Numerical Issues Affecting LDPC Error Floors

Presenter: Brian K. Butler, PhD Candidate
Advisor: Paul H. Siegel, Professor

CMRR and Department of Electrical and Computer Engineering,
University of California, San Diego

October 25–26, 2012
“Numerical Issues Affecting LDPC Error Floors”

- To be presented at the Globecom Conference, Dec 2012.
- Studies the high-SNR performance of LDPC codes to help understand their suitability to very low error-rate applications (e.g., broadcast, deep-space communications, storage)
- Concludes that even high-accuracy arithmetic (e.g., 64-bit floating-point) can impart an error floor if not careful.
Iterative Belief Propagation (BP) Decoding

Channel Quality/SNR

Decoder Output

Error Rate

High

Low

High Error Rates are Bad

“Waterfall”

Slope transition

“Error Floor”

Low

Moderate

High
LDPC codes, a class of linear block codes, are a very important part of modern error correction [R. Gallager, 1962].

A binary linear block code may be defined by the set of codewords \( \mathcal{C} = \{ \mathbf{c} \in \mathbb{F}_2^n : \mathbf{H} \mathbf{c} = \mathbf{0} \} \), where

- \( \mathbf{H} \) is the \( r \times n \) parity-check matrix over \( \mathbb{F}_2 \).
- \( \mathbf{c} \) is the codeword column vector of \( n \) elements.
- \( \mathbf{0} \) is the all-zero column vector of \( r \) elements.

\( \mathbf{H} \) must be sparse for the code to be LDPC.
A **Tanner graph** is a bipartite graph (without parallel edges) that describes \( H = (h_{ij}) \).

Tanner graph \( B = (V, C, E) \).

Vertices are partitioned into variable nodes \( V \) and check nodes \( C \).

The edge \((c_i, v_j)\) is in \( E \) if and only if \( h_{ij} = 1 \).
Trapping sets (i.e., near codewords) are tightly interconnected graphical structures within the Tanner graph. When the variables of the TS are in error relatively few check nodes are unsatisfied (shaded).

\[(a, b) = (4, 2)\) example \hspace{2cm} (a, b) = (5, 3)\) example
During the first half-iteration, we compute the log-likelihood ratio (LLR) messages to be sent from the CNs to the connected VNs as

$$\lambda_{i \leftarrow j}^{[i \leftarrow j]} = 2 \tanh^{-1} \left( \prod_{k \in \mathcal{N}(j) \setminus i} \tanh \frac{\lambda_{k \rightarrow j}^{[k \rightarrow j]}}{2} \right) .$$  \hspace{1cm} (1)
Now, during the second half-iteration, we sum incoming LLR messages at the VNs to be sent back to the connected CNs as

\[
\lambda_{j}^{[i \rightarrow j]} = \lambda^{[i]} + \sum_{k \in \mathcal{N}(i) \setminus j} \lambda_{j}^{[i \leftarrow k]}.
\]  
(2)

High SNR drives LLRs \(\lambda_{i}^{[i \rightarrow j]}\) to high magnitude.

On a graph with cycles the LLR magnitudes may grow indefinitely.
Numerical Problems of the SPA Decoder
for CN update of highly certain messages

Double precision floating point (DP-FP) (i.e., 64-bit IEEE 754) computations maintain \( p = 53 \) bits worth of precision.

- tanh-based LLR-domain: suffers a round-to-\( \pm 1 \) problem, due to the limited precision, for any log-likelihood ratio (LLR) \(|\lambda| > 38.12309 = (p + 2) \ln(2)\).

- Jacobian-based LLR-domain: has no numerical issues

- Gallager’s involution transform (GIT) approach suffers an intermediate loss of precision.

- LR-domain is limited due to intermediate overflow.

- LD-domain inherently suffers a round-to-\( \pm 1 \) problem.

Our examination of published error floor results suggests that LLR limiting is commonly employed implicitly.
Numerical Summary of BP Algorithms/SPA

<table>
<thead>
<tr>
<th>Technique</th>
<th>LLR-equivalent limit (approx.)</th>
<th>LLR limit for DP-FP</th>
<th>LLR limit for QP-FP</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLR(^a)</td>
<td>((p + 2) \ln 2) (2^{emax+1})</td>
<td>38.12</td>
<td>79.72</td>
</tr>
<tr>
<td>Jacob.</td>
<td>((p + 2) \ln 2) (2^{emax+1})</td>
<td>1.798 (\times 10^{308})</td>
<td>1.190 (\times 10^{4932})</td>
</tr>
<tr>
<td>MSA(^b)</td>
<td>((p + 2) \ln 2) (2^{emax+1})</td>
<td>38.12</td>
<td>79.72</td>
</tr>
<tr>
<td>GIT</td>
<td>((p + 2) \ln 2) ((emax + p) \ln 2)</td>
<td>745.8</td>
<td>11434</td>
</tr>
<tr>
<td>GIT2</td>
<td>((p + 2) \ln 2) ((emax + p) \ln 2)</td>
<td>354.9</td>
<td>5678</td>
</tr>
<tr>
<td>LR</td>
<td>((p + 1) \ln 2) ((emax + 1) \ln 2/2)</td>
<td>709.8</td>
<td>11357</td>
</tr>
<tr>
<td>LR2</td>
<td>((p + 1) \ln 2) ((emax + 1) \ln 2)</td>
<td>37.43</td>
<td>79.02</td>
</tr>
<tr>
<td>LD</td>
<td>((p + 1) \ln 2) ((emax + p) \ln 2)</td>
<td>745.8</td>
<td>11434</td>
</tr>
</tbody>
</table>

\(^a\)SPA decoder using \(\text{tanh}\)() function on LLRs.

\(^b\)MSA decoder approximates SPA-LLR with a loss of 0.6 to 1.22 dB.
SPA without an upper limit on LLR

It is easy to extend a large dynamic range algorithm to one with no upper limit.

Generally, all SPA algorithms become insensitive to scaling as LLRs grow large.

The Min-Sum Algorithm is always invariant to scaling.
Numerical Issues Affecting LDPC Error Floors

Floating-point Simulation Results

Maximum of 200 Iterations with Varying LLR Saturation Limits

Margulis LDPC code w/ \( n = 2640 \), rate \( R = 0.5 \)

![Graph](image)

- **FER vs.** \( E_b/N_0 \) in dB for the AWGN channel.
- Floating-point simulation.

Maximum 200 iterations.
Semi-analytic technique performed at 2.8 dB.
Semi-analytic technique counts only (12,4) and (14,4) trapping sets.
Numerical Issues Affecting LDPC Error Floors

Floating-point Simulation Results

Maximum of 200 Iterations with Varying LLR Saturation Limits

802.3an LDPC Code w/ \( n = 2048 \), rate \( R \approx 0.84 \)

BER vs. \( E_b/N_0 \) in dB for the AWGN channel.

Floating-point simulation.
SPA without an upper limit on LLR (non-sat.)

A regular LDPC code with block length \( n = 1057 \), rate \( R \approx 0.77 \), and \( d_{\text{min}} = 8 \)

Floating-point simulation.

FER vs. \( E_b/N_0 \) in dB for the AWGN channel.

Uses Richter's approximation to \( \ln(1+\exp(-|x|)) \) in SPA for speed.
Several numeric improvements to existing SPA algorithms are identified to increase the range of highly certain messages.

Two new SPA algorithms (i.e., OLD and SPA without upper limit) are also proposed to increase the range of highly certain messages.

Increased insight into the behavior of iterative-BP decoding on graphs with cycles at high SNR.
Thank You