Insight from Simple Questions: Three Examples

Henry D. Pfister
Duke University

Jack Keil Wolf Lecture on Information Theory and Applications
Center for Magnetic Recording Research
University of California, San Diego
December 1st, 2017
Insight from Simple Questions: Three Examples
(and some things that I learned from Jack)

Henry D. Pfister
Duke University

Jack Keil Wolf Lecture on Information Theory and Applications
Center for Magnetic Recording Research
University of California, San Diego
December 1st, 2017
Acknowledgments

Thanks to my past and present collaborators. In particular, this presentation is based on work with

- Jack Wolf, Roberto Padovani, Jilei Hou, Alessandro Vanelli-Coralli
- John Smee, Joseph Soriaga, Stefano Tomasin
- Santhosh Kumar, Robert Calderbank
- Rüdiger Urbanke, Marco Mondelli, Shrinivas Kudekar, Eren Şaşoğlu

Thanks to Paul Siegel and the selection committee for the invitation to speak today
Jack Wolf liked to ask simple questions whose answers could provide insight.
Jack Wolf liked to ask simple questions whose answers could provide insight.

- Often, they were playful curiosities.
Jack Wolf liked to ask simple questions whose answers could provide insight.

- Often, they were playful curiosities
- But, some had a more profound impact
Jack Wolf liked to ask simple questions whose answers could provide insight. Often, they were playful curiosities. But, some had a more profound impact. This style of inquiry resonated with me.
Jack Wolf liked to ask simple questions whose answers could provide insight

- Often, they were playful curiosities
- But, some had a more profound impact

This style of inquiry resonated with me

But it is an art...

- Simple questions can be quite hard to answer
- Answer doesn’t always give insight
- Even when it should, you might miss it
Jack Wolf liked to ask simple questions whose answers could provide insight

- Often, they were playful curiosities
- But, some had a more profound impact

This style of inquiry resonated with me

But it is an art...

- Simple questions can be quite hard to answer
- Answer doesn’t always give insight
- Even when it should, you might miss it

One approach I like is based on Polya’s Dictum:
If you can’t solve a problem, then there is an easier problem inside it that you can’t solve
Three Simple Questions

- Can one compute the amplitude and frequency of 2 complex sinusoids using only 3 complex samples?

- How can cellular systems with successive cancellation approach the equal-rate point of the multiple-access channel with equal-power users?

- Can Reed-Muller codes achieve capacity?
Can one compute the amplitude and frequency of 2 complex sinusoids using only 3 complex samples?
Can one compute the amplitude and frequency of 2 complex sinusoids using only 3 complex samples?

Assume the samples are $s_0, s_1, s_2$ where

$$s_n = a_1 e^{n\omega_1 T\sqrt{-1}} + a_2 e^{n\omega_2 T\sqrt{-1}} = a_1 \gamma_1^n + a_2 \gamma_2^n$$

$$S(z) = \sum_{n=0}^{\infty} s_n z^{-n} = a_1 \frac{1}{1-\gamma_1 z^{-1}} + a_2 \frac{1}{1-\gamma_2 z^{-1}}.$$
In 1795, Baron Gaspard De Prony published a method that recovers the parameters of a linear combination of $t$ exponentials from $2t$ uniform samples $s_1, \ldots, s_{2t}$.
Prony’s Method

- In 1795, Baron Gaspard De Prony published a method that recovers the parameters of a linear combination of \( t \) exponentials from \( 2t \) uniform samples \( s_1, \ldots, s_{2t} \)

- For \( t = 2 \), Prony’s method designs a 3-tap FIR filter (e.g., with Z-transform \( B(z) = b_0 + b_1z^{-1} + b_2z^{-2} \)) whose zeros cancel the poles in the Z-transform of the \( s_n \) sequence:

\[
B(z)S(z) = \frac{b_0(1 - \gamma_1z^{-1})(1 - \gamma_2z^{-1})}{B(z)} \left[ a_1 \frac{1}{1 - \gamma_1z^{-1}} + a_2 \frac{1}{1 - \gamma_2z^{-1}} \right]
\]

\[
= b_0a_1(1 - \gamma_2z^{-1}) + b_0a_2(1 - \gamma_1z^{-1}).
\]
Prony’s Method

- In 1795, Baron Gaspard De Prony published a method that recovers the parameters of a linear combination of $t$ exponentials from $2t$ uniform samples $s_1, \ldots, s_{2t}$

- For $t = 2$, Prony’s method designs a 3-tap FIR filter (e.g., with Z-transform $B(z) = b_0 + b_1 z^{-1} + b_2 z^{-2}$) whose zeros cancel the poles in the Z-transform of the $s_n$ sequence:

$$B(z)S(z) = b_0(1 - \gamma_1 z^{-1})(1 - \gamma_2 z^{-1}) \left[ a_1 \frac{1}{1 - \gamma_1 z^{-1}} + a_2 \frac{1}{1 - \gamma_2 z^{-1}} \right]$$

$$= b_0 a_1 (1 - \gamma_2 z^{-1}) + b_0 a_2 (1 - \gamma_1 z^{-1}).$$

- Expand the LHS and match $z^{-2}$ and $z^{-3}$ coefficients to get

$$\begin{bmatrix} s_0 & s_1 \\ s_1 & s_2 \end{bmatrix} \begin{bmatrix} b_2 \\ b_1 \end{bmatrix} = -b_0 \begin{bmatrix} s_2 \\ s_3 \end{bmatrix}$$
In 1795, Baron Gaspard De Prony published a method that recovers the parameters of a linear combination of $t$ exponentials from $2t$ uniform samples $s_1, \ldots, s_{2t}$.

For $t = 2$, Prony’s method designs a 3-tap FIR filter (e.g., with Z-transform $B(z) = b_0 + b_1z^{-1} + b_2z^{-2}$) whose zeros cancel the poles in the Z-transform of the $s_n$ sequence:

$$B(z)S(z) = b_0(1 - \gamma_1 z^{-1})(1 - \gamma_2 z^{-1}) \left[ \frac{1}{1 - \gamma_1 z^{-1}} + a_2 \frac{1}{1 - \gamma_2 z^{-1}} \right]$$

$$= b_0 a_1 (1 - \gamma_2 z^{-1}) + b_0 a_2 (1 - \gamma_1 z^{-1}).$$

Expand the LHS and match $z^{-2}$ and $z^{-3}$ coefficients to get

$$\begin{bmatrix} s_0 & s_1 \\ s_1 & s_2 \end{bmatrix} \begin{bmatrix} b_2 \\ b_1 \end{bmatrix} = -b_0 \begin{bmatrix} s_2 \\ s_3 \end{bmatrix}$$

$B(z)$ has zeros at poles of $S(z)$: Factor $B(z)$ to find $\gamma_1, \gamma_2$. 
In 1967, Wolf observed (in a 3/4 page correspondence) that the celebrated PGZ decoding algorithm for RS/BCH codes was mathematically identical to Prony's method.

Decoding of Bose–Chaudhuri–Hocquenghem Codes and Prony's Method for Curve Fitting

The literature is filled with examples of a common set of equations which arise in two or more diverse applications. The purpose of this correspondence is to point out that such a situation has occurred in the two fields of 1) algebraic coding theory, and 2) curve fitting. Two aspects are of particular interest: the large time span which separates the fundamental works in these two areas, and that the methods of solution are identical. Since it is assumed that the readers of this *Transactions* are more familiar with the coding application, the equations are first discussed in terms of the curve fitting application.

Prony[1] has considered the problem of approximating a curve \( f(x) \) by a finite set of exponentials as

\[
f(x) \approx C_1 e^{\gamma_1 x} + C_2 e^{\gamma_2 x} + \cdots + C_N e^{\gamma_N x}
\]

where the \( C_i \) and the \( \gamma_i \) are unknowns. It is assumed that \( f(x) \) is specified at \( N \) equally spaced points and with appropriate normalization of the \( x \) scale we can call these points \( x = 0, 1, 2, \ldots, N - 1 \).

The coding problem one is concerned with a finite field. The methods of solution, however, are identical.

It appears that this analogy will help us little in the decoding problem. On the contrary, it is possible that some of the techniques developed for decoding will have application to the curve fitting problem. For example, Forney's technique[2] for filling in erasures is applicable to the curve fitting problem where some of the exponentials are known.

Jack Keil Wolf  
Dept. of Elec. Engng.  
Polytechnic Institute of Brooklyn  
Brooklyn, N. Y.

References

To solve for $b_1, b_2$, we want 2 linear equations

But 3 samples only give 1 equation: $s_0 b_2 + s_1 b_1 = -b_0 s_2$

Can we use $s_0, s_1, s_2$ to get another independent equation?
Getting Away with Fewer Samples

To solve for $b_1, b_2$, we want 2 linear equations

- But 3 samples only give 1 equation: $s_0 b_2 + s_1 b_1 = -b_0 s_2$
- Can we use $s_0, s_1, s_2$ to get another independent equation?

For $\gamma_i = e^{\omega_i T \sqrt{-1}}$, consider the time-reversed conjugate $s_{-n}^*$

- The sequence $s_{-n}^*$ has the same frequency content as $s_n$:

$$s_{-n}^* = \sum_{i=1}^{2} a_i^* \left(e^{-n\omega_i T \sqrt{-1}}\right)^* = \sum_{i=1}^{2} a_i^* e^{n\omega_i T \sqrt{-1}}$$
To solve for $b_1, b_2$, we want 2 linear equations

- But 3 samples only give 1 equation: $s_0 b_2 + s_1 b_1 = -b_0 s_2$
- Can we use $s_0, s_1, s_2$ to get another independent equation?

For $\gamma_i = e^{\omega_i T \sqrt{-1}}$, consider the *time-reversed conjugate* $s_{-n}^*$

- The sequence $s_{-n}^*$ has the same frequency content as $s_n$:

$$s_{-n}^* = \sum_{i=1}^{2} a_i^* \left( e^{-n\omega_i T \sqrt{-1}} \right)^* = \sum_{i=1}^{2} a_i^* e^{n\omega_i T \sqrt{-1}}$$

Thus, we can use 3 samples to write

$$\begin{bmatrix} s_0 & s_1 \\ s_2^* & s_1^* \end{bmatrix} \begin{bmatrix} b_2 \\ b_1 \end{bmatrix} = -b_0 \begin{bmatrix} s_2 \\ s_0^* \end{bmatrix}$$

- If $a_1, a_2$ have uniform phase, matrix is almost surely invertible
Have We Cheated?

- Prony's method
  - Works for general complex frequencies like $\gamma_1 = 2e^{3\sqrt{-1}}$
  - Counting $\mathbb{R}$ dimensions: 8 parameters from 8 samples
Prony's method
- Works for general complex frequencies like $\gamma_1 = 2e^{3\sqrt{-1}}$
- Counting $\mathbb{R}$ dimensions: 8 parameters from 8 samples

New method based on time reversal and conjugacy
- Requires that $(\gamma_i^*)^{-1} = \gamma_i$ which implies $|\gamma_i| = 1$
- Counting $\mathbb{R}$ dimensions: 6 parameters from 6 samples
Have We Cheated?

- Prony’s method
  - Works for general complex frequencies like $\gamma_1 = 2e^{3\sqrt{-1}}$
  - Counting $\mathbb{R}$ dimensions: 8 parameters from 8 samples

- New method based on time reversal and conjugacy
  - Requires that $(\gamma_i^*)^{-1} = \gamma_i$ which implies $|\gamma_i| = 1$
  - Counting $\mathbb{R}$ dimensions: 6 parameters from 6 samples

- Yes, we have cheated!
  - We can find $t$ “sinusoids” using $\lceil 3t/2 \rceil$ samples
  - but not $t$ “complex exponentials”
  - The gain comes from reducing $\gamma_i$ from 2-dim to 1-dim
“Rediscovering Our Roots: Coding Theory and Reed-Solomon Codes”

Three Questions

- How many equally spaced samples are required to find the amplitude and frequency of the sum of $t$ sinusoids?

- How many errors can be corrected by a $[17, 8, 10]$ shortened Reed-Solomon code over the Galois field $\mathbb{F}_{256}$?

- How are these questions related?
“Rediscovering Our Roots: Coding Theory and Reed-Solomon Codes”

Three Questions

- How many equally spaced samples are required to find the amplitude and frequency of the sum of $t$ sinusoids? \[ \left\lceil \frac{3t}{2} \right\rceil \]

- How many errors can be corrected by a $[17, 8, 10]$ shortened Reed-Solomon code over the Galois field $\mathbb{F}_{256}$?

- How are these questions related?
“Rediscovering Our Roots: Coding Theory and Reed-Solomon Codes”

<table>
<thead>
<tr>
<th>Three Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>- How many equally spaced samples are required to find the amplitude and frequency of the sum of $t$ sinusoids? $\lceil \frac{3t}{2} \rceil$</td>
</tr>
<tr>
<td>- How many errors can be corrected by a $[17, 8, 10]$ shortened Reed-Solomon code over the Galois field $\mathbb{F}_{256}$? 6</td>
</tr>
<tr>
<td>- How are these questions related?</td>
</tr>
</tbody>
</table>
“Rediscovering Our Roots: Coding Theory and Reed-Solomon Codes”

Three Questions

- How many equally spaced samples are required to find the amplitude and frequency of the sum of \( t \) sinusoids? \( \left\lceil \frac{3t}{2} \right\rceil \)

- How many errors can be corrected by a \([17, 8, 10]\) shortened Reed-Solomon code over the Galois field \( \mathbb{F}_{256} \)? 6

- How are these questions related? via Jack’s 1967 paper
Let $F = \mathbb{F}_{q^2}$ and let $\alpha \in F$ be a primitive element

- Choose $\beta = \alpha^{q-1}$ and consider its conjugate $\beta^q$ (w.r.t. $\mathbb{F}_q$)
  - A little algebra shows $\beta^q = \alpha^{q^2-q} = \alpha^{1-q} = \beta^{-1}$

Since $\beta^{q+1} = \alpha^{(q-1)(q+1)} = \alpha^{q^2-1}$, we see $\beta$ has order $q + 1$

- Element $\beta$ gives $[q + 1, k, q + 2 - k]_F$ shortened RS code
- Syndrome sequence has time-reversal conjugate symmetry
- Decoding trick can succeed with $\left\lfloor \frac{2(n - k)}{3} \right\rfloor$ errors
First Interlude

- In 1998, I took Algebraic Coding Theory from Jack.
First Interlude

- In 1998, I took Algebraic Coding Theory from Jack
  - His use of simple examples made the class very accessible
  - That’s where I learned the decoding link to Prony’s method
In 1998, I took Algebraic Coding Theory from Jack

- His use of simple examples made the class very accessible
- That’s where I learned the decoding link to Prony’s method

In Dec. 1999, Jack was on my prelim exam committee
In 1998, I took Algebraic Coding Theory from Jack

- His use of simple examples made the class very accessible
- That’s where I learned the decoding link to Prony’s method

In Dec. 1999, Jack was on my prelim exam committee

- He asked the simple question:

  “Can we analyze typical-set decoding via weight enumerators?”
How can cellular systems with successive cancellation approach the equal-rate point of the multiple-access channel with equal-power users?
How can cellular systems with successive cancellation approach the equal-rate point of the multiple-access channel with equal-power users?

Asked (in spirit) at Qualcomm in 2003 by Roberto Padovani

- Initially, a small group (including Jack) met weekly with Roberto to discuss this
- Later, the group grew and started implementing pilot interference cancellation for CSM 6800
- When I left Qualcomm in 2005, PIC was working, the TIC system design was drafted, and the real work began: Jilei Hou, John Smee, Joseph Soriaga, and others built it
Interference cancellation viewed with skepticism at Qualcomm

- Viterbi’s book (Vembu paper) argues that gains are limited
- This negative view was controversial in the IT community
Interference cancellation viewed with skepticism at Qualcomm
- Viterbi’s book (Vembu paper) argues that gains are limited
- This negative view was controversial in the IT community

After Viterbi left Qualcomm, this view was revisited
- First task was to understand what gains are possible:
  “Capacity of Multi-antenna Gaussian Channels” by Telatar
Interference cancellation viewed with skepticism at Qualcomm

- Viterbi’s book (Vembu paper) argues that gains are limited
- This negative view was controversial in the IT community

After Viterbi left Qualcomm, this view was revisited

- First task was to understand what gains are possible:
  “Capacity of Multi-antenna Gaussian Channels” by Telatar
- Second task was to consider practical limitations:
  e.g., channel estimation and MMSE cancellation
Qualcomm EV-DO Reverse Link

- 1228800 chips/sec, 1 slot = 2048 chips, 1 slot = 1.67 ms
- Release 0 uses a single 16-slot packet for reverse link
  - Users staggered in 16 “phases” for hardware load balancing
- Release A uses Hybrid ARQ with four 4-slot sub-packets
  - Packets in 3 interlaced streams to allow time for ACK/NACK
  - Users staggered across 4 “phases” for hardware load balancing

(Figure courtesy of Soriaga et al., GLOBECOM 2006 [SHS06])
3-User Gaussian MAC w/successive interference cancellation
- Capacity achieved by an exponentially decaying power profile
- Also by equal-power staggered users (c.f., rate-splitting)

$$3 \log_2 \left( \frac{1+1}{1} \right) = \log_2 \left( \frac{1+7/3+7/3+7/3}{1+7/3+7/3} \right) + \log_2 \left( \frac{1+7/3+7/3}{1+7/3} \right) + \log_2 \left( \frac{1+7/3}{1} \right)$$

(Figure courtesy of Hou et al., IEEE Comm. Mag. 2006 [HSPT06])
Staggered Successive Interference Cancellation

Graphical Representation of Decoding

Uses single-user coding and decoding of packets
Cancellation occurs in receive buffer after decoding
Conceptually similar to D-BLAST and rate-splitting
Due to CRC, peeling similar to LDPC codes on the BEC
For uncoordinated users, called iterative collision resolution
Staggered Successive Interference Cancellation

Graphical Representation of Decoding

Uses single-user coding and decoding of packets
Cancellation occurs in receive buffer after decoding
Conceptually similar to D-BLAST and rate-splitting
Due to CRC, peeling similar to LDPC codes on the BEC
For uncoordinated users, called iterative collision resolution
Staggered Successive Interference Cancellation

Graphical Representation of Decoding

- Uses single-user coding and decoding of packets
- Cancellation occurs in receive buffer after decoding
- Conceptually similar to D-BLAST and rate-splitting
- Due to CRC, peeling similar to LDPC codes on the BEC
- For uncoordinated users, called iterative collision resolution
Staggered Successive Interference Cancellation

Graphical Representation of Decoding

- Uses single-user coding and decoding of packets
- Cancellation occurs in receive buffer after decoding
- Conceptually similar to D-BLAST and rate-splitting
- Due to CRC, peeling similar to LDPC codes on the BEC
- For uncoordinated users, called iterative collision resolution
Staggered Successive Interference Cancellation

Graphical Representation of Decoding

- Uses single-user coding and decoding of packets
- Cancellation occurs in receive buffer after decoding
- Conceptually similar to D-BLAST and rate-splitting
- Due to CRC, peeling similar to LDPC codes on the BEC
- For uncoordinated users, called iterative collision resolution
For the $A$-th user, the slot SINR achieved before IC from antenna combining with a weighting vector $w$ is $$\theta$$.

Performance Comparison

In [5] are straightforward as in the previous subsection.

Matrix of the residual signal. Note that this term is related to level as that of a receiver without IC, then any power receiver to maintain the same post-cancellation interference buffer traffic. In this case, if we constrain the IC-enabled cancellation of user $A$.

Modeled as a uniform random variable over the interval $[0, \pi)$.

Moreover, similar to equation (4), the change in SINR from a reduction in $N$ as compared to (8), $\beta_0$.

The SINR in (8) then simplifies to $\beta$.

Combining, we use $R_{0}$.

As a reminder, in the case of MRC, the weighting vector $h$.

For simulations we assume that all ATs continuously $\theta_1$.

Sector Throughput for Full Buffer Traffic

Figure 3. For systems loaded with 10-16 ATs per sector, we demonstrate that the original grade-of-service provided by IC does indeed increase the throughput for users at every $N_t$.

The cumulative gain as ATs can use higher T2P ratios (i.e., data rates, cancelled at the receiver translates directly into a throughput $\beta_{1}$).

IC systems with comparable interference levels are given in following the MAC protocol in [1]). Results for IC versus no-$$N_c$$.

EV-DO does not change significantly after IC. $\beta_{2}$.

In Figure 4 we illustrate the distribution among ATs of $\beta_{3}$.

Average sector throughput for the EV-DO reverse-link. $\beta_{4}$.

Distribution of throughput across ATs in the network. $\beta_{5}$.

<table>
<thead>
<tr>
<th>Number of ATs per Sector</th>
<th>AT Throughput (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>400</td>
</tr>
<tr>
<td>6</td>
<td>600</td>
</tr>
<tr>
<td>8</td>
<td>800</td>
</tr>
<tr>
<td>10</td>
<td>1000</td>
</tr>
<tr>
<td>12</td>
<td>1200</td>
</tr>
<tr>
<td>14</td>
<td>1400</td>
</tr>
<tr>
<td>16</td>
<td>1600</td>
</tr>
</tbody>
</table>

- IC gains approx. 70% for 4 antennas and 50% for 2 antennas

(Figure courtesy of Soriaga et al., GLOBECOM 2006 [SHS06])
Performance in practice was *surprisingly good*
Performance in practice was surprisingly good

Key role of staggered users was only recognized later

- Fortunately, staggering was there for hardware load balancing
- Adding/removing staggering would have been impossible

Cancellation project at Qualcomm started in 2003, before the magic of spatial coupling was known

Lefty Gomez said famously, “I’d rather be lucky than good.”

Definitely true for innovation and backwards compatibility!
Historical Perspective

- Performance in practice was **surprisingly good**
- **Key role of staggered users** was only recognized later
  - Fortunately, staggering was there for **hardware load balancing**
  - Adding/removing staggering would have been impossible
- Cancellation project at Qualcomm started in 2003, before the magic of spatial coupling was known
Performance in practice was **surprisingly good**

**Key role of staggered users** was only recognized later
- Fortunately, staggering was there for **hardware load balancing**
- Adding/removing staggering would have been impossible

Cancellation project at Qualcomm started in 2003, before the magic of spatial coupling was known

Lefty Gomez said famously, “I’d rather be lucky than good.”
- Definitely true for innovation and backwards compatibility!
What is Spatial Coupling?

Spatially-Coupled Factor Graphs

Variable nodes have a natural global orientation

Boundaries help variables to be recovered in an ordered fashion
What is Spatial Coupling?

- **Spatially-Coupled Factor Graphs**
  - Variable nodes have a natural global orientation
  - Boundaries help variables to be recovered in an ordered fashion
Spatially-Coupled Sudoku Example
Spatially-Coupled Sudoku Example

```
6 5 4
7 3 9
8 1 2
1 3 5 9 4 7
2 9 4 3 8 6
8 7 6 1 2 5 9
2 5 6 3
3 8
1 4 6 2
9 2
3 1
4 7
```

Spatially-Coupled Sudoku Example
Spatially-Coupled Sudoku Example
Spatially-Coupled Sudoku Example

```

```

```

```

```

```

```
Spatially-Coupled Sudoku Example
Second Interlude

- Insight missed: From the staggered MAC to Spatial Coupling
Insight missed: From the staggered MAC to Spatial Coupling
- There were signs that the staggered structure was important
- But, I didn’t appreciate spatial coupling until the KRU proof
Insight missed: From the staggered MAC to Spatial Coupling

- There were signs that the staggered structure was important
- But, I didn’t appreciate spatial coupling until the KRU proof

If you want more details on this topic, google for Jack Wolf’s plenary slides from ISSSTA 2006
Insight missed: From the staggered MAC to Spatial Coupling
   - There were signs that the staggered structure was important
   - But, I didn’t appreciate spatial coupling until the KRU proof

If you want more details on this topic, google for Jack Wolf’s plenary slides from ISSSTA 2006

When I was ready to leave Qualcomm, I sought Jack’s advice
   - I was deciding between a post-doc and few different startups
Third Question

Can Reed-Muller codes achieve capacity?
Can Reed-Muller codes achieve capacity?

- Discussed by Kasami, Lin, and Peterson in late 1960s
- Mentioned by Lin in ITW 1993 talk
- Conjectured for BEC in a paper by Dumer and Farrell in 1994
- Costello and Forney in 2007: “The road to channel capacity”
- Asked by Arikan in 2010 after the invention of polar codes
- Shown for rate $\to 0/1$ by Abbe, Shpilka, Widgerson in 2014
Coding for the BEC: Problem Setup

- Length-$N$ binary linear code $C$

![Diagram showing message flow through encoder and BEC(p)]

**Uniform codeword:** $X = (X_0, \ldots, X_{N-1}) \in \{0, 1\}^N$

**Bernoulli-$p$ erasures:** $Z = (Z_0, \ldots, Z_{N-1}) \in \{0, 1\}^N$

**BEC($p$) observation of $X$:**

$$Y_i = \begin{cases} X_i & \text{if } Z_i = 0 \\ ? & \text{if } Z_i = 1 \end{cases}$$
Example: \( y = (1 \ 0 \ ? \ ? \ ? \ ?) \) with

\[
C = \begin{cases}
(0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0) \\
(0 \ 0 \ 1 \ 1 \ 1 \ 0) \\
(0 \ 1 \ 0 \ 0 \ 1 \ 1) \\
(0 \ 1 \ 1 \ 1 \ 0 \ 1) \\
(1 \ 0 \ 0 \ 1 \ 0 \ 1) \\
(1 \ 0 \ 1 \ 0 \ 1 \ 1) \\
(1 \ 1 \ 0 \ 1 \ 1 \ 0) \\
(1 \ 1 \ 1 \ 0 \ 0 \ 0)
\end{cases}
\]
Example: \( y = (1 \ 0 \ ? \ ? \ ? \ ?) \) with

\[
C = \{ (0 \ 0 \ 0 \ 0 \ 0 \ 0) , 
(0 \ 0 \ 1 \ 1 \ 1 \ 0) , 
(0 \ 1 \ 0 \ 0 \ 1 \ 1) , 
(0 \ 1 \ 1 \ 1 \ 0 \ 1) , 
(1 \ 0 \ 0 \ 1 \ 0 \ 1) , 
(1 \ 0 \ 1 \ 0 \ 1 \ 1) , 
(1 \ 1 \ 0 \ 1 \ 1 \ 0) , 
(1 \ 1 \ 1 \ 0 \ 0 \ 0) \}
\]

Consistent Codewords: \( C(y) \triangleq \{ x \in C \mid x_i = y_i \text{ when } y_i \neq ? \} \)
Example: \( \underline{y} = (1 \ 0 \ ? \ ? \ ? \ ?) \) with

\[
C = \begin{cases}
(0 \ 0 \ 0 \ 0 \ 0 \ 0)
, \\
(0 \ 0 \ 1 \ 1 \ 1 \ 0)
, \\
(0 \ 1 \ 0 \ 0 \ 1 \ 1)
, \\
(0 \ 1 \ 1 \ 1 \ 0 \ 1)
, \\
(1 \ 0 \ 0 \ 1 \ 0 \ 1)
, \\
(1 \ 0 \ 1 \ 0 \ 1 \ 1)
, \\
(1 \ 1 \ 0 \ 1 \ 1 \ 0)
, \\
(1 \ 1 \ 1 \ 0 \ 0 \ 0)
\end{cases}
\]

Consistent Codewords: \( C(\underline{y}) \triangleq \{ x \in C \mid x_i = y_i \text{ when } y_i \neq ? \} \)

Decoder declares block erasure iff \( |C(\underline{y})| > 1 \)
Example: $y = (1 \ 0 \ ? \ ? \ ? \ ?)$ with

$$C = \begin{cases} 
(0 \ 0 \ 0 \ 0 \ 0 \ 0) \\
(0 \ 0 \ 1 \ 1 \ 1 \ 0) \\
(0 \ 1 \ 0 \ 0 \ 1 \ 1) \\
(0 \ 1 \ 1 \ 1 \ 0 \ 1) \\
(1 \ 0 \ 0 \ 1 \ 0 \ 1) \\
(1 \ 0 \ 1 \ 0 \ 1 \ 1) \\
(1 \ 1 \ 0 \ 1 \ 1 \ 0) \\
(1 \ 1 \ 1 \ 0 \ 0 \ 0) 
\end{cases}$$

Consistent Codewords: $C(y) \triangleq \{x \in C \mid x_i = y_i \text{ when } y_i \neq ?\}$

Decoder declares block erasure iff $|C(y)| > 1$

Bit-$i$ recoverable iff $x_i = x'_i$ for all $x, x' \in C(y)$

- If $x_i$ unrecoverable, half the consistent codewords have $x_i = 1$
- Since they are equiprobable, $H(X_i \mid Y = y)$ is either 0 or 1
The MAP erasure rate of bit-\(i\), \(P_{b,i}(p)\), satisfies
\[
P_{b,i}(p) \triangleq H(X_i|Y) = P(Y_i = ?) H(X_i|Y, Y_i = ?),
\]
where \(h_i(p)\) is called the MAP EXIT function of bit-\(i\).
The MAP erasure rate of bit-$i$, $P_{b,i}(p)$, satisfies

$$P_{b,i}(p) \triangleq H(X_i|Y) = \mathbb{P}(Y_i = ?) \cdot H(X_i|Y, Y_i = ?),$$

where $h_i(p)$ is called the MAP EXIT function of bit-$i$

<table>
<thead>
<tr>
<th>MAP erasure $P_b(p)$</th>
<th>MAP EXIT $h(p)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_b(p) \triangleq \frac{1}{N} \sum_{i=0}^{N-1} P_{b,i}(p) = ph(p)$</td>
<td>$h(p) \triangleq \frac{1}{N} \sum_{i=0}^{N-1} h_i(p)$</td>
</tr>
</tbody>
</table>
The MAP erasure rate of bit-$i$, $P_{b,i}(p)$, satisfies
\[
P_{b,i}(p) \triangleq H(X_i | Y) = \Pr(Y_i = ?) H(X_i | Y, Y_i = ?),
\]
where $h_i(p)$ is called the MAP EXIT function of bit-$i$.

<table>
<thead>
<tr>
<th>MAP erasure $P_b(p)$</th>
<th>MAP EXIT $h(p)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_b(p) \triangleq \frac{1}{N} \sum_{i=0}^{N-1} P_{b,i}(p) = ph(p)$</td>
<td>$h(p) \triangleq \frac{1}{N} \sum_{i=0}^{N-1} h_i(p)$</td>
</tr>
</tbody>
</table>

Achieving Capacity $C = 1 - p$ under Bit-MAP Decoding

- Code sequence $C^{(n)}$ with rates $R_n \rightarrow R$
  
  i. For any $p < 1 - R$, the MAP erasure rate $P_{b}^{(n)}(p) \rightarrow 0$
  
  ii. For any $p < 1 - R$, the MAP EXIT function $h^{(n)}(p) \rightarrow 0$
EXtrinsic Information Transfer (EXIT) Curves

In 1999, ten Brink defined to explain iterative decoding

\[ \text{Area} = \frac{K}{N} \]

\[ h(p) \]

\[ \int_0^1 h(p) \, dp = R \text{ (code rate)} \]

If \( C(n) \) capacity achieving, it implies \( h(n) \rightarrow \text{step function} \)
EXtrinsic Information Transfer (EXIT) Curves

- In 1999, ten Brink defined $h(p)$ to explain iterative decoding.
- EXIT Area Theorem [ABK04]

$$\int_{0}^{1} h(p) \, dp = R \quad \text{(code rate)}$$

- Conservation principle by Ashikhmin, ten Brink, and Kramer.
In 1999, ten Brink defined \( h(p) \) to explain iterative decoding.

**EXIT Area Theorem [ABK04]**

\[
\int_{0}^{1} h(p) \, dp = R \quad \text{(code rate)}
\]

- conservation principle by Ashikhmin, ten Brink, and Kramer
- If \( C^{(n)} \) capacity achieving, it implies \( h^{(n)}(p) \to \) step function
The MAP EXIT Curve of a Capacity-Achieving Code

Area Theorem implies sharp transition iff capacity achieving
The MAP EXIT Curve of a Capacity-Achieving Code

Area Theorem implies sharp transition iff capacity achieving
The MAP EXIT Curve of a Capacity-Achieving Code

Area Theorem implies sharp transition iff capacity achieving
The MAP EXIT Curve of a Capacity-Achieving Code

Area Theorem implies sharp transition iff capacity achieving
Symmetry alone is sufficient to achieve capacity!!! [KKMPSU16]

Theorem 1: Let \( \{C_n\} \) be a sequence of \( \mathbb{F}_q \)-linear codes with

- blocklengths \( N_n \to \infty \) and rates \( R_n \to R \in (0,1) \), where
- the permutation group of each \( C_n \) is doubly transitive.

Then, \( \{C_n\} \) achieves capacity on the \( q \)-ary erasure channel (QEC) under symbol-MAP decoding.
Symmetry alone is sufficient to achieve capacity!!! [KKMPSU16]

**Theorem 1:** Let \( \{C_n\} \) be a sequence of \( \mathbb{F}_q \)-linear codes with
- blocklengths \( N_n \to \infty \) and rates \( R_n \to R \in (0,1) \), where
- the permutation group of each \( C_n \) is **doubly transitive**.

Then, \( \{C_n\} \) **achieves capacity** on the \( q \)-ary erasure channel (QEC) under symbol-MAP decoding.

- Previous work to show RM codes achieve capacity
  - Many approaches tried based on the code / weight distribution
Symmetry alone is sufficient to achieve capacity!!! [KKMPSU16]

Theorem 1: Let \( \{C_n\} \) be a sequence of \( \mathbb{F}_q \)-linear codes with
- blocklengths \( N_n \to \infty \) and rates \( R_n \to R \in (0, 1) \), where
- the permutation group of each \( C_n \) is doubly transitive.

Then, \( \{C_n\} \) achieves capacity on the \( q \)-ary erasure channel (QEC) under symbol-MAP decoding.

- Previous work to show RM codes achieve capacity
  - Many approaches tried based on the code / weight distribution
- Only known proof works for a much wider class!
Hamming (8,4) Code
(columns are codewords)

Permutation Group $G$
Code Symmetry in Pictures (1)

Hamming $(8,4)$ Code
(columns are codewords)

Permutation Group $\mathcal{G}$
Hamming (8,4) Code
(columns are codewords)

Permutation Group $G$
Code Symmetry in Pictures (1)

Hamming (8,4) Code
(columns are codewords)

Permutation Group \( G \)
For a set $\mathcal{A} \subseteq \mathcal{X}^N$ of vectors

- Permutations $\pi \in S_N$ act on $\mathcal{X}^N$ via: $b = \pi(a) \iff b_{\pi(i)} = a_i$
- The permutation group $\mathcal{G}$ of $\mathcal{A}$ is defined to be

$$
\mathcal{G} \triangleq \{ \pi \in S_N | \pi(a) \in \mathcal{A} \ \forall \ a \in \mathcal{A} \}
$$
For a set $\mathcal{A} \subseteq \mathcal{X}^N$ of vectors

- Permutations $\pi \in S_N$ act on $\mathcal{X}^N$ via: $b = \pi(a) \iff b_{\pi(i)} = a_i$
- The permutation group $G$ of $\mathcal{A}$ is defined to be
  \[ G \triangleq \{ \pi \in S_N \mid \pi(a) \in \mathcal{A} \forall a \in \mathcal{A} \} \]

Permutation group of $\mathcal{A}$

\[ \mathcal{A} = \begin{cases} 
(0 0 0 0) \\
(0 0 1 1) \\
(1 1 0 0) \\
(1 1 1 1) 
\end{cases} \begin{array}{c}
\implies \\
G = \begin{cases} 
(0 1 2 3) & (0 1 2 3) \\
(0 1 2 3) & (0 1 2 3) \\
(0 1 2 3) & (0 1 2 3) \\
(0 1 2 3) & (0 1 2 3) \\
(2 3 0 1) & (3 2 0 1) \\
(2 3 1 0) & (3 2 1 0) 
\end{cases} \end{array} \]
The Permutation Group of a Set of Vectors

- For a set $\mathcal{A} \subseteq \mathcal{X}^N$ of vectors
  - Permutations $\pi \in S_N$ act on $\mathcal{X}^N$ via: $b = \pi(a) \iff b_{\pi(i)} = a_i$
  - The permutation group $G$ of $\mathcal{A}$ is defined to be
    $$G \triangleq \{ \pi \in S_N \mid \pi(a) \in \mathcal{A} \ \forall \ a \in \mathcal{A} \}$$
  - A permutation group $G$ is transitive if, for all $i, j \in \mathbb{Z}_N$, there exists $\pi \in G$ such that $\pi(i) = j$

- Permutation group of $\mathcal{A}$ is transitive

\[
\mathcal{A} = \left\{ \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \right\} \quad \implies \quad G = \left\{ \begin{pmatrix} 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 3 & 2 \\ 0 & 1 & 2 & 3 \\ 2 & 3 & 0 & 1 \\ 0 & 1 & 2 & 3 \\ 2 & 3 & 1 & 0 \end{pmatrix} \right\}
\]
The Permutation Group of a Set of Vectors

- For a set $\mathcal{A} \subseteq \mathcal{X}^N$ of vectors
  - Permutations $\pi \in S_N$ act on $\mathcal{X}^N$ via: $b = \pi(a) \iff b_{\pi(i)} = a_i$
  - The permutation group $G$ of $\mathcal{A}$ is defined to be
    $$G \triangleq \{ \pi \in S_N \mid \pi(a) \in \mathcal{A} \ \forall \ a \in \mathcal{A} \}$$
  - A permutation group $G$ is transitive if, for all $i, j \in \mathbb{Z}_N$, there exists $\pi \in G$ such that $\pi(i) = j$
  - $G$ is doubly transitive if, for any $i, j, k, l \in \mathbb{Z}_N$ with $i \neq j$ and $k \neq l$, there exists $\pi \in G$ such that $\pi(i) = k$ and $\pi(j) = l$.

- Permutation group of $\mathcal{A}$ is transitive but not doubly transitive

\[
\mathcal{A} = \left\{ \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \right\} \quad \Longrightarrow \quad G = \left\{ \begin{pmatrix} 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 2 & 3 \\ 0 & 1 & 3 & 2 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 3 & 2 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 2 & 3 \\ 2 & 3 & 0 & 1 \\ 2 & 3 & 0 & 1 \\ 2 & 3 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 2 & 3 \\ 2 & 3 & 1 & 0 \\ 2 & 3 & 1 & 0 \\ 2 & 3 & 1 & 0 \end{pmatrix} \right\}
\]
Code Symmetry in Pictures (2)

Hamming (8,4) Code
(columns are codewords)

Permutation Group $G$
Hamming (8,4) Code
(columns are codewords)

Permutation Group \( \mathcal{G} \)
Hamming (8,4) Code
(columns are codewords)

Permutation Group $\mathcal{G}$
Hamming (8,4) Code
(columns are codewords)

Permutation Group $G$
For linear codes, the recovery of $X_i$ from $Y_{\sim i} = y_{\sim i}$

- is independent of the transmitted codeword $X$
- only depends on erasure indicator $z_i = 1\{?\}(y_i)$
- is a zero-one boolean function of $z_{\sim i}$
For linear codes, the recovery of $X_i$ from $Y_{\sim i} = y_{\sim i}$

- is independent of the transmitted codeword $X$
- only depends on erasure indicator $z_i = 1\{?\}(y_i)$
- is a zero-one boolean function of $z_{\sim i}$

Recovery indicator of bit-$i$ is a monotone boolean function

$$H(X_i|Y_{\sim i} = y_{\sim i}, Z_{\sim i} = z_{\sim i}) = f_i(z_{\sim i}) \in \{0, 1\},$$

where $z_{\sim i} \triangleq (z_0, \ldots, z_{i-1}, z_{i+1}, \ldots, z_{N-1})$
For linear codes, the recovery of $X_i$ from $Y_{\sim i} = y_{\sim i}$
- is independent of the transmitted codeword $X$
- only depends on erasure indicator $z_i = 1_{\{?\}}(y_i)$
- is a zero-one boolean function of $z_{\sim i}$

Recovery indicator of bit-$i$ is a monotone boolean function

$$H(X_i | Y_{\sim i} = y_{\sim i}, Z_{\sim i} = z_{\sim i}) = f_i(z_{\sim i}) \in \{0, 1\},$$
where $z_{\sim i} \triangleq (z_0, \ldots, z_{i-1}, z_{i+1}, \ldots, z_{N-1})$

Bit-$i$ EXIT function equals the expectation of $f_i(z_{\sim i})$

$$h_i(p) \triangleq H(X_i | Y, Y_i = ?) = \mathbb{E}[f_i(Z_{\sim i})]$$
If the permutation group of a code is transitive, then

\[ h_i(p) = h(p), \quad i \in \{1, 2, \ldots, N\} \]

Thus, EXIT Area Theorem implies

\[ \int_0^1 h_i(p) \, dp = R \]
If the permutation group of a code is transitive, then
\[ h_i(p) = h(p), \quad i \in \{1, 2, \ldots, N\} \]

Thus, EXIT Area Theorem implies
\[ \int_0^1 h_i(p) \, dp = R \]

If the permutation group is doubly transitive, then
\[ f_i(z_1, z_2, \ldots, z_{N-1}) = f_i(z_{\pi(1)}, z_{\pi(2)}, \ldots, z_{\pi(N-1)}) \quad \forall \pi \in G_i \]

where \( G_i \) is a transitive group of permutations.
If the permutation group of a code is **transitive**, then
\[ h_i(p) = h(p), \quad i \in \{1, 2, \ldots, N\} \]

Thus, EXIT Area Theorem implies
\[ \int_0^1 h_i(p) \, dp = R \]

If the permutation group is **doubly transitive**, then
\[ f_i(z_1, z_2, \ldots, z_{N-1}) = f_i(z_{\pi(1)}, z_{\pi(2)}, \ldots, z_{\pi(N-1)}) \quad \forall \pi \in G_i \]

where \( G_i \) is a **transitive group of permutations**

Thus, \( f_i \) is a **symmetric monotone boolean function**
EVERY MONOTONE GRAPH PROPERTY HAS A SHARP THRESHOLD

EHUD FRIEDGUT AND GIL KALAI

(Communicated by Jeffry N. Kahn)

Abstract. In their seminal work which initiated random graph theory Erdős and Rényi discovered that many graph properties have sharp thresholds as the number of vertices tends to infinity. We prove a conjecture of Linial that every monotone graph property has a sharp threshold. This follows from the following theorem.

Let $V_n(p) = \{0,1\}^n$ denote the Hamming space endowed with the probability measure $\mu_p$ defined by $\mu_p(\epsilon_1, \epsilon_2, \ldots, \epsilon_n) = p^k \cdot (1-p)^{n-k}$, where $k = \epsilon_1 + \epsilon_2 + \cdots + \epsilon_n$. Let $A$ be a monotone subset of $V_n$. We say that $A$ is symmetric if there is a transitive permutation group $\Gamma$ on $\{1,2,\ldots,n\}$ such that $A$ is invariant under $\Gamma$.

Theorem. For every symmetric monotone $A$, if $\mu_p(A) > \epsilon$ then $\mu_q(A) > 1-\epsilon$ for $q = p + c_1 \log(1/2\epsilon)/\log n$. ($c_1$ is an absolute constant.)

For any $\epsilon > 0$, the EXIT function $h(p)$ transitions from $\epsilon$ to $1-\epsilon$ as $p$ changes by $O(1/\log(N))$.
Beyond Double Transitivity

The previous result also holds with less symmetry [KCP16].

**Theorem 2:** Let \( \{C_n\} \) be a sequence of codes over \( \mathbb{F}_q \) with

- transitive perm. groups \( \mathcal{G}^{(n)} \) and rates \( R_n \rightarrow R \in (0, 1) \),
- where the size of the smallest orbit of \( \mathcal{G}_0^{(n)} \) is unbounded

Then, \( \{C_n\} \) **achieves capacity** on the \( q \)-ary erasure channel (QEC) under symbol-MAP decoding.
Example: Product Codes

<table>
<thead>
<tr>
<th></th>
<th>(0, 0)</th>
<th>(0, 1)</th>
<th>(0, 2)</th>
<th>(0, 3)</th>
<th>(0, 4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1, 0)</td>
<td>(1, 1)</td>
<td>(1, 2)</td>
<td>(1, 3)</td>
<td>(1, 4)</td>
<td></td>
</tr>
<tr>
<td>(2, 0)</td>
<td>(2, 1)</td>
<td>(2, 2)</td>
<td>(2, 3)</td>
<td>(2, 4)</td>
<td></td>
</tr>
</tbody>
</table>

Setup
- \( M \times N \) product code with doubly transitive component codes
- \( \mathcal{G}_0 \) permutations fix \((0, 0)\) but transitive on other rows/cols
- Each orbit has a own color, smallest \( \geq \min\{M - 1, N - 1\} \)
Multi-Dimensional Product Codes

- Codeword of \textit{m-dimensional product code} is array of $n^m$ bits
- Axis-aligned 1D subarrays are codewords of $(n, k)$ linear code
- Overall Rate: $R_m = (k/n)^m$
Multi-Dimensional Product Codes

- Codeword of \textit{m-dimensional product code} is array of \(n^m\) bits
- Axis-aligned 1D subarrays are codewords of \((n, k)\) linear code
- Overall Rate: \(R_m = \left(\frac{k}{n}\right)^m\)
- If \((n, k)\) code is \((n, n-1)\) single parity-check (SPC) code
  - Rate \(R_m = \left(\frac{n-1}{n}\right)^m\) satisfies \(\lim_{n \to \infty} R_n = e^{-1}\)
High-Dimensional SPC Product Codes

MAP Performance for the $n$-D Product of $(n, n-1)$ SPC Codes

- $n=3$, $R=0.30$, $N=3^3=27$
- $n=4$, $R=0.32$, $N=4^4=256$
- $n=5$, $R=0.33$, $N=5^5=3125$
- $n=6$, $R=0.34$, $N=6^6=46656$

- Capacity for $R=0.34$
Insight Achieved!

Simple question only involved RM codes
Insight Achieved!

- Simple question only involved RM codes
  - But its solution provided much more
  - Interplay between symmetry and capacity has long history
Insight Achieved!

- Simple question only involved RM codes
  - But its solution provided much more
  - Interplay between symmetry and capacity has long history

- Can this be extended beyond the BEC?
  - Either way, I believe the answer will provide new insight
Summary

Three simple questions and their (partial) answers
Summary

- Three simple questions and their (partial) answers
  - What insights did they provide?
  - What is the right simple question for your research?

Preparing this talk

Great opportunity to reflect on Jack Wolf's impact on my life
Three simple questions and their (partial) answers

- What insights did they provide?
- What is the right simple question for your research?

Preparing this talk

- Great opportunity to reflect on Jack Wolf’s impact on my life
Thank You!
For a group $G$ acting on $\mathbb{Z}_N$, the orbit of $i \in \mathbb{Z}_N$ is

$$O_i \triangleq \{j \in \mathbb{Z}_N \mid \exists \pi \in G, \pi(i) = j\}.$$ 

Note that for $i, j \in \mathbb{Z}_N$, either $O_i = O_j$ or $O_i \cap O_j = \emptyset$. Thus, the set of orbits, $\{O_\ell\}$, partitions the set $\mathbb{Z}_N$. 
Orbits and Stabilizers

- For a group $G$ acting on $\mathbb{Z}_N$, the orbit of $i \in \mathbb{Z}_N$ is
  \[ O_i \triangleq \{ j \in \mathbb{Z}_N \mid \exists \pi \in G, \pi(i) = j \} . \]

- Note that for $i, j \in \mathbb{Z}_N$, either $O_i = O_j$ or $O_i \cap O_j = \emptyset$. Thus, the set of orbits, $\{O_\ell\}$, partitions the set $\mathbb{Z}_N$.

- For group $G$ acting on $\mathbb{Z}_N$, the stabilizer subgroup of $i \in \mathbb{Z}_N$ is
  \[ G_i \triangleq \{ \pi \in G \mid \pi(i) = i \} . \]
For a group $G$ acting on $\mathbb{Z}_N$, the orbit of $i \in \mathbb{Z}_N$ is

$$O_i \triangleq \{ j \in \mathbb{Z}_N \mid \exists \pi \in G, \pi(i) = j \}.$$ 

Note that for $i, j \in \mathbb{Z}_N$, either $O_i = O_j$ or $O_i \cap O_j = \emptyset$. Thus, the set of orbits, $\{O_\ell\}$, partitions the set $\mathbb{Z}_N$.

For group $G$ acting on $\mathbb{Z}_N$, the stabilizer subgroup of $i \in \mathbb{Z}_N$ is

$$G_i \triangleq \{ \pi \in G \mid \pi(i) = i \}.$$ 

For $G_i$, let the size of the smallest non-trivial orbit be

$$O_{\text{min}}(G_i) \triangleq \min_{j \in \mathbb{Z}_N \setminus \{i\}} |O_j(G_i)|.$$
Multi-Dimensional Product Codes (2)

- For $m$-D product of $(n, k)$ codes, we index bits by $v \in \mathbb{Z}_n^m$
  - If $(n, k)$ code transitive, then product code transitive
  - We permute code bits by *permuting code dimensions*
  - Code is preserved because component codes identical

Consider size of the smallest non-trivial orbit $O_{\min}(G_0)$

Assume bit 0 associated with index $(0, \ldots, 0)$
Then, permuting code dimensions maps bit 0 to bit 0
Permuting code dimensions induces orbits defined by the empirical distribution of the index vector
Minimal orbits have one non-zero entry and $O_{\min}(G_0) \geq m^n$

$m$-D product of $(n, n-1)$ SPCs achieves capacity as $n \to \infty$

Some prior work on SPC product codes [CTB95, RG01]
For $m$-D product of $(n, k)$ codes, we index bits by $\mathbf{v} \in \mathbb{Z}_n^m$

- If $(n, k)$ code transitive, then product code transitive
- We permute code bits by permuting code dimensions
- Code is preserved because component codes identical

Consider size of the smallest non-trivial orbit $O_{\text{min}}(G_0)$

- Assume bit 0 associated with index $(0, \ldots, 0)$
- Then, permuting code dimensions maps bit 0 to bit 0
- Permuting code dimensions induces orbits defined by the empirical distribution of the index vector
- Minimal orbits have one non-zero entry and $O_{\text{min}}(G_0) \geq m$
For $m$-D product of $(n, k)$ codes, we index bits by $\mathbf{v} \in \mathbb{Z}_n^m$

- If $(n, k)$ code transitive, then product code transitive
- We permute code bits by permuting code dimensions
- Code is preserved because component codes identical

Consider size of the smallest non-trivial orbit $O_{\text{min}}(G_0)$

- Assume bit 0 associated with index $(0, \ldots, 0)$
- Then, permuting code dimensions maps bit 0 to bit 0
- Permuting code dimensions induces orbits defined by the empirical distribution of the index vector
- Minimal orbits have one non-zero entry and $O_{\text{min}}(G_0) \geq m$

$n$-D product of $(n, n-1)$ SPCs achieves capacity as $n \to \infty$!
- Some prior work on SPC product codes [CTB95, RG01]
The influence of variables in product spaces.

[CTB95] Giuseppe Caire, Giorgio Taricco, Gérard Battail.
Weight distribution and performance of the iterated product of single-parity-check codes.

[FK96] Ehud Friedgut, Gil Kalai.
Every monotone graph property has a sharp threshold.

[KCP16] Santhosh Kumar, Robert Calderbank, Henry D. Pfister.
Beyond double transitivity: Capacity-achieving cyclic codes on erasure channels.
The influence of variables on boolean functions.

Reed-Muller codes achieve capacity on erasure channels.

[RG01] David M Rankin, T Aaron Gulliver.
Single parity check product codes.