Variations of Rank Modulation for Flash Memories

Zhiying Wang

Joint work with

Anxiao(Andrew) Jiang
Jehoshua Bruck
Flash Memory

Control Gate
Floating Gate
Source
Drain
Substrate

Block erasure
Flash Memory

- Iterative writing
- Writing speed
Flash Memory

- Charge leakage
Flash Memory

- Charge leakage
- Data reliability
Rank Modulation

- Writing speed
- Data reliability

Decoding Rank Modulation

- Maximal level identification; maximal level removal
- For m cells, need (m-1) iterations
- Capacity: \( \log(5!)/5 \)
Variation I: Partial Rank Modulation

- Reduce the complexity of decoding
- Only $k$ iterations ($k=2$)
- Capacity: $\log(5*4)/5$
- $k$-partial rank modulation ($k$-permutations): top $k$ cells; total of $m$ cells

(5 2)

(5 2 | 4 1 3)
Partial Rank Modulation: Updating

- Push-to-the-top operation

1 2 3 4 5

5 2 4 1 3

1 5 2 4 3

1 5 2 4 3
Gray Code for Partial Rank Modulation

- A cycle of all k-permutations
- Transition: push-to-the-top
Gray Code for Rank Modulation*

Gray Code for Rank Modulation*

$m = k + 1 = 3$

---

Gray Code as a Counter

- Counter from 0 to 5

\[ m = k + 1 = 3 \]

Absolute Value Counter

Rank Modulation Counter
Gray Code as a Counter

- Counter from 0 to 5

\[ m = k + 1 = 3 \]

Absolute Value Counter

Rank Modulation Counter
Gray Code as a Counter

- Counter from 0 to 5

\[ m = k + 1 = 3 \]
Gray Code as a Counter

- Counter from 0 to 5

\[ m = k + 1 = 3 \]

Absolute Value Counter

Rank Modulation Counter
Gray Code as a Counter

- Counter from 0 to 5

\[ m = k + 1 = 3 \]

![Diagram showing Gray code counter]

Absolute Value Counter

Rank Modulation Counter
Gray Code as a Counter

- Counter from 0 to 5

\[ m = k + 1 = 3 \]

Absolute Value Counter

Rank Modulation Counter
Gray Code as a Counter

- Counter from 0 to 5

\[ m = k + 1 = 3 \]
Universal Cycles for k-permutations*

- A sequence \((u_1, u_2, \ldots, u_N)\), \(N = m!/(m-k)!\)
- Each k-permutation is represented by exactly one \((u_{i+1}, u_{i+2}, \ldots, u_{i+k})\)

\[m = k + 1 = 3\]

Universal cycle

\[2 \quad 1 \quad 2 \quad 3 \quad 1 \quad 3\]

\[21 \quad 12 \quad 23 \quad 31 \quad 13 \quad 32\]

B.W. Jackson, “Universal cycles for k-subsets and k-permutations”, 1993
F. Ruskey and A. Williams, “An explicit universal cycle for the \((n - 1)\)-permutations of an n-set”, 2009
Universal Cycles for k-permutations

\[ m = k + 1 = 3 \]

\[ 321 \quad 312 \quad 123 \quad 231 \quad 213 \quad 132 \]
Example: $m=4$, $k=2$
Example: $m=4$, $k=2$
Construction of Gray Code

Theorem:

- There exists a Gray code for k-partial rank modulation, for all $0 < k < m$.
- For a given permutation, the next push-to-the-top operation can be decided in time $O(k \log(k))$ on average.
Partial Rank Modulation: Summary

- Reduce decoding complexity
- $k$-permutations out of $m$ cells
- Gray code of partial rank modulation

- Given permutation size $m$, what can we do if we have more than $m$ cell levels?
Variation II: Bounded Rank Modulation

- Permutation size $m$
- Maximum cell level $D > m$
Bounded Rank Modulation: Example

- 8 cells; cell levels \{1,2,3,4,5,6\};
  permutation size 4;

\[ \begin{align*}
1 & \quad 2 & \quad 3 & \quad 4 & \quad 5 & \quad 6 & \quad 2 & \quad 3 \\
\end{align*} \rightarrow 4321, 2143 \]

\[ \begin{align*}
1 & \quad 2 & \quad 3 & \quad 4 & \quad 3 & \quad 4 & \quad 1 & \quad 2 \\
\end{align*} \rightarrow 4321, 2143 \]

Only 4 levels needed

- Can we do better?

\[ \begin{align*}
1 & \quad 2 & \quad 3 & \quad 4 & \quad 5 & \quad 6 & \quad 2 & \quad 3 \\
\end{align*} \rightarrow 4321, 4321, 2143 \]

overlap

All 6 levels needed
Computing the Capacity

- Cell levels \{1,2,3,4,5,6,7,8,9,10\} ; 8 cells; permutation size 4

\[
cap = \log(4!^2) / 8 = 1.146
\]

\[
cap = \log(4! \times (4 \times 3)^2) / 8 = 1.469
\]

\[
cap = \log(4! \times 4^4) / 8 = 1.573
\]
Overlap Increases Capacity

- Theorem: $\text{cap}(m,D,v=1) > \text{cap}(m,D,v=0)$, for given $m > 1$, $D > m + 1$

  $m$: permutation size
  $D$: maximum cell level
  $v$: overlap
Encoding

- Max level $D=4$, size $m=2$, overlap $v=1$
- $001\ 100\ 000\ 010\ ...$ message
- $\rightarrow 1001\ 0100\ 1110\ 1010\ ...$ permutation
- $\rightarrow 14314\ 1431\ 2341\ 4142\ ...$ cell level

$(21)=1$, $(12)=0$

Convolutional

Rate = 3:4
Decoding

- Max level $D=4$, size $m=2$, overlap $v=1$
- 14314 1431 2341 4142... cell level
- $\rightarrow$ 1001 0100 1110 1010 ... permutation
- $\rightarrow$ 001 100 000 010 ... message

<table>
<thead>
<tr>
<th>First Two Permutation Bits</th>
<th>Permutation Sequence</th>
<th>Information Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equal</td>
<td>0010</td>
<td>000</td>
</tr>
<tr>
<td></td>
<td>1110</td>
<td>000</td>
</tr>
<tr>
<td></td>
<td>0001</td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>1101</td>
<td>111</td>
</tr>
<tr>
<td>Not equal</td>
<td>1011</td>
<td>011</td>
</tr>
<tr>
<td></td>
<td>0100</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>0101</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td>0110</td>
<td>110</td>
</tr>
</tbody>
</table>
Summary

- Partial rank modulation
  - Less sorting iterations
  - Smaller permutation size
  - Gray code

- Bounded rank modulation
  - Given permutation size & maximal cell level
  - Overlap and capacity
  - Encoder and decoder
Open Problems

• Partial rank modulation
  ◦ Error-correcting codes using lower levels as redundancy
  ◦ Efficient mapping between permutations and values
  ◦ Gap distributions after push-to-the-top

• Bounded rank modulation
  ◦ Exact optimal overlap values for \( m < D < \infty \)
  ◦ Error-correcting codes
  ◦ Efficient encoding/decoding
Thank you!