The Secant Method Decoder

Nandakishore Santhi[1]

[1]T-13 / CCS-3 / T-CNLS
Los Alamos National Laboratory

Algorithms, Inference and Statistical Physics
Santa Fe, May 3, 2007

Outline of the Talk

Introduction

- LDPC codes can be decoded near optimally using various forms of the *Belief-Propagation (BP)* algorithm.
- These sub-optimal LDPC decoders can also be understood in terms of Linear Programming.
- In this talk, we formulate an LDPC decoder using some iterative techniques for solving non-linear systems of equations and for non-linear minimization and present some preliminary results.
- Because of the method's origins in the *Secant Method* for NL systems, we call this decoder *The Secant Method Decoder*.

References

P. Wolfe, "The Secant Method for Simultaneous Nonlinear Equations," *Communications of the ACM*, **2**, No. 12, pp. 12-13, Dec. 1959.

The Secant Method for NL Equations

- A sub-optimal algorithm for solving simultaneous non-linear equations, introduced in 1959 by P. Wolfe.
- Say, we wish to solve the following system of m non-linear equations:

$$f_i(\mathbf{x}) = 0; \quad \text{for } i = 1, \ldots, m \tag{1}$$

- Given a set of $(m+1)$ trial solutions $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m+1}\}$ find parameters $\{\pi_1, \pi_2, \ldots, \pi_{m+1}\}$ by solving a linear system:

$$\sum_{i=1}^{m+1} \pi_i = 1; \tag{2}$$

$$\sum_{j=1}^{m+1} \pi_j f_i(\mathbf{x}_j) = 0; \quad \text{for } i = 1, \ldots, m \tag{3}$$

- Replace $\mathbf{x}_\ell$ with the largest cost $\sum_{i=1}^{m} |f_i(\mathbf{x}_\ell)|^2$ with a new estimate $\mathbf{x}'_\ell = \sum_{j=1}^{m+1} \pi_j \mathbf{x}_j$.

Local convergence of the Secant Method

- With some reasonable assumptions, it has been shown that the method has a super-linear order of local convergence.
- For the case of a single equation, the method can be shown to have an order of convergence equal to the *golden ratio*, $\frac{1+\sqrt{5}}{2}$. That is, the magnitude of error in any iteration is the product of the magnitude of error in the previous two iterations.
- As with any general low complexity algorithm for solving non-linear equations, there is no global convergence guarantee. However, the method is found to be useful in many practical applications.

Decoding a linear code used over a BSC

- Let $\mathbb{C}$ be an $[n, k, d]$ linear code. We will formulate the BSC decoding problem as the solution of a non-linear system.
- $\mathrm{BSC}(p)$: For $a \in \{0, 1\}$,

$$\begin{aligned} \Pr(a \to a) &= (1-p) \\ \Pr(a \to \bar{a}) &= p \end{aligned}$$

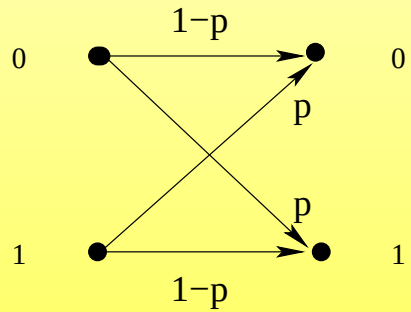

Figure: BSC Transition Diagram

- If $\mathbf{r} \in \mathbb{F}_2^n$ is the received vector, then we wish to find $\mathbf{c} \in \mathbb{C}$ such that $d_H(\mathbf{c}, \mathbf{r})$ is minimized for the Hamming distance. Here we need to consider only the cases $0 \leq d_H(\mathbf{c}, \mathbf{r}) \leq (n-k)$.

Hard Decoder as solving NL Equations: PROB A

- We use the following mapping from $\mathbb{F}_2$ to $\mathbb{R}$: $(0 \rightarrow +1)$ and $(1 \rightarrow -1)$. Also let $\mathbf{H} = [\mathbf{h}_{ij}]$ be a parity check matrix for $\mathbb{C}$.
- Constraint $\mathbf{c} \in \mathbb{F}_2^n$:

$$1 - c_j^2 = 0; \quad \text{for } j = 1, \ldots, n \tag{4}$$

- Constraint $\mathbf{c} \in \mathbb{C}$:

$$1 - \prod_{j:\mathbf{h}_{ij}=1} c_j = 0; \quad \text{for } i = 1, \ldots, (n-k) \tag{5}$$

- Constraint $d_H(\mathbf{c}, \mathbf{r}) = \delta$:

$$(n - 2\delta) - \sum_{j=1}^{n} r_j c_j = 0 \tag{6}$$

- Attempt to solve the above NL system for increasing values of δ from 0 to $(n-k)$. When estimates no longer change, STOP.

Secant Method applied on PROB A

- We can employ the *Secant Method* to try and obtain a solution to PROB A.
- The effectiveness of this method depends on the sparsity of the **H** matrix through the non-linearity of Eq(5) and on the choice of the $(2n-k+2)$ *trial solutions*.
- A possible choice for trial vectors is:

$$\begin{aligned} \mathbf{x}_j &= \mathbf{r}+\mathbf{e}_j, \quad j=1,\ldots,n; \\ \mathbf{x}_{n+1} &= \mathbf{r}; \\ \mathbf{x}_{j+n+1} &= \bar{\mathbf{r}}+\mathbf{e}_j, \quad j=1,\ldots,(n-k); \quad \text{and} \\ \mathbf{x}_{2n-k+2} &= \bar{\mathbf{r}} \end{aligned}$$

Decoding a linear code used over an AWGN channel

- We will formulate the AWGN decoding problem as a non-linear minimization problem.
- $\mathrm{AWGN}(0, \sigma)$:

$$r_j = c_j + \mathcal{N}(0, \sigma)$$

- If $\mathbf{r} \in \mathbb{R}^n$ is the received vector, then we wish to find $\mathbf{c} \in \mathbb{C}$ (mapped from $\mathbb{F}_2^n$ to $\mathbb{R}^n$) such that $d_2(\mathbf{c}, \mathbf{r})$ is minimized.

AWGN Soft Decoder as a NL minimization: PROB B

- We use the following mapping from $\mathbb{F}_2$ to $\mathbb{R}$: $(0 \rightarrow +1)$ and $(1 \rightarrow -1)$. Also let $\mathbf{H} = [\mathbf{h}_{ij}]$ be a parity check matrix for $\mathbb{C}$.
- A cost function for $\mathbf{c} \in \mathbb{F}_2^n$:

$$(1 - c_j^2)^2; \quad \text{for } j = 1, \ldots, n \tag{7}$$

- A cost function for $\mathbf{c} \in \mathbb{C}$:

$$(1 - \prod_{j:\mathbf{h}_{ij}=1} c_j)^2; \quad \text{for } i = 1, \ldots, (n-k) \tag{8}$$

- A cost function for $d_2(\mathbf{c}, \mathbf{r})$:

$$(r_j - c_j)^2; \quad \text{for } j = 1, \ldots, n \tag{9}$$

- Attempt to find a solution $\mathbf{c}$ which minimizes the total cost function. When the costs no longer improve, STOP.

Modified Secant Method for PROB B

- We propose a modified *Secant Method* to solve the NL minimization of PROB B.
- Sparsity of the **H** matrix is important.
- A possible set of trial vectors:

$$\begin{aligned} \mathbf{x}_j &= \mathbf{r} - 2r_j\mathbf{e}_j, \quad j = 1, \ldots, n; \\ \mathbf{x}_{n+1} &= \mathbf{r} \end{aligned}$$

- There are $(3n-k)$ cost functions and only $(n+1)$ parameters $\{\pi_j\}_{j=1}^{n+1}$ and trial solutions. Moreover we are interested in NL minimization.
- So we propose to use QR transform properties to obtain an iterative algorithm.

QR transform properties

- Any $M \times N$ matrix $\mathbf{A}$ can be decomposed as:

$$\mathbf{A} = \mathbf{Q}\mathbf{R}\mathbf{P}^T$$

  where $\mathbf{Q}$ is an $M \times M$ orthogonal matrix, $\mathbf{R}$ is an upper triangular matrix with the number of rows being equal to the $\mathrm{rank}(A)$, and $\mathbf{P}$ is a column permutation matrix.
- An over-complete linear system of the form:

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

  can be solved as the triangular system, $\mathbf{R}\mathbf{y} = \mathbf{Q}^T\mathbf{b}$, $\mathbf{x}^\star = \mathbf{P}\mathbf{y}$
- The solution $\mathbf{x}^\star$ has the property that it minimizes $||\mathbf{A}\mathbf{x}^\star - \mathbf{b}||_2$.

The Modified Secant Method

- Let the $m = (3n - k)$ non-linear cost functions be:

$$\begin{aligned} f_j(\mathbf{x}) &= x_j^2; \quad \text{for } j = 1, \ldots, n \\ f_{n+i}(\mathbf{x}) &= \prod_{j:\mathbf{h}_{ij}=1} x_j; \quad \text{for } i = 1, \ldots, (n-k) \\ f_{2n-k+j}(\mathbf{x}) &= x_j; \quad \text{for } j = 1, \ldots, n \end{aligned}$$

- Let $\mathbf{A} = [a_{ij}]$, where $a_{ij} = f_i(\mathbf{x}_j)$. Also let $\mathbf{b} = [b_i]$; for $j = 1, \ldots, (3n - k)$, where

$$\begin{aligned} b_i &= 1; \quad \text{for } j = 1, \ldots, (2n-k) \\ b_{2n-k+j} &= r_j; \quad \text{for } j = 1, \ldots, n \end{aligned}$$

- Least square solve the system $\mathbf{Ax} = \mathbf{b}$ using the QR transformation.
- Replace $\mathbf{x}_{n+1}$ with the solution $\mathbf{x}^\star$ of the QR step. Repeat till a local solution is reached.

Reducing iteration Complexity

- The QR method in general requires $\mathscr{O}(n^3)$ operations.
- Further changes to the matrix $\mathbf{A}$ occur only on column $(n+1)$ in each iteration. Therefore the QR decomposition can be modified using a *rank*-1 update which requires only $\mathscr{O}(n^2)$ operations.
- Observation: in practice about 5 iterations are required.

AWGN Performance of the Modified Secant Method Decoder

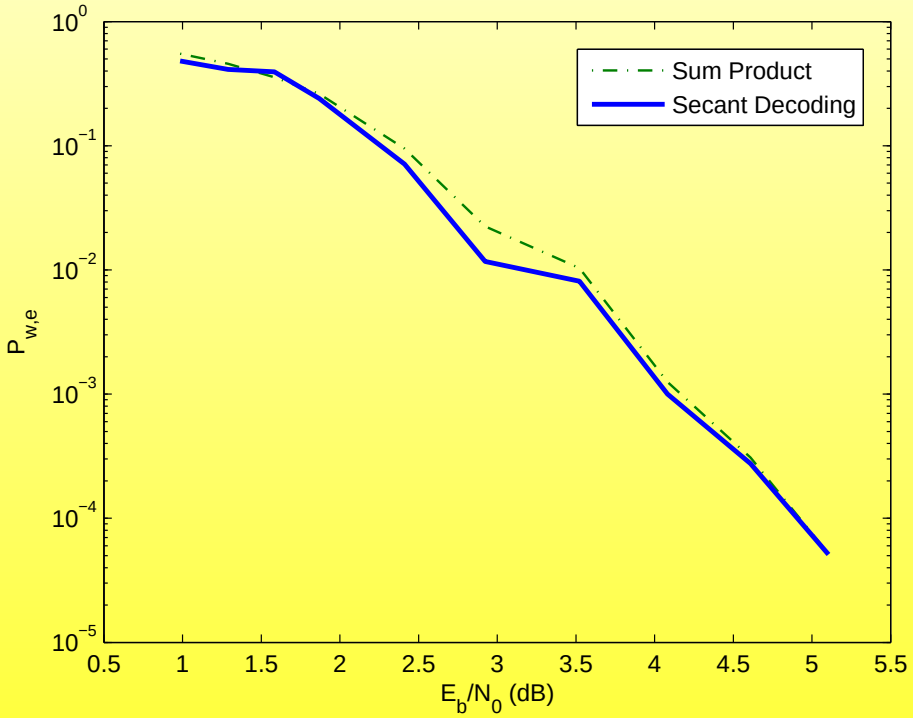

Figure: Word Error Rate as a function of $\frac{E_b}{N_0}$ for MacKay's 96.3.963 LDPC code. This code has length 96 and the **H** matrix has 48 rows.

Summary and further directions

- Presented a decoder for LDPC codes based on the Secant Method.
- Several modifications are possible, including a different choice of the trial solutions, as well as the cost functions.
- See if BP can be used in conjunction with similar non-linear techniques to improve the error floor behavior.
- The algorithm is more complex than BP, which has roughly linear complexity. For certain trial solution choices, it can be optimized to have a quadratic complexity. Are further improvements possible?