Repeat-Accumulate Codes

Telecommunications Laboratory
Alex Balatsoukas-Stimming

Technical University of Crete

July 2, 2009
Outline

1 Repeat-Accumulate codes
   - Motivation
   - Definition
   - Encoder Structure

2 Irregular Repeat-Accumulate Codes
   - Motivation
   - Definition

3 Other RA Code Extensions
   - Structured IRA Codes (S-IRA)
   - Accumulate-Repeat-Accumulate Codes (ARA)

4 References
Divsalar et al. [1] attempted to prove AWGN coding theorems for a class of codes they call “turbo-like”.

Their proof technique used the ensemble input-output weight enumerator (IOWE) and combined this with the classical union bound to show that the ML word error probability reaches zero as $N \to \infty$ for some SNR threshold.

The difficulty in calculating the IOWE restricted them to very simple coding systems which they called *repeat and accumulate* codes.
Repeat-Accumulate Codes

The class of RA codes can be viewed as a subclass of LDPC codes (or Turbo codes)

Their encoding is done as follows:

1. A frame of information symbols of length $N$ is repeated $q$ times, resulting in a length $qN$ frame.
2. A random (but fixed) permutation is applied to the resulting frame.
3. The permuted frame is fed to a rate-1 accumulator with transfer function $1/(1 + D)$. 
Divsalar et al. prefer to think of the accumulator as a rate-1 block code whose input block $[x_1, \ldots, x_{qN}]$ and output block $[y_1, \ldots, y_{qN}]$ are related by the following formula:

\[
\begin{align*}
    y_1 &= x_1 \\
    y_2 &= x_1 + x_2 \\
    \vdots \\
    y_n &= x_1 + x_2 + \cdots + x_{qN}
\end{align*}
\]

Which corresponds to the following generator matrix:

\[
G_2 = \begin{bmatrix}
    1 & 1 & 1 & \ldots & 1 \\
    0 & 1 & 1 & \ldots & 1 \\
    0 & 0 & 1 & \ldots & 1 \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & 0 & \ldots & 1
\end{bmatrix}_{qN \times qN}
\]
The final stage of the encoder can be thought of either as an accumulator as pictured above (resulting in a “Turbo-like” code), or as the block code defined previously (resulting in an “LDPC-like” code).
The resulting systematic generator matrix is:

$$G = \begin{bmatrix} I & G_1G_2 \end{bmatrix}$$

where $G_1$ is a $N \times qN$ matrix representing both the $q$-times repetition and the permutation of the $N$ information bits.

This encoding scheme results in a code of rate $1/(q + 1)$.

For a non systematic generator matrix we simply omit the systematic part (i.e. the identity matrix):

$$G = [G_1G_2]$$

This encoding scheme results in a code of rate $1/q$. 
Repeat-Accumulate Codes

+ Simple structure.
+ Linear encoding complexity.
+ Efficient iterative decoding using belief propagation.
+ Good performance.

- High error floors.
- Small choice of rates and only rates below 1/2 or equal to 1 (since $q \geq 1$).
As with the superset of LDPC codes, irregular RA codes were introduced Jin et al. in 2000.

Motivation: irregular LDPC codes generally perform a lot better than regular LDPC codes.

Each information bit is not repeated a fixed number of times as with regular RA codes.
The Tanner graph of an IRA code with parameters \((f_1, \ldots, f_J, \alpha)\) where \(f_i \geq 0\) and \(\sum_i f_i = 1\) is as follows:
Irregular Repeat-Accumulate Codes

- The $k$ variable nodes on the left are the information nodes.
- There are $r = (k \sum_i if_i)/a$ check nodes.
- There are $r$ variable nodes, called parity nodes, connected to the $r$ check nodes in a simple zigzag manner.
- The recursive formula for the calculation of the parity bits is as follows:

\[ x_j = x_{j-1} + \sum_{i=1}^{\alpha} v_{(j-1)\alpha+i} \]
Irregular Repeat-Accumulate Codes

- For the non systematic version of the above code, the codeword is:
  \[(x_1, \ldots, x_r)\]
  and the corresponding rate is:
  \[
  \text{Rate} = \frac{k}{r} = \frac{\alpha}{\sum_i i f_i}
  \]

- For the systematic version of the above code, the codeword is:
  \[(u_1, \ldots, u_k; x_1, \ldots, x_r)\]
  and the corresponding rate is:
  \[
  \text{Rate} = \frac{k}{r + k} = \frac{\alpha}{\alpha + \sum_i i f_i}
  \]
Irregular Repeat-Accumulate Codes

As with LDPC codes, density evolution can be used to find good degree distributions by solving a linear program.

By using the Gaussian approximation for the messages exchanged by the BP algorithm, the problem is greatly simplified.

The best rate-$1/2$ IRA code by Divsalar et al. [2] has a threshold of 0.266 dB, while the best rate-$1/2$ irregular LDPC code found in [3] has a threshold of 0.25 dB (both calculated by using exact density evolution).
Irregular Repeat-Accumulate Codes

- Simple structure.
- Linear encoding complexity.
- Efficient iterative decoding using belief propagation.
- Good performance.
- Disadvantages of regular RA codes are fixed.
Structured IRA Codes

The parity-check matrix of an RA code can be written as follows:

\[ H = [ H_1 \quad H_2 ] \]

where

\[
H_2 = \begin{bmatrix}
1 & 0 & 0 & 0 & \ldots & 0 \\
1 & 1 & 0 & 0 & \ldots & 0 \\
0 & 1 & 1 & 0 & \ldots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & 1 & 1 & 0 \\
0 & 0 & \ldots & 0 & 1 & 1 \\
\end{bmatrix}
\]
Structured IRA Codes

We define the matrix $P$ as follows:

$$P = \begin{bmatrix}
\pi^{b_{0,0}} & \pi^{b_{0,1}} & \ldots & \pi^{b_{0,J-1}} \\
\pi^{b_{1,0}} & \pi^{b_{1,1}} & \ldots & \pi^{b_{1,J-1}} \\
\vdots & \vdots & \ddots & \vdots \\
\pi^{b_{L-1,0}} & \pi^{b_{L-1,1}} & \ldots & \pi^{b_{L-1,J-1}}
\end{bmatrix}_{L \times J}$$

where $\pi$ is a right cyclic shift $Q \times Q$ permutation matrix and $b_{i,j} \in \{0, 1, \ldots, Q - 1, \infty\}$ are the corresponding exponents.

We define $\pi^\infty$ as the zero matrix.
If we use $H_1 = P$, the resulting code has a poor minimum distance.

We use a permuted version of $P$ instead, where the permutation is chosen so that the minimum codeword weight for low-weight inputs is increased.

So, the final parity-check matrix will be:

$$H = [\Pi P \ H_2]$$

The resulting code is regular unless we choose to mask out some entries (by choosing $b_{i,j} = \infty$) in the $P$ matrix in accordance with a targeted repetition profile.
In [5], the addition of a rate-1 accumulator before the repetition code was proposed.

Using density evolution, it was shown that this precoding improves performance.

For example, for a rate-1/3 RA code, the iterative decoding threshold is 0.73 dB, while with the addition of the precoder the threshold is lowered to −0.048 dB.

This improvement in performance is called precoding gain.
Irregular ARA codes (IARA) were also proposed.

Rates greater than 1/2 can be achieved by puncturing.
References


