Not Just Regular Decoding: Asymptotics and Improvements of Regular Syndrome Decoding Attacks

Andre Esser[1], Paolo Santini[2]

[1]Technology Innovation Institute, UAE
[2] Polytechnic University of Marche, Italy

August 21, 2024
Crypto 2024

Regular Syndrome Decoding

Definition: We say that $\mathbf{e} \in \mathbb{F}_2^n$ with Hamming weight w is *regular* if

$$\mathbf{e} = \left(\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_w\right),$$

where each $\mathbf{e}_i$ has length $b = \frac{n}{w}$ and Hamming weight one.

Regular Syndrome Decoding (RSD)

Given $\mathbf{H} \in \mathbb{F}_2^{r \times n}$, $\mathbf{s} \in \mathbb{F}_2^r$ and $w \in \mathbb{N}$, find $\mathbf{e} \in \mathbb{F}_2^n$ such that

- $\mathrm{wt}(\mathbf{e}) = w$
- $\mathbf{e}$ is regular
- $\mathbf{He} = \mathbf{s}$

Regular Syndrome Decoding

Definition: We say that $\mathbf{e} \in \mathbb{F}_2^n$ with Hamming weight w is *regular* if

$$\mathbf{e} = \left(\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_w\right),$$

where each $\mathbf{e}_i$ has length $b = \frac{n}{w}$ and Hamming weight one.

Regular Syndrome Decoding (RSD)

Given $\mathbf{H} \in \mathbb{F}_2^{r \times n}$, $\mathbf{s} \in \mathbb{F}_2^r$ and $w \in \mathbb{N}$, find $\mathbf{e} \in \mathbb{F}_2^n$ such that

- $\mathrm{wt}(\mathbf{e}) = w$
- $\mathbf{e}$ is regular
- $\mathbf{He} = \mathbf{s}$

RSD has been employed in many cryptographic constructions, including:

- SHA-3 candidate FSB (Augot et al., 2003)

Regular Syndrome Decoding

Definition: We say that $\mathbf{e} \in \mathbb{F}_2^n$ with Hamming weight w is *regular* if

$$\mathbf{e} = (\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_w),$$

where each $\mathbf{e}_i$ has length $b = \frac{n}{w}$ and Hamming weight one.

Regular Syndrome Decoding (RSD)

Given $\mathbf{H} \in \mathbb{F}_2^{r \times n}$, $\mathbf{s} \in \mathbb{F}_2^r$ and $w \in \mathbb{N}$, find $\mathbf{e} \in \mathbb{F}_2^n$ such that

- $\mathrm{wt}(\mathbf{e}) = w$
- $\mathbf{e}$ is regular
- $\mathbf{He} = \mathbf{s}$

RSD has been employed in many cryptographic constructions, including:

- SHA-3 candidate FSB (Augot et al., 2003)
- MPC protocols (Hazay et al., 2018)
- Pseudorandom Correlation Generators (Boyle et al., 2018, Boyle et al. 2019)

Regular Syndrome Decoding

Definition: We say that $\mathbf{e} \in \mathbb{F}_2^n$ with Hamming weight w is *regular* if

$$\mathbf{e} = \left(\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_w\right),$$

where each $\mathbf{e}_i$ has length $b = \frac{n}{w}$ and Hamming weight one.

Regular Syndrome Decoding (RSD)

Given $\mathbf{H} \in \mathbb{F}_2^{r \times n}$, $\mathbf{s} \in \mathbb{F}_2^r$ and $w \in \mathbb{N}$, find $\mathbf{e} \in \mathbb{F}_2^n$ such that

- $\mathrm{wt}(\mathbf{e}) = w$
- $\mathbf{e}$ is regular
- $\mathbf{He} = \mathbf{s}$

RSD has been employed in many cryptographic constructions, including:

- SHA-3 candidate FSB (Augot et al., 2003)
- MPC protocols (Hazay et al., 2018)
- Pseudorandom Correlation Generators (Boyle et al., 2018, Boyle et al. 2019)
- Post-quantum signatures (Carozza et al., 2023, Hongrui et al., 2024)

Uniqueness bound

Random RSD instance:

- sample uniformly random $\mathbf{H} \in \mathbb{F}_2^{r \times n}$
- sample uniformly random regular vector $\mathbf{e} \in \mathbb{F}_2^n$ with weight w
- set $\mathbf{s} = \mathbf{He}$

Uniqueness bound

Random RSD instance:

- sample uniformly random $\mathbf{H} \in \mathbb{F}_2^{r \times n}$
- sample uniformly random regular vector $\mathbf{e} \in \mathbb{F}_2^n$ with weight w
- set $\mathbf{s} = \mathbf{He}$

Average number of solutions: $S_{RSD} = \max\left\{1 \; ; \; \left(\frac{n}{w}\right)^w 2^{-r} = b^w 2^{-r}\right\}$

Uniqueness bound

Random RSD instance:

- sample uniformly random $\mathbf{H} \in \mathbb{F}_2^{r \times n}$
- sample uniformly random regular vector $\mathbf{e} \in \mathbb{F}_2^n$ with weight w
- set $\mathbf{s} = \mathbf{He}$

Average number of solutions: $S_{RSD} = \max\left\{1 \ ; \ \left(\frac{n}{w}\right)^w 2^{-r} = b^w 2^{-r}\right\}$

Let $r = (1 - \kappa) \cdot n$, with $\kappa \in [0; 1]$ being the code rate

Uniqueness bound

Let $w = \omega n$ with $\omega \in [0 \ ; \ 0.5]$, $\omega = \frac{1}{b}$ and $b \in \mathbb{N}$.
We have $S_{RSD} = 1$ if

$$-\omega \cdot \log_2(\omega) \leq 1 - \kappa.$$

Analogue of Gilbert-Varshamov (GV) bound for the standard Syndrome Decoding (SD)

Uniqueness bound

Random RSD instance:

- sample uniformly random $\mathbf{H} \in \mathbb{F}_2^{r \times n}$
- sample uniformly random regular vector $\mathbf{e} \in \mathbb{F}_2^n$ with weight w
- set $\mathbf{s} = \mathbf{He}$

Average number of solutions: $S_{RSD} = \max\left\{1\ ;\ \left(\frac{n}{w}\right)^w 2^{-r} = b^w 2^{-r}\right\}$

Let $r = (1 - \kappa) \cdot n$, with $\kappa \in [0; 1]$ being the code rate

Uniqueness bound

Let $w = \omega n$ with $\omega \in [0\ ;\ 0.5]$, $\omega = \frac{1}{b}$ and $b \in \mathbb{N}$.
We have $S_{RSD} = 1$ if

$$-\omega \cdot \log_2(\omega) \leq 1 - \kappa.$$

Analogue of Gilbert-Varshamov (GV) bound for the standard Syndrome Decoding (SD)

Solving RSD

Information Set Decoding (ISD) is known to be the best solver for SD

Hazay et al., 2018: even if tailored to the RSD setting, ISD obtains about the same complexity as direct SD attacks

Liu et al., 2022: standard ISD attacks perform best for most of the suggested parameters

Solving RSD

Information Set Decoding (ISD) is known to be the best solver for SD

Hazay et al., 2018: even if tailored to the RSD setting, ISD obtains about the same complexity as direct SD attacks

Liu et al., 2022: standard ISD attacks perform best for most of the suggested parameters

Solving RSD

Information Set Decoding (ISD) is known to be the best solver for SD

Hazay et al., 2018: even if tailored to the RSD setting, ISD obtains about the same complexity as direct SD attacks

Liu et al., 2022: standard ISD attacks perform best for most of the suggested parameters

It seems regularity cannot be used to speed-up attacks...

Solving RSD

Information Set Decoding (ISD) is known to be the best solver for SD

Hazay et al., 2018: even if tailored to the RSD setting, ISD obtains about the same complexity as direct SD attacks

Liu et al., 2022: standard ISD attacks perform best for most of the suggested parameters

It seems regularity cannot be used to speed-up attacks...

Briaud and Øygarden, 2023: algebraic-based solvers, cryptanalysis of many RSD instances

Carozza, Couteau and Joux, 2023:

- New solvers (**CCJ-Linearization**, **CCJ-Enum**, **CCJ-MO**)
- Comparison of SD and RSD using number of solutions

Solving RSD

Information Set Decoding (ISD) is known to be the best solver for SD

Hazay et al., 2018: even if tailored to the RSD setting, ISD obtains about the same complexity as direct SD attacks

Liu et al., 2022: standard ISD attacks perform best for most of the suggested parameters

It seems regularity cannot be used to speed-up attacks...

Briaud and Øygarden, 2023: algebraic-based solvers, cryptanalysis of many RSD instances

Carozza, Couteau and Joux, 2023:

- New solvers (**CCJ-Linearization**, **CCJ-Enum**, **CCJ-MO**)
- Comparison of SD and RSD using number of solutions

Our contributions

Regular-ISD algorithms

- Design of new algorithms (**Perm**, **Enum**, **Rep**, **Rep-MO**)
- Fastest solvers for worst case RSD instances
- Cryptanalysis of many RSD-based schemes

Our contributions

Regular-ISD algorithms

- Design of new algorithms (**Perm**, **Enum**, **Rep**, **Rep-MO**)
- Fastest solvers for worst case RSD instances
- Cryptanalysis of many RSD-based schemes

Towards a rigorous hardness classification

- Comparison with SD using (also) costs of known algorithms
- Identification of easy regimes (low dimension and exponentially many solutions)

Our contributions

Regular-ISD algorithms

- Design of new algorithms (**Perm**, **Enum**, **Rep**, **Rep-MO**)
- Fastest solvers for worst case RSD instances
- Cryptanalysis of many RSD-based schemes

Towards a rigorous hardness classification

- Comparison with SD using (also) costs of known algorithms
- Identification of easy regimes (low dimension and exponentially many solutions)
- Identification of regimes in which RSD is harder than SD
- Worst case RSD is harder than worst case SD

Our contributions

Regular-ISD algorithms

- Design of new algorithms (**Perm**, **Enum**, **Rep**, **Rep-MO**)
- Fastest solvers for worst case RSD instances
- Cryptanalysis of many RSD-based schemes

Towards a rigorous hardness classification

- Comparison with SD using (also) costs of known algorithms
- Identification of easy regimes (low dimension and exponentially many solutions)
- Identification of regimes in which RSD is harder than SD
- Worst case RSD is harder than worst case SD

Artifact

- Estimator for concrete and asymptotics costs
- Proof-of-concept implementation of **Perm** and **Enum**

Regularity-encoding parity-check equations

Technique already used in Briaud and Øygarden, 2023 and Carozza, Couteau, Joux, 2023

Any RSD instance $\{\mathbf{H}, \mathbf{s}, w\}$ can be transformed into a new RSD instance $\{\mathbf{H}', \mathbf{s}', w\}$ by encoding regularity:

$$\mathbf{H}' = \begin{array}{c} \begin{pmatrix} \multicolumn{4}{c}{\mathbf{H}} \\ \hline 1\;1\cdots1 & & & \\ & 1\;1\cdots1 & & \\ & & \ddots & \\ & & & 1\;1\cdots1 \end{pmatrix} \\ \begin{array}{cccc} \xleftrightarrow{\quad b \quad} & \xleftrightarrow{\quad b \quad} & \qquad & \xleftrightarrow{\quad b \quad} \end{array} \end{array} \qquad \mathbf{s}' = \begin{pmatrix} \mathbf{s} \\ \hline 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \updownarrow w$$

Regularity-encoding parity-check equations

Technique already used in Briaud and Øygarden, 2023 and Carozza, Couteau, Joux, 2023

Any RSD instance $\{\mathbf{H}, \mathbf{s}, w\}$ can be transformed into a new RSD instance $\{\mathbf{H}', \mathbf{s}', w\}$ by encoding regularity:

$$\mathbf{H}' = \begin{pmatrix} & & \mathbf{H} & & \\ \hline 1\ 1 \cdots 1 & & & & \\ & 1\ 1 \cdots 1 & & & \\ & & \ddots & & \\ & & & 1\ 1 \cdots 1 & \end{pmatrix} \qquad \mathbf{s}' = \begin{pmatrix} \mathbf{s} \\ \hline 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \updownarrow w$$

$$\underbrace{\quad}_{b} \quad \underbrace{\quad}_{b} \quad \cdots \quad \underbrace{\quad}_{b}$$

New code dimension: $k' = n - r - w$

New code rate: $\kappa' = \max\left\{\kappa - \frac{w}{n}\ ;\ 0\right\}$ (with large probability)

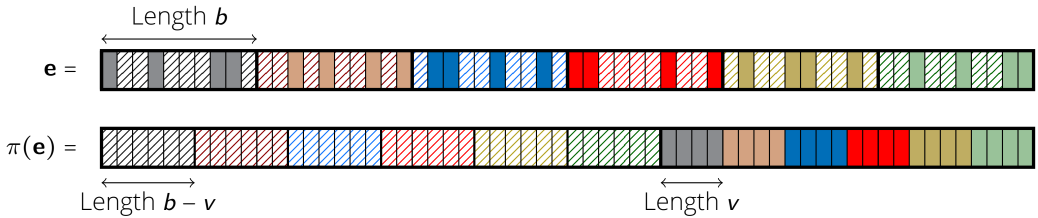
Regular permutations

Definition: Let $\mathbf{e} = (\mathbf{e}_1, \ldots, \mathbf{e}_w) \in (\mathbb{F}_2^b)^w$. For an integer $v \le b$ and a permutation matrix $\mathbf{P}$ let

$$\mathbf{Pe} = (\mathbf{e}'_1, \ldots, \mathbf{e}'_w, \mathbf{e}''_1, \ldots, \mathbf{e}''_w),$$

with $\mathbf{e}'_i \in \mathbb{F}_2^{b-v}$ and $\mathbf{e}''_i \in \mathbb{F}_2^v$. We call $\mathbf{P}$ a v-regular permutation if each $\mathbf{e}'_i$ and each $\mathbf{e}''_i$ are formed only by coordinates from $\mathbf{e}_i$.

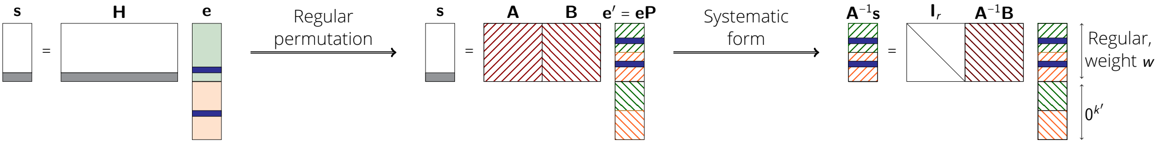
Example: $w = 6$, $b = 10$, $v = 4$



Perm: Permutation-Based Regular ISD

Adaptation of Prange's ISD to the regular setting, using regularity encoding parity-checks and regular permutation

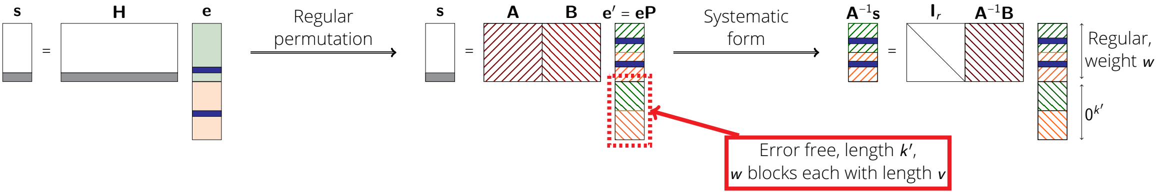
The information set is constituted by sampling $v := \frac{k'}{w} = \frac{n-r-w}{w}$ coordinates from each block



Perm: Permutation-Based Regular ISD

Adaptation of Prange's ISD to the regular setting, using regularity encoding parity-checks and regular permutation

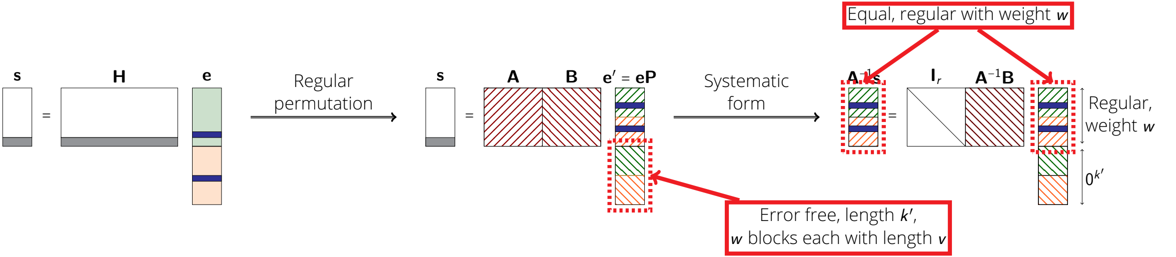
The information set is constituted by sampling $v := \frac{k'}{w} = \frac{n-r-w}{w}$ coordinates from each block



Perm: Permutation-Based Regular ISD

Adaptation of Prange's ISD to the regular setting, using regularity encoding parity-checks and regular permutation

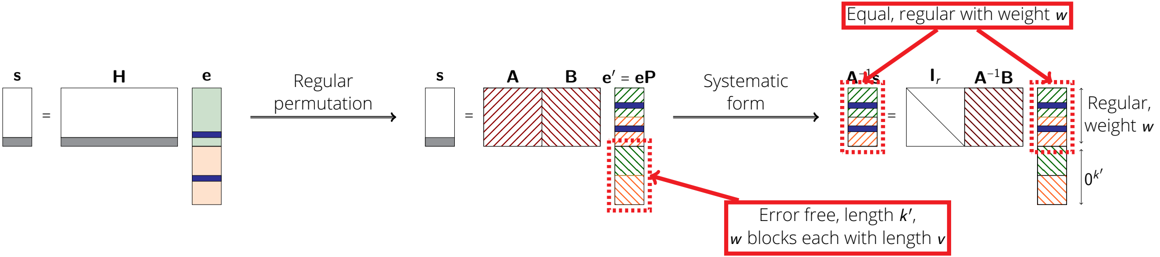
The information set is constituted by sampling $v := \frac{k'}{w} = \frac{n-r-w}{w}$ coordinates from each block



Perm: Permutation-Based Regular ISD

Adaptation of Prange's ISD to the regular setting, using regularity encoding parity-checks and regular permutation

The information set is constituted by sampling $v := \frac{k'}{w} = \frac{n-r-w}{w}$ coordinates from each block



Rounding issues: if v is not integer: select $\lfloor v \rfloor$ from some blocks, $\lceil v \rceil$ from the other blocks. Very mild impact on complexity

Advanced Regular ISD algorithms

Translation of advanced techniques from the SD setting. Rounding issues have mild impact on complexity

Best solvers for worst case RSD instances (asymptotic time complexity expressed as $T = 2^{cn}$):

- CCJ-MO: $c = 0.1281$
- Rep-MO: $c = 0.1117$ (0.1119 after resolving rounding issues)

Advanced Regular ISD algorithms

Translation of advanced techniques from the SD setting. Rounding issues have mild impact on complexity

Best solvers for worst case RSD instances (asymptotic time complexity expressed as $T = 2^{cn}$):

- CCJ-MO: $c = 0.1281$
- Rep-MO: $c = 0.1117$ (0.1119 after resolving rounding issues)

Advanced Regular ISD algorithms

Translation of advanced techniques from the SD setting. Rounding issues have mild impact on complexity

Best solvers for worst case RSD instances (asymptotic time complexity expressed as $T = 2^{cn}$):

- CCJ-MO: $c = 0.1281$
- Rep-MO: $c = 0.1117$ (0.1119 after resolving rounding issues)

For many RSD-based schemes, regular-ISD algorithms result in the fastest attacks

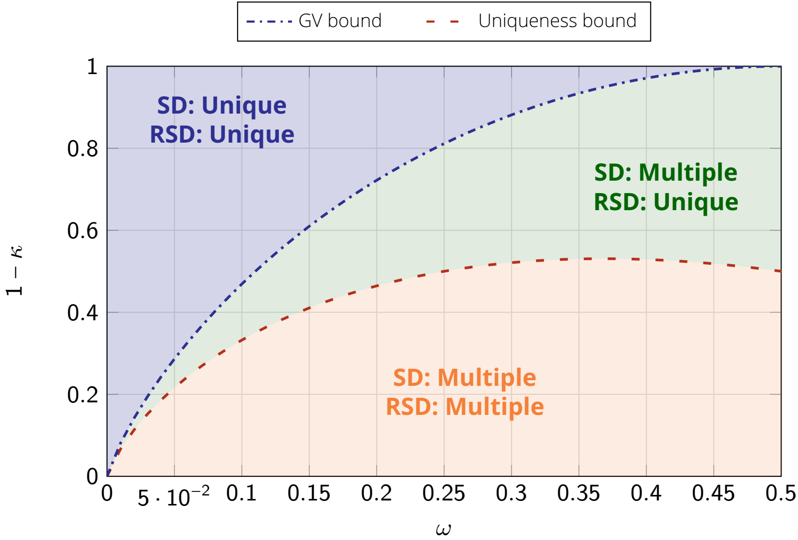
source	(n, k, w)	previous best	regular-ISD
Hazay et al., 2018	$(1280, 860, 80)$	132	114
Liu et al., 2024	$(2^{10}, 652, 57)$	90	76
	$(2^{10}, 652, 106)$	129	113
	$(2^{12}, 1589, 172)$	132	109
	$(2^{14}, 3482, 338)$	135	118
	$(2^{16}, 7391, 667)$	139	126
Carozza et al., 2024	$(1842, 825, 307)$	183	153

Table: Bit security for selected instances considering regular-ISD in comparison to previous best approaches.

Hardness classification: SD vs RSD

Comparison of hardness to solve SD and RSD with same parameters (n, κ, ω)

Number of solutions:
SD is harder than RSD when solution is unique

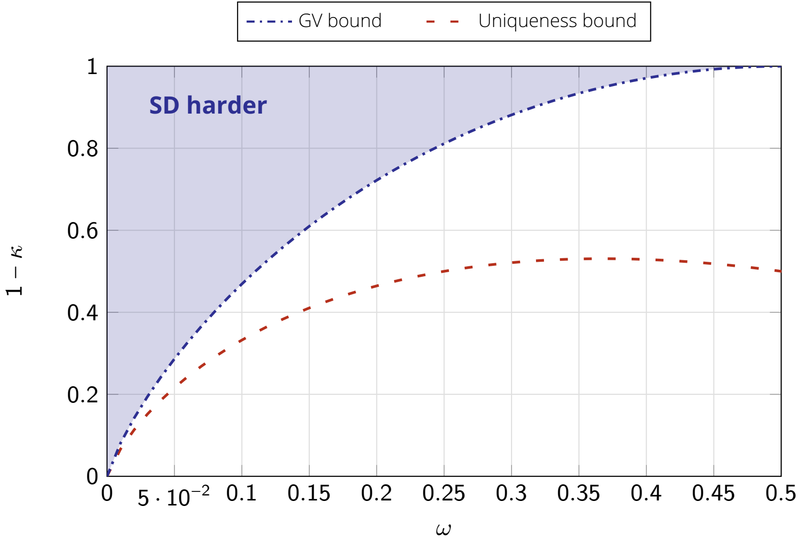


Hardness classification: SD vs RSD

Comparison of hardness to solve SD and RSD with same parameters (n, κ, ω)

Number of solutions:

SD is harder than RSD when solution is unique



Hardness classification: SD vs RSD

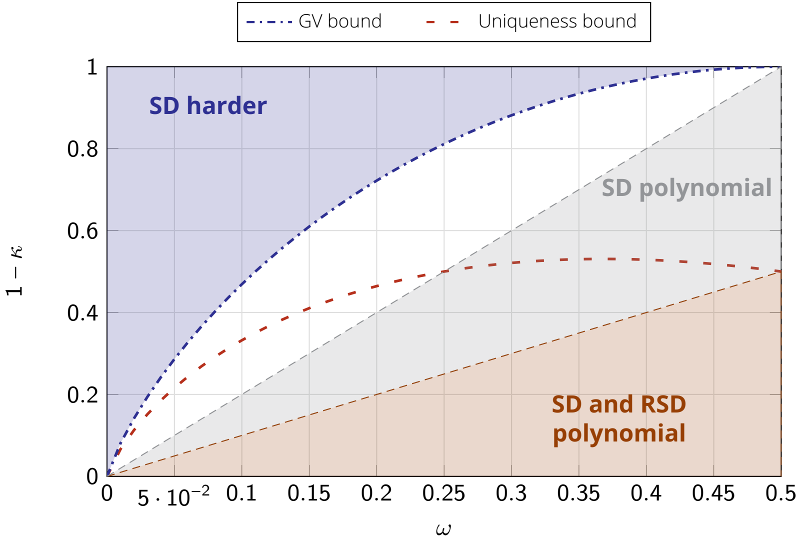
Comparison of hardness to solve SD and RSD with same parameters (n, κ, ω)

Number of solutions:

SD is harder than RSD when solution is unique

Easy RSD Regimes

- $\kappa \geq \frac{1}{2}$ and $\omega \geq 1 - \kappa$: Exponentially many solutions



Hardness classification: SD vs RSD

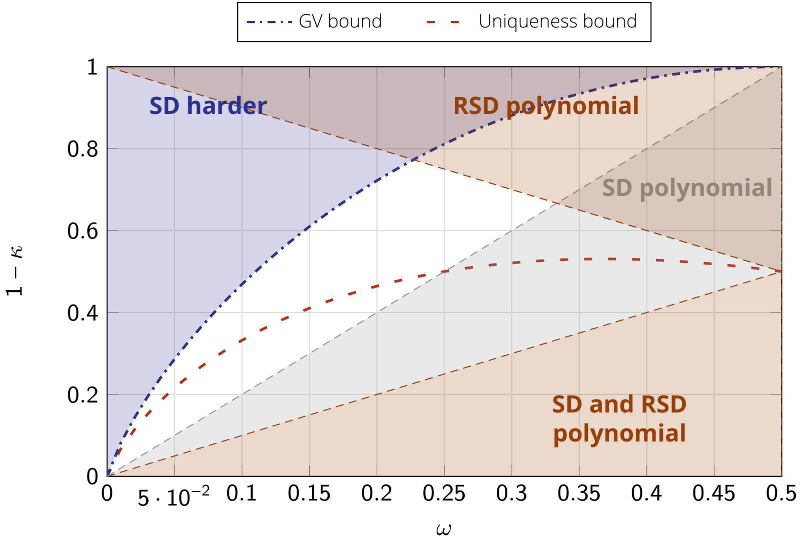
Comparison of hardness to solve SD and RSD with same parameters (n, κ, ω)

Number of solutions:

SD is harder than RSD when solution is unique

Easy RSD Regimes

- $\kappa \geq \frac{1}{2}$ and $\omega \geq 1 - \kappa$: Exponentially many solutions
- $\omega \geq \kappa$: $\kappa' = 0$ with large probability



Hardness classification: SD vs RSD

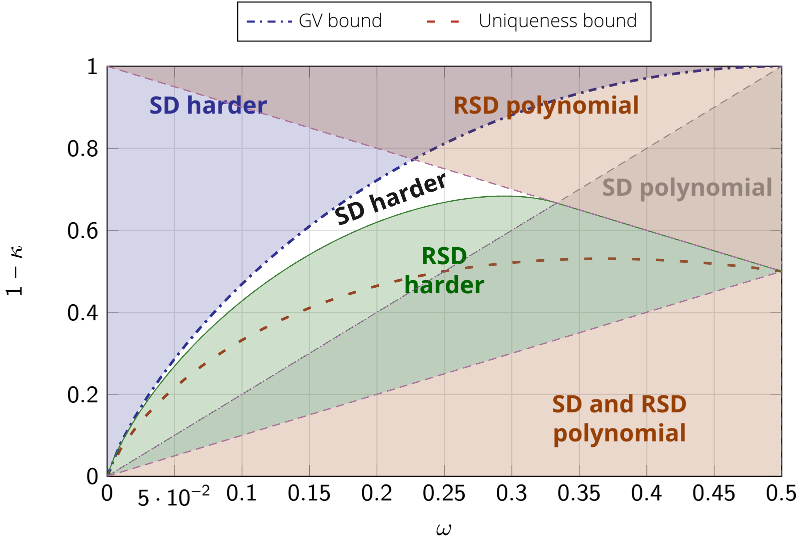
Comparison of hardness to solve SD and RSD with same parameters (n, κ, ω)

Number of solutions:

SD is harder than RSD when solution is unique

Easy RSD Regimes

- $\kappa \geq \frac{1}{2}$ and $\omega \geq 1 - \kappa$:
  Exponentially many solutions
- $\omega \geq \kappa$:
  $\kappa' = 0$ with large probability



Hardness classification: worst case RSD instances

Worst case RSD instances: $w = \omega^* n$, with $\omega^* \approx \min\left\{\frac{\kappa}{2}\ ;\ UB(\kappa)\right\}$

Worst case RSD is harder than worst case SD

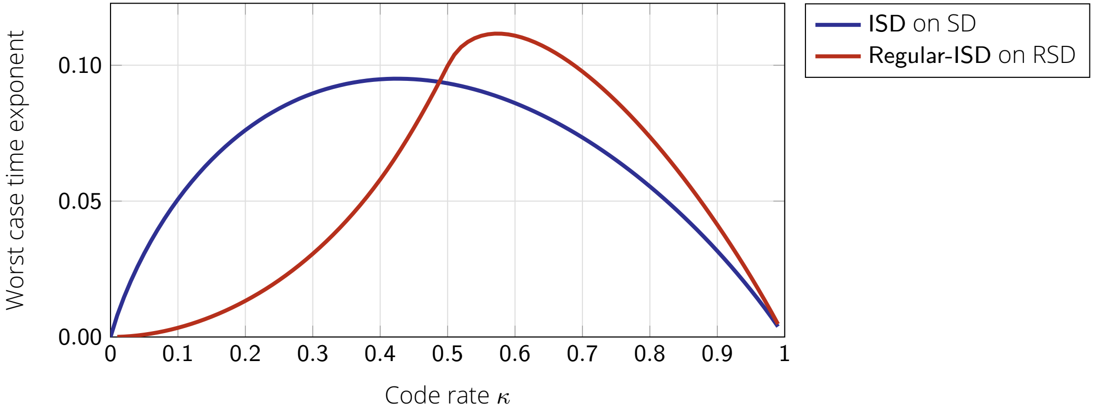


Figure: Comparison of running time of best ISD algorithm on worst case SD instances and best regular-ISD algorithms on RSD worst case instances

Hardness classification: worst case RSD instances

Worst case RSD instances: $w = \omega^* n$, with $\omega^* \approx \min\left\{\frac{\kappa}{2}\ ;\ UB(\kappa)\right\}$

Worst case RSD is harder than worst case SD

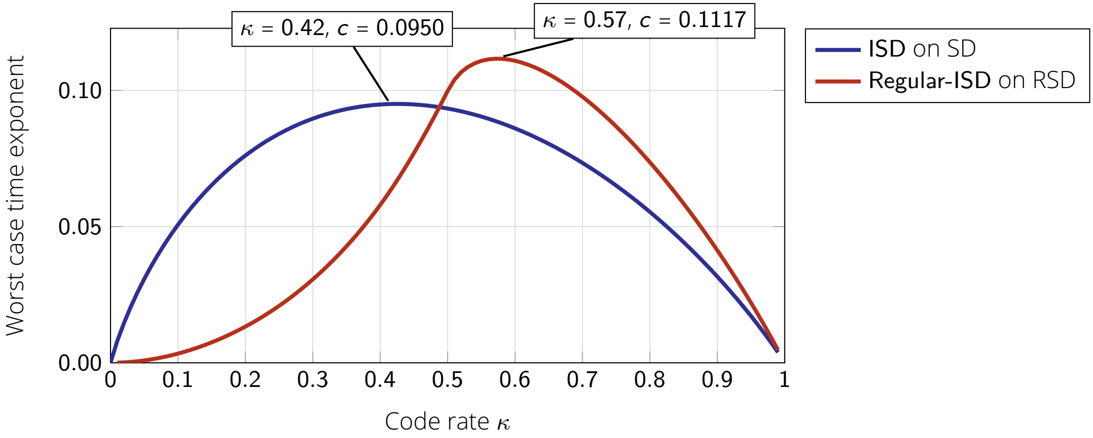


Figure: Comparison of running time of best ISD algorithm on worst case SD instances and best regular-ISD algorithms on RSD worst case instances

Conclusions

Regular-ISD: translation of ISD from the standard case to the regular case

Regular-ISD algorithms setting are the best solvers for RSD in many concrete applications and for worst case instances

Hardness classification for RSD and how to choose worst case RSD instances

Worst case RSD are harder-to-solve than worst case SD for all code rates approximately ≥ 0.5

Full version: **`Eprint 2023/1568`**

Conclusions

Regular-ISD: translation of ISD from the standard case to the regular case

Regular-ISD algorithms setting are the best solvers for RSD in many concrete applications and for worst case instances

Hardness classification for RSD and how to choose worst case RSD instances

Worst case RSD are harder-to-solve than worst case SD for all code rates approximately ≥ 0.5

Full version: `Eprint 2023/1568`

Conclusions

Regular-ISD: translation of ISD from the standard case to the regular case

Regular-ISD algorithms setting are the best solvers for RSD in many concrete applications and for worst case instances

Hardness classification for RSD and how to choose worst case RSD instances

Worst case RSD are harder-to-solve than worst case SD for all code rates approximately ≥ 0.5

Full version: `Eprint 2023/1568`

Conclusions

Regular-ISD: translation of ISD from the standard case to the regular case

Regular-ISD algorithms setting are the best solvers for RSD in many concrete applications and for worst case instances

Hardness classification for RSD and how to choose worst case RSD instances

Worst case RSD are harder-to-solve than worst case SD for all code rates approximately ≥ 0.5

Full version: `Eprint 2023/1568`

THANKS FOR THE ATTENTION ;)