April 5, 2010

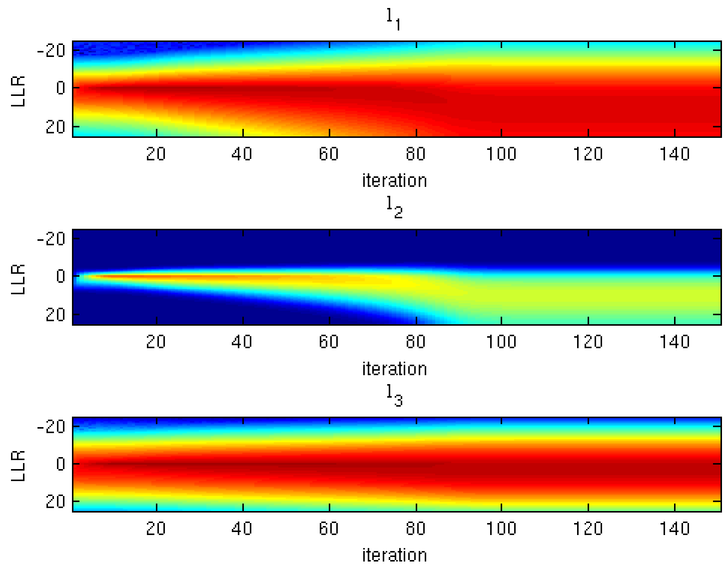

Figure 1: $\sigma = 0.9, q = 4, d = 256, min_{llr} = -20, max_{llr} = 20, niter = 150$

Density Evolution for Non-binary LT-Codes

I have implemented the density evolution method for Non-binary LT-codes. The parameters are

- Output degree distribution
- Size of the finite field, q
- Number of quantization for each dimension
- Minimum and maximum of LL values

In the current implementation the following issues exists

- The number of iteration is fixed,however, it should run until the difference between the probability of the error is less than a threshold. I have an ambiguity here how to calculate probability of error in non-binary case.
- Only the AWGN is implemented. Should I also consider non-binary symmetric channel?

Figure 1 shows a simulation for the degree distribution in Payam's paper in $GF(4)$. Three plots correspond to the densities of LLR values (y axis) λ_1, λ_2, and λ_3. The noise parameter of AWGN channel is 0.9 and each dimension is quantized into 256 values uniformly between -20 and 20. The x axis is the iteration number which is fixed to 150.

TODO:

- Checking for possible bug in calculating messages

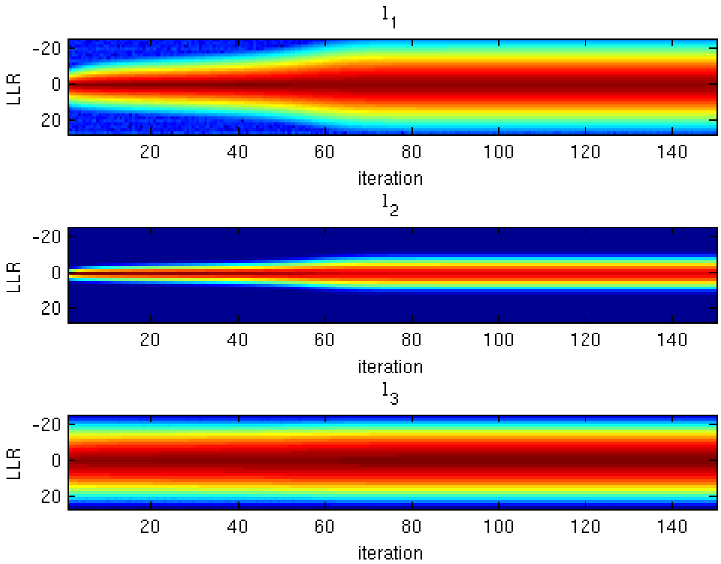

Figure 2: $\sigma = 0.9, q = 4, d = 128, min_{llr} = -20, max_{llr} = 20, niter = 150$

- Adding a parameter for the degree of the Raptor precode
- Optimizing the degree distribution for a given overhead and noise level?
- Using multi dimensional clustering to accelerate the procedure
- Adding support for LDPC codes (How to create random LDPC form a given degree distribution? random labels?)
- Using mixture of Gaussians to approximate the densities.

Figure 2 is another simulation using the following degree distribution:

```
1 0.017336
2 0.40023
3 0.21963
5 0.19276
10 0.068901
15 0.047744
36  0.02306
40 0.014646
102 0.0011277
113 0.0093442
300 0.005212
```

LP-Decoding and Lasserre SDP Relaxation

The ML-decoding of a linear binary code $\mathcal{C}$ can be stated as the following integer programming

$$x_{LP} := \underset{x\in\mathcal{C}}{\operatorname{argmin}} \langle \gamma, x\rangle \,, \tag{1}$$

where γ is the log-likelihood vector. The natural relaxation of this LP can be obtained by relaxing $x \in \mathcal{C}$ to $x \in [0,1]$ and the following constraints for each parity check,

$$\sum_{i\in V} x_i - \sum_{i\in N\setminus V} x_i \leq |V| - 1, \quad \forall V \subset N,\ |V| \text{ is odd}, \tag{2}$$

where N is a set of variable nodes adjacent to the check.

In order to get a comparable running time to BP decoding, several methods have been proposed to solve this LP more efficiently by exploiting the properties of this polytope. Here we are going to apply lift-and-project methods to the ML-decoding IP directly. Let K be a polytope defined by set of linear constraints $g_1, \ldots g_m$:

$$K = \{x \in [0,1]^n | g_l(x) \geq \text{ for } l = 1, \ldots m\}. \tag{3}$$

In our problem K is the LP relaxation defined in Equation 2. We are interested in optimizing objective function 1 over the convex hull $P = \text{conv}(K \cap \{0,1\}^n)$ which is the integral points in K. We are interested in strengthening the relaxation by removing more fractional points from K to get closer to integral solutions. This procedure can be done by introducing more constraints to K which is violated by some fractional points. We define a sequence of $h^1, h^2, \ldots h^r$ a hierarchy of relaxation if

$$K \subseteq h^1(K) \subseteq h^2(K) \subseteq \cdots \subseteq h^r(K) = P, \tag{4}$$

and r is called rank of the hierarchy.

In a more general setting we assume $g_1, \ldots g_m$ are polynomials in $x_1, \ldots x_n$. Since $x_i \in 0,1$ we have $x_i^2 = x_i$, therefore, we can assume each variable has degree at most 1 and we $g_l(x)$ can be factored as

$$\sum_{I\subseteq V} g_l(I) \prod_{i\in I} x_i, \tag{5}$$

where $g_l(I)$ is the coefficient associated with the monomial expansion. Let $V = \{1, \ldots, n\}$ and Let $\mathcal{P}_q(V)$ denotes the set of all subsets of V of cardinality at most q. We introduce real vectors $y \in \mathbb{R}^{\mathcal{P}(V)}$ and denote them by y_I or $y(I)$and set $y_0 = y_\emptyset, y_i = y_{\{i\}}$, and $y_{ij} = y_{\{i,j\}}$. Given $1 \leq t \leq n$ and $U \subseteq V$ define matrices

$$M_t(y) := (y(I \cup J))_{|I|,|J|\leq t} \tag{6}$$

$$M_U(y) := (y(I \cup J))_{I,J\subseteq U}. \tag{7}$$

The matrix $M_V(y) = M_n(y)$ is known as the moment matrix of y. For $x, y \in \mathbb{R}^{\mathcal{P}(V)}$ we define the shift operator $x * y \in \mathbb{R}^{\mathcal{P}(V)}$ by

$$x * y := M_V(y)x; \text{ that is, } x * y(I) := \sum_{F \subseteq V} x_F y_{I \cap K} \tag{8}$$

Lemma 1. *[2] Given $x \in K \cap \{0,1\}^n$, the vector $y \in \mathbb{R}^{\mathcal{P}(V)}$ with entries $y(I) := \prod_{i \in I} x_i$ satisfies*

$$M_V(y) \succeq 0, M_V(g_l * y) \succeq \text{ for } l = 1, \ldots, m. \tag{9}$$

The Sherai-Adams (SA) and Lasserre hierarchies are two systematic ways to construct additional constraints. In SA hierarchy, added constraints are linear, whereas in Lasserre's we introduce a set of positive semi-definite constraints.

Lasserre Hierarchy

Let w_l denote the degree of the polynomial g_l and set

$$v_l := \lceil \frac{w_l}{2} \rceil, \ w := \max w_l, \ v := \max v_l \tag{10}$$

For $t \leq v - 1$ set

$$P_t(K) := \{y \in \mathbb{R}^{\mathcal{P}_{2t+2}(V)} | M_{t+1}(y) \succeq 0, \ M_{t+1-v_l}(g_l * y) \succeq 0 \text{ for } l = 1, \ldots, m\} \tag{11}$$

The Lasserre relaxation of order t is defined by intersecting the first coordinate with hyperplane $y_0 = 1$ and then projecting the resulting set onto the coordinates associated with first-order moments [3] i.e. projection of $P_t(K) \cap y | y_0 = 1$ onto the subspace $mathbbR^n$ indexed by singletons. Lasserre shows that

$$P = Q_{n+v-1}(K) \subseteq \ldots \subseteq Q_v(K) \subseteq Q_{v-1}(K). \tag{12}$$

Laurent [2] applies this hierarchy to the stable set problem and max-cut problem and [1] applies it for Knapsack problem. However, I am having problem to apply this method to the LP decoding polytope. Another important result [3] shows that the rank of the relaxation is related to tree-width of the hypergraph of polynomial system, it is also interesting to see what is the tree-width related to LP decoding polytope. After the program formulation I would like to see if it is possible to implement this relaxation and compare it with constraints added in other approches such as Taghavi's.

Bibliography

[1] A. Karlin, C. Mathieu, and C.T. Nguyen. Integrality gaps of linear and semi-definite programming relaxations for knapsack, 2009.

[2] M. Laurent. A comparison of the Sherali-Adams, Lovász-Schrijver, and Lasserre relaxations for 0-1 programming. *Mathematics of Operations Research*, 28(3):470–496, 2003.

[3] M.J. Wainwright and M.I. Jordan. Treewidth-based conditions for exactness of the Sherali-Adams and Lasserre relaxations. *Univ. California, Berkeley, Technical Report*, 671, 2004.