Sampling-Based Estimates of Weight Enumerators of Reed-Muller Codes

Navin Kashyap

Indian Institute of Science

in collaboration with

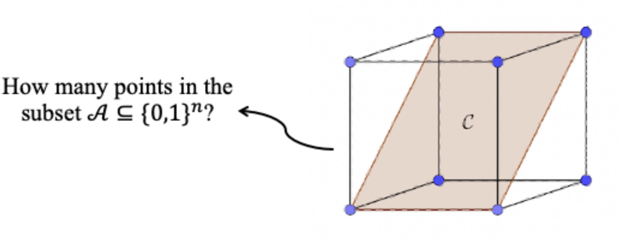
V. Arvind Rameshwar — Indian Institute of Science

Shreyas Jain — IISER, Mohali

ITA 2024

What is the talk about?

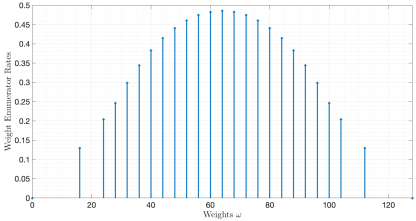
Let $\mathcal{C} \subseteq \{0,1\}^n$ be a binary Reed-Muller (RM) code.



This talk discusses a sampling-based algorithmic approach for obtaining reliable numerical estimates of the above count.

Our specific interest is in sets of the form $\mathcal{A}_\mathrm{w} = \{\mathbf{x} : w_H(\mathbf{x}) = \mathrm{w}\}$, and the weight enumerators $A_\mathrm{w} := |\mathcal{C} \cap \mathcal{A}_\mathrm{w}|$.

Why are weight enumerators useful?



- They bound the probability of ML decoding error over a binary-input memoryless symmetric (BMS) channel:

$$P_{\mathrm{err}} \le \sum_{\mathrm{w}=1}^{n} A_{\mathrm{w}} z^{\mathrm{w}},$$

where $z = \sum_y \sqrt{P(y|0)P(y|1)}$ is the Bhattacharyya parameter.

- This connection has been exploited in many papers to analyze the performance of ML/MAP decoding over BMS channels:
  [Abbe-Shpilka-Wigderson (T-IT 2015)], [Kudekar et al. (ISIT 2016)], [Sberlo-Shpilka (arXiv:1811.12447)]

Brief Background

Definition of an RM code

- Fix $m \geq 1$ and consider the points $(x_1, \ldots, x_m)$ of the Boolean hypercube $\{0,1\}^m$.
- Define $x_S := \prod_{i \in S} x_i$, where $S \subseteq [m]$.
- Pick a multilinear polynomial $f = \sum_{S \in \mathcal{S}} x_S$, where $\mathcal{S} \subseteq 2^{[m]}$, with

$$\deg(f) = \max_{S \in \mathcal{S}} |S| \leq r.$$

- Evaluate f at all points in $\{0,1\}^m$ in the (lexicographic) order:

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

  and call the resultant vector $\mathrm{Eval}(f)$. Here, blocklength $n = 2^m$.
- The code $\mathrm{RM}(m, r)$ consists of all $\mathrm{Eval}(f)$, where f is as above.

Dimension, $d_{\min}$, and other useful things

- Dimension :

$$\begin{aligned}\dim(\mathrm{RM}(m,r)) &= \#\{x_S : \ \deg(x_S) = |S| \leq r\} \\ &= \sum_{i=0}^{r} \binom{m}{i} =: \binom{m}{\leq r}.\end{aligned}$$

Dimension, $d_{\min}$, and other useful things

- Dimension :

$$\dim(\mathrm{RM}(m,r)) = \#\{x_S : \deg(x_S) = |S| \le r\}$$
$$= \sum_{i=0}^{r} \binom{m}{i} =: \binom{m}{\le r}.$$

- Minimum distance :

$$d_{\min}(\mathrm{RM}(m,r)) = w_H(\mathrm{Eval}(x_1 x_2 \ldots x_r)) = 2^{m-r}.$$

- Every minimum-weight codeword $\boldsymbol{v} \in \mathrm{RM}(m,r)$ can be expressed as

$$\boldsymbol{v} = \left(\mathbb{1}_H(\boldsymbol{z}) : \ \boldsymbol{z} \in \{0,1\}^m\right),$$

  for an $(m-r)$-dimensional affine subspace H of $\mathbb{F}_2^m$.

- The minimum-weight codewords span $\mathrm{RM}(m,r)$.

Prior Work on Weight Enumerators of RM Codes

Exact expressions/values

- RM$(m, 1)$: $A_0 = A_{2^m} = 1$, $A_{2^{m-1}} = 2^{m+1} - 2$
- RM$(m, 2)$: Sloane-Berlekamp (1970)
- RM$(7, 3)$: Sugino-Ienaga-Tokura-Kasami (1971)
  RM$(9, 3)$: Sugita-Kasami-Fujiwara (1996)
  RM$(9, 4)$: Markov-Borissov (2023)
- RM(m, r): Exact A_w known for $w < 2.5 \cdot 2^{m-r}$
  Kasami-Tokura (1970), Kasami-Tokura-Azumi (1976)

Analytical bounds

- Kaufman-Lovett-Porat (2012), Sberlo-Shpilka (2015), Samorodnitsky (2020), Rao-Sprumont (2022):
  Bounds on A_w via Fourier analysis on the hypercube

Algorithms

- Sarwate (1973): Recursive algorithm using Plotkin decomposition

Weight Spectra of RM Codes

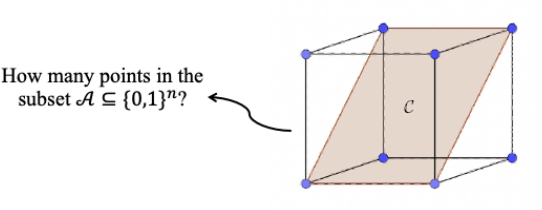
The weight spectrum of a code is the set $\mathcal{W} = \{\mathrm{w} : A_{\mathrm{w}} \neq 0\}$.

We will denote the weight spectrum of RM(m, r) by $\mathcal{W}_{m,r}$.

- McEliece (1972): $\mathcal{W}_{m,r} \subset \{\text{multiples of } 2^{\lfloor \frac{m-1}{r} \rfloor}\}$
- Carlet and Solé (2023):
  - $\mathcal{W}_{m,m-3}$ for $m \geq 6$, and $\mathcal{W}_{m,m-4}$ for $m \geq 8$
  - $\mathcal{W}_{8,3}$ and $\mathcal{W}_{9,4}$
- Carlet (2023): $\mathcal{W}_{m,m-5}$ for $m \geq 10$

What are we shooting for?

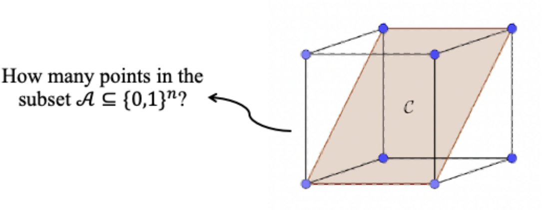
Let $\mathcal{C} \subseteq \{0,1\}^n$ be a binary Reed-Muller code.



- Exact computation of A_w is hard (algebraically) and computationally intractable (numerically)

What are we shooting for?

Let $\mathcal{C} \subseteq \{0,1\}^n$ be a binary Reed-Muller code.

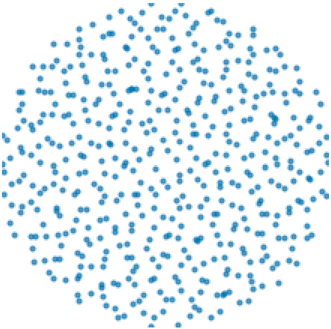


- Exact computation of A_{w} is hard (algebraically) and computationally intractable (numerically)
- Can we efficiently (poly. time?) obtain an estimate $\widehat{A}_{\mathrm{w}}$, such that with high probability,

$$A_{\mathrm{w}} \in [(1-\epsilon)A_{\mathrm{w}}, (1+\epsilon)A_{\mathrm{w}}],$$

for some arbitrarily small $\epsilon > 0$?

Sampling-Based Algorithms



A naïve first pass

- Suppose that we try to construct A_{w} via "rejection sampling":

1. Draw L uniformly random codewords from $\mathrm{RM}(m, r)$.
2. Set $\widehat{A}_{\mathrm{w}} = |\mathrm{RM}(m, r)| \times \left(\frac{\#\{\text{samples of weight w}\}}{L} \right)$.

A naïve first pass

- Suppose that we try to construct A_{w} via "rejection sampling":

> 1. Draw L uniformly random codewords from $\mathrm{RM}(m, r)$.
> 2. Set $\widehat{A}_{\mathrm{w}} = |\mathrm{RM}(m, r)| \times \left(\frac{\#\{\text{samples of weight w}\}}{L} \right)$.

Clearly, for L large, $\widehat{A}_{\mathrm{w}} \in [(1-\epsilon)A_{\mathrm{w}}, (1+\epsilon)A_{\mathrm{w}}]$ w.h.p.

A naïve first pass

- Suppose that we try to construct A_{w} via "rejection sampling":

> 1. Draw L uniformly random codewords from $\mathrm{RM}(m, r)$.
> 2. Set $\widehat{A}_{\mathrm{w}} = |\mathrm{RM}(m, r)| \times \left(\frac{\#\{\text{samples of weight w}\}}{L} \right)$.

Clearly, for L large, $\widehat{A}_{\mathrm{w}} \in [(1-\epsilon)A_{\mathrm{w}}, (1+\epsilon)A_{\mathrm{w}}]$ w.h.p.

- For most weights, A_{w} is exp. smaller than $\mathrm{RM}(m, r)$!
  see e.g., [Rao and Sprumont (2022)]

Hence, exponentially many (in blocklength n) draws needed (bad)!

Key insight

- Note that $A_{\mathrm{w}} = Z$, the partition function of the distrib. p given by

$$p(\boldsymbol{x}) = \frac{1}{Z} \cdot \mathbb{1}_{\mathcal{C} \cap \mathcal{A}_{\mathrm{w}}}(\boldsymbol{x}), \quad \boldsymbol{x} \in \{0, 1\}^n.$$

Key insight

- Note that $A_{\mathrm{w}} = Z$, the partition function of the distrib. p given by

$$p(\boldsymbol{x}) = \frac{1}{Z} \cdot \mathbb{1}_{\mathcal{C} \cap \mathcal{A}_{\mathrm{w}}}(\boldsymbol{x}), \quad \boldsymbol{x} \in \{0,1\}^n.$$

Hard to sample from p or compute Z

Key insight

- Note that $A_{\mathrm{w}} = Z$, the partition function of the distrib. p given by

$$p(\boldsymbol{x}) = \frac{1}{Z} \cdot \mathbb{1}_{\mathcal{C} \cap \mathcal{A}_{\mathrm{w}}}(\boldsymbol{x}), \quad \boldsymbol{x} \in \{0,1\}^n.$$

- Consider the following Gibbs distribution p_β, for $\beta > 0$:

$$p_\beta(\boldsymbol{x}) = \frac{1}{Z_\beta} \cdot e^{-\beta \cdot E(\boldsymbol{x})}, \quad \boldsymbol{x} \in \mathcal{C},$$

where $E(\boldsymbol{x}) = |w_H(\boldsymbol{x}) - \mathrm{w}|$ and $Z_\beta = \sum_{\mathbf{c} \in \mathcal{C}} e^{-\beta \cdot E(\mathbf{x})}$.

Key insight

- Note that $A_{\mathrm{w}} = Z$, the partition function of the distrib. p given by

$$p(\boldsymbol{x}) = \frac{1}{Z} \cdot \mathbb{1}_{\mathcal{C} \cap \mathcal{A}_{\mathrm{w}}}(\boldsymbol{x}), \quad \boldsymbol{x} \in \{0, 1\}^n.$$

- Consider the following Gibbs distribution p_β, for $\beta > 0$:

$$p_\beta(\boldsymbol{x}) = \frac{1}{Z_\beta} \cdot e^{-\beta \cdot E(\boldsymbol{x})}, \quad \boldsymbol{x} \in \mathcal{C},$$

where $E(\boldsymbol{x}) = |w_H(\boldsymbol{x}) - \mathrm{w}|$ and $Z_\beta = \sum_{\mathbf{c} \in \mathcal{C}} e^{-\beta \cdot E(\mathbf{x})}$.

- Importantly,

$$\lim_{\beta \to \infty} p_\beta(\boldsymbol{x}) = p(\boldsymbol{x}), \quad \text{and}$$
$$\lim_{\beta \to \infty} Z_\beta = Z.$$

Key insight

- Note that $A_{\mathrm{w}} = Z$, the partition function of the distrib. p given by

$$p(\boldsymbol{x}) = \frac{1}{Z} \cdot \mathbb{1}_{\mathcal{C} \cap \mathcal{A}_{\mathrm{w}}}(\boldsymbol{x}), \quad \boldsymbol{x} \in \{0,1\}^n.$$

- Consider the following Gibbs distribution p_β, for $\beta > 0$:

$$p_\beta(\boldsymbol{x}) = \frac{1}{Z_\beta} \cdot e^{-\beta \cdot E(\boldsymbol{x})}, \quad \boldsymbol{x} \in \mathcal{C},$$

where $E(\boldsymbol{x}) = |w_H(\boldsymbol{x}) - \mathrm{w}|$ and $Z_\beta = \sum_{\mathbf{c} \in \mathcal{C}} e^{-\beta \cdot E(\mathbf{x})}$.

- Importantly,

$$\lim_{\beta \to \infty} p_\beta(\boldsymbol{x}) = p(\boldsymbol{x}), \quad \text{and}$$
$$\lim_{\beta \to \infty} Z_\beta = Z.$$

We use $Z_{\beta^\star}$, for large $\beta^\star$, as a "good" approximation to $Z = A_{\mathrm{w}}$.

Key insight

- Note that $A_{\mathrm{w}} = Z$, the partition function of the distrib. p given by

$$p(\boldsymbol{x}) = \frac{1}{Z} \cdot \mathbb{1}_{\mathcal{C} \cap \mathcal{A}_{\mathrm{w}}}(\boldsymbol{x}), \quad \boldsymbol{x} \in \{0,1\}^n.$$

- Consider the following Gibbs distribution p_β, for $\beta > 0$:

$$p_\beta(\boldsymbol{x}) = \frac{1}{Z_\beta} \cdot e^{-\beta \cdot E(\boldsymbol{x})}, \quad \boldsymbol{x} \in \mathcal{C},$$

where $E(\boldsymbol{x}) = |w_H(\boldsymbol{x}) - \mathrm{w}|$ and $Z_\beta = \sum_{\mathbf{c} \in \mathcal{C}} e^{-\beta \cdot E(\mathbf{x})}$.

- Importantly,

$$\lim_{\beta \to \infty} p_\beta(\boldsymbol{x}) = p(\boldsymbol{x}), \quad \text{and}$$
$$\lim_{\beta \to \infty} Z_\beta = Z.$$

We next illustrate how to approximately compute $Z_{\beta^\star}$.

Counting via sampling - I

The following technique from statistical physics is well-known [Valleau and Card (1972)]:

- Fix a large $\ell > 0$ and a "cooling schedule" of β parameters:

$$0 =: \beta_0 < \beta_1 < \ldots < \beta_\ell =: \beta^\star,$$

where $\beta_i = \beta_{i-1} + 1/n$, for $1 \leq i \leq \ell$.

Counting via sampling - I

The following technique from statistical physics is well-known [Valleau and Card (1972)]:

- Fix a large $\ell > 0$ and a "cooling schedule" of β parameters:

$$0 =: \beta_0 < \beta_1 < \ldots < \beta_\ell =: \beta^\star,$$

  where $\beta_i = \beta_{i-1} + 1/n$, for $1 \le i \le \ell$.

- Write

$$Z_{\beta^\star} = Z_{\beta_0} \times \prod_{i=1}^{\ell} \frac{Z_{\beta_i}}{Z_{\beta_{i-1}}},$$

  where $Z_{\beta_0} = Z_0 = |\mathcal{C}|$.

Counting via sampling - I

The following technique from statistical physics is well-known [Valleau and Card (1972)]:

- Fix a large $\ell > 0$ and a "cooling schedule" of β parameters:

$$0 =: \beta_0 < \beta_1 < \ldots < \beta_\ell =: \beta^\star,$$

where $\beta_i = \beta_{i-1} + 1/n$, for $1 \leq i \leq \ell$.

- Write

$$Z_{\beta^\star} = Z_{\beta_0} \times \prod_{i=1}^{\ell} \frac{Z_{\beta_i}}{Z_{\beta_{i-1}}},$$

where $Z_{\beta_0} = Z_0 = |\mathcal{C}|$.

- Observe that

$$\frac{Z_{\beta_i}}{Z_{\beta_{i-1}}} = \frac{1}{Z_{\beta_{i-1}}} \sum_{\mathbf{c} \in \mathcal{C}} \exp(-\beta_i E(\mathbf{c}))$$
$$= \mathbb{E}_{p_{\beta_{i-1}}}[\exp(\underbrace{(\beta_{i-1} - \beta_i)}_{=-1/n} E(\mathbf{c}))].$$

Counting via sampling - II

$$Z_{\beta^\star} = Z_{\beta_0} \times \prod_{i=1}^{\ell} \mathbb{E}_{p_{\beta_{i-1}}}\left[\exp(-\tfrac{1}{n} \underbrace{|w_H(\mathbf{c}) - \mathrm{w}|}_{= E(\mathbf{c})})\right].$$

Suppose that we have **black-box access** to samples from p_β. Then,

1. Estimate $\mathbb{E}_{p_{\beta_{i-1}}}\left[\exp(-\frac{1}{n}E(\mathbf{c}))\right]$ as
$$Y_i = \frac{1}{t}\sum_{j=1}^{t} X_{i,j},$$
where $X_{i,j} \overset{\text{iid}}{\sim} p_{\beta_{i-1}}$ and t is "large".
2. Return $\widehat{Z}_{\beta^\star} = Z_0 \times \prod_{i=1}^{\ell} Y_i$.

Counting via sampling - II

$$Z_{\beta^\star} = Z_{\beta_0} \times \prod_{i=1}^{\ell} \mathbb{E}_{p_{\beta_{i-1}}}\left[\exp(-\tfrac{1}{n} \underbrace{|w_H(\mathbf{c}) - \mathrm{w}|}_{= E(\mathbf{c})})\right].$$

Suppose that we have **black-box access** to samples from p_β. Then,

1. Estimate $\mathbb{E}_{p_{\beta_{i-1}}}\left[\exp(-\frac{1}{n}E(\mathbf{c}))\right]$ as

$$Y_i = \frac{1}{t}\sum_{j=1}^{t} X_{i,j},$$

where $X_{i,j} \overset{\text{iid}}{\sim} p_{\beta_{i-1}}$ and t is "large".

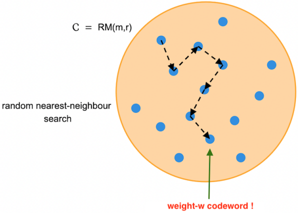
2. Return $\widehat{Z}_{\beta^\star} = Z_0 \times \prod_{i=1}^{\ell} Y_i$.

How to sample from p_β?

How large is ℓ?

How large is t?

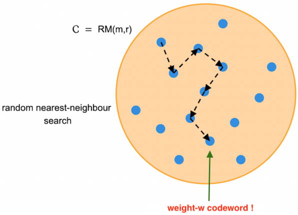
How to sample from p_β?



procedure MCMC-SAMPLER
  Start at an arbitrary codeword $\boldsymbol{c}_0 \in \mathcal{C}$.
  Fix a large epoch length τ.
  for $i = 1 : \tau$ **do**
    Sample a uniformly random min.-wt. codeword $\boldsymbol{c}^\#$, and set $\boldsymbol{c}_{\text{proposed}} \leftarrow \boldsymbol{c}_{i-1} + \boldsymbol{c}^\#$
    $$\boldsymbol{c}_i \leftarrow \begin{cases} \boldsymbol{c}_{\text{proposed}} & \text{w.p. } p_{\text{accept}} \\ \boldsymbol{c}_{i-1} & \text{w.p. } 1 - p_{\text{accept}} \end{cases}$$
  Output $\boldsymbol{c}_\tau$.

$$p_{\text{accept}} := \min\left(1, \exp(-\beta(E(\mathbf{c}_{\text{proposed}}) - E(\mathbf{c}_{(i-1)})))\right)$$

How to sample from p_β?



procedure MCMC-SAMPLER
    Start at an arbitrary codeword $\boldsymbol{c}_0 \in \mathcal{C}$.
    Fix a large epoch length τ.
    for $i = 1 : \tau$ **do**
        Sample a uniformly random min.-wt. codeword $\boldsymbol{c}^{\#}$, and set $\boldsymbol{c}_{\text{proposed}} \leftarrow \boldsymbol{c}_{i-1} + \boldsymbol{c}^{\#}$
        $\boldsymbol{c}_i \leftarrow \begin{cases} \boldsymbol{c}_{\text{proposed}} & \text{w.p. } p_{\text{accept}} \\ \boldsymbol{c}_{i-1} & \text{w.p. } 1 - p_{\text{accept}} \end{cases}$
    Output $\boldsymbol{c}_\tau$.

$$p_{\text{accept}} := \min\left(1, \exp(-\beta(E(\mathbf{c}_{\text{proposed}}) - E(\mathbf{c}_{(i-1)})))\right)$$

Observations:

1. We can efficiently draw unif. random. min.-wt. $\boldsymbol{c}^{\#}$ using the corresp. with $(m-r)$-dimensional affine subspaces.
2. The Metropolis Markov chain above is irreducible (min.-wt. codewords span $\mathcal{C}$) and has p_β as stationary distribution.

How large is ℓ? How large is t?

We now provide bounds on the sample complexity of the approx. counting algorithm.

- Setting $t = \Theta(n^3)$, we have [Dyer and Frieze (1991)]

$$\Pr[(1-\epsilon)Z_{\beta^\star} \leq \widehat{Z}_{\beta^\star} \leq (1+\epsilon)Z_{\beta^\star}] \geq \frac{3}{4}.$$

How large is ℓ? How large is t?

We now provide bounds on the sample complexity of the approx. counting algorithm.

- Setting $t = \Theta(n^3)$, we have [Dyer and Frieze (1991)]

$$\Pr[(1-\epsilon)Z_{\beta^\star} \leq \widehat{Z}_{\beta^\star} \leq (1+\epsilon)Z_{\beta^\star}] \geq \frac{3}{4}.$$

- Setting $\ell = O(n)$, we get [A. Sinclair, lecture notes (2020)]

$$(1-\delta_n)Z \leq Z_{\beta^\star} \leq (1+\delta_n)Z,$$

  where $\delta_n \to 0$ exponentially quickly.

How large is ℓ? How large is t?

We now provide bounds on the sample complexity of the approx. counting algorithm.

- Setting $t = \Theta(n^3)$, we have [Dyer and Frieze (1991)]

$$\Pr[(1-\epsilon)Z_{\beta^\star} \leq \widehat{Z}_{\beta^\star} \leq (1+\epsilon)Z_{\beta^\star}] \geq \frac{3}{4}.$$

- Setting $\ell = O(n)$, we get [A. Sinclair, lecture notes (2020)]

$$(1-\delta_n)Z \leq Z_{\beta^\star} \leq (1+\delta_n)Z,$$

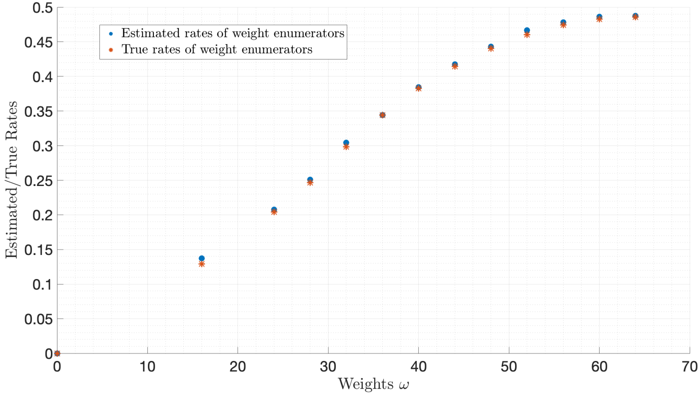
  where $\delta_n \to 0$ exponentially quickly.

- Thus, using $\Theta(n^6)$ samples overall, we obtain that

$$\Pr[(1-\gamma_n)A_{\mathrm{w}} \leq \widehat{Z}_{\beta^\star} \leq (1+\gamma_n)A_{\mathrm{w}}] \geq \frac{3}{4}.$$

- The constant $3/4$ can be improved to $1-\alpha$, for $\alpha > 0$ arbit. small, using a "median-of-batches" trick.

Numerical Results - I

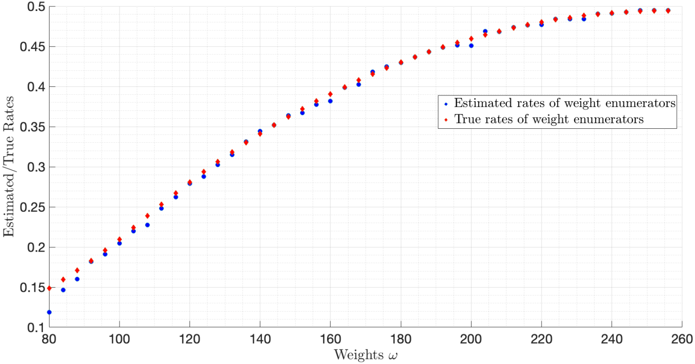


Plots of estimated and exact[1] rates of weight enumerators of RM(7, 3)

- The rate of a weight enumerator is $\frac{1}{n}\log_2 A_w$.

[1]from [Sugino-Ienaga-Tokura-Kasami (1971)]

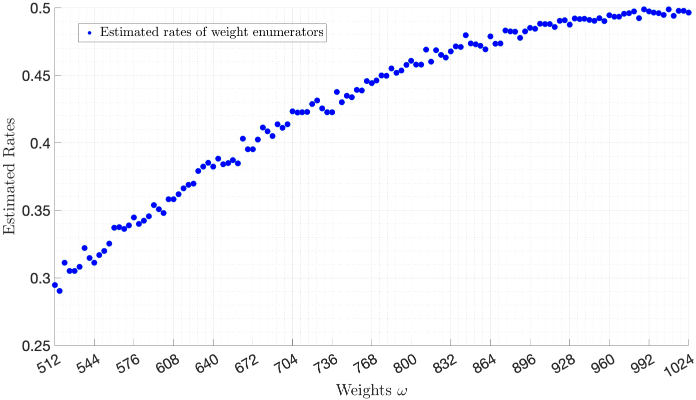
Numerical Results - II



Plots of estimated and exact[2] rates of weight enumerators of RM$(9, 4)$ in the range $80 \leq w \leq 256$

[2]from [Markov-Borissov (2023)]

Numerical Results - III



Plot of estimated rates of weight enumerators of $\mathrm{RM}(11, 5)$ in the range $512 \leq w \leq 1024$

Weight Spectra of RM(10, 3) and RM(10, 4)

Recall that

- For $\mathrm{RM}(m, r)$, A_{w} for $\mathrm{w} < 2.5 \cdot 2^{m-r}$ are known exactly. [Kasami-Tokura (1970)], [Kasami-Tokura-Azumi (1976)]

  By symmetry, these are also known for $\mathrm{w} > 2^m - 2.5 \cdot 2^{m-r}$.

- $\mathcal{W}_{m,r} \subset \{\text{multiples of } 2^{\lfloor \frac{m-1}{r} \rfloor}\}$ [McEliece (1972)]

Weight Spectra of RM(10, 3) and RM(10, 4)

Recall that

- For $\mathrm{RM}(m, r)$, A_{w} for $\mathrm{w} < 2.5 \cdot 2^{m-r}$ are known exactly. [Kasami-Tokura (1970)], [Kasami-Tokura-Azumi (1976)]

  By symmetry, these are also known for $\mathrm{w} > 2^m - 2.5 \cdot 2^{m-r}$.

- $\mathcal{W}_{m,r} \subset \{\text{multiples of } 2^{\lfloor \frac{m-1}{r} \rfloor}\}$ [McEliece (1972)]

Using our sampling algorithm, we can find witnesses (codewords) that prove the following result.

Theorem

For $(m, r) = (10, 3)$ or $(10, 4)$, the weight spectrum in the range $2.5 \cdot 2^{m-r} \leq \mathrm{w} \leq 2^m - 2.5 \cdot 2^{m-r}$ is composed exactly of the multiples of $2^{\lfloor \frac{m-1}{r} \rfloor}$ in that range.

Open Questions

- For $\mathrm{RM}(11, 5)$, can our sampling-based estimates be used in conjunction with algebraic methods (such as the MacWilliams' identities) to determine the exact weight enumerators?

- What can we do to improve our estimates at low weights?

- MCMC-based decoders for RM codes?
  J.-T. Huang and Y.-H. Kim have recently (GLOBECOM'20, ISIT'23) considered MCMC decoders for linear codes.