Computer-Aided Investigation of Information-Theoretic Limits: An Overview

Chao Tian

Texas A&M University

June 2024

Based on joint work with Jim Plank, Tie Liu, Jun Chen, Hua Sun, Tao Guo, Ruida Zhou, Wenjing Chen, Brent Hurst,

and 

Outline

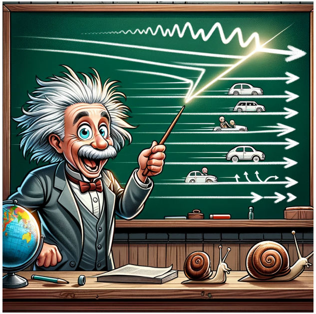
1. Fundamental Limits of Information Systems
2. Symmetry-Reduced Entropy LP
3. Beyond Bounds and Proofs
   - Reverse engineering optimal codes
   - Data-driven outer bound hypotheses
   - Computer-aided exploration
4. A New Software Toolbox (CAI)
5. Two New Directions
   - Utilizing non-Shannon-type inequalities
   - A new decomposition approach
6. Summary

Outline

Information Systems

- Information processing for certain purpose;
- Mostly noiseless (wireline) & contents are independent (files or bits);
- Not including noisy channels (e.g., Li TIT-23).

Fundamental Limits of Information Systems



Fundamental limits: hard limit, regardless of the engineering design

- Usually obtained through some counting arguments: in information theory, we use *entropy* to count.

Information Theoretic Limits: Conventional Approach

An art more than a science:

1. Develop a good understanding of the engineering problem;
2. Chain of inequalities: translate the understanding + trial-and-error.

$\Downarrow$

Heavy reliance on humans: human ingenuity and diligence

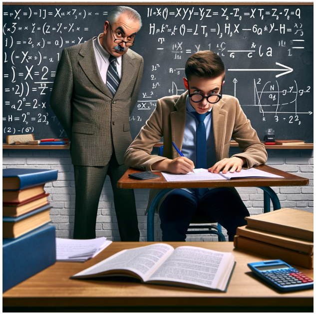
Information Theoretic Limits: Conventional Approach

An art more than a science:

1. Develop a good understanding of the engineering problem;
2. Chain of inequalities: translate the understanding + trial-and-error.

$$\Downarrow$$

Heavy reliance on humans: human ingenuity and diligence



Information Theoretic Limits: New Approaches?

Question: how can we reduce the human factors?

An optimization view: find the "best" combination of information inequalities

$\Downarrow$

Idea: computers to do some or all the work?

$\Uparrow$

A key driver: development in optimization software and computer hardware.

Information Theoretic Limits: New Approaches?

Question: how can we reduce the human factors?

An optimization view: find the "best" combination of information inequalities

$\Downarrow$

Idea: computers to do some or all the work?

$\Uparrow$

A key driver: development in optimization software and computer hardware.

Information Theoretic Limits: New Approaches?

Question: how can we reduce the human factors?

An optimization view: find the "best" combination of information inequalities

$\Downarrow$

Idea: computers to do some or all the work?

$\Uparrow$

A key driver: development in optimization software and computer hardware.

Information Theoretic Limits: New Approaches?

Question: how can we reduce the human factors?

An optimization view: find the "best" combination of information inequalities

$\Downarrow$

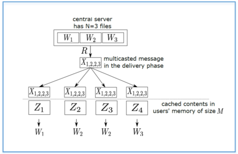
Idea: computers to do some or all the work?

$\Uparrow$

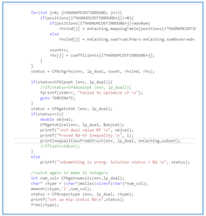
A key driver: development in optimization software and computer hardware.

A Computational Approach for the Fundamental Limits

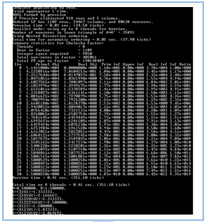
System model

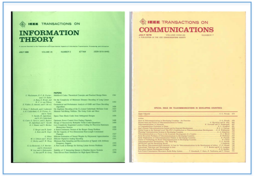


Computer code

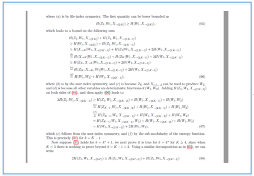


LP/dual-LP optimization

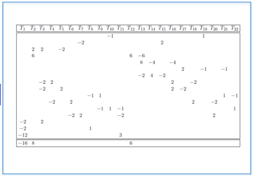




Publication



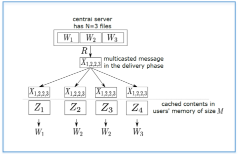
Research paper



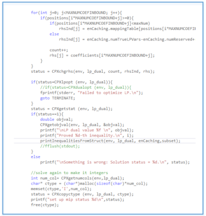
Bounds & proof tables

Goal: To solve real difficult research problems and obtain new engineering ideas.

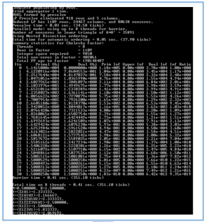
System model

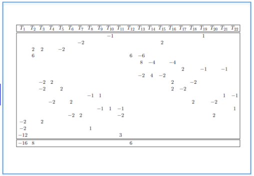


Computer code

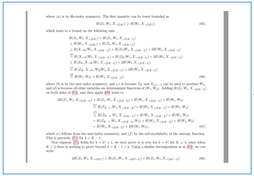


LP/dual-LP optimization

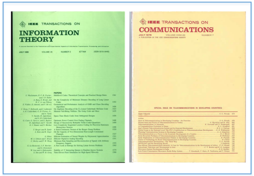




Bounds & proof tables



Research paper



Publication

Goal: To solve real difficult research problems and obtain new engineering ideas.

The Mathematical Foundation: Yeung's Entropy Linear Program

Is a certain information inequality true? "Yes or can't-determine"

Example

With three random variables Y_1, Y_2, Y_3, does the inequality $H(Y_1) \geq H(Y_2)$ hold?

$$x_{001} \triangleq H(Y_1),\ x_{010} \triangleq H(Y_2),\ x_{100} \triangleq H(Y_2),\ x_{011} \triangleq H(Y_1, Y_2),$$
$$x_{110} \triangleq H(Y_2, Y_3),\ x_{101} \triangleq H(Y_1, Y_3),\ x_{111} \triangleq H(Y_1, Y_2, Y_3).$$

We can consider the optimization problem:

$$\begin{aligned} \text{minimize:}\ & x_{001} - x_{010} \\ \text{subject to:}\ & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0 \\ & x_{001} + x_{010} - x_{011} \geq 0,\ \dots\dots \\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0. \end{aligned}$$

This looks weird, but let's translate: $x_{001} + x_{010} - x_{011} \Leftrightarrow H(Y_1) + H(Y_2) - H(Y_1, Y_2) = I(Y_1; Y_2) \geq 0.$

The Mathematical Foundation: Yeung's Entropy Linear Program

Is a certain information inequality true? "Yes or can't-determine"

Example

With three random variables Y_1, Y_2, Y_3, does the inequality $H(Y_1) \geq H(Y_2)$ hold?

$$x_{001} \triangleq H(Y_1),\ x_{010} \triangleq H(Y_2),\ x_{100} \triangleq H(Y_2),\ x_{011} \triangleq H(Y_1, Y_2),$$
$$x_{110} \triangleq H(Y_2, Y_3),\ x_{101} \triangleq H(Y_1, Y_3),\ x_{111} \triangleq H(Y_1, Y_2, Y_3).$$

We can consider the optimization problem:

$$\begin{aligned} \text{minimize: } & x_{001} - x_{010} \\ \text{subject to: } & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0 \\ & x_{001} + x_{010} - x_{011} \geq 0,\ \dots\dots \\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0. \end{aligned}$$

This looks weird, but let's translate: $x_{001} + x_{010} - x_{011} \Leftrightarrow H(Y_1) + H(Y_2) - H(Y_1, Y_2) = I(Y_1; Y_2) \geq 0.$

The Mathematical Foundation: Yeung's Entropy Linear Program

Is a certain information inequality true? "Yes or can't-determine"

Example

With three random variables Y_1, Y_2, Y_3, does the inequality $H(Y_1) \geq H(Y_2)$ hold?

$$x_{001} \triangleq H(Y_1),\ x_{010} \triangleq H(Y_2),\ x_{100} \triangleq H(Y_2),\ x_{011} \triangleq H(Y_1, Y_2),$$
$$x_{110} \triangleq H(Y_2, Y_3),\ x_{101} \triangleq H(Y_1, Y_3),\ x_{111} \triangleq H(Y_1, Y_2, Y_3).$$

We can consider the optimization problem:

$$\begin{aligned}\text{minimize:}\ & x_{001} - x_{010}\\ \text{subject to:}\ & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0\\ & x_{001} + x_{010} - x_{011} \geq 0,\ \ldots\ldots\\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0.\end{aligned}$$

This looks weird, but let's translate: $x_{001} + x_{010} - x_{011} \Leftrightarrow H(Y_1) + H(Y_2) - H(Y_1, Y_2) = I(Y_1; Y_2) \geq 0$.

The Mathematical Foundation: Yeung's Entropy Linear Program

Is a certain information inequality true? "Yes or can't-determine"

Example

With three random variables Y_1, Y_2, Y_3, does the inequality $H(Y_1) \geq H(Y_2)$ hold?

$$x_{001} \triangleq H(Y_1),\ x_{010} \triangleq H(Y_2),\ x_{100} \triangleq H(Y_2),\ x_{011} \triangleq H(Y_1, Y_2),$$
$$x_{110} \triangleq H(Y_2, Y_3),\ x_{101} \triangleq H(Y_1, Y_3),\ x_{111} \triangleq H(Y_1, Y_2, Y_3).$$

We can consider the optimization problem:

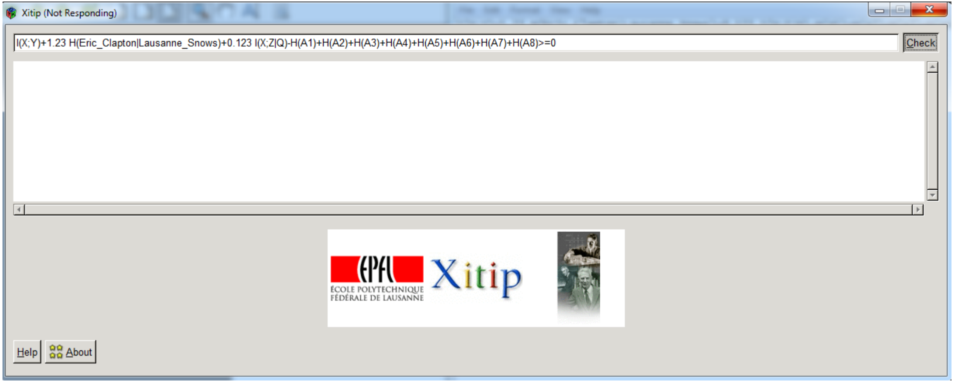
$$\begin{aligned} \text{minimize: } & x_{001} - x_{010} \\ \text{subject to: } & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0 \\ & x_{001} + x_{010} - x_{011} \geq 0,\ \dots\dots \\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0. \end{aligned}$$

This looks weird, but let's translate: $x_{001} + x_{010} - x_{011} \Leftrightarrow H(Y_1) + H(Y_2) - H(Y_1, Y_2) = I(Y_1; Y_2) \geq 0$.

Software and Libraries

ITIP, Xitip, and Citip libraries (1997, 2007, 2020), which were used to

- Study the entropic regions;
- Verify simple conjectured inequalities.



Proving Inequalities → Converse for Coding Problems

Example

A source Y_1 of unit rate, is encoded into Y_2 and Y_3 (maybe with additional randomness) of equal rates, that can be used to jointly recover Y_1. What is the minimum coding rate of Y_2?

Translation: $H(Y_1) = 1$, $H(Y_2) = H(Y_3)$, $H(Y_1|Y_2, Y_3) = 0$, lower bound on $H(Y_2)$?

$$\text{minimize: } x_{010}$$
$$\text{subject to: } x_{001} = 1,\ x_{010} = x_{100},\ x_{111} - x_{110} = 0$$

Are these all the constraints? Should also include the elemental inequalities:

$$\begin{aligned}\text{subject also to: } & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0\\ & x_{001} + x_{010} - x_{011} \geq 0,\ \ldots\\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0.\end{aligned}$$

Proving Inequalities $\to$ Converse for Coding Problems

Example

A source Y_1 of unit rate, is encoded into Y_2 and Y_3 (maybe with additional randomness) of equal rates, that can be used to jointly recover Y_1. What is the minimum coding rate of Y_2?

Translation: $H(Y_1) = 1$, $H(Y_2) = H(Y_3)$, $H(Y_1|Y_2, Y_3) = 0$, lower bound on $H(Y_2)$?

$$\text{minimize: } x_{010}$$

$$\text{subject to: } x_{001} = 1,\ x_{010} = x_{100},\ x_{111} - x_{110} = 0$$

Are these all the constraints? Should also include the elemental inequalities:

$$\begin{aligned} \text{subject also to: } & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0 \\ & x_{001} + x_{010} - x_{011} \geq 0,\ \ldots \\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0. \end{aligned}$$

Example

A source Y_1 of unit rate, is encoded into Y_2 and Y_3 (maybe with additional randomness) of equal rates, that can be used to jointly recover Y_1. What is the minimum coding rate of Y_2?

Translation: $H(Y_1) = 1$, $H(Y_2) = H(Y_3)$, $H(Y_1|Y_2, Y_3) = 0$, lower bound on $H(Y_2)$?

$$\begin{aligned} \text{minimize: } & x_{010} \\ \text{subject to: } & x_{001} = 1,\ x_{010} = x_{100},\ x_{111} - x_{110} = 0 \end{aligned}$$

Are these all the constraints? Should also include the elemental inequalities:

$$\begin{aligned} \text{subject also to: } & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0 \\ & x_{001} + x_{010} - x_{011} \geq 0,\ \ldots \\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0. \end{aligned}$$

Example

A source Y_1 of unit rate, is encoded into Y_2 and Y_3 (maybe with additional randomness) of equal rates, that can be used to jointly recover Y_1. What is the minimum coding rate of Y_2?

Translation: $H(Y_1) = 1$, $H(Y_2) = H(Y_3)$, $H(Y_1|Y_2, Y_3) = 0$, lower bound on $H(Y_2)$?

$$\begin{aligned} \text{minimize: } & x_{010} \\ \text{subject to: } & x_{001} = 1,\ x_{010} = x_{100},\ x_{111} - x_{110} = 0 \end{aligned}$$

Are these all the constraints? Should also include the elemental inequalities:

$$\begin{aligned} \text{subject also to: } & x_{111} - x_{001} \geq 0,\ x_{111} - x_{010} \geq 0,\ x_{111} - x_{100} \geq 0 \\ & x_{001} + x_{010} - x_{011} \geq 0,\ \ldots \\ & x_{011} + x_{110} - x_{111} - x_{010} \geq 0. \end{aligned}$$

Why Are We Still Here?

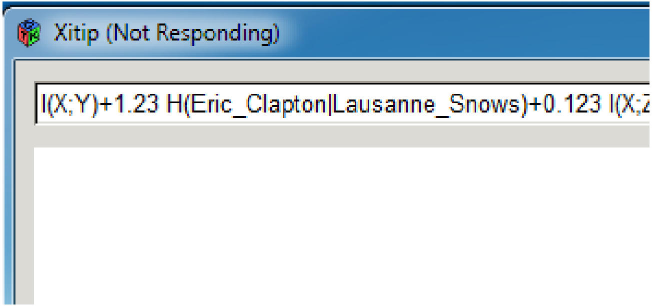
Exponential in the number of random variables: storage and computation constrained

- n random variables: $2^n - 1$ LP variables and $n + \binom{n}{2} 2^{n-2}$ LP constraints.

Why Are We Still Here?

Exponential in the number of random variables: storage and computation constrained

- n random variables: $2^n - 1$ LP variables and $n + \binom{n}{2} 2^{n-2}$ LP constraints.

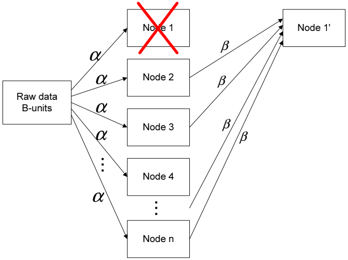


Outline

Symmetry-Reduced Entropy LP

Motivation:

- In regenerating code, the simplest non-trivial case had at least 16 random variables;
- Translate to roughly 2 million inequality constraints! Too complex ☹
- However, the problem is highly symmetric.



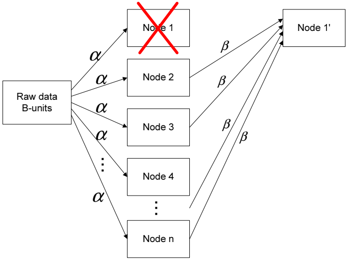
Therefore, we built a customized approach (T. JSAC-14):

1. Symmetry and other factors to reduce LP;
2. Compute the outer bounds;
3. LP dual to generate human-readable proofs.

Symmetry-Reduced Entropy LP

Motivation:

- In regenerating code, the simplest non-trivial case had at least 16 random variables;
- Translate to roughly 2 million inequality constraints! Too complex ☹
- However, the problem is highly symmetric.



Therefore, we built a customized approach (T. JSAC-14):

1. Symmetry and other factors to reduce LP;
2. Compute the outer bounds;
3. LP dual to generate human-readable proofs.

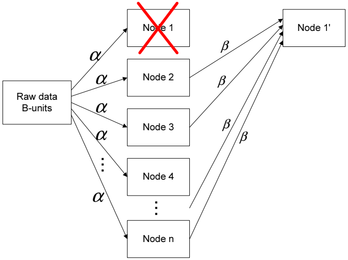
Symmetry-Reduced Entropy LP

Motivation:

- In regenerating code, the simplest non-trivial case had at least 16 random variables;
- Translate to roughly 2 million inequality constraints! Too complex ☹
- However, the problem is highly symmetric.

Therefore, we built a customized approach (T. JSAC-14):

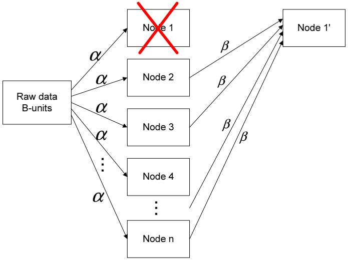
1. Symmetry and other factors to reduce LP;
2. Compute the outer bounds;
3. LP dual to generate human-readable proofs.



Symmetry-Reduced Entropy LP

Motivation:

- In regenerating code, the simplest non-trivial case had at least 16 random variables;
- Translate to roughly 2 million inequality constraints! Too complex ☹
- However, the problem is highly symmetric.



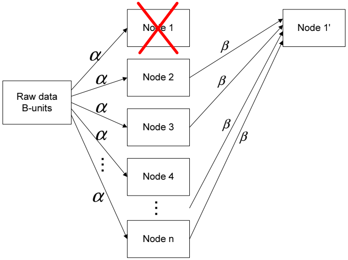
Therefore, we built a customized approach (T. JSAC-14):

1. Symmetry and other factors to reduce LP;
2. Compute the outer bounds;
3. LP dual to generate human-readable proofs.

Symmetry-Reduced Entropy LP

Motivation:

- In regenerating code, the simplest non-trivial case had at least 16 random variables;
- Translate to roughly 2 million inequality constraints! Too complex ☹
- However, the problem is highly symmetric.



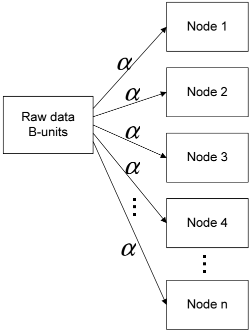
Therefore, we built a customized approach (T. JSAC-14):

1. Symmetry and other factors to reduce LP;
2. Compute the outer bounds;
3. LP dual to generate human-readable proofs.

First Setting: The Regenerating Code Problem

Dimakis et al. Infocom-07

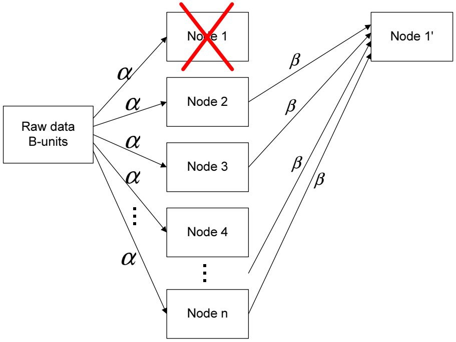
- (n, k) property: any k in n nodes can recover the B-units of total data;
- Node of size α;
- Repair to access any d remaining nodes for β each.



First Setting: The Regenerating Code Problem

Dimakis et al. Infocom-07

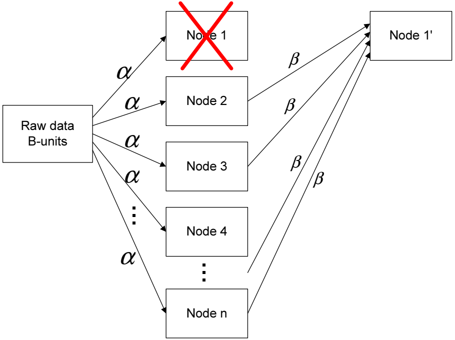
- (n, k) property: any k in n nodes can recover the B-units of total data;
- Node of size α;
- Repair to access any d remaining nodes for β each.



First Setting: The Regenerating Code Problem

Dimakis et al. Infocom-07

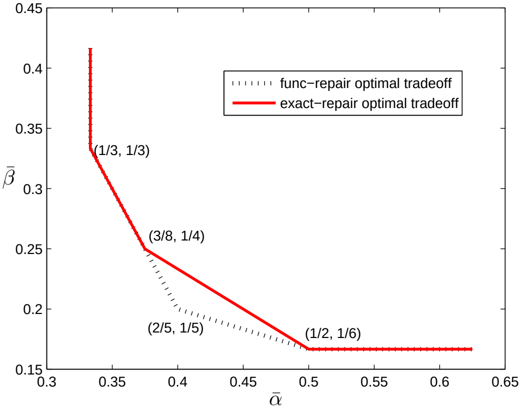
- (n, k) property: any k in n nodes can recover the B-units of total data;
- Node of size α;
- Repair to access any d remaining nodes for β each.



Optimal tradeoff of $\{(\bar{\alpha}, \bar{\beta})\}$ for fixed (n, k, d)?

Bounds and Implication

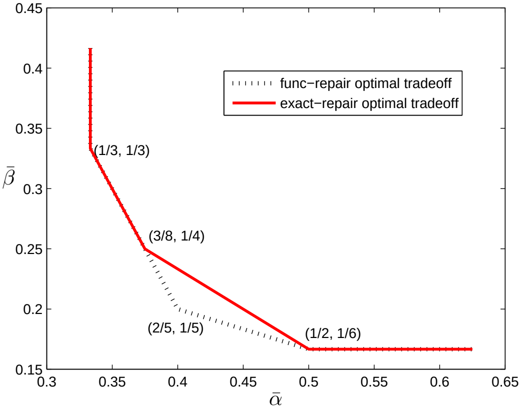
Exact-repair optimal tradeoff $\neq$ Functional-repair optimal tradeoff?



Key: establish an outer bound for exact-repair codes.

Bounds and Implication

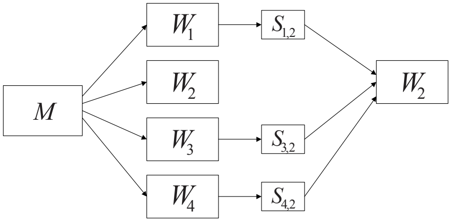
Exact-repair optimal tradeoff $\neq$ Functional-repair optimal tradeoff?



Key: establish an outer bound for exact-repair codes.

Translation: Regenerating Codes $(n, k, d) = (4, 3, 3)$

Define random variables and write the conditions



$$M, W_1, W_2, W_3, W_4$$
$$S_{1,2}, S_{1,3}, S_{1,4}$$
$$S_{2,1}, S_{2,3}, S_{2,4}$$
$$S_{3,1}, S_{3,2}, S_{3,4}$$
$$S_{4,1}, S_{4,2}, S_{4,3}$$

$$H(M) = B, \quad H(W_i) \leq \alpha, \quad H(S_{i,j}) \leq \beta,$$
$$H(S_{i,j}|W_i) = 0, \quad H(W_i|\{S_{j,i}, j \neq i\}) = 0, \quad H(W_i, W_j, W_k) = B$$

The Main Idea: Symmetry Reduction

Proposition (Informal)

There is no loss in using (considering) only symmetric codes.

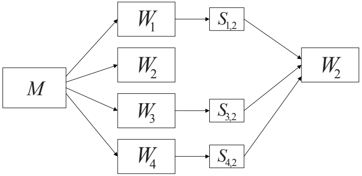
Intuition: storage nodes have the same role, so permutation does not jeopardize performance.

- Symmetry reduction, e.g.,

$$H(W_1, W_2, S_{1,3}, S_{2,4}) = H(W_2, W_3, S_{2,4}, S_{3,1}).$$

- Other reductions:

$$H(W_i, W_j, W_k) = \ldots = H(\{W_i\}, \{S_{i,j}\}) = B.$$



Many joint entropy terms have the same values

$\Downarrow$

No need to represent them using different variables in LP!

The Main Idea: Symmetry Reduction

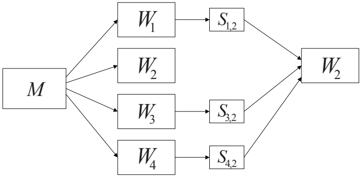
Proposition (Informal)

There is no loss in using (considering) only symmetric codes.

Intuition: storage nodes have the same role, so permutation does not jeopardize performance.

- Symmetry reduction, e.g.,

$$H(W_1, W_2, S_{1,3}, S_{2,4}) = H(W_2, W_3, S_{2,4}, S_{3,1}).$$



- Other reductions:

$$H(W_i, W_j, W_k) = \ldots = H(\{W_i\}, \{S_{i,j}\}) = B.$$

Many joint entropy terms have the same values

$\Downarrow$

No need to represent them using different variables in LP!

The Main Idea: Symmetry Reduction

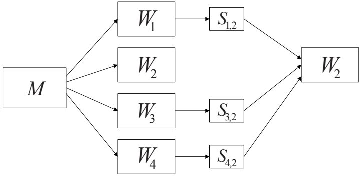
Proposition (Informal)

There is no loss in using (considering) only symmetric codes.

Intuition: storage nodes have the same role, so permutation does not jeopardize performance.

- Symmetry reduction, e.g.,

$$H(W_1, W_2, S_{1,3}, S_{2,4}) = H(W_2, W_3, S_{2,4}, S_{3,1}).$$



- Other reductions:

$$H(W_i, W_j, W_k) = \ldots = H(\{W_i\}, \{S_{i,j}\}) = B.$$

Many joint entropy terms have the same values

$\Downarrow$

No need to represent them using different variables in LP!

The Reduced LP

Use these reductions to remove redundant variables and constraints in LP

$$\boxed{\begin{array}{c}\text{LP with 65535 variables} + \text{2 million constraints} \\ \Downarrow \\ \text{LP with 176 variables} + \text{6152 constraints}\end{array}}$$

Now the LP is small

- Trace out the boundary with some discrete (α, β) pairs;
  - From this we identify $4\alpha + 6\beta \geq 3B$
- Only numerical result: not good for understanding the problem;
- Note: There are not minimal; see Guo et al. 2024.

The Reduced LP

Use these reductions to remove redundant variables and constraints in LP

$$\boxed{\begin{array}{c}\text{LP with 65535 variables} + \text{2 million constraints} \\ \Downarrow \\ \text{LP with 176 variables} + \text{6152 constraints}\end{array}}$$

Now the LP is small

- Trace out the boundary with some discrete (α, β) pairs;
  - From this we identify $4\alpha + 6\beta \geq 3B$
- Only numerical result: not good for understanding the problem;
- Note: There are not minimal; see Guo et al. 2024.

The Reduced LP

Use these reductions to remove redundant variables and constraints in LP

$$\boxed{\begin{array}{c}\text{LP with 65535 variables} + \text{2 million constraints}\\ \Downarrow \\ \text{LP with 176 variables} + \text{6152 constraints}\end{array}}$$

Now the LP is small

- Trace out the boundary with some discrete (α, β) pairs;
  - From this we identify $4\alpha + 6\beta \geq 3B$
- Only numerical result: not good for understanding the problem;
- Note: There are not minimal; see Guo et al. 2024.

Explicit Proof: LP Dual to the Rescue

What is an explicit proof?

- Conventionally and on the first sight: a (mysterious) chain of inequalities
  - Analogy: want to show $3a - d \geq 0$, but only know

$$\begin{array}{ccccc} a & +b & -c & & \geq 0 \\ & -4b & +c & +d & \geq 0 \\ & b & +2c & -2d & \geq 0 \end{array}$$

- More fundamentally: a linear combination of known inequalities;
- $4\alpha + 6\beta \geq 3B$ is such a linear combination;
- But the LP solver must have already found this combination.

Explicit Proof: LP Dual to the Rescue

What is an explicit proof?

- Conventionally and on the first sight: a (mysterious) chain of inequalities
  - Analogy: want to show $3a - d \geq 0$, but only know

$$
\begin{array}{lllll}
3a & +3b & -3c & & \geq 0 \\
 & -4b & +c & +d & \geq 0 \\
 & b & +2c & -2d & \geq 0
\end{array}
$$

- More fundamentally: a linear combination of known inequalities;
- $4\alpha + 6\beta \geq 3B$ is such a linear combination;
- But the LP solver must have already found this combination.

Explicit Proof: LP Dual to the Rescue

What is an explicit proof?

- Conventionally and on the first sight: a (mysterious) chain of inequalities
  - Analogy: want to show $3a - d \geq 0$, but only know

$$\begin{array}{ccccc} 3a & +3b & -3c & & \geq 0 \\ & -4b & +c & +d & \geq 0 \\ & b & +2c & -2d & \geq 0 \end{array}$$

- More fundamentally: a linear combination of known inequalities;
- $4\alpha + 6\beta \geq 3B$ is such a linear combination;
- But the LP solver must have already found this combination.

Explicit Proof: LP Dual to the Rescue

What is an explicit proof?

- Conventionally and on the first sight: a (mysterious) chain of inequalities
  - Analogy: want to show $3a - d \geq 0$, but only know

$$
\begin{array}{lllll}
3a & +3b & -3c & & \geq 0 \\
 & -4b & +c & +d & \geq 0 \\
 & b & +2c & -2d & \geq 0
\end{array}
$$

- More fundamentally: a linear combination of known inequalities;
- $4\alpha + 6\beta \geq 3B$ is such a linear combination;
- But the LP solver must have already found this combination.

Explicit Proof: LP Dual to the Rescue

What is an explicit proof?

- Conventionally and on the first sight: a (mysterious) chain of inequalities
  - Analogy: want to show $3a - d \geq 0$, but only know

$$
\begin{array}{lllll}
3a & +3b & -3c & & \geq 0 \\
 & -4b & +c & +d & \geq 0 \\
 & b & +2c & -2d & \geq 0
\end{array}
$$

- More fundamentally: a linear combination of known inequalities;
- $4\alpha + 6\beta \geq 3B$ is such a linear combination;
- But the LP solver must have already found this combination.

Explicit Proof: LP Dual to the Rescue

What is an explicit proof?

- Conventionally and on the first sight: a (mysterious) chain of inequalities
  - Analogy: want to show $3a - d \geq 0$, but only know

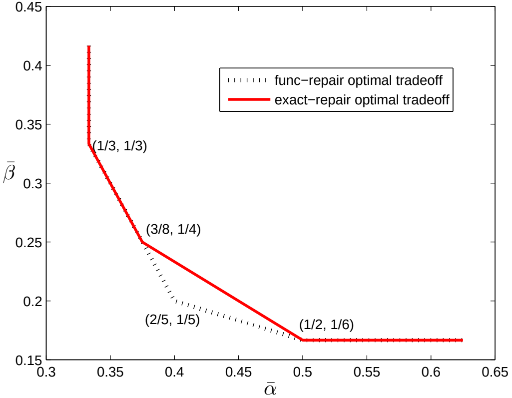
$$\begin{array}{lllll} 3a & +3b & -3c & & \geq 0 \\ & -4b & +c & +d & \geq 0 \\ & b & +2c & -2d & \geq 0 \end{array}$$

- More fundamentally: a linear combination of known inequalities;
- $4\alpha + 6\beta \geq 3B$ is such a linear combination;
- But the LP solver must have already found this combination.

Solution: solve the LP dual problem.

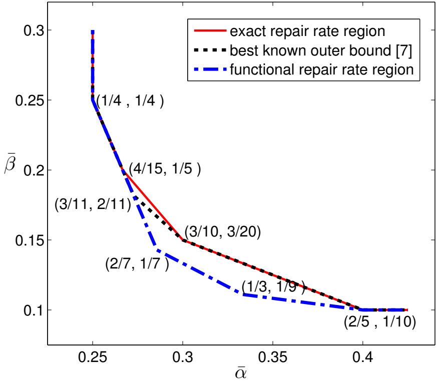
The Proof Table

Coefficients	Inequalities
7	$I(S_{i,j}; W_k) \geq 0$
3	$I(S_{k,j}; S_{t,j} \vert W_i) \geq 0$
1	$I(W_i; W_j \vert S_{i,j}) \geq 0$
1	$I(W_i; S_{t,k} \vert W_j) \geq 0$
1	$I(W_i; W_j S_{k,t} \vert S_{i,t} S_{j,i} W_t) \geq 0$
1	$I(W_i; S_{k,t} \vert S_{k,j} S_{t,j} W_j) \geq 0$
1	$I(S_{k,i}; S_{k,j} \vert S_{j,i} W_i) \geq 0$
1	$H(S_{t,i} \vert S_{k,i} W_i W_j) \geq 0$



Adding them up and canceling out terms $\Rightarrow 4\alpha + 6\beta \geq 3B$

Generalization to Larger Instances of Regenerating Codes



- Complete solution for the $(5, 4, 4)$ case;
  - 24 random variables; $\sim$1.16 billion constraints before reduction.
- Solution for the $(6, 5, 5)$ setting does not match the inner bound.

Outline

What Else?

The general computational approach: we built a hammer

1. Symmetry and symmetry-reduced entropic LP
2. Generating human readable proofs

How to better use the hammer?

3. Reverse engineering optimal codes
4. Data-driven outer bound hypotheses
5. Computer-aided exploration

We next use the coded caching system as our running example.

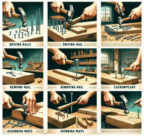
What Else?

The general computational approach: we built a hammer

1. Symmetry and symmetry-reduced entropic LP
2. Generating human readable proofs

How to better use the hammer?

3. Reverse engineering optimal codes
4. Data-driven outer bound hypotheses
5. Computer-aided exploration



We next use the coded caching system as our running example.

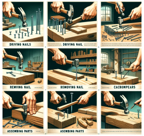
What Else?

The general computational approach: we built a hammer

1. Symmetry and symmetry-reduced entropic LP
2. Generating human readable proofs

How to better use the hammer?

3. Reverse engineering optimal codes
4. Data-driven outer bound hypotheses
5. Computer-aided exploration

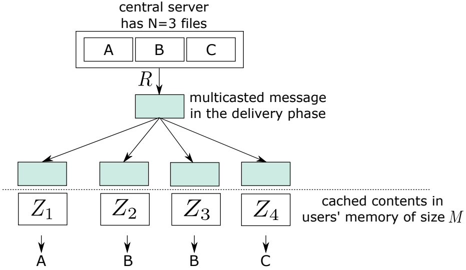


We next use the coded caching system as our running example.

Second Setting: Coded Caching

Proposed by Maddah-Ali & Niesen (IT-14)

- N files, K users, each user has a cache of size M;
- Placement phase vs. delivery phase.

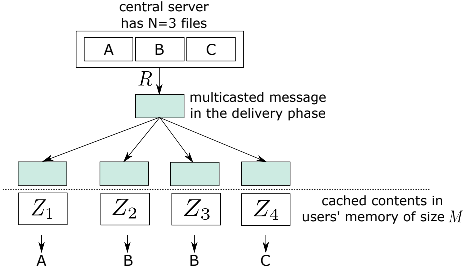


What is the optimal tradeoff between M and R?

Second Setting: Coded Caching

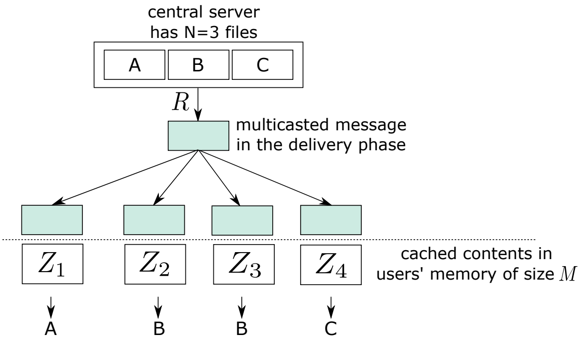
Proposed by Maddah-Ali & Niesen (IT-14)

- N files, K users, each user has a cache of size M;
- Placement phase vs. delivery phase.



What is the optimal tradeoff between M and R?

Random Variables in the Caching Problem



Random variables in the problem: $n = N + K + N^K$

- N files: $\mathcal{W} = \{W_1, W_2, ..., W_N\}$;
- Cached contents at K users: $\mathcal{Z} = \{Z_1, Z_2, ..., Z_K\}$;
- Transmission for demands $(d_1, d_2, \ldots, d_K)$: $\mathcal{X} = \{X_{d_1,d_2,...,d_K}\}$.

A Linear Program (Before Reduction)

Objective function:

$$\text{minimize: } M$$

Problem specific constraints:

$$
\begin{aligned}
H(Z_k|W_1, W_2, \dots, W_N) &= 0, \quad k = 1, 2, \dots, K; \\
H(X_{d_1,d_2,\dots,d_K}|W_1, W_2, \dots, W_N) &= 0, \quad d_k \in \{1, 2, \dots, N\}; \\
H(W_{d_k}|Z_k, X_{d_1,d_2,\dots,d_K}) &= 0, \quad d_k \in \{1, 2, \dots, N\}, \; k = 1, \dots, k; \\
H(Z_k) &\le M, \quad k = 1, 2, \dots, K; \\
H(X_{d_1,d_2,\dots,d_K}) &\le R, \quad d_k \in \{1, 2, \dots, N\}.
\end{aligned}
$$

Generic constraints: elemental entropic inequalities for a set of R.V.s Ω

$$
\begin{aligned}
H(A|\Omega \setminus \{A\}) &\ge 0, \quad A \in \Omega; \\
I(A; B|T) &\ge 0, \quad \text{where } T \subseteq \Omega \setminus \{A, B\}, \; A, B \in \Omega
\end{aligned}
$$

A Linear Program (Before Reduction)

Objective function:

$$\text{minimize: } M$$

Problem specific constraints:

$$\begin{aligned}
H(Z_k|W_1, W_2, \ldots, W_N) &= 0, \quad k = 1, 2, \ldots, K; \\
H(X_{d_1,d_2,\ldots,d_K}|W_1, W_2, \ldots, W_N) &= 0, \quad d_k \in \{1, 2, \ldots, N\}; \\
H(W_{d_k}|Z_k, X_{d_1,d_2,\ldots,d_K}) &= 0, \quad d_k \in \{1, 2, \ldots, N\},\ k = 1, \ldots, k; \\
H(Z_k) &\leq M, \quad k = 1, 2, \ldots, K; \\
H(X_{d_1,d_2,\ldots,d_K}) &\leq R, \quad d_k \in \{1, 2, \ldots, N\}.
\end{aligned}$$

Generic constraints: elemental entropic inequalities for a set of R.V.s Ω

$$\begin{aligned}
H(A|\Omega \setminus \{A\}) &\geq 0, \quad A \in \Omega; \\
I(A; B|T) &\geq 0, \quad \text{where } T \subseteq \Omega \setminus \{A, B\},\ A, B \in \Omega
\end{aligned}$$

A Linear Program (Before Reduction)

Objective function:

$$\text{minimize: } M$$

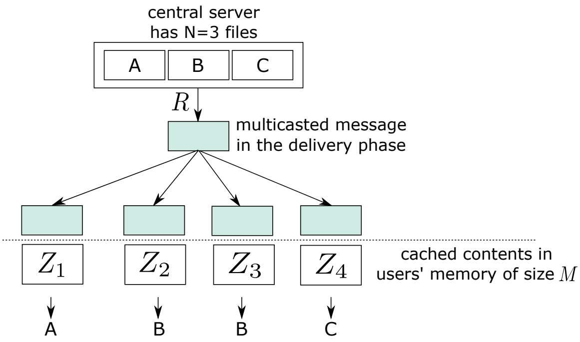
Problem specific constraints:

$$\begin{aligned}
H(Z_k|W_1, W_2, \ldots, W_N) &= 0, \quad k = 1, 2, \ldots, K;\\
H(X_{d_1,d_2,\ldots,d_K}|W_1, W_2, \ldots, W_N) &= 0, \quad d_k \in \{1, 2, \ldots, N\};\\
H(W_{d_k}|Z_k, X_{d_1,d_2,\ldots,d_K}) &= 0, \quad d_k \in \{1, 2, \ldots, N\},\ k = 1, \ldots, k;\\
H(Z_k) &\leq M, \quad k = 1, 2, \ldots, K;\\
H(X_{d_1,d_2,\ldots,d_K}) &\leq R, \quad d_k \in \{1, 2, \ldots, N\}.
\end{aligned}$$

Generic constraints: elemental entropic inequalities for a set of R.V.s Ω

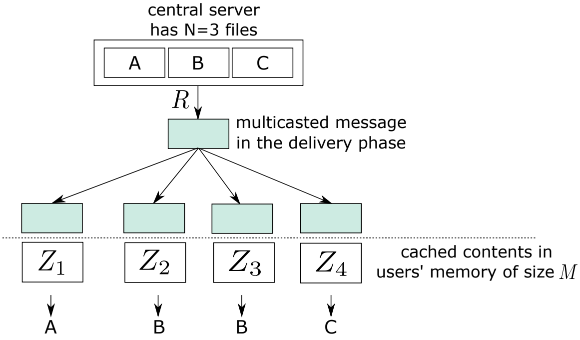
$$\begin{aligned}
H(A|\Omega \setminus \{A\}) &\geq 0, \quad A \in \Omega;\\
I(A; B|T) &\geq 0, \quad \text{where } T \subseteq \Omega \setminus \{A, B\},\ A, B \in \Omega
\end{aligned}$$

Symmetry in the Caching Problem



- User index symmetry $\bar{\pi}$: permute the cached contents Z_i at users
- File index symmetry $\hat{\pi}$: permute the files before encoding

Symmetry in the Caching Problem



- User index symmetry $\bar{\pi}$: permute the cached contents Z_i at users
- File index symmetry $\hat{\pi}$: permute the files before encoding

The Existence of Optimal Symmetric Codes

Proposition

For any caching code, there is a code with the same or smaller caching memory and transmission rate, which is both user-index-symmetric and file-index-symmetric.

$\Rightarrow$ Without loss of optimality, can consider only symmetric codes.

Example: $(N, K) = (3, 4)$

- User-index: $\bar{\pi} = \begin{pmatrix} 1234 \\ 2314 \end{pmatrix}$, $H(W_2, Z_2, X_{1,2,3,2}) = H(W_2, Z_3, X_{3,1,2,2})$
- File-index: $\hat{\pi} = \begin{pmatrix} 123 \\ 231 \end{pmatrix}$, $H(W_3, Z_3, X_{1,2,3,2}) = H(W_1, Z_3, X_{2,3,1,3})$

$\Rightarrow$ LP significantly reduced (e.g. 10^8 to 10^4).

The Existence of Optimal Symmetric Codes

Proposition

For any caching code, there is a code with the same or smaller caching memory and transmission rate, which is both user-index-symmetric and file-index-symmetric.

$\Rightarrow$ Without loss of optimality, can consider only symmetric codes.

Example: $(N, K) = (3, 4)$

- User-index: $\bar{\pi} = \begin{pmatrix} 1234 \\ 2314 \end{pmatrix}$, $H(W_2, Z_2, X_{1,2,3,2}) = H(W_2, Z_3, X_{3,1,2,2})$
- File-index: $\hat{\pi} = \begin{pmatrix} 123 \\ 231 \end{pmatrix}$, $H(W_3, Z_3, X_{1,2,3,2}) = H(W_1, Z_3, X_{2,3,1,3})$

$\Rightarrow$ LP significantly reduced (e.g. 10^8 to 10^4).

The Existence of Optimal Symmetric Codes

Proposition

For any caching code, there is a code with the same or smaller caching memory and transmission rate, which is both user-index-symmetric and file-index-symmetric.

$\Rightarrow$ Without loss of optimality, can consider only symmetric codes.

Example: $(N, K) = (3, 4)$

- User-index: $\bar{\pi} = \begin{pmatrix} 1234 \\ 2314 \end{pmatrix}$, $H(W_2, Z_2, X_{1,2,3,2}) = H(W_2, Z_3, X_{3,1,2,2})$
- File-index: $\hat{\pi} = \begin{pmatrix} 123 \\ 231 \end{pmatrix}$, $H(W_3, Z_3, X_{1,2,3,2}) = H(W_1, Z_3, X_{2,3,1,3})$

$\Rightarrow$ LP significantly reduced (e.g. 10^8 to 10^4).

The Existence of Optimal Symmetric Codes

Proposition

For any caching code, there is a code with the same or smaller caching memory and transmission rate, which is both user-index-symmetric and file-index-symmetric.

$\Rightarrow$ Without loss of optimality, can consider only symmetric codes.

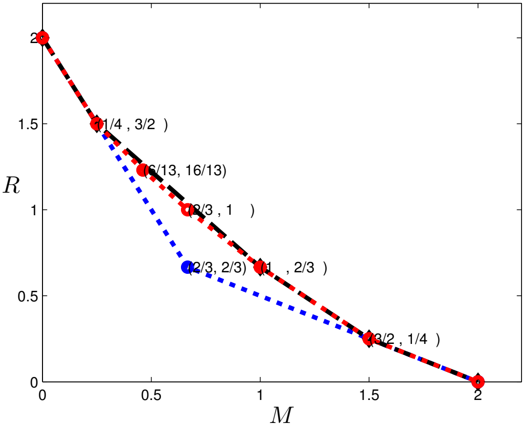
Example: $(N, K) = (3, 4)$

- User-index: $\bar{\pi} = \begin{pmatrix} 1234 \\ 2314 \end{pmatrix}$, $H(W_2, Z_2, X_{1,2,3,2}) = H(W_2, Z_3, X_{3,1,2,2})$
- File-index: $\hat{\pi} = \begin{pmatrix} 123 \\ 231 \end{pmatrix}$, $H(W_3, Z_3, X_{1,2,3,2}) = H(W_1, Z_3, X_{2,3,1,3})$

$\Rightarrow$ LP significantly reduced (e.g. 10^8 to 10^4).

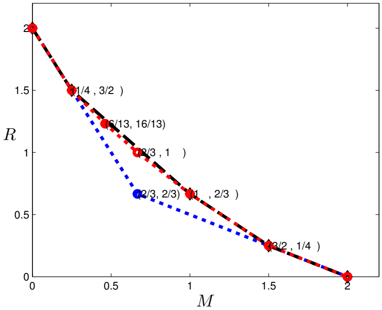
Outline

Reverse-Engineering Codes for $(N, K) = (2, 4)$



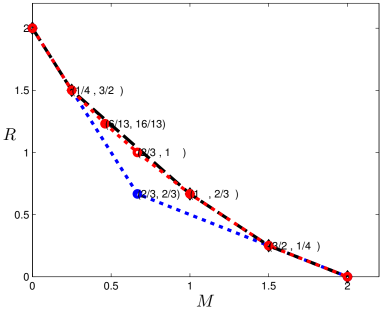
- Simple bounds already tight for $M \in [0, 1/4] \cup [1, 2]$;
- Investigate the bounds, identify a corner point not achieved yet;
- ASSUME it achievable: attempt to design codes.

Extracting Joint Entropies & Reverse Engineering



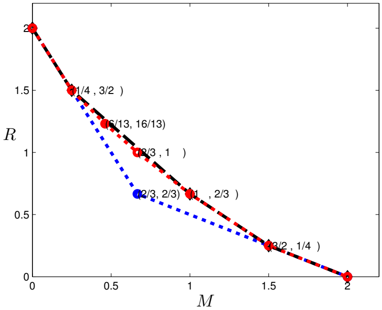
- $(M, R) = (2/3, 1)$: file A, B each has 6 symbols in a finite field;
  - $A = \{A_1, A_2, ..., A_6\}$ and $B = \{B_1, B_2, ..., B_6\}$;
  - Target: a linear code that caches 4 symbols, and delivers 6 symbols?
  - Still hard to design directly.
- New idea: the LP also finds the joint entropy vector in the optimal solution
  - $\Rightarrow$ New target: find a linear code with this particular entropy structure.

Extracting Joint Entropies & Reverse Engineering



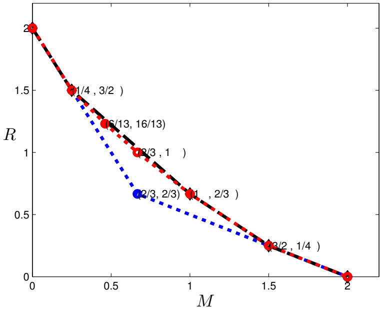
- $(M, R) = (2/3, 1)$: file A, B each has 6 symbols in a finite field;
  - $A = \{A_1, A_2, ..., A_6\}$ and $B = \{B_1, B_2, ..., B_6\}$;
  - Target: a linear code that caches 4 symbols, and delivers 6 symbols?
  - Still hard to design directly.
- New idea: the LP also finds the joint entropy vector in the optimal solution
  - $\Rightarrow$ New target: find a linear code with this particular entropy structure.

Extracting Joint Entropies & Reverse Engineering



- $(M, R) = (2/3, 1)$: file A, B each has 6 symbols in a finite field;
  - $A = \{A_1, A_2, ..., A_6\}$ and $B = \{B_1, B_2, ..., B_6\}$;
  - Target: a linear code that caches 4 symbols, and delivers 6 symbols?
  - Still hard to design directly.
- New idea: the LP also finds the joint entropy vector in the optimal solution
  - $\Rightarrow$ New target: find a linear code with this particular entropy structure.

Extracting Joint Entropies & Reverse Engineering



- $(M, R) = (2/3, 1)$: file A, B each has 6 symbols in a finite field;
  - $A = \{A_1, A_2, ..., A_6\}$ and $B = \{B_1, B_2, ..., B_6\}$;
  - Target: a linear code that caches 4 symbols, and delivers 6 symbols?
  - Still hard to design directly.
- New idea: the LP also finds the joint entropy vector in the optimal solution
  - $\Rightarrow$ New target: find a linear code with this particular entropy structure.

Extracting Joint Entropies & Reverse Engineering

For the delivery part:

Joint entropy	value
$H(X_{1,1,1,2})$	6
$H(X_{1,1,2,2})$	6
$H(X_{1,1,1,2}\|A)$	3
$H(X_{1,1,1,2}\|B)$	3
$H(X_{1,1,2,2}\|A)$	3

$X_{1,1,1,2} =$

$$\left[\begin{array}{ccccc|cccc} * & * & * & \cdots & * & \# & \# & \cdots & \# \\ * & * & * & \cdots & * & \# & \# & \cdots & \# \\ * & * & * & \cdots & * & \# & \# & \cdots & \# \\ * & * & * & \cdots & * & \# & \# & \cdots & \# \\ * & * & * & \cdots & * & \# & \# & \cdots & \# \\ * & * & * & \cdots & * & \# & \# & \cdots & \# \end{array}\right] \cdot \begin{bmatrix} A_1 \\ \cdots \\ A_6 \\ - \\ B_1 \\ \cdots \\ B_6 \end{bmatrix}$$

$H(X_{1,1,1,2}|A) = H(X_{1,1,1,2}|B) = 3$

$\Rightarrow$ The linear combinations of B's span dimension 3

$\Rightarrow$ The linear combinations of A's span dimension 3

$\Rightarrow$ Recall $X_{1,1,1,2}$'s has dimension 6

$\Rightarrow$ No need to mix A and B in the delivery $X_{1,1,1,2}$!

For the delivery part:

Joint entropy	value
$H(X_{1,1,1,2})$	6
$H(X_{1,1,2,2})$	6
$H(X_{1,1,1,2}\vert A)$	3
$H(X_{1,1,1,2}\vert B)$	3
$H(X_{1,1,2,2}\vert A)$	3

$$X_{1,1,1,2} = \left[\begin{array}{ccccc|cccc} * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \end{array}\right] \cdot \begin{bmatrix} A_1 \\ \dots \\ A_6 \\ - \\ B_1 \\ \dots \\ B_6 \end{bmatrix}$$

$H(X_{1,1,1,2}|A) = H(X_{1,1,1,2}|B) = 3$

$\Rightarrow$ The linear combinations of B's span dimension 3

$\Rightarrow$ The linear combinations of A's span dimension 3

$\Rightarrow$ Recall $X_{1,1,1,2}$'s has dimension 6

$\Rightarrow$ No need to mix A and B in the delivery $X_{1,1,1,2}$!

Extracting Joint Entropies & Reverse Engineering

For the delivery part:

Joint entropy	value
$H(X_{1,1,1,2})$	6
$H(X_{1,1,2,2})$	6
$H(X_{1,1,1,2}\|A)$	3
$H(X_{1,1,1,2}\|B)$	3
$H(X_{1,1,2,2}\|A)$	3

$$X_{1,1,1,2} = \left[\begin{array}{ccccc|cccc} * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \\ * & * & * & \dots & * & \# & \# & \dots & \# \end{array}\right] \cdot \begin{bmatrix} A_1 \\ \dots \\ A_6 \\ - \\ B_1 \\ \dots \\ B_6 \end{bmatrix}$$

$H(X_{1,1,1,2}|A) = H(X_{1,1,1,2}|B) = 3$

$\Rightarrow$ The linear combinations of B's span dimension 3

$\Rightarrow$ The linear combinations of A's span dimension 3

$\Rightarrow$ Recall $X_{1,1,1,2}$'s has dimension 6

$\Rightarrow$ No need to mix A and B in the delivery $X_{1,1,1,2}$!

Extracting Joint Entropies & Reverse Engineering

For the placement part:

Joint entropy	value
$H(Z_1\|A)$	3
$H(Z_1, Z_2\|A)$	5
$H(Z_1, Z_2, Z_3\|A)$	6
$H(Z_1, Z_2, Z_3, Z_4\|A)$	6

Any one user cache $\rightarrow$ 3 pieces of B_i's, any two-user caches $\rightarrow$ 5, any three-user caches $\rightarrow$ 6

$\Rightarrow$ Each symbol placed at 2 users's cache, as a component of linear combinations.

Extracting Joint Entropies & Reverse Engineering

For the placement part:

Joint entropy	value
$H(Z_1 \vert A)$	3
$H(Z_1, Z_2 \vert A)$	5
$H(Z_1, Z_2, Z_3 \vert A)$	6
$H(Z_1, Z_2, Z_3, Z_4 \vert A)$	6

Any one user cache $\rightarrow$ 3 pieces of B_i's, any two-user caches $\rightarrow$ 5, any three-user caches $\rightarrow$ 6
$\Rightarrow$ Each symbol placed at 2 users's cache, as a component of linear combinations.

User 1		B_1	B_2	B_3
User 2	B_1		B_4	B_5
User 3	B_2	B_4		B_6
User 4	B_3	B_5	B_6	

A New Code for $(N, K) = (2, 4)$

Much easier to construct the code with those clues.

Z_1	$A_1 + B_1$	$A_2 + B_2$	$A_3 + B_3$	$A_1 + A_2 + A_3 + 2(B_1 + B_2 + B_3)$
Z_2	$A_1 + B_1$	$A_4 + B_4$	$A_5 + B_5$	$A_1 + A_4 + A_5 + 2(B_1 + B_4 + B_5)$
Z_3	$A_2 + B_2$	$A_4 + B_4$	$A_6 + B_6$	$A_2 + A_4 + A_6 + 2(B_2 + B_4 + B_6)$
Z_4	$A_3 + B_3$	$A_5 + B_5$	$A_6 + B_6$	$A_3 + A_5 + A_6 + 2(B_3 + B_5 + B_6)$

Requests are (A, A, A, B), send $X_{1,1,1,2}$

$$B_1, B_2, B_4; A_3 + 2A_5 + 3A_6, A_3 + 3A_5 + 4A_6; A_1 + A_2 + A_4.$$

Requests are (A, A, B, B), send $X_{1,1,2,2}$

$$B_1, A_6; A_2 + 2A_4, A_3 + 2A_5, B_2 + 2B_3, B_4 + 2B_5$$

A New Code for $(N, K) = (2, 4)$

Much easier to construct the code with those clues.

Z_1	$A_1 + B_1$	$A_2 + B_2$	$A_3 + B_3$	$A_1 + A_2 + A_3 + 2(B_1 + B_2 + B_3)$
Z_2	$A_1 + B_1$	$A_4 + B_4$	$A_5 + B_5$	$A_1 + A_4 + A_5 + 2(B_1 + B_4 + B_5)$
Z_3	$A_2 + B_2$	$A_4 + B_4$	$A_6 + B_6$	$A_2 + A_4 + A_6 + 2(B_2 + B_4 + B_6)$
Z_4	$A_3 + B_3$	$A_5 + B_5$	$A_6 + B_6$	$A_3 + A_5 + A_6 + 2(B_3 + B_5 + B_6)$

Requests are (A, A, A, B), send $X_{1,1,1,2}$

$$B_1, B_2, B_4; A_3 + 2A_5 + 3A_6, A_3 + 3A_5 + 4A_6; A_1 + A_2 + A_4.$$

Requests are (A, A, B, B), send $X_{1,1,2,2}$

$$B_1, A_6; A_2 + 2A_4, A_3 + 2A_5, B_2 + 2B_3, B_4 + 2B_5$$

Generalization to Other (N, K)

Code can be generalized (T. & Chen TIT-2018):

- Choose the numbers of combinations to cache and transmit;
- Choose the coefficients nicely: full rank conditions.

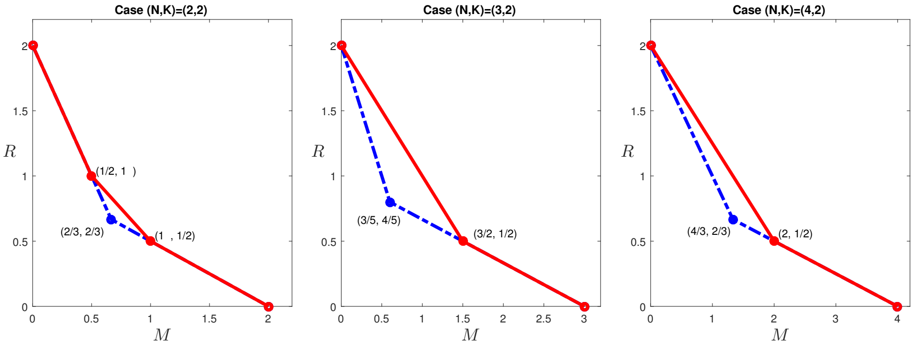
Theorem

For $N \in \mathbb{N}$ files and $K \in \mathbb{N}$ users each with a cache of size M, and $N \leq K$, the following (M, R) pairs are achievable

$$\left(\frac{t[(N-1)t+K-N]}{K(K-1)}, \frac{N(K-t)}{K}\right), \qquad t = 0, 1, \ldots, K.$$

Outline

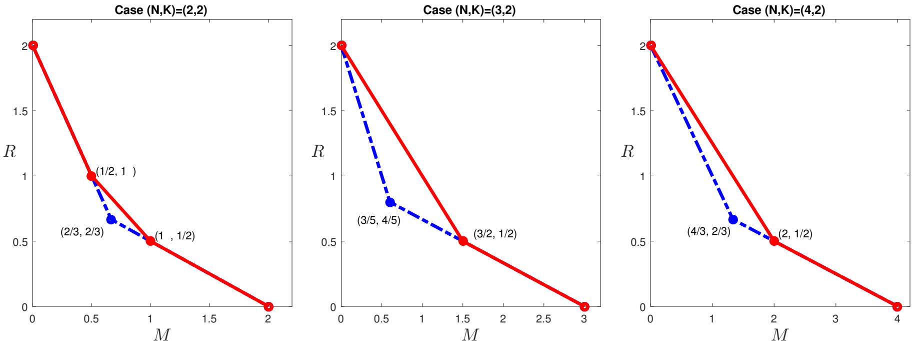
A Data Driven Hypothesis: Connection Cross Instances



Red line: optimal tradeoff; Blue dash-dot: cutset outer bound

- Use the computational approach to first find solutions for $N = 3, 4$;
- For $N = 3, 4$, the upper corner point disappears (surprise!);
- Hypothesis: one corner point $(M, R) = (N/2, 1/2)$ if $(N \geq 3, K = 2)$.

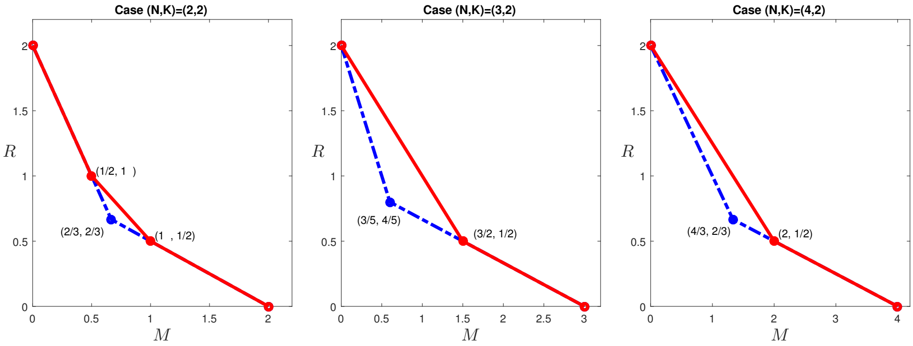
A Data Driven Hypothesis: Connection Cross Instances



Red line: optimal tradeoff; Blue dash-dot: cutset outer bound

- Use the computational approach to first find solutions for $N = 3, 4$;
- For $N = 3, 4$, the upper corner point disappears (surprise!);
- Hypothesis: one corner point $(M, R) = (N/2, 1/2)$ if $(N \geq 3, K = 2)$.

A Data Driven Hypothesis: Connection Cross Instances



Red line: optimal tradeoff; Blue dash-dot: cutset outer bound

- Use the computational approach to first find solutions for $N = 3, 4$;
- For $N = 3, 4$, the upper corner point disappears (surprise!);
- Hypothesis: one corner point $(M, R) = (N/2, 1/2)$ if $(N \geq 3, K = 2)$.

A Complete Characterization for $K = 2$

Theorem

Converse: for $(N, K) = (N, 2)$ *and* $N \geq 3$*, the* (M, R) *pair must satisfy*

$$3M + NR \geq 2N, \quad M + NR \geq N. \tag{1}$$

Forward: any nonnegative (M, R) *pair satisfying (1) is achievable.*

- The first collection of cases to have a complete solution;
- Generate explicit proofs using LP-dual, and find a general pattern;
- This generalization is not computer-produced ☹, but inspired by it.

A Complete Characterization for $K = 2$

Theorem

Converse: for $(N, K) = (N, 2)$ and $N \geq 3$, the (M, R) pair must satisfy

$$3M + NR \geq 2N, \quad M + NR \geq N. \tag{1}$$

Forward: any nonnegative (M, R) pair satisfying (1) is achievable.

- The first collection of cases to have a complete solution;
- Generate explicit proofs using LP-dual, and find a general pattern;
- This generalization is not computer-produced ☹, but inspired by it.

A Complete Characterization for $K = 2$

Theorem

Converse: for $(N, K) = (N, 2)$ and $N \geq 3$, the (M, R) pair must satisfy

$$3M + NR \geq 2N, \quad M + NR \geq N. \tag{1}$$

Forward: any nonnegative (M, R) pair satisfying (1) is achievable.

- The first collection of cases to have a complete solution;
- Generate explicit proofs using LP-dual, and find a general pattern;
- This generalization is not computer-produced ☹, but inspired by it.

A Complete Characterization for $K = 2$

Theorem

Converse: for $(N, K) = (N, 2)$ *and* $N \geq 3$*, the* (M, R) *pair must satisfy*

$$3M + NR \geq 2N, \quad M + NR \geq N. \tag{1}$$

Forward: any nonnegative (M, R) *pair satisfying (1) is achievable.*

- The first collection of cases to have a complete solution;
- Generate explicit proofs using LP-dual, and find a general pattern;
- This generalization is not computer-produced ☹, but inspired by it.

Outline

Difficulty for Larger Cases

Complexity increases quickly with problem parameters:

- Number of R.V.s in caching: $N + K + N^K$;
- Number of LP constraints after symmetry-reduction:

$$\approx \frac{\binom{N+K+N^K}{2} 2^{N+K+N^K-2}}{N!K!}$$

- $(N, K) = (6, 3)$, 225 R.V.s, $\approx 7.8 \times 10^{67}$ LP constraints after symmetry reduction (there are $\approx 1.33 \times 10^{50}$ atoms on earth ☹).

Run out of memory $\Rightarrow$ Idea: Relax the problem strategically

Difficulty for Larger Cases

Complexity increases quickly with problem parameters:

- Number of R.V.s in caching: $N + K + N^K$;
- Number of LP constraints after symmetry-reduction:

$$\approx \frac{\binom{N+K+N^K}{2} 2^{N+K+N^K-2}}{N!K!}$$

- $(N, K) = (6, 3)$, 225 R.V.s, $\approx 7.8 \times 10^{67}$ LP constraints after symmetry reduction (there are $\approx 1.33 \times 10^{50}$ atoms on earth ☹).

Run out of memory $\Rightarrow$ Idea: Relax the problem strategically

Difficulty for Larger Cases

Complexity increases quickly with problem parameters:

- Number of R.V.s in caching: $N + K + N^K$;
- Number of LP constraints after symmetry-reduction:

$$\approx \frac{\binom{N+K+N^K}{2} 2^{N+K+N^K-2}}{N!K!}$$

- $(N, K) = (6, 3)$, 225 R.V.s, $\approx 7.8 \times 10^{67}$ LP constraints after symmetry reduction (there are $\approx 1.33 \times 10^{50}$ atoms on earth ☹).

Run out of memory ⇒ Idea: Relax the problem strategically

Difficulty for Larger Cases

Complexity increases quickly with problem parameters:

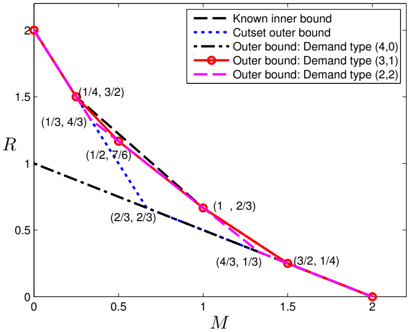
- Number of R.V.s in caching: $N + K + N^K$;
- Number of LP constraints after symmetry-reduction:

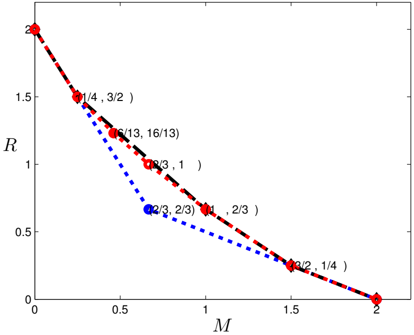
$$\approx \frac{\binom{N+K+N^K}{2} 2^{N+K+N^K-2}}{N!K!}$$

- $(N, K) = (6, 3)$, 225 R.V.s, $\approx 7.8 \times 10^{67}$ LP constraints after symmetry reduction (there are $\approx 1.33 \times 10^{50}$ atoms on earth ☹).

Run out of memory $\Rightarrow$ Idea: Relax the problem strategically

Equivalent Bounds Using Few Demands

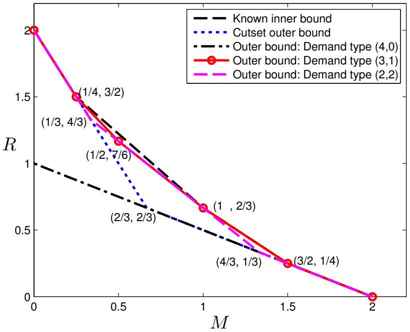


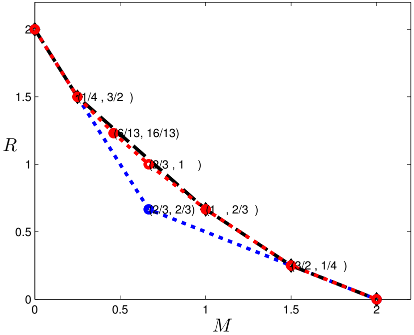


Finding: Equivalent bounds can be obtained with only some demands.

- $(N, K) = (2, 4)$, only $\mathcal{W} \cup \mathcal{Z} \cup \{X_{1,1,1,2}, X_{1,1,2,2}\}$: $22 \Rightarrow 8$ R.V.s;
- $(N, K) = (3, 3)$, only $\mathcal{W} \cup \mathcal{Z} \cup \{X_{2,1,1}, X_{3,1,1}, X_{3,2,1}\}$: $30 \Rightarrow 9$ R.V.s .

Equivalent Bounds Using Few Demands

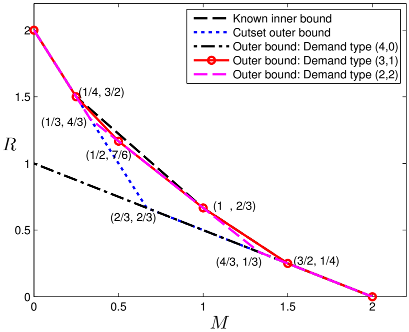


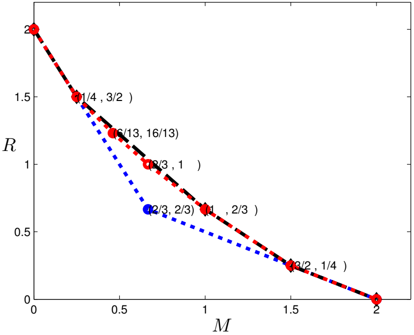


Finding: Equivalent bounds can be obtained with only some demands.

- $(N, K) = (2, 4)$, only $\mathcal{W} \cup \mathcal{Z} \cup \{X_{1,1,1,2}, X_{1,1,2,2}\}$: $22 \Rightarrow 8$ R.V.s;
- $(N, K) = (3, 3)$, only $\mathcal{W} \cup \mathcal{Z} \cup \{X_{2,1,1}, X_{3,1,1}, X_{3,2,1}\}$: $30 \Rightarrow 9$ R.V.s .

Equivalent Bounds Using Few Demands

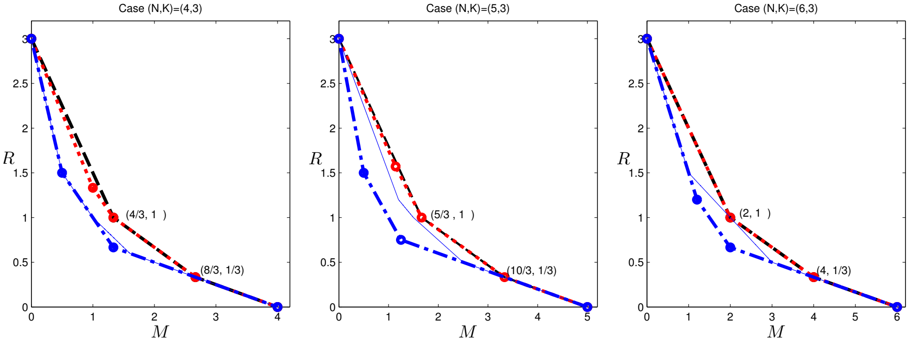




Finding: Equivalent bounds can be obtained with only some demands.

- $(N, K) = (2, 4)$, only $\mathcal{W} \cup \mathcal{Z} \cup \{X_{1,1,1,2}, X_{1,1,2,2}\}$: $22 \Rightarrow 8$ R.V.s;
- $(N, K) = (3, 3)$, only $\mathcal{W} \cup \mathcal{Z} \cup \{X_{2,1,1}, X_{3,1,1}, X_{3,2,1}\}$: $30 \Rightarrow 9$ R.V.s .

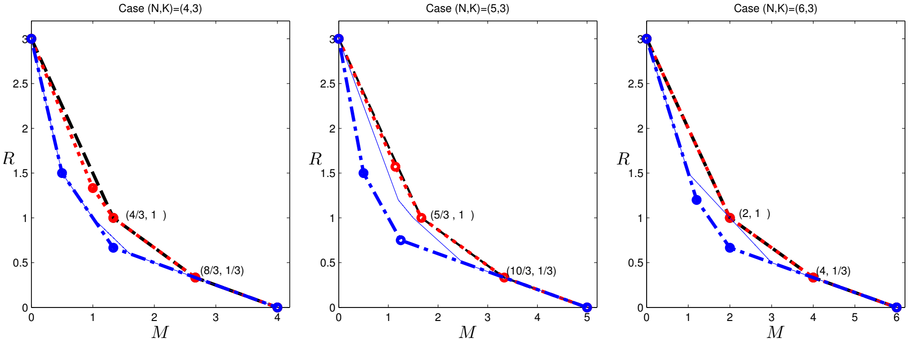
Exploration and More Bounds



Red dotted line: computed outer bounds; blue dashed-dot lines: cut-set outer bounds; black dashed lines: inner bounds; thin blue lines: outer bounds by Ghasemi and Ramamoorthy.

- Many new observations and hypotheses;
- Recall $(6,3)$, we had 10^{67} LP constraints: Happen to solve this case completely!

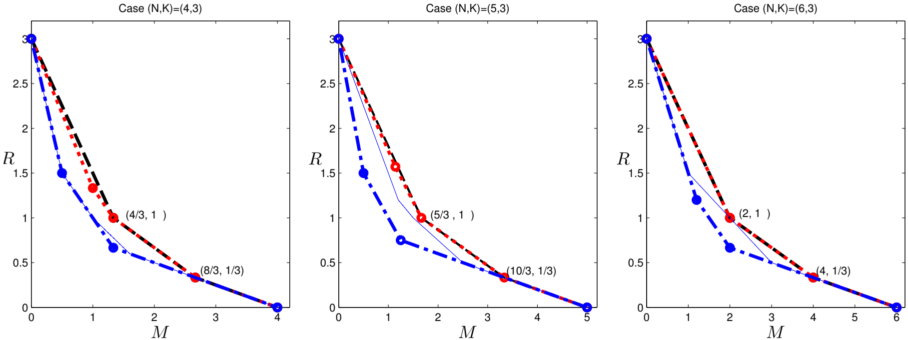
Exploration and More Bounds



Red dotted line: computed outer bounds; blue dashed-dot lines: cut-set outer bounds; black dashed lines: inner bounds; thin blue lines: outer bounds by Ghasemi and Ramamoorthy.

- Many new observations and hypotheses;
- Recall $(6,3)$, we had 10^{67} LP constraints: Happen to solve this case completely!

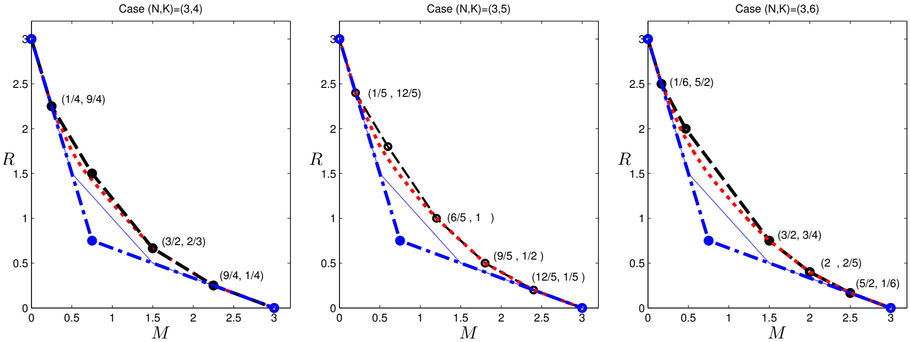
Exploration and More Bounds



Red dotted line: computed outer bounds; blue dashed-dot lines: cut-set outer bounds; black dashed lines: inner bounds; thin blue lines: outer bounds by Ghasemi and Ramamoorthy.

- Many new observations and hypotheses;
- Recall $(6,3)$, we had 10^{67} LP constraints: Happen to solve this case completely!

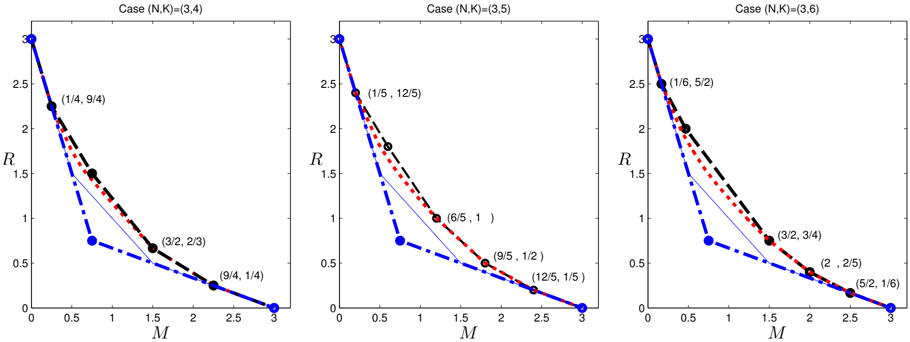
Exploration and More Bounds



Red dotted line: computed outer bounds; blue dashed-dot lines: cut-set outer bounds; black dashed lines: inner bounds; thin blue lines: outer bounds by Ghasemi and Ramamoorthy.

- Significant improved outer bounds, and some further generalized by Yu et al. TIT-18.
- More details: T. Entropy MDPI-18.

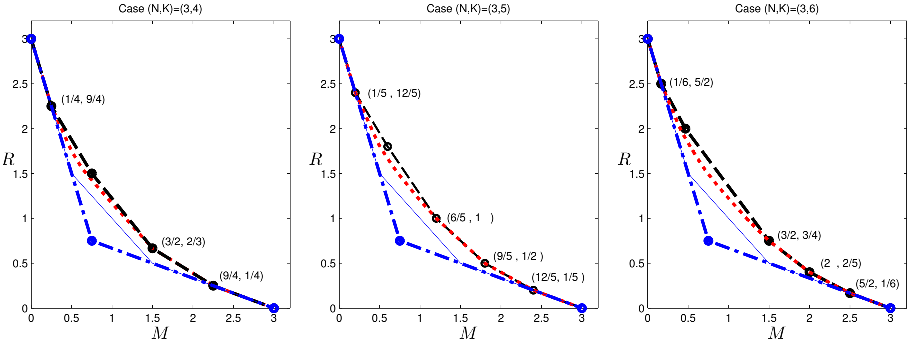
Exploration and More Bounds



Red dotted line: computed outer bounds; blue dashed-dot lines: cut-set outer bounds; black dashed lines: inner bounds; thin blue lines: outer bounds by Ghasemi and Ramamoorthy.

- Significant improved outer bounds, and some further generalized by Yu et al. TIT-18;
- More details: T. Entropy MDPI-18.

Exploration and More Bounds



Red dotted line: computed outer bounds; blue dashed-dot lines: cut-set outer bounds; black dashed lines: inner bounds;
thin blue lines: outer bounds by Ghasemi and Ramamoorthy.

- Significant improved outer bounds, and some further generalized by Yu et al. TIT-18;
- More details: T. Entropy MDPI-18.

Outline

The Computer-Aided Investigation (CAI) Toolbox

We open-sourced a package to streamline many of the functionalities (C/C++/Python):

- Formatted problem description file: specify the coding problem;
- Use symmetry to perform reduction;
- Compute bounds, generate proofs, trace out convex hull, readout joint entropy values, sensitivity analysis, etc.

One caveat:

- Requires a local LP solver backend: Cplex or Gurobi
- Cplex or Gurobi are commercial solvers but free for academic users;
- Known to be significantly faster than open-source solvers and have various additional functionalities;
- General academic license not set up for online computing access.

https://github.com/ct2641/CAI

The Computer-Aided Investigation (CAI) Toolbox

We open-sourced a package to streamline many of the functionalities (C/C++/Python):

- Formatted problem description file: specify the coding problem;
- Use symmetry to perform reduction;
- Compute bounds, generate proofs, trace out convex hull, readout joint entropy values, sensitivity analysis, etc.

One caveat:

- Requires a local LP solver backend: Cplex or Gurobi
- Cplex or Gurobi are commercial solvers but free for academic users;
- Known to be significantly faster than open-source solvers and have various additional functionalities;
- General academic license not set up for online computing access.

`https://github.com/ct2641/CAI`

An Example: $(4, 3, 3)$ Regenerating Code

```
//PDRG4x3x3.txt: problem description file for the (4,3,3) regenerating code problem.

Random variables:
W1,W2,W3,W4,S12,S13,S14,S21,S23,S24,S31,S32,S34,S41,S42,S43

Additional LP variables:
A,B

Objective:
A+B

Dependency:
S12,S13,S14:W1
S21,S23,S24:W2
S31,S32,S34:W3
S41,S42,S43:W4
W1:S21,S31,S41
W2:S12,S32,S42
W3:S13,S23,S43
W4:S14,S24,S34

Constant bounds:
H(W1)-A<=0
H(S12)-B<=0
H(W1,W2,W3,W4)>=1

Symmetry:
W1,W2,W3,W4,S12,S13,S14,S21,S23,S24,S31,S32,S34,S41,S42,S43
W1,W2,W4,W3,S12,S14,S13,S21,S24,S23,S41,S42,S43,S31,S32,S34
W1,W3,W2,W4,S13,S12,S14,S31,S32,S34,S21,S23,S24,S41,S43,S42
W1,W4,W3,W2,S14,S13,S12,S41,S43,S42,S31,S34,S32,S21,S24,S23
```

An Example: $(4, 3, 3)$ Regenerating Code – Continued

```
...
W3,W2,W1,W4,S32,S31,S34,S23,S21,S24,S13,S12,S14,S43,S42,S41
W3,W2,W4,W1,S32,S34,S31,S23,S24,S21,S43,S42,S41,S13,S12,S14
W3,W1,W2,W4,S31,S32,S34,S13,S12,S14,S23,S21,S24,S43,S41,S42
W3,W1,W4,W2,S31,S34,S32,S13,S14,S12,S43,S41,S42,S23,S21,S24
W3,W4,W1,W2,S34,S31,S32,S43,S41,S42,S13,S14,S12,S23,S24,S21
W3,W4,W2,W1,S34,S32,S31,S43,S42,S41,S23,S24,S21,S13,S14,S12
W4,W2,W3,W1,S42,S43,S41,S24,S23,S21,S34,S32,S31,S14,S12,S13
W4,W2,W1,W3,S42,S41,S43,S24,S21,S23,S14,S12,S13,S34,S32,S31
W4,W1,W3,W2,S41,S43,S42,S14,S13,S12,S34,S31,S32,S24,S21,S23
W4,W1,W2,W3,S41,S42,S43,S14,S12,S13,S24,S21,S23,S34,S31,S32
W4,W3,W1,W2,S43,S41,S42,S34,S31,S32,S14,S13,S12,S24,S23,S21
W4,W3,W2,W1,S43,S42,S41,S34,S32,S31,S24,S23,S21,S14,S13,S12

Bounds to prove:
8A+12B>=6

end
```

An Example: $(4, 3, 3)$ Regenerating Code Result

Simple lower bound computation:

```
***************************************************************
-The following 16 random variables were found:
W1 W2 W3 W4 S12 S13 S14 S21 S23 S24 S31 S32 S34 S41 S42 S43
---The problem has 2 additional LP variables.
---The objective function has 2 non-zero terms.
---The problem has 8 dependency relations.
---The problem has 3 constant value bounds.
---Permutations in the symmetry relation =  24.
---Number of bounds to prove =  1.
***************************************************************
Total number of elements to reduce: 65536
CPXPARAM_Read_DataCheck                          1
Tried aggregator 1 time.
DUAL formed by presolve
LP Presolve eliminated 38643 rows and 3 columns.
Reduced LP has 177 rows, 5084 columns, and 17831 nonzeros.
Presolve time = 0.05 sec. (24.27 ticks)
Parallel mode: using up to 20 threads for barrier.
Number of nonzeros in lower triangle of A*A' = 5960
Using Approximate Minimum Degree ordering
Total time for automatic ordering = 0.00 sec. (0.44 ticks)
Summary statistics for Cholesky factor:
  Threads                  = 20
  Rows in Factor           = 177
  Integer space required   = 923
  Total non-zeros in factor = 12808
  Total FP ops to factor   = 1232248
 Itn      Primal Obj        Dual Obj  Prim Inf Upper Inf  Dual Inf Inf Ratio
```

An Example: $(4,3,3)$ Regenerating Code Result – Continued

```
Itn      Primal Obj        Dual Obj  Prim Inf Upper Inf  Dual Inf Inf Ratio
   0   1.0000000e+01   0.0000000e+00  3.77e+04  0.00e+00  5.26e+03  1.00e+00
   1   6.1373374e+00   1.5078468e+00  2.60e+04  0.00e+00  3.34e+03  2.44e+00
   2   2.2445860e+00   1.4338756e+00  1.07e+04  0.00e+00  1.06e+03  8.08e+04
   3   1.5228039e+00   9.6993458e-01  4.73e+03  0.00e+00  1.81e+02  9.34e+01
   4   1.0830951e+00   9.9552024e-01  7.79e+02  0.00e+00  1.94e+01  1.01e+03
   5   1.0262480e+00   9.8688399e-01  3.25e+02  0.00e+00  2.65e+00  7.06e+03
   6   1.0237260e+00   9.7472519e-01  3.14e+02  0.00e+00  1.78e+00  6.55e+03
   7   9.5978567e-01   9.5337070e-01  4.00e+01  0.00e+00  2.14e-01  5.25e+04
   8   9.0444925e-01   8.9317974e-01  2.54e+01  0.00e+00  1.26e-01  6.17e+04
   9   8.0010814e-01   8.3512233e-01  7.88e+00  0.00e+00  7.35e-02  8.41e+04
  10   7.6631321e-01   7.7048934e-01  5.74e+00  0.00e+00  4.38e-02  1.33e+05
  11   7.0912848e-01   7.0989975e-01  2.21e+00  0.00e+00  1.64e-02  3.38e+05
  12   6.5432023e-01   6.5701740e-01  6.27e-01  0.00e+00  5.20e-03  9.72e+05
  12   6.5432023e-01   6.5701740e-01  6.27e-01  0.00e+00  5.20e-03  9.72e+05
  13   6.2554618e-01   6.2728184e-01  1.45e-02  0.00e+00  3.91e-04  1.33e+07
  14   6.2499995e-01   6.2500055e-01  1.44e-06  0.00e+00  1.13e-07  5.06e+10
  15   6.2500000e-01   6.2500000e-01  2.11e-10  0.00e+00  1.58e-11  4.65e+14
Barrier time = 0.11 sec. (33.54 ticks)

Total time on 20 threads = 0.11 sec. (33.54 ticks)

*****************************************************************
Optimal value is 0.625000.
Values achieving the optimal solution:
 0.375000   0.250000
*****************************************************************
```

An Example: $(4, 3, 3)$ Regenerating Code Result – Continued

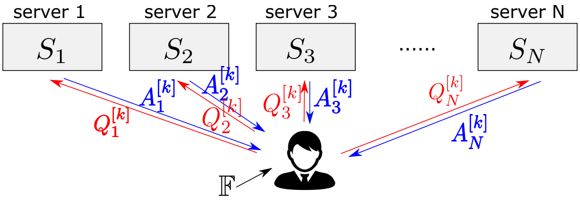
Trace out the convex hull:

```
*****************************************************************
-The following 16 random variables were found:
W1 W2 W3 W4 S12 S13 S14 S21 S23 S24 S31 S32 S34 S41 S42 S43
---The problem has 2 additional LP variables.
---The objective function has 2 non-zero terms.
---The problem has 8 dependency relations.
---The problem has 3 constant value bounds.
---Permutations in the symmetry relation =  24.
---Number of bounds to prove =  1.
*****************************************************************
Total number of elements to reduce: 65536
New point (0.333333, 0.333333).
New point (0.500000, 0.166667).
New point (0.375000, 0.250000).


List of found points on the hull:
(0.333333, 0.333333).
(0.375000, 0.250000).
(0.500000, 0.166667).
End of list of found points.
```

Outline

Third Setting: Private Information Retrieval



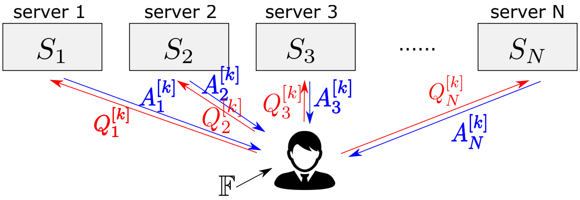
Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \ldots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \ldots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Third Setting: Private Information Retrieval



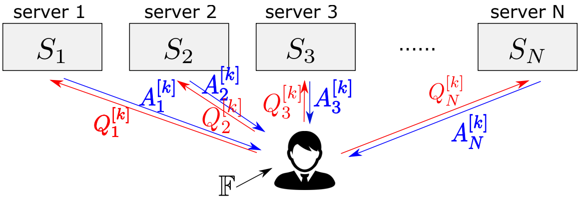
Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \ldots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \ldots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Third Setting: Private Information Retrieval



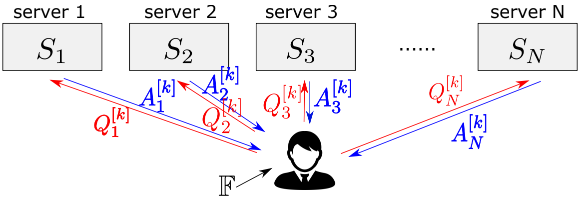
Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \ldots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \ldots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Third Setting: Private Information Retrieval



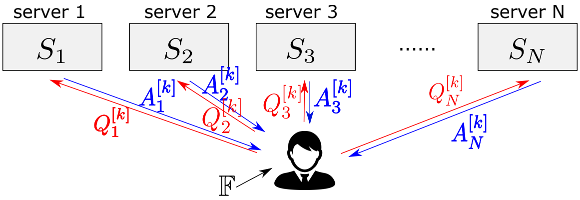
Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \ldots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \ldots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Third Setting: Private Information Retrieval



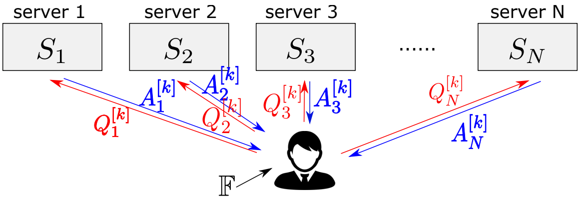
Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \dots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \dots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Third Setting: Private Information Retrieval



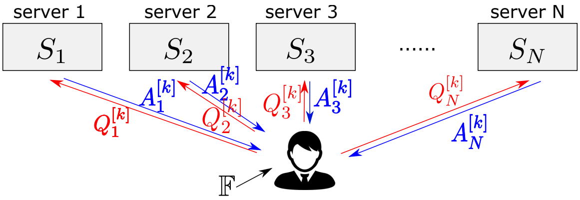
Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \ldots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \ldots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Third Setting: Private Information Retrieval



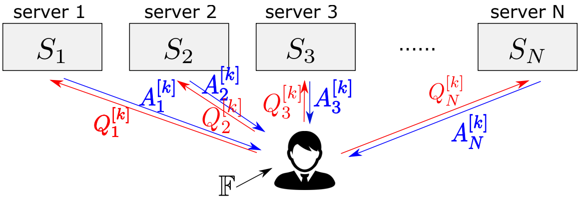
Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \ldots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \ldots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Third Setting: Private Information Retrieval



Retrieval protocols: K messages (of unit rate each) & N servers

- To request W_k: with a random key $\mathbb{F}$, user generates queries $Q_1^{[k]}, \ldots, Q_N^{[k]}$;
- Servers: return answers $A_1^{[k]}, \ldots, A_N^{[k]}$ after receiving the queries;
- User recovers $\hat{W}_k = \psi(A_{1:N}^{[k]}, k, \mathbb{F})$.

Requirements: retrieve correctly, but keep the identity of the message private

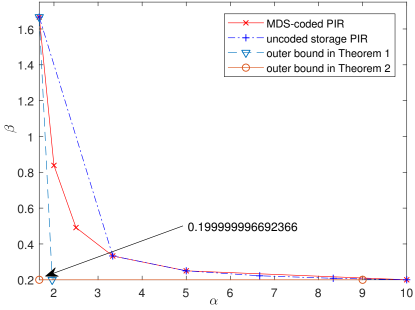
- Correctness: $W_k = \hat{W}_k$;
- Privacy: the query distribution $\mathbf{Pr}(Q_n^{[k]} = q) = \mathbf{Pr}(Q_n^{[k']} = q)$.

Two Bounds Obtained Through Computer-Aided Exploration

Problem setup: $2NK + K + 1$ random variables.

Identical distribution for retrieving different message $\Rightarrow$ Constraints on entropy as equality.

i.e., $(A_n^{[k]}, Q_n^{[k]}, W_{1:K}, S_{1:N}) \sim (A_n^{[k']}, Q_n^{[k']}, W_{1:K}, S_{1:N}) \Rightarrow$
$H(Q_n^{[k]}, W_1, W_2) = H(Q_n^{[k']}, W_1, W_2)$, $H(A_n^{[k]}, W_1) = H(A_n^{[k']}, W_1)$, ...



Two tradeoff bounds between storage α and download β:

Almost vertical bound:

$\beta + (N-1)\alpha \geq K.$

Almost horizontal bound:

$$\frac{\alpha+(N-1)\beta}{N-2} + N^{K-1}\beta \geq \frac{K}{N-2} + \frac{N^K-1}{N(N-1)}.$$

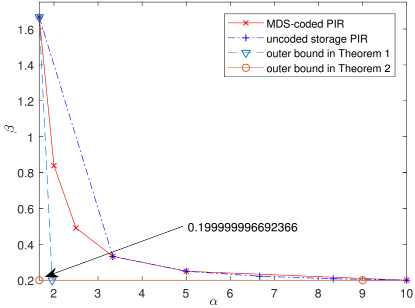
More details: T. TIT-20; Guo et al., JSAIT-21.

Two Bounds Obtained Through Computer-Aided Exploration

Problem setup: $2NK + K + 1$ random variables.

Identical distribution for retrieving different message $\Rightarrow$ Constraints on entropy as equality.

i.e., $(A_n^{[k]}, Q_n^{[k]}, W_{1:K}, S_{1:N}) \sim (A_n^{[k']}, Q_n^{[k']}, W_{1:K}, S_{1:N}) \Rightarrow$
$H(Q_n^{[k]}, W_1, W_2) = H(Q_n^{[k']}, W_1, W_2)$, $H(A_n^{[k]}, W_1) = H(A_n^{[k']}, W_1)$, ...



Two tradeoff bounds between storage α and download β:

Almost vertical bound:

$$\beta + (N-1)\alpha \geq K.$$

Almost horizontal bound:

$$\frac{\alpha+(N-1)\beta}{N-2} + N^{K-1}\beta \geq \frac{K}{N-2} + \frac{N^K-1}{N(N-1)}.$$

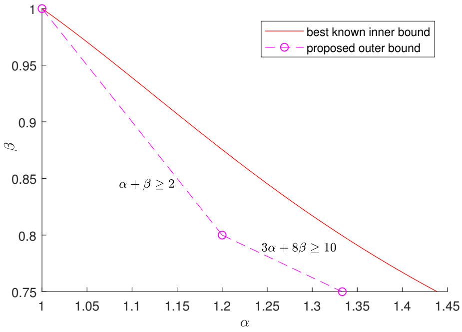
More details: T. TIT-20; Guo et al., JSAIT-21.

Outline

For the PIR problem with storage constraint:

Bound 3:

When $N = K = 2$, $3\alpha + 8\beta \geq 10$.



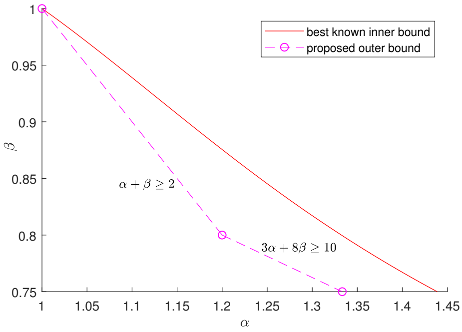
- Relies on a novel pseudo-message technique: non-Shannon-type inequality (T. TIT-20).

Utilizing Non-Shannon-Type Inequalities

For the PIR problem with storage constraint:

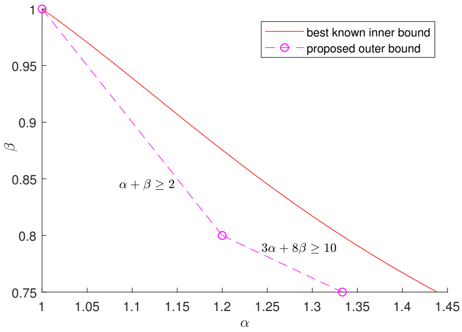
Bound 3:

When $N = K = 2$, $3\alpha + 8\beta \geq 10$.



- Relies on a novel pseudo-message technique: non-Shannon-type inequality (T. TIT-20).

Some Details



Three steps to derive this new bound:

1. Symmetry reduction: w.l.o.o., assume equal rate for all answers, so are storage contents;
2. Consider a subtle dependence structure among answers;
3. Introducing pseudo-messages: extended probability space to derive the bound.

Note: a different problem representation from that to derive the generic bounds just now.

Proof of Bound 3: Step 1

In a general PIR storage code:

- Storage contents may have different rates at different servers;
- Different answers may have different rates.

Symmetrize the code: through server-symmetry and variety-symmetry (Tian et al. IT-19)

- Storage rate: $H(S_n) = H(S_{n'})$ for any two servers $n, n' \in \{1, 2, \ldots, N\} \Rightarrow H(S_n) \leq \alpha$;
- Answer rate: $H(A_n^{(q)}) = H(A_{n'}^{(q')})$, q, q' are the query indices $\Rightarrow H(A_n^{(q)}) \leq \beta$;
- Joint entropy rate: $H(A_n^q, W_k) = H(A_{n'}^{(q')}, W_{k'})$, $n, n' \in \{1, 2, \ldots, N\}$, q, q' are query indices, $k, k' \in \{1, 2, \ldots, K\}$.

Proof of Bound 3: Step 1

In a general PIR storage code:

- Storage contents may have different rates at different servers;
- Different answers may have different rates.

Symmetrize the code: through server-symmetry and variety-symmetry (Tian et al. IT-19)

- Storage rate: $H(S_n) = H(S_{n'})$ for any two servers $n, n' \in \{1, 2, \ldots, N\} \Rightarrow H(S_n) \leq \alpha$;
- Answer rate: $H(A_n^{(q)}) = H(A_{n'}^{(q')})$, q, q' are the query indices $\Rightarrow H(A_n^{(q)}) \leq \beta$;
- Joint entropy rate: $H(A_n^q, W_k) = H(A_{n'}^{(q')}, W_{k'})$, $n, n' \in \{1, 2, \ldots, N\}$, q, q' are query indices, $k, k' \in \{1, 2, \ldots, K\}$.

Proof of Bound 3: Step 1

In a general PIR storage code:

- Storage contents may have different rates at different servers;
- Different answers may have different rates.

Symmetrize the code: through server-symmetry and variety-symmetry (Tian et al. IT-19)

- Storage rate: $H(S_n) = H(S_{n'})$ for any two servers $n, n' \in \{1, 2, \ldots, N\} \Rightarrow H(S_n) \leq \alpha$;
- Answer rate: $H(A_n^{(q)}) = H(A_{n'}^{(q')})$, q, q' are the query indices $\Rightarrow H(A_n^{(q)}) \leq \beta$;
- Joint entropy rate: $H(A_n^q, W_k) = H(A_{n'}^{(q')}, W_{k'})$, $n, n' \in \{1, 2, \ldots, N\}$, q, q' are query indices, $k, k' \in \{1, 2, \ldots, K\}$.

Proof of Bound 3: Step 1

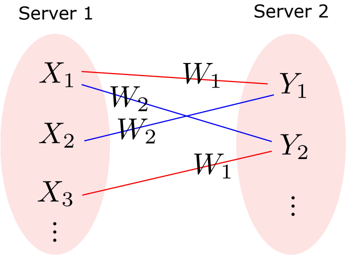
In a general PIR storage code:

- Storage contents may have different rates at different servers;
- Different answers may have different rates.

Symmetrize the code: through server-symmetry and variety-symmetry (Tian et al. IT-19)

- Storage rate: $H(S_n) = H(S_{n'})$ for any two servers $n, n' \in \{1, 2, \ldots, N\} \Rightarrow H(S_n) \leq \alpha$;
- Answer rate: $H(A_n^{(q)}) = H(A_{n'}^{(q')})$, q, q' are the query indices $\Rightarrow H(A_n^{(q)}) \leq \beta$;
- Joint entropy rate: $H(A_n^q, W_k) = H(A_{n'}^{(q')}, W_{k'})$, $n, n' \in \{1, 2, \ldots, N\}$, q, q' are query indices, $k, k' \in \{1, 2, \ldots, K\}$.

Proof of Bound 3: Step 2



A subtle dependence structure:

1. To retrieve W_1: server-1 answer $X_1 = A_1^{(q_1)}$ & server-2 answer $Y_1 = A_2^{(q_2)}$;
2. X_1 can also retrieve W_2 (due to privacy): together with $Y_2 = A_2^{(q_2')}$;
3. Y_1 can also retrieve W_2: together with $X_2 = A_1^{(q_1')}$;
4. Y_2 can also retrieve W_1: together with $X_3 = A_1^{(q_1'')}$.

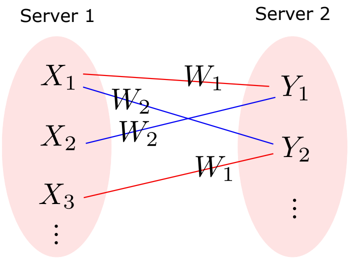
$$H(W_1|X_1, Y_1) = 0,\ H(W_2|X_1, Y_2) = 0,\ H(W_2|X_2, Y_1) = 0,\ H(W_1|X_3, Y_2) = 0.$$

Additional dependence:

- $(X_1, X_2, X_3, Y_1, Y_2)$ are deterministic functions of W_1, W_2;
- Answer encoding function: $H(X_1, X_2, X_3|S_1) = 0$ and $H(Y_1, Y_2|S_2) = 0$, implying

$$\alpha \geq H(S_1) \geq H(X_1, X_2, X_3), \quad \alpha \geq H(S_2) \geq H(Y_1, Y_2).$$

Proof of Bound 3: Step 2



A subtle dependence structure:

1. To retrieve W_1: server-1 answer $X_1 = A_1^{(q_1)}$ & server-2 answer $Y_1 = A_2^{(q_2)}$;
2. X_1 can also retrieve W_2 (due to privacy): together with $Y_2 = A_2^{(q_2')}$;
3. Y_1 can also retrieve W_2: together with $X_2 = A_1^{(q_1')}$;
4. Y_2 can also retrieve W_1: together with $X_3 = A_1^{(q_1'')}$.

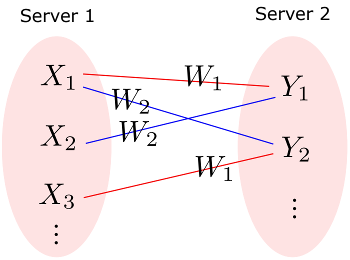
$$H(W_1|X_1, Y_1) = 0,\ H(W_2|X_1, Y_2) = 0,\ H(W_2|X_2, Y_1) = 0,\ H(W_1|X_3, Y_2) = 0.$$

Additional dependence:

- $(X_1, X_2, X_3, Y_1, Y_2)$ are deterministic functions of W_1, W_2;
- Answer encoding function: $H(X_1, X_2, X_3|S_1) = 0$ and $H(Y_1, Y_2|S_2) = 0$, implying

$$\alpha \geq H(S_1) \geq H(X_1, X_2, X_3), \quad \alpha \geq H(S_2) \geq H(Y_1, Y_2).$$

Proof of Bound 3: Step 2



A subtle dependence structure:

1. To retrieve W_1: server-1 answer $X_1 = A_1^{(q_1)}$ & server-2 answer $Y_1 = A_2^{(q_2)}$;
2. X_1 can also retrieve W_2 (due to privacy): together with $Y_2 = A_2^{(q_2')}$;
3. Y_1 can also retrieve W_2: together with $X_2 = A_1^{(q_1')}$;
4. Y_2 can also retrieve W_1: together with $X_3 = A_1^{(q_1'')}$.

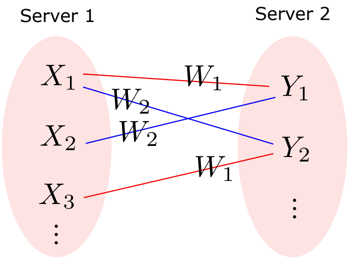
$$H(W_1|X_1, Y_1) = 0,\ H(W_2|X_1, Y_2) = 0,\ H(W_2|X_2, Y_1) = 0,\ H(W_1|X_3, Y_2) = 0.$$

Additional dependence:

- $(X_1, X_2, X_3, Y_1, Y_2)$ are deterministic functions of W_1, W_2;
- Answer encoding function: $H(X_1, X_2, X_3|S_1) = 0$ and $H(Y_1, Y_2|S_2) = 0$, implying

$$\alpha \geq H(S_1) \geq H(X_1, X_2, X_3), \quad \alpha \geq H(S_2) \geq H(Y_1, Y_2).$$

Proof of Bound 3: Step 2



A subtle dependence structure:

1. To retrieve W_1: server-1 answer $X_1 = A_1^{(q_1)}$ & server-2 answer $Y_1 = A_2^{(q_2)}$;
2. X_1 can also retrieve W_2 (due to privacy): together with $Y_2 = A_2^{(q_2')}$;
3. Y_1 can also retrieve W_2: together with $X_2 = A_1^{(q_1')}$;
4. Y_2 can also retrieve W_1: together with $X_3 = A_1^{(q_1'')}$.

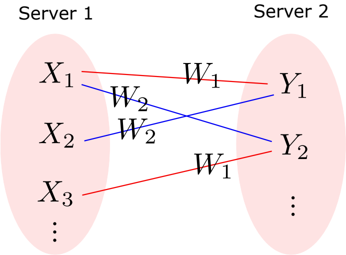
$$H(W_1|X_1,Y_1)=0,\ H(W_2|X_1,Y_2)=0,\ H(W_2|X_2,Y_1)=0,\ H(W_1|X_3,Y_2)=0.$$

Additional dependence:

- (X_1,X_2,X_3,Y_1,Y_2) are deterministic functions of W_1,W_2;
- Answer encoding function: $H(X_1,X_2,X_3|S_1)=0$ and $H(Y_1,Y_2|S_2)=0$, implying

$$\alpha \geq H(S_1) \geq H(X_1,X_2,X_3), \quad \alpha \geq H(S_2) \geq H(Y_1,Y_2).$$

Proof of Bound 3: Step 2



A subtle dependence structure:

1. To retrieve W_1: server-1 answer $X_1 = A_1^{(q_1)}$ & server-2 answer $Y_1 = A_2^{(q_2)}$;
2. X_1 can also retrieve W_2 (due to privacy): together with $Y_2 = A_2^{(q_2')}$;
3. Y_1 can also retrieve W_2: together with $X_2 = A_1^{(q_1')}$;
4. Y_2 can also retrieve W_1: together with $X_3 = A_1^{(q_1'')}$.

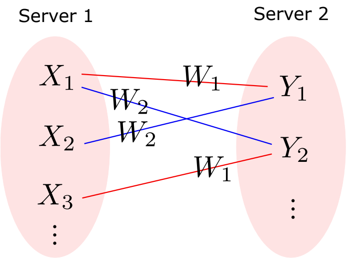
$$H(W_1|X_1, Y_1) = 0,\ H(W_2|X_1, Y_2) = 0,\ H(W_2|X_2, Y_1) = 0,\ H(W_1|X_3, Y_2) = 0.$$

Additional dependence:

- $(X_1, X_2, X_3, Y_1, Y_2)$ are deterministic functions of W_1, W_2;
- Answer encoding function: $H(X_1, X_2, X_3|S_1) = 0$ and $H(Y_1, Y_2|S_2) = 0$, implying

$$\alpha \geq H(S_1) \geq H(X_1, X_2, X_3), \quad \alpha \geq H(S_2) \geq H(Y_1, Y_2).$$

Proof of Bound 3: Step 2



A subtle dependence structure:

1. To retrieve W_1: server-1 answer $X_1 = A_1^{(q_1)}$ & server-2 answer $Y_1 = A_2^{(q_2)}$;
2. X_1 can also retrieve W_2 (due to privacy): together with $Y_2 = A_2^{(q_2')}$;
3. Y_1 can also retrieve W_2: together with $X_2 = A_1^{(q_1')}$;
4. Y_2 can also retrieve W_1: together with $X_3 = A_1^{(q_1'')}$.

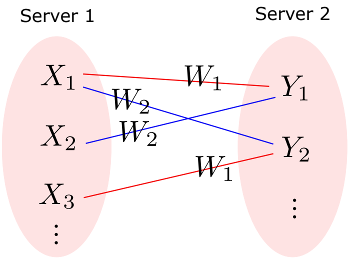
$$H(W_1|X_1, Y_1) = 0, \; H(W_2|X_1, Y_2) = 0, \; H(W_2|X_2, Y_1) = 0, \; H(W_1|X_3, Y_2) = 0.$$

Additional dependence:

- $(X_1, X_2, X_3, Y_1, Y_2)$ are deterministic functions of W_1, W_2;
- Answer encoding function: $H(X_1, X_2, X_3|S_1) = 0$ and $H(Y_1, Y_2|S_2) = 0$, implying

$$\alpha \geq H(S_1) \geq H(X_1, X_2, X_3), \quad \alpha \geq H(S_2) \geq H(Y_1, Y_2).$$

Proof of Bound 3: Step 2



A subtle dependence structure:

1. To retrieve W_1: server-1 answer $X_1 = A_1^{(q_1)}$ & server-2 answer $Y_1 = A_2^{(q_2)}$;
2. X_1 can also retrieve W_2 (due to privacy): together with $Y_2 = A_2^{(q_2')}$;
3. Y_1 can also retrieve W_2: together with $X_2 = A_1^{(q_1')}$;
4. Y_2 can also retrieve W_1: together with $X_3 = A_1^{(q_1'')}$.

$$H(W_1|X_1, Y_1) = 0,\ H(W_2|X_1, Y_2) = 0,\ H(W_2|X_2, Y_1) = 0,\ H(W_1|X_3, Y_2) = 0.$$

Additional dependence:

- $(X_1, X_2, X_3, Y_1, Y_2)$ are deterministic functions of W_1, W_2;
- Answer encoding function: $H(X_1, X_2, X_3|S_1) = 0$ and $H(Y_1, Y_2|S_2) = 0$, implying

$$\alpha \geq H(S_1) \geq H(X_1, X_2, X_3), \quad \alpha \geq H(S_2) \geq H(Y_1, Y_2).$$

Proof of Bound 3: Step 3

$$V_1, V_2\ \underline{Y_1, Y_2, W_1, W_2, X_1, X_2, X_3}\ U_1, U_2$$

Extend the probability space: the copy lemma technique

- Pseudo messages V_1, V_2: $(V_1, V_2) \leftrightarrow (Y_1, Y_2) \leftrightarrow (W_1, W_2, X_1, X_2, X_3)$

  Identical distribution: $(Y_1, Y_2, V_1, V_2) \sim (Y_1, Y_2, W_1, W_2)$.

- Pseudo messages (U_1, U_2): $(U_1, U_2) \leftrightarrow (X_1, X_2, X_3) \leftrightarrow (W_1, W_2, Y_1, Y_2, V_1, V_2)$

  Identical distribution: $(X_1, X_2, X_3, U_1, U_2) \sim (X_1, X_2, X_3, W_1, W_2)$.

In this extended probability space, terms can be canceled via the above distribution relation.

Proof of Bound 3: Step 3

$$V_1, V_2 \ \underline{Y_1, Y_2, W_1, W_2, X_1, X_2, X_3} \ U_1, U_2$$

Extend the probability space: the copy lemma technique

- Pseudo messages V_1, V_2: $(V_1, V_2) \leftrightarrow (Y_1, Y_2) \leftrightarrow (W_1, W_2, X_1, X_2, X_3)$

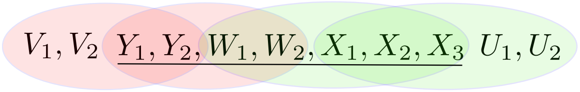
  Identical distribution: $(Y_1, Y_2, V_1, V_2) \sim (Y_1, Y_2, W_1, W_2)$.

- Pseudo messages (U_1, U_2): $(U_1, U_2) \leftrightarrow (X_1, X_2, X_3) \leftrightarrow (W_1, W_2, Y_1, Y_2, V_1, V_2)$

  Identical distribution: $(X_1, X_2, X_3, U_1, U_2) \sim (X_1, X_2, X_3, W_1, W_2)$.

In this extended probability space, terms can be canceled via the above distribution relation

Proof of Bound 3: Step 3



Extend the probability space: the copy lemma technique

- Pseudo messages V_1, V_2: $(V_1, V_2) \leftrightarrow (Y_1, Y_2) \leftrightarrow (W_1, W_2, X_1, X_2, X_3)$

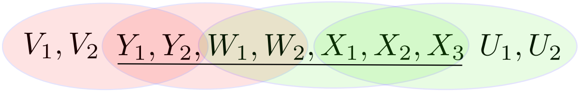
$$\text{Identical distribution: } (Y_1, Y_2, V_1, V_2) \sim (Y_1, Y_2, W_1, W_2).$$

- Pseudo messages (U_1, U_2): $(U_1, U_2) \leftrightarrow (X_1, X_2, X_3) \leftrightarrow (W_1, W_2, Y_1, Y_2, V_1, V_2)$

$$\text{Identical distribution: } (X_1, X_2, X_3, U_1, U_2) \sim (X_1, X_2, X_3, W_1, W_2).$$

In this extended probability space, terms can be canceled via the above distribution relatio

Proof of Bound 3: Step 3



Extend the probability space: the copy lemma technique

- Pseudo messages V_1, V_2: $(V_1, V_2) \leftrightarrow (Y_1, Y_2) \leftrightarrow (W_1, W_2, X_1, X_2, X_3)$

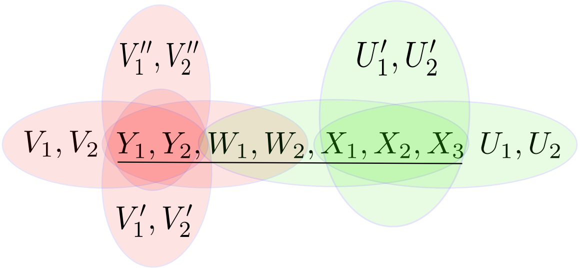
$$\text{Identical distribution: } (Y_1, Y_2, V_1, V_2) \sim (Y_1, Y_2, W_1, W_2).$$

- Pseudo messages (U_1, U_2): $(U_1, U_2) \leftrightarrow (X_1, X_2, X_3) \leftrightarrow (W_1, W_2, Y_1, Y_2, V_1, V_2)$

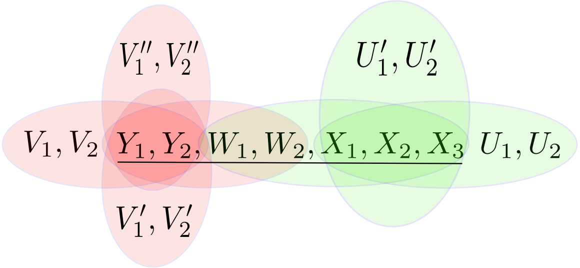
$$\text{Identical distribution: } (X_1, X_2, X_3, U_1, U_2) \sim (X_1, X_2, X_3, W_1, W_2).$$

In this extended probability space, terms can be canceled via the above distribution relation.

Hardness of Automatic Incorporating Non-Shannon



- We are expanding but not reducing the probability space.
- How do we know what to expand? Infinite many choices.
- The expansion given above for the PIR problem leads to slightly better bounds.
- Requires (automatic) exploration and machine learning of patterns.



- We are expanding but not reducing the probability space.
- How do we know what to expand? Infinite many choices.
- The expansion given above for the PIR problem leads to slightly better bounds.
- Requires (automatic) exploration and machine learning of patterns.

Outline

1. Fundamental Limits of Information Systems
2. Symmetry-Reduced Entropy LP
3. Beyond Bounds and Proofs
   - Reverse engineering optimal codes
   - Data-driven outer bound hypotheses
   - Computer-aided exploration
4. A New Software Toolbox (CAI)
5. Two New Directions
   - Utilizing non-Shannon-type inequalities
   - A new decomposition approach
6. Summary

The Reformulated Optimization Problem

The original optimization problem is

$$\min_{\mathrm{I\&II}} f_0$$

where type-I constraints are elemental inequalities, and type-II are problem specific ones.

Assuming the number of effective type-I inequalities is very small $\leq \kappa$, then

$$\min_{\mathrm{I\&II}} f_0 = \max_{\mathrm{I}_p \subseteq \mathrm{I}:|\mathrm{I}_p|=\kappa} \min_{\mathrm{I}_p\&\mathrm{II}} f_0.$$

We do have some empirical evidence: all the results given previously only required a small number of inequalities (e.g., 8 inequalities for the (4,3,3) code)

The Reformulated Optimization Problem

The original optimization problem is

$$\min_{\mathsf{I\&II}} f_0$$

where type-I constraints are elemental inequalities, and type-II are problem specific ones.

Assuming the number of effective type-I inequalities is very small $\leq \kappa$, then

$$\min_{\mathsf{I\&II}} f_0 = \max_{\mathsf{I}_p \subseteq \mathsf{I} : |\mathsf{I}_p| = \kappa} \min_{\mathsf{I}_p\mathsf{\&II}} f_0.$$

We do have some empirical evidence: all the results given previously only required a small number of inequalities (e.g., 8 inequalities for the (4,3,3) code)

The Reformulated Optimization Problem

The original optimization problem is

$$\min_{\mathsf{I}\&\mathsf{II}} f_0$$

where type-I constraints are elemental inequalities, and type-II are problem specific ones.

Assuming the number of effective type-I inequalities is very small $\leq \kappa$, then

$$\min_{\mathsf{I}\&\mathsf{II}} f_0 = \max_{\mathsf{I}_p \subseteq \mathsf{I}: |\mathsf{I}_p| = \kappa} \min_{\mathsf{I}_p \& \mathsf{II}} f_0.$$

We do have some empirical evidence: all the results given previously only required a small number of inequalities (e.g., 8 inequalities for the (4,3,3) code)

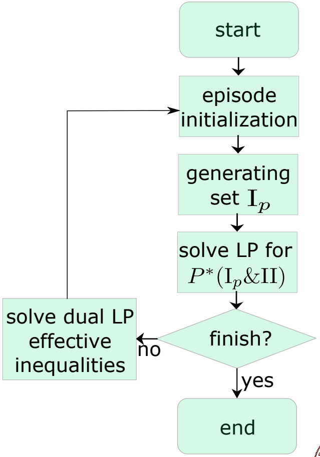
The New Approach

Idea: "Guess" on the effective inequalities

- With this conjectured set of effective inequalities, compute a bound;
- Many attempts can be made to find the best bound;
- The effective inequalities are likely important constraints: reuse them in future episodes.

Similar to how a human does it:

- Try to understand the problem and find the most constraining parts;
- Attempt to construct outer bounds and improve on it through some trial-and-error.



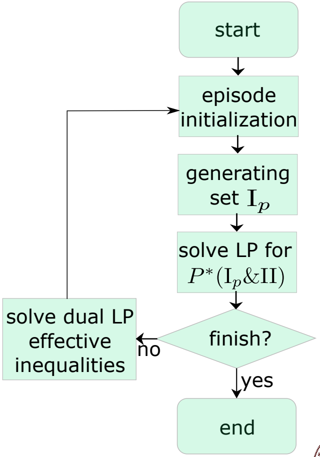
The New Approach

Idea: "Guess" on the effective inequalities

- With this conjectured set of effective inequalities, compute a bound;
- Many attempts can be made to find the best bound;
- The effective inequalities are likely important constraints: reuse them in future episodes.

Similar to how a human does it:

- Try to understand the problem and find the most constraining parts;
- Attempt to construct outer bounds and improve on it through some trial-and-error.



Incorporating Side Information

There are many “intuitions” that human researchers rely on

- Start from smaller instances and extend it to large instances;
- Use genie-aided arguments;
- Relaxed the problem constraints;
- Utilize potentially optimal codes.

Incorporating Side Information

There are many “intuitions” that human researchers rely on

- Start from smaller instances and extend it to large instances;
- Use genie-aided arguments;
- Relaxed the problem constraints;
- Utilize potentially optimal codes.

How to Utilize Potentially Optimal Codes

Recall our example where $min(3a + c) = 8$:

$$\begin{aligned}
a \quad & +b & & \geq 1 \\
& -4b & -c & \geq 2 \\
& +b & +2c & \geq 3 \\
a \quad & & +4c & \geq 0
\end{aligned}$$

- Suppose the physical meaning of the system leads us to an assignment $(a, b, c) = (2, -1, 2)$, which we suspect is optimal;
- $min(3a + c) = 8$ would imply that the first 3 inequalities are effective, and the last not;
- Indeed, the last inequality with $(a, b, c) = (2, -1, 2)$ here will hold with strict inequality, while the first three with equality.

$\Rightarrow$Exclude inequalities not equal to zero with the suspected optimal solution; more generally, select those with a mismatch gap less than a threshold for a given assignment.

How to Utilize Potentially Optimal Codes

Recall our example where $min(3a + c) = 8$:

$$
\begin{array}{llll}
a & +b & & \geq 1 \\
 & -4b & -c & \geq 2 \\
 & +b & +2c & \geq 3 \\
a & & +4c & \geq 0
\end{array}
$$

- Suppose the physical meaning of the system leads us to an assignment $(a, b, c) = (2, -1, 2)$, which we suspect is optimal;
- $min(3a + c) = 8$ would imply that the first 3 inequalities are effective, and the last not;
- Indeed, the last inequality with $(a, b, c) = (2, -1, 2)$ here will hold with strict inequality, while the first three with equality.

$\Rightarrow$Exclude inequalities not equal to zero with the suspected optimal solution; more general: select those with a mismatch gap less than a threshold for a given assignment.

How to Utilize Potentially Optimal Codes

Recall our example where $min(3a + c) = 8$:

$$\begin{array}{llll} a & +b & & \geq 1 \\ & -4b & -c & \geq 2 \\ & +b & +2c & \geq 3 \\ a & & +4c & \geq 0 \end{array}$$

- Suppose the physical meaning of the system leads us to an assignment $(a, b, c) = (2, -1, 2)$, which we suspect is optimal;
- $min(3a + c) = 8$ would imply that the first 3 inequalities are effective, and the last not;
- Indeed, the last inequality with $(a, b, c) = (2, -1, 2)$ here will hold with strict inequality, while the first three with equality.

$\Rightarrow$Exclude inequalities not equal to zero with the suspected optimal solution; more generally, select those with a mismatch gap less than a threshold for a given assignment.

How to Utilize Potentially Optimal Codes

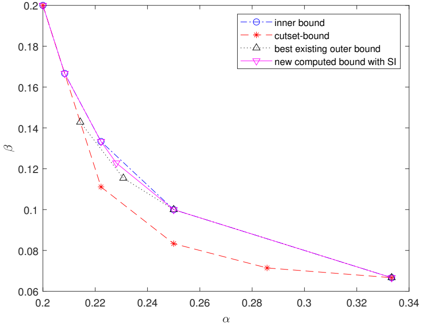
Recall our example where $min(3a + c) = 8$:

$$
\begin{array}{llll}
a & +b & & \geq 1 \\
 & -4b & -c & \geq 2 \\
 & +b & +2c & \geq 3 \\
a & & +4c & \geq 0
\end{array}
$$

- Suppose the physical meaning of the system leads us to an assignment $(a, b, c) = (2, -1, 2)$, which we suspect is optimal;
- $min(3a + c) = 8$ would imply that the first 3 inequalities are effective, and the last not;
- Indeed, the last inequality with $(a, b, c) = (2, -1, 2)$ here will hold with strict inequality, while the first three with equality.

$\Rightarrow$Exclude inequalities not equal to zero with the suspected optimal solution; more generally, select those with a mismatch gap less than a threshold for a given assignment.

Applying on the Regenerating Code Problem $(6, 5, 5)$



- The inner bound is due to the layered coding scheme in TIT-15: optimal for linear codes;
- The best outer bound was due to Mohajer and Tandon ISIT-15;
- Details in Chen & T. ISIT-22.

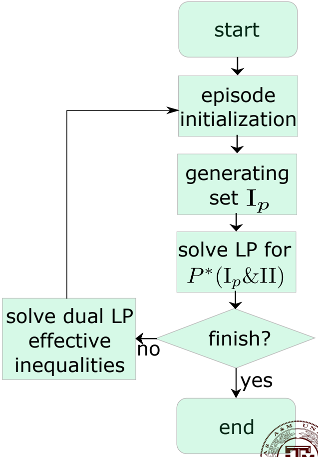
Pros and Cons

Pros:

- Each LP solves a small problem: both the number of variables and the number of constraints;
- Reduction-based approach: when the problem is large, no good way to even enumerate and start the reduction;
- Various intuition/side-information can be incorporated;
- Mimic human behaviors, potentially more efficient.

Cons:

- Only works when a small set of inequalities are effective;
- Hard to identify good combination of entropy terms and inequalities, especially at the beginning;
- Worse case complexity may even be worse than directly solving the entropy LP.



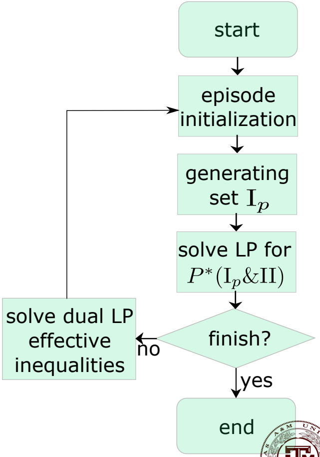
Pros and Cons

Pros:

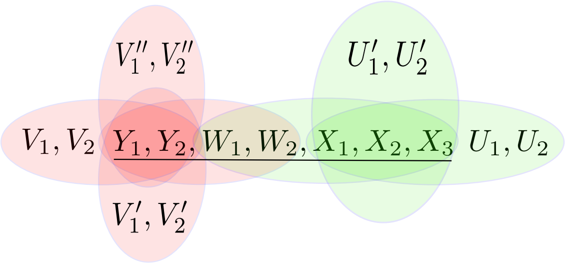
- Each LP solves a small problem: both the number of variables and the number of constraints;
- Reduction-based approach: when the problem is large, no good way to even enumerate and start the reduction;
- Various intuition/side-information can be incorporated;
- Mimic human behaviors, potentially more efficient.

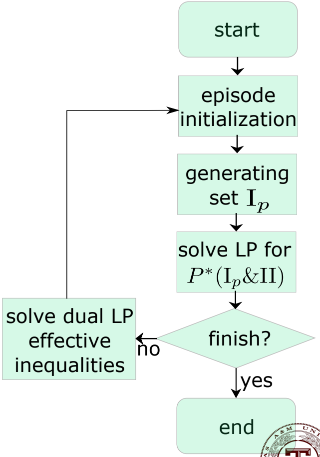
Cons:

- Only works when a small set of inequalities are effective;
- Hard to identify good combination of entropy terms and inequalities, especially at the beginning;
- Worse case complexity may even be worse than directly solving the entropy LP.



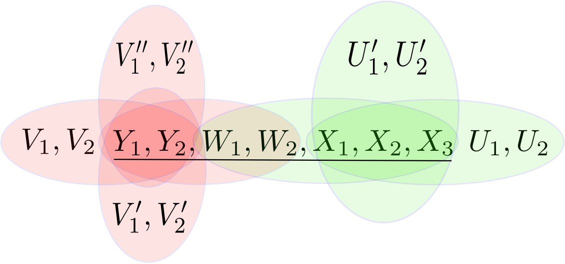
Prediction vs Generation

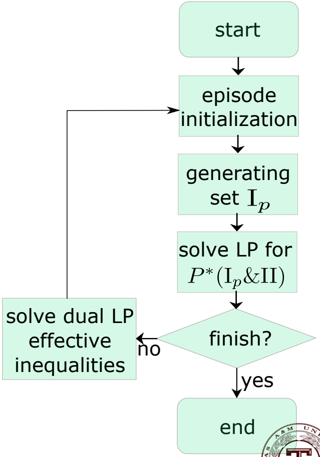




- We want to "predict/find" the best extended probability space or inequality combinations;
- Actually, perhaps not the right way to look at it: we only need to generate such patterns efficiently and improve the accuracy online.

Prediction vs Generation





- We want to "predict/find" the best extended probability space or inequality combinations;
- Actually, perhaps not the right way to look at it: we only need to generate such patterns efficiently and improve the accuracy online.

Outline

Summary

The conventional approach relies too much on human efforts

- Relieve us of tedious work by introducing more machine intelligence;
- A computational and data-driven approach;
- Application on real research problems proves its effectiveness.

- The two new directions both point to machine learning:
  - How to effectively and automatically select the probability space to extend?
  - How to intelligently select subsets of inequalities in the loop?

  Possible answer: reinforcement learning and generative models?

Disclaimer: It is an incomplete overview, as several related and recent efforts were not included.

Summary

The conventional approach relies too much on human efforts

- Relieve us of tedious work by introducing more machine intelligence;
- A computational and data-driven approach;
- Application on real research problems proves its effectiveness.

- The two new directions both point to machine learning:
  - How to effectively and automatically select the probability space to extend?
  - How to intelligently select subsets of inequalities in the loop?

  Possible answer: reinforcement learning and generative models?

Disclaimer: It is an incomplete overview, as several related and recent efforts were not included.

Summary

The conventional approach relies too much on human efforts

- Relieve us of tedious work by introducing more machine intelligence;
- A computational and data-driven approach;
- Application on real research problems proves its effectiveness.

- The two new directions both point to machine learning:
  - How to effectively and automatically select the probability space to extend?
  - How to intelligently select subsets of inequalities in the loop?

  Possible answer: reinforcement learning and generative models?

Disclaimer: It is an incomplete overview, as several related and recent efforts were not included.

Summary

The conventional approach relies too much on human efforts

- Relieve us of tedious work by introducing more machine intelligence;
- A computational and data-driven approach;
- Application on real research problems proves its effectiveness.

- The two new directions both point to machine learning:
  - How to effectively and automatically select the probability space to extend?
  - How to intelligently select subsets of inequalities in the loop?

  Possible answer: reinforcement learning and generative models?

Disclaimer: It is an incomplete overview, as several related and recent efforts were not included.

Summary

The conventional approach relies too much on human efforts

- Relieve us of tedious work by introducing more machine intelligence;
- A computational and data-driven approach;
- Application on real research problems proves its effectiveness.

- The two new directions both point to machine learning:
  - How to effectively and automatically select the probability space to extend?
  - How to intelligently select subsets of inequalities in the loop?

  Possible answer: reinforcement learning and generative models?

Disclaimer: It is an incomplete overview, as several related and recent efforts were not included.

Le Fin

Questions, please!