\mathbf{C}$. Since $\text{wt}(a) \leq k < d(\mathbf{C})$, the only possible case is $a = 0$, which means that the columns of $\mathbf{L}$ are linearly independent and therefore $\text{rk}(\mathbf{L})$ is maximum. $\square$

This means that for codes with a sufficiently large minimum distance, there is no optimal choice for the matrix $\mathbf{L}$. The situation is different if

$d(\mathbf{C}) \leq k$. In this case, we consider a generalization of the minimum distance, the *generalized Hamming weights*. In order to define them, we start by defining the *support* of a codeword $c = (c_1, \dots, c_n) \in \mathbf{C}$ as

$$\text{supp}(c) = \{i \in [n] \mid c_i \neq 0\}.$$

The support of a subcode $\mathbf{D} \leq \mathbf{C}$ is defined as

$$\text{supp}(\mathbf{D}) = \bigcup_{x \in \mathbf{D}} \text{supp}(x).$$

Definition 2.2 (Generalized Hamming weights – GHWs). For $1 \leq i \leq k$, the i -th *generalized Hamming weight* of an $[n, k]$ -linear code $\mathbf{C}$ is

$$d_i(\mathbf{C}) = \min\{|\text{supp}(\mathbf{D})| : \mathbf{D} \leq \mathbf{C}, \dim(\mathbf{D}) = i\}.$$

Notice that $d_1(\mathbf{C}) = d(\mathbf{C})$. For the main properties of GHWs, we refer to [HP03, Chapter 7.10], where one can find the next result and more.

Theorem 2.15 (Monotonicity and Generalized Singleton bound). *For an $[n, k]$ -linear code $\mathbf{C}$, we have*

- *monotonicity*: $1 \leq d_1(\mathbf{C}) < d_2(\mathbf{C}) < \dots < d_k(\mathbf{C}) \leq n$;
- *generalized Singleton bound*: $d_i(\mathbf{C}) \leq n - k + i$.

If $d(\mathbf{C}) \leq k$, one can look for the maximum $i \in [k]$ such that $d_i(\mathbf{C}) \leq k$. This idea is related to our problem as follows.

Proposition 2.16. *Given a parity check matrix $\mathbf{H} = (\mathbf{L} \mid \mathbf{I}_{n-k})$ of an $[n, k]$ -linear code $\mathbf{C}$ such that $d(\mathbf{C}) \leq k$, we have*

$$\text{rk}(\mathbf{L}) \geq k - \max\{i \in [k] \mid d_i(\mathbf{C}) \leq k\}.$$

Proof. Since $d(\mathbf{C}) \leq k$, the set $\{i \in [k] \mid d_i(\mathbf{C}) \leq k\}$ is not-empty, and therefore we can consider its maximum $r = \max\{i \in [k] \mid d_i(\mathbf{C}) \leq k\}$. If $r = k$, then $\text{rk}(\mathbf{L}) \geq 0$ is obviously true. Otherwise, we have that for any $j > r$, $d_j(\mathbf{C}) > k$, and therefore for any subcode $\mathbf{D}$ of $\mathbf{C}$ such that $\dim(\mathbf{D}) = j > r$, we have $|\text{supp}(\mathbf{D})| > k$. Therefore, any subcode $\mathbf{D} \leq \mathbf{C}$ such that $|\text{supp}(\mathbf{D})| \leq k$, we have $\dim(\mathbf{D}) \leq r$.

We now consider the subcode $\mathbf{V} = \mathbf{C} \cap \{(x_1, \dots, x_n) \in \mathbb{F}_2^n \mid x_{k+1} = \dots = x_n = 0\}$. The support of $\mathbf{V}$ is contained in $\{1, \dots, k\}$, and therefore

$|\text{supp}(\mathbf{V})| \leq k$, and thus $\dim(\mathbf{V}) \leq r$ for the above considerations. It is also true that $\mathbf{V} \cong \text{Ker}(\mathbf{L})$, since for any $x \in \mathbb{F}_2^k$ it holds

$$\mathbf{L} \cdot x^\top = 0 \iff \mathbf{L} \cdot x^\top + \mathbf{I}_{n-k} \cdot 0^\top = 0 \iff \mathbf{H} \cdot (x \mid 0)^\top = 0 \iff (x \mid 0) \in \mathbf{V}.$$

Therefore, we have $\dim(\text{Ker}(\mathbf{L})) \leq r$, which implies $\text{rk}(\mathbf{L}) = \dim(\text{Im}(\mathbf{L})) \geq k - r$ as desired. $\square$

We now have a lower bound for $\text{rk}(\mathbf{L})$, which can be used to refine the bound given in Corollary 2.13. However, there are two main difficulties with this approach. First, computing the GHWs of a code may be hard. Although there are many theoretical results, there is no general efficient algorithm to compute the GHWs of a generic linear code (see e.g. [San25]). Since the problem of computing the minimum distance $d_1(\mathbf{C})$ of a code is NP-complete [BMT78], the problem of computing the GHWs is also NP-complete. Second, it is not clear how to compute the matrix $\mathbf{L}$ of least rank, or even one of relatively small rank. This problem is related to understanding the relations of linear dependence among the columns of $\mathbf{H}$, which is in turn related to the problem of finding the *minimal codewords* of $\mathbf{C}$. The minimal codewords are defined as the codewords of $\mathbf{C}$ whose support does not properly contain the support of any other non-zero codeword. For a general linear code, the problem of finding its minimal codewords is again NP-complete. Nevertheless, the GHWs (or bounds on them) are known for several families of codes. See for example [Wei91; FTW92; HP98; Lee15; LCQ22].

3 Some variants of the binary ESDP

In this section, we briefly discuss how to apply some of the ideas from the previous section to other contexts. More specifically, we discuss the BSDP, fields of size $q > 2$, and the Regular variant.

3.1 Bounded Syndrome Decoding Problem – BSDP

So far we discussed the ESDP, where we search for an error of weight exactly t . This is the case of main interest from the cryptographic point of view. Intuitively, in cryptography we prefer to add as much noise as possible to the plaintext, therefore we often add an error of weight t equal to the maximum error-correction capability of the code. In coding theory, one is more likely to be interested in the BSDP, as one wants to correct every error whose

weight is smaller than or equal to the maximum error-correction capability of the code. The two variants of the SDP, however, are strictly related. In fact, one can solve the ESDP for a target weight t by solving the BSDP for $t - 1$ and t , while solving the ESDP for $1, \dots, t$ allows to solve the BSDP for t . One expects that the complexity of the ESDP grows with t and that the complexity of the BSDP for t is dominated by the last step, i.e. by the ESDP for t . If this is the case, then the ESDP and the BSDP for the same t have the same asymptotic complexity.

We now briefly describe how to apply the ideas from the previous section to the BSDP. The core is the following lemma, which is a variation of Lemma 2.2:

Lemma 3.1. *For any target weight $0 \leq t \leq n$,*

$$\{x \in \mathbb{F}_2^n \mid e_{n,j}(x) = 0 \text{ for all } j = t + 1, \dots, n\} = \{x \in \mathbb{F}_2^n \mid \text{wt}(x) \leq t\}.$$

Proof. " $\subseteq$ ": suppose that $\text{wt}(x) > t$. Then, by Lemma 2.2, we have that $e_{n,\text{wt}(x)}(x) = \binom{\text{wt}(x)}{\text{wt}(x)} = 1$, which is a contradiction. Therefore, $\text{wt}(x) \leq t$;

" $\supseteq$ ": for any $j = t + 1, \dots, n$, by Lemma 2.2 we have $e_{n,j}(x) = \binom{\text{wt}(x)}{j}$, which is equal to 0 since $j > t \geq \text{wt}(x)$. $\square$

This allows us to use the elementary symmetric polynomials to efficiently encode the weight constraint in the BSDP. Let $S_{3B} = \{e_{n,j}(x) \mid j = t + 1, \dots, n\}$. The resulting polynomial system $S_B = S_1 \cup S_2 \cup S_{3B}$ is such that the solutions $S_B = 0$ are exactly the solutions of the BSDP.

Notice that, if $x \in \mathbb{F}_2^n$ is such that $\text{wt}(x) = t < 2^r$, where $r = \lfloor \log_2 n \rfloor$, then, due to (2), we have $e_{n,j}(x) = 0$ for any $j \geq 2^r$, since $t_r = 0$ in the binary decomposition of t . Therefore, if the target weight is $t < 2^r$, it suffices to let $S_{3B} = \{e_{n,j}(x) \mid j = t + 1, \dots, 2^r\}$, since the equations corresponding to higher degrees are also satisfied. This has the effect of reducing the degree of the polynomials in the system, similarly to the model S' for the ESDP.

Concerning the complexity of this model, the same considerations as in Section 2.1 lead to focus on the degree of regularity of the symmetric ideal $(e_{n,t+1}, \dots, e_{n,n}) + Q_n$. Applying Proposition 2.5 with $u = t$, we obtain

$$d_{\text{reg}}(S_B) \leq \left\lfloor \frac{n+t}{2} \right\rfloor + 1.$$

The bound $d_{\text{reg}}(S_B) \leq k + 1$ holds also in this case, since this holds for every system that contains $n - k$ linear equations and the field equations. A bound on the solving degree can be derived from Theorem 1.1 as usual.

The idea of dropping the equation of largest degree, instead, may not lead to similar results, it introduces too many possible weights among the solutions of the polynomial system. Therefore, with the variable substitution we may lower the degree of regularity or get a dependence with the specific instance of the BSDP, but we will not get a better bound for the solving degree.

3.2 Field of size $q > 2$

If we consider a field $\mathbb{F}_q$ of size $q > 2$, the proposed model for the ESDP does not have the same efficiency, as it relies on Lemma 2.2, which is not true in the general case of $q \geq 2$. One may solve several polynomial systems, each one corresponding to a specific combination of t non-zero coefficients in $\mathbb{F}_q$, where t is the target weight, but it would soon be computationally unfeasible.

Another way may be, as in [Cam+25], to add some new variables $\{z_i\}_{i=1}^n$ which represent the non-zero positions of the vector. This was done by adding the set of polynomials $\{x_i^{q-1} - z_i \mid i = 1, \dots, n\}$ to their model. In combination with field equations $\{x_i^q - x_i \mid i = 1, \dots, n\}$, they force z_i to be 0 if $x_i = 0$, and 1 otherwise. In our case, we can slightly modify our model for the ESDP over $\mathbb{F}_2$, by adding this set of polynomials to the usual parity check encoding, the finite field equations and the weight constraint encoding, where the last one is expressed by $\{e_{n,2^i}(z) - t_i \mid i = 0, \dots, r\}$, $r = \lfloor \log_2 n \rfloor$. Since we can also add the equations $\{z_i^2 - z_i \mid i = 1, \dots, n\}$, the degree of regularity of the given system is the degree of regularity of the ideal $I_n^q = (x_i^{q-1}, z_i^2 \mid i = 1, \dots, n) + (e_{n,2^i}(z) \mid i = 0, \dots, r) + (\sum_{j=1}^n h_{i,j} x_j \mid i = 1, \dots, n-k)$. As in the proof of Proposition 2.5, one can show that every z -variable permutation of $z_1 \cdots z_{\lfloor \frac{n}{2} \rfloor + 1}$ belongs to this ideal. Therefore, in the quotient $\mathbb{F}_q[x_1, \dots, x_n, z_1, \dots, z_n]/I_n^q$, a non-zero monomial of highest degree must divide a z -variable permutation of $x_1^{q-2} \cdots x_n^{q-2} \cdot z_1 \cdots z_{\lfloor \frac{n}{2} \rfloor}$. After the x -variable substitution given by the parity check encoding, we have that the degree of regularity is bounded from above by $k(q-2) + \lfloor \frac{n}{2} \rfloor + 1$. As q increases, this bound becomes significantly larger than the one obtained in the binary case.

For the BSDP in the case $q > 2$, one may use the same model proposed in Section 3.1, with the proper field equations. The degree of regularity of any system that contains the field equations of $\mathbb{F}_q$ is bounded from above

by $k(q-1)+1$. This yields a simple, although inaccurate, bound for the solving degree of the system.

3.3 Regular Syndrome Decoding Problem

The Regular variant of the SDP, first introduced in [AFS05], is defined, in the usual setting, as follows:

Problem 4 (RSDP: Regular Syndrome Decoding Problem). Find a vector $e \in \mathbb{F}_q^n$ such that $\mathbf{H} \cdot e^\top = s^\top$ and $e = (e_1, \dots, e_t)$, where $t \mid n$, $e_\ell \in \mathbb{F}_q^{\frac{n}{t}}$ and $\text{wt}(e_\ell) = 1$ for $\ell = 1, \dots, t$.

In this variant, therefore, the solution vector e is divided into t equal-length blocks, each containing exactly one non-zero element. The model described in Section 2 can be applied as follows. If we divide the variables into t sets $X_1 = \{x_1, \dots, x_h\}, \dots, X_t = \{x_{n-h+1}, \dots, x_n\}$, where $h = \frac{n}{t}$, we can consider the weight constraint encoding over each set. For example, in the binary case, for any $\ell = 1, \dots, t$ we can impose the condition $\text{wt}(e_\ell) = 1$ by considering the set $S_{3_\ell} = \{e_{h,1}(X_\ell)+1\} \cup \{e_{h,2^i}(X_\ell) \mid i = 1, \dots, \lfloor \log_2 h \rfloor\}$.

Therefore, the set $S_R = S_1 \cup S_2 \cup \bigcup_{\ell=1}^t S_{3_\ell}$ is such that the solutions of $S_R = 0$ are the solutions of the RSDP instance.

To study the complexity of this variant model, we again consider the corresponding degree of regularity. Thanks to Lemma 2.3, we can prove that for any $\ell = 1, \dots, t$ the ideal generated by S_R^{top} contains all the elementary symmetric polynomials $e_{h,j}(X_\ell)$, $j = 1, \dots, h$, and by Lemma 2.9 it is possible to recursively construct the elementary symmetric polynomials in an ever-increasing number of variables, until reaching the polynomials in the full set of n variables. We can therefore apply Proposition 2.5 and obtain

$$d_{\text{reg}}(S_R) \leq \left\lfloor \frac{n}{2} \right\rfloor + 1.$$

This time, the degrees of the system polynomials are lower than the degree of regularity, since $h \leq \frac{n}{2}$, therefore Theorem 1.1 gives us directly that

$$d_{\text{sol}}(S_R) \leq \left\lfloor \frac{n}{2} \right\rfloor + 2.$$

Although this result is less favourable than the one for the model for the RSDP proposed by Briaud and Øygarden in [BØ23], we notice that our model could more easily be adapted to a *less regular* version of the RSDP, such as, for example, to the case of blocks of different lengths or with different weights.

References

- [Ala+25] G. Alagic et al. *Status Report on the Fourth Round of the NIST Post-Quantum Cryptography Standardization Process*. NIST Internal Report 8545. National Institute of Standards and Technology, Mar. 2025. DOI: [10.6028/NIST.IR.8545](https://doi.org/10.6028/NIST.IR.8545).
- [Ala+26] G. Alagic et al. *Status Report on the Second Round of the Additional Digital Signature Schemes for the NIST Post-Quantum Cryptography Standardization Process*. NIST Internal Report 8610. National Institute of Standards and Technology, May 2026. DOI: [10.6028/NIST.IR.8610](https://doi.org/10.6028/NIST.IR.8610).
- [AFS05] D. Augot, M. Finiasz, and N. Sendrier. “A Family of Fast Syndrome Based Cryptographic Hash Functions”. In: *Progress in Cryptology – Mycrypt 2005*. Ed. by E. Dawson and S. Vaudenay. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 64–83. DOI: [10.1007/11554868_6](https://doi.org/10.1007/11554868_6).
- [BMT78] E. Berlekamp, R. McEliece, and H. van Tilborg. “On the inherent intractability of certain coding problems (Corresp.)” In: *IEEE Transactions on Information Theory* 24.3 (1978), pp. 384–386. DOI: [10.1109/TIT.1978.1055873](https://doi.org/10.1109/TIT.1978.1055873).
- [BØ23] P. Briaud and M. Øygarden. “A New Algebraic Approach to the Regular Syndrome Decoding Problem and Implications for PCG Constructions”. In: *Advances in Cryptology – EUROCRYPT 2023*. Ed. by C. Hazay and M. Stam. Cham: Springer Nature Switzerland, 2023, pp. 391–422. DOI: [10.1007/978-3-031-30589-4_14](https://doi.org/10.1007/978-3-031-30589-4_14).
- [Buc06] B. Buchberger. “Bruno Buchberger’s PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal”. In: *Journal of Symbolic Computation* 41.3 (2006), pp. 475–511. DOI: [10.1016/j.jsc.2005.09.007](https://doi.org/10.1016/j.jsc.2005.09.007).
- [Buc+09] J. A. Buchmann, J. Ding, M. S. E. Mohamed, and W. S. A. E. Mohamed. “MutantXL: Solving Multivariate Polynomial Equations for Cryptanalysis”. In: *Symmetric Cryptography*. Ed. by H. Handschuh, S. Lucks, B. Preneel, and P. Rogaway. Vol. 9031. Dagstuhl Seminar Proceedings (DagSemProc). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2009, pp. 1–7. DOI: [10.4230/DagSemProc.09031.10](https://doi.org/10.4230/DagSemProc.09031.10).

- [Cam+25] A. Caminata, R. Cartor, A. Meneghetti, R. Mora, and A. Pellegrini. “Quadratic Modelings of Syndrome Decoding”. In: *Post-Quantum Cryptography*. Ed. by R. Niederhagen and M.-J. O. Saarinen. Cham: Springer Nature Switzerland, 2025, pp. 35–70. DOI: [10.1007/978-3-031-86599-2_2](https://doi.org/10.1007/978-3-031-86599-2_2).
- [CG21] A. Caminata and E. Gorla. “Solving Multivariate Polynomial Systems and an Invariant from Commutative Algebra”. In: *Arithmetic of Finite Fields*. Springer International Publishing, 2021, pp. 3–36. DOI: [10.1007/978-3-030-68869-1_1](https://doi.org/10.1007/978-3-030-68869-1_1).
- [CG23] A. Caminata and E. Gorla. “Solving degree, last fall degree, and related invariants”. In: *Journal of Symbolic Computation* 114 (2023), pp. 322–335. DOI: [10.1016/j.jsc.2022.05.001](https://doi.org/10.1016/j.jsc.2022.05.001).
- [Cla22] Classic McEliece Team. *Classic McEliece: conservative code-based cryptography: cryptosystem specification*. Oct. 2022. URL: <https://classic.mceliece.org/mceliece-spec-20221023.pdf>.
- [Cou+00] N. Courtois, A. Klimov, J. Patarin, and A. Shamir. “Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations”. In: *Advances in Cryptology — EUROCRYPT 2000*. Ed. by B. Preneel. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 392–407. DOI: [10.1007/3-540-45539-6_27](https://doi.org/10.1007/3-540-45539-6_27).
- [Fau02] J. C. Faugère. “A new efficient algorithm for computing Gröbner bases without reduction to zero (F5)”. In: *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*. ISSAC ’02. Lille, France: Association for Computing Machinery, 2002, pp. 75–83. DOI: [10.1145/780506.780516](https://doi.org/10.1145/780506.780516).
- [Fau99] J.-C. Faugère. “A new efficient algorithm for computing Gröbner bases (F4)”. In: *Journal of Pure and Applied Algebra* 139.1 (1999), pp. 61–88. DOI: [10.1016/S0022-4049\(99\)00005-5](https://doi.org/10.1016/S0022-4049(99)00005-5).
- [FBS04] J.-C. Faugère, M. Bardet, and B. Salvy. “On the complexity of Gröbner basis computation of semi-regular overdetermined algebraic equations”. In: 2004. URL: <https://api.semanticscholar.org/CorpusID:7982329>.
- [FTW92] G. Feng, K. Tzeng, and V. Wei. “On the generalized Hamming weights of several classes of cyclic codes”. In: *IEEE Transactions on Information Theory* 38.3 (1992), pp. 1125–1130. DOI: [10.1109/18.135653](https://doi.org/10.1109/18.135653).

- [FY79] A. Fraenkel and Y. Yesha. “Complexity of problems in games, graphs and algebraic equations”. In: *Discrete Applied Mathematics* 1.1 (1979), pp. 15–30. DOI: [10.1016/0166-218X\(79\)90012-X](https://doi.org/10.1016/0166-218X(79)90012-X).
- [GG13] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. 3rd ed. Cambridge University Press, 2013. DOI: [10.1017/CB09781139856065](https://doi.org/10.1017/CB09781139856065).
- [GV13] G. H. Golub and C. F. Van Loan. *Matrix Computations - 4th Edition*. Philadelphia, PA: Johns Hopkins University Press, 2013. URL: <https://epubs.siam.org/doi/book/10.1137/1.9781421407944>.
- [HP98] P. Heijnen and R. Pellikaan. “Generalized Hamming weights of q-ary Reed-Muller codes”. In: *IEEE Transactions on Information Theory* 44.1 (1998), pp. 181–196. DOI: [10.1109/18.651015](https://doi.org/10.1109/18.651015).
- [HP03] W. C. Huffman and V. Pless. *Fundamentals of Error-Correcting Codes*. Cambridge, U.K.: Cambridge University Press, 2003. DOI: [10.1017/CB09780511807077](https://doi.org/10.1017/CB09780511807077).
- [KR00] M. Kreuzer and L. Robbiano. *Computational Commutative Algebra*. Vol. 1. Springer Berlin, Heidelberg, 2000. DOI: [10.1007/978-3-540-70628-1](https://doi.org/10.1007/978-3-540-70628-1).
- [Lee15] K. Lee. “Bounds for generalized Hamming weights of general AG codes”. In: *Finite Fields and Their Applications* 34 (2015), pp. 265–279. ISSN: 1071-5797. DOI: [10.1016/j.ffa.2015.02.006](https://doi.org/10.1016/j.ffa.2015.02.006).
- [LCQ22] K. Li, H. Chen, and L. Qu. “Generalized hamming weights of linear codes from cryptographic functions”. In: *Advances in Mathematics of Communications* 16.4 (2022), pp. 859–877. ISSN: 1930-5346. DOI: [10.3934/amc.2022081](https://doi.org/10.3934/amc.2022081).
- [Luc91] É. Lucas. *Théorie des nombres*. Vol. 1. Digitized by Internet Archive (University of Toronto). Paris: Gauthier-Villars, 1891. URL: <https://archive.org/details/thoriedesnombr01lucauoft>.
- [McE78] R. J. McEliece. “A public-key cryptosystem based on algebraic coding theory”. In: *DSN progress report* 42-44 (1978), pp. 114–116. URL: https://tda.jpl.nasa.gov/progress_report/42-44/44N.PDF.

- [MPS23] A. Meneghetti, A. Pellegrini, and M. Sala. “On the equivalence of two post-quantum cryptographic families”. In: *Annali di Matematica Pura ed Applicata* 202.2 (Apr. 2023), pp. 967–991. DOI: [10.1007/s10231-022-01267-x](https://doi.org/10.1007/s10231-022-01267-x).
- [Moh+08] M. S. E. Mohamed, W. S. A. E. Mohamed, J. Ding, and J. Buchmann. “MXL2: Solving Polynomial Equations over $GF(2)$ Using an Improved Mutant Strategy”. In: *Post-Quantum Cryptography*. Ed. by J. Buchmann and J. Ding. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 203–215. DOI: [10.1007/978-3-540-88403-3_14](https://doi.org/10.1007/978-3-540-88403-3_14).
- [Nie86] H. Niederreiter. “Knapsack-type cryptosystems and algebraic coding theory”. In: *Problems of Control and Information Theory* 15.2 (1986), pp. 159–166. URL: <https://scispace.com/papers/knapsack-type-cryptosystems-and-algebraic-coding-theory-s94u2oxhfw>.
- [Pra62] E. Prange. “The use of information sets in decoding cyclic codes”. In: *IRE Transactions on Information Theory* 8.5 (1962), pp. 5–9. DOI: [10.1109/TIT.1962.1057777](https://doi.org/10.1109/TIT.1962.1057777).
- [Sal25] F. Salizzoni. “An upper bound for the solving degree in terms of the degree of regularity”. In: *Transactions on Mathematical Cryptology* 5.1 (Sept. 2025), pp. 1–7. URL: <https://journals.flvc.org/mathcryptology/article/view/134601>.
- [San25] R. San-José. “An Algorithm for Computing Generalized Hamming Weights and the Sage Package GHWs”. In: *ACM Transactions on Mathematical Software* 51.4 (Dec. 2025). DOI: [10.1145/3773284](https://doi.org/10.1145/3773284).
- [WGR24] V. Weger, N. Gassner, and J. Rosenthal. *A Survey on Code-Based Cryptography*. 2024. DOI: [10.48550/arXiv.2201.07119](https://doi.org/10.48550/arXiv.2201.07119).
- [Wei91] V. Wei. “Generalized Hamming weights for linear codes”. In: *IEEE Transactions on Information Theory* 37.5 (1991), pp. 1412–1418. DOI: [10.1109/18.133259](https://doi.org/10.1109/18.133259).
- [Wil90] R. M. Wilson. “A Diagonal Form for the Incidence Matrices of t -Subsets vs. k -Subsets”. In: *European Journal of Combinatorics* 11.6 (1990), pp. 609–615. DOI: [10.1016/S0195-6698\(13\)80046-7](https://doi.org/10.1016/S0195-6698(13)80046-7).