Towards Low-Complexity Algorithms for LP Decoding

Pascal O. Vontobel
Information Theory Research Group
Hewlett-Packard Laboratories

USC, Los Angeles, CA, November 10, 2006

Overview

- Primal LP
- Dual LP
- Coordinate-Ascent Algorithm

Note: These slides are rather brief and not self-contained. For more information please consult the paper referenced at the end.

The Primal LP (Part 1)

$$\begin{aligned} \text{minimise} \quad & \sum_{i \in \mathcal{I}} \lambda_i x_i \\ \text{subject to} \quad & \mathbf{x} \in \operatorname{conv}(\mathcal{C}_j) \quad (j \in \mathcal{J}). \end{aligned}$$

Definitions (Part 1)

Here we used the following codes, variables and vectors. The code

- $\mathcal{A}_i \subseteq \{0,1\}^{|\{0\} \cup \mathcal{J}_i|}$, $(i \in \mathcal{I})$, is the set containing the all-zeros vector and the all-ones vector of length $|\mathcal{J}_i| + 1$
- $\mathcal{B}_j \subseteq \{0,1\}^{|\mathcal{I}_j|}$, $j \in \mathcal{J}$, is the code $\mathcal{C}_j$ punctured at the positions $\mathcal{I} \setminus \mathcal{I}_j$. (For the codes $\mathcal{C}$ under consideration this means that $\mathcal{B}_j$ contains all vectors of length $|\mathcal{I}_j|$ of even parity.)
- For $i \in \mathcal{I}$ we will also use the vectors $\mathbf{u}_i$ where the entries are indexed by $\{0\} \cup \mathcal{J}_i$ and denoted by $u_{i,j} \triangleq [\mathbf{u}_i]_j$, and for $j \in \mathcal{J}$ we will use the vectors $\mathbf{v}_j$ where the entries are indexed by $\mathcal{I}_j$ and denoted by $v_{j,i} \triangleq [\mathbf{v}_j]_i$.
- We will use a similar notation for the entries of $\mathbf{a}_i$ and $\mathbf{b}_j$, i.e. we will use $a_{i,j} \triangleq [\mathbf{a}_i]_j$ and $b_{j,i} \triangleq [\mathbf{b}_j]_i$, respectively.

The Primal LP (Part 2)

$$
\begin{aligned}
\text{min.} \quad & \sum_{i\in\mathcal{I}} \lambda_i x_i \\
\text{subj. to} \quad & x_i = u_{i,0} \quad (i\in\mathcal{I}), \\
& u_{i,j} = v_{j,i} \quad ((i,j)\in\mathcal{E}), \\
& \sum_{\mathbf{a}_i\in\mathcal{A}_i} \alpha_{i,\mathbf{a}_i}\mathbf{a}_i = \mathbf{u}_i \quad (i\in\mathcal{I}), \\
& \sum_{\mathbf{b}_j\in\mathcal{B}_j} \beta_{j,\mathbf{b}_j}\mathbf{b}_j = \mathbf{v}_j \quad (j\in\mathcal{J}), \\
& \alpha_{i,\mathbf{a}_i} \geq 0 \quad (i\in\mathcal{I},\ \mathbf{a}_i\in\mathcal{A}_i), \\
& \beta_{j,\mathbf{b}_j} \geq 0 \quad (j\in\mathcal{J},\ \mathbf{b}_j\in\mathcal{B}_j), \\
& \sum_{\mathbf{a}_i\in\mathcal{A}_i} \alpha_{i,\mathbf{a}_i} = 1 \quad (i\in\mathcal{I}), \\
& \sum_{\mathbf{b}_j\in\mathcal{B}_j} \beta_{j,\mathbf{b}_j} = 1 \quad (j\in\mathcal{J}).
\end{aligned}
$$

Definitions (Part 1)

If A is a statement, then

$$
[A] \triangleq \begin{cases} 1 & \text{if } A \text{ is true} \\ 0 & \text{if } A \text{ is false} \end{cases}
$$

$$
[\![A]\!] \triangleq -\log[A] = \begin{cases} 0 & \text{if } A \text{ is true} \\ +\infty & \text{if } A \text{ is false} \end{cases}
$$

The Primal LP (Part 3)

With suitable cost functions, the above optimization problem can also be formulated as an unconstrained optimization problem. Minimize

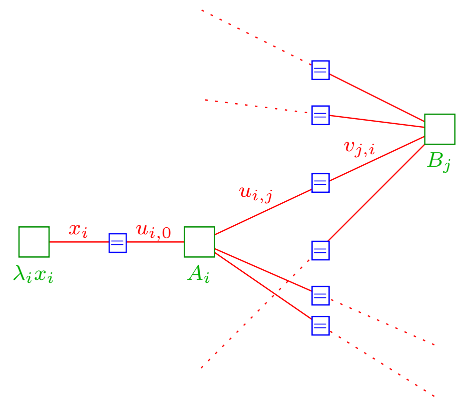
$$
\sum_{i\in\mathcal{I}} \lambda_i x_i + \sum_{i\in\mathcal{I}} [\![x_i = u_{i,0}]\!] + \sum_{(i,j)\in\mathcal{E}} [\![u_{i,j} = v_{j,i}]\!] + \sum_{i\in\mathcal{I}} A_i(\mathbf{u}_i) + \sum_{j\in\mathcal{J}} B_j(\mathbf{v}_j),
$$

where for all $i\in\mathcal{I}$ and all $j\in\mathcal{J}$, respectively, we introduced

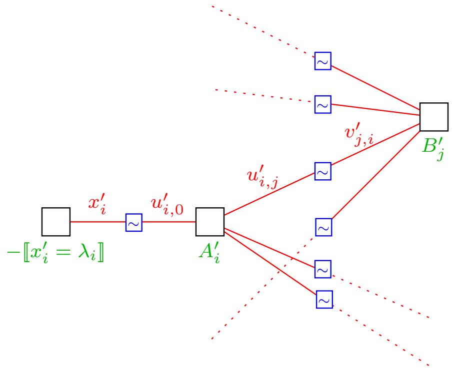
$$
A_i(\mathbf{u}_i) \triangleq \left[\!\!\left[\sum_{\mathbf{a}_i\in\mathcal{A}_i} \alpha_{i,\mathbf{a}_i}\mathbf{a}_i = \mathbf{u}_i \right]\!\!\right] + \sum_{\mathbf{a}_i\in\mathcal{A}_i} [\![\alpha_{i,\mathbf{a}_i} \geq 0]\!] + \left[\!\!\left[\sum_{\mathbf{a}_i\in\mathcal{A}_i} \alpha_{i,\mathbf{a}_i} = 1 \right]\!\!\right],
$$

$$
B_j(\mathbf{v}_j) \triangleq \left[\!\!\left[\sum_{\mathbf{b}_j\in\mathcal{B}_j} \beta_{j,\mathbf{b}_j}\mathbf{b}_j = \mathbf{v}_j \right]\!\!\right] + \sum_{\mathbf{b}_j\in\mathcal{B}_j} [\![\beta_{j,\mathbf{b}_j} \geq 0]\!] + \left[\!\!\left[\sum_{\mathbf{b}_j\in\mathcal{B}_j} \beta_{j,\mathbf{b}_j} = 1 \right]\!\!\right].
$$

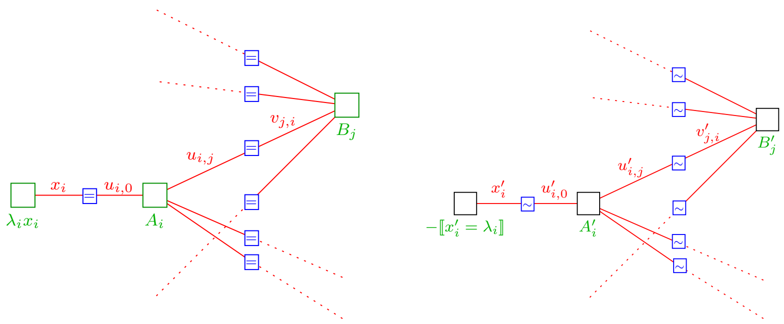
FFG representin the Primal LP



FFG representin the Dual LP



Comparison of FFGs Representin the Primal and Dual LP



Dual LP

$$
\begin{aligned}
\text{max.} \quad & \sum_{i \in \mathcal{I}} \phi'_i + \sum_{j \in \mathcal{J}} \theta'_j \\
\text{subj. to} \quad & \phi'_i \leq \min_{\mathbf{a}_i \in \mathcal{A}_i} \langle -\mathbf{u}'_i, \mathbf{a}_i \rangle \quad (i \in \mathcal{I}), \\
& \theta'_j \leq \min_{\mathbf{b}_j \in \mathcal{B}_j} \langle -\mathbf{v}'_j, \mathbf{b}_j \rangle \quad (j \in \mathcal{J}), \\
& u'_{i,j} = -v'_{j,i} \quad ((i,j) \in \mathcal{E}), \\
& u'_{i,0} = -x'_i \quad (i \in \mathcal{I}), \\
& x'_i = \lambda_i \quad (i \in \mathcal{I}).
\end{aligned}
$$

Coordinate- scent l orithm (Part 1)

We propose is a coordinate-ascent-type algorithm for solving the dual LP. The main idea is to select edges $(i,j) \in \mathcal{E}$ according to some update schedule: for each selected edge $(i,j) \in \mathcal{E}$ we then replace the old values of $u'_{i,j}$, ϕ'_i, and θ'_j by new values such that the dual cost function is increased (or at least not decreased). Practically, this means that we have to find a $\overline{u}'_{i,j}$ such that $h'(\overline{u}'_{i,j}) \geq h'(u'_{i,j})$, where

$$h'(u'_{i,j}) \triangleq \min_{\mathbf{a}_i \in \mathcal{A}_i} \langle -\mathbf{u}'_i, \mathbf{a}_i \rangle + \min_{\mathbf{b}_j \in \mathcal{B}_j} \langle -\mathbf{v}'_j, \mathbf{b}_j \rangle.$$

A simple way to achieve this is by setting

$$\overline{u}'_{i,j} \triangleq \arg\max_{u'_{i,j}} h'(u'_{i,j}).$$

The variables ϕ'_i and θ'_j are then updated accordingly so that we obtain a new (dual) feasible point.

Coordinate- scent l orithm (Part 2)

Lemma

The function $h'(u'_{i,j})$ is maximised by any value $u'_{i,j}$ that lies in the closed interval between

$$(S'_{i,0} - S'_{i,1}) \quad \text{and} \quad -(T'_{j,0} - T'_{j,1}),$$

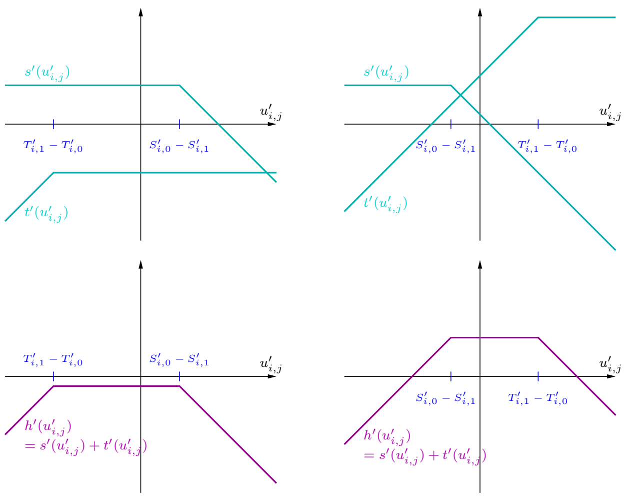
where

$$S'_{i,0} \triangleq -\min_{\substack{\mathbf{a}_i \in \mathcal{A}_i \\ a_{i,j}=0}} \langle -\tilde{\mathbf{u}}_i, \tilde{\mathbf{a}}_i \rangle, \qquad T'_{j,0} \triangleq -\min_{\substack{\mathbf{b}_j \in \mathcal{B}_j \\ b_{j,i}=0}} \langle -\tilde{\mathbf{v}}_j, \tilde{\mathbf{b}}_j \rangle,$$

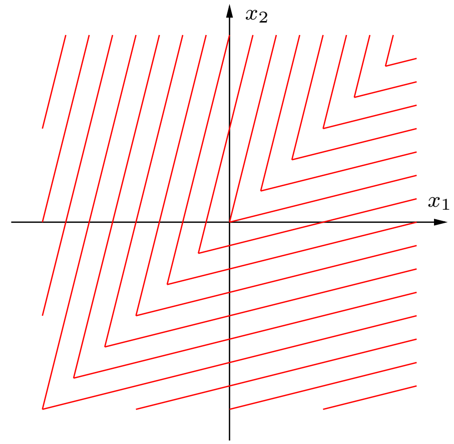
$$S'_{i,1} \triangleq -\min_{\substack{\mathbf{a}_i \in \mathcal{A}_i \\ a_{i,j}=1}} \langle -\tilde{\mathbf{u}}_i, \tilde{\mathbf{a}}_i \rangle, \qquad T'_{j,1} \triangleq -\min_{\substack{\mathbf{b}_j \in \mathcal{B}_j \\ b_{j,i}=1}} \langle -\tilde{\mathbf{v}}_j, \tilde{\mathbf{b}}_j \rangle.$$

Note that the differences $S'_{i,0} - S'_{i,1}$ and $T'_{i,0} - T'_{i,1}$, which are required for computing $\overline{u}'_{i,j}$, can be obtained very efficiently by using the min-sum algorithm.

Coordinate- scent l orithm (Part 3)



Problematic Contour for Coordinate- scent l orithm



References

For more info, see e.g.

- P. O. Vontobel and R. Koetter Towards low-complexity linear-programming decoding [arxiv] Proc. 4th Intern. Conf. on Turbo Codes and Related Topics, Munich, Germany, Apr. 3-7, 2006. [http://www.arxiv.org/abs/cs.IT/0602088]

invent