A Rate-Distortion Perspective on Multiple Decoding of Reed-Solomon Codes

Phong S. Nguyen
(joint work with H. D. Pfister & K. R. Narayanan)

Department of Electrical & Computer Engineering

Texas A&M University

Brief background on Reed-Solomon (RS) codes

- RS codes are one of the most widely used error-correcting codes in digital communication and data storage.
- (n, k) RS codes have rate $\frac{k}{n}$ and minimum distance $d_{min} = n - k + 1$ which is the best possible (MDS).
- Can correct both random and burst errors.
- Efficient conventional hard-decision (HD) decoding algorithms such as Berlekamp-Massey (BM) can correct up to half d_{min}.

Brief background on Reed-Solomon (RS) codes

- RS codes are one of the most widely used error-correcting codes in digital communication and data storage.
- (n, k) RS codes have rate $\frac{k}{n}$ and minimum distance $d_{min} = n - k + 1$ which is the best possible (MDS).
- Can correct both random and burst errors.
- Efficient conventional hard-decision (HD) decoding algorithms such as Berlekamp-Massey (BM) can correct up to half d_{min}.

Brief background on Reed-Solomon (RS) codes

- RS codes are one of the most widely used error-correcting codes in digital communication and data storage.
- (n, k) RS codes have rate $\frac{k}{n}$ and minimum distance $d_{min} = n - k + 1$ which is the best possible (MDS).
- Can correct both random and burst errors.
- Efficient conventional hard-decision (HD) decoding algorithms such as Berlekamp-Massey (BM) can correct up to half d_{min}.

Brief background on Reed-Solomon (RS) codes

- RS codes are one of the most widely used error-correcting codes in digital communication and data storage.
- (n, k) RS codes have rate $\frac{k}{n}$ and minimum distance $d_{min} = n - k + 1$ which is the best possible (MDS).
- Can correct both random and burst errors.
- Efficient conventional hard-decision (HD) decoding algorithms such as Berlekamp-Massey (BM) can correct up to half d_{min}.

Decoding algorithms of RS codes

- Researchers have put a considerable effort into improving the decoding performance at the expense of complexity.
  - Guruswami-Sudan (GS) - can correct errors beyond half d_{min}.
  - Koetter-Vardy (KV) - algebraic soft-decision decoding (ASD) to achieve a substantial coding gain.
- Both algorithms, however, have significant computational complexity.

Decoding algorithms of RS codes

- Researchers have put a considerable effort into improving the decoding performance at the expense of complexity.
    - Guruswami-Sudan (GS) - can correct errors beyond half d_{min}.
    - Koetter-Vardy (KV) - algebraic soft-decision decoding (ASD) to achieve a substantial coding gain.
- Both algorithms, however, have significant computational complexity.

Decoding algorithms of RS codes

- Researchers have put a considerable effort into improving the decoding performance at the expense of complexity.
    - Guruswami-Sudan (GS) - can correct errors beyond half d_{min}.
    - Koetter-Vardy (KV) - algebraic soft-decision decoding (ASD) to achieve a substantial coding gain.
- Both algorithms, however, have significant computational complexity.

Decoding algorithms of RS codes

- Researchers have put a considerable effort into improving the decoding performance at the expense of complexity.
    - Guruswami-Sudan (GS) - can correct errors beyond half d_{min}.
    - Koetter-Vardy (KV) - algebraic soft-decision decoding (ASD) to achieve a substantial coding gain.
- Both algorithms, however, have significant computational complexity.

Motivation

- Multiple runs of some low complexity algorithm.
- For example: multiple runs of error-and-erasure decoding (BM algorithm), each time with a different set of erasure patterns.

Definition

- $\hat{x}^n \in \mathbb{Z}_2^n \triangleq \{0,1\}^n$ as an erasure pattern: at index i,

$$\hat{x}_i = \begin{cases} 0, & \text{if symbol is erased} \\ 1, & \text{if symbol is not erased} \end{cases}$$

- $x^n \in \mathbb{Z}_2^n$ as an error pattern: at index i,

$$x_i = \begin{cases} 0, & \text{if error occurs} \\ 1, & \text{if error does not occur} \end{cases}$$

Motivation

- Multiple runs of some low complexity algorithm.
- For example: multiple runs of error-and-erasure decoding (BM algorithm), each time with a different set of erasure patterns.

Definition

- $\hat{x}^n \in \mathbb{Z}_2^n \triangleq \{0,1\}^n$ as an erasure pattern: at index i,

$$\hat{x}_i = \begin{cases} 0, & \text{if symbol is erased} \\ 1, & \text{if symbol is not erased} \end{cases}$$

- $x^n \in \mathbb{Z}_2^n$ as an error pattern: at index i,

$$x_i = \begin{cases} 0, & \text{if error occurs} \\ 1, & \text{if error does not occur} \end{cases}$$

Motivation

- Multiple runs of some low complexity algorithm.
- For example: multiple runs of error-and-erasure decoding (BM algorithm), each time with a different set of erasure patterns.

Definition

- $\hat{x}^n \in \mathbb{Z}_2^n \triangleq \{0,1\}^n$ as an erasure pattern: at index i,

$$\hat{x}_i = \begin{cases} 0, \text{ if symbol is erased} \\ 1, \text{ if symbol is not erased} \end{cases}$$

- $x^n \in \mathbb{Z}_2^n$ as an error pattern: at index i,

$$x_i = \begin{cases} 0, \text{ if error occurs} \\ 1, \text{ if error does not occur} \end{cases}$$

Motivation

- Multiple runs of some low complexity algorithm.
- For example: multiple runs of error-and-erasure decoding (BM algorithm), each time with a different set of erasure patterns.

Definition

- $\hat{x}^n \in \mathbb{Z}_2^n \triangleq \{0,1\}^n$ as an erasure pattern: at index i,

$$\hat{x}_i = \begin{cases} 0, & \text{if symbol is erased} \\ 1, & \text{if symbol is not erased} \end{cases}$$

- $x^n \in \mathbb{Z}_2^n$ as an error pattern: at index i,

$$x_i = \begin{cases} 0, & \text{if error occurs} \\ 1, & \text{if error does not occur} \end{cases}$$

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED$(3,2)$

111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...

001111...

000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

$\mathrm{SED}(l, f)$ repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...

001111...

010111...

100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern
001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD
111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)
111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern
001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD
111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3, 2)
111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...

001111...

000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...

001111...

010111...

100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

$\mathrm{SED}(l, f)$ repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111...

001111...

000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111...

001111...

010111...

100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern

001011 ...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD

111111 ...

001111 ...

000011 ...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)

111111 ...

001111 ...

010111 ...

100111 ...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Motivation

- Sort the codeword positions in increasing reliability order.

Error pattern
001011...

- Generalized Minimum Distance (GMD) decoding [Forney]

Repeat error-and-erasure decoding while successively erasing an even number of the least reliable positions (LRPs).

Erasure patterns of GMD
111111...
001111...
000011...

- Successive Error-and-erasure Decoding (SED) [Lee-Kumar]

SED(l, f) repeats error-and-erasure decoding with every combination of an even number $\leq f$ of erasures within the l LRPs.

Erasure patterns of SED(3,2)
111111...
001111...
010111...
100111...

- **Motivation:** How can one construct the "best" set of erasure patterns (in the sense of performance vs complexity)?

Main results

- Design a R-D framework to analyze the asymptotic trade-off between performance vs complexity of multiple error-and-erasure decoding.
- The framework is extended to analyze multiple ASD decoding.
- Propose a family of multiple-decoding algorithms are that achieve better performance-vs-complexity trade-off than other algorithms.
- The algorithm that achieves the best trade-off uses the set of patterns generated by random codes combined with covering codes.

Main results

- Design a R-D framework to analyze the asymptotic trade-off between performance vs complexity of multiple error-and-erasure decoding.
- The framework is extended to analyze multiple ASD decoding.
- Propose a family of multiple-decoding algorithms are that achieve better performance-vs-complexity trade-off than other algorithms.
- The algorithm that achieves the best trade-off uses the set of patterns generated by random codes combined with covering codes.

Main results

- Design a R-D framework to analyze the asymptotic trade-off between performance vs complexity of multiple error-and-erasure decoding.
- The framework is extended to analyze multiple ASD decoding.
- Propose a family of multiple-decoding algorithms are that achieve better performance-vs-complexity trade-off than other algorithms.
- The algorithm that achieves the best trade-off uses the set of patterns generated by random codes combined with covering codes.

Main results

- Design a R-D framework to analyze the asymptotic trade-off between performance vs complexity of multiple error-and-erasure decoding.
- The framework is extended to analyze multiple ASD decoding.
- Propose a family of multiple-decoding algorithms are that achieve better performance-vs-complexity trade-off than other algorithms.
- The algorithm that achieves the best trade-off uses the set of patterns generated by random codes combined with covering codes.

Multiple error-and-erasure decoding

Single error-and-erasure decoding threshold

Consider an (n, k) RS code. If e symbols are erased, the BM algorithm can correct v errors in unerased positions if $2v + e < n - k + 1$.

- A multiple error-and-erasure decoding is considered to succeed if the decoding threshold is satisfied for at least one round of decoding.

- **Our idea**: connect to a R-D (covering) problem where the multiple-decoding succeeds if the error pattern ■ is covered by at least one ball centered at an erasure pattern •

Multiple error-and-erasure decoding

Single error-and-erasure decoding threshold

Consider an (n, k) RS code. If e symbols are erased, the BM algorithm can correct v errors in unerased positions if $2v + e < n - k + 1$.

- A multiple error-and-erasure decoding is considered to succeed if the decoding threshold is satisfied for at least one round of decoding.

- **Our idea**: connect to a R-D (covering) problem where the multiple-decoding succeeds if the error pattern ■ is covered by at least one ball centered at an erasure pattern •

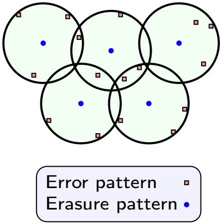
Multiple error-and-erasure decoding

Single error-and-erasure decoding threshold

Consider an (n, k) RS code. If e symbols are erased, the BM algorithm can correct v errors in unerased positions if $2v + e < n - k + 1$.

- A multiple error-and-erasure decoding is considered to succeed if the decoding threshold is satisfied for at least one round of decoding.

- **Our idea**: connect to a R-D (covering) problem where the multiple-decoding succeeds if the error pattern ■ is covered by at least one ball centered at an erasure pattern •



Distortion measure

Definition

Given a *letter-by-letter* distortion measure δ, the distortion between and error pattern x^n and an erasure pattern $\hat{x}^n$ is $d(x^n, \hat{x}^n) = \sum_{i=1}^{n} \delta(x_i, \hat{x}_i)$.

Proposition

If we choose $\delta : \mathbb{Z}_2^n \times \mathbb{Z}_2^n \to \mathbb{R}_{\geq 0}$ as follows

$$\begin{array}{ll} \delta(0,0) = 1 & \delta(0,1) = 2 \\ \delta(1,0) = 1 & \delta(1,1) = 0 \end{array}$$

	Erasure	
Error	Yes (0)	No (1)
Yes (0)	1	2
No (1)	1	0

then $2v + e < n - k + 1$ reduces to the form $d(x^n, \hat{x}^n) < n - k + 1$.

- **Question:** How many decoding attempts are needed to achieve a fixed distortion threshold?

Distortion measure

Definition

Given a *letter-by-letter* distortion measure δ, the distortion between and error pattern x^n and an erasure pattern $\hat{x}^n$ is $d(x^n, \hat{x}^n) = \sum_{i=1}^{n} \delta(x_i, \hat{x}_i)$.

Proposition

If we choose $\delta : \mathbb{Z}_2^n \times \mathbb{Z}_2^n \longrightarrow \mathbb{R}_{\geq 0}$ as follows

$$\delta(0,0) = 1 \quad \delta(0,1) = 2$$
$$\delta(1,0) = 1 \quad \delta(1,1) = 0$$

Error	Erasure: Yes (0)	Erasure: No (1)
Yes (0)	1	2
No (1)	1	0

then $2v + e < n - k + 1$ reduces to the form $d(x^n, \hat{x}^n) < n - k + 1$.

- **Question:** How many decoding attempts are needed to achieve a fixed distortion threshold?

Distortion measure

Definition

Given a *letter-by-letter* distortion measure δ, the distortion between and error pattern x^n and an erasure pattern $\hat{x}^n$ is $d(x^n, \hat{x}^n) = \sum_{i=1}^{n} \delta(x_i, \hat{x}_i)$.

Proposition

If we choose $\delta : \mathbb{Z}_2^n \times \mathbb{Z}_2^n \to \mathbb{R}_{\geq 0}$ as follows

$$\delta(0,0) = 1 \quad \delta(0,1) = 2$$
$$\delta(1,0) = 1 \quad \delta(1,1) = 0$$

Error	Erasure: Yes (0)	Erasure: No (1)
Yes (0)	1	2
No (1)	1	0

then $2v + e < n - k + 1$ reduces to the form $d(x^n, \hat{x}^n) < n - k + 1$.

- **Question:** How many decoding attempts are needed to achieve a fixed distortion threshold?

Our approach

- **Problem statement**: Build a set $\mathcal{B}$ of no more than 2^R erasure patterns $\hat{x}^n$ in order to

$$\max_{\mathcal{B}:|\mathcal{B}|\leq 2^R} \Pr\{\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) < n-k+1\}.$$

- **Asymptotic solution 1**:
  - View the error pattern x^n as a source sequence and the erasure pattern $\hat{x}^n$ as a reproduction sequence $\rightarrow$ a source coding problem.
  - R-D theory: the set $\mathcal{B}$ of 2^R reproduction sequences (erasure patterns) can be generated randomly according to $\underline{q}$ so that

$$\lim_{n\rightarrow\infty} E\left[\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n)\right] \leq D$$

  - Thus, for large enough n, with high probability we have

$$\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) \leq D$$

Our approach

- **Problem statement**: Build a set $\mathcal{B}$ of no more than 2^R erasure patterns $\hat{x}^n$ in order to

$$\max_{\mathcal{B}:|\mathcal{B}|\leq 2^R} \Pr\{\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) < n-k+1\}.$$

- **Asymptotic solution 1**:
  - View the error pattern x^n as a source sequence and the erasure pattern $\hat{x}^n$ as a reproduction sequence → a source coding problem.
  - R-D theory: the set $\mathcal{B}$ of 2^R reproduction sequences (erasure patterns) can be generated randomly according to $\underline{q}$ so that

$$\lim_{n\to\infty} E\left[\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n)\right] \leq D$$

  - Thus, for large enough n, with high probability we have

$$\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) \leq D$$

Our approach

- **Problem statement**: Build a set $\mathcal{B}$ of no more than 2^R erasure patterns $\hat{x}^n$ in order to

$$\max_{\mathcal{B}:|\mathcal{B}|\leq 2^R} \Pr\{\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) < n-k+1\}.$$

- **Asymptotic solution 1**:
  - View the error pattern x^n as a source sequence and the erasure pattern $\hat{x}^n$ as a reproduction sequence → a source coding problem.
  - R-D theory: the set $\mathcal{B}$ of 2^R reproduction sequences (erasure patterns) can be generated randomly according to $\underline{q}$ so that

$$\lim_{n\to\infty} E\left[\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n)\right] \leq D$$

  - Thus, for large enough n, with high probability we have

$$\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) \leq D$$

Our approach

- **Problem statement**: Build a set $\mathcal{B}$ of no more than 2^R erasure patterns $\hat{x}^n$ in order to

$$\max_{\mathcal{B}:|\mathcal{B}|\leq 2^R} \Pr\{\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) < n-k+1\}.$$

- **Asymptotic solution 1**:
  - View the error pattern x^n as a source sequence and the erasure pattern $\hat{x}^n$ as a reproduction sequence $\rightarrow$ a source coding problem.
  - R-D theory: the set $\mathcal{B}$ of 2^R reproduction sequences (erasure patterns) can be generated randomly according to $\underline{q}$ so that

$$\lim_{n\to\infty} E\left[\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n)\right] \leq D$$

  - Thus, for large enough n, with high probability we have

$$\min_{\hat{x}^n\in\mathcal{B}} d(x^n,\hat{x}^n) \leq D$$

Proposed general multiple-decoding algorithm

- **Phase I**: Compute rate-distortion function (run one time)
  - Step 1: Empirically compute the reliability matrix $[\mathbf{P}_1^{(t)}]_{j,i}$ during time t for $t = 1, \ldots \tau$.
  - Step 2: Sort the probabilities in increasing order reliability order of codeword positions and get an average matrix $\bar{\mathbf{P}}$ over all time t.
  - Step 3: Compute the R-D function using probability matrix $\bar{\mathbf{P}}$. Determine the point on the R-D curve that corresponds to a designated rate R along with the test-channel input-distribution $\underline{q}$ that achieves that point.

Proposed general multiple-decoding algorithm

- **Phase I**: Compute rate-distortion function (run one time)
  - Step 1: Empirically compute the reliability matrix $[\mathbf{P}_1^{(t)}]_{j,i}$ during time t for $t = 1, \ldots \tau$.
  - Step 2: Sort the probabilities in increasing order reliability order of codeword positions and get an average matrix $\bar{\mathbf{P}}$ over all time t.
  - Step 3: Compute the R-D function using probability matrix $\bar{\mathbf{P}}$. Determine the point on the R-D curve that corresponds to a designated rate R along with the test-channel input-distribution $\underline{q}$ that achieves that point.

Proposed general multiple-decoding algorithm

- **Phase I**: Compute rate-distortion function (run one time)
  - Step 1: Empirically compute the reliability matrix $[\mathbf{P}_1^{(t)}]_{j,i}$ during time t for $t = 1, \ldots \tau$.
  - Step 2: Sort the probabilities in increasing order reliability order of codeword positions and get an average matrix $\bar{\mathbf{P}}$ over all time t.
  - Step 3: Compute the R-D function using probability matrix $\bar{\mathbf{P}}$. Determine the point on the R-D curve that corresponds to a designated rate R along with the test-channel input-distribution $\underline{q}$ that achieves that point.

Proposed general multiple-decoding algorithm (cont.)

- **Phase II**: Run actual decoder.
  - Step 4: Based on the actual received signal sequence, determine the permutation σ that gives the reliability order of codeword positions.
  - Step 5: Randomly generate a set of 2^R erasure patterns using the test-channel input-probability distribution vector $\underline{q}$ and permute the indices of each erasure pattern by the permutation σ^{-1}.
  - Step 6: Run multiple attempts of the corresponding decoding scheme (e.g. error-and erasure decoding) using the set of erasure patterns in Step 5 to produce a list of candidate codewords.
  - Step 7: Use Maximum-Likelihood (ML) decoding to pick the best codeword on the list.

Proposed general multiple-decoding algorithm (cont.)

- **Phase II**: Run actual decoder.
  - Step 4: Based on the actual received signal sequence, determine the permutation σ that gives the reliability order of codeword positions.
  - Step 5: Randomly generate a set of 2^R erasure patterns using the test-channel input-probability distribution vector $\underline{q}$ and permute the indices of each erasure pattern by the permutation σ^{-1}.
  - Step 6: Run multiple attempts of the corresponding decoding scheme (e.g. error-and erasure decoding) using the set of erasure patterns in Step 5 to produce a list of candidate codewords.
  - Step 7: Use Maximum-Likelihood (ML) decoding to pick the best codeword on the list.

Proposed general multiple-decoding algorithm (cont.)

- **Phase II**: Run actual decoder.
    - Step 4: Based on the actual received signal sequence, determine the permutation σ that gives the reliability order of codeword positions.
    - Step 5: Randomly generate a set of 2^R erasure patterns using the test-channel input-probability distribution vector $\underline{q}$ and permute the indices of each erasure pattern by the permutation σ^{-1}.
    - Step 6: Run multiple attempts of the corresponding decoding scheme (e.g. error-and erasure decoding) using the set of erasure patterns in Step 5 to produce a list of candidate codewords.
    - Step 7: Use Maximum-Likelihood (ML) decoding to pick the best codeword on the list.

Proposed general multiple-decoding algorithm (cont.)

- **Phase II**: Run actual decoder.
  - Step 4: Based on the actual received signal sequence, determine the permutation σ that gives the reliability order of codeword positions.
  - Step 5: Randomly generate a set of 2^R erasure patterns using the test-channel input-probability distribution vector $\underline{q}$ and permute the indices of each erasure pattern by the permutation σ^{-1}.
  - Step 6: Run multiple attempts of the corresponding decoding scheme (e.g. error-and erasure decoding) using the set of erasure patterns in Step 5 to produce a list of candidate codewords.
  - Step 7: Use Maximum-Likelihood (ML) decoding to pick the best codeword on the list.

Generalizations and extentions

- This general R-D framework can be extended in two ways:
  - Generalize error and erasure patterns in multiple error-and-erasure decoding to make better use of the soft information.
  - Analyze multiple ASD decoding.
- The trick is to find an appropriate distortion measure δ to convert the decoding threshold to the form $d(x^n,\hat{x}^n) < D$.
- Can also combine covering codes with random codes to work with finite n.

Generalizations and extentions

- This general R-D framework can be extended in two ways:
  - Generalize error and erasure patterns in multiple error-and-erasure decoding to make better use of the soft information.
  - Analyze multiple ASD decoding.
- The trick is to find an appropriate distortion measure δ to convert the decoding threshold to the form $d(x^n, \hat{x}^n) < D$.
- Can also combine covering codes with random codes to work with finite n.

Generalizations and extentions

- This general R-D framework can be extended in two ways:
  - Generalize error and erasure patterns in multiple error-and-erasure decoding to make better use of the soft information.
  - Analyze multiple ASD decoding.
- The trick is to find an appropriate distortion measure δ to convert the decoding threshold to the form $d(x^n, \hat{x}^n) < D$.
- Can also combine covering codes with random codes to work with finite n.

Generalizations and extentions

- This general R-D framework can be extended in two ways:
  - Generalize error and erasure patterns in multiple error-and-erasure decoding to make better use of the soft information.
  - Analyze multiple ASD decoding.
- The trick is to find an appropriate distortion measure δ to convert the decoding threshold to the form $d(x^n, \hat{x}^n) < D$.
- Can also combine covering codes with random codes to work with finite n.

Generalizations and extentions

- This general R-D framework can be extended in two ways:
  - Generalize error and erasure patterns in multiple error-and-erasure decoding to make better use of the soft information.
  - Analyze multiple ASD decoding.
- The trick is to find an appropriate distortion measure δ to convert the decoding threshold to the form $d(x^n, \hat{x}^n) < D$.
- Can also combine covering codes with random codes to work with finite n.

Generalized multiple error-and-erasure decoding

Definition

Consider a positive integer l.

- $x^n \in \mathbb{Z}_{l+1}^n$ as an generalized error pattern: at index i,

$$x_i = \begin{cases} 0, & \text{none of the first } l \text{ most likely symbols is correct} \\ j, & \text{if the } j\text{-th most likely symbol is correct } (j = 1, 2, \ldots, l) \end{cases}$$

- $\hat{x}^n \in \mathbb{Z}_{l+1}^n$ as an generalized erasure pattern: at index i,

$$\hat{x}_i = \begin{cases} 0, & \text{if an easure is used} \\ j, & \text{if the } j\text{-th most likely symbol is used as HD } (j = 1, 2, \ldots, l) \end{cases}$$

Theorem

We choose $\delta : \mathbb{Z}_{l+1} \times \mathbb{Z}_{l+1} \to \mathbb{R}_{\geq 0}$ defined by $\delta(x, \hat{x}) = [\Delta]_{x,\hat{x}}$ in terms of

$$\Delta = \begin{pmatrix} 1 & 2 & \ldots & 2 & 2 \\ 1 & 0 & \ldots & 2 & 2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 2 & \ldots & 0 & 2 \\ 1 & 2 & \ldots & 2 & 0 \end{pmatrix} \Longrightarrow$$

decoding threshold becomes $d(x^n, \hat{x}^n) < n - k + 1$.

Generalized multiple error-and-erasure decoding

Definition

Consider a positive integer l.

- $x^n \in \mathbb{Z}_{l+1}^n$ as an generalized error pattern: at index i,

$$x_i = \begin{cases} 0, & \text{none of the first } l \text{ most likely symbols is correct} \\ j, & \text{if the } j\text{-th most likely symbol is correct } (j = 1, 2, \ldots, l) \end{cases}$$

- $\hat{x}^n \in \mathbb{Z}_{l+1}^n$ as an generalized erasure pattern: at index i,

$$\hat{x}_i = \begin{cases} 0, & \text{if an easure is used} \\ j, & \text{if the } j\text{-th most likely symbol is used as HD } (j = 1, 2, \ldots, l) \end{cases}$$

Theorem

We choose $\delta : \mathbb{Z}_{l+1} \times \mathbb{Z}_{l+1} \to \mathbb{R}_{\geq 0}$ defined by $\delta(x, \hat{x}) = [\Delta]_{x,\hat{x}}$ in terms of

$$\Delta = \begin{pmatrix} 1 & 2 & \ldots & 2 & 2 \\ 1 & 0 & \ldots & 2 & 2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 2 & \ldots & 0 & 2 \\ 1 & 2 & \ldots & 2 & 0 \end{pmatrix} \Longrightarrow$$

decoding threshold becomes $d(x^n, \hat{x}^n) < n - k + 1.$

Split-covering approach

- For a finite n, the random coding approach may have problems with only a few LRPs.
- Can instead use good covering codes to handle these LRPs.
- **Split-covering approach**:
  Sort the RS c.w. positions in increasing reliability order.

$$x^n = \left[\underbrace{x_1 x_2 \ldots x_{n_c}}_{\text{use covering codes}} \;\middle|\; \underbrace{x_{n_c+1} \ldots x_n}_{\text{use random codes}} \right]$$

Split-covering approach

- For a finite n, the random coding approach may have problems with only a few LRPs.
- Can instead use good covering codes to handle these LRPs.
- **Split-covering approach**:
  Sort the RS c.w. positions in increasing reliability order.

$$x^n = \left[\underbrace{x_1 x_2 \ldots x_{n_c}}_{\text{use covering codes}} \;\middle|\; \underbrace{x_{n_c+1} \ldots x_n}_{\text{use random codes}} \right]$$

Split-covering approach

- For a finite n, the random coding approach may have problems with only a few LRPs.
- Can instead use good covering codes to handle these LRPs.
- **Split-covering approach**:
  Sort the RS c.w. positions in increasing reliability order.

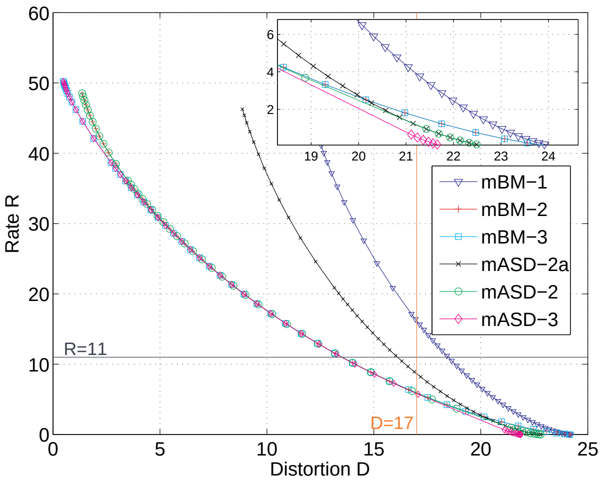
$$x^n = \left[\underbrace{x_1 x_2 \ldots x_{n_c}}_{\substack{\text{use covering} \\ \text{codes}}} \;\middle|\; \underbrace{x_{n_c+1} \ldots x_n}_{\substack{\text{use random} \\ \text{codes}}} \right]$$

Split-covering approach

- For a finite n, the random coding approach may have problems with only a few LRPs.
- Can instead use good covering codes to handle these LRPs.
- **Split-covering approach**:
  Sort the RS c.w. positions in increasing reliability order.

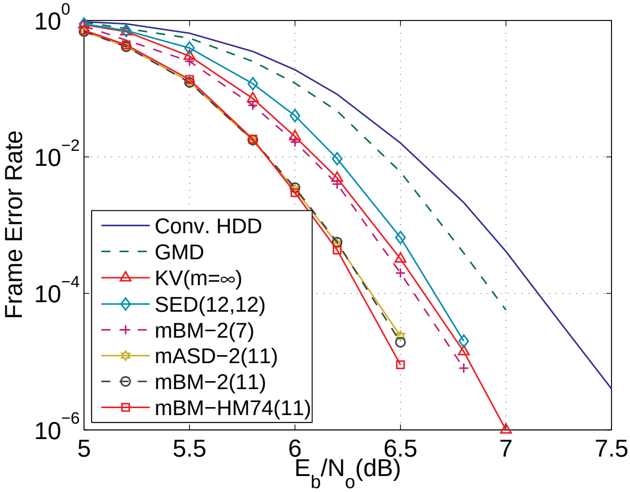
$$x^n = [\ \underbrace{x_1 x_2 \ldots x_{n_c}}_{\text{use covering codes}} \ |\ \underbrace{x_{n_c+1} \ldots x_n}_{\text{use random codes}}\ ]$$

Simulation results



A realization of R-D curves at $E_b/N_0 = 5.2$dB for various decoding algorithms for the (255,239) RS code over an AWGN channel.

Simulation results



Performance of various decoding algorithms for the (255,239) RS code over an AWGN channel.

Summary and open problems

- A unified R-D framework to analyze multiple decoding trials, with various algorithms, of RS codes in terms of performance vs complexity.
- Connect complexity-vs-performance to rate-vs-distortion relationship of an associated R-D problem.
- Covering codes are also combined to mitigate the suboptimality of random codes when the n is not large.
- Open problems:
  - Can further improve by focusing on the R-D error-exponent.
  - Can even decrease the overall complexity by using clever techniques to lower the complexity per decoding trial.

Summary and open problems

- A unified R-D framework to analyze multiple decoding trials, with various algorithms, of RS codes in terms of performance vs complexity.
- Connect complexity-vs-performance to rate-vs-distortion relationship of an associated R-D problem.
- Covering codes are also combined to mitigate the suboptimality of random codes when the n is not large.
- Open problems:
  - Can further improve by focusing on the R-D error-exponent.
  - Can even decrease the overall complexity by using clever techniques to lower the complexity per decoding trial.

Summary and open problems

- A unified R-D framework to analyze multiple decoding trials, with various algorithms, of RS codes in terms of performance vs complexity.
- Connect complexity-vs-performance to rate-vs-distortion relationship of an associated R-D problem.
- Covering codes are also combined to mitigate the suboptimality of random codes when the n is not large.
- Open problems:
  - Can further improve by focusing on the R-D error-exponent.
  - Can even decrease the overall complexity by using clever techniques to lower the complexity per decoding trial.

Summary and open problems

- A unified R-D framework to analyze multiple decoding trials, with various algorithms, of RS codes in terms of performance vs complexity.
- Connect complexity-vs-performance to rate-vs-distortion relationship of an associated R-D problem.
- Covering codes are also combined to mitigate the suboptimality of random codes when the n is not large.
- Open problems:
  - Can further improve by focusing on the R-D error-exponent.
  - Can even decrease the overall complexity by using clever techniques to lower the complexity per decoding trial.

Summary and open problems

- A unified R-D framework to analyze multiple decoding trials, with various algorithms, of RS codes in terms of performance vs complexity.
- Connect complexity-vs-performance to rate-vs-distortion relationship of an associated R-D problem.
- Covering codes are also combined to mitigate the suboptimality of random codes when the n is not large.
- Open problems:
  - Can further improve by focusing on the R-D error-exponent.
  - Can even decrease the overall complexity by using clever techniques to lower the complexity per decoding trial.

THANK YOU!

THANK YOU!