LP Decoding for Non-Uniform Sources and for the Non-Ergodic Polya Channel

A. Cohen, F. Alajaji, N. Kashyap, G. Takahara

Department of Math and Stats
Queen's University

Jul. 9 / 2008

What is a Linear Program?

- A linear program (LP) is an optimization problem over some set of variables, say $\underline{x} = (x_1, ..., x_n)$, in which the objective function is linear in the x_i variables and the region in which the solution is allowed to lie can be defined by a series of linear constraints.
- More precisely, an LP can be described as:

$$\begin{array}{ll} \text{minimize} & \underline{c}^T \underline{x} \\ \text{subject to} & A\underline{x} \leq \underline{b} \end{array}$$

.

- Alternatively, we can talk about optimizing over polytopes (*i.e*, the generalization of a polygon). Given any polytope, there exists a set of linear constraints (or facets) that describe the polytope.

Linear Programming Decoding

- For a length-n code C, the following LP is equivalent to ML decoding [1]:

$$\underline{\hat{x}} = \underset{\underline{x} \in poly(C)}{\text{argmax}} \; \underline{\gamma} \cdot \underline{x},$$

where $poly(C)$ is the convex hull of C defined by

$$poly(C) = \left\{ \sum_{\underline{y} \in C} \lambda_{\underline{y}} \underline{y} : \lambda_{\underline{y}} \geq 0, \sum_{\underline{y} \in C} \lambda_{\underline{y}} = 1 \right\}$$

and $\underline{\gamma} = (\gamma_1, \cdots, \gamma_n)$ is defined as the log-likelihood ratio (LLR) $\gamma_i = \log\left(\frac{P(y_i|c_i=1)}{P(y_i|c_i=0)}\right)$.

- Since solving this LP is equivalent to solving the ML decoding problem, we know that it is NP-hard.

Feldman's Codeword Polytope Relaxation

- A "relaxed" polytope can be used in order to solve the problem with managable complexity.
- The following relaxation is due to Feldman *et. al.* [1]. Given an (n, k) linear code C, and $H \subset C^{\perp}$, the relaxation is defined as:

$$\mathcal{Q}(H) = \bigcap_{\underline{h} \in H} \mathcal{P}(\underline{h}^{\perp}),$$

  where $\mathcal{P}(\underline{h}^{\perp})$ is the codeword polytope of the code $\underline{h}^{\perp} = \{\underline{c} \in \{0,1\}^n : \underline{h} \cdot \underline{c} \equiv 0 \pmod 2\}$.
- The following LP uses the above polytope to implement a sub-optimal ML decoder:

$$\hat{\underline{x}} = \underset{\underline{x} \in \mathcal{Q}(H)}{\operatorname{argmax}}\ \underline{\gamma} \cdot \underline{x}.$$

Properties of the Polytope $\mathcal{Q}(H)$

The relaxed polytope $\mathcal{Q}(H)$ has the following desirable properties:

1. The polytope can be expressed *easily* in terms of linear constraints (this is important, as such an expression is required by any LP solver).
2. For LDPC codes, the polytope $\mathcal{Q}(H)$ can be expressed *efficiently*, which means that an LP using this polytope can also be solved efficiently.
3. If we choose H to span $C^{\perp}$, then the integral vertices of $\mathcal{Q}(H)$ are exactly the codewords of C, (*i.e,* $\{0,1\}^n \bigcap \mathcal{Q}(H) = C$).
4. The above gives us the so-called "ML Certificate" property. That is, if the LP converges to an integral vertex, then it is known that this vertex must be the ML solution.

Exploiting Source Redundancy at the Decoder

- Non-uniformity at the source can be exploited at the decoder.
- Assuming a systematic (n, k) code C, it is possible to linearize the MAP decoding metric so as to exploit non-uniformity in an LP decoder:

$$\begin{aligned} \hat{\underline{c}} &= \operatorname*{argmax}_{\underline{c} \in C} P(\underline{c})P(\underline{y}|\underline{c}) \\ &= \operatorname*{argmax}_{\underline{c} \in C} \underline{\gamma^*} \cdot \underline{c} \end{aligned}$$

where $\underline{\gamma^*} = (\gamma_1^*, \cdots, \gamma_n^*)$ is defined by

$$\gamma_i^* = \begin{cases} \gamma_i + \log \frac{p_1}{p_0} & , \text{for } 1 \le i \le k \\ \gamma_i & , \text{for } k < i \le n. \end{cases}$$

Relaxation for MAP Decoding

- Using the cost function $\underline{\gamma}^*$ above, we can define an LP over the polytope $\mathcal{Q}(H)$ to obtain an LP relaxation for the MAP decoding problem:

$$\max_{\underline{x} \in \mathcal{Q}(H)} \underline{\gamma}^* \cdot \underline{x}.$$

- It is known that non-systematic codes perform better than systematic codes in scenarios with non-uniformity at the source; however, $\underline{\gamma}^*$ relies on the code being systematic.
- So, we would like to find an LP formulation for MAP decoding which does not require the transmission of a systematic code.

Puncturing

- One way to incorporate the *a-priori* codeword information into an LP decoder without transmitting a systematic code is to encode using a systematic code of rate lower than desired, and then puncture the systematic bits before transmission.
- More precisely, suppose that we wish to use a code of rate $R = \frac{k}{n}$, and blocklength n:
  1. We select a systematic $(n+k, k)$ code, $\widetilde{C}$.
  2. We encode source symbols $\underline{s}$ using $\widetilde{C}$, but, before transmission, we strip away the first k (systematic) bits.
- LP decoding can be performed over the "extended polytope" $\mathcal{Q}(\widetilde{H})$, $\widetilde{H} \subset \widetilde{C}^{\perp}$ using a modified cost function $\underline{\gamma}' = (\gamma'_1, \ldots, \gamma'_{n+k})$:

$$\gamma'_i = \begin{cases} \log\left(\frac{p_1}{p_0}\right), & \text{for } 1 \le i \le k \\ \log\left(\frac{P(y_{i-k}|x_i=1)}{P(y_{i-k}|x_i=0)}\right), & \text{for } k < i \le n+k. \end{cases}$$

Simulation Results for LP Decoding w/ Non-Uniform Sources

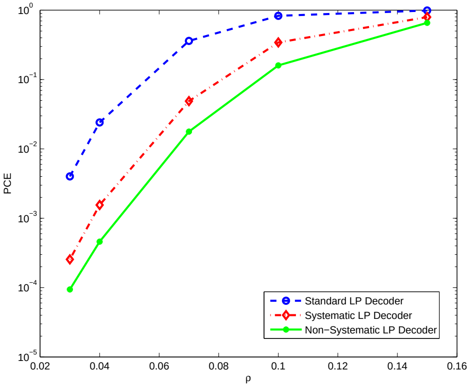


Figure: Source $p_1 = 0.9$. Top two curves: regular systematic $(200, 100)$ LDPC code. Bottom curve: regular $(300, 100)$ LDPC code with the first 100 (systematic) bits punctured.

Infinite-Memory Polya Contagion Channel

- The infinite-memory Polya-contagion communication channel is a binary non-ergodic channel in which the noise is modeled by the Polya-contagion urn scheme.
- At time i, a ball is drawn from an urn containing R red balls and B black balls ($T = R + B > 0$).
- If the ball is red, then the noise, z_i, at time slot i is 1, and otherwise, it is 0.
- After each draw, $1 + \Delta$ balls of the colour selected are added to the urn, where $\Delta \geq 0$.

ML Decoding for the Polya Channel w/ All-ones Codeword

- It can be shown (by an extension of results from [2]) that for any linear code C containing the *all-ones* codeword that if $\rho = \frac{R}{T} < 0.5$, then ML decoding is equivalent to minimum Hamming distance decoding (MDD).
- Further, if we consider $\underline{\gamma}^+$

$$\gamma_i^+ = \begin{cases} -1, & \text{if } y_i = 1 \\ 1, & \text{if } y_i = 0. \end{cases},$$

then we have the following, where $\omega()$ represents the Hamming weight [3]:

$$\underline{\gamma}^+ \cdot \underline{x} \quad = \quad d(\underline{x}, \underline{y}) - \omega(\underline{y})$$

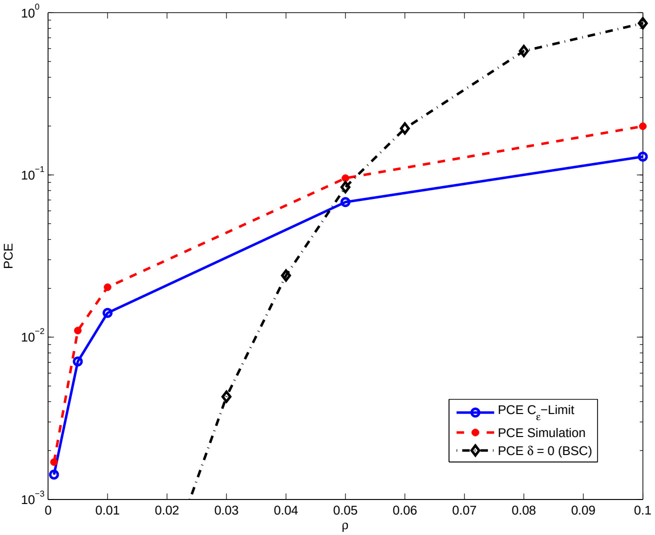
- So, it follows that

$$\min_{\underline{x} \in C} \underline{\gamma}^+ \cdot \underline{x} \quad = \quad \min_{\underline{x} \in C} d(\underline{x}, \underline{y}).$$

LP Decoding for the Polya Channel

- Using the cost function $\underline{\gamma}^{+}$ we can then define a relaxed LP which represents ML (MDD) decoding for the infinite-memory Polya channel assuming that our code contains the *all-ones* codeword.
- We implement this scheme using a regular $(200, 100)$ LDPC code with row degree 6. Having even row degree guarantees that the *all-ones* codeword is in the code.
- We compare the results to the channel ϵ-capacity, C_ϵ.
- For a given $\epsilon > 0$, the ϵ-capacity, C_ϵ, of a channel is defined as the maximum rate, R, for which there exist, given sufficiently large block length, codes having rate arbitrarily close to R and probability of error at most ϵ.
- For comparison purposes, we also include the code performance over the BSC (*i.e*,, when $\delta = 0$), corresponding to the situation where an ideal (infinite-depth) interleaver is applied to the channel.

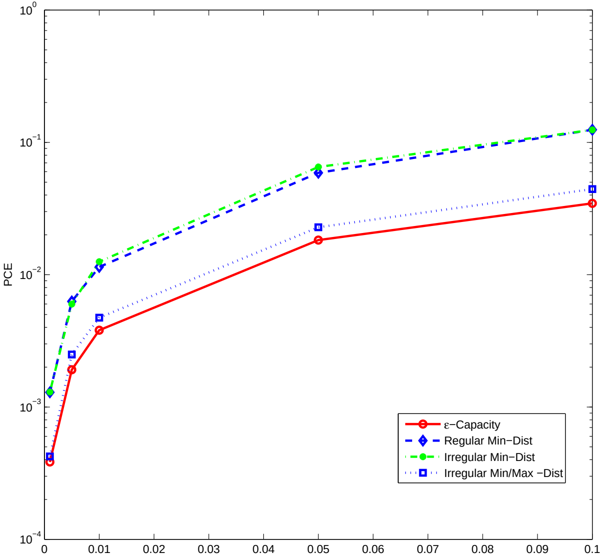
Simulation Results $\Delta = 2$



ML Decoding for the Polya Channel for Arbitrary Codes

- In [2], a formulation is given for ML decoding of arbitrary codes transmitted over the Polya channel. It is shown that ML decoding is achieved by either decoding to the minimum or maximum Hamming distance codeword (w.r.t. the received vector) depending on the channel parameters and the received vector.
- Maximum distance decoding can be approximated by an LP by using the negative of the cost function for minimum distance decoding, defined earlier.
- Using the conditions from [2] and the approximate min and max distance LP decoders, approximate ML decoding can be implemented for arbitrary codes.
- The next plot shows results for an irregular (w/out the all-ones codeword) $(200, 100)$ LDPC code under min/max distance decoding and under min distance only decoding as well as a regular (w/ the all ones codeword) $(200, 100)$ LDPC code under min distance only decoding.

Comparing Codes and Decoders for $\Delta = 10$



References

- J. Feldman, M. J. Wainright, and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Trans. Inform. Theory*, vol. 51, no. 3, pp. 954–972, 2005.
- F. Alajaji and T. Fuja, "A communication channel modeled on contagion," *IEEE Tras. Inform. Theory*, vol. 60, no. 6, pp. 2035–2041, 1994.
- J. Feldman, "Decoding error-correcting codes via linear programming," Ph.D. dissertation, Massachusetts Institute of Technology, 2003.