FDT

Bob, Cindy, Arash

September 18, 2018

Abstract

TO ADD.

1 Introduction

Mixed-integer linear programming (MILP), the optimization of a linear objective function subject to linear and integrality constraints, is a classic NP-hard problem [GJ79]. In fact it is NP-hard in general to determine if there is a feasible solution to an MILP problem [GJ79]. However, intelligent branch-and-bound

1.1 Our results

1.2 Notation and Preliminaries

2 FDT on binary MIPs

In this section we present the fractional decomposition tree (FDT) algorithm and prove its correctness.

Consider the set of point described by a set

$$S = \{(x,y) \in \mathbb{R}^{n\times p} \ : \ Ax + Gy \geq b, \ x \in \{0,1\}\}, \tag{1}$$

and a linear relaxation of set S

$$P = \{(x,y) \in \mathbb{R}^{n\times p} \ : \ Ax + Gy \geq b, \ 0 \leq x \leq 1\}. \tag{2}$$

Also assume we are given a point $(x^*, y^*) \in P$. For instance, (x^*, y^*) can be the optimal solution of minimizing a cost function cx over set P, which provides a lower bound on

$\min_{(x,y)\in S} cx$. The integrality gap of S with its relaxation P is defined as

$$g_S = \max_{c\in R^{n\times p}_{\geq 0},\, c\neq 0} \frac{\min_{(x,y)\in S} cx}{\min_{(x,y)\in P} cx}. \tag{3}$$

In this paper discuss sets S for which g_S is a finite number.

Theorem 1. *There is a polynomial time algorithm (the FDT algorithm) that produces $\lambda \in [0,1]^k$ and $(z^1, w^1), \ldots, (z^k, w^k) \in S$ such that $k \leq n$, $\sum_{i=1}^k \lambda_i z^i \leq Cx^*$, and $\sum_{i=1}^k \lambda_i = 1$. Moreover, $C \leq g_S^n$.*

Although the theoretical worst-case upper bound on C in the statement of Theorem 1 can be very large, we will show that in practice C can be really small and hence FDT can provide good LP-based approximation algorithms in many cases. We also remark that if $g_S = 1$, i.e. P is integral, then the algorithm will give an exact decomposition of any feasible solution. In the remainder of this section we explain the FDT algorithm and prove Theorem 1.

We begin with the following subroutine that we call *branching on a variable.*

Lemma 1. *Given $(x', y') \in \mathcal{D}(P)$ and $\ell \in 1, \ldots, n$, we can find in polynomial time vector $(\hat{x}^0, \hat{y}^0), (\hat{x}^1, \hat{y}^1)$ and scalars $\gamma_0, \gamma_1 \in [0,1]$ such that*

(i) $\gamma_0 + \gamma_1 \geq \frac{1}{g_S}$,

(ii) $(\hat{x}^0, \hat{y}^0)$ *and* $(\hat{x}^1, \hat{y}^1)$ *are in* P,

(iii) $\hat{x}^0_\ell = 0$ *and* $\hat{x}^1_\ell = 1$,

(iv) $\gamma_0 \hat{x}^0 + \gamma_1 \hat{x}^1 \leq x'$.

In order to prove Lemma 1 we need the following theorem due to Carr and Vempala [CV04].

Theorem 2 (Carr, Vempala [CV04])**.** *Let $(x', y') \in \mathcal{D}(P)$, there exists $\theta \in [0,1]^k$, where $\sum_{i=1}^k \theta_i = 1$ and $(\tilde{x}^i, \tilde{y}^i) \in S$ for $i = 1, \ldots, k$ such that $\sum_{i=1}^k \theta_i \tilde{x}^i \leq g_S x'$.*

Proof of Lemma 1. Consider the following linear programing which we denote by Branching(ℓ, x', y').

The variables of Branching(ℓ, x', y') are γ_0, γ_1 and (x^0, y^0) and (x^1, y^1).

$$
\begin{aligned}
\text{Branching}(\ell, x', y') \qquad & \max \quad \lambda_0 + \lambda_1 && (4)\\
& \text{s.t.} \quad Ax^0 + Gy^0 \geq b\lambda_0 && (5)\\
& \qquad Ax^1 + Gy^1 \geq b\lambda_1 && (6)\\
& \qquad 0 \leq x^0 \leq \lambda_0 && (7)\\
& \qquad 0 \leq x^1 \leq \lambda_1 && (8)\\
& \qquad x^0_\ell = 0,\ x^1_\ell = \lambda_1 && (9)\\
& \qquad x^0 + x^1 \leq x' && (10)\\
& \qquad \lambda_0, \lambda_1 \geq 0 && (11)
\end{aligned}
$$

Let $(x^0, y^0), (x^1, y^1)$, and λ_0, λ_1 be an optimal solution solution to the LP above. Let $(\hat{x}^0, \hat{y}^0) = (\frac{x^0}{\lambda_0}, \frac{y^0}{\lambda_0})$, $(\hat{x}^1, \hat{y}^1) = (\frac{x^1}{\lambda_1}, \frac{y^1}{\lambda_1})$, and $\gamma_0 = \lambda_0$ and $\gamma_1 = \lambda_1$. Observe that (ii), (iii), (iv) are satisfied with this choice. In order to show that (i) is also satisfied we prove the following claim.

Claim 1. We have $\lambda_0 + \lambda_1 \geq \frac{1}{g_S}$.

Proof. We show that there is a feasible solution that achieves the objective value of $\frac{1}{g_S}$. By Theorem 2 there exists $\theta \in [0, 1]^k$, with $\sum_{i=1}^k \theta_i = 1$ and $(\tilde{x}^i, \tilde{y}^i) \in S$ for $i = 1, \ldots, k$ such that $\sum_{i=1}^k \theta_i \tilde{x}^i \leq g_S x'$.

$$
x' \geq \sum_{i=1}^{k} \frac{\theta_i}{g_S} \tilde{x}^i = \sum_{i \in [k]: \tilde{x}^i_\ell = 0} \frac{\theta_i}{g_S} \tilde{x}^i + \sum_{i \in [k]: \tilde{x}^i_\ell = 1} \frac{\theta_i}{g_S} \tilde{x}^i \tag{12}
$$

For $j = 0, 1$ let $(x^j, y^j) = \sum_{i \in [k]: \tilde{x}^i_\ell = j} \frac{\theta_i}{g_S} (\tilde{x}^i, \tilde{y}^i)$. Also let $\lambda_0 = \sum_{i \in [k]: \tilde{x}^i_\ell = 0} \frac{\theta_i}{g_S}$ and $\lambda_1 = \sum_{i \in [k]: \tilde{x}^i_\ell = 1} \frac{\theta_i}{g_S}$. Note that $\lambda_0 + \lambda_1 = \frac{1}{g_S}$. Constraint 10 is satisfied by Inequality 12. Also, for $j = 0, 1$ we have

$$
Ax^j + Gy^j = \sum_{i \in [k], \tilde{x}^i_\ell = j} \frac{\theta_i}{g_S} (A\tilde{x}^i + G\tilde{y}^i) \leq b \sum_{i \in [k], \tilde{x}^i_\ell = j} \frac{\theta_i}{g_S} = b\lambda_j. \tag{13}
$$

Hence, Constraints 5 and 6 hold. Constraint 9 also holds since x^0_ℓ is obviously 0 and $x^1_\ell = \sum_{i \in [k]: \tilde{x}^i_\ell = 1} \frac{\theta_i}{g_S} = \lambda_1$. The rest of the constraints trivially hold. $\square$

This concludes the proof. $\square$

We now show if x' in the statement of of Lemma 1 is partially integral, we can find solutions with more integral components.

Lemma 2. *Given $(x', y') \in \mathcal{D}(P)$, such that $x'_1, \ldots, x'_{\ell-1} \in \{0,1\}$ for some $\ell \geq 1$, we can find in polynomial time vector $(\hat{x}^0, \hat{y}^0), (\hat{x}^1, \hat{y}^1)$ and scalars $\gamma_0, \gamma_1 \in [0,1]$ such that*

(i) $\gamma_0 + \gamma_1 \geq \frac{1}{g_S}$,

(ii) $(\hat{x}^0, \hat{y}^0)$ *and* $(\hat{x}^1, \hat{y}^1)$ *are in* $\mathcal{D}(P)$,

(iii) $\hat{x}^0_\ell = 0$ *and* $\hat{x}^1_\ell = 1$,

(iv) $\gamma_0 \hat{x}^0 + \gamma_1 \hat{x}^1 \leq x'$,

(v) $\hat{x}^i_j \in \{0,1\}$ *for* $i = 0,1$ *and* $j = 1, \ldots, \ell - 1$.

Proof. By Lemma 1 we can find $(\bar{x}^0, \bar{y}^0)$, $(\bar{x}^1, \bar{y}^1)$, λ_0 and λ_1 that satisfy (i), (ii), (iii), and (iv). We define $\hat{x}^0$ and $\hat{x}^1$ as follows. For $i = 0, 1$, for $j = 1, \ldots, \ell - 1$, let $\hat{x}^i_j = \lceil \bar{x}^i_j \rceil$, for $j = \ell, \ldots, n$ let $\hat{x}^i_j = \bar{x}^i_j$. We now show that $(\hat{x}^0, \bar{y}^0)$, $(\hat{x}^1, \bar{y}^1)$, λ_0, and λ_1 satisfy all the conditions. Note that conditions (i), (ii), (iii), and (v) are trivially satisfied. Thus we only need to show (iv) holds. We need to show that $\lambda_0 \hat{x}^0_j + \lambda_1 \hat{x}^1_j \leq g_S x'_j$. If $j = \ell, \ldots, n$, then this clearly holds. Hence, assume $j \leq \ell - 1$. By the property of x' we have $x'_j \in \{0,1\}$. If $x'_j = 0$, then by Constraint 10 we have $\bar{x}^0_j = \bar{x}^1_j = 0$. Therefore, $\hat{x}^i_j = 0$ for $i = 0, 1$, so (iv) holds. Otherwise if $x'_j = 1$, then we have

$$\lambda_0 \hat{x}^0_j + \lambda_1 \hat{x}^1_j \leq \lambda_0 + \lambda_1 \leq 1 \leq x'_j.$$

Therefore (v) holds. □

The FDT algorithm grows a tree similar to the classic branch-and-bound search tree for integer programs. Each node represents a partially integral vector $(\bar{x}, \bar{y})$ in $\mathcal{D}(P)$ together with a multiplier $\bar{\lambda}$. The solutions contained in the nodes of the tree become progressively more integral at each level. Leaves of the FDT tree contain solutions with integer values for all the x variables that dominate a point in S. We will later see how we can turn these into points in S.

Let us define the FDT algorithm more formally. The algorithm maintains nodes L_i in iteration i of the algorithm. The nodes in each level L_i are progressively more integral. The points in the leaves of the FDT, or L_n, are points in $\mathcal{D}(P)$ that are integral for all integer variables.

Lemma 3. *There is a polynomial time algorithm that produces sets $L_0, \ldots, L_n$ of pairs of $(x, y) \in \mathcal{D}(P)$ together with multipliers λ with the following properties for $i = 0, \ldots, n$.*

a. If $(x, y) \in L_i$, then $x_j \in \{0,1\}$ for $j = 1, \ldots, i$, i.e. the first i coordinates of a solution in level i are integral.

b. $\sum_{[(x,y),\lambda] \in L_i} \lambda \geq \frac{1}{g^i_S}$.

c. $\sum_{[(x,y),\lambda]\in L_i} \lambda x \leq x^*$.

d. $|L_i| \leq n$.

Proof. We prove this lemma using induction but one can clearly see how to turn this proof into a polynomial time algorithm. Let L_0 be the set that contains a single node (*root of the FDT tree*) with (x^*, y^*) and multiplier 1. It is easy to check all the requirements in the lemma are satisfied for this choice.

Suppose by induction that we have constructed sets $L_0, \ldots, L_i$. Let the solutions in L_i be (x^j, y^j) for $j = 1, \ldots, k$ and λ_j be their multipliers, respectively. For each $j \in 1, \ldots, k$ by Lemma 2 (setting $(x', y') = (x^j, y^j)$ and $\ell = i$) we can find $(x^{j0}, y^{j0}), (x^{j1}, y^{j1})$ and λ_j^0, λ_j^1 with the properties (i) to (v) in Lemma 2. Define L' to be the set of node with solutions $(x^{j0}, y^{j0}), (x^{j1}, y^{j1})$ and multipliers $\lambda_j\lambda_j^0$, $\lambda_j\lambda_j^1$, respectively, for $j = 1, \ldots, \ell$.

Notice that for each $j \in \{1, \ldots, k\}$ by property (v) in Lemma 2 we have $x_\ell^{j0}, x_\ell^{j1} \in \{0, 1\}$ for $\ell = 0, \ldots, i+1$. We also have

$$\begin{aligned}
\sum_{[(x,y),\lambda]\in L'} \lambda &= \sum_{j=1}^{k} \lambda_j\lambda_j^0 + \lambda_j\lambda_j^1 \\
&\geq \sum_{j=1}^{k} \frac{\lambda_j}{g_S} && \text{(By property (i) in Lemma 2)} \\
&\geq \frac{1}{g_S} \sum_{[(x,y),\lambda]\in L_i} \lambda \\
&\geq \frac{1}{g_S^{i+1}}. && \text{(By induction hypothesis)}
\end{aligned}$$

Also

$$\begin{aligned}
\sum_{[(x,y),\lambda]\in L'} \lambda x &= \sum_{j=1}^{k} \lambda_j \left(\lambda_j^0 x^{j0} + \lambda_j^1 x^{j1}\right) \\
&\leq \sum_{j=1}^{k} \lambda_j x^j && \text{(By property (iv) in Lemma 2)} \\
&\leq \sum_{[(x,y),\lambda]\in L_i} \lambda x \\
&\leq x^*. && \text{(By induction hypothesis)}
\end{aligned}$$

Set L' seems like a suitable candidate for L_{i+1}, however we might have $|L'| > n$. We call the following linear programming Pruning(L'). Let $L' = \{[(x^1, y^1), \lambda_1], \ldots, [(x^k, y^k), \lambda_k]\}$. The variables of Pruning(L') is a scalar variable θ_j for each node j in L'.

$$\text{Pruning}(L') \qquad \max \quad \sum_{j=1}^{k} \theta_j \tag{14}$$

$$\text{s.t.} \quad \sum_{j=1}^{k} \theta_j x^j \leq x^* \tag{15}$$

$$\theta \geq 0 \tag{16}$$

Notice that $\theta = \lambda$ is in fact a feasible solution to Pruning(L'). Let θ^* be the optimal vertex point solution to this LP. Since the problem is in $\mathbb{R}^k$, θ^* has to satisfy k linearly independent constraints at equality. However, there are only n constraints of type 15. Therefore, there are at most n coordinates of θ^*_i that are non-zero. We claim that L_{i+1} which consists of (x^j, y^j) for $j = 1, \dots, k$ and their corresponding multipliers θ^*_j satisfy the properties in the statement of the lemma. Notice that, we can discard the nodes in L_{i+1} that have $\theta^*_j = 0$, so $|L_{i+1}| \leq n$. Also, since θ^* is optimal and λ is feasible for Pruning(L'), we have $\sum_{j=1}^{k} \theta^*_j \geq \sum_{j=1}^{k} \lambda_j \geq \frac{1}{g_S^{i+1}}$. $\square$

The leaves of the FDT tree, or L_n have the property that every solution (x, y) in L_n has $x \in \{0, 1\}^n$ and $(x, y) \in \mathcal{D}(P)$. Our goal to now replace these leaves solutions in $\mathcal{D}(P)$ with solutions in S. To this end, we need to reduce some variables with value of 1 into variables of value 0. We introduce a procedure that given a point $(x, y) \in \mathcal{D}(P)$ with $x \in \{0, 1\}^n$ outputs a point $z \leq x$ with $z \in \{0, 1\}^n$ such that (z, w) is in S for some vector $w \in \mathbb{R}^p$.

The algorithm "fixes" the variables iteratively, from x_1 to x_n. Suppose we run the algorithm for $\ell \in \{0, \dots, n-1\}$ iterations and we have $(x^\ell, y^\ell) \in \mathcal{D}(P)$ such that $x_i \in \{0, 1\}$ for $i = 1, \dots, \ell$. Now consider the following linear programming. The variables of this LP are the $z \in \mathbb{R}^n$ variables and $w \in \mathbb{R}^p$.

$$\text{DominantToFeasbile}(x^\ell) \qquad \min \quad z_{\ell+1} \tag{17}$$

$$\text{s.t.} \quad Az + Gw \geq b \tag{18}$$

$$z_j = x^\ell_j \quad \forall\, j \leq \ell \tag{19}$$

$$z_j \leq x^\ell_j \quad \forall\, j \geq \ell + 1 \tag{20}$$

$$z \geq 0 \tag{21}$$

If the optimal value to DominantToFeasbile(x^ℓ) is 0, then let $x^{\ell+1}_{\ell+1} = 0$. Otherwise if the optimal value is strictly positive let $x^{\ell+1}_{\ell+1} = 1$. Let $x^{\ell+1}_j = x^\ell_j$ for $j \in \{1, \dots, n\} \setminus \{\ell + 1\}$. This procedure is summarized in Algorithm 1.

Algorithm 1: IP dominant to IP feasible procedure

Input: $(x, y) \in \mathcal{D}(S)$

Output: $(x^n, y^n) \in S$, $x^n \leq x$

$x^0 \leftarrow x$
for $\ell = 0$ ***to*** $n-1$ **do**
  $x^{\ell+1} \leftarrow x^\ell$
  $\eta \leftarrow$ optimal value of DominantToFeasbile(x^ℓ)
  $y^{\ell+1} \leftarrow$ optimal solution for w variables in DominantToFeasbile(x^ℓ)
  if $\eta = 0$ **then**
    $x^{\ell+1}_{\ell+1} \leftarrow 0$
  else
    $x^{\ell+1}_{\ell+1} \leftarrow 1$
  end
end

Theorem 3. *Given $(x, y) \in \mathcal{D}(P)$ Algorithm 1 correctly finds $(x^n, y^n) \in S$ in polynomial time.*

Proof. First, we need to show that in any iteration $\ell \in \{0, \dots, n-1\}$ of Algorithm 1 the DominantToFeasbile(x^ℓ) is feasible. We show something stronger. For $\ell = 0, \dots, n-1$ let

$$\mathrm{LP}^{(\ell)} = \{(z, w) \in P \ : \ z \leq x^\ell \text{ and } z_j = x^\ell_j \text{ for } j = 1, \dots, \ell\}, \text{ and}$$
$$\mathrm{IP}^{(\ell)} = \{(z, w) \in \mathrm{LP}^{(\ell)} \ : \ z \in \{0,1\}^n\}.$$

Notice that if $\mathrm{LP}^{(\ell)}$ is an non-empty set then DominantToFeasbile(x^ℓ) is feasible. We show by induction on ℓ that $\mathrm{LP}^{(\ell)}$ and $\mathrm{IP}^{(\ell)}$ are not empty sets for $\ell = 0, \dots, n-1$. First notice that $\mathrm{LP}^{(0)}$ is clearly feasible since by definition $(x^0, y^0) \in \mathcal{D}(P)$, meaning there exists $(z, w) \in P$ such that $z \leq x^0$. By Theorem 2 there exists $(\tilde{z}^i, \tilde{w}^i) \in S$ and $\theta_i \geq 0$ for $i = 1, \dots, k$ such that $\sum_{i=1}^k \theta_i = 1$ and $\sum_{i=1}^k \theta_i \tilde{z}^i \leq g_S z$. Hence, $\sum_{i=1}^k \theta_i \tilde{z}^i \leq g_S z \leq g_S x^0$. So if x^0_j is zero, then $\sum_{i=1}^k \theta_i \tilde{z}^i_j = 0$, which implies that $z^i_j = 0$ for all $i = 1, \dots, k$ and $j = 1, \dots, n$ where $x^0_j = 0$. Hence, $z^i \leq x^0$ for $i = 1, \dots, k$. Therefore $\mathrm{IP}^{(0)} \neq \emptyset$.

Now assume $\mathrm{IP}^{(\ell)}$ are all non-empty for some $\ell \in \{0, \dots, n-2\}$. Since $\mathrm{IP}^{(\ell)} \subseteq \mathrm{LP}^{(\ell)}$ we have $\mathrm{LP}^{(\ell)} \neq \emptyset$ and hence the DominantToFeasbile(x^ℓ) has an optimal solution (z^*, w^*).

We consider two cases. In the first case, we have $z^*_{\ell+1} = 0$. In this case we have $x^{\ell+1}_{\ell+1} = 0$. Since $z^* \leq x^{\ell+1}$, we have $z^* \in \mathrm{LP}^{(\ell+1)}$. Also, $(z^*, w^*) \in P$. By Theorem 2 there exists

$(\tilde{z}^i, \tilde{w}^i) \in S$ and $\theta_i \geq 0$ for $i = 1, \ldots, k$ such that $\sum_{i=1}^k \theta_i = 1$ and $\sum_{i=1}^k \theta_i \tilde{z}^i \leq g_S z^*$. We have

$$\sum_{i=1}^{k} \theta_i \tilde{z}^i \leq g_S z^* \leq g_S x^{\ell+1} \tag{22}$$

So for $j \in \{1, \ldots, n\}$ where $x_j^{(\ell+1)} = 0$, we have $z_j^i = 0$ for $i = 1, \ldots, k$. Hence, $z^i \leq x^{(\ell+1)}$ for $i = 1, \ldots, k$. Hence, there exists $(z, w) \in S$ such that $z \leq x^{\ell+1}$. We claim that $(z, w) \in \mathrm{IP}^{(\ell+1)}$. If $(z, w) \notin \mathrm{IP}^{(\ell+1)}$ we must have $1 \leq j \leq \ell$ such that $z_j < x_j^{\ell+1}$, and thus $z_j = 0$ and $x_j^{\ell+1} = 1$. Without loss of generality assume j is minimum number satisfying $z_j < x_j^{\ell+1}$. Consider iteration j of Algorithm 1. Notice that $z \leq x^{\ell+1} \leq x^j$. We have $x_j^j = 1$ which implies when we solved DominantToFeasbile(x^{j-1}) the optimal value was strictly larger than zero. However, (z, w) is a feasible solution to DominantToFeasbile(x^{j-1}) and gives an objective value of 0. This is a contradiction, so $(z, w) \in \mathrm{IP}^{(\ell+1)}$.

Now for the second case, assume $z_{\ell+1}^* > 0$. We have $x_{\ell+1}^{\ell+1} = 1$. Notice that for each point $z \in \mathrm{LP}^{(\ell)}$ we have $z_{\ell+1} > 0$, so for each $z \in \mathrm{IP}^{(\ell)}$ we have $z_{\ell+1} > 0$, i.e. $z_{\ell+1} = 1$. This means that $z \in \mathrm{IP}^{(\ell+1)}$, and $\mathrm{IP}^{(\ell+1)} \neq \emptyset$.

Now consider (x^n, y^n). Let (z, y^n) be the optimal solution to $\mathrm{LP}^{(n-1)}$. If $x^n = 0$, we have $x^n = z$, which implies that $(x^n, y^n) \in P$, and since $x^n \in \{0, 1\}^n$ we have $(x^n, y^n) \in S$. If $x^n = 1$, it must be the case that $z_n > 0$. By the argument above there is a point $(z', w') \in \mathrm{IP}^{(n-1)}$. We show that $x^n = z'$. Observe that for $j = 1, \ldots, n-1$ we have $z'_j = x_j^{n-1} = x_j^n$. We just need to show that $z'_j = 1$. Assume $z'_j = 0$ for contradiction, then $(z', w') \in \mathrm{LP}^{(n-1)}$ has objective value of 0 for DominantToFeasbile(x^{n-1}), this is a contradiction to (z, w) being the optimal solution. $\square$

This completes the FDT algorithm.

Algorithm 2: Fractional Decomposition Tree Algorithm

Input: $P = \{(x,y) \in \mathbb{R}^{n \times p} : Ax + Gy \geq b\}$ and $S = \{(z,w) \in P : z \in \{0,1\}^n\}$ such that $g_S = \max_{c \in \mathbb{R}^n_+} \frac{\min_{(x,y)\in} cx}{\min_{(x,y)\in P} cx}$ is finite, $(x^*, y^*) \in P$

Output: $(z^i, w^i) \in S$ and $\lambda_i \geq 0$ for $i = 1, \ldots, k$ such that $\sum_{i=1}^k \lambda_i = 1$, and $\sum_{i=1}^k \lambda_i z^i \leq g^n x^*$

$L^0 \leftarrow [(x^*, y^*), 1]$
for $i = 1$ ***to*** n **do**
  $L' \leftarrow \emptyset$
  for $[(x,y), \lambda] \in L^i$ **do**
    Apply Lemma 2 to obtain $[(\hat{x}^0, \hat{y}^0), \gamma_0]$ and $[(\hat{x}^1, \hat{y}^1), \gamma_1]$
    $L' \leftarrow L' \cup \{[(\hat{x}^0, \hat{y}^0), \gamma_0]\} \cup \{[(\hat{x}^1, \hat{y}^1), \gamma_1]\}$
  end
  Apply Lemma 3 to L' to obtain L^{i+1}.
end
for $[(x,y), \lambda] \in L^n$ **do**
  Apply Algorithm 1 to (x,y) to obtain $(z,w) \in S$
  $F \leftarrow F \cup \{[(z,w), \lambda]\}$
end
return F

3 TO DO

- Implement FDT on SND 0,1,2 and point-to-point SND 0,1,2
- Implement of TAP
- Write paper for IPCO

References

[CCZ14] Michele Conforti, Gerard Cornuejols, and Giacomo Zambelli. *Integer Programming*. Springer Publishing Company, Incorporated, 2014.

[CV04] Robert Carr and Santosh Vempala. On the held-karp relaxation for the asymmetric and symmetric traveling salesman problems. *Mathematical Programming*, 100(3):569–587, Jul 2004.

[GJ79] M.R. Garey and D.S. Johnson. *Computers and Intractability – A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA, 1979.

[MS14] Monaldo Mastrolilli and Georgios Stamoulis. Bi-criteria and approximation algorithms for restricted matchings. *Theoretical Computer Science*, 540-541:115 – 132, 2014. Combinatorial Optimization: Theory of algorithms and Complexity.

[Sch03] A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency*. Springer, 2003.