Decomposition Methods for Global Solutions of Mixed-Integer Linear Programs

Kaizhao Sun · Mou Sun · Wotao Yin

February 23, 2021

Abstract This paper introduces two decomposition-based methods for two-block mixed-integer linear programs (MILPs), which break the original problem into a sequence of smaller MILP subproblems. The first method is based on the \( \ell_1 \)-augmented Lagrangian. The second method is based on the alternating direction method of multipliers. When the original problem has a block-angular structure, the subproblems of the first block have low dimensions and can be solved in parallel. We add reverse-norm cuts and augmented Lagrangian cuts to the subproblems of the second block. For both methods, we show asymptotic convergence to globally optimal solutions and present iteration upper bounds. Numerical comparisons with recent decomposition methods demonstrate the exactness and efficiency of our proposed methods.

Keywords Mixed-integer linear programming · Augmented Lagrangian · Alternating direction method of multipliers

1 Introduction

In this paper, we consider the generic two-block mixed-integer linear program (MILP)

\[
p^* := \min_{x,z} \quad c^\top x + g^\top z \\
\text{subject to} \quad Ax + Bz = 0 \\
x \in X, z \in Z,
\]

where we have real decision variables \( x \in \mathbb{R}^n \) and \( z \in \mathbb{R}^d \), rational parameters \( c \in \mathbb{Q}^n \), \( g \in \mathbb{Q}^d \), \( A \in \mathbb{Q}^{m \times n} \), and \( B \in \mathbb{Q}^{m \times d} \), and compact sets \( X \subset \mathbb{R}^n \) and \( Z \subset \mathbb{R}^d \). The sets require some coordinates to take only integer values. Without loss of generality\(^1\), we have used a zero vector in the right-hand side of (1b). Let us write \( x \) in the block-coordinate form \( x = [x_1^\top, \ldots, x_p^\top]^\top \) for \( p \geq 1 \). Assume that \( A \) is block diagonal and \( X = X_1 \times \cdots \times X_p \).

---

1. Any \( Ax + Bz = b \) can be equivalently expressed as \( Ax + B\tilde{z} = 0 \) where \( B = [B, b] \) and \( \tilde{z} = [z^\top, -1]^\top \).

K. Sun
School of Industrial and Systems Engineering, Georgia Institute of Technology
E-mail: ksun46@gatech.edu

M. Sun
Decision Intelligence Lab, DAMO Academy, Alibaba Group
E-mail: mou.sunm@alibaba-inc.com

W. Yin
Decision Intelligence Lab, DAMO Academy, Alibaba Group (US)
E-mail: wotao.yin@alibaba-inc.com
This paper introduces methods to take advantages of the lower dimensions of the separable blocks \( x_1, \ldots, x_p \). The methods decouple them into a set of mixed-integer subproblems, each involving one \( x_i \in X_i \), so that \( p \) of them can be solved in parallel on multiple computing cores. Of course, when \( p = 1 \), our methods still offer a decomposition between \( x \) and \( z \). They coordinate the subproblem solutions and converge to a solution of the global problem.

1.1 Background

A mixed-integer program (MIP) is an optimization problem involving integer variables. A MILP is a MIP with affine objectives and constraints. The MIP is a common formulation in decision sciences and data sciences. As data have become much larger, decision problems with higher dimensions have arisen. For example, in online marketing, decisions are made on billions of consumers and millions of SKUs, which are significantly more than those in classic marketing. As another example, dispatching resources in cloud computing has a typical scale that is order-of-magnitude larger than dispatching a fleet of vehicles. The scale increase of these MIPs has surpassed the evolution of MIP method. Therefore, it is of central importance to develop algorithms that can scale well enough to solve huge MIPs.

Huge MIPs typically have decomposable structures. The coefficient matrix of (1) may take the classic block-angular structure:

\[
\begin{bmatrix}
A & B \\
A_2 & B_2 \\
\vdots & \vdots \\
A_p & B_p
\end{bmatrix}
\]

where \( A \) has a block-diagonal structure and the rows of \( B \) are divided into groups accordingly. The matrix in (2) naturally arises in two-stage optimization problems: in \( Ax + Bz = 0 \), the \( z \) variable represents the first-stage decision variable and couples with the second-stage variables \( x_i \) via \( A_ix_i + B_iz = 0 \) in each scenario \( i \in [p] \). More often than not, each second-stage variable \( x_i \) is independently constrained by \( x_i \in X_i \).

We can also transform the following multi-block problem into problem (1):

\[
\min_{x_1 \in X_1, \ldots, x_p \in X_p} \sum_{i=1}^{p} c_i^T x_i, \quad \text{subject to } \sum_{i=1}^{p} G_i x_i = h.
\]

There possibly exist dense rows of the constraints that couple all the \( p \) blocks of variables. To obtain (1), introduce auxiliary variable \( z_i \) for each block variable \( x_i \): let \( X := \prod_{i=1}^{p} X_i \), \( Z := \{ z = [z_1^T, \ldots, z_p^T]^T \mid \sum_{i=1}^{p} G_i z_i = h \} \), and impose consensus constraints \( x_i = z_i \) for all \( i \in [p] \). Formulation (3) has many important applications including, but not limited to, the consensus and sharing problems in utility maximization.

1.2 Augmented Lagrangian Decomposition and Challenges

Decomposing a large global problem into smaller local subproblems is not new. The most common methods are based on the augmented Lagrangian (AL) and the alternating direction method of multipliers (ADMM). However, theoretical and practical issues arise when it comes to decomposing MILPs. Consider the following \( \ell_1 \)-penalty AL for problem (1):

\[
L(x, z, \lambda, \rho) := c^T x + g^T z + \langle \lambda, Ax + Bz \rangle + \rho \| Ax + Bz \|_1, \quad \lambda \in \mathbb{R}^m, \quad \rho > 0,
\]
and the corresponding dual function

\[ d(\lambda, \rho) := \min_{x \in X, z \in Z} L(x, z, \lambda, \rho). \]  

We call the minimization problem in (5) the \textit{primal subproblem} and call the following problem:

\[ \sup_{\lambda \in \mathbb{R}^m, \rho > 0} d(\lambda, \rho) \]  

the \textit{dual problem}. Note that the penalty parameter \( \rho \) is a variable. Feizollahi, Ahmed, and Sun [13] established that one can solve (1) by solving (6).

The classic AL uses the squared penalty \( \frac{1}{2} \| Ax + Bz \|_2^2 \), which couples the blocks of \( x \) and \( z \) in a nonlinear way. The subproblems of the classic AL are, therefore, more difficult to solve than those given by the \( \ell_1 \)-penalty AL.

The augmented Lagrangian method (ALM) was proposed in the late 1960s by Hestenes [15] and Powell [22], and is still a popular method for constrained programs. Convergence of ALM has been established for convex programs [23, 24] and smooth nonlinear programs [3].

For MILP (1) and other nonsmooth nonconvex problems, ALM-based algorithms are developed based on the theory of nonconvex AL duality [14, 16, 25] and aim to solve the max-min problem (5)–(6). Since \( d(\lambda, \rho) \) is concave and upper-semicontinuous in \((\lambda, \rho)\) [25, Exercise 11.56], one can maximize it with existing methods for nonsmooth convex optimization. In the context of a more general equality-constrained nonlinear program, works [6–8] introduced various modified subgradient methods, and [10] proposed an inexact bundle method for solving the dual problem (6); convergence of the dual variable is established as long as the dual optimal set is nonempty, while convergence of the primal variable can be ensured even when the dual optimal set is empty.

When we consider MILP (1), the subproblem (5) has nonconvex mixed-integer constraint sets \( X \) or \( Z \). To our knowledge, there is no efficient decomposition method for (5) that can guarantee convergence to a globally optimal solution. Methods [6–8, 10] all need an exact or inexact oracle of \( d(\lambda, \rho) \), thus requiring a (nearly) global minimize of subproblem (5). The method in [10] alternatively updates two blocks of variables in (5), yet block coordinate descent (BCD) methods, in general, may get stuck at a non-stationary point when a nonsmooth coupling function, such as \( \| Ax + Bz \|_1 \) in (4), is present [34]. For the special case of (1) with \( B = I \), \( g = 0 \), and \( Z \) being a linear subspace, recent work [5] applied proximal ALM to a convexified formulation, which may not return a feasible solution \( x \in X \).

1.3 ADMM and Challenges

The ADMM successively updates each of the primal variable blocks, rather than update them together, so it has simpler subproblems than ALM. Proving ADMM converges is more challenging.

Although the majority of the ADMM literature assumes convexity, convergence results of ADMM for nonconvex multi-block problems have appeared [32, 17, 21, 27]. So have the ADMM-type methods involving discrete variables. Authors of [36] reformulated a vehicle routing problem as a multi-block MILP and applied ADMM where blocks are updated in the Gauss-Seidel style. They discovered that each mixed-integer quadratic program (MIQP) subproblem can be reduced to MILP since the coupling terms in the objective are binary. Problems studied in \([2, 18, 19, 28, 29, 35]\) can be abstracted into the form \( \min_{x \in X \cap Z} \{ f(x) \} \), where \( f \) and \( X \) are continuous and easy to optimize with but \( Z \) is discrete and poses challenges. Introduce variable \( z \) to obtain

\[ \min_{x, z} \{ f(x) \} \quad \text{subject to} \quad x = z, x \in X, z \in Z, \]  

on which ADMM can be applied. The \( x \)-subproblem becomes a convex program (except in [19], where it is a mixed-integer nonlinear program); the \( z \)-subproblem is a projection onto a discrete set, which may admit a closed-form solution and may be also parallelized.
The aforementioned works of ADMM cannot ensure convergence to the global solution. The conditions in [32] for ADMM to converge to a stationary point of a nonconvex problem also fail to hold for (7). Indeed, one common set of structural conditions in [32] are 1) $\text{Im}(A) \subseteq \text{Im}(B)$, and 2) $Z = \mathbb{R}^m$. They fail to hold for (7) when $Z$ is not $\mathbb{R}^m$.

The authors of [33] proposed an ADMM-based method to solve problems involving binary variables; the binary constraint $x \in \{0, 1\}^n$ is formulated as the intersection of $[0, 1]^n$ and a shifted $\ell_p$-sphere, and the resulting ADMM is shown to converge to a stationary point of a perturbed problem. It is, however, unclear whether such a stationary point is always a (nearly) optimal binary solution of the original problem.

Encouraging numerical results of ADMM on discrete optimization are reported in [2,18,19,28,29,35,36], where valuable experimental observations on adaptive tuning of the penalty parameter and restart heuristics are also discussed. None of these works offers convergence guarantees to global solutions. Indeed, there are examples on which ADMM either diverges or converges to local or infeasible solutions.

1.4 Our Approach

We aim to overcome the challenges outlined in Sections 1.2 and 1.3.

Our first method is based on ALM with its subproblem (5) solved by a new method we call Alternating Update Scheme for the Sharp AL function or AUSAL. Here, “sharp” refers to the exact penalty property in (4). AUSAL has dual-stage loops. The inner stage minimizes over $x \in X$ with $z$ fixed. The outer-stage objective is a Lipschitz function in $z$, which we can evaluate but do not have an explicit form. Borrowing a classic idea in global Lipschitz minimization from [20], we approximate this objective using the point-wise maximum of a set of nonconvex minorants. AUSAL generates nonconvex minorants using reverse norm cuts, which are obtained from the inner-stage solutions. After many cuts generated, AUSAL returns a nearly optimal $z$ in the outer stage and a nearly optimal $x$ in the inner stage for (5). We can bound the error by the number of cuts and other constants. The $x$ and $z$ are used to update the Lagrange multipliers and penalty parameter.

Our second method is based on the ADMM, where we update the Lagrange multipliers and penalty parameter more frequently. Since a historical reverse norm cut cannot be effectively reused in a new AUSAL subroutine, we instead use a class of nonconvex minorants known as AL cuts (though we use them in ADMM, not AL), which were proposed recently by Ahmed et al. [1] and Zhang and Sun [37] in nonconvex stochastic optimization. Each such cut can be obtained by minimizing over $x$ for a fixed Lagrange multipliers, a penalty parameter, and $z$.

We show that the two methods are guaranteed to converge to a global solution of the original MIP (1) under proper conditions. We summarize our contributions in the next subsection.

1.5 Contributions

We propose AUSAL to solve the primal subproblem (5), and we bound its iteration complexities by $O(p^d/d^\epsilon)$ to find an $\epsilon$-optimal solution, where $d$ is the dimension of $z$. In order to obtain an $\epsilon$-solution of MILP (1) (see Definition 1), we can either apply AUSAL to the penalty formulation, i.e., $\lambda = 0$ in (5), or apply AUSAL inside an ALM. The former requires properly increasing the penalty parameter to reach exact penalty. The latter updates both Lagrange multipliers and the penalty parameter. We provide two variants of the latter based on two subgradient methods.

We also propose an ADMM-based method that uses AL cuts. Our approach differs from those in [1,37]. In particular, we obtain an AL cut by solving a single MIP subproblem in $x$ with $z$ fixed rather than getting an AL cut after solving a series of subproblems as in [1,37]. Under conditions regarding the sequence of Lagrange multipliers and penalty parameters, the method finds an $\epsilon$-solution
in $\mathcal{O}(1/\epsilon^d)$ iterations. The conditions allow flexible ways to update the Lagrange multipliers and penalty parameter.

The proposed ALM-based and ADMM-based methods maintain the separable structure of the original problem, where the update of the $x$ variable can be decomposed into lower-dimensional MILPs and solved simultaneously. They are applicable to the motivating examples discussed in Section 1.1.

We conduct numerical experiments and compare our methods with existing decomposition algorithms on a class of randomly generated MILP instances. The proposed methods demonstrate advantages in either solution time or quality, and often both.

A disadvantage of our methods is that the $z$-subproblem grows in size as more cuts are added, which makes the $z$-subproblem increasingly difficult to solve. This is a common issue associated with cut-based MILP methods. We believe, however, we can alleviate this issue by generating more efficient cuts, applying row generation, or using other means to solve the $z$-subproblem efficiently. We leave them to future work.

The proposed methods are not replacement of MILP solvers but means to scale the existing solvers to larger MILP instances, especially for those with block separable structures.

1.6 Other Related Works

References [9, 31] study the multi-block MILP problem, where the number of blocks is far greater than the dimension of coupling constraints. Paper [9] applies **primal decomposition** to the multi-block problem (3), giving rise to a bi-level optimization framework, where each lower-level subproblem involves only one pair of cost vector $c_i$ and constraint set $\{ x \in X_i : G_i x_i \leq y_i \}$ and the high-level problem coordinates their local resource variables $y_1, \ldots, y_p$. Paper [31] and its follow-up works [11, 12] are based on **dual decomposition**, where a restricted Lagrangian dual problem reduces to subproblems that each involves only one pair of $(c_i, G_i)$ and constraint set $X_i$. In addition to pioneering in distributed computation of multi-block MILPs, both primal and dual decompositions are also theoretically supported: under mild assumptions, a feasible primal solution can be recovered with a suboptimality bound by invoking the Shapley-Folkman Theorem [3, Section 5.6]. Paper [30] considers a generic MILP in a fully decentralized setting, where constraints are distributed over a time-varying network. During each communication round, nodal agents solve localized linear program (LP) subproblems, generate Gomory Mixed-Integer (GMI) cuts, and exchange their cuts and local bases with neighbors; finite time convergence to an optimal solution is established with the lexicographical optimality of the solution to each LP relaxation.

1.7 Notation and Organization

We let $\mathbb{R}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{N}$ denote the sets of real, integer, rational, and natural numbers. Write $[k] = \{1, \cdots, k\}$. For a vector $x \in \mathbb{R}^n$, use $\|x\|_p$ as the $\ell_p$-norm of $x$ for $1 \leq p \leq \infty$. The inner product of $x, y \in \mathbb{R}^n$ is denoted by $\langle x, y \rangle$ or $x^\top y$. For a matrix $A \in \mathbb{R}^{m \times n}$, $\|A\|_p$ denotes its induced (operator) norm, and $\text{Im}(A)$ denotes its column space. We introduce $\overline{B}_p(x; R) = \{ y \in \mathbb{R}^n \mid \|x - y\|_p \leq R \}$ and $D_p(X) = \sup \{ \|x - y\|_p \mid x, y \in X \}$. We let $\mathbb{I}_X(x)$ be the $0/\infty$-indicator function of $X$, which equals 0 if $x \in X$ and $+\infty$ otherwise.

The rest of the paper is organized as follows. We state our assumption on problem (1) and provide a more detailed review of background materials in Section 2. We present the proposed ALM framework in Section 3 and ADMM variant in Section 4, together with their convergence results. In Section 5, we discuss implementation issues and present numerical experiments. Finally, we give some concluding remarks in Section 6.
2 Preliminary

2.1 Assumption and Approximate Solution

Throughout this paper, we make the following assumption on MILP (1).

**Assumption 1** Problem (1) is feasible, and constraint sets $X$ and $Z$ in (8) are compact and mixed-integer representable, i.e.,
\[
X = \{ x \in \mathbb{R}^{n_1}_+ \times Z^{n_2}_+ \mid Ex = f \}, \quad Z = \{ z \in \mathbb{R}^{d_1}_+ \times Z^{d_2}_+ \mid Gx = h \},
\]
for some rational matrices (vectors) $E$ and $G$ ($f$ and $h$), where $n_1 + n_2 = n$, $d_1 + d_2 = d$.

We measure the accuracy of an approximate solution as follows.

**Definition 1** Let $\epsilon > 0$. We say $(x^*, z^*)$ is an $\epsilon$-solution of the MIP (1) if $x^* \in X$, $z^* \in Z$, and
\[
c^\top x^* + g^\top z^* \leq p^* + \epsilon, \quad \| Ax^* + Bz^* \|_1 \leq \epsilon.
\]
(9)

Note that infeasibility is measured in the $\ell_1$-norm due to our analysis. Since $(x^*, z^*)$ may be infeasible, it is possible that $c^\top x^* + g^\top z^* < p^*$.

2.2 Exact Penalization

Since the AL dual problem in (6) has weak duality $\sup_{\lambda \in \mathbb{R}^m, \rho \geq 0} d(\lambda, \rho) \leq p^*$, two important follow-up questions are: 1) how to obtain strong duality, i.e., “=” holds; 2) how to obtain an optimal primal solution by solving the dual problem? Paper [13] provides positive answers to both questions for MILP.

**Definition 2** (Exact Penalization [25, Definition 11.60]) A dual variable $\lambda \in \mathbb{R}^m$ is said to support exact penalization if, for all sufficiently large $\rho > 0$, it holds that
\[
\operatorname{Argmin}_{x \in X, z \in Z} \{ c^\top x + g^\top z \mid Ax + Bz = 0 \} = \operatorname{Argmin}_{x \in X, z \in Z} L(x, z, \lambda, \rho).
\]
We say a pair $(\lambda, \rho)$ supports exact penalization if the above equation holds. We simply say $\rho$ supports exact penalization when $(0, \rho)$ does so.

**Theorem 1** (Exact Penalization for MILP [13, Proposition 1 and Theorem 5]) Suppose Assumption 1 holds. Strong duality holds for (6), i.e., $\sup_{\lambda \in \mathbb{R}^m, \rho \geq 0} d(\lambda, \rho) = p^*$. For any $\lambda \in \mathbb{R}^m$, there exists a $\rho > 0$ such that every $(\lambda, \rho)$ with $\rho \in [\rho, +\infty)$ supports exact penalization.

The exact penalization result proved in [13] applies to a broader class of penalty functions, including all norms in the finite Euclidean space. This paper focuses on the $\ell_1$-norm for component-wise separability. The augmented Lagrangian given by $\frac{1}{2}\| \cdot \|_2^2$ does not support exact penalization; in general, no finite penalty $\rho$ can eliminate the duality gap [13, Proposition 7]. A complete characterization of exact penalization is given in the following theorem.

**Theorem 2** (Criterion for Exact Penalization [25, Theorem 11.61]) Suppose Assumption 1 holds. The following statements are equivalent:

1. The pair $(\lambda, \rho)$ supports exact penalization.
2. The pair $(\lambda, \rho)$ solves the dual problem (6).
3. There is $r > 0$ such that $p(u) \geq p(0) + \langle \lambda, u \rangle - \rho \| u \|_1$, $\forall u \in \overline{B}_1(0; r)$, for
\[
p(u) := \min_{x, z} \{ c^\top x + g^\top z \mid Ax + Bz + u = 0, x \in X, z \in Z \}.
\]
(10)
3 An ALM Method empowered by AUSAL

In this section, we introduce an ALM framework for MILP (1). In Section 3.1, we present the AUSAL algorithm for subproblem (5) with convergence guarantees. In order to find an ϵ-solution of MILP (1), AUSAL is further applied to the penalty formulation in Section 3.2 or embedded in the ALM in Section 3.3. We present two variants of ALM based on different subgradient updates.

3.1 AUSAL

In this section, we propose AUSAL to solve the primal subproblem in (5):

\[
d(\lambda, \rho) = \min_{x \in \mathcal{X}, z \in \mathcal{Z}} c^T x + g^T z + \langle \lambda, Ax + Bz \rangle + \rho \| Ax + Bz \|_1,
\]

where \( \lambda \in \mathbb{R}^m \) and \( \rho \) are considered as constants. We decompose the minimization into two stages: the inner stage minimizes over \( x \in \mathcal{X} \) with \( z \) fixed and the outer stage minimizes over \( Z \), respectively,

\[
R(z) := \min_{x \in \mathcal{X}} \langle c + A^T \lambda, x \rangle + \rho \| Ax + Bz \|_1,
\]

\[
d(\lambda, \rho) = \min_{z \in \mathcal{Z}} \langle g + B^T \lambda, z \rangle + R(z).
\]

Note that \( R(z) \) is well defined over \( z \in \mathbb{R}^d \) since \( X \) is compact. The function \( R(z) \) captures the dependency of the \( x \)'s objective value on variable \( z \), and is closely related to the concept of value function or cost-to-go function in the context of sequential decision making problems.

Since the minimizer \( x \) does not have a closed form, \( R(z) \) lacks an explicit formula. We can only compute it for each \( z \). One approach is to solve (13) with a blackbox optimization method, which we do not study here.

We present an alternative approach that replaces \( R(z) \) by an explicit function that is its lower approximation and reduce (13) to a MILP. To this end, we need the following properties of \( R(z) \):

**Lemma 1** Suppose Assumption 1 holds, and \( X \) is compact for any right-hand side vector \( f \) (possibly empty). Then \( R(z) \) is piecewise-linear and \( K_\rho \)-Lipschitz continuous with respect to the \( \ell_1 \)-norm over \( \mathbb{R}^d \), where \( K_\rho := \rho \| B \|_1 \).

**Proof** We firstly show \( R(z) \) is piecewise linear by considering the standard-form MILP problem:

\[
v(b) := \min_z \left\{ c^T x \middle| Ax := \begin{bmatrix} A_1 x \\ A_2 x \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \right\}.
\]

Let \( v(b) \) equal \( +\infty \) if it is infeasible and \( -\infty \) if it is unbounded. Define \( D := \{ b \mid v(b) < +\infty \} \). It is shown in [4] that if the MILP is described by rational data and \( v(b) > -\infty \) for all \( b \in D \), then \( v(b) \) is a piecewise linear function over \( D \). Now fix \( b_2 \in \text{Im}(A_2) \) and define the partial value function with respect to \( b_1 \) by \( v_1(b_1) = v(b_1, b_2^\top) \). Then \( v_1(b_1) \) is also piecewise linear over its domain \( D_1 := \{ b_1 \mid [b_1^\top, b_2^\top]^\top \in D \} \). Notice that the premise of [4] is satisfied for problem (12) by assumption, and we can equivalently write \( R(z) = \min_{x \in \mathcal{X} \cup \{ z \}} \langle c + A^T \lambda, x \rangle + \rho \| Ax + Bu \|_1 \mid u = z \rangle \), where \( z \) plays the role of \( b_1 \). Since the rest of the problem defining \( R(z) \) can be cast as a standard MILP problem, we conclude that \( R(z) \) is a piecewise linear function in \( \mathbb{R}^d \).

To prove \( R(z) \) is Lipschitz, take any \( \bar{z}, \tilde{z} \in \mathbb{R}^d \), and let \( \bar{x} \) and \( \tilde{x} \) be the optimal solution to (12) with \( z = \bar{z} \) and \( z = \tilde{z} \), respectively. It holds that

\[
R(\bar{z}) - R(\tilde{z}) \\
\leq \langle c + A^T \lambda, \bar{x} \rangle + \rho \| A\bar{x} + B\bar{z} \|_1 - \langle c + A^T \lambda, \tilde{x} \rangle - \rho \| A\tilde{x} + B\tilde{z} \|_1 \\
\leq \rho \| B \|_1 \| \bar{z} - \tilde{z} \|_1,
\]
where the first inequality is due to the optimality of \( \bar{x} \), and the second inequality is by the triangle inequality. Similarly we have \( R(\bar{z}) - R(z) \leq \rho \| B \|_1 \| \bar{z} - z \|_1 \), which concludes the proof with the claimed modulus \( K_\rho \).

Since \( R(z) \) is \( K_\rho \)-Lipschitz, it has the lower approximation:

\[
r(z; \bar{z}) := R(\bar{z}) - K_\rho \| z - \bar{z} \|_1,
\]

where, for any given \( \bar{z} \in Z \), we can compute \( R(\bar{z}) \). Let us call \( -\| z - \bar{z} \|_1 \) “reverse norm”; hence, we add \( z \) changes, we keep track of the best (i.e., lowest) upper bound. On the other side, we can reduce the gap between the best upper bound and the lower bound.

\[ R(\bar{z}; \bar{z}) := \sup \{ r(z; \bar{z}) \mid z \in \bar{Z} \}, \]

which becomes a tighter lower approximation of \( R(z) \) as \( \bar{Z} \) gets larger. Eventually, if \( \bar{Z} = Z \), we have \( R(z; \bar{Z}) = R(z) \) over all \( z \in Z \).

Function \( R(z; \bar{Z}) \) is piece-wise linear and \( K_\rho \)-Lipschitz under the \( \ell_1 \) norm. So, we solve

\[
\min_{z \in \bar{Z}} \langle g + B^T \lambda, z \rangle + R(z; \bar{Z})
\]

either with a method for piecewise linear objectives or with a MILP solver (after expanding \( R(z; \bar{Z}) \) into a set of linear constraints).

Suppose (16) returns a solution \( z' \). Then, \( \langle g + B^T \lambda, z' \rangle + R(z') \) is an upper bound of (13). When \( \bar{Z} \) changes, we keep track of the best (i.e., lowest) upper bound. On the other side, \( \langle g + B^T \lambda, z' \rangle + R(z'; \bar{Z}) \) is a lower bound of (13). Adding \( z' \) to \( \bar{Z} \) can reduce the gap between the best upper bound and the lower bound. Indeed, even if \( z' \) is a minimizer of (13), the upper bound is tight but the lower bound may not be tight since \( R(z'; \bar{Z}) < R(z') \) is possible. If \( z' \) is not a minimizer of (13), the lower bound must be non-tight. Adding \( z' \) to \( \bar{Z} \) leads to \( R(z'; \bar{Z}) = R(z') \) and tightens the lower bound. This adding step requires computing \( R(z') \) according to (12).

The approach described in the last paragraph is known to the community of global Lipschitz minimization [20] and is recently used in stochastic optimization [1]. Applying this idea to the AL framework has advantages that are not noted before: the explicit Lipschitz constant \( K_\rho = \rho \| B \|_1 \) of \( R(z) \) and that problem (12) is always feasible without assuming the complete continuous recourse (CCR) condition [38].

We present the proposed AUSAL method in Algorithm 1, which returns an \( \epsilon \)-optimal to the primal subproblem (5).

**Algorithm 1 :** AUSAL for subproblem (11)

1: Input \( \lambda \in \mathbb{R}^m; \rho > 0, \epsilon \geq 0; \)
2: Initialize \( z^0 \in \bar{Z}, R(z^0), Z_0 \leftarrow \{ z^0 \}, \text{UB} \leftarrow +\infty; \)
3: for \( k = 1, 2, \ldots \) do
4: \quad compute \( z^k \in \text{Argmin}_{z \in \bar{Z}} \langle g + B^T \lambda, z \rangle + R(z; Z_{k-1}); \)
5: \quad UB \leftarrow \min(UB, \langle g + B^T \lambda, z^k \rangle + R(z^k));
6: \quad if UB \geq \epsilon then return \( (x^k, z^k) \);
7: \quad if UB \geq \epsilon then return \( (x^k, z^k) \);
8: \quad Z_k \leftarrow Z_{k-1} \cup \{ z_k \};
9: end if
10: end for
3.1.1 Convergence of AUSAL

We present the asymptotic convergence and iteration complexity of AUSAL.

**Lemma 2** Let \( z^k \) be an iterate in AUSAL (Algorithm 1). Then \( R(z) - R(z; Z_k) \leq \epsilon \) for all \( z \in Z \) such that \( \|z - z^k\|_1 \leq \epsilon/(2K_\rho) \).

**Proof** Notice for all \( z \in Z \) and \( \|z - z^k\|_1 \leq \epsilon/(2K_\rho) \), we have

\[
R(z) - R(z; Z_k) \leq R(z) - R(z^k) + K_\rho \|z - z^k\|_1 \\
\leq R(z^k) + K_\rho \|z - z^k\| - R(z^k) + K_\rho \|z - z^k\|_1 \\
= 2K_\rho \|z - z^k\|_1 \leq \epsilon,
\]

where the first inequality is due to \( R(z; Z_k) \geq R(z^k) - K_\rho \|z - z^k\|_1 \), and the second inequality is due to \( R \) being \( K_\rho \)-Lipschitz. \( \square \)

In words, Lemma 2 means that, as \( Z_k \) includes \( z_k \), the function \( R(z; Z_k) \) (and all future \( R(z; Z_k+1), \ldots \)) well approximates \( R \) near \( z_k \). Consider two consecutive iterates \( z^k \) and \( z^{k+1} \). If \( \|z^{k+1} - z^k\| \leq \epsilon/(2K_\rho) \), then the stopping criteria in line 7 of AUSAL is satisfied at iteration \( k + 1 \); otherwise, \( z^{k+1} \) must be bounded away from \( z^k \) by some positive distance.

**Theorem 3** Suppose Assumption 1 holds.

1. (Convergence) Let \( \{ (x^k, z^k) \}_{k \in \mathbb{N}} \) be the sequence generated by AUSAL, and define \( t^k = R(z^k; Z_{k-1}) \). The sequence \( \{ (g + B^T \lambda, z^k) + t^k \}_{k \in \mathbb{N}} \) converges to \( d(\lambda, \rho) \) monotonically from below. Any limit point \( (x^*, z^*) \) of \( \{ (x^k, z^k) \}_{k \in \mathbb{N}} \) is a global solution to the primal subproblem (5).

2. (Complexity) Let \( \epsilon > 0 \), and suppose \( Z \subseteq \overline{B}_1(\bar{z}; R) \) for some \( \bar{z} \in \mathbb{R}^d \) and radius \( R > 0 \). Then AUSAL terminates in no more than

\[
\left( 1 + \frac{4\rho\|B\|\|R\|}{\epsilon} \right)^d = O\left( \frac{p^d}{\epsilon^d} \right)
\]

iterations.

**Proof**

1. Notice \( R(z; Z_{k-1}) \leq R(z; Z_k) \) for all \( k \in \mathbb{N} \) and \( z \in Z \), so the sequence \( \{ (g + B^T \lambda, z^k) + t^k \}_{k \in \mathbb{N}} \) is nondecreasing and bounded from above by \( d(\lambda, \rho) \) and thus converges to some \( d \leq d(\lambda, \rho) \). Let \( (x^*, z^*) \) be a limit point and \( \{ (x^k, z^k) \}_{k \in \mathbb{N}} \) be the subsequence convergent to \( (x^*, z^*) \). Then we have

\[
\langle g + B^T \lambda, z^{k_j} \rangle + t^{k_j} \geq \langle g + B^T \lambda, z^{k_j} \rangle + R(z^{k_j}) - R(z^{k_{j-1}}) \|z^{k_j} - z^{k_{j-1}}\|_1.
\]

Since \( \|z^{k_j} - z^{k_{j-1}}\|_1 \to 0 \), letting \( j \to \infty \) on (18) gives \( d \geq \langle g + B^T \lambda, z^* \rangle + R(z^*) \geq d(\lambda, \rho) \), where the first inequality follows from the continuity of \( R \) and the second inequality is due to \( z^* \in Z \). As a result, we have \( d = d(\lambda, \rho) = \langle g + B^T \lambda, z^* \rangle + R(z^*) \). Finally, taking the limit on both sides of \( R(z^{k_j}) = \langle c + A^T \lambda, x^{k_j} \rangle + \rho \|Ax^{k_j} + Bz^{k_j}\|_1 \) over \( j \) implies that \( (x^*, z^*) \) is optimal to (5).

2. Since \( \lim_{j \to \infty} z^{k_j} = z^* \), we have \( \|z^{k_j+1} - z^{k_j}\|_1 \leq \epsilon/(2K_\rho) \) for all large enough \( j \in \mathbb{N} \). Notice \( k_{j+1} \geq k_j + 1 \); by Lemma 2, it follows \( R(z^{k_{j+1}}) - t^{k_{j+1}} \leq R(z^{k_{j+1}}) - R(z^{k_j}; Z_{k_j}) \leq \epsilon \). Now let \( K \) be the first index such that \( R(z^K) - t^K \leq \epsilon \), which is sufficient to ensure the termination of AUSAL (line 7). For all \( 0 \leq i < j \leq K - 1 \), we claim that \( \|z^i - z^j\|_1 > \epsilon/(2K_\rho) \); suppose not, then Lemma 2 suggests that \( R(z^i) - t^j \leq R(z^j) - R(z^j; Z_i) \leq \epsilon \), contradicting the choice of \( K \). Let \( r = \epsilon/(4K_\rho) \). Since \( z^i \in Z \) for all \( 0 \leq i \leq K - 1 \) and \( Z \subseteq \overline{B}_1(\bar{z}; R) \),

\[
\bigcup_{i=0}^{K-1} \overline{B}_1(z^i; r) \subseteq Z + \overline{B}_1(0; r) \subseteq \overline{B}_1(z; R + r);
\]
Theorem 5 Let \( \lambda, \rho \in \mathbb{R} \) satisfy (1), \( \lambda \geq \| \lambda \|_\infty \) and (2) \( \lambda, \rho - 1 \) supports exact penalization. Then AUSAL applied to the primal subproblem (5) returns an \( \epsilon \)-solution of MILP (1).

Proof Let \((x^*, z^*)\) denote the solution returned by Algorithm 1. Clearly, \( x^* \in X \) and \( z^* \in Z \). We first show \( c^\top x^* + g^\top z^* \leq p + \epsilon \):

\[
    c^\top x^* + g^\top z^* \leq c^\top x^* + g^\top z^* + (\rho - \| \lambda \|_\infty) \| Ax^* + Bz^* \|_1 \\
    \leq c^\top x^* + g^\top z^* + \langle \lambda, Ax^* + Bz^* \rangle + \rho \| Ax^* + Bz^* \|_1 \\
    \leq d(\lambda, \rho) + \epsilon \leq p^* + \epsilon. \tag{19}
\]

Furthermore, (19) also implies

\[
    \rho \| Ax^* + Bz^* \|_1 \leq p^* + \epsilon - c^\top x^* - g^\top z^* - \langle \lambda, Ax^* + Bz^* \rangle;
\]

since \( \lambda, \rho - 1 \) also supports exact penalization,

\[
    p^* \leq c^\top x^* + g^\top z^* + \langle \lambda, Ax^* + Bz^* \rangle + (\rho - 1) \| Ax^* + Bz^* \|_1.
\]

The above two inequalities together imply \( \| Ax^* + Bz^* \| \leq \epsilon \).

Since the current \((\lambda, \rho)\) may not support exact penalization, we must address the question: given any \( \lambda \in \mathbb{R}^m \), what is the sufficiently large \( \rho \) for AUSAL to deliver an \( \epsilon \)-solution of MILP (1). Note that \( \lambda = 0 \) is possible, and if this is the case, the primal subproblem (5) becomes the penalty formulation of (1).

Theorem 5 Let \( \lambda \in \mathbb{R}^m \) and \( \epsilon > 0 \). Then AUSAL applied to the primal subproblem (5) with

\[
    \rho = \frac{\| c \|_{\infty} D_1(X) + \| g \|_{\infty} D_1(Z)}{\epsilon} + \| \lambda \|_{\infty} + 1
\]

returns an \( \epsilon \)-solution \((x^*, z^*)\) of the MILP (1) in at most

\[
    \left[ 1 + \frac{4 \| B \|_1 R}{\epsilon} \left( \frac{\| c \|_{\infty} D_1(X) + \| g \|_{\infty} D_1(Z)}{\epsilon} + \| \lambda \|_{\infty} + 1 \right) \right]^d. \tag{20}
\]

iterations, where \( D_1(\cdot) \) returns the diameter of the argument, and \( R \) is the radius of \( Z \).
Proof We have $c^T x^* + g^T z^* \leq p^* + \epsilon$ from (19). It remains to show $\|Ax^* + Bz^*\|_1 \leq \epsilon$. Let $(\bar{x}, \bar{z})$ be an optimal solution of MILP (1). Again from (19), we have

$$\|Ax^* + Bz^*\|_1 \leq \frac{c^T (\bar{x} - x^*) + g^T (\bar{z} - z^*) + \epsilon}{\rho - \|\lambda\|_\infty} \leq \frac{\|c\|_\infty D_1(X) + \|g\|_\infty D_1(Z) + \epsilon}{\rho - \|\lambda\|_\infty}.$$ 

It is straightforward to verify the choice of $\rho$ ensures $\|Ax^* + Bz^*\|_1 \leq \epsilon$, and finally the complexity result is a direct consequence of Theorem 3. \hfill \Box

The choice of $\rho$ in Theorem 5 serves as a sufficient condition for AUSAL to deliver an $\epsilon$-solution of (1) given any $\lambda \in \mathbb{R}^m$; however, the $O(1/\epsilon^2 d)$ iteration complexity is undesirable. In practice, instead of calling AUSAL just once with some fixed $(\lambda, \rho)$, we should update them in an iterative scheme, which we discuss in the next section.

3.3 ALM and Dual Updates

We present an ALM that uses AUSAL for its subproblem and two different subgradient methods for the update of $(\lambda, \rho)$. This method is appropriate when a pair $(\lambda, \rho)$ that supports exact penalization is not known initially.

The dual function $d(\lambda, \rho)$ is a concave and upper-semicontinuous function in $(\lambda, \rho)$. Let $(\bar{x}, \bar{z})$ be a pair of solution returned by AUSAL with input $(\bar{\lambda}, \bar{\rho}, \epsilon)$. Then for all $(\lambda, \rho) \in \mathbb{R}^m \times \mathbb{R}_+$, it holds

$$d(\lambda, \rho) \leq c^T \bar{x} + g^T \bar{z} + \langle \lambda, A\bar{x} + B\bar{z} \rangle + \rho \|A\bar{x} + B\bar{z}\|_1$$

$$= c^T \bar{x} + g^T \bar{z} + \langle \lambda - \bar{\lambda}, A\bar{x} + B\bar{z} \rangle + \bar{\rho} \|A\bar{x} + B\bar{z}\|_1$$

$$+ \langle \bar{\lambda} - \lambda, A\bar{x} + B\bar{z} \rangle + (\rho - \bar{\rho}) \|A\bar{x} + B\bar{z}\|_1$$

$$\leq d(\bar{\lambda}, \bar{\rho}) + \epsilon + \langle \lambda - \bar{\lambda}, A\bar{x} + B\bar{z} \rangle + (\rho - \bar{\rho}) \|A\bar{x} + B\bar{z}\|_1,$$

where the last inequality is due to the $\epsilon$-optimality of $(\bar{x}, \bar{z})$. Therefore, we have

$$(-d)(\lambda, \rho) \geq (-d)(\bar{\lambda}, \bar{\rho}) - \left[ \frac{A\bar{x} + B\bar{z}}{\|A\bar{x} + B\bar{z}\|_1} \right]^T \left[ \frac{\lambda - \bar{\lambda}}{\rho - \bar{\rho}} \right] - \epsilon,$$

or equivalently,

$$- \left[ \frac{A\bar{x} + B\bar{z}}{\|A\bar{x} + B\bar{z}\|_1} \right] \in \partial_\epsilon (-d)(\lambda, \rho). \quad (21)$$

When $\epsilon = 0$, the above inclusion yields a convex subgradient. Therefore, we can use the primal information $(\bar{x}, \bar{z})$ to construct an $\epsilon$-subgradient of $-d$ at $(\bar{\lambda}, \bar{\rho})$ and apply an inexact subgradient method to solve the dual problem (6). Note that the compactness on $X$ and $Z$ ensures that $d(\lambda, \rho)$ is well defined for all $\lambda \in \mathbb{R}^m$ and $\rho \geq 0$; the exact penalization of MILP from Theorem 1 guarantees that $\text{Argmax } d(\lambda, \rho)$ is nonempty.

There are various subgradient methods under different assumptions and variations, such as the exactness of subgradient, choices of step size, and stopping criteria. Below we provide two specific implementations. Since the analysis is standard, we place the proof in Appendix A only for completeness. Beyond the two implementations below, we can apply methods [6–8, 10] with AUSAL as a subroutine to solve the dual problem (6).
Algorithm 2: A Subgradient Variant with Iteration Complexity on Objective Gap

1: **Input** $\epsilon_p \geq 0$;
2: **Initialize** $(\lambda^1, \rho^1)$, and a sequence $\{\tau_k\}_{k \in \mathbb{N}}$ such that $\tau_k > 0$, $\tau_k \to 0$, and $\sum_{k \in \mathbb{N}} \tau_k = +\infty$;
3: for $k = 1, 2, \ldots$ do
4: $(x^k, z^k) \leftarrow \text{AUSAL}(\lambda^k, \rho^k, \epsilon_p)$;
5: set $\alpha_k = \tau_k / \sqrt{2 \|Ax + Bz\|_1}$;
6: $\lambda^{k+1} \leftarrow \lambda^k + \alpha_k (Ax^k + Bz^k)$, $\rho^{k+1} \leftarrow \rho^k + \alpha_k \|Ax^k + Bz^k\|_1$
7: end for

3.3.1 First Subgradient Method and Iteration Complexity

The first subgradient algorithm is presented as Algorithm 2. The constant $\epsilon_p \geq 0$ controls the exactness of the subgradient in (21). If AUSAL returns $(x^k, z^k)$ with $Ax^k + Bz^k = 0$, $(x^k, z^k)$ is an $\epsilon_p$-solution to MILP (1). To study convergence and complexity, we assume $\|Ax^k + Bz^k\|_1 > 0$ for all $k \in \mathbb{N}$, so that Algorithm 2 will produce an infinite sequence $\{(\lambda^k, \rho^k)\}_{k \in \mathbb{N}}$. We introduce the following constants for our analysis:

- $D^* := \text{Argmax}_{\lambda, \rho \geq 0} d(\lambda, \rho)$,
- $d_0 := \min_{(\lambda, \rho) \in D^*} \|\lambda^1 - \lambda\|_1 + |\rho^1 - \rho|$, and
- $M := \max_{x \in \mathcal{X}, z \in \mathcal{Z}} \|Ax + Bz\|_1$.

Consider the step size $\alpha_k$ chosen as in line 5 of Algorithm 2. Other standard choices include $\alpha_k = \epsilon_d/(2M^2)$ or $\alpha_k = \epsilon_d/(2\|Ax^k + Bz^k\|_1^2)$; their convergence results are similar and thus omitted.

**Theorem 6** The following statements hold.

1. Let $\epsilon_d > 0$. Suppose Algorithm 2 performs $K = [2M^2d_0^2/\epsilon_d^2]$ iterations with $\tau_i = d_0/\sqrt{K}$. Then,

   $$\min_{k \in [K]} \rho^* - d(\lambda^k, \rho^k) \leq \epsilon_p + \epsilon_d.$$

2. Suppose $\epsilon_p = 0$, and the sequence $\{\tau_k\}_{k \in \mathbb{N}}$ also satisfies $0 < \tau_k \leq \tau$ for all $k \in \mathbb{N}$ and $\sum_{k \in \mathbb{N}} \tau_k^2 < +\infty$. Then $(\lambda^k, \rho^k)$ converges to some $(\lambda^*, \rho^*) \in D^*$.

**Proof** See Appendix A.1.

In addition, we can impose a compact feasible region for $(\lambda, \rho)$ and apply the projected subgradient method. The results in Theorem 6 still apply with essentially the same proof. The rationale is that, if Algorithm 2 does not find the optimal dual variable after $K$ iterations, then by Theorem 6, we can estimate the optimality in objective, and expect the best-so-far iterate $(\lambda^k, \rho^k)$ to be close to some optimal dual solution $(\lambda^*, \rho^*)$. Consequently, we can post-process to recover an optimal dual solution.

**Corollary 1** Suppose Algorithm 2 generates a pair $(\lambda^k, \rho^k)$ such that $\|(\lambda^k, \rho^k) - (\lambda^*, \rho^*)\|_\infty \leq l$ for some $(\lambda^*, \rho^*) \in D^*$. Then $(\lambda^k, \rho^k + 2l)$ supports exact penalization. Applying AUSAL with $\lambda = \lambda^k$, $\rho = \max\{\|\lambda^k\|_\infty, \rho^k + 2l + 1\}$, and $\epsilon > 0$ will return an $\epsilon$-solution of the MILP (1).

**Proof** By Theorem 2, there exists $r > 0$ such that for all $u \in B_1(0; r)$, it holds

$$p(u) \geq p(0) + (\lambda^*, u) - \rho^*\|u\|_1 \geq p(0) + (\lambda^k, u) - \rho^k\|u\|_1 - \|\lambda^k - \lambda^*\|_\infty\|u\|_1 - |\rho^k - \rho^*|\|u\|_1 \geq p(0) + (\lambda^k, u) - (\rho^k + 2l)\|u\|_1,$$

so that $(\lambda^k, \rho^k + 2l)$ supports exact penalization. The second claim follows from Theorem 4. \(\Box\)
3.3.2 Second Subgradient Method with Finite Convergence to an Approximate Solution

We present the second variant in Algorithm 3. Now we assume $\epsilon_p$ is strictly positive, and we are interested in finding an $\epsilon_p$-solution of MILP (1).

Algorithm 3: A Subgradient Variant with Finite Convergence to Approximate Solution

1: Input $\epsilon_p > 0$;
2: Initialize $(\lambda^1, \rho^1)$ with $\rho^1 \geq \|\lambda^1\|_\infty$, and some $\tau > 0$;
3: for $k = 1, 2, \cdots$ do
4: $(x^k, z^k) \leftarrow \text{AUSAL}(\lambda^k, \rho^k, \epsilon_p)$;
5: if $\|Ax^k + Bz^k\|_1 \leq \epsilon_p$ then
6: return $(x^k, z^k)$.
7: set $\alpha_k = \tau/\|Ax^k + Bz^k\|_1$;
8: $\lambda^{k+1} \leftarrow \lambda^k + \alpha_k (Ax^k + Bz^k)$, $\rho^{k+1} \leftarrow \max\{\|\lambda^{k+1}\|_\infty, \rho^k + \alpha_k \|Ax^k + Bz^k\|_1\}$;
9: end for

If Algorithm 3 terminates with $(x^k, z^k)$, then $x^k \in X$, $z^k \in Z$, and $\|Ax^k + Bz^k\|_1 \leq \epsilon_p$. Since $\rho^k \geq \|\lambda^k\|_\infty$, the same derivation in (19) gives $c^T x^k + g^T z^k \leq p^* + \epsilon_p$. Therefore, $(x^k, z^k)$ is indeed an $\epsilon_p$-solution of the MILP (1).

Theorem 7 Algorithm 3 returns an $\epsilon_p$-solution of MILP (1) in a finite number of iterations. 

Proof See Appendix A.2. \qed

4 An ADMM-Based Method

In the section, we present an ADMM variant and state its convergence result. We start by looking into the penalty formulation of MILP (1)

$$\min_{x \in X, z \in \mathbb{R}} c^T x + g^T z + \rho \|Ax + Bz\|_1$$

for some $\rho > 0$. We further adopt the new notation

$$R_\rho (z) := \min_{x \in X} c^T x + \rho \|Ax + Bz\|_1, \quad (22)$$

in replace of $R(z)$ in (12), where the dependency on $\lambda$ vanishes and $R_\rho$ is only parameterized by the penalty $\rho$. As a special case of (12), $R_\rho$ is also Lipschitz continuous with modulus $\rho \|B\|_1$.

We firstly introduce a new point of view of generating valid lower approximation for $R_\rho$ in Section 4.1, and then present our proposed ADMM variant in Section 4.2. Convergence and complexity results are given in Section 4.3.

4.1 Generating Cuts from the Dual Perspective

To begin with, for a given $z \in Z$ and $\rho > 0$, the evaluation $R_\rho (z)$ requires solving the optimization problem (22), which can be regarded as a primal problem in $x$. We consider a relaxation of this primal problem as follows: given $\mu \in \mathbb{R}^m$ and $\beta \geq 0$, let

$$P(z, \mu, \beta) := \min_x c^T x + \langle \mu, Ax + Bz \rangle + \beta \|Ax + Bz\|_1. \quad (23)$$

Notice that

$$P(z, \mu, \beta) \leq \min_x c^T x + (\beta + \|\mu\|_\infty) \|Ax + Bz\|_1 \leq R_\rho (z) \quad (24)$$
for all $(\mu, \beta) \in A(\rho)$, where
\[ A(\rho) := \{ (\mu, \beta) \in \mathbb{R}^{m+1} \mid \beta \geq 0, \beta + \|\mu\|_\infty \leq \rho \}. \] (25)

Inequality (24) is a weak duality result in the sense that $P(z, \mu, \beta)$ provides a lower bound for $R_\rho(z)$ when the pair $(\mu, \beta)$ is constrained in $A(\rho)$. It turns out that strong duality also holds:
\[
R_\rho(z) = \max_{(\mu, \beta) \in A(\rho)} P(z, \mu, \beta) \\
= \max_{(\mu, \beta) \in A(\rho)} \min_{x \in X} c^T x + \langle \mu, Ax + Bz \rangle + \beta \|Ax + Bz\|_1,
\] (26)
simply due to the fact that $(0, \rho) \in A(\rho)$ and $P(z, 0, \rho) = R_\rho(z)$ by definition. Given some $\bar{z}$ and $(\bar{\mu}, \bar{\beta}) \in A(\rho)$, suppose we solve the problem (23), then it holds that for any $z \in Z$
\[
\hat{r}(z; \bar{z}, \bar{\mu}, \bar{\beta}) := P(\bar{z}, \bar{\mu}, \bar{\beta}) + \langle \bar{\mu}, Bz - B\bar{z} \rangle - \bar{\beta} \|Bz - B\bar{z}\|_1
\] (27)
is a lower approximation of $R_\rho(z)$.

**Definition 4** We call the inequality $R_\rho(z) \geq \hat{r}(z; \bar{z}, \bar{\mu}, \bar{\beta})$ an augmented Lagrangian cut (AL cut) at $\bar{z}$ parameterized by $(\bar{\mu}, \bar{\beta})$. We say the cut is tight at $z$ if $R_\rho(z) = \hat{r}(z; \bar{z}, \bar{\mu}, \bar{\beta})$.

We note that an AL cut is not necessarily tight since $P(\bar{z}, \bar{\mu}, \bar{\beta}) < R_\rho(\bar{z})$ when $(\bar{\mu}, \bar{\beta}) \in A(\rho)$ is not optimal for (26). The additional linear term $\langle \bar{\mu}, Bz - B\bar{z} \rangle$ corresponds to a rotation around the pivot $(\bar{z}, P(\bar{z}, \bar{\mu}, \bar{\beta})) \in \mathbb{R}^{d+1}$. So unlike the reverse norm cut, an AL cut is not symmetrically pointing downwards. In addition, since $\|\bar{\mu}\|_\infty + \bar{\beta} \leq \rho$, the AL cut may have a smaller Lipschitz constant than $R_\rho$. Geometrically, the rotation effect and smaller Lipschitz constant allow an AL cut to be “fatter” than $R_\rho$ and thus covers a wider range in $Z$ than a reverse norm cut. Moreover, a smaller value of $\bar{\beta} + \|\bar{\mu}\|_\infty$ can be preferable for optimization solvers.

AL cuts appear recently in [1, 37] to approximate nonconvex value functions in multistage stochastic optimization. Ahmed et al. [1] observed that AL cuts can be numerically more efficient than reverse norm cuts, but they didn’t provide a systematic analysis for the convergence of their algorithm using AL cuts. In order to get a tight cut, Zhang and Sun [37] assumed an oracle for (a similar version of) the max-min problem (26) is available; in practice, this requires solving a series of MILP problems of the form (23), for example, in a double-looped ALM. As we will show in the next subsection, the proposed ADMM variant avoids the max-min problem (26), and only solves a single MILP of the form (23) in each iteration.

### 4.2 ADMM with AL cuts

We can approximate $R_\rho(z)$ using AL cuts (27). The $x$-subproblem has the following form: at iteration $k$, given $z^{k-1}$ and some dual information $(\mu^k, \beta^k)$, we obtain $x^k$ as
\[
x^k \in \operatorname{Argmin}_{x \in X} c^T x + \langle \mu^k, Ax + Bz^{k-1} \rangle + \beta^k \|Ax + Bz^{k-1}\|_1.
\] (28)
Then an AL cut (27) is generated and appended to the $z$-subproblem to obtain $z^k$:

$$\min_{z \in \mathbb{Z}, t} g^\top z + t$$

subject to

$$t \geq \bar{r}(z; z^{j-1}, \mu^j, \beta^j) \quad \forall j \in [k].$$

(29a) (29b)

Algorithm 4: An ADMM Framework with AL Cuts

1: Initialize $z^0 \in \mathbb{Z}$, $(\mu^1, \beta^1) \in \mathbb{R}^m \times \mathbb{R}_+$
2: for $k = 1, 2, \cdots$ do
3: obtain $x^k$ as a minimizer of (28);
4: obtain $(z^k, t^k)$ as a minimizer of (29);
5: update $(\mu^{k+1}, \beta^{k+1}) \in \mathbb{R}^m \times \mathbb{R}_+$;
6: end for

A conceptual ADMM is described in Algorithm 4. One major difference between our proposed ADMM variant and the classic ADMM lies in the $z$-subproblem. In a traditional ADMM framework, the $z$-subproblem has a similar structure as the $x$-subproblem, i.e.,

$$\min_{z \in \mathbb{Z}} g^\top z + \left( c^\top x^k + (\mu^k, Ax^k + Bz) + \beta^k \sigma(Ax^k + Bz) \right),$$

(30)

where $\sigma(\cdot) = \frac{1}{2}\|\cdot\|_2^2$ (proximal Lagrangian) or $\sigma(\cdot) = \|\cdot\|_1$ (sharp Lagrangian) is usually used. Problem (29) consists of valid lower approximation for $R_\rho(z)$, which will be refined over iterations. In contrast, (30) approximates this dependency locally by only one piece (inside the parenthesis), which is not necessarily a lower approximation. This might shed some light on why the classic ADMM may not be able to converge to global optimal solutions.

We do not specify how $(\mu^{k+1}, \beta^{k+1})$ is updated in Algorithm 4. Instead, we present a set of assumptions on the selection of $(\mu^{k+1}, \beta^{k+1})$ to establish convergence results. We provide a geometric intuition in the next section. Any updates that meet the assumptions will work.

4.3 Convergence and Complexity

In this subsection, we study the convergence of the ADMM framework. In particular, we establish asymptotic convergence in Theorem 8 under Assumption 2 and an iteration complexity result in Theorem 9 under (a slightly stronger) Assumption 3.

Assumption 2 Let $\rho > 0$ be the minimum penalty that supports exact penalization for MILP (1). Suppose $(\mu^k, \beta^k)$ are chosen such that

1. $\beta^k - \|\mu^k\|_\infty \geq \rho$ for all sufficiently large $k \in \mathbb{N}$;
2. $\beta^k + \|\mu^k\|_\infty \leq \overline{\rho}$ for all $k \in \mathbb{N}$ for some $\overline{\rho} > 0$.

Remark 1

1. Part 1 of Assumption 2 avoids too many loose cuts. It guarantees that for sufficiently large $k$, the peak of the AL cut reaches at least $R_\mu$. Otherwise the objective of the $z$-subproblem is always a loose lower bound of $\rho$.
2. Part 2 can be satisfied if $\beta^k$ is bounded away from $\|\mu^k\|_\infty$ by some constant. However, we do not want $\beta^k$ to go to infinity, as the resulting generalized cut is very "slim". This is ensured by part 2. "Slim" cuts are not desirable since we will need a lot more such cuts to construct a good approximation. The constant $\overline{\rho}$ also appears in the complexity result in Theorem 9.
3. The exact values of $\rho$ and $\overline{\rho}$ are not required during numerical implementation. For example, one can update the dual variables like in the classic ADMM: $\mu^{k+1} = \mu^k + \beta^k (Ax^k + Bz^k)$, and then increase $\beta^{k+1}$ accordingly. The bound $\overline{\rho}$ is required for analysis only; treat it as a constant large enough such that projecting $(\mu, \beta)$ onto some bounded set is not necessary.

Recall $p^*$ is the optimal value of the MILP (1) and that

$$t^k = \max_{j \in [k]} \{\bar{r}(z^k; z^{j-1}, \mu^j, \beta^j)\}.$$ 

**Theorem 8** Suppose Assumption 2 holds. Let $\{(z^k, t^k, x^{k+1})\}_{k \in \mathbb{N}}$ be the sequence generated by Algorithm 4, and $(x^*, z^*)$ be a limit point of $\{(z^k, x^{k+1})\}_{k \in \mathbb{N}}$. The following claims holds.

1. $\{g^T z^k + t^k\}_{k \in \mathbb{N}}$ converges to $p^*$ monotonically from below.
2. $p^* = g^T z^* + R_p(z^*) = g^T z^* + R_\overline{\rho}(z^*)$.
3. $(x^*, z^*)$ is an optimal solution to MILP (1).

**Proof**

1. We firstly prove the sequence $\{g^T z^k + t^k\}_{k \in \mathbb{N}}$ converges to $p^*$ monotonically from below. Since $g^T z^k + t^k$ is the optimal value of problem (29), whose feasible region is shrinking over $k \in \mathbb{N}$, we know $\{g^T z^k + t^k\}_{k \in \mathbb{N}}$ is monotone non-decreasing. Since $\|\mu^k\|_\infty + \beta^k \leq \overline{\rho}$, it follows that $(\mu^*, \beta^*) \in \Lambda(\overline{\rho})$ for all for all $k \in \mathbb{N}$, i.e., all cuts of the form (29b) are valid lower approximations of $R_\overline{\rho}(z)$. As a result, we have

$$g^T z^k + t^k \leq \min_{z \in \mathcal{Z}} \ g^T z + R_\overline{\rho}(z) = p^*,$$

where the equality is due to $\overline{\rho}$ supports exact penalization. The sequence $\{g^T z^k + t^k\}_{k \in \mathbb{N}}$ is non-decreasing and bounded from above by $p^*$, so it converges to some $\overline{p} \leq p^*$. Next let $z^*$ be a limit point, and $\{z^{k_j}\}_{j \in \mathbb{N}}$ be the subsequence convergent to $z^*$. Notice that by the Hölder’s inequality and Assumption 2, we have for large enough $j$,

$$P(z^{k_j}, \mu^{k_j-1+1}, \beta^{k_j-1+1})$$

$$= \min_{x \in X} c^T x + \langle \mu^{k_j-1+1}, Ax + Bz^{k_j-1} \rangle + \beta^{k_j-1+1} \|Ax + Bz^{k_j-1}\|_1$$

$$\geq \min_{x \in X} c^T x + (\beta^{k_j-1+1} - \|\mu^{k_j-1+1}\|_\infty) \|Ax + Bz^{k_j-1}\|_1$$

$$\geq \min_{x \in X} c^T x + \overline{\rho} \|Ax + Bz^{k_j-1}\|_1 = R_\overline{\rho}(z^{k_j-1}).$$

(31)

In addition,

$$g^T z^{k_j} + t^{k_j}$$

$$\geq g^T z^{k_j} + P(z^{k_j-1}, \mu^{k_j-1+1}, \beta^{k_j-1+1}) + \langle \mu^{k_j-1+1}, Bz^{k_j} - Bz^{k_j-1} \rangle$$

$$- \beta^{k_j-1+1} \|Bz^{k_j} - Bz^{k_j-1}\|_1$$

$$\geq g^T z^{k_j} + R_\overline{\rho}(z^{k_j-1}) + \langle \mu^{k_j-1+1}, Bz^{k_j} - Bz^{k_j-1} \rangle - \beta^{k_j-1+1} \|Bz^{k_j} - Bz^{k_j-1}\|_1,$$

where the first inequality is due to (29b), and the second inequality is due to (31). Taking limit on both sides of the above inequality gives

$$p^* = \bar{p} = \lim_{j \to \infty} g^T z^{k_j} + t^{k_j} \geq g^T z^* + R_\overline{\rho}(z^*) \geq p^*,$$

where the second inequality is due to the continuity of $R_\overline{\rho}$, the fact that $Bz^{k_j} - Bz^{k_j-1}$ vanishes, and the boundedness of $\{(\mu^k, \beta^k)\}_{k \in \mathbb{N}}$ by Assumption 2. So we conclude that $g^T z^* + R_\overline{\rho}(z^*) = p^* = \bar{p}$. Theorem 8 is proved.
2. We have already shown the first equality. Let

\[ x(\rho, z^*) \in \text{Argmin}_{x \in X} c^T x + \rho \| A x + B z^* \|_1. \]

Then it holds that

\[ (x(\rho, z^*), z^*) \in \text{Argmin}_{x \in X, z \in Z} c^T x + g^T z + \rho \| A x + B z \|_1 \]

\[ = \text{Argmin}_{x \in X, z \in Z} \{ c^T x + g^T z \mid A x + B z = 0 \}. \]

So we know \( A x(\rho, z^*) + B z^* = 0 \) and \( c^T x(\rho, z^*) + g^T z^* = p^* \). Consequently,

\[ p^* = g^T z^* + R_\rho(z^*) \leq g^T z^* + R_\rho(z^*) \]

\[ = g^T z^* + \min_{x \in X} c^T x + \rho \| A x + B z^* \|_1 \]

\[ \leq g^T z^* + c^T x(\rho, z^*) = p^*. \]

This proves the second equality.

3. It remains to show \( (x^*, z^*) \) is an optimal solution to MILP (1). Again, let \( \{(z^{k_j}, x^{k_j+1})\}_{j \in \mathbb{N}} \) be the subsequence convergent to \( (z^*, x^*) \). Notice that

\[ g^T z^{k_j} + R_\rho(z^{k_j}) \]

\[ \leq g^T z^{k_j} + c^T x^{k_j+1} + (\mu^{k_j+1}, A x^{k_j+1} + B z^{k_j}) + \beta^{k_j+1} \| A x^{k_j+1} + B z^{k_j} \|_1 \]

\[ \leq g^T z^{k_j} + R_\rho(z^{k_j}); \]

assuming without loss generality that \( \lim_{j \to \infty} (\mu^{k_j+1}, z^{k_j+1}) = (\mu^*, z^*) \) and taking limit on both sides of the above two inequalities, we have

\[ c^T x^* + g^T z^* + (\mu^*, A x^* + B z^*) + \beta^* \| A x^* + B z^* \|_1 = p^*, \]

where the equality holds due to the second claim. It suffices to show \( A x^* + B z^* = 0 \). Suppose not, then by Theorem 1, \( (x^*, z^*) \notin \text{Argmin}_{x \in X, z \in Z} c^T x + g^T z + \rho \| A x + B z \|_1 \); therefore,

\[ p^* = c^T x^* + g^T z^* + (\mu^*, A x^* + B z^*) + \beta^* \| A x^* + B z^* \|_1 \]

\[ \geq c^T x^* + g^T z^* + \rho \| A x^* + B z^* \|_1 \]

\[ > \min_{x \in X, z \in Z} c^T x + g^T z + \rho \| A x + B z \|_1 = p^*, \]

where is a desired contradiction. This completes the proof. \( \square \)

In order to establish iteration complexity of Algorithm 4, we need a slightly stronger version of Assumption 2.

**Assumption 3** Let \( \bar{\rho} - 1 > 0 \) be the minimum penalty that supports exact penalization for MILP (1). Suppose \( (\mu^k, \beta^k) \) are chosen such that

1. \( \beta^k - \| \mu^k \|_\infty \geq \bar{\rho} \) for all \( k \in \mathbb{N} \);
2. \( \beta^k + \| \mu^k \|_\infty \leq \bar{\rho} \) for all \( k \in \mathbb{N} \).

**Theorem 9** Suppose Assumption 3 holds, and \( Z \subseteq \overline{B}_1(\bar{z}; R) \) for some \( \bar{z} \in \mathbb{R}^d \) and radius \( R > 0 \). Let \( \epsilon > 0 \), and \( \{(z^k, t^k)\}_{k \in \mathbb{N}} \) be the sequence generated by line 4 of Algorithm 4. Then Algorithm 4 finds a solution \( (z^K, t^K) \) satisfying

1. (optimality gap) \( p^* - (g^T z^K + t^K) \leq \epsilon, \)
2. (approximate solution) \((\hat{x}^K, z^K)\) is an \(\epsilon\)-solution to MILP (1), where
\[
\hat{x}^K \in \text{Argmin}_{x \in X} c^\top x + \rho \|A x + B z^K\|_1,
\]
in no more than
\[
K \leq \left(1 + \frac{2(\rho + \overline{\rho})\|B\|_1 R}{\epsilon}\right)^d
\] iterations.

Proof For all nonnegative integers \(i < j\), we have
\[
\max\{p^* - (g^\top z^j + t^j), g^\top z^j + R_\mu(z^j) - p^*\} \\
\leq R_p(z^j) - t^j \\
\leq R_p(z^j) - P(z^j, \mu^{i+1}, \beta^{i+1}) - (\mu^{i+1}, B z^j - B z^i) + \beta^{i+1}\|B z^j - B z^i\|_1 \\
\leq R_p(z^j) - R_\mu(z^j) + (\|\mu^{i+1}\|_\infty + \beta^{i+1})\|B z^j - B z^i\|_1 \\
\leq (\rho + \overline{\rho})\|B\|_1\|z^j - z^i\|_1,
\]
where the first inequality is due to \(g^\top z^j + t^j \leq p^* \leq g^\top z^j + R_\mu(z^j)\), the second inequality is due to (31), the third inequality is due to (31), and the last inequality is due to Assumption 3 and \(R_p\) being \(\rho\|B\|_1\)-Lipschitz. Let \(K\) be the smallest index such that
\[
\max\{p^* - (g^\top z^K + t^K), g^\top z^K + R_\mu(z^K) - p^*\} \leq \epsilon.
\]
Then we must have \(\|z^i - z^j\|_1 > \epsilon/\|B\|_1\) for all \(0 \leq i < j \leq K - 1\), since otherwise (33) implies
\[
\max\{p^* - (g^\top z^j + t^j), g^\top z^j + R_\mu(z^j) - p^*\} \leq (\rho + \overline{\rho})\|B\|_1\|z^j - z^i\|_1 \leq \epsilon,
\]
contradicting to the choice of \(K\). Using the same argument as the proof of Theorem 3, Part 2, we can bound \(K\) by (32).

It remains to show the claimed \((\hat{x}^K, z^K)\) is an \(\epsilon\)-solution of the MILP (1). Since \(g^\top z^K + R_\mu(z^K) - p^* \leq \epsilon\) and \(c^\top \hat{x}^K + \overline{\rho}\|A \hat{x}^K + B z^K\| = R_\mu(z^K)\), we know \((\hat{x}^K, z^K)\) is an \(\epsilon\)-optimal solution to the problem \(\min_{x \in X, z \in Z} c^\top x + g^\top z + \rho \|A x + B z\|_1\). Now by Assumption 3 and the proof of Theorem 4, we conclude that \((\hat{x}^K, z^K)\) is an \(\epsilon\)-solution of MILP (1).

\section{5 Implementation Issues and Numerical Experiments}

5.1 Modification of Historical Cuts in ALM

Consider the ALM framework in Section 3. Given the current \((\bar{\lambda}, \bar{\rho})\), we use AUSAL (Algorithm 1) to solve the primal subproblem (5); in each iteration, given some \(\bar{z} \in Z\), we approximate the function \(R(z; \bar{\lambda}, \bar{\rho})\) by adding a reverse norm cut of the form
\[
R(z; \bar{\lambda}, \bar{\rho}) \geq R(\bar{z}; \bar{\lambda}, \bar{\rho}) - K_\rho\|z - \bar{z}\|_1
\]
to the \(z\)-subproblem, where \(R(z; \bar{\lambda}, \bar{\rho})\) is defined in (12) and explicitly parameterized by \((\bar{\lambda}, \bar{\rho})\). When the current AUSAL terminates, we update a new pair of dual variables \((\lambda, \rho)\), and start the next AUSAL.
Notice that cuts (34) generated with \((\lambda, \rho)\) are not in general valid anymore for \(R(z; \lambda, \rho)\). Naively, we can remove all previous cuts when starting a new AUSAL, but this will cause a loss of historical information. Instead, we can modify old cuts so that they always stay valid for the latest dual information \((\lambda, \rho)\). Assuming the penalty is nondecreasing, i.e., \(\rho \geq \bar{\rho}\), we have

\[
R(z; \lambda, \rho) = \min_{x \in X} c^T x + \langle \lambda, Ax \rangle + \rho \|Ax + B\bar{z}\|_1 \\
= \min_{x \in X} c^T x + \langle \bar{\lambda}, Ax \rangle + \bar{\rho} \|Ax + B\bar{z}\|_1 \\
+ \langle \lambda - \bar{\lambda}, Ax \rangle + (\rho - \bar{\rho}) \|Ax + B\bar{z}\|_1 \\
\geq R(\bar{z}; \bar{\lambda}, \bar{\rho}) - \|\lambda - \bar{\lambda}\|_\infty \left( \max_{x \in X} \|Ax\|_1 \right),
\]

which follows

\[
R(z; \lambda, \rho) \geq R(\bar{z}; \bar{\lambda}, \bar{\rho}) - \|\lambda - \bar{\lambda}\|_\infty \left( \max_{x \in X} \|Ax\|_1 \right) - K_\rho \|z - \bar{z}\|_1
\]

is a valid inequality for all \(R(z; \lambda, \rho)\) over \(z \in Z\). The new cuts (35) can be obtained by modifying the coefficient \(K_\rho\) and constant \(R(\bar{z}; \bar{\lambda}, \bar{\rho})\) in old cuts (34). Though these modified cuts may not be tight, empirically we do observe they help ALM to maintain a more stable lower bound.

5.2 Comparison with Primal and Dual Decompositions

In this section, we present numerical experiments on the generic multi-block MILP problem for the purpose of proof of concept:

\[
\min_{x_1 \in X_1, \ldots, x_p \in X_p} c_1^T x_1 + \cdots + c_p^T x_p \quad \text{subject to} \quad A_1 x_1 + \cdots + A_p x_p \leq b. \tag{36}
\]

In view of the discussion on (3), our proposed algorithms are applicable to the two-block reformulation of the above problem. Authors of [9, 31] apply primal and dual decomposition algorithms, respectively, to solve a restricted problem, where the right-hand side vector \(b\) is replaced by \(b - \sigma\). The introduction of the vector \(\sigma \in \mathbb{R}^{m_1}\) makes it possible to recover a feasible solution of (36) from the Lagrangian dual framework.

We compare the performance of the proposed algorithms with the primal and dual decomposition algorithms. We set \(p = 100\) and \(m = 10\). For each block \(i \in [p]\), we let \(X_i = \{x \in [0, 1, \ldots, 60] \times [-60, 60] \mid E_i x_i \leq f_i\}\); we generate \((E_i, f_i, c_i)\) according to [9, Section 5] and \(A_i\) with standard Gaussian entries. The restriction vector \(\sigma\) is calculated according to [9, Section 4.1] for both primal and dual decomposition algorithms, as the one used in [31] is typically larger in magnitude and results in infeasibility for tested instances. For different right-hand side vector \(b \in \{1500, 1200, 1000, 800, 600\} \times e\), where \(e\) is the vector of all ones, we generate five instances. Notice that as the resource vector \(b\) decreases, we expect the problem to become more challenging.

We initialize the ALM dual variable and penalty as \((\lambda^1, \rho^1) = (0, 0.01)\). Each call of AUSAL is terminated if either the relative gap is less than 5% or a total of 30 iterations are reached. When the current AUSAL returns \((x^k, z^k)\), we update the dual variable \(\lambda^{k+1}\) as in line 6 of Algorithm 2 with \(\tau_k = 0.1/k\), and set \(\rho^{k+1} = 1.1 \rho^k\). For ADMM, we initialize the dual variable and penalty as \((\mu^1, \beta^1) = (0, 0.01)\), and update them by

\[
\mu^{k+1} = \mu^k + \beta^k (A x^k + B z^{k-1}), \quad \text{and} \quad \beta^{k+1} = \begin{cases} 1.1 \beta^k & \text{if } 5 \text{ divides } k \\ \beta^k & \text{otherwise.} \end{cases}
\]

All MILP subproblems are solved by Gurobi 8.1.0. Since the comparison involves both single-looped (ADMM and dual decomposition) and double-looped (ALM and primal decomposition) algorithms,
we set a time limit of 20 minutes (MAX) for each generated instance, and report the best duality gap achieved, i.e., (upper bound – lower bound)/|upper bound|, as well as the corresponding time in Table 1.

| b   | Gap(%) | Time(s) | Gap(%) | Time(s) | Gap(%) | Time(s) | Gap(%) | Time(s) |
|-----|--------|---------|--------|---------|--------|---------|--------|---------|
| 1500 | 0.00   | 1.21    | 0.00   | 1.97    | 133.06 | 577.09  | 4.74   | 2.94    |
| 1200 | 0.00   | 2.12    | 0.00   | 2.04    | 151.11 | 356.16  | 14.96  | 2.50    |
| 1000 | 0.00   | 2.14    | 0.00   | 2.53    | 162.88 | 619.04  | 12.97  | 2.90    |
| 800  | *4.82  | MAX     | 0.00   | 233.12  | 136.37 | 938.75  | 36.94  | 2.95    |
| 600  | *6.72  | MAX     | 0.00   | 160.13  | 144.29 | 420.02  | 24.02  | 2.53    |
| 400  | *5.73  | MAX     | 0.00   | 10.76   | 126.77 | 778.19  | 10.28  | 2.12    |
| 200  | *7.14  | MAX     | 0.00   | 57.99   | 178.48 | 447.01  | 40.04  | 2.77    |

Table 1: Comparison with primal and dual decomposition on multi-block MILP

The proposed ALM and ADM are able to find global optimal solutions in a few iterations for relatively large values of $b$. However, as the magnitude of $b$ decreases, ALM and ADM have difficulties locating feasible solutions for some cases, where we use the augmented Lagrangian function value as an upper bound estimate to calculate the gap (marked by “*”), and report “MAX” in the time column. For such cases, ALM and ADM cannot terminate successfully within the time limit, and we observe the $z$-subproblem suffers from increasingly long computation time as the two algorithms proceed. This is because the number of binary variables and constraints in the $z$-subproblem will grow linearly with respect to the number of cuts, which makes the subproblem more difficult to solve over iterations. In contrast, the primal and dual decomposition algorithms are able to obtain feasible solutions in all tested instances. Nevertheless, since they are essentially solving a restricted problem, convergence to solutions with large duality gaps are observed.

In terms of computation time, the dual decomposition algorithm is consistently fast since it is single-looped and maintains simple subproblems. The primal decomposition algorithm is double-looped, and each local subproblem is solved by a cutting-plane subroutine, which can affect the overall computation time from two perspectives: on one hand, the cutting-plane subroutine itself can be slow when applied to nonsmooth problems; on the other hand, the quality of solutions returned by the cutting-plane subroutine can further influence the convergence of the outer-level subgradient method. We indeed observe that sometimes the cutting-plane subroutine is not able to terminate within 200 iterations, and consequently only an inexact subgradient is available for the outer level update. This might also count for the large duality gaps in Table 1. Finally, the proposed ALM and ADM are usually fast in the early stage: when they converge successfully for instances in Table 1, the solution time is less than or comparable to that of dual decomposition. However, as we have commented earlier, they can be slow in the long run, especially after over a few hundreds of cuts are appended to the $z$-subproblem.
6 Concluding Remarks

In this paper, we study generic MILP problems with two blocks of variables \(x\) and \(z\). We propose an algorithm named AUSAL that alternatively updates \(x\) and \(z\) in the augmented Lagrangian function, and provide its convergence and complexity results. AUSAL can be directly applied to the penalty formulation, or embedded inside ALM to solve the augmented Lagrangian dual problem. We also propose a single-looped ADMM variant, which is built upon the AL cut introduced in [1, 37]. Different from the procedure used in the previous two references, we obtain an AL cut by solving a single augmented Lagrangian relaxation in variable \(x\); compared to existing ADMM works, our ADMM variant allows a more flexible update scheme for the dual variable and penalty, and is guaranteed to converge to a global optimal solution with iteration complexity estimate. When certain block-angular structure is present, the update of \(x\) can be further decomposed and solved in parallel in both algorithm.

We test the proposed ALM and ADMM variant on multi-block MILP problems, and compare with existing decompositions algorithms. Admittedly, the update of \(z\) variable in both algorithms requires solving a MILP problem with an increasing size of variables and constraints, which can be the computational bottleneck for large and dense problems. We are interested in investigating more practical subproblem oracles, i.e., managing a controllable size of cuts, or new methodologies to approximate the dependency between \(x\) and \(z\). We leave these in the future work.

Acknowledgements We would like to thank the authors of [9] for making their primal-decomposition code available and the authors of [37] for the discussion on AL cuts’ numerical performance. A part of this work was done during with the internship of Alibaba (US) Innovation Research.

References

1. Ahmed, S., Cabral, F.G., da Costa, B.F.P.: Stochastic lipschitz dynamic programming. Mathematical Programming pp. 1–39 (2020)
2. Alavian, A., Rotkowitz, M.C.: Improving ADMM-based optimization of mixed integer objectives. In: 2017 51st Annual Conference on Information Sciences and Systems (CISS). IEEE (2017)
3. Bertsekas, D.P.: Constrained Optimization and Lagrange Multiplier Methods. Academic press (2014)
4. Blair, C.E., Jeroslow, R.G.: The value function of a mixed integer program: I. Discrete Mathematics 19(2), 121–138 (1977)
5. Boland, N., Christiansen, J., Dandurand, B., Eberhard, A., Oliveira, F.: A parallelizable augmented Lagrangian method applied to large-scale non-convex-constrained optimization problems. Mathematical Programming 175(1-2), 503–536 (2019)
6. Burachik, R.S., Gasimov, R.N., Ismayilova, N.A., Kaya, C.Y.: On a modified subgradient algorithm for dual problems via sharp augmented Lagrangian. Journal of Global Optimization 34(1), 55–78 (2006)
7. Burachik, R.S., Iusem, A.N., Melo, J.G.: A primal dual modified subgradient algorithm with sharp Lagrangian. Journal of Global Optimization 46(3), 347–361 (2010)
8. Burachik, R.S., Iusem, A.N., Melo, J.G.: An inexact modified subgradient algorithm for primal-dual problems via augmented Lagrangians. Journal of Optimization Theory and Applications 157(1), 108–131 (2013)
9. Camisa, A., Notarnicola, I., Notarstefano, G.: A primal decomposition method with suboptimality bounds for distributed mixed-integer linear programming. In: 2018 IEEE Conference on Decision and Control (CDC), pp. 3391–3396, IEEE (2018)
10. Cordova, M., de Oliveira, W., Sagastizábal, C.: Revisiting augmented Lagrangian duals. Available on: http://www.optimization-online.org/DB_HTML/2020/03/7709.html (2020)
11. Falsone, A., Margellos, K., Prandini, M.: A distributed iterative algorithm for multi-agent MILPs: finite-time feasibility and performance characterization. IEEE control systems letters 2(4), 563–568 (2018)
12. Falsone, A., Margellos, K., Prandini, M.: A decentralized approach to multi-agent MILPs: finite-time feasibility and performance guarantees. Automatica 103, 141–150 (2019)
13. Feizollahi, M.J., Ahmed, S., Sun, A.: Exact augmented Lagrangian duality for mixed integer linear programming. Mathematical Programming 161(1-2), 365–387 (2017)
14. Gasimov, R.N.: Augmented Lagrangian duality and nondifferentiable optimization methods in nonconvex programming. Journal of Global Optimization 24(2), 187–203 (2002)
15. Hestenes, M.R.: Multiplier and gradient methods. Journal of optimization theory and applications 4(5), 303–320 (1969)
Lemma 3 Let \( \{ (\lambda^k, \rho^k) \}_{k \in \mathbb{N}} \) be the sequence of iterates generated by Algorithm 2. Then for all \( K \geq 1 \), it holds that

\[
\min_{k \in [K]} p^* - d(\lambda^k, \rho^k) \leq \frac{M}{\sqrt{2}} \sum_{k=1}^{K} \tau_k + \epsilon_p.
\]

Proof To simplify notation, we denote \( w^k = (\lambda^k, \rho^k) \), and denote \( w^* = (\lambda^*, \rho^*) \) to be the maximizer in the definition of \( d_0 \). We also denote the \( \epsilon \)-subgradient of \( -d(w) \) by \( \gamma^k \), and it holds

\[
\| \gamma^k \|_2^2 = \| A\lambda^k + Bz^k \|_2^2 + \| Az^k + Bz^k \|_2^2 \leq 2 \| A\lambda^k + Bz^k \|_2^2 \leq 2M^2.
\]

A Proof

A.1 Proof of Theorem 6

We first state a standard lemma regarding the progress in objective value.

Lemma 3 Let \( \{ (\lambda^k, \rho^k) \}_{k \in \mathbb{N}} \) be the sequence of iterates generated by Algorithm 2. Then for all \( K \geq 1 \), it holds that

\[
\min_{k \in [K]} p^* - d(\lambda^k, \rho^k) \leq \frac{M}{\sqrt{2}} \sum_{k=1}^{K} \tau_k + \epsilon_p.
\]

Proof To simplify notation, we denote \( w^k = (\lambda^k, \rho^k) \), and denote \( w^* = (\lambda^*, \rho^*) \) to be the maximizer in the definition of \( d_0 \). We also denote the \( \epsilon \)-subgradient of \( -d(w) \) by \( \gamma^k \), and it holds

\[
\| \gamma^k \|_2^2 = \| A\lambda^k + Bz^k \|_2^2 + \| Az^k + Bz^k \|_2^2 \leq 2 \| A\lambda^k + Bz^k \|_2^2 \leq 2M^2.
\]

A Proof

A.1 Proof of Theorem 6

We first state a standard lemma regarding the progress in objective value.

Lemma 3 Let \( \{ (\lambda^k, \rho^k) \}_{k \in \mathbb{N}} \) be the sequence of iterates generated by Algorithm 2. Then for all \( K \geq 1 \), it holds that

\[
\min_{k \in [K]} p^* - d(\lambda^k, \rho^k) \leq \frac{M}{\sqrt{2}} \sum_{k=1}^{K} \tau_k + \epsilon_p.
\]

Proof To simplify notation, we denote \( w^k = (\lambda^k, \rho^k) \), and denote \( w^* = (\lambda^*, \rho^*) \) to be the maximizer in the definition of \( d_0 \). We also denote the \( \epsilon \)-subgradient of \( -d(w) \) by \( \gamma^k \), and it holds

\[
\| \gamma^k \|_2^2 = \| A\lambda^k + Bz^k \|_2^2 + \| Az^k + Bz^k \|_2^2 \leq 2 \| A\lambda^k + Bz^k \|_2^2 \leq 2M^2.
\]
Also notice
\[\|w^{k+1} - w^*\|_2^2 = \|w^k - \alpha_k g^k - w^*\|_2^2 \]
\[= \|w^k - w^*\|_2^2 + \sum_{j=1}^K \alpha_j \|g_j\|_2^2 - 2\alpha_k \langle g^k, w^k - w^* \rangle \]
\[\leq \|w^k - w^*\|_2^2 + \sum_{j=1}^K \alpha_j \|g_j\|_2^2 - 2\alpha_k (p^* - d(w^k) - \epsilon_p), \]
where we use the fact that \(g^k \in \partial_k (-d(w^k))\) in the last inequality. Re-arranging terms and summing over \(k = 1, \cdots, K\), we have
\[2 \left( \sum_{k=1}^K \alpha_k \right) \sum_{k=1}^K \alpha_k \]
\[\leq 2 \sum_{k=1}^K \alpha_k (p^* - d(\lambda^k, \rho^k) - \epsilon_k) \leq \|w^1 - w^*\|_2^2 + \sum_{k=1}^K \alpha_k \|g^k\|_2^2 \]
\[\leq \|w^1 - w^*\|_2^2 + \sum_{k=1}^K \alpha_k^2 \|Ax^k + Bz^k\|_1^2 = d_0^2 + \sum_{k=1}^K \tau_k^2, \]
which further implies
\[\min_{k \in [K]} p^* - d(\lambda^k, \rho^k) \leq \frac{d_0^2 + \sum_{k=1}^K \tau_k^2}{2} + \epsilon_p \leq \frac{M d_0^2 + \sum_{k=1}^K \tau_k^2}{\sqrt{2}} + \epsilon_p. \]

**Proof (Proof of Theorem 6)** The first claim follows from Lemma 3 and the choices of \(\tau_k\) and \(K\) so that:
\[\frac{M d_0^2 + \sum_{k=1}^K \tau_k^2}{\sqrt{2}} \leq \frac{\sqrt{2} M d_0}{\sqrt{\sum_{k=1}^K \tau_k}} + \epsilon_p \leq \epsilon_d. \]
Let \(\gamma_\tau = \alpha_k/\tau_k\). Recall the bound in (38), and we have \(\gamma_\tau \|g^k\|_2 \leq 1\); the second claim is then proved in [26, Theorem 7.4].

**A.2 Proof of Theorem 7**

**Proof (Proof of Theorem 7)** Firstly notice that according to the penalty update, we have
\[\rho^k = \rho^1 + \sum_{j=2}^K \rho^j - \rho^{j-1} = \rho^1 + \sum_{j=2}^K \max\{\|\lambda^j\|_\infty, \alpha_j \|Ax^j + Bz^j\|_1\} \]
\[\geq \rho^1 + \sum_{j=2}^K \alpha_j \|Ax^j + Bz^j\|_1 = \rho^1 + (k - 1)\tau. \]

For the purpose of contradiction, suppose \(\|Ax^k + Bz^k\|_1 \geq \rho_p\) for all \(k \in \mathbb{N}\), and thus Algorithm 3 will generate an unbounded sequence \(\{\rho^k\}_{k \in \mathbb{N}}\). Let \((\lambda^*, \rho^*)\) be an optimal solution to the dual problem (6). Then we have
\[\|\lambda^{k+1} - \lambda^*\|_2^2 = \|\lambda^k + \alpha_k (Ax^k + Bz^k) - \lambda^*\|_2^2 \]
\[= \|\lambda^k - \lambda^*\|_2^2 + \alpha_k^2 \|Ax^k + Bz^k\|_2^2 + 2\alpha_k \langle Ax^k + Bz^k, \lambda^k - \lambda^* \rangle \]
\[\leq \|\lambda^k - \lambda^*\|_2^2 + \alpha_k^2 \|Ax^k + Bz^k\|_2^2 + 2\alpha_k \left(d(\lambda^k, \rho^k) - \rho^* + \epsilon_p + \|Ax^k + Bz^k\|_1 (\rho^* - \rho^p)\right) \]
\[\leq \|\lambda^k - \lambda^*\|_2^2 + \sum_{j=1}^K \alpha_j \|Ax^k + Bz^k\|_2^2 + 2\alpha_k \left(\epsilon_p + \|Ax^k + Bz^k\|_1 (\rho^* - \rho^k)\right), \]
(39)
where the first inequality is due to (21), and the second inequality is due to \(d(\lambda^k, \rho^k) \leq \rho^*\). In view of the definition of \(\alpha_k\) and the fact that \(\|Ax^k + Bz^k\|_1 > \rho_p\), (39) further implies that
\[\|\lambda^{k+1} - \lambda^*\|_2^2 \leq \|\lambda^k - \lambda^*\|_2^2 + \tau^2 + 2r\rho^* - 2r \rho^k. \]
(40)
Notice that when \(\rho^k \geq \rho^* + \tau/2 + 1\), we have \(\|\lambda^{k+1} - \lambda^*\|_2 \leq \|\lambda^k - \lambda^*\|_2\), and thus the dual sequence \(\lambda^k\) stays bounded; now letting \(k \to \infty\) on (40), the left-hand side is nonnegative while the right-hand side goes to \(-\infty\), which is a desired contradiction. \(\square\)