# Large Scale Mixed-Integer Optimization: a Solution Method with Supply Chain Applications

Robin Vujanic, Peyman Mohajerin Esfahani, Paul Goulart and Manfred Morari

***Abstract*— In this paper we investigate lagrangian duality for a class of mixed integer programs which is of wide practical interest as it appears in many application domains, such as power systems or logistics. For this problem structure, we provide a new solution method that is simple to implement, is distributable and has convergence and performance guarantees. To obtain it, we borrow ideas and results from the convex optimization field, and exploit the special geometric features arising from the specific structure studied. The performance bound indicates that the quality of the solutions recovered improves as the size of the problem increases, making it particularly useful for very large instances. We verify the efficacy of the proposed method on industrial-sized instances of a problem stemming from supply chain optimization.**

## I. Introduction

In this paper we investigate mixed integer optimization programs in the form

$$\left\{\begin{array}{ll} \underset{x}{\text{minimize}} & \sum_{i\in I} c_i^\top x_i \\ \text{subject to} & \sum_{i\in I} H_i x_i \leq b \\ & x_i \in X_i \qquad \forall i \in I, \end{array}\right. \tag{$\mathcal{P}$}$$

in which the sets $X_i$ are mixed integer polyhedral sets that can be written as

$$X_i = \left\{ x \in \mathbb{R}^{r_i} \times \mathbb{Z}^{z_i} \;\middle|\; A_i x \leq d_i \right\},$$

for some given $A_i \in \mathbb{R}^{m_i \times n_i}$, where $n_i = r_i + z_i$, $d_i \in \mathbb{R}^{m_i}$, and the coupling constraints are determined by the matrices $H_i \in \mathbb{R}^{m \times n_i}$ and the vector $b \in \mathbb{R}^m$.

The structure given in $\mathcal{P}$ can be thought of as modeling problems in which a large number of local *subsystems* $X_i$, whose description can contain discrete variables, is coupled through a relatively small number of constraints budgeted by the resource vector $b$. We are particularly interested in large scale instances of this problem, i.e., instances for which the number of subsystems included – the cardinality of the index set $I$ – is significantly larger than $m$, the number of coupling constraints.

Such instances arise in a variety of application domains. The simplest examples are traditional combinatorial programs, such as the knapsack problem, in which $X_i = \{0,1\}$, $c_i \geq 0$ and $H_i \geq 0$. More detailed models appear, for instance, in power systems control. Here the subsystems are the generators to be controlled, integer variables arise due to start-up or shut-down costs, and the coupling constraints are related to the requirement that production must match the load [20]. Portfolio optimization problems for small investors, in which integer variables are used to encode various investment frictions, is another example application in which models fitting $\mathcal{P}$ have been proposed [4]. Problems structured as $\mathcal{P}$ are also considered in the literature on network utility maximization [16]. In this paper we study an application from supply chains related to the problem of partial shipments, discussed in [10], [3]. Finally, many problems that do not naturally possess the structure of $\mathcal{P}$ can be brought back to it by adequate permutations of the rows and columns of the constraint matrices. In [5], the authors discuss algorithms to automate this procedure.

Lagrangian duality is a particularly useful framework in this context, because it allows one to separate $\mathcal{P}$. However, in contrast to the convex case, duality in the mixed integer case is known to be generally unable to produce optimal solutions directly. In fact, primal solutions recovered may even be infeasible (we illustrate this issue with an example in Section II). It is for this reason that, often, duality in the mixed integer case is only used to produce tight lower bounds to the optimal objective of $\mathcal{P}$.

In this paper we show that, owing to its geometric structural properties, it is indeed possible to recover a good, feasible primal solution to $\mathcal{P}$ from its dual.

### *Previous Results and Current Contribution*

Duality for the specific structure in $\mathcal{P}$ has been already investigated. In [2] the authors show that as the size of the problem is increased by adding subsystems, the duality gap *vanishes*. This indicates that as the size of the problem increases, it more closely resembles a convex program. In practical applications, vanishing duality gap has been observed in the context of power systems [7]. It also appears in the literature for multistage stochastic programs [9], [8], where it is used to assert that for the problem structures of interest there, lagrangian duality is almost strong, but in which no further relation is derived in order to retrieve primal solutions. Another example is in the field of communications, in problems related to multicarrier systems optimization, where the duality gap arises due to non-convexity of the objective function rather than the presence of integer variables [21].

Here we further investigate duality for problem $\mathcal{P}$. The main contributions of this paper are the following:

- We propose a method to solve the mixed-integer program $\mathcal{P}$ with convergence guarantees, borrowing ideas and results from the convex literature, in particular a primal recovery scheme based on averaging.
- From an application point of view, we assess the performance of the proposed method on a problem stemming from supply chain optimization. General purpose solvers are unable to solve these models, and our results indicate that our proposed method is an attractive alternative.

The paper is structured as follows: in Section II we briefly

R. Vujanic, P. Mohajerin Esfahani, P. Goulart and M. Morari are with the Department of Information Technology and Electrical Engineering, Automatic Control Laboratory, Swiss Federal Institute of Technology, ETH Zurich, Switzerland. Contact: vujanicr(at)control.ee.ethz.ch

introduce duality and discuss special geometric properties arising from the specific structure of $\mathcal{P}$. Based on these geometric properties, in Section III we propose a solution algorithm with convergence and performance guarantees. In Section IV we verify its efficacy on a model stemming from supply chains.

*Notation*

Given an optimization problem $\mathcal{A}$, we indicate with $J^\star_{\mathcal{A}}$ its optimal objective value and with $J_{\mathcal{A}}(x)$ the performance of the solution $x$ with respect to the objective function of $\mathcal{A}$. For a given set $X$, we denote by $\mathrm{conv}(X)$ its convex hull and by $\mathrm{vert}(X)$ the set of vertices of $\mathrm{conv}(X)$. The inequality sign "$\geq$" used between vectors (or matrices) is always intended component-wise. In the algorithms, we use brackets to indicate iterations, e.g., $\lambda^{[k]}$ is the value of the variable $\lambda$ at iteration $k$. In order to avoid confusion of sub- and superscripts when describing particular subsystems we use parentheses, e.g., we denote by $(x_{\mathrm{LP}})_i$ the part of $(x_{\mathrm{LP}})$ related to subsystem $i \in I$. With $U[a,b]$ we denote the uniform distribution between $a$ and $b$. We indicate with $|I|$ the cardinality of the set $I$.

## II. Exploiting the special features of $\mathcal{P}$ using duality

In the Lagrangian duality framework, we relax (or dualize) the coupling constraints of $\mathcal{P}$ in the objective, leading to the following dual function

$$d(\lambda) \;\doteq\; \min_{x\in X}\left(\sum_{i\in I} c_i^\top x_i + \lambda^\top\Big(\sum_{i\in I} H_i x_i - b\Big)\right).$$

The dual function is known to provide lower bounds to the objective of $\mathcal{P}$, i.e., $d(\lambda) \leq J^\star_{\mathcal{P}}$ for all $\lambda \succeq 0$. We are then interested in the best (largest) lower bound duality can provide, which is why we pose the following *dual problem* (after some algebraic reorganization)

$$\left\{\begin{array}{ll}\sup\limits_{\lambda} & -\lambda^\top b + \sum\limits_{i\in I}\min\limits_{x_i\in X_i}\left(c_i^\top x_i + \lambda^\top H_i x_i\right)\\ \text{s.t.} & \lambda \succeq 0.\end{array}\right. \qquad (\mathcal{D})$$

When we solve the dual (outer maximization) problem in $\mathcal{D}$ using an iterative algorithm, at each iterate $\lambda^{[k]}$ we have to solve the following inner minimization problem

$$\sum_{i\in I}\min_{x_i\in X_i}\left(c_i^\top x_i + (\lambda^{[k]})^\top H_i x_i\right). \qquad (1)$$

The central object of this paper are the solutions of this inner problem (1). These are of substantial practical interest, because they are obtained as by–products of methods used to solve $\mathcal{D}$, and because they are computed in a distributed fashion. Notice in fact that solving the inner problem amounts to solving a collection of $|I|$ decoupled minimizations. We thus denote by $\mathcal{X}(\lambda)$ the set of minimizers of (1) for some $\lambda$. An *inner solution* is then any selection from $\mathcal{X}(\lambda)$, and it is denoted by $x(\lambda)$. Here we are particularly interested in the behavior of $x(\lambda)$ when $\lambda^{[k]} \to \lambda^\star$, where $\lambda^\star$ is an optimizer of the dual problem.

It is well known that for problems affected by a *duality gap* (i.e., for which $J^\star_{\mathcal{D}} < J^\star_{\mathcal{P}}$), these solutions are usually non-unique, suboptimal, even infeasible, and hence inadequate candidates for solving the primal problem. We illustrate this with an example.

*Example 1:* Consider the following problem:

$$\begin{array}{ll}\text{minimize} & -x_1\\ \text{subject to} & x_1 - x_2 \leq 0.5\\ & x_1 + x_2 \leq 1.5\\ & x_1 \in X_1, x_2 \in X_2\end{array} \qquad (2)$$

with $X_i = \{x_i \in \mathbb{Z} \,|\, 0 \leq x_i \leq 1\}$, $i = 1,2$. Figure 1(a) depicts its geometry. The only two feasible (and also optimal) points for this problem are $(x_1, x_2) = (0,0)$ and $(0,1)$. Relaxing the constraints $x_1 - x_2 \leq 0.5$ and $x_1 + x_2 \leq 1.5$, and introducing the corresponding multipliers $(\lambda_1, \lambda_2) \succeq 0$, leads to the dual function

$$d(\lambda) = \begin{cases} -0.5\lambda_1 + 0.5\lambda_2 - 1 & \lambda \in \textcircled{1}\\ -1.5\lambda_1 - 0.5\lambda_2 & \lambda \in \textcircled{2}\\ -0.5\lambda_1 - 1.5\lambda_2 & \lambda \in \textcircled{3}\\ +0.5\lambda_1 - 0.5\lambda_2 - 1 & \lambda \in \textcircled{4}, \end{cases}$$

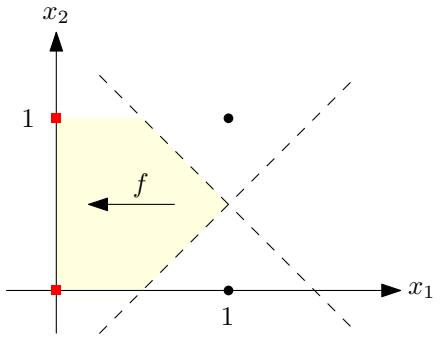
where the regions are arranged according to Figure 1(b). It can be seen that any point on the intersection of regions 1 and 4 attains the largest value for the dual function, and is therefore a solution for the dual problem, i.e.,

$$\begin{bmatrix}\lambda_1^\star\\ \lambda_2^\star\end{bmatrix} = \begin{bmatrix}1\\ 1\end{bmatrix}\cdot\theta, \quad \theta \in [0, 0.5]. \qquad (3)$$

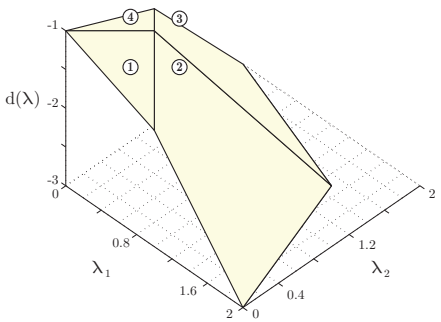
For any $\theta \in [0, 0.5)$ the set of primal solutions recovered is

$$\mathcal{X}_1(\lambda^\star) = \arg\min_{x_1\in X_1}\{x_1(\lambda_1^* + \lambda_2^* - 1)\} = \{1\}$$
$$\mathcal{X}_2(\lambda^\star) = \arg\min_{x_2\in X_2}\{x_2(-\lambda_1^* + \lambda_2^*)\} = \{0,1\}.$$

Thus, at these dual optimizers, the inner solution is non-unique, and for any selection made, e.g. $x_1(\lambda^*) = 1 \in \mathcal{X}_1(\lambda^*)$ and $x_2(\lambda^\star) = 0 \in \mathcal{X}_2(\lambda^\star)$, the pair $(x_1(\lambda^\star), x_2(\lambda^\star))$ is infeasible for problem (2).




(a) Primal Problem




(b) Dual Function

Fig. 1. Illustrations of Example 1.

Our objective in this paper is to provide a method for recovering *good* and *feasible* solutions for the particular structure of $\mathcal{P}$ from the inner solutions $x(\lambda)$.

### A. A convexification of $\mathcal{P}$ and its structure

It is known that lagrangian duality for mixed integer programs is tightly related to the following convexification of the primal problem

$$\left\{\begin{array}{ll} \underset{x}{\text{minimize}} & \sum\limits_{i\in I} c_i^\top x_i \\ \text{subject to} & \sum\limits_{i\in I} H_i x_i \leq b \\ & x_i \in \operatorname{conv}(X_i) \qquad \forall i \in I. \end{array}\right. \tag{$\mathcal{P}_{\mathrm{LP}}$}$$

It is in fact well known that $\mathcal{P}_{\mathrm{LP}}$ satisfies the (non-obvious) relation $J^\star_{\mathcal{P}_{\mathrm{LP}}} = J^\star_{\mathcal{D}}$ [11, Thm. 1d]. Note that we do not usually have direct access to $\mathcal{P}_{\mathrm{LP}}$ because we do not have an explicit description of $\operatorname{conv}(X_i)$. It must also be emphasized that while $\mathcal{P}_{\mathrm{LP}}$ is a linear program, it *does not* coincide with the traditional linear relaxation obtained when relaxing the integrality constraints in $X_i$ to an interval. In fact, $\mathcal{P}_{\mathrm{LP}}$ is usually tighter [11, Thm. 1a].

In this paper we will exploit the fact that it is possible to recover a solution to $\mathcal{P}_{\mathrm{LP}}$, denoted by $x^\star_{\mathrm{LP}}$, when solving $\mathcal{D}$. For this, we borrow a method from convex optimization which was initially proposed in [18]. We will outline this method in the next section. For the moment, notice that a solution to the convexification $\mathcal{P}_{\mathrm{LP}}$ may, in general, not necessarily satisfy the local constraints (in particular, the integrality conditions on the discrete variables). However, in the following Theorem we establish that, for the specific structure of $\mathcal{P}$, a vertex solution of $\mathcal{P}_{\mathrm{LP}}$ is indeed useful as it may violate the local constraints only for *a few* subsystems.

*Theorem 1:* Let $\tilde{x}$ be a vertex of $\mathcal{P}_{\mathrm{LP}}$. Then $\tilde{x}_i \in \operatorname{conv}(X_i)$ for all $i \in I$, and there exists $I_1 \subseteq I$, with cardinality at least $|I_1| \geq |I| - m - 1$, so that $\tilde{x}_i \in X_i$ for all $i \in I_1$.

*Proof:* See Appendix A. ∎

Note that Theorem 1 holds for any vertex of the feasible set of $\mathcal{P}_{\mathrm{LP}}$, and thus in particular also for any vertex optimizer $x^\star_{\mathrm{LP}}$. In the next subsection we show how to recover $x^\star_{\mathrm{LP}}$, while in Section III we exploit it in order to recover a *good* feasible solution for the original $\mathcal{P}$.

### B. Averaging scheme

In this section we discuss a method to obtain a solution $x^\star_{\mathrm{LP}}$ to $\mathcal{P}_{\mathrm{LP}}$. It is based on a method developed for convex problems, which was first proposed in [18], was then further developed in [12] and has been recently extended by [15]. Its theoretical properties in the convex case are summarized in [1]. It consists in deploying a combination of the subgradient method and an averaging scheme for the inner solutions.

The subgradient method is a well known method used to solve duals. According to it, given an initialization $\lambda^{[1]}$, the subsequent dual iterates $\lambda^{[k]}$ are updated as

$$\lambda^{[k+1]} = \mathrm{P}_+\left(\lambda^{[k]} + s^{[k]} \cdot \gamma^{[k]}\right), \tag{4}$$

in which $\mathrm{P}_+(\cdot)$ denotes the projection onto $\mathbb{R}^m_+$, $s^{[k]}$ is the steplength and $\gamma^{[k]}$ is a subgradient of the dual function. When solving a dual problem, a valid subgradient is given by the residuals of the current inner solution, i.e., $\gamma^{[k]} = \sum_{i\in I} H_i x_i(\lambda^{[k]}) - b$. Following the scheme proposed in [1], we use the steplength rule

$$s^{[k]} = \frac{\alpha}{k}, \tag{5}$$

and as we produce the dual iterates $\lambda^{[k]}$ during the subgradient optimization (4), we construct an average of the inner solutions encountered:

$$\bar{x}^{[k]} = \frac{1}{k}\sum_{j=1}^{k} x(\lambda^{[j]}). \tag{6}$$

Surprisingly[1], the sequence $\left\{\bar{x}^{[k]}\right\}$ accumulates at the optimizers of $\mathcal{P}_{\mathrm{LP}}$ [1, Cor. 5]. For the sake of simplicity, we now make the following assumption.

*Assumption 1:* The optimization problem $\mathcal{P}_{\mathrm{LP}}$ has the unique solution $x^\star_{\mathrm{LP}}$.

Necessary and sufficient conditions for the uniqueness of solutions to linear programs are discussed in [14]. The exceptional case in which Assumption 1 is not satisfied can be avoided by adding negligible perturbation terms to the cost vector. Assumption 1 eliminates the need to talk about accumulation points, enabling the following concise convergence result.

*Theorem 2 (primal and dual convergence):* Suppose that the subgradient method (4) is applied to the dual problem $\mathcal{D}$. Then $\lambda^{[k]} \to \lambda^\star \in \Lambda^\star$, where $\Lambda^\star$ is the set of optimal dual solutions. Further, if Assumption 1 holds, then $\bar{x}^{[k]} \to x^\star_{\mathrm{LP}}$.

*Proof:* Dual convergence $\lambda^{[k]} \to \lambda^\star \in \Lambda^\star$ is proven in [1, Thm. 2]. For primal convergence, see Appendix B. ∎

*Remark 1:* Dual convergence asserted in Theorem 2 can be generalized to any steplength rule which satisfies

$$s^{[k]} \to 0, \qquad \sum_{k=1}^{\infty} s^{[k]} = \infty, \qquad \sum_{k=1}^{\infty} (s^{[k]})^2 < \infty, \tag{7}$$

see [1, Thm. 3]. Other averaging schemes that generalize (6) can be found in [17].

## III. A Solution algorithm with convergence and performance guarantees

In this section we propose an algorithm to recover a good solution to $\mathcal{P}$. We take advantage of the properties of vertex solutions of $\mathcal{P}_{\mathrm{LP}}$ established in Theorem 1, and the primal solution recovery scheme of Theorem 2. In order to simplify the analysis, we first make the following assumption.

*Assumption 2:* The local systems are such that, for all $i \in I$, $0 \in X_i$ and $H_i x_i \geq 0$ for all $x_i \in X_i$.

Assumption 2 holds for a number of traditional combinatorial problems (e.g. the knapsack problem) as well as more refined models used in practice, such as portfolio optimization problems for small investors, for which models fitting $\mathcal{P}$ have been proposed [4]. Another example application is the supply chain problem presented in Section IV.

Our proposed method to solve $\mathcal{P}$ under Assumptions 1, and 2 is reported in Algorithm 1. It is a two–phases method.

In a first phase, using averaging we construct $x^\star_{\mathrm{LP}}$ and detect $I_1$, the subset of subsystems whose solution already

[1]Recall that, as shown in Example 1, the iterates $x(\lambda^{[k]})$ are generally non-unique, suboptimal, or even infeasible points.

**Algorithm 1** Primal Solution Recovery

Initialization:

$k = 1$

$\lambda^{[1]} = 0$

Construction of the Ergodic Sequence:

**while** $k \leq k_{\max}$ **do**

$x_i(\lambda^{[k]}) = \arg\min_{x_i \in X_i} (c_i + \lambda^{[k]} H_i) x_i$

$\bar{x}_i^{[k]} = \frac{1}{k} \sum_{j=1}^{k} x_i(\lambda^{[j]})$

$\gamma^{[k]} = \sum_{i \in I} H_i x_i(\lambda^{[k]}) - b$

$\lambda^{[k+1]} = \max(\lambda^{[k]} + \frac{\alpha}{k} \gamma^{[k]}, 0)$

$k = k + 1$

**end while**

Rectification of the Solution:

determine the partitioning of $I = I_1 \cup I_2$, such that for all $i \in I_1$, $\bar{x}_i^{[k_{\max}]} \in X_i$ (in particular, it satisfies integrality). $|I_1| \geq |I| - m - 1$.

**for** $i \in I_1$ **do**

$\hat{x}_i^\star = \bar{x}_i^{[k_{\max}]}$

**end for**

$\rho = \sum_{i \in I_1} H_i \hat{x}_i^\star$

$$(\hat{x}^\star)_{i \in I_2} = \begin{cases} \arg\min & \sum_{i \in I_2} c_i x_i \\ \text{s.t.} & \sum_{i \in I_2} H_i x_i \leq b - \rho \\ & x_i \in X_i, i \in I_2 \end{cases} \qquad (\mathcal{P}_{\text{FIX}})$$

satisfies all the local constraints (including the integrality requirements). We let the averaging run for $k_{\max}$ iterations, where $k_{\max}$ is a parameter chosen such that the averaging sequence has settled. Recall that the convergence of this sequence is guaranteed by Theorem 2.

In the second phase, we reoptimize over the smaller set of subsystems indexed by $I_2 = I \setminus I_1$. The second optimization is low dimensional since it is guaranteed to entail at most $m+1$ subsystems. The input of the algorithm is the data of the problem, i.e. the tuple $(c_i, H_i, A_i, d_i)_{i \in I}$ for each subsystem, and the resource vector $b \in \mathbb{R}^m$. The output is $\hat{x}^\star$, a feasible solution to $\mathcal{P}$ that satisfies the performance bound given in the following Theorem.

*Theorem 3:* Under Assumptions 1 and 2, the solution $\hat{x}^\star$ produced by Algorithm 1 is a feasible solution to $\mathcal{P}$ which satisfies the following performance bound:

$$J_{\mathcal{P}}(\hat{x}^\star) - J_{\mathcal{P}}^\star \leq (m+1) \max_{i \in I} \max_{x_i \in X_i} c_i^\top x_i. \qquad (8)$$

*Proof:* See Appendix B. ∎

According to the performance bound (8), if $J_{\mathcal{P}}^\star$ grows linearly as we increase the size of the problem $|I|$, and if the sets $\{X_i\}_{i \in I}$ are uniformly bounded, then

$$\frac{J(\hat{x}_i^\star) - J_{\mathcal{P}}^\star}{J_{\mathcal{P}}^\star} \to 0 \quad \text{as} \quad |I| \to \infty, \qquad (9)$$

indicating that the quality of the solutions recovered improves as the problem size is increased.

*Remark 1:* The reoptimization in $\mathcal{P}_{\text{FIX}}$ is not necessary, one can safely pick $\hat{x}_i^\star = 0$ for $i \in I_2$. The advantage of opting for this variation is that then the algorithm is *fully* distributed. And while the performance of the solutions is decreased in this case, it still satisfies the bound (8).

## IV. Application Example: Partial Shipments

The main objective of this section is to assess how the proposed method performs on a mixed integer optimization problem that is of practical interest: the problem of partial shipments.

In this problem setting, a distributor of some products has to supply multiple customers. Due to uncertainties in the demand and high storage costs, as well as restrictions on manufacturing capacities (especially true for seasonal items such as ski equipment or pharmaceuticals), the products available in the inventory that are ready for shipping is less than the total demand. Under these circumstances, more often than not, the distributor chooses to satisfy partial orders of more customers, instead of fully satisfying only a few of them [10]. On the other hand, these shipments cannot be too small, due to transportation costs and the additional paperwork and tracking costs. Distributors are therefore faced with the problem of allocating the available product inventories to customers in the presence of these shipping restrictions [10].

### A. Model for the Optimization Problem

We use the formulation proposed in [10], which we summarize here for completeness. We are given a demand of $M$ products from $N$ customers ($N > M$). $D_j^i$ is the demand of product $j$ from customer $i$, while $I_j$ is the amount of product $j$ available in the inventory. If a shipment is made, it must at least amount to the $\beta$-fraction of the total demand over all products. The optimization variables are $w^i \in \{0, 1\}$, which decide whether customer $i$ gets a partial shipment, and $S_j^i$, which is the amount of product $j$ shipped to customer $i$.

The rewards are composed of two components. A fixed reward amounting to $K^i$ is obtained if a shipment is made to customer $i$. This can encode the customer appreciation for receiving shipment, even if partial, or the weight that the distributor attaches to each customer. There is an additional revenue $r_j^i$ that the supplier retrieves based on what fraction $S_j^i$ of the total demand $D_j^i$ is shipped.

We can formulate this problem as the following optimization program

$$\underset{w^i, S_j^i}{\text{minimize}} \quad -\sum_{i=1}^{|N|} K^i w^i - \sum_{i=1}^{|N|} \sum_{j=1}^{|M|} r_j^i \frac{S_j^i}{D_j^i} \qquad (10a)$$

$$\text{subject to} \quad \sum_{i=1}^{|N|} S_j^i \leq I_j, \quad j \in M \qquad (10b)$$

$$\sum_{j=1}^{|M|} S_j^i \geq \beta \cdot w^i \sum_{j=1}^{|M|} D_j^i, \quad i \in N \qquad (10c)$$

$$0 \leq S_j^i \leq w^i \cdot D_j^i, \quad i \in N,\ j \in M \qquad (10d)$$

$$w^i \in \{0, 1\}, \quad i \in N. \qquad (10e)$$

| $\|N\|$ | $\|M\|$ | $I_j$ |
|---|---|---|
| 100 | 25 | U[1000-2500] |
| 300 | 75 | U[4000-5000] |
| 500 | 50 | U[7000-8500] |
| 600 | 50 | U[8500-9500] |
| 1000 | 100 | U[14500-15500] |

TABLE I

SIZES OF THE PROBLEM STUDIED AND THE CORRESPONDING INVENTORIES.

### B. Applying the Proposed Method and Results

The partial shipment problem (10) is NP-hard, which is proven by reduction to a knapsack problem, see [10, Lemma 2.1]. It is also known that greedy strategies perform poorly on it [10]. We tried to solve it using general purpose solvers, but these are also inadequate for the larger instances (see Table II). In [10], the authors propose a heuristic to obtain good partial assignments that can be fixed, thus reducing the number of integer variables to be solved. The method used to retrieve the assignment is relatively sophisticated and must be adapted when the model is subject to minor changes. Even under these circumstances, solving the problem remains a challenge: reported solve times for medium sized instances are up to 6 hours, with an average optimality gap of the recovered solutions of 6.2%.

We tackle the problem using our proposed method. Here duality can be exploited by relaxing the budget constraint (10b). This leads to the following dual problem

$$\begin{aligned} \sup_{\lambda\geq 0} \quad \min_{w^i, S_j^i} \quad & -\sum_{i=1}^{|N|} K^i w^i - \sum_{i=1}^{|N|}\sum_{j=1}^{|M|}\left(\frac{r_j^i}{D_j^i} - \lambda_j\right) S_j^i - \sum_{j=1}^{|M|} \lambda_j I_j \\ & \sum_{j=1}^{|M|} S_j^i \geq \beta \cdot w^i \sum_{j=1}^{|M|} D_j^i \\ & 0 \leq S_j^i \leq w^i \cdot D_j^i \\ & w^i \in \{0,1\}\,, \end{aligned} \tag{11}$$

whose inner problem is decoupled in $N$ separate optimization programs, one for each customer. We apply Algorithm 1 using this relaxation. For the tests, we generate 5 instances of the problem using random parameters sampled from a uniform distribution over the following ranges: $D_j^i = U[1-100]$, $K^i = U[1-100]$, $r_j^i = U[1-15]$, $\beta = 0.6$. The budgets $I_j$ are assigned according to Table I. These parameter values are taken from [10], and the sizes of the problems considered are representative for industrial instances related to a pharmaceutical company. The computations are carried on a Laptop PC with 4GB of RAM and a 2.67GHz processor.

The outer (maximization) problem in (11) is solved using a subgradient method. Before activating the steplength rule "$\alpha/k$", we first perform 50 iterations with fixed steplength, in order to localize a better initialization point $\lambda^{[1]}$ for Algorithm 1.

At each iteration, the inner (minimization) problem in (11) is solved as a generic optimization program. As opposed to the fully coupled system (10), general purpose solvers solve the decoupled problem very rapidly. The inner problems are constructed using Yalmip [13] and solved by CPLEX 12.5.

Figure 2(a) depicts the typical dual convergence observed during the execution of Algorithm 1. The vertical dashed line indicates the iteration from which we activate the "$\alpha/k$" stepsize rule and start to compute the average sequence. Figure 2(b) shows primal convergence of the averaged $\bar{x}^{[k]}$. In our experiments the "heavy tail" behavior shown is dominated by the continuous part of the problem, i.e., by the convergence rate of $(\bar{S}^{[k]})_j^i \to (S_{\text{LP}}^\star)_j^i$. In contrast, the convergence of the integer part, i.e., $(\bar{w}^{[k]})^i \to (w_{\text{LP}}^\star)^i$ is much faster. In order to speed up the computations, we thus fix the integer parts of $(w^{[k_{\max}]})^i$, and in the rectification part of the Algorithm, where we recompute a solution over $I_2$ using the remaining budget, we also recompute the continuous part of the problem $(S_j^i)$ for all the subsystems.

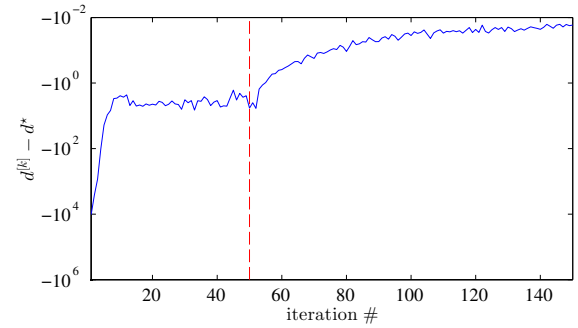
| N | M | Proposed Method Gap (%)† | Proposed Method Time (s)† | CPLEX Time (s)† |
|---|---|---|---|---|
| 100 | 25 | .11–.38–.60 | 16.7–18.5–21.5 | 10.3–63.5–202.5 |
| 300 | 75 | .01–.03–.07 | 61.0–62.5–63.4 | – |
| 500 | 50 | .04–.06–.11 | 95.0–101.0–110.6 | – |
| 600 | 50 | .01–.06–.10 | 113.7–116.6–119.5 | – |
| 1000 | 100 | .00–.03–.05 | 225.3–245.2–288.8 | – |

†indicated are min., average and max. computation times.
(–) runs out of memory before solving.

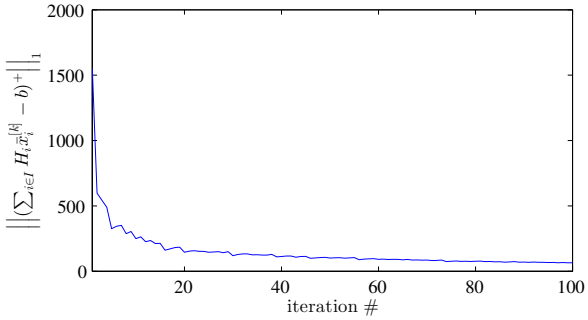
TABLE II

PERFORMANCE OF THE PROPOSED METHOD COMPARED TO CPLEX 12.5.

The results are reported in Table II, where we compare the performance of our proposed method with CPLEX 12.5 when it is applied to the fully coupled system (10). CPLEX is generally unable to provide exact solutions to the problem before running out of memory. Our proposed method provides nearly optimal feasible solutions, the computation times are acceptably short ($\leq$ 5 min) and are affected by a low degree of variance. We tested the algorithm on a single processor, but since the bulk of the computational effort lies in the computation of solutions to the inner problem, solve times can be substantially improved by exploiting parallelism.




(a) Dual convergence.




(b) Primal convergence of the ergodic sequence.

Fig. 2. Primal and dual convergence.

## APPENDIX

### A. Proofs of Theorem 1

Our theorem relies on the following crucial result, known in the literature as the Shapley–Folkman–Starr Theorem.

*Theorem 4 (Shapley–Folkman–Starr):* Let $S_i \subseteq \mathbb{R}^{m+1}$, $i \in I$, be nonempty sets with $|I| > m+1$, and let $S = S_1 + \cdots + S_{|I|}$. Then every vector $s \in \text{conv}(S)$ can be represented as $s = s_1 + \cdots + s_{|I|}$, where $s_i \in \text{conv}(S_i)$ for all $i \in I$, and $s_i \notin S_i$ for at most $m+1$ indices $i$.

*Proof:* See [6, Prop. 5.7.1]. ∎

*Proof:* [Theorem 1] Let $F(\mathcal{P}_{\text{LP}})$ be the feasible set of the program $\mathcal{P}_{\text{LP}}$, i.e.,

$$F(\mathcal{P}_{\text{LP}}) \doteq \left\{ x = \begin{bmatrix} x_1 \\ \vdots \\ x_{|I|} \end{bmatrix} \;\middle|\; \sum_{i \in I} H_i x_i \leq b, \quad x_i \in \text{conv}(X_i) \right\}.$$

Suppose that $\tilde{x} \doteq [\tilde{x}_1^\top, \cdots, \tilde{x}_{|I|}^\top]^\top$ is a vertex of $F(\mathcal{P}_{\text{LP}})$. Then there exists a vector $\tilde{c} \doteq [\tilde{c}_1^\top, \cdots, \tilde{c}_{|I|}^\top]^\top$ such that $\tilde{x}$ is the unique optimizer of

$$\tilde{x} = \underset{x \in F(\mathcal{P}_{\text{LP}})}{\text{argmin}} \; \tilde{c}^\top x.$$

Consider now the set $S \doteq S_1 + \cdots + S_{|I|}$ where

$$S_i \doteq \left\{ \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \in \mathbb{R}^{m+1} \;\middle|\; s_1 = H_i x_i, \quad s_2 = \tilde{c}_i^\top x_i, \right. \\ \left. \text{for some} \quad x_i \in X_i \right\}.$$

Since $\text{conv}(H \cdot X) = H \cdot \text{conv}(X)$ (where $H$ is a matrix, $X$ is a set – possibly non-convex – and the product is intended to be the multiplication of each vector in $X$ by $H$), we have that

$$\text{conv}(S_i) = \left\{ \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \in \mathbb{R}^{m+1} \;\middle|\; s_1 = H_i x_i, \quad s_2 = \tilde{c}_i^\top x_i, \right. \\ \left. \text{for some} \quad x_i \in \text{conv}(X_i) \right\}.$$

Then, in view of Theorem 4, for every $x = [x_1^\top, \cdots, x_{|I|}^\top]^\top$, $x_i \in \text{conv}(X_i)$, there exist $I_1 \subset I$, $|I_1| \geq m+1$, and a representation $y = [y_1^\top, \cdots, y_{|I|}^\top]^\top$ such that $y_i \in \text{conv}(X_i)$ for all $i \in I$, $y_i \in X_i$ for all $i \in I_1$ and

$$\sum_{i \in I} \tilde{H}_i x_i = \sum_{i \in I} \tilde{H}_i y_i, \qquad \sum_{i \in I} \tilde{c}_i^\top x_i = \sum_{i \in I} \tilde{c}_i^\top y_i.$$

This implies that the representation $y$ of any $x \in F(\mathcal{P}_{\text{LP}})$ also belongs to $F(\mathcal{P}_{\text{LP}})$ and attains the same objective value. But since $\tilde{x}$ is the unique minimizer, it must coincide with its representation $\tilde{y}$, which concludes the proof. ∎

### B. Proofs of Theorem 2 and 3

For reasons of space, the other proofs have been moved to [19], which is available on–line.

## REFERENCES


[1] Kurt M. Anstreicher and Laurence A. Wolsey. Two "well-known" properties of subgradient optimization. *Math. Program.*, 120(1):213–220, April 2009.

[2] J. P. Aubin and I. Ekeland. Estimates of the duality gap in nonconvex optimization. *Mathematics of Operations Research*, 1(3):225–245, January 1976.

[3] Snehamay Banerjee, Avijit Banerjee, Jonathan Burton, and William Bistline. Controlled partial shipments in two-echelon supply chain networks: a simulation study. *International Journal of Production Economics*, 71(13):91–100, May 2001.

[4] Philipp Baumann and Norbert Trautmann. Portfolio-optimization models for small investors. *Mathematical Methods of Operations Research*, 77(3):345–356, June 2013.

[5] Martin Bergner, Alberto Caprara, Fabio Furini, Marco Luebbecke, Enrico Malaguti, and Emiliano Traversi. Partial convexification of general mips by dantzig-wolfe reformulation. In *Proceedings of the 15th international conference on Integer Programming and combinatorial optimization*, pages 39–51, 2011.

[6] Dimitri P. Bertsekas. *Convex Optimization Theory*. Athena Scientific, 1st edition, June 2009.

[7] Dimitri P. Bertsekas, G. Lauer, N. Sandell, and T. Posbergh. Optimal short-term scheduling of large-scale power systems. *IEEE Transactions on Automatic Control*, 28(1):1–11, January 1983.

[8] John R. Birge and M. a. H. Dempstert. Stochastic programming approaches to stochastic scheduling. *Journal of Global Optimization*, 9(3-4):417–451, December 1996.

[9] Claus C. Caroe and Ruediger Schultz. Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24(1):37–45, 1999.

[10] Milind Dawande, Srinagesh Gavirneni, and Sridhar Tayur. Effective heuristics for multiproduct partial shipment models. *Operations Research*, 54(2):337–352, January 2006.

[11] A. M. Geoffrion. *Lagrangean relaxation for integer programming*, volume 2 of *Mathematical Programming Studies*. Springer Berlin Heidelberg, 1974.

[12] T. Larsson and Z. Liu. A lagrangean relaxation scheme for structured linear programs with application to multicommodity network flows. *Optimization*, 40(3):247–284, 1997.

[13] J. Loefberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[14] O.L. Mangasarian. Uniqueness of solution in linear programming. *Linear Algebra and its Applications*, 25:151–162, June 1979.

[15] Angelia Nedic and Asuman Ozdaglar. Approximate primal solutions and rate analysis for dual subgradient methods. *SIAM J. on Optimization*, 19(4):1757–1780, February 2009.

[16] D.P. Palomar and Mung Chiang. A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communications*, 24(8):1439–1451, 2006.

[17] Hanif D. Sherali and Gyunghyun Choi. Recovery of primal solutions when using subgradient optimization methods to solve lagrangian duals of linear programs. *Operations Research Letters*, 19(3):105–113, September 1996.

[18] N.Z. Shor, K.C. Kiwiel, and A. Ruszczynski. *Minimization methods for non-differentiable functions*, volume 121. Springer Verlag, 1985.

[19] Robin Vujanic, Peyman Mohajerin Esfahani, Paul Goulart, and Manfred Morari. Proofs of the Theorems 1, 2 and 3. `http://control.ee.ethz.ch/index.cgi?page=publications&action=details&id=4717`, 2014.

[20] H.Y. Yamin. Review on methods of generation scheduling in electric power systems. *Electric Power Systems Research*, 69:227–248, May 2004.

[21] Wei Yu and R. Lui. Dual methods for nonconvex spectrum optimization of multicarrier systems. *IEEE Transactions on Communications*, 54(7):1310–1322, 2006.