LECTURE 1
INTRODUCTION TO CODE-BASED CRYPTOGRAPHY
DECODING A RANDOM CODE

Summer School: *Introduction to Quantum-Safe Cryptography*

Thomas Debris-Alazard

July 01, 2024

Inria, École Polytechnique

THE TEAM

- Maxime Bombar (Post-doc at CWI, Netherland)

  maxime.bombar@cwi.nl

- Thomas Debris-Alazard (Researcher at Inria, France)

  thomas.debris@inria.fr

Course Content:

1. An Intractable Problem Related to Codes, Decoding
2. Random Codes
3. Information Set Decoding (ISD) Algorithms and Duals Attacks
4. Duality, Fourier Theory and Decoding Self-Reducibility (Worst-to-Average Case Reduction)
5. McEliece and Alekhnovitch Encryption' Schemes (From Original Propositions to Instantiations)

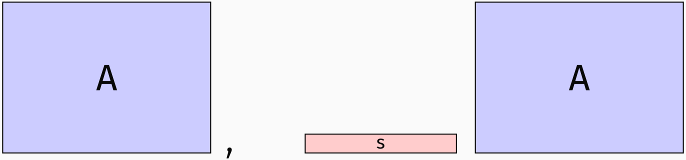
$\longrightarrow$ 3 lectures notes (long, for further reading): https://arxiv.org/pdf/2304.03541

Exercise Sessions:

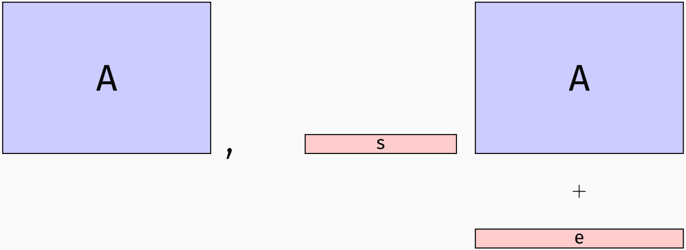
1. Starting Exercises to Get Familiar with Linear Codes & Crypto
2. Programming Session: Implement Basic ISDs and Breaking Challenges
3. Advanced Exercises About Code-Based Cryptography and Duality

$\longrightarrow$ 2 long exercise sheets: cryptanalyses of code-based encryption schemes

Code-Based Cryptography?


A
,
s
A

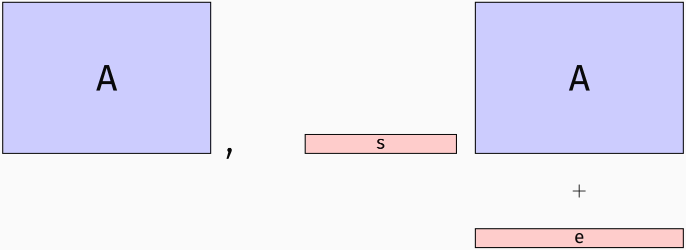
Shannon (1948/1949) introduced the following problem (decoding),



Aim:

Recover s

Shannon (1948/1949) introduced the following problem (decoding),



with few 1

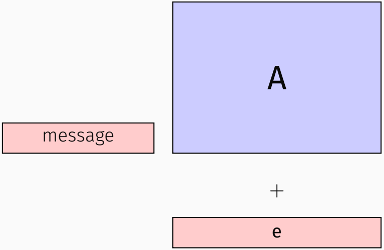
Aim:

Recover s

$\longrightarrow$ Matrix **A** and vectors **s**, **e** are binary ($\in \mathbb{F}_2$)

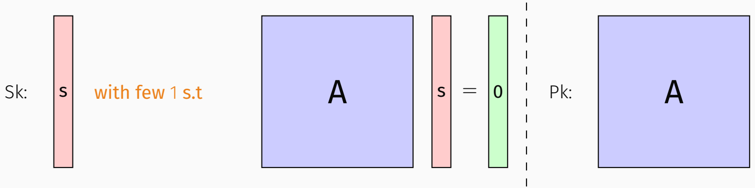
McEliece (1978):

$$\mathbf{A} \leftarrow \text{Trapdoor}(): \text{ public-key}$$

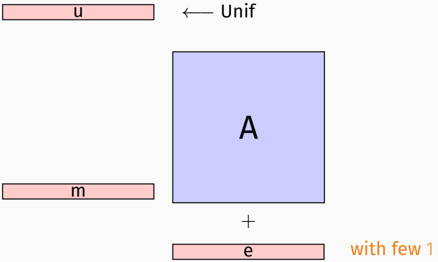


- With the trapdoor: easy to recover message if **e** "short" (with few 1, a lot of 0),
- Without: hard

Alekhnovich (2003):



- To encrypt $b = 1$, send
- To encrypt $b = 0$, send



But how to decrypt?

YOU SAID CODE?

$$s \; A \quad : \quad s \in \mathbb{F}_2^k$$

is known as a linear code!

Understanding what is a linear code: useful to

1. build trapdoors
2. understand the hardness of decoding

The first purpose of linear codes was not cryptography. . .

*It was **telecommunication**!*

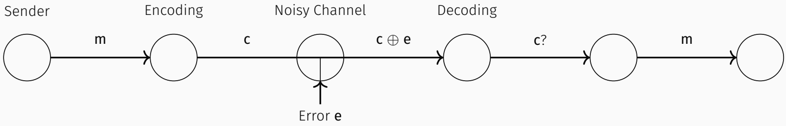
$\longrightarrow$ Codes are at the core of information theory (and friends)

How to transmit k bits over a noisy channel?

LINEAR CODES AND TELECOMMUNICATION

How to transmit k bits over a noisy channel?

1. Fix $\mathcal{C}$ subspace $\subseteq \mathbb{F}_2^n$ of dimension k
2. Map $(m_1, \ldots, m_k) \longrightarrow \mathbf{c} = (c_1, \ldots, c_n) \in \mathcal{C}$ $\big($adding $n-k$ bits redundancy$\big)$
3. Send $\mathbf{c}$ across the noisy channel



$\longrightarrow$ from $\mathbf{c} \oplus \mathbf{e}$: how to recover $\mathbf{e}$ and then $\mathbf{c}$?

$\big($Decoding Problem$\big)$

Real life scenario: $\mathbf{c}+\mathbf{e}$ with $\mathbf{e}=(e_1,\ldots,e_n)$ *such that,*

$$\forall i \in [1,n], \quad \mathbb{P}(e_i = 1) = p \text{ and } \mathbb{P}(e_i = 0) = 1-p$$

$\longrightarrow$ Each bit of $\mathbf{c}$ is flipped with probability p

Given a received corrupted word y:

$$\mathbb{P}\left(\mathbf{c} \text{ was sent} \mid \mathbf{y} \text{ is received}\right) = p^{d_{\mathrm{H}}(\mathbf{c},\mathbf{y})}(1-p)^{n-d_{\mathrm{H}}(\mathbf{c},\mathbf{y})}$$

$$\text{where } d_{\mathrm{H}}(\mathbf{c},\mathbf{y}) \stackrel{\text{def}}{=} \sharp\left\{i \in [1,n] \ : \ c_i \neq y_i\right\} \quad (\text{Hamming distance})$$

Real life scenario: $\mathbf{c}+\mathbf{e}$ with $\mathbf{e}=(e_1,\ldots,e_n)$ *such that,*

$$\forall i \in [1,n], \quad \mathbb{P}(e_i=1)=p \;\text{ and }\; \mathbb{P}(e_i=0)=1-p$$

$\longrightarrow$ Each bit of $\mathbf{c}$ is flipped with probability p

Given a received corrupted word y:

$$\mathbb{P}\left(\mathbf{c} \text{ was sent} \mid \mathbf{y} \text{ is received}\right) = p^{d_{\mathrm{H}}(\mathbf{c},\mathbf{y})}(1-p)^{n-d_{\mathrm{H}}(\mathbf{c},\mathbf{y})}$$

where $d_{\mathrm{H}}(\mathbf{c},\mathbf{y}) \stackrel{\text{def}}{=} \sharp\left\{i \in [1,n] \;:\; c_i \neq y_i\right\}$ (Hamming distance)

Any decoding candidate $\mathbf{c} \in \mathcal{C}$ is even more likely

as it is close to the received message $\mathbf{y}$ for the Hamming distance.

$\longrightarrow$ It explains why historically the Hamming distance has been the considered metric

when dealing with codes. . .

BASICS ON LINEAR CODES

$\mathbb{F}_q$: finite field with q elements

Linear Code:

A linear code $\mathcal{C}$ of length n and dimension k $\left([n,k]_q\text{-code}\right)$:

subspace of $\mathbb{F}_q^n$ of dimension k

First Examples:

1. $\left\{(f(x_1),\dots,f(x_n)) : \ f \in \mathbb{F}_q[X] \text{ and } \mathbf{deg}(f) < k\right\}$ where the x_i's are distinct elements of $\mathbb{F}_q$ is an $[n,k]_q$-code
2. $\left\{(\mathbf{u},\mathbf{u}+\mathbf{v}) : \ \mathbf{u} \in U \text{ and } \mathbf{v} \in V\right\}$ where U (*resp.* V) is an $[n,k_U]_q$-code (*resp.* $[n,k_V]_q$-code) is an $[2n, k_U+k_V]_q$-code

Hamming Weight:

Given $\mathbf{x} \in \mathbb{F}_q^n$, its Hamming weight is:

$$|\mathbf{x}| \stackrel{\text{def}}{=} \sharp\left\{i \in [1, n] : \ x_i \neq 0\right\}$$

Minimum Distance:

The minimum distance of $\mathcal{C}$ is:

$$d_{\min}(\mathcal{C}) \stackrel{\text{def}}{=} \min\left\{|\mathbf{c}| \ : \ \mathbf{c} \in \mathcal{C}, \ \mathbf{c} \neq \mathbf{0}\right\}$$

$d_{\min}(\mathcal{C})$ is an important quantity:

"geometry" of $\mathcal{C}$; "efficiency" of $\mathcal{C}$; "security" of $\mathcal{C}$

$\mathcal{C}$ be an $[n,k]_q$-code

Basis representation: $\mathbf{g}_1, \ldots, \mathbf{g}_k$ basis of $\mathcal{C}$,

$$\mathcal{C} = \left\{ \mathbf{m}\mathbf{G} : \ \mathbf{m} \in \mathbb{F}_q^k \right\} \text{ where the rows of } \mathbf{G} \in \mathbb{F}_q^{k \times n} \text{ are the } \mathbf{g}_i$$

Reciprocally, any $\mathbf{G} \in \mathbb{F}_q^{k \times n}$ of rank k defines the $[n,k]_q$-code,

$$\mathcal{C} \stackrel{\text{def}}{=} \left\{ \mathbf{m}\mathbf{G} : \ \mathbf{m} \in \mathbb{F}_q^k \right\}$$

Generator Matrix:

$\mathbf{G}$ is called a generator matrix

Dual Code:

Given $\mathcal{C}$, its dual $\mathcal{C}^{\perp}$ is the $[n, n-k]_q$-code,

$$\mathcal{C}^{\perp} \stackrel{\text{def}}{=} \left\{ \mathbf{c}^{\perp} \in \mathbb{F}_q^n : \ \forall \mathbf{c} \in \mathcal{C}, \ \mathbf{c} \cdot \mathbf{c}^{\perp} \stackrel{\text{def}}{=} \sum_{i=1}^{n} c_i \ c_i^{\perp} = 0 \in \mathbb{F}_q \right\}$$

$\longrightarrow$ Wait Lecture 4 to understand the rational behind this definition!

HOW TO REPRESENT A CODE (II)?

Dual Code:

Given $\mathcal{C}$, its dual $\mathcal{C}^{\perp}$ is the $[n, n-k]_q$-code,

$$\mathcal{C}^{\perp} \stackrel{\text{def}}{=} \left\{ \mathbf{c}^{\perp} \in \mathbb{F}_q^n : \ \forall \mathbf{c} \in \mathcal{C}, \ \mathbf{c} \cdot \mathbf{c}^{\perp} \stackrel{\text{def}}{=} \sum_{i=1}^{n} c_i \, c_i^{\perp} = 0 \in \mathbb{F}_q \right\}$$

$\longrightarrow$ Wait Lecture 4 to understand the rational behind this definition!

Parity-check representation: $\mathbf{h}_1, \dots, \mathbf{h}_{n-k}$ basis of $\mathcal{C}^{\perp}$,

$$\mathcal{C} = \left\{ \mathbf{c} \in \mathbb{F}_q^n : \ \mathbf{H}\mathbf{c}^{\mathsf{T}} = \mathbf{0} \right\} \text{ where the rows of } \mathbf{H} \in \mathbb{F}_q^{(n-k)\times n} \text{ are the } \mathbf{h}_i$$

Reciprocally, any $\mathbf{H} \in \mathbb{F}_q^{(n-k)\times n}$ of rank $n-k$ defines the $[n, k]_q$-code,

$$\mathcal{C} \stackrel{\text{def}}{=} \left\{ \mathbf{c} \in \mathbb{F}_q^n : \ \mathbf{H}\mathbf{c}^{\mathsf{T}} = \mathbf{0} \right\}$$

Parity-Check Matrix:

$\mathbf{H}$ is called a parity-check matrix

A REMARK

- $\mathbf{G} \in \mathbb{F}_q^{k \times n}$ generator matrix of $\mathcal{C}$ $\left(\textit{i.e.}, \mathcal{C} = \left\{ \mathbf{mG} : \ \mathbf{m} \in \mathbb{F}_q^k \right\} \right)$, $\mathbf{S} \in \mathbb{F}_q^{k \times k}$ non-singular,

  $\longrightarrow$ $\mathbf{SG}$ still generator matrix of $\mathcal{C}$

- $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ parity-check matrix of $\mathcal{C}$ $\left(\textit{i.e.}, \mathcal{C} = \left\{ \mathbf{c} \in \mathbb{F}_q^n : \ \mathbf{Hc}^\mathsf{T} = \mathbf{0} \right\} \right)$, $\mathbf{S} \in \mathbb{F}_q^{(n-k) \times (n-k)}$ non-singular,

  $\longrightarrow$ $\mathbf{SH}$ still parity-check matrix of $\mathcal{C}$

$\mathbf{G} \in \mathbb{F}_q^{k \times n}$ generator matrix $\xleftrightarrow{\text{easy to compute?}}$ $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ parity-check matrix

$$\mathbf{G} \in \mathbb{F}_q^{k \times n} \text{ generator matrix } \xleftrightarrow{\text{easy to compute?}} \mathbf{H} \in \mathbb{F}_q^{(n-k)\times n} \text{ parity-check matrix}$$

Yes!

1. Show that if $\mathbf{H} \in \mathbb{F}_q^{(n-k)\times n}$ has rank $n-k$ and $\mathbf{G}\mathbf{H}^{\mathsf{T}} = \mathbf{0}$, then $\mathbf{H}$ parity-check (exercise)
2. Perform a Gaussian elimination: $\mathbf{S}\mathbf{G} = \left(\mathbf{I}_k \mid \mathbf{A}\right)$, then $\mathbf{H} = \left(-\mathbf{A}^\top \mid \mathbf{I}_{n-k}\right)$ is a parity-check matrix

Would you rather choose generator or parity-check representation?

Would you rather choose generator or parity-check representation?

Sorry for the team generator matrix :(

Usually, the parity-check representation is more convenient

Let $\mathcal{C}_{\text{Ham}}$ be the $[7,4]_2$-code of generator matrix:

$$\mathbf{G} \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\mathbf{H} \stackrel{\text{def}}{=} \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

has rank 3 and verifies $\mathbf{G}\mathbf{H}^{\mathsf{T}} = \mathbf{0}$.

Let $\mathbf{c}+\mathbf{e}$ where $\begin{cases} \mathbf{c} \in \mathcal{C}_{\text{Ham}} \\ |\mathbf{e}| = 1 \end{cases}$: how to easily recover $\mathbf{e}$?

Given $\mathbf{c}+\mathbf{e}$: recover $\mathbf{e}$

$\longrightarrow$ Make modulo $\mathcal{C}$ to extract the information about $\mathbf{e}$

Coset Space: $\mathbb{F}_q^n/\mathcal{C}$

Given an $[n,k]_q$-code $\mathcal{C}$, $\quad \sharp\, \mathbb{F}_q^n/\mathcal{C} = q^{n-k}$ and $\mathbb{F}_q^n/\mathcal{C} = \left\{ \mathbf{x}_i + \mathcal{C} \ : \ 1 \leq i \leq q^{n-k} \right\}$

A natural set of representatives via a parity-check $\mathbf{H}$: syndromes

$\mathbf{x}_i + \mathcal{C} \in \mathbb{F}_q^n/\mathcal{C} \longmapsto \mathbf{H}\mathbf{x}_i^{\mathsf{T}} \in \mathbb{F}_q^{n-k}$ (called a syndrome)

is an isomorphism

$\mathcal{C}$ be an $[n, k]_q$-code of parity-check matrix $\mathbf{H}$

Noisy codeword	Syndrome
$\mathbf{c}+\mathbf{e}$	$\mathbf{H}\mathbf{e}^{\intercal}$

- From $\mathbf{c}+\mathbf{e}$: $\mathbf{H}(\mathbf{c}+\mathbf{e})^{\intercal} = \mathbf{H}\mathbf{c}^{\intercal} + \mathbf{H}\mathbf{e}^{\intercal} = \mathbf{H}\mathbf{e}^{\intercal}$
- From $\mathbf{H}\mathbf{e}^{\intercal}$: compute with linear algebra $\mathbf{y}$ s.t
$$\mathbf{H}\mathbf{y}^{\intercal} = \mathbf{H}\mathbf{e}^{\intercal} \iff \mathbf{H}(\mathbf{y}-\mathbf{e})^{\intercal} = \mathbf{0} \iff \mathbf{y}-\mathbf{e} \in \mathcal{C} \iff \mathbf{y} = \mathbf{c}+\mathbf{e}$$

THE WORST-CASE DECODING PROBLEM

Two formulations for the worst-case decoding:

Problem (Noisy Codeword Decoding):

- **Given:** $\mathbf{G} \in \mathbb{F}_q^{k \times n}$ of rank k, $t \in [0, n]$, $\mathbf{y} \in \mathbb{F}_q^n$ where $\mathbf{y} = \mathbf{c} + \mathbf{e}$ with $\mathbf{c} = \mathbf{m}\mathbf{G}$ for some $\mathbf{m} \in \mathbb{F}_q^k$ and $|\mathbf{e}| = t$
- **Find:** $\mathbf{e}$ $\left(\text{or equivalently } \mathbf{m}\right)$

Problem (Syndrome Decoding):

- **Given:** $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$ of rank $n-k$, $t \in [0, n]$, $\mathbf{s} \in \mathbb{F}_q^{n-k}$ where $\mathbf{H}\mathbf{e}^{\mathsf{T}} = \mathbf{s}^{\mathsf{T}}$ with $|\mathbf{e}| = t$
- **Find:** $\mathbf{e}$

$\longrightarrow$ These problems are equivalent!

n length ; k dimension ; t decoding distance

Let, $\mathcal{A}$ be an algorithm such that $\mathcal{A}(\mathbf{G}, \mathbf{mG} + \mathbf{e}) \longmapsto \mathbf{e}$

Given $(\mathbf{H}, \mathbf{He}^{\intercal})$: our aim, recover $\mathbf{e}$ using $\mathcal{A}$

1. Compute with linear algebra $\mathbf{G}$ (rank k) such that $\mathbf{GH}^{\intercal} = \mathbf{0}$
2. Compute (again) with linear algebra $\mathbf{y}$ such that $\mathbf{Hy}^{\intercal} = \mathbf{He}^{\intercal}$
3. Notice that $\mathbf{H}(\mathbf{y} - \mathbf{e})^{\intercal} = \mathbf{0} \iff \mathbf{y} - \mathbf{e} = \mathbf{mG}$ for some $\mathbf{m} \in \mathbb{F}_q^k$
4. Feed $(\mathbf{G}, \mathbf{y})$ to $\mathcal{A}$: it recovers $\mathbf{e}$

Exercise: show that the reciprocal holds

In what follows, we will mainly keep the parity-check representation!

Worst-Case Decisional Decoding Problem

- Input: $\mathbf{H} \in \mathbb{F}_q^{(n-k)\times n}$, $\mathbf{s} \in \mathbb{F}_q^{n-k}$ where $n, k \in \mathbb{N}$ with $k \leq n$ and an integer $t \leq n$.
- Decision: it exists $\mathbf{e} \in \mathbb{F}_q^n$ of Hamming weight t such $\mathbf{H}\mathbf{e}^{\mathsf{T}} = \mathbf{s}^{\mathsf{T}}$?

This problem is NP-complete

Is it useful?

Be careful of the input set!

The above NP-completeness shows that (if $\mathrm{P} \neq \mathrm{NP}$)

We cannot easily solve the decoding problem for all codes and all decoding distances. . .

$\longrightarrow$ There are codes for which decoding is hard!

Not a safety guarantee for cryptographic applications!

Is decoding hard for all codes?

No! (remember Hamming code. . .)

Generalized Reed-Solomon (GRS) Codes:

Given $\mathbf{z} \in (\mathbb{F}_q^{\star})^n$ and $\mathbf{x} \in \mathbb{F}_q^n$ s.t $x_i \neq x_j$ (in particular $n \leq q$) and $k \leq n$.

The code $\mathrm{GRS}_k(\mathbf{x}, \mathbf{z})$ is defined as:

$$\mathrm{GRS}_k(\mathbf{x}, \mathbf{z}) \stackrel{\text{def}}{=} \left\{ \Big(z_1 f(x_1), \ \dots, \ z_n f(x_n) \Big) \ : \ f \in \mathbb{F}_q[X] \ \text{ and } \ \mathbf{deg}(f) < k \right\}$$

$\longrightarrow$ GRS are used in QR-codes!

Exercise: $\mathrm{GRS}_k(\mathbf{x}, \mathbf{z})$ has generator matrix:

$$\mathbf{G} \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ x_1^2 & x_2^2 & \cdots & x_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^k & x_2^k & \cdots & x_n^k \end{pmatrix} \begin{pmatrix} z_1 & & & 0 \\ & z_2 & & \\ & & \ddots & \\ 0 & & & z_n \end{pmatrix}$$

Decoding Algorithm:

Given, $\mathrm{GRS}_k(\mathbf{x}, \mathbf{z})$ and $\mathbf{c} + \mathbf{e}$ such that $\begin{cases} \mathbf{c} \in \mathrm{GRS}_k(\mathbf{x}, \mathbf{z}) \\ |\mathbf{e}| \leq \left\lfloor \frac{n-k}{2} \right\rfloor \end{cases}$

Then, we can recover $(\mathbf{c}, \mathbf{e})$ in polynomial time in the size of inputs, *i.e.*, $O\left(n^\ell\right)$ for some ℓ.

$\longrightarrow$ See Exercise Session

- There are codes for which decoding is hard (NP-Completeness)
- Decoding is easy for some family of codes (for instance Generalized-Reed-Solomon codes)

Is decoding hard for almost all codes?

AVERAGE DECODING PROBLEM

THE AVERAGE DECODING PROBLEM

DP(n, q, R, τ), $k \overset{\text{def}}{=} Rn$ **and** $t \overset{\text{def}}{=} \tau n$

Sample: $\boxed{\mathbf{H}} \longleftarrow \mathrm{Unif}\left(\mathbb{F}_q^{(n-k)\times n}\right), \quad \boxed{\mathbf{x}} \longleftarrow \mathrm{Unif}\big(\mathbf{z} \ : |\mathbf{z}| = t\big)$

Input: $\boxed{\mathbf{H}}\ , \ \boxed{\mathbf{s}} = \boxed{\mathbf{H}}\ \boxed{\mathbf{x}}$

Recover: $\boxed{\mathbf{e}}$ s.t $\boxed{\mathbf{H}}\ \boxed{\mathbf{e}} = \boxed{\mathbf{s}}$ and $\boxed{\mathbf{e}} \in \left\{\mathbf{z} \ : \ |\mathbf{z}| = t\right\}$

For a fixed $R = k/n$, with respect to $\tau = t/n$, the solution will be unique or not!

$$\text{Let, } \varepsilon = \mathbb{P}_{\mathbf{H},\mathbf{x}}\Big(\mathcal{A}(\mathbf{H}, \mathbf{s} = \mathbf{x}\mathbf{H}^{\intercal}) = \mathbf{e} \ \text{ such that } \ |\mathbf{e}| = t \ \text{ and } \ \mathbf{e}\mathbf{H}^{\intercal} = \mathbf{s}\Big)$$

Using the law of total probability:

$$\varepsilon = \frac{1}{q^{k\times(n-k)}\times(q-1)^t\binom{n}{t}} \sum_{\substack{\mathbf{x}_0\in\mathbb{F}_q^n,\ |\mathbf{x}_0|=t \\ \mathbf{H}_0\in\mathbb{F}_q^{(n-k)\times n}}} \mathbb{P}\Big(\mathcal{A}(\mathbf{H}_0, \mathbf{s} = \mathbf{x}_0\mathbf{H}^{\intercal}) = \mathbf{e} \ \text{ s.t } \ |\mathbf{e}| = t \ \text{ and } \ \mathbf{e}\mathbf{H}^{\intercal} = \mathbf{s}\Big)$$

$\longrightarrow$ ε is the average success probability of $\mathcal{A}$ over all fixed possible inputs

$\Big($above probabilities are computed over the internal randomness of $\mathcal{A}\Big)$

Consequence:

If ε is negligible, then $\mathcal{A}$ fails to decode almost all codes

AVERAGE DECODING HARDNESS

Exponential Complexity for Decoding in Average:

For all known algorithms $\mathcal{A}$ (T running time of one iteration $\mathcal{A}$)

$$\frac{T}{\varepsilon} = 2^{\alpha(q,R,\tau)\ n(1+o(1))} \text{ for some } \alpha(q,R,\tau) \geq 0$$

Figure 1: Hardness of DP(n, q, R, τ) as function of τ

AVERAGE DECODING HARDNESS

Exponential Complexity for Decoding in Average:

For all known algorithms $\mathcal{A}$ (T running time of one iteration $\mathcal{A}$)

$$\frac{T}{\varepsilon} = 2^{\alpha(q,R,\tau)\ n(1+o(1))} \text{ for some } \alpha(q,R,\tau) \geq 0$$

Figure 1: Hardness of $\mathrm{DP}(n, q, R, \tau)$ as function of τ

- McEliece encryption: $t = \tau n = \Theta\left(\frac{n}{\log n}\right)$
- Other encryptions: $t = \tau n = \Theta\left(\sqrt{n}\right)$
- Authenticated protocols: $t = \tau n = Cn$ where C constant quite small
- Wave Signature: $t = \tau n = Cn$ where C large constant, $C \approx 0.95$

$\mathrm{DP}'(n, q, R, \tau)$. Let $k \stackrel{\text{def}}{=} \lfloor Rn \rfloor$ and $t \stackrel{\text{def}}{=} \lfloor \tau n \rfloor$

- **Input:** $(\mathbf{G}, \mathbf{y} \stackrel{\text{def}}{=} \mathbf{sG} + \mathbf{x})$ where $\mathbf{G}$, $\mathbf{s}$ and $\mathbf{x}$ are uniformly distributed over $\mathbb{F}_q^{k \times n}$, $\mathbb{F}_q^k$ and words of Hamming weight t in $\mathbb{F}_q^n$.
- **Output:** an error $\mathbf{e} \in \mathbb{F}_q^n$ of Hamming weight t such that $\mathbf{y} - \mathbf{e} = \mathbf{mG}$ for some $\mathbf{m} \in \mathbb{F}_q^k$.

Exercise Session:

For any algorithm $\mathcal{A}$ solving DP' with probability ε and time T:

Describe an algorithm $\mathcal{B}$ solving DP in the $\approx$ same time with probability $\geq \varepsilon - O\left(q^{-\min(k, n-k)}\right)$

(and the reciprocal)

$\longrightarrow$ Same average hardness with syndromes or noisy codewords formalism!