6.962 Week 5

Topic: Irregular Repeat-Accumulate (IRA) Codes

Presenter: Stark Draper

Thanks to: F. R. Kschischang and R. McEliece
for use of their slides.

**Note:** The slides available on-line from a subset of those presented in class. Some slides of F. Kschischang and R. McEliece were also used.
The “Final” Problem

For a given channel, find an ensemble of codes that:

(1) Has a linear-time encoding algorithm.

(2) Can be decoded reliably in linear time at rates arbitrarily close to channel capacity.

⇒ IRA codes satisfy (1), (2) on Binary Erasure Channel (BEC)!!
Outline

I. IRA Codes & Factor Graphs

II. BSC

III. BEC

IV. Density Evolution

V. Computational Complexity

VI. IRA Code Performance
IRA codes vs. RA codes

RA codes

<table>
<thead>
<tr>
<th>Info nodes const degree</th>
<th>Chk nodes left degree 1</th>
<th>Parity nodes degree 2</th>
<th>Nonsystematic</th>
</tr>
</thead>
</table>

IRA codes

<table>
<thead>
<tr>
<th>Info nodes var degree</th>
<th>Chk nodes left degree $a$</th>
<th>Parity nodes degree 2</th>
<th>Systematic</th>
</tr>
</thead>
</table>

4
Factor Graph of IRA Code

Observations

Information Nodes (Visible Variables)

Check Nodes (Valid Behaviors)

Parity Nodes (Visible Variables)

Observations

\[ f(y_0|x_0) \quad f(y_1|x_1) \quad \ldots \quad f(y_5|x_5) \]
Outline

I. IRA Codes & Factor Graphs

II. BSC

III. BEC

IV. Density Evolution

V. Computational Complexity

VI. IRA Code Performance
A Posteriori Probabilities

We transmit \((x_1, \ldots, x_N) = x_1^N\) over a memoryless channel.

We observe \((y_1, \ldots, y_N) = y_1^N\).

\[
f(x_1^N | y_1^N) = \frac{f(y_1^N | x_1^N)p(x_1^N)}{f(y_1^N)} \propto p(x_1^N) \prod_i f(y_i | x_i) \\
= \frac{1}{|C|} \chi_{\text{Behavior}}^C(x_1^N) \prod_i f(y_i | x_i) \text{ Probabilities}
\]
BSC: Probabilities

Joint Probabilities:

<table>
<thead>
<tr>
<th></th>
<th>$x = 0$</th>
<th>$x = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y = 0$</td>
<td>$0.5(1 - p)$</td>
<td>$0.5p$</td>
</tr>
<tr>
<td>$y = 1$</td>
<td>$0.5p$</td>
<td>$0.5(1 - p)$</td>
</tr>
</tbody>
</table>

Initial probabilities to pass:

| $f(y = 0|x)$ | $x = 0$ | $x = 1$ |
|--------------|---------|---------|
|               | $(1 - p)$ | $p$     |

| $f(y = 1|x)$ | $x = 0$ | $x = 1$ |
|--------------|---------|---------|
| $p$          |         | $(1 - p)$ |

But, these change during decoding...
Binary Probability Gates 2

Can build multiple-input probability gates in terms of two-input gates, since:

\[ \text{VAR}(x_1, x_2, \ldots, x_n) = \text{VAR}(x_1, \text{VAR}(x_2, \ldots x_n)) \]
\[ \text{CHK}(x_1, x_2, \ldots, x_n) = \text{CHK}(x_1, \text{CHK}(x_2, \ldots x_n)) \]

Messages parameterize the density.

\[ m = \log \left[ \frac{p(x = 0; y)}{p(x = 1; y)} \right]. \]

Multiplication is carried out on densities, not messages.
Outline

I. IRA Codes & Factor Graphs

II. BSC

III. BEC

IV. Density Evolution

V. Computational Complexity

VI. IRA Code Performance
BEC: Probabilities

Joint Probabilities:

<table>
<thead>
<tr>
<th></th>
<th>$x = 0$</th>
<th>$x = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y = 0$</td>
<td>$0.5(1-p)$</td>
<td>0</td>
</tr>
<tr>
<td>$y = 1$</td>
<td>0</td>
<td>$0.5(1-p)$</td>
</tr>
<tr>
<td>$y = e$</td>
<td>$0.5p$</td>
<td>$0.5p$</td>
</tr>
</tbody>
</table>

Initial probabilities to pass:

<table>
<thead>
<tr>
<th></th>
<th>$x = 0$</th>
<th>$x = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(y = 0</td>
<td>x)$</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x = 0$</th>
<th>$x = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(y = 1</td>
<td>x)$</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x = 0$</th>
<th>$x = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(y = e</td>
<td>x)$</td>
<td>$0.5$</td>
</tr>
</tbody>
</table>
BEC: Probabilities

Probabilities simplify greatly.

Summary:

1. $f(y|x) \sim B(1, 0)$ if $y = 0$.

2. $f(y|x) \sim B(0, 1)$ if $y = 1$.

3. $f(y|x) \sim B(0.5, 0.5)$ if $y = e$. 
BEC: Messages Alphabet

1. \( \log \frac{f(y=0|x=0)}{f(y=0|x=1)} = \log[1/0] = +\infty. \)

\( \Rightarrow m = m_0. \)

2. \( \log \frac{f(y=1|x=0)}{f(y=1|x=1)} = \log[0/1] = -\infty. \)

\( \Rightarrow m = m_1. \)

3. \( \log \frac{f(y=e|x=0)}{f(y=e|x=1)} = \log[0.5/0.5] = 0. \)

\( \Rightarrow m = m_e. \)

Message alphabet doesn’t change during decoding!!!
BEC: Message Alphabet

Message alphabet reduces to three letters:

1. $m = m_0$ if $\log \left[ \frac{f(y=0|x=0)}{f(y=0|x=1)} \right] = +\infty$

2. $m = m_1$ if $\log \left[ \frac{f(y=1|x=0)}{f(y=1|x=1)} \right] = -\infty$.

3. $m = m_e$ if $\log \left[ \frac{f(y=e|x=0)}{f(y=e|x=1)} \right] = 0$. 
BEC: Factor-Graphs rules

Node output rules simplify:

- A variable node outgoing message is $m_0$ ($m_1$) if any incoming messages is $m_0$ ($m_1$) (i.e. not all incoming messages are erasures, $m_e$). Having one incoming message $m_0$ and another $m_1$ is impossible on the BEC.

- A check node outgoing message is $m_0$ ($m_1$) only if all incoming messages are non-erasures and sum to $m_0$ ($m_1$). (A single erasure incoming message completely randomizes output of parity check.)

- If neither of the above holds, the outgoing message is $m_e$. 
Outline

I. IRA Codes & Factor Graphs

II. BSC

III. BEC

IV. **Density Evolution**

V. Computational Complexity

VI. IRA Code Performance
Fixed-Point Analysis

1. Assume decoding has converged.

2. Calculate ensemble-average erasure probabilities \((x_0, x_1, x_2, x_3)\)

3. Eliminate \(x_1, x_2, x_3\) to leave an equation parameterized by \(x_0\).

4. Show \(x_0 \notin (0, 1]\).

\[\Rightarrow x_0 = \Pr(\text{erasure on info bit}) \rightarrow 0.\]
Fixed-point $\Pr(\text{erasure})$ Calculation

Condition on the degree of the node:

$$x_1 = \Pr(m = m_e, \text{chk} \rightarrow \text{parity})$$

$$= \sum_{i=1}^{D_{\text{max}}} \Pr(m_e, \text{chk node deg } i)$$

$$= \sum_{i=1}^{D_{\text{max}}} [1 - \Pr(m_e | \text{deg } i)] \Pr(\text{deg } i)$$

$$= \sum_{i=1}^{D_{\text{max}}} [1 - (1 - x_2) \prod_{j=1}^{i} (1 - x_0)] \Pr(\text{deg } i)$$

$$= 1 - (1 - x_2) \sum_{i} (1 - x_0)^i \left[ \frac{\rho_i}{i} \frac{1}{\sum (\rho_j/j)} \right]$$

$$= 1 - (1 - x_2) R(1 - x_0).$$

And similarly for $x_0, x_2, x_3$. 
Summary

$x_i^{(L)}$ is the value of $x_i$ on the $L$th iteration.

\[
\begin{align*}
  x_1[L] &= 1 - (1 - x_2[L])R(1 - x_0[L]), \\
  x_2[L] &= px_1[L], \\
  x_3[L] &= 1 - (1 - x_2[L])^2 \rho(1 - x_0), \\
  x_0[L + 1] &= p\lambda(x_3[L]). \\
\end{align*}
\]

Which becomes

\[
x_0[L+1] = p\lambda \left( 1 - \left[ \frac{1 - p}{1 - pR(1 - x_0[L])} \right]^2 \rho(1 - x_0[L]) \right)
\]

Show must converge to a point $x_0 \notin (0, 1]$. 
Outline

I. IRA Codes & Factor Graphs

II. BSC

III. BEC

IV. Density Evolution

V. Computational Complexity

VI. IRA Code Performance
Outline

I. IRA Codes & Factor Graphs

II. BSC

III. BEC

IV. Density Evolution

V. Computational Complexity

VI. IRA Code Performance