Coding Schemes for Input-Constrained Channels

V. Arvind Rameshwar

(guided by) Navin Kashyap

Indian Institute of Science, Bangalore

Networks Seminar 2023

Coding Schemes for Input-Constrained Channels

V. Arvind Rameshwar

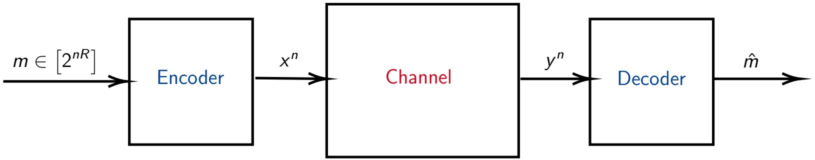
(guided by) Navin Kashyap

Indian Institute of Science, Bangalore

Networks Seminar 2023

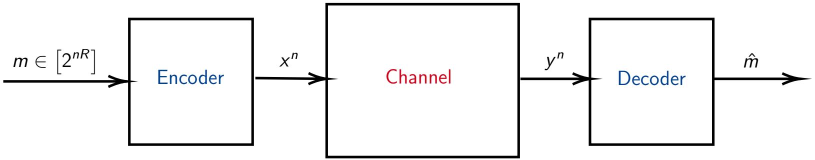
The big picture

We are broadly interested in the design of coding schemes that allow for reliable communication over noisy channels with memory.



The big picture

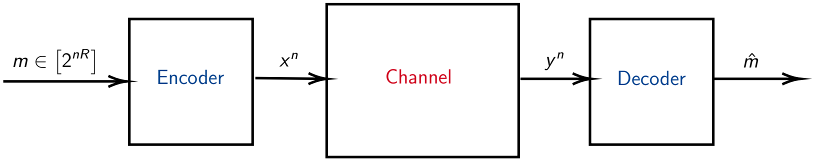
We are broadly interested in the design of coding schemes that allow for reliable communication over noisy channels with memory.



- The "memory" of the channel is encapsulated in a channel state s_i, at every time instant i, with the transitions between states (possibly) driven by the inputs.
- Examples: ISI channels, Gilbert-Elliott channels, input-constrained channels
- Our focus: Input-constrained channels

The big picture

We are broadly interested in the design of coding schemes that allow for reliable communication over noisy channels with memory.



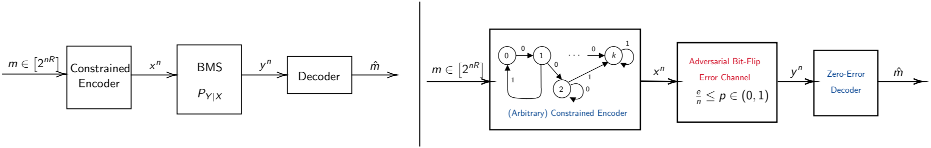
- The "memory" of the channel is encapsulated in a channel state s_i, at every time instant i, with the transitions between states (possibly) driven by the inputs.
- Examples: ISI channels, Gilbert-Elliott channels, input-constrained channels
- Our focus: Input-constrained channels

Broad question: Can we design reliable coding schemes, with large rate $R \in (0, 1)$, over such channels?

Channel models and constraints: an overview

We consider the setting of the transmission of binary constrained codes over noisy channels.

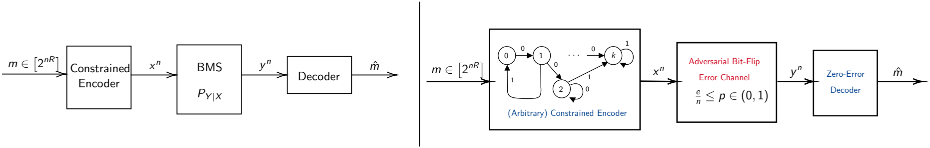
- We consider both stochastic and combinatorial (worst-case) noise models.



Channel models and constraints: an overview

We consider the setting of the transmission of binary constrained codes over noisy channels.

- We consider both stochastic and combinatorial (worst-case) noise models.

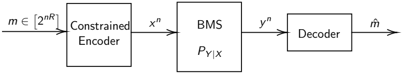


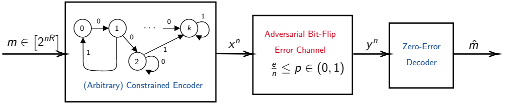
- The constrained sequences that we consider find application in a number of domains:

Channel models and constraints: an overview

We consider the setting of the transmission of binary constrained codes over noisy channels.

- We consider both stochastic and combinatorial (worst-case) noise models.



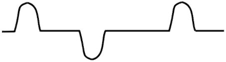


- The constrained sequences that we consider find application in a number of domains:

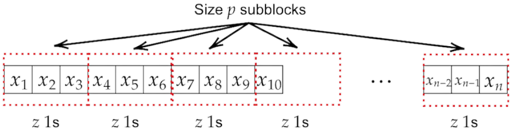
Some constraints of interest

- Runlength-limited (RLL) constraints: Help alleviate ISI in magneto-optical recording

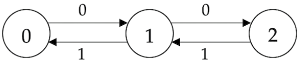
. . . 0 1 0 0 0 1 0 0 0 0 0 1 0 0 . . . $\longleftrightarrow$

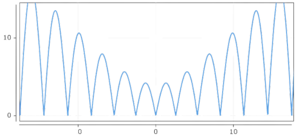


- Subblock composition constraints: Maintain receiver battery levels in energy-harvesting communication



- Charge constraints: Ensure spectral nulls (DC-freeness) in frequency spectrum





Talk outline

Part 1 Coding schemes over input-constrained symmetric channels via linear codes

A Explicit coding schemes over rulength-limited channels can be designed using Reed-Muller (RM) codes!

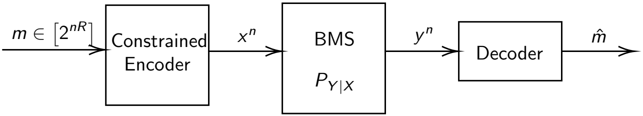
B A simple Fourier-analytic identity can help compute rates of arbitrarily constrained subcodes!

Part 2 Bounds on the resilience of constrained codes to worst-case (combinatorial) symmetric errors

- Delsarte's linear program (LP) can be extended to yield good bounds!

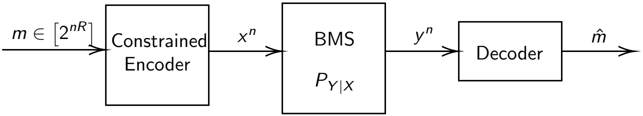
Part 1: Coding schemes over input-constrained symmetric channels

The channel model



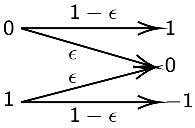
We first focus on (stochastic) input-constrained Binary-Input Memoryless Symmetric (BMS) channels:

The channel model

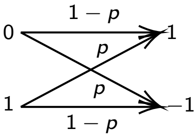


We first focus on (stochastic) input-constrained Binary-Input Memoryless Symmetric (BMS) channels:

Examples:



Binary Erasure Channel (BEC)



Binary Symmetric Channel (BSC)

Main idea and objectives

- Key Idea: Use explicit codes that achieve capacity over unconstrained BMS channels and select subcodes that comply with the input constraint.

- We know from [Reeves and Pfister (2022), Arikan (2009), Richardson and Urbanke (2001)] that there exist *linear codes* that achieve capacity over any BMS channel, under suitable decoding procedures.
  - Hence, *constrained subcodes* of such linear codes also enjoy vanishing error probabilities, under bit-MAP decoding.

- Goals:
  - Design explicit constrained coding schemes, using capacity-achieving linear codes, for select constraints.
  - Obtain estimates of the sizes of the largest constrained subcodes of general linear codes.

Part 1A: Constrained coding schemes using RM codes

The input constraint

The input constraint of interest to us is the (d, ∞)-runlength limited (RLL) input-constraint:

Definition

A binary sequence is said to satisfy the (d, ∞)-RLL constraint if there exist at least d 0s between every pair of successive 1s.

The input constraint

The input constraint of interest to us is the (d, ∞)-runlength limited (RLL) input-constraint:

Definition

A binary sequence is said to satisfy the (d, ∞)-RLL constraint if there exist at least d 0s between every pair of successive 1s.

- Example: For the $(2, \infty)$-RLL constraint,

$$1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1 \quad \checkmark$$

The input constraint

The input constraint of interest to us is the (d, ∞)-runlength limited (RLL) input-constraint:

Definition

A binary sequence is said to satisfy the (d, ∞)-RLL constraint if there exist at least d 0s between every pair of successive 1s.

- Example: For the $(2, \infty)$-RLL constraint,

$$1\;0\;0\;0\;1\;0\;0\;0\;0\;1\;0\;0\;1 \quad \checkmark$$

$$1\;0\;0\;1\;0\;1\;0\;0\;0\;1\;0\;0\;1 \quad \text{X}$$

The input constraint

The input constraint of interest to us is the (d, ∞)-runlength limited (RLL) input-constraint:

Definition

A binary sequence is said to satisfy the (d, ∞)-RLL constraint if there exist at least d 0s between every pair of successive 1s.

- Example: For the $(2, \infty)$-RLL constraint,

$$1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1 \qquad \checkmark$$

$$1\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1 \qquad \text{X}$$

- $(1, \infty)$-RLL $\equiv$ no-consecutive-ones.
- The (d, ∞)-RLL constraint on data sequences ensures that successive voltage peaks (or 1 bits) are spaced far apart, in order to alleviate ISI in magnetic recording systems.

Select prior art

Coding-theoretic work:

- Patapoutian and Kumar (1992): Coset-averaging lower bound on rates of constrained subcodes of cosets of linear codes.

Select prior art

Coding-theoretic work:

- Patapoutian and Kumar (1992): Coset-averaging lower bound on rates of constrained subcodes of cosets of linear codes.

Information-theoretic work:

- Zehavi and Wolf (1988), Shamai and Kofman (1990): Achievable rates over a BSC with runlength constraints at the input, using (random) Markov inputs
- Arnold et al. (2006), Han (2013, 2015), Li and Han (2018): Numerically computable achievable rates over general finite-state channels (FSCs), using Markov inputs
- Thangaraj (2017), Huleihel, Sabag, Permuter, Kashyap (2019): Upper bounds on the capacities of FSCs using the "dual capacity" method

Select prior art

Coding-theoretic work:

- Patapoutian and Kumar (1992): Coset-averaging lower bound on rates of constrained subcodes of cosets of linear codes.

Information-theoretic work:

- Zehavi and Wolf (1988), Shamai and Kofman (1990): Achievable rates over a BSC with runlength constraints at the input, using (random) Markov inputs
- Arnold et al. (2006), Han (2013, 2015), Li and Han (2018): Numerically computable achievable rates over general finite-state channels (FSCs), using Markov inputs
- Thangaraj (2017), Huleihel, Sabag, Permuter, Kashyap (2019): Upper bounds on the capacities of FSCs using the "dual capacity" method

In this part: Constrained code constructions using RM codes and explicit bounds on rates

Brief Background on RM Codes

- Codewords of RM codes consist of evaluation vectors of multivariate polynomials over $\mathbb{F}_2$.
- For a polynomial $f \in \mathbb{F}_2[x_1, x_2, \ldots, x_m]$ and a binary vector $\boldsymbol{z} = (z_1, \ldots, z_m)$, let $\text{Eval}_{\boldsymbol{z}}(f) := f(z_1, \ldots, z_m)$.
- Let the evaluation points $\boldsymbol{z}$ be ordered according to the standard lexicographic ordering:

$$000\ldots00 \to 000\ldots01 \to 000\ldots10 \to \ldots \to 111\ldots11$$

- We denote by $\text{Eval}(f) := (\text{Eval}_{\boldsymbol{z}}(f))_{\boldsymbol{z} \in \mathbb{F}_2^m}$, the 2^m-length vector of evaluations of f at points ordered in the lexicographic order

Brief Background on RM Codes

- Codewords of RM codes consist of evaluation vectors of multivariate polynomials over $\mathbb{F}_2$.
- For a polynomial $f \in \mathbb{F}_2[x_1, x_2, \ldots, x_m]$ and a binary vector $\boldsymbol{z} = (z_1, \ldots, z_m)$, let $\text{Eval}_{\boldsymbol{z}}(f) := f(z_1, \ldots, z_m)$.
- Let the evaluation points $\boldsymbol{z}$ be ordered according to the standard lexicographic ordering:

$$000\ldots00 \rightarrow 000\ldots01 \rightarrow 000\ldots10 \rightarrow \ldots \rightarrow 111\ldots11$$

- We denote by $\text{Eval}(f) := (\text{Eval}_{\boldsymbol{z}}(f))_{\boldsymbol{z} \in \mathbb{F}_2^m}$, the 2^m-length vector of evaluations of f at points ordered in the lexicographic order

Definition

The r^{th} order binary RM code $\text{RM}(m, r)$ is defined as the set of binary vectors:

$$\text{RM}(m, r) := \{\text{Eval}(f) : f \in \mathbb{F}_2[x_1, x_2, \ldots, x_m],\ \deg(f) \leq r\},$$

where $\deg(f)$ is the degree of the largest monomial in f and the degree of a monomial $x_S := \prod_{j \in S : S \subseteq [m]} x_j$ is simply $|S|$.

Brief Background on RM Codes

- Dimension and rate:

$$\dim(\mathrm{RM}(m,r)) = \#\{x_S \in \mathbb{F}_2[x_1,\ldots,x_m] : \ \deg(x_S) = |S| \le r\}$$
$$= \sum_{i=0}^{r} \binom{m}{i} =: \binom{m}{\le r}.$$

Hence, $\mathrm{rate}(\mathrm{RM}(m,r)) = \frac{\binom{m}{\le r}}{2^m}$.

Brief Background on RM Codes

- Dimension and rate:

$$\dim(\mathrm{RM}(m,r)) = \#\{x_S \in \mathbb{F}_2[x_1,\dots,x_m] : \deg(x_S) = |S| \le r\}$$
$$= \sum_{i=0}^{r} \binom{m}{i} =: \binom{m}{\le r}.$$

Hence, $\mathrm{rate}(\mathrm{RM}(m,r)) = \frac{\binom{m}{\le r}}{2^m}$.

- Example: For $R \in (0,1)$, consider the sequence of codes $\{\mathcal{C}_m(R) = \mathrm{RM}(m, r_m)\}_{m\ge 1}$, with

$$r_m := \max\left\{ \left\lfloor \frac{m}{2} + \frac{\sqrt{m}}{2} Q^{-1}(1-R) \right\rfloor, 0 \right\},$$

where

$$Q(t) = \frac{1}{\sqrt{2\pi}} \int_t^\infty e^{-\tau^2/2} d\tau, \ t \in \mathbb{R}.$$

- It can be checked that $\mathrm{rate}(\mathcal{C}_m) \xrightarrow{m\to\infty} R$.

Selected results

A simple construction using linear subcodes:

Theorem

For any $R \in [0, C)$, there exists a sequence of (d, ∞)-RLL linear subcodes $\{\mathcal{C}_m^{(d,\infty)}(R)\}$, where, for $t = \lceil \log_2(d+1) \rceil$,

$$\mathcal{C}_m^{(d,\infty)}(R) = \left\{ Eval(f) : f = \left(\prod_{i=m-t+1}^{m} x_i \right) \cdot h(x_1, \ldots, x_{m-t}), \; deg(h) \leq r_m - t \right\}.$$

Moreover, $rate\left(\mathcal{C}_m^{(d,\infty)}(R)\right) \xrightarrow{m \to \infty} R \cdot 2^{-\lceil \log_2(d+1) \rceil}$.

Selected results

A simple construction using linear subcodes:

Theorem

For any $R \in [0, C)$, there exists a sequence of (d,∞)-RLL linear subcodes $\{\mathcal{C}_m^{(d,\infty)}(R)\}$, where, for $t = \lceil \log_2(d+1) \rceil$,

$$\mathcal{C}_m^{(d,\infty)}(R) = \left\{ Eval(f) : f = \left(\prod_{i=m-t+1}^{m} x_i \right) \cdot h(x_1, \ldots, x_{m-t}), \; deg(h) \leq r_m - t \right\}.$$

Moreover, $rate\left(\mathcal{C}_m^{(d,\infty)}(R)\right) \xrightarrow{m \to \infty} R \cdot 2^{-\lceil \log_2(d+1) \rceil}$.

For example, when $d = 1$, these subcodes have all the symbols in odd coordinate positions as 0.

Selected results

Converse results for linear subcodes:

Theorem

For any sequence $\left\{\hat{\mathcal{C}}_m(R)\right\}_{m\geq 1}$ *of RM codes, following the lexicographic coordinate ordering, such that* $\text{rate}\left(\hat{\mathcal{C}}_m(R)\right) \xrightarrow{m\to\infty} R \in (0,1)$*, the largest rate of linear* (d,∞)*-RLL constrained subcodes is* $\frac{R}{d+1}$*.*

Selected results

Converse results for linear subcodes:

Theorem

For any sequence $\left\{\hat{\mathcal{C}}_m(R)\right\}_{m\geq 1}$ *of RM codes, following the lexicographic coordinate ordering, such that* $\text{rate}\left(\hat{\mathcal{C}}_m(R)\right) \xrightarrow{m\to\infty} R \in (0,1)$*, the largest rate of linear* (d,∞)*-RLL constrained subcodes is* $\frac{R}{d+1}$*.*

We had previously identified (d,∞)-RLL linear subcodes $\left\{\mathcal{C}_m^{(d,\infty)}(R)\right\}_{m\geq 1}$ of rate $R \cdot 2^{-\lceil \log_2(d+1) \rceil} \approx \frac{R}{d+1}$.

Hence, the subcodes $\left\{\mathcal{C}_m^{(d,\infty)}(R)\right\}_{m\geq 1}$ are essentially rate-optimal.

Selected results

Converse results for linear subcodes (contd.):

Theorem

Under almost all coordinate orderings, the largest rate of linear (d, ∞)-RLL subcodes of RM codes of rate R, is at most $\frac{R}{d+1} + \epsilon$, for $\epsilon > 0$ arbitrarily small.

Existence results for non-linear subcodes:

Theorem

For any sequence of RM codes $\left\{\hat{\mathcal{C}}_m(R)\right\}_{m\geq 1}$, under the lexicographic coordinate ordering, such that $\text{rate}\left(\hat{\mathcal{C}}_m(R)\right) \xrightarrow{m\to\infty} R \in (0,1)$, there exist $(1,\infty)$-RLL subcodes $\tilde{\mathcal{C}}_m^{(1,\infty)}(R) \subseteq \hat{\mathcal{C}}_m(R)$, of rate at least $\max\left(0, R - \frac{3}{8}\right)$.

Selected results

Converse results for linear subcodes (contd.):

Theorem

Under almost all coordinate orderings, the largest rate of linear (d,∞)-RLL subcodes of RM codes of rate R, is at most $\frac{R}{d+1}+\epsilon$, for $\epsilon > 0$ arbitrarily small.

Existence results for non-linear subcodes:

Theorem

For any sequence of RM codes $\left\{\hat{\mathcal{C}}_m(R)\right\}_{m\geq 1}$, under the lexicographic coordinate ordering, such that $\text{rate}\left(\hat{\mathcal{C}}_m(R)\right) \xrightarrow{m\to\infty} R \in (0,1)$, there exist $(1,\infty)$-RLL subcodes $\tilde{\mathcal{C}}_m^{(1,\infty)}(R) \subseteq \hat{\mathcal{C}}_m(R)$, of rate at least $\max\left(0, R-\frac{3}{8}\right)$.

These subcodes are necessarily non-linear for $R > 0.75$, since then $R - \frac{3}{8} > \frac{R}{2}$.

Plots and Comparisons - I

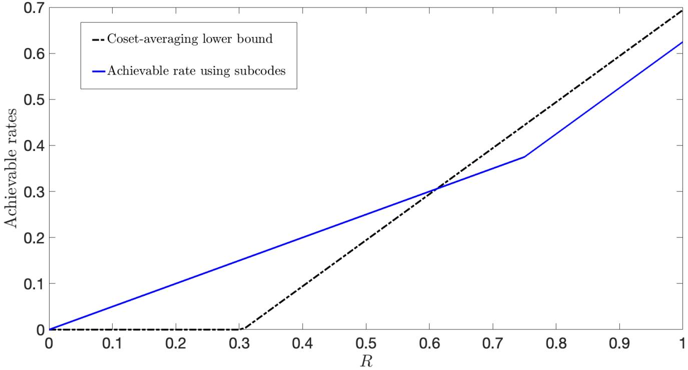


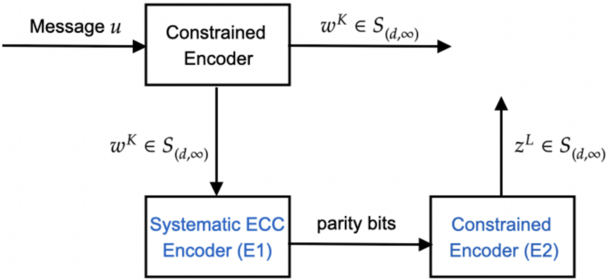
Figure: Plot comparing the achievable rates using $(1, \infty)$-RLL RM subcodes with the coset-averaging lower bound that is approximately $R - 0.3058$, of [Patapoutian and Kumar (1992)]

A concatenated coding scheme

We adopt the "reverse concatenation" strategy of [Bliss (1981)] and [Mansuripur (1991)] that is commonly used to limit error propagation during decoding of constrained codes.

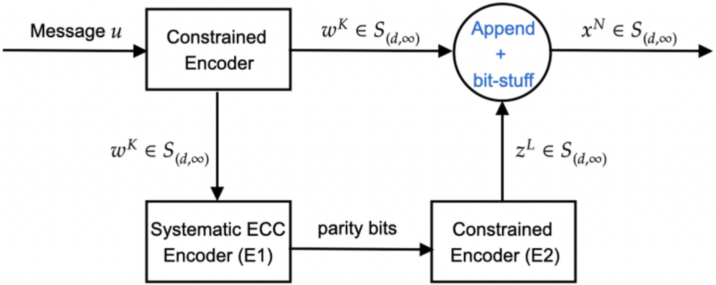
A concatenated coding scheme

Encoding (the Bliss scheme):



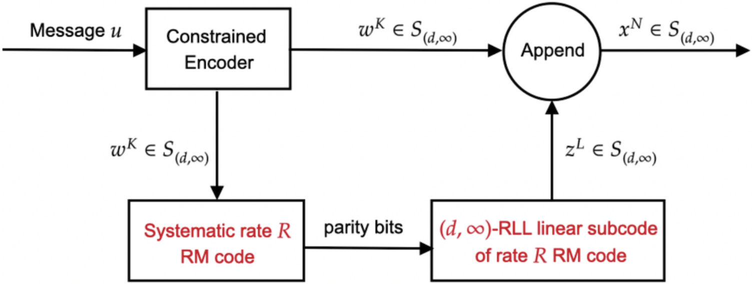
A concatenated coding scheme

Encoding (the Bliss scheme):



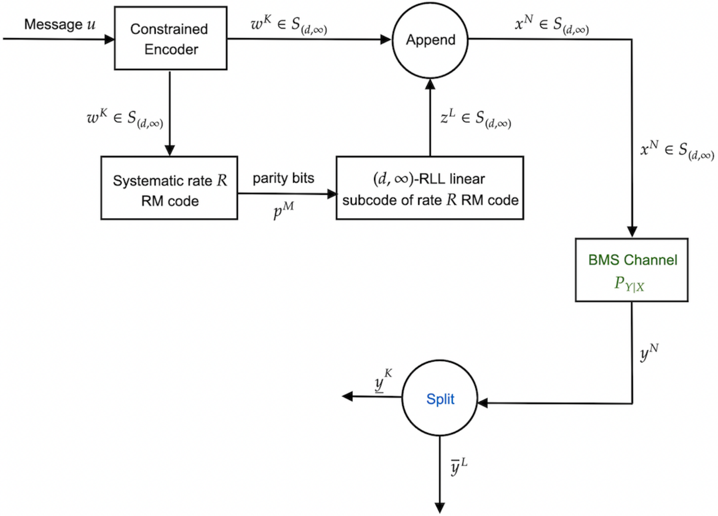
A concatenated coding scheme

Encoding:



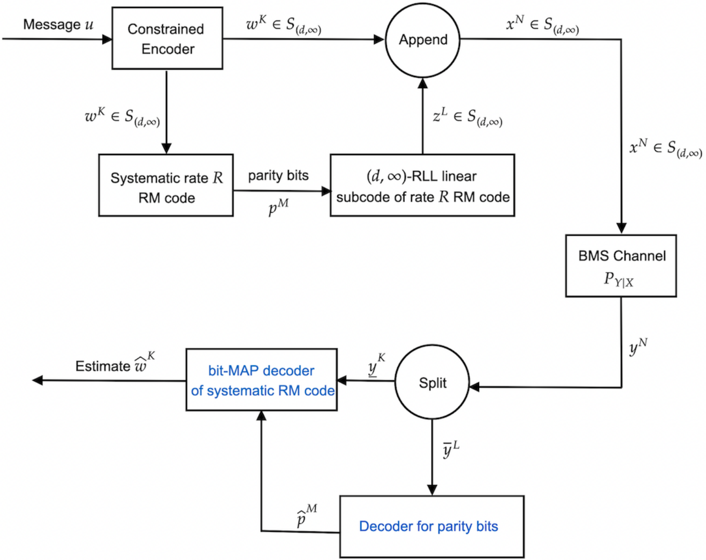
A concatenated coding scheme

Encoding+Decoding:



A concatenated coding scheme

Encoding+Decoding:



Coding theorem

Let C be the capacity of the unconstrained BMS channel.

Slight modifications to the previous concatenated coding scheme yield the following coding theorem:

Coding theorem

Let C be the capacity of the unconstrained BMS channel.

Slight modifications to the previous concatenated coding scheme yield the following coding theorem:

Theorem

For any $R \in (0, C)$, there exists a sequence of (d, ∞)-RLL constrained concatenated codes $\{\mathcal{C}_m^{conc}\}_{m \geq 1}$ that achieves a rate lower bound given by

$$\liminf_{m \to \infty} rate(\mathcal{C}_m^{conc}) \geq \frac{C_0^{(d)} \cdot R^2 \cdot 2^{-\lceil \log_2(d+1) \rceil}}{R^2 \cdot 2^{-\lceil \log_2(d+1) \rceil} + 1 - R + \epsilon},$$

over (d, ∞)-RLL input-constrained BMS channels, where $\epsilon > 0$ can be arbitrarily small.

Plots and Comparisons - II

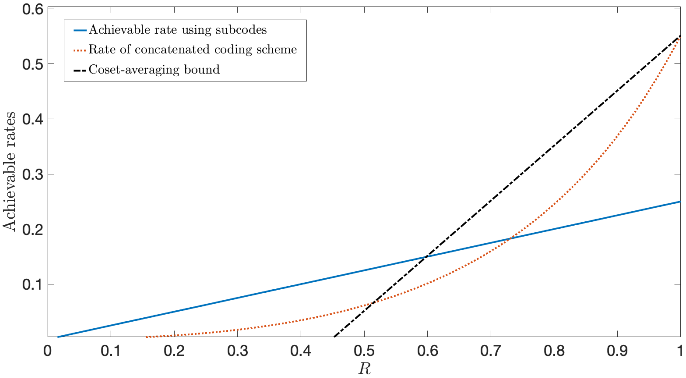


Figure: Plot comparing the achievable rates using $(2, \infty)$-RLL linear RM subcodes with the coset-averaging lower bound and the rate achieved by the concatenated coding scheme

Some questions

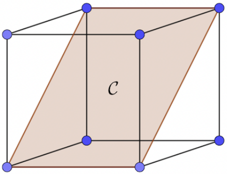
- Can we obtain better estimates of the weight distributions of RM codes, which will help sharpen our bounds?

- Can we extend the techniques in our work to design good codes over other finite-state channels such as Gilbert-Elliott channels (GECs) or ISI channels?

Part 1B: Counting constrained codewords in linear codes

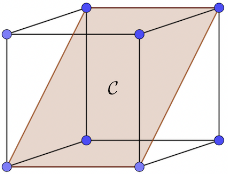
The problem

- Motivated by the previous section, we now consider the problem of computing rates of (arbitrary) constrained codewords in general linear codes $\mathcal{C}$.



The problem

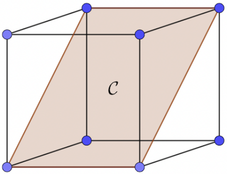
- Motivated by the previous section, we now consider the problem of computing rates of (arbitrary) constrained codewords in general linear codes $\mathcal{C}$.



- Such an approach can help identify linear codes with large constrained subcodes.

The problem

- Motivated by the previous section, we now consider the problem of computing rates of (arbitrary) constrained codewords in general linear codes $\mathcal{C}$.

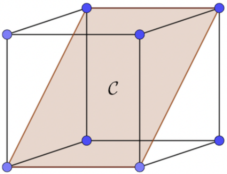


- Such an approach can help identify linear codes with large constrained subcodes.
- The problem: Given a set of constrained codewords $\mathcal{A} \subseteq \mathbb{F}_2^n$, we would like to gain insight into

$$N(\mathcal{C};\mathcal{A}) = \sum_{\boldsymbol{x}\in\mathcal{C}} \mathbb{1}\{\boldsymbol{x} \in \mathcal{A}\} = \sum_{\boldsymbol{x}\in\{0,1\}^n} \mathbb{1}\{\boldsymbol{x} \in \mathcal{A}\} \cdot \mathbb{1}\{\boldsymbol{x} \in \mathcal{C}\}.$$

The problem

- Motivated by the previous section, we now consider the problem of computing rates of (arbitrary) constrained codewords in general linear codes $\mathcal{C}$.



- Such an approach can help identify linear codes with large constrained subcodes.
- The problem: Given a set of constrained codewords $\mathcal{A} \subseteq \mathbb{F}_2^n$, we would like to gain insight into

$$N(\mathcal{C}; \mathcal{A}) = \sum_{\boldsymbol{x} \in \mathcal{C}} \mathbb{1}\{\boldsymbol{x} \in \mathcal{A}\} = \sum_{\boldsymbol{x} \in \{0,1\}^n} \mathbb{1}\{\boldsymbol{x} \in \mathcal{A}\} \cdot \mathbb{1}\{\boldsymbol{x} \in \mathcal{C}\}.$$

This looks like an inner product between Boolean functions. . .

A brief refresher on Fourier analysis on $\mathbb{F}_2^n$

- Given any function $f : \{0,1\}^n \to \mathbb{R}$ and a vector $\boldsymbol{s} = (s_1, \ldots, s_n) \in \{0,1\}^n$, the Fourier coefficient of f at $\boldsymbol{s}$ is

$$\widehat{f}(\boldsymbol{s}) := \frac{1}{2^n} \sum_{\boldsymbol{x} \in \{0,1\}^n} f(\boldsymbol{x}) \cdot (-1)^{\boldsymbol{x} \cdot \boldsymbol{s}}.$$

- The functions $(\chi_{\boldsymbol{s}} : \boldsymbol{s} \in \{0,1\}^n)$, where $\chi_{\boldsymbol{s}}(\boldsymbol{x}) := (-1)^{\boldsymbol{x} \cdot \boldsymbol{s}}$, form a basis for the vector space V of functions $f : \{0,1\}^n \to \mathbb{R}$. Further, they are orthonormal with respect to the inner product $\langle \cdot, \cdot \rangle$, where:

$$\langle f, g \rangle := \frac{1}{2^n} \sum_{\boldsymbol{x} \in \{0,1\}^n} f(\boldsymbol{x}) g(\boldsymbol{x}).$$

Theorem (Plancherel's Theorem)

For any $f, g \in \{0,1\}^n \to \mathbb{R}$, we have that

$$\langle f, g \rangle = \sum_{\boldsymbol{s} \in \{0,1\}^n} \widehat{f}(\boldsymbol{s}) \widehat{g}(\boldsymbol{s}).$$

Workhorse

Observe that

$$N(\mathcal{C};\mathcal{A}) = \sum_{\boldsymbol{x}\in\{0,1\}^n} \mathbb{1}\{\boldsymbol{x}\in\mathcal{A}\}\cdot\mathbb{1}\{\boldsymbol{x}\in\mathcal{C}\}$$

$$= 2^n \cdot \sum_{\boldsymbol{s}\in\{0,1\}^n} \widehat{\mathbb{1}_{\mathcal{A}}}(\boldsymbol{s})\cdot\widehat{\mathbb{1}_{\mathcal{C}}}(\boldsymbol{s}).$$

Workhorse

Observe that

$$N(\mathcal{C};\mathcal{A}) = \sum_{\boldsymbol{x}\in\{0,1\}^n} \mathbb{1}\{\boldsymbol{x}\in\mathcal{A}\}\cdot\mathbb{1}\{\boldsymbol{x}\in\mathcal{C}\}$$
$$= 2^n \cdot \sum_{\boldsymbol{s}\in\{0,1\}^n} \widehat{\mathbb{1}_{\mathcal{A}}}(\boldsymbol{s})\cdot\widehat{\mathbb{1}_{\mathcal{C}}}(\boldsymbol{s}).$$

For linear codes $\mathcal{C}$, it is easy to show that

$$\widehat{\mathbb{1}_{\mathcal{C}}}(\boldsymbol{s}) = \frac{|\mathcal{C}|}{2^n}\cdot\mathbb{1}_{\mathcal{C}^\perp}(\boldsymbol{s}).$$

Hence, we have that

$$N(\mathcal{C};\mathcal{A}) = |\mathcal{C}|\cdot\sum_{\boldsymbol{s}\in\mathcal{C}^\perp}\widehat{\mathbb{1}_{\mathcal{A}}}(\boldsymbol{s}).$$

Workhorse

Observe that

$$N(\mathcal{C};\mathcal{A}) = \sum_{\boldsymbol{x}\in\{0,1\}^n} \mathbb{1}\{\boldsymbol{x}\in\mathcal{A}\}\cdot\mathbb{1}\{\boldsymbol{x}\in\mathcal{C}\}$$
$$= 2^n \cdot \sum_{\boldsymbol{s}\in\{0,1\}^n} \widehat{\mathbb{1}_{\mathcal{A}}}(\boldsymbol{s})\cdot\widehat{\mathbb{1}_{\mathcal{C}}}(\boldsymbol{s}).$$

For linear codes $\mathcal{C}$, it is easy to show that

$$\widehat{\mathbb{1}_{\mathcal{C}}}(\boldsymbol{s}) = \frac{|\mathcal{C}|}{2^n}\cdot\mathbb{1}_{\mathcal{C}^\perp}(\boldsymbol{s}).$$

Hence, we have that

$$N(\mathcal{C};\mathcal{A}) = |\mathcal{C}|\cdot\sum_{\boldsymbol{s}\in\mathcal{C}^\perp}\widehat{\mathbb{1}_{\mathcal{A}}}(\boldsymbol{s}).$$

Observations:

1. If $\dim(\mathcal{C}) \gg n/2$, then we can employ our insight to count over a low-dimensional space!
2. For many constraints of interest, the Fourier transform above is computable!

Example 0: warm-up

- Consider the constant weight constraint that admits only binary sequences of a fixed weight $i \in [0 : n]$ (we let W_i denote the set of such words). Let us write $a_i(\mathcal{C}) := N(\mathcal{C}; W_i)$.
- Applying our method, we get that

Example 0: warm-up

- Consider the constant weight constraint that admits only binary sequences of a fixed weight $i \in [0:n]$ (we let W_i denote the set of such words). Let us write $a_i(\mathcal{C}) := N(\mathcal{C}; W_i)$.

- Applying our method, we get that

$$a_i(\mathcal{C}) = \frac{1}{|\mathcal{C}^\perp|} \sum_{j=0}^{n} K_i^{(n)}(j) \cdot a_j(\mathcal{C}^\perp),$$

where $K_i^{(n)}(x)$ is the i^{th}-Krawtchouk polynomial at length n, with $K_i^{(n)}(z) = \sum_{\ell=0}^{i} (-1)^\ell \binom{z}{\ell} \binom{n-z}{i-\ell}$.

Example 0: warm-up

- Consider the constant weight constraint that admits only binary sequences of a fixed weight $i \in [0:n]$ (we let W_i denote the set of such words). Let us write $a_i(\mathcal{C}) := N(\mathcal{C}; W_i)$.

- Applying our method, we get that

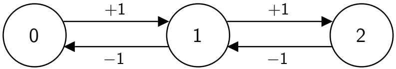
$$a_i(\mathcal{C}) = \frac{1}{|\mathcal{C}^\perp|} \sum_{j=0}^{n} K_i^{(n)}(j) \cdot a_j(\mathcal{C}^\perp),$$

where $K_i^{(n)}(x)$ is the i^{th}-Krawtchouk polynomial at length n, with $K_i^{(n)}(z) = \sum_{\ell=0}^{i} (-1)^\ell \binom{z}{\ell}\binom{n-z}{i-\ell}$.

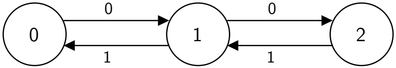
These are simply MacWilliams' identities for linear codes.

Example 1: 2-charge constraint

- We now consider a spectral null constraint, whose sequences in $\{+1,-1\}^n$ have a null at zero frequency (sometimes called a DC-free constraint). Our constraint is the so-called 2-charge constraint.



- We let S_2 denote those sequences in $\{0,1\}^n$ that can be mapped to 2-charge constrained sequences via the map $x \mapsto (-1)^x$, for $x \in \{0,1\}$.



Sequences in S_2 can be read off the labels of paths in the above graph.

Computation of Fourier coefficients

- We define the set of vectors (when n is odd)

$$\boldsymbol{b}_0 := 100\ldots00, \qquad \boldsymbol{b}_1 := 0\underbrace{11}00\ldots00,$$

$$\boldsymbol{b}_2 := 000\underbrace{11}\ldots00, \quad \ldots \quad \boldsymbol{b}_{\lceil\frac{n}{2}\rceil} := 000\ldots00\underbrace{11},$$

  and similarly when n is even.

- We let $V_{\mathcal{B}}$ denote the span$(\{\boldsymbol{b}_0, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_{\lceil\frac{n}{2}\rceil}\})$. Then,

Computation of Fourier coefficients

- We define the set of vectors (when n is odd)

$$\boldsymbol{b}_0 := 100\ldots00, \qquad \boldsymbol{b}_1 := 0\underbrace{11}00\ldots00,$$

$$\boldsymbol{b}_2 := 000\underbrace{11}\ldots00, \quad \ldots \quad \boldsymbol{b}_{\lceil\frac{n}{2}\rceil} := 000\ldots00\underbrace{11},$$

and similarly when n is even.

- We let $V_{\mathcal{B}}$ denote the span$(\{\boldsymbol{b}_0, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_{\lceil\frac{n}{2}\rceil}\})$. Then,

Theorem

For $\boldsymbol{a} = \left(a_0, a_1, \ldots, a_{\lceil\frac{n}{2}\rceil-1}\right) \in \{0,1\}^{\lceil\frac{n}{2}\rceil}$, *consider* $\boldsymbol{s} = \sum_{i=0}^{\lceil\frac{n}{2}\rceil-1} a_i \cdot \boldsymbol{b}_i$. *It holds that*

$$\widehat{\mathbb{1}_{S_2}}(\boldsymbol{s}) = 2^{\lfloor\frac{n}{2}\rfloor-n} \cdot (-1)^{w(\boldsymbol{a})-a_0}.$$

Further, for $\boldsymbol{s} \notin V_{\mathcal{B}}$, *we have that* $\widehat{\mathbb{1}_{S_2}}(\boldsymbol{s}) = 0$.

Some consequences

Consider now the following condition (**C**):

$$(\mathbf{C}) \quad \text{For all } \boldsymbol{s} \in \mathcal{C}^{\perp} \cap V_{\mathcal{B}}, \text{ it holds that } \widehat{\mathbb{1}_{S_2}}(\boldsymbol{s}) \geq 0.$$

Then, applying our approach and using the previous theorem, we get that

Some consequences

Consider now the following condition (**C**):

$$(\mathbf{C}) \quad \text{For all } \boldsymbol{s} \in \mathcal{C}^{\perp} \cap V_{\mathcal{B}}, \text{ it holds that } \widehat{\mathbb{1}_{S_2}}(\boldsymbol{s}) \geq 0.$$

Then, applying our approach and using the previous theorem, we get that

- If condition (**C**) is not satisfied, then, $N(\mathcal{C}; S_2) = 0$.
- If condition (**C**) is satisfied and there exist $t_n \in \left[1 : \left\lceil \frac{n}{2} \right\rceil - 1\right]$ linearly independent vectors $(\boldsymbol{s}_1, \ldots, \boldsymbol{s}_{t_n})$ in $\mathcal{C}^{\perp}$ with $\widehat{\mathbb{1}_{S_2}}(\boldsymbol{s}_i) > 0$, for all $1 \leq i \leq t_n$, then, $N(\mathcal{C}; S_2) = |\mathcal{C}| \cdot 2^{t_n + \left\lfloor \frac{n}{2} \right\rfloor - n}$.

Some consequences

Consider now the following condition (**C**):

$$(\mathbf{C}) \quad \text{For all } \boldsymbol{s} \in \mathcal{C}^\perp \cap V_{\mathcal{B}}, \text{ it holds that } \widehat{\mathbb{1}_{S_2}}(\boldsymbol{s}) \geq 0.$$

Then, applying our approach and using the previous theorem, we get that

- If condition (**C**) is not satisfied, then, $N(\mathcal{C}; S_2) = 0$.
- If condition (**C**) is satisfied and there exist $t_n \in \left[1 : \left\lceil \frac{n}{2} \right\rceil - 1\right]$ linearly independent vectors $(\boldsymbol{s}_1, \ldots, \boldsymbol{s}_{t_n})$ in $\mathcal{C}^\perp$ with $\widehat{\mathbb{1}_{S_2}}(\boldsymbol{s}_i) > 0$, for all $1 \leq i \leq t_n$, then, $N(\mathcal{C}; S_2) = |\mathcal{C}| \cdot 2^{t_n + \left\lfloor \frac{n}{2} \right\rfloor - n}$.

Hence, it is possible to construct a sequence $\{\mathcal{C}^{(n)}\}_{n \geq 1}$ of linear codes of rate R such that the rate of their constrained subcodes is

$$\liminf_{n \to \infty} \text{rate}\left(\mathcal{C}_2^{(n)}\right) = R - \frac{1}{2} + \liminf_{n \to \infty} \frac{t_n}{n}.$$

Therefore, we can obtain rates better than the coset-averaging bound of [Patapoutian and Kumar (1992)], using subcodes!

Applications to specific linear codes

We also applied our approach to count the number of words in S_2, in specific linear codes.

Applications to specific linear codes

We also applied our approach to count the number of words in S_2, in specific linear codes.

Hamming codes:

Theorem

For $m \geq 3$ and for $\mathcal{C}$ being the $[2^m - 1, 2^m - 1 - m]$ Hamming code under a canonical coordinate ordering, we have that

$$N(\mathcal{C}; S_2) = 2^{\left\lfloor \frac{2^m - 1}{2} \right\rfloor - 1}.$$

Applications to specific linear codes

We also applied our approach to count the number of words in S_2, in specific linear codes.

Hamming codes:

Theorem

For $m \geq 3$ and for $\mathcal{C}$ being the $[2^m - 1, 2^m - 1 - m]$ Hamming code under a canonical coordinate ordering, we have that

$$N(\mathcal{C}; S_2) = 2^{\left\lfloor \frac{2^m-1}{2} \right\rfloor - 1}.$$

Reed-Muller codes:

(m, r)	$(5, 3)$	$(6, 4)$	$(7, 5)$	$(8, 6)$
$N(\mathrm{RM}(m, r); S_2)$	2048	6.711×10^7	1.441×10^{17}	1.329×10^{36}

Some sample numerical values for high rate RM codes

Example 2: (d, ∞)-RLL constraint

- We return to our familiar runlength-limited constraint, which requires that there be at least d 0s between successive 1s. Let S^d denote the set of constrained sequences.

- Let $\widehat{\mathbb{1}_{S^d}}^{(n)}$ denote the Fourier transform at blocklength $n \geq 1$.

Example 2: (d, ∞)-RLL constraint

- We return to our familiar runlength-limited constraint, which requires that there be at least d 0s between successive 1s. Let S^d denote the set of constrained sequences.

- Let $\widehat{\mathbb{1}_{S^d}}^{(n)}$ denote the Fourier transform at blocklength $n \geq 1$.

Theorem

For $n \geq d+2$ and for $\mathbf{s} = (s_1, \ldots, s_n) \in \{0,1\}^n$, it holds that

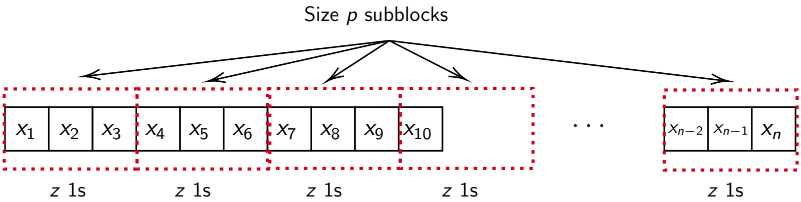
$$\widehat{\mathbb{1}_{S^d}}^{(n)}(\mathrm{s}) = 2^{-1} \cdot \widehat{\mathbb{1}_{S^d}}^{(n-1)}(s_2^n) + (-1)^{s_1} \cdot 2^{-(d+1)} \cdot \widehat{\mathbb{1}_{S^d}}^{(n-d-1)}\left(s_{d+2}^n\right).$$

- The above theorem gives rise to a recursive algorithm for computing $\widehat{\mathbb{1}_{S^d}}^{(n)}$.

This recursive procedure is faster than the Fast Walsh-Hadamard Transform!

Example 3: Subblock-composition constraints

- Next, consider the so-called subblock composition constraint, which admits binary sequences that have a fixed number of 1s in each "subblock".
  - Application: In energy-harvesting communication, ensures that the receiver's battery does not drain out during periods of low symbol energy[1].



- Let C_z^p denote the set of such constrained sequences.

[1] A. Tandon, M. Motani, and L. R. Varshney, "Subblock-constrained codes for real-time simultaneous energy and information transfer," T-IT, 2016.

Computation of Fourier coefficients

- Standard arguments lead us to the characterization of Fourier coefficients:

Theorem

For $\mathbf{s} \in \{0,1\}^n$ with $\mathbf{s} = \mathbf{s}_1\mathbf{s}_2\ldots\mathbf{s}_p$, we have that

$$2^n \cdot \widehat{\mathbb{1}_{C_z^p}}(\mathbf{s}) = \prod_{\ell=1}^{p} K_z^{(n/p)}(w(\mathbf{s}_\ell)),$$

where $K_i^{(n/p)}(j) = \sum_{t=0}^{i}(-1)^t\binom{j}{t}\binom{n/p-j}{i-t}$ is the i^{th}-Krawtchouk polynomial, for the length n/p.

[2] V. A. R. and N. Kashyap, "Estimating the sizes of binary error-correcting constrained codes," submitted to the IEEE JSAIT.

Computation of Fourier coefficients

- Standard arguments lead us to the characterization of Fourier coefficients:

Theorem

For $\mathbf{s} \in \{0,1\}^n$ *with* $\mathbf{s} = \mathbf{s}_1\mathbf{s}_2\ldots\mathbf{s}_p$, *we have that*

$$2^n \cdot \widehat{\mathbb{1}_{C_z^p}}(\mathbf{s}) = \prod_{\ell=1}^{p} K_z^{(n/p)}(w(\mathbf{s}_\ell)),$$

where $K_i^{(n/p)}(j) = \sum_{t=0}^{i}(-1)^t\binom{j}{t}\binom{n/p-j}{i-t}$ *is the* i^{th}*-Krawtchouk polynomial, for the length* n/p.

- We show[2] that this characterization can help speed up the computation of constrained subcodes of RM codes, for select values of p.

[2] V. A. R. and N. Kashyap, "Estimating the sizes of binary error-correcting constrained codes," submitted to the IEEE JSAIT.

Some questions

- Can we come up with recursive algorithms for efficient computation of the Fourier coefficients for a large class of constraints (say, finite-type constraints, representable by a finite, labelled, directed graph)?

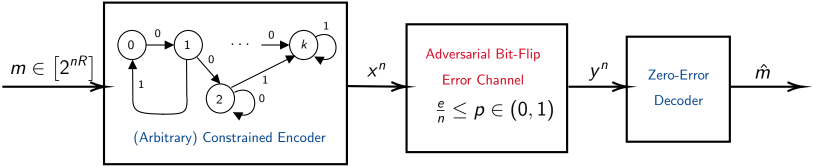
- Can we obtain estimates of the asymptotic rates of constrained subcodes of specific linear codes, using our methods?

- Can we find more use cases of a Fourier-analytic approach to counting constrained codewords?
  - For example, our Fourier coefficients can also be used to compute the weight distributions of constrained words in $\mathbb{F}_2^n$ and in a given linear code $\mathcal{C}$.

Part 2: A version of Delsarte's LP for constrained systems

The setting

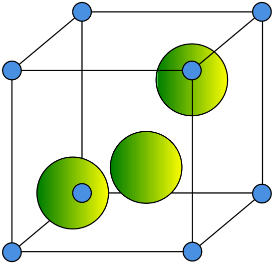
- Consider the channel model where the constrained codewords we intend transmitting, are subjected to adversarial (or worst-case) bit-flip errors (or erasures).



- We wish to uniquely recover the transmitted codeword, with zero error.
- Equivalently, we wish to come up with bounds on the sizes of constrained codes with a given minimum Hamming distance.

The problem

- Given a length-n constrained system represented by a set $\mathcal{A}$ of constrained words, what is the largest size of a constrained code with minimum distance at least d?



- This is a packing problem for which there exist generalized sphere packing bounds [Fazeli, Vardy, Yaakobi (2015), Cullina, Kiyavash (2016)]

Can we improve on these bounds?

Delsarte's LP (for unconstrained systems)

Let $A(n, d)$ denote the size of the largest (not necessarily linear) length-n code with minimum distance at least d.

Consider the following LP L:

Delsarte's LP (for unconstrained systems)

Let $A(n,d)$ denote the size of the largest (not necessarily linear) length-n code with minimum distance at least d.

Consider the following LP L:

$$\begin{aligned}
\text{maximize} \quad & \sum_{w=0}^{n} a_w \\
\text{subj. to} \quad & a_w \geq 0, \text{ for all } w \in [0:n], \\
& \sum_{j=0}^{n} a_j \cdot K_w(j) \geq 0, \text{ for all } w \in [0:n], \\
& a_w = 0, \text{ for } w \in [1:d-1], \\
& a_0 = 1.
\end{aligned}$$

- Delsarte (1973) proved that the distance distribution $(b_w : 0 \leq w \leq n)$ of *any* binary length-n code $\mathcal{C}$ of minimum distance at least d is a feasible solution to the LP above, with objective value $|\mathcal{C}|$.
- Hence, $A(n,d) \leq \text{val}(\text{L})$.

Taking a step back: an alternative formulation

Del(n, d)

$$\underset{f\colon \{0,1\}^n \to \mathbb{R}}{\text{maximize}} \quad \sum_{\boldsymbol{x} \in \{0,1\}^n} f(\boldsymbol{x})$$

subject to:

$$f(\boldsymbol{x}) \geq 0, \ \forall \ \boldsymbol{x} \in \{0,1\}^n,$$

$$\widehat{f}(\boldsymbol{s}) \geq 0, \ \forall \ \boldsymbol{s} \in \{0,1\}^n,$$

$$f(\boldsymbol{x}) = 0, \ \text{if } 1 \leq w(\boldsymbol{x}) \leq d-1,$$

$$f(0^n) = 1.$$

Taking a step back: an alternative formulation

Del(n, d)

$$\underset{f\colon \{0,1\}^n \to \mathbb{R}}{\text{maximize}} \quad \sum_{\boldsymbol{x} \in \{0,1\}^n} f(\boldsymbol{x})$$

subject to:

$$f(\boldsymbol{x}) \geq 0, \ \forall\ \boldsymbol{x} \in \{0,1\}^n,$$
$$\widehat{f}(\boldsymbol{s}) \geq 0, \ \forall\ \boldsymbol{s} \in \{0,1\}^n,$$
$$f(\boldsymbol{x}) = 0, \ \text{if } 1 \leq w(\boldsymbol{x}) \leq d-1,$$
$$f(0^n) = 1.$$

- A feasible solution to the above LP is $\frac{2^n}{|\mathcal{C}|} \cdot (\mathbb{1}_{\mathcal{C}} \star \mathbb{1}_{\mathcal{C}})$, for $\mathcal{C}$ of minimum distance at least d. Here, for $f, g : \{0,1\}^n \to \mathbb{R}$,

$$(f \star g)(\boldsymbol{x}) := \frac{1}{2^n} \sum_{\boldsymbol{z} \in \{0,1\}^n} f(\boldsymbol{z}) \cdot g(\boldsymbol{x} + \boldsymbol{z}).$$

- The objective value again is $|\mathcal{C}|$.

Our LP for constrained systems

Let $A(n, d; \mathcal{A})$ denote the largest length-n $\mathcal{A}$-constrained code of minimum distance at least d.

$\mathrm{Del}(n, d; \mathcal{A})$

$$\underset{f\colon \{0,1\}^n \to \mathbb{R}}{\text{maximize}} \quad \sum_{\boldsymbol{x} \in \{0,1\}^n} f(\boldsymbol{x})$$

subject to:

$$f(\boldsymbol{x}) \geq 0, \ \forall \ \boldsymbol{x} \in \{0,1\}^n,$$

$$\widehat{f}(\boldsymbol{s}) \geq 0, \ \forall \ \boldsymbol{s} \in \{0,1\}^n,$$

$$f(\boldsymbol{x}) = 0, \ \text{if } 1 \leq w(\boldsymbol{x}) \leq d-1,$$

$$f(0^n) \leq \mathrm{val}(\mathrm{Del}(n, d)),$$

$$f(\boldsymbol{x}) \leq 2^n \cdot (\mathbb{1}_{\mathcal{A}} \star \mathbb{1}_{\mathcal{A}})(\boldsymbol{x}), \ \forall \ \boldsymbol{x} \in \{0,1\}^n.$$

- A feasible solution to $\mathrm{Del}(n, d; \mathcal{A})$ can again be constructed, whose objective value is $(A(n, d; \mathcal{A}))^2$.

How good is the LP?

- We ran our LP on the $(1,\infty)$-RLL constrained system ($n = 10$) ...

d	$\text{Del}(n, d; S_{(1,\infty)})$	$\text{GenSph}(n, d; S_{(1,\infty)})$	$\text{Del}(n, d)$
2	128.557	144	512
3	74.762	111	85.333
4	42.048	111	42.667
5	12	63	12
6	6	63	6
7	3.2	26	3.2

How good is the LP?

- We ran our LP on the $(1,\infty)$-RLL constrained system ($n = 10$) ...

d	$\mathrm{Del}(n,d;S_{(1,\infty)})$	$\mathrm{GenSph}(n,d;S_{(1,\infty)})$	$\mathrm{Del}(n,d)$
2	128.557	144	512
3	74.762	111	85.333
4	42.048	111	42.667
5	12	63	12
6	6	63	6
7	3.2	26	3.2

- ... and on the $(2,\infty)$-RLL constrained system ($n = 10$):

d	$\mathrm{Del}(n,d;S_{(2,\infty)})$	$\mathrm{GenSph}(n,d;S_{(2,\infty)})$	$\mathrm{Del}(n,d)$
2	49.578	60	512
3	32.075	46.5	85.333
4	21.721	46.5	42.667
5	7.856	34	12
6	4.899	34	6
7	2.529	19	3.2

How good is the LP?

- We ran our LP on the $(1,\infty)$-RLL constrained system $(n = 10)$ ...

d	$\mathrm{Del}(n, d; S_{(1,\infty)})$	$\mathrm{GenSph}(n, d; S_{(1,\infty)})$	$\mathrm{Del}(n, d)$
2	128.557	144	512
3	74.762	111	85.333
4	42.048	111	42.667
5	12	63	12
6	6	63	6
7	3.2	26	3.2

- ... and on the $(2,\infty)$-RLL constrained system $(n = 10)$:

d	$\mathrm{Del}(n, d; S_{(2,\infty)})$	$\mathrm{GenSph}(n, d; S_{(2,\infty)})$	$\mathrm{Del}(n, d)$
2	49.578	60	512
3	32.075	46.5	85.333
4	21.721	46.5	42.667
5	7.856	34	12
6	4.899	34	6
7	2.529	19	3.2

- Our LP appears to perform better than the generalized sphere packing upper bounds.

Some comments

- The LP $\mathrm{Del}(n, d; \mathcal{A})$ as stated has 2^n variables for a blocklength n. For select constraints, it is indeed possible to "symmetrize" the LP using the symmetry group of the constraint (i.e., the group of index permutations that leave $\mathbb{1}_{\mathcal{A}}$ unchanged).

  This symmetrization can greatly reduce the size of the LP (sometimes resulting in only polynomially many variables + constraints!)

Some comments

- The LP $\mathsf{Del}(n, d; \mathcal{A})$ as stated has 2^n variables for a blocklength n. For select constraints, it is indeed possible to "symmetrize" the LP using the symmetry group of the constraint (i.e., the group of index permutations that leave $\mathbb{1}_{\mathcal{A}}$ unchanged).

  This symmetrization can greatly reduce the size of the LP (sometimes resulting in only polynomially many variables + constraints!)

- The LP also satisfies the sanity check:

$$\mathsf{val}(\mathsf{Del}(n, d; \mathcal{A}))^{1/2} \leq \min\{\mathsf{val}(\mathsf{Del}(n, d)), |\mathcal{A}|\}.$$

Some comments

- The LP $\mathrm{Del}(n, d; \mathcal{A})$ as stated has 2^n variables for a blocklength n. For select constraints, it is indeed possible to "symmetrize" the LP using the symmetry group of the constraint (i.e., the group of index permutations that leave $\mathbb{1}_{\mathcal{A}}$ unchanged).

  This symmetrization can greatly reduce the size of the LP (sometimes resulting in only polynomially many variables + constraints!)

- The LP also satisfies the sanity check:

$$\mathrm{val}(\mathrm{Del}(n, d; \mathcal{A}))^{1/2} \leq \min\{\mathrm{val}(\mathrm{Del}(n, d)), |\mathcal{A}|\}.$$

- Future work can apply modern Fourier analytic techniques à la [Navon, Samorodnitsky (2005)], [Loyfer, Linial (2022)] or expander graph-related tools [Friedman, Tillich (2005)] to obtain asymptotic rate-distance tradeoff upper bounds using our LP.

Some questions

- Can we derive asymptotic upper bounds on the rate-distance tradeoff for constrained codes, using our LP formulation?

- Can we connect our Fourier-analytic techniques of counting, with this packing problem?

Thank You!