An Exact Algorithm for Two-stage Robust Optimization with Mixed Integer Recourse Problems

Long Zhao and Bo Zeng

Department of Industrial and Management Systems Engineering
University of South Florida
Email: longzhao@mail.usf.edu, bzeng@usf.edu,

January, 2012

Abstract

In this paper, we consider a linear two-stage robust optimization model with a mixed integer recourse problem. Currently, this type of two-stage robust optimization model does not have any exact solution algorithm available. We first present a set of sufficient conditions under which the existence of an optimal solution is guaranteed. Then, we present a *nested column-and-constraint generation* algorithm that can derive an exact solution in finite steps. The algorithm development also yields novel solution methods to solve general mixed integer bi-level programs and some four-level programs. Finally, the proposed framework is demonstrated through an application of a robust rostering problem with part-time staff scheduling in the second stage.
Key words: two-stage robust optimization, mixed integer recourse problem, tri-level program, bi-level program

1 Introduction

Robust optimization (RO) is a recent optimization approach that deals with data uncertainty. Different from stochastic programming, another well-known and popular modeling method, RO does not assume probability distributions of random parameters. Also, rather than looking for an optimal solution with respect to the expected objective value as in stochastic programming, RO pursues a solution that guarantees the best performance in any of the worst cases (Ben-Tal and Nemirovski (1998, 1999, 2000), El Ghaoui et al. (1998), Bertsimas and Sim (2003, 2004)). Since the solution is expected to hedge against any possibility, a *single-stage* RO model tends to be over-conservative (Atamturk and Zhang 2007). To address this issue, especially in situations where some recourse decisions can be made after the uncertainty is revealed, *two-stage RO*, or robust *adjustable* or *adaptable* optimization, has been proposed and studied (Ben-Tal et al. 2004, Atamturk and Zhang 2007, Bertsimas et al. 2011a). Equation (1) presents the most popular form of two-stage RO, where vector $\mathbf{y}$ denotes the first-stage decision variables, $\mathbf{u}$ is a point in the uncertainty set $\mathbb{U}$, and vector $\mathbf{x}$ denotes the recourse decision variables in the second stage.

$$\min_{\mathbf{y}\in\mathbb{Y}} \max_{\mathbf{u}\in\mathbb{U}} \min_{\mathbf{x}\in\mathbb{F}(\mathbf{y},\mathbf{u})} \mathbf{h}(\mathbf{y},\mathbf{u},\mathbf{x}). \tag{1}$$

The uncertainty set, which is beyond the control of the decision maker, often takes the form of a mixed integer set, e.g., a finite discrete set or a polytope, or a nonlinear set, e.g., an ellipsoid.

In very recent years, two-stage RO has gained popularity in solving practical problems where the randomness is significant but its probability distribution is difficult to characterize, or implementing an infeasible solution is extremely costly. Those applications include network/transportation systems (Atamturk and Zhang 2007, Ordonez and Zhao 2007, Gabrel et al. 2011), portfolio optimization (Takeda et al. 2008), and power systems control and scheduling problems (Zhao and Zeng 2010, Jiang et al. 2011, Bertsimas et al. 2011b). However, different from stochastic programming, which is in a monolithic form, the two-stage RO is actually a tri-level optimization model that is very challenging to compute. Even a simple formulation with linear programming (LP) problems in both stages could be NP-hard (Ben-Tal et al. 2004). To solve this complex problem, Ben-Tal et al. (2004) assume that recourse decision variables take values according to some affine functions of the value of uncertainty. Then, a two-stage RO can be reduced to a formulation close to a single-stage RO model and could be solved by fast algorithms. This approximation strategy is called the *affine rule* method. Different from that approximation method, a master-subproblem framework, in two variants, recently has been developed and implemented to derive the solution of a few two-stage RO problems (Thiele et al. 2010, Zhao and Zeng 2010, Bertsimas et al. 2011b, Jiang et al. 2011, Gabrel et al. 2011). The first type of master-subproblem procedure is developed in a spirit similar to Benders decomposition, in which cutting planes supplied to the master problem are defined with the dual information of the recourse problem and revealed uncertainty information (Thiele et al. 2010, Zhao and Zeng 2010, Bertsimas et al. 2011b, Jiang et al. 2011, Gabrel et al. 2011). The second algorithm generates recourse variables and associated constraints for significant scenarios in the uncertainty set and supplies them to the master problem as cutting planes, which is a *column-and-constraint* generation procedure (Zhao and Zeng 2010, Zeng and Zhao 2011). It can be proven that when the recourse problem is LP, both algorithms converge to an optimal solution in finite iterations (Thiele et al. 2010, Zhao and Zeng 2010, Bertsimas et al. 2011b, Jiang et al. 2011, Gabrel et al. 2011, Zeng and Zhao 2011). Nevertheless, they perform very differently in the computational aspect. As shown in (Zhao and Zeng 2010, Zeng and Zhao 2011), for robust power system scheduling problems and robust location-transportation problems, the column-and-constraint generation algorithm performs an order of magnitude faster than an implementation of Benders type algorithms.

Although most existing research focuses on two-stage RO with LP recourse problems in the second stage, we observe that many real problems require discrete recourse decisions. Over the last 20 years, stochastic programming with integer or binary recourse variables has received tremendous research attention in both application and algorithmic aspects (Ahmed et al. 2004, Carøe and Tind 1998, Laporte and Louveaux 1993, Sen and Sherali 2006, Sen 2003). As another optimization approach to deal with uncertainty, it is worth developing efficient algorithms for two-stage RO with discrete recourse problems and applying them to solve practical issues. Nevertheless, compared with two-stage RO with LP recourse problems, solving two-stage RO models with mixed integer programming (MIP) recourse problems is more challenging. To the best of our knowledge, very limited research has been done to solve this type of problem. The only approach of which we are aware is the revised affine rule based approximation algorithms to deal with discrete recourse variables (Bertsimas et al. 2011a), which could derive reasonable and computationally-effective solutions in many applications. Note that discrete variables, especially binary variables, often are used to capture the complicated logic requirements, where it is difficult to identify some simple decision rules to determine variable values. If this is the case with the recourse problem, affine rule based approximation methods may not be able to produce good solutions. Also, given that strong duality does not hold in general MIP problems, it is difficult to extend current Benders-type algorithms to generate cutting planes (Thiele et al. 2010, Zhao and Zeng 2010, Bertsimas et al. 2011b, Jiang et al. 2011, Gabrel et al. 2011, Zeng and Zhao 2011).

As mentioned in (Zeng and Zhao 2011), the column-and-constraint generation algorithm, which simply creates cutting planes with primal decision variables, may provide an effective framework to solve those difficult problems. In fact, when the uncertainty set is a finite discrete set, it is clear that we can extend the column-and-constraint generation strategy to deal with MIP recourse problems. However, a few issues remain unsolved for general cases. Among them, the critical one is how to quickly identify the significant scenarios other than enumeration. In this paper, we address those issues by deriving structural insights and developing computation methods to exactly solve two-stage RO with an MIP recourse problem. We first provide a sufficient condition to guarantee the existence of an optimal solution. Then, we describe the application of a column-and-constraint generation algorithm to solve the mixed integer bi-level programs, which serves to identify significant scenarios. This algorithm can be embedded into an outer-level column-and-constraint generation scheme to obtain a complete algorithm for two-stage RO with a MIP recourse problem. We refer to it as *nested column-and-constraint generation algorithm.* Finally, we demonstrate the strength of our algorithm on a simple two-stage robust employee scheduling model. Our contribution includes: (i) the properties to ensure the existence of optimal solutions to two-stage RO with an MIP recourse problem; (ii) an algorithm to exactly solve this type of problems; (iii) a preliminary computational study showing the effectiveness of the proposed algorithm on a two-stage robust rostering problem. Our algorithm development also yields algorithms to solve general mixed integer bi-level program and some four-level programs.

The remainder of this paper is organized as follows. Two-stage robust optimization with an MIP recourse is analyzed in Section 2. The solution method is described in Section 3. An application of solving the two-stage rostering problem is presented in Section 4. Finally, Section 5 concludes the paper.

2 Two-stage Robust Optimization with MIP Recourse

In this section, the general form of linear two-stage RO formulation with an MIP recourse is analyzed, and some conditions under which an exact solution exists are identified. Considered first is the general two-stage robust model with an MIP recourse (**two-stage RO(MIP)**) as follows:

$$\inf_{\mathbf{y}\in\mathbb{Y}} \mathbf{cy} + \sup_{\mathbf{u}\in\mathbb{U}} \inf_{\mathbf{z},\mathbf{x}\in\mathbb{F}(\mathbf{y},\mathbf{u})} \mathbf{dx} + \mathbf{gz}, \tag{2}$$

where $\mathbb{Y} = \{\mathbf{y} \in \mathbb{R}^m_+ \times \mathbb{Z}^{m'}_+ : \mathbf{Ay} \geq \mathbf{b}\}$, $\mathbb{F}(\mathbf{y},\mathbf{u}) = \{(\mathbf{z},\mathbf{x}) \in \mathbb{Z}^n_+ \times \mathbb{R}^p_+ : \mathbf{Ex} + \mathbf{Gz} \geq \mathbf{f} - \mathbf{Ru} - \mathbf{Dy}, \mathbf{Tz} \geq \mathbf{v}\}$, and the uncertainty set $\mathbb{U}$ is a bounded mixed integer set in the form of $\mathbb{U} = \{\mathbf{u} \in \mathbb{Z}^q_+ \times \mathbb{R}^{q'}_+ : \mathbf{Hu} \leq \mathbf{a}\}$, with all coefficients/parameters rational in those expressions. Without loss of generality, one can assume that $Proj_{\mathbf{z}}\mathbb{F}(\mathbf{y},\mathbf{u}) \subseteq \Phi = \{\mathbf{z} \in \mathbb{Z}^n_+ : \mathbf{Tz} \geq \mathbf{v}\}$ for any $\mathbf{y} \in \mathbb{Y}$ and $\mathbf{u} \in \mathbb{U}$. Note in (2) that the first-stage decision $\mathbf{y}$ takes into account all possible future uncertain situations, which are captured by $\mathbb{U}$ without probability information, and the recourse decision variables $\mathbf{z}$ and $\mathbf{x}$ represent the recourse actions that can be implemented after the first-stage decision is made and the uncertainty is revealed. Clearly, such a framework allows us to model a sequential decision-making environment where (discrete) recourse actions can be made in a wait-and-see fashion.

Because the recourse problem is a linear MIP problem, it can be treated simply as a regular minimization problem, i.e., the infimum can be replaced by its minimum. We are now interested in conditions under which (2) is equivalent to

$$\min_{\mathbf{y}\in\mathbb{Y}} \mathbf{cy} + \max_{\mathbf{u}\in\mathbb{U}} \min_{\mathbf{z},\mathbf{x}\in\mathbb{F}(\mathbf{y},\mathbf{u})} \mathbf{dx} + \mathbf{gz}. \tag{3}$$

It is obvious that the reduction from (2) to (3) is valid if $\mathbb{U}$ is a finite discrete set. In fact, if that is the case, the tri-level formulation in (2) can be further reformulated into a monolithic mixed integer optimization problem. Assume that $\mathbb{U} = \{\mathbf{u}^i\}_{i=1}^I$. The following formulation is equivalent to (2) and (3):

$$\begin{aligned}
\textbf{MIP}-\textbf{Equivalent}: \quad & \min\ \mathbf{cy} + \eta & (4)\\
& st.\ \mathbf{Ay} \geq \mathbf{b} & (5)\\
& \eta \geq \mathbf{dx}^i + \mathbf{gz}^i,\ i = 1,\ldots,I & (6)\\
& \mathbf{Ex}^i + \mathbf{Gz}^i \geq \mathbf{f} - \mathbf{Ru}^i - \mathbf{Dy}, \quad \mathbf{Tz}^i \geq \mathbf{v},\ i = 1,\ldots,I & (7)\\
& \mathbf{y} \in \mathbb{R}^m_+ \times \mathbb{Z}^{m'}_+, \mathbf{z}^i \in \mathbb{Z}^n_+, \mathbf{x}^i \in \mathbb{R}^p_+, i = 1,\ldots,I. & (8)
\end{aligned}$$

Nevertheless, such a reduction may not be feasible for a general multi-level formulation. For example, consider the following bi-level problem:

$$\sup_{0\leq d\leq 1} \min_{x_1,x_2} \{x_1 + x_2 : x_1 \geq d, x_2 \geq 1 - d, x_1 \in \mathbb{Z}_+, x_2 \geq 0\}.$$

If $d = 0$, solving the above bi-level formulation will yield $x_1 = 0$, $x_2 = 1$, and $x_1 + x_2 = 1$. When $d = \delta \in (0, 1]$, we have $x_1 = 1$, $x_2 = 1 - \delta$ and $x_1 + x_2 = 2 - \delta$. Clearly, no optimal d can be derived to achieve the supremum, which is 2. Therefore, this bi-level formulation cannot be reduced to a $\max - \min$ problem.

Although the aforementioned reduction is not valid in general for multi-level programs, it can be proven that (3) is equivalent to (2) under some mild conditions. Next, we present such a sufficient condition adapted from (Hoang 1998, Takeda et al. 2008).

Proposition 1. *Assume that the uncertainty set* $\mathbb{U}$ *is a polytope, i.e,* $q = 0$*. If* $\theta(\mathbf{u}) := \min_{\mathbf{z},\mathbf{x}\in\mathbb{F}(\hat{\mathbf{y}},\mathbf{u})} \mathbf{dx} + \mathbf{gz}$ *is a quasiconvex function over* $\mathbb{U}$ *for any given* $\hat{\mathbf{y}}$*, then (2) reduces to (3). Furthermore, both of them are equivalent to*

$$\min_{\mathbf{y}\in\mathbb{Y}} \mathbf{cy} + \max_{\mathbf{u}\in\mathbf{U}} \min_{\mathbf{z},\mathbf{x}\in\mathbb{F}(\mathbf{y},\mathbf{u})} \mathbf{dx} + \mathbf{gz}, \tag{9}$$

where $\mathbf{U} = \{\mathbf{u}^j\}_{j=1}^J$ *is the set of all extreme points of* $\mathbb{U}$*.* □

It is easy to generalize this result to the case where $\mathbb{U}$ is a bounded mixed integer set. Under this situation, note that $\mathbb{U}$ is the union of a collection of polytopes, i.e. $\mathbb{U} = \mathbb{U}^1 \cup \ldots \mathbb{U}^K$, and each of them corresponds to a particular assignment to the set of integer variables.

Corollary 1. *If* $\theta(\mathbf{u})$ *is quasiconvex over* $\mathbb{U}^k, k = 1, ..., K$ *for any given* $\hat{\mathbf{y}}$*, then* (2) *reduces to* (3)*. Furthermore, both of them are equivalent to*

$$\min_{\mathbf{y}\in\mathbb{Y}} \mathbf{cy} + \max_{\mathbf{u}\in\hat{\mathbf{U}}} \min_{\mathbf{z},\mathbf{x}\in\mathbb{F}(\mathbf{y},\mathbf{u})} \mathbf{dx} + \mathbf{gz}, \tag{10}$$

where $\hat{\mathbf{U}} = \{\mathbf{U}^k\}_{k=1}^K$ *and* $\mathbf{U}^k$ *is the set of all extreme points of* $\mathbb{U}^k$*.* □

We mention that the quasiconvex requirement is not restrictive because a broad range of functions is quasiconvex. The next corollary lists a few typical functions that are quasiconvex.

Corollary 2. $\theta(\mathbf{u})$ *is quasiconvex if it is (i) a convex function; or (ii) a non-increasing function; or (iii) a non-decreasing function.* □

To the best of our knowledge, in all existing applications of two-stage RO, including (Thiele et al. 2010, Zhao and Zeng 2010, Bertsimas et al. 2011b, Jiang et al. 2011, Gabrel et al. 2011), $\theta(\mathbf{u})$ is either non-decreasing or non-increasing in $\mathbf{u}$ because $\mathbf{u}$ represents random factor(s) that lead to either a positive or negative impact to the decision maker. So, unless explicitly mentioned, we always assume that $\theta(\mathbf{u})$ is a quasiconvex function in the remainder of this paper. Note that when the recourse problem is an LP problem, it is not necessary to carry this assumption, as the equivalence among (2), (3), and (10) is always valid (Zeng and Zhao 2011).

It is worth pointing out that the equivalence among (2), (3), and (10) is non-trivial, as (10) is a computationally convenient formulation. Similar to the case where the uncertainty set is a finite discrete set, we can develop the corresponding MIP-Equivalent reformulation in the form of (4-8) to solve (10). However, enumerating all possible scenarios (or the extreme points in $\hat{\mathbf{U}}$) and listing all corresponding variables and constraints is not realistic. In fact, we would like to highlight that the constraints in (6) indicate that not all scenarios (and their variables and constraints) in $\hat{\mathbf{U}}$ are necessary in defining the optimal solution and the worst-case objective value. That is, probably only a small subset of $\hat{\mathbf{U}}$ and the associated variables and constraints play a significant role. Such an observation motivates the development of the column-and-constraint generation algorithm (Zeng and Zhao 2011, Zhao and Zeng 2010), a solution procedure that creates extra recourse variables and related constraints for significant uncertainty scenarios that are generated on the fly. This procedure has been successfully applied to solve a few two-stage RO problems with LP recourse problems (Zeng and Zhao 2011, Zhao and Zeng 2010) and has exhibited a superior computational performance over Benders-type algorithms. In the next section, we first briefly review the column-and-constraint generation algorithm in its basic form and then present its extension to solve two-stage RO with an MIP recourse problem.

3 Solving two-stage Robust Problem with an MIP Recourse Problem

As mentioned, the column-and-constraint generation procedure creates recourse decision variables (and related constraints) for dynamically-identified significant uncertainty scenarios. Because those variables and the corresponding constraints are primal to the decision maker, no dual information is necessary in generating those variables and constraints, which is different from Benders cutting plane solution procedures. Hence, we believe that it could evolve into a more capable algorithm to solve two-stage RO with an MIP in the second stage, for which no exact algorithm has been reported.

We first briefly review the column-and-constraint generation algorithm in its basic form, along with some results on its complexity. Then, we study an extension of the basic form, the nested column-and-constraint generation algorithm, to solve two-stage RO with an MIP recourse problem. Our solution procedure also provides a novel approach to solve the challenging bi-level max − min problem with discrete variables in the second level.

Remark 1. *An optimal solution to the problem in* (3) *is just an optimal* $\mathbf{y}$ *that performs best, with the help of recourse actions, in any worst situations. The actual recourse decisions will be made after* $\mathbf{u}$ *is revealed. To engage in a meaningful discussion, we assume that there exists an optimal* $\mathbf{y} \in \mathbb{Y}$ *to problem (3) with a bounded objective value.*

3.1 A Review of the Column-and-Constraint Generation Algorithm

The column-and-constraint generation procedure is implemented in a two-level master-subproblem framework. We first assume that an oracle is available to solve the following bi-level max − min problem, i.e., the subproblem in this procedure:

$$\begin{aligned} \mathbf{SP}: \quad & \mathcal{Q}(\hat{\mathbf{y}}) = \max_{\mathbf{u}\in\mathbb{U}} \min_{\mathbf{z},\mathbf{x}} \mathbf{dx} + \mathbf{gz} \\ \text{s.t.} \quad & \mathbf{Ex} + \mathbf{Gz} \geq \mathbf{f} - \mathbf{Ru} - \mathbf{D}\hat{\mathbf{y}} \\ & \mathbf{Tz} \geq \mathbf{v} \\ & \mathbf{z} \in \mathbb{Z}^n_+, \mathbf{x} \in \mathbf{R}^p_+. \end{aligned} \tag{11}$$

This oracle can either derive an optimal scenario $\mathbf{u}^* \in \mathbb{U}$ with a finite optimal value or identify some $\mathbf{u}^*$ for which the recourse problem is infeasible, i.e., $\mathcal{Q}(\hat{\mathbf{y}}) = +\infty$ by convention. Note that any feasible solution to (3) provides an upper bound. Also, solving a relaxation of its monolithic equivalent form, i.e., the formulation in the form of **MIP−Equivalent** with respect to a subset of $\hat{\mathbf{U}}$, yields a lower bound. By making use of upper and lower bounds, we can develop an exact algorithm to solve (3). Let LB and UB be the lower and upper bounds, respectively, and ϵ be the optimality tolerance. We have

Column-and-Constraint Generation Algorithm

---

1. Set $LB = -\infty$, $UB = +\infty$ and $k = 0$.
2. Solve the following master problem.

$$\begin{aligned} \mathbf{MP}: \min_{\mathbf{y},\eta,\mathbf{x}} \quad & \mathbf{cy} + \eta \\ \text{s.t.} \quad & \mathbf{Ay} \geq \mathbf{b} \\ & \eta \geq \mathbf{dx}^l + \mathbf{gz}^l, \ \forall 1 \leq l \leq k \\ & \mathbf{Dy} + \mathbf{Ex}^l + \mathbf{Gz}^l \geq \mathbf{f} - \mathbf{Ru}^*_l, \ \forall 1 \leq l \leq k \\ & \mathbf{Tz}^l \geq \mathbf{v}, \ \forall 1 \leq l \leq k \\ & \mathbf{y} \in \mathbb{Y}, \ \eta \in \mathbb{R}, \ \mathbf{z}^l \in \mathbb{Z}^n_+, \mathbf{x}^l \in \mathbb{R}^p_+, \ \forall 1 \leq l \leq k \end{aligned} \tag{12}$$

   Derive an optimal solution $(\mathbf{y}^*_{k+1}, \eta^*_{k+1}, \mathbf{z}^{1^*}, \ldots, \mathbf{z}^{k*}, \mathbf{x}^{1*}, \ldots, \mathbf{x}^{k*})$ and update $LB = \mathbf{cy}^*_{k+1} + \eta^*_{k+1}$. If $UB - LB \leq \epsilon$, return $\mathbf{y}^*_{k+1}$ and terminate.
3. Call the oracle to solve subproblem $\mathcal{Q}(\mathbf{y}^*_{k+1})$ in (11) and update

$$UB = \min\{UB, \mathbf{cy}^*_{k+1} + \mathcal{Q}(\mathbf{y}^*_{k+1})\}.$$

   If $UB - LB \leq \epsilon$, return $\mathbf{y}^*_{k+1}$ and terminate.
4. *Create* variables $(\mathbf{z}^{k+1}, \mathbf{x}^{k+1})$ and *add* the following constraints:

$$\begin{aligned} & \eta \geq \mathbf{dx}^{k+1} + \mathbf{gz}^{k+1} \\ & \mathbf{Dy} + \mathbf{Ex}^{k+1} + \mathbf{Gz}^{k+1} \geq \mathbf{f} - \mathbf{Ru}^*_{k+1}, \\ & \mathbf{Tz}^{k+1} \geq \mathbf{v} \\ & \mathbf{z}^{k+1} \in \mathbb{Z}^n_+, \mathbf{x}^{k+1} \in \mathbb{R}^p_+ \end{aligned}$$

   to **MP** where u^*_{k+1} is the scenario solving $\mathcal{Q}(\mathbf{y}^*_{k+1})$. Update $k = k + 1$ and go to Step 2. □

Next, we present a result that is adapted from (Zeng and Zhao 2011) on the finite convergence of this algorithm. This result holds in two-stage models with both LP recourse and MIP recourse problems, as it is up to only the description of the uncertainty set.

Proposition 2. *(Zeng and Zhao 2011) Given that an oracle can find a worst-case* $\mathbf{u}^*$ *for any given* $\mathbf{y}^*$*, the column-and-constraint generation algorithm converges an optimal solution in a finite number of iterations. The number of iterations is bounded by* $|\hat{\mathbf{U}}|$. □

The above column-and-constraint generation algorithm dynamically converts a two-stage RO problem into a monolithic MIP formulation, **MP** in (12), which actually is a partial enumeration of the **MIP − Equivalent** form. In fact, **MP** has a structure that is very close to the deterministic equivalent form of two-stage stochastic programming. Hence, it probably builds a connection to stochastic programming and, therefore, existing computational techniques for two-stage stochastic programming could be useful to solve **MP** efficiently.

Then comes the critical step for successfully implementing the column-and-constraint generation procedure: solving **SP** and identifying a significant scenario $\mathbf{u}^*$. As **SP** is a bi-level program, general existing algorithms to bi-level programs can be adopted to solve it. Nevertheless, the majority of existing study on bi-level programs focuses on formulations with an LP as the lower-level problem. A very limited study is available on solving those with an MIP in the lower level (Shimizu et al. 1997). Such a situation may be explained by the fact that the strong duality does not hold, and less structural information can be used in algorithm development. As a result, this type of problem remains a challenging one in both theoretical and computational aspects. However, to solve a two-stage RO with an MIP recourse problem, it is necessary to incorporate an efficient algorithm to solve **SP** within the column-and-constraint generation procedure. Towards this end, we propose to solve the bi-level **SP** with a MIP recourse problem through its tri-level structure, which allows us to make use of the idea behind the column-and-constraint generation method.

3.2 Solving Bi-level SP by Its Tri-level Form

Different from the mainstream idea that directly reduces a bi-level program into a monolithic optimization problem, our strategy is to first expand it into a tri-level problem in a structure similar to two-stage RO. Then, by using the column-and-constraint generation approach, we can dynamically convert the tri-level problem into an (equivalent) monolithic form. The feasible set of discrete recourse variables can be *considered* as the uncertainty set, while the optimal solution we are looking for is in terms of $\mathbb{U}$.

To focus on the main results and algorithm developments in this paper, we make a few very mild assumptions. First, we assume $p \geq 1$, i.e., the MIP recourse problem has at least one continuous recourse variable. Second, we assume that the LP problem, obtained by fixing $\mathbf{y}$, $\mathbf{u}$ and $\mathbf{z}$ to their any possible values, is always feasible and bounded, which is referred to as the *extended relatively complete recourse* property. Finally, we assume that the feasible set of discrete recourse variables, i.e., Φ, is bounded, which is not restrictive for practical problems.

Observe that the formulation of **SP** defined in (11) is equivalent to the following tri-level formulation obtained by separating discrete variables from continuous variables:

$$\mathcal{Q}(\hat{\mathbf{y}}) = \max_{\mathbf{u}\in\mathbb{U}} \quad \min_{\mathbf{z}\in\Phi} \mathbf{g}\mathbf{z} + \quad \min_{\mathbf{x}} \mathbf{d}\mathbf{x} \quad st.\ \mathbf{E}\mathbf{x} \geq \mathbf{f} - \mathbf{R}\mathbf{u} - \mathbf{D}\hat{\mathbf{y}} - \mathbf{G}\mathbf{z}, \quad \mathbf{x} \in \mathbb{R}^p_+ \tag{13}$$

Denote $\Phi = \{\mathbf{z}^r\}_{r=1}^{\mathcal{R}}$. By making use of the countability of Φ, we have

$$\begin{aligned} \mathbf{Bi/Tri{-}Equivalent\ I}: \mathcal{Q}(\hat{\mathbf{y}}) = &\max\ \theta \\ &\theta \leq \mathbf{g}\mathbf{z}^r + \min\{\mathbf{d}\mathbf{x}^r : \mathbf{E}\mathbf{x}^r \geq f - Ru - D\hat{y} - Gz^r, \mathbf{x} \in \mathbb{R}^p_+\} \\ &r = 1, \ldots, \mathcal{R} \\ &\mathbf{u} \in \mathbb{U}. \end{aligned} \tag{14}$$

In the above formulation, $\mathbf{x}^r$ are the corresponding decision variables for $\mathbf{z}^r$, a particular assignment of $\mathbf{z}$. Clearly, the whole **Bi/Tri−Equivalent I** shares a great similarity to **MIP−Equivalent**, which motivates us to investigate the idea of column-and-constraint method.

We first present the derivation of its monolithic formulation. Note that the minimization problem in each constraint is an LP model. If the extended relatively complete recourse assumption holds, we can apply the classical Karush-Kuhn-Tucker (KKT) condition to convert this minimization problem into a feasibility problem. Specifically, let π^r be the dual variables of the minimization problem in the r^{th} constraint of **Bi/Tri−Equivalent I** with a compatible dimension p'. Then, deriving an optimal $\mathbf{x}^{r*}$ of that minimization problem is equivalent to obtaining a feasible solution $(\mathbf{x}^r, \pi^r)$ that satisfies the following constraints:

$$\mathbf{E}\mathbf{x}^r \geq \mathbf{f} - \mathbf{R}\mathbf{u} - \mathbf{D}\hat{\mathbf{y}} - \mathbf{G}\mathbf{z}^r \tag{15}$$

$$\mathbf{E}^t \pi^r \leq \mathbf{d}^t \tag{16}$$

$$\mathbf{x}^r(\mathbf{d}^t - \mathbf{E}^t \pi^r) = 0 \tag{17}$$

$$\pi^r(\mathbf{E}\mathbf{x}^r - \mathbf{f} + \mathbf{R}\mathbf{u} + \mathbf{D}\hat{\mathbf{y}} + \mathbf{G}\mathbf{z}^r) = 0 \tag{18}$$

$$\mathbf{x}^r \in \mathbb{R}^p_+,\ \pi^r \in \mathbb{R}^{p'}_+. \tag{19}$$

The last two constraints are complementary slackness conditions, which ensure that two feasible solutions are optimal to primal and dual problems, respectively. In fact, those complementary constraints can be linearized by introducing binary variables and making use of *big-M*, i.e. a sufficiently large number. For example, a constraint in (17) can be reformulated as

$$\mathbf{x}^r_j \leq M\delta^r_j,\ (\mathbf{d}^t - \mathbf{E}^t\pi^r)_j \leq M(1-\delta^r_j),\ \delta^r_j \in \{0,1\},\ j = 1, \ldots, p.$$

Hence, the above feasibility problem can be converted into a binary MIP formulation.

Consequently, using constraints (15-19) to identify feasible (optimal) solutions, we now can reformulate **Bi/Tri−Equivalent I** as a monolithic model. To simplify our exposition, constraints of nonlinear complementary slackness conditions are kept in the model while the *big-M* method can be called to linearize them.

Proposition 3. *The bi-level program,* **SP** *defined in* (11)*, is equivalent to the following formulation:*

$$\begin{aligned}
\mathcal{Q}(\hat{\mathbf{y}}) = \max\ & \theta \\
\theta \leq\ & \mathbf{g}\mathbf{z}^r + \mathbf{d}\mathbf{x}^r,\ r = 1, \ldots, \mathcal{R} \\
\mathbf{E}\mathbf{x}^r \geq\ & \mathbf{f} - \mathbf{R}\mathbf{u} - \mathbf{D}\hat{\mathbf{y}} - \mathbf{G}\mathbf{z}^r,\ r = 1, \ldots, \mathcal{R} \\
\mathbf{E}^t\pi^r \leq\ & \mathbf{d}^t,\ r = 1, \ldots, \mathcal{R} \\
\mathbf{x}^r(\mathbf{d}^t - \mathbf{E}^t\pi^r) =\ & 0,\ r = 1, \ldots, \mathcal{R} \\
\pi^r(\mathbf{E}\mathbf{x}^r - \mathbf{f} + \mathbf{R}\mathbf{u} + \mathbf{D}\hat{\mathbf{y}} + \mathbf{G}\mathbf{z}^r) =\ & 0,\ r = 1, \ldots, \mathcal{R} \\
\mathbf{u} \in \mathbb{U},\ \mathbf{x}^r \in\ & \mathbb{R}^p_+,\ \pi^r \in \mathbb{R}^{p'}_+,\ r = 1, \ldots, \mathcal{R}.
\end{aligned} \tag{20}$$

□

Using an argument similar to that for the column-and-constraint generation algorithm, we anticipate that it is not necessary to enumerate variables and constraints for all the possible $\mathbf{z}^r$'s and to construct the complete equivalent model in (20) to solve the bi/tri-level **SP**. Probably only a small number of $\mathbf{z}^r$'s and their associated $(\mathbf{x}^r, \pi^r)$ and constraints are sufficient. So, we propose to re-apply the column-and-constraint generation strategy to solve **SP**. To distinguish from the previously-described column-and-constraint generation procedure, we denote the one described following as the *inner-level column-and-constraint generation procedure* and denote the previous one as the *outer-level column-and-constraint generation procedure.*

The Inner-Level Column-and-Constraint Generation Algorithm

---

1. Set $LB = -\infty$, $UB = +\infty$ and $k = 0$.

2. Solve the following master problem (in its linearized form) of **SP**

$$\begin{aligned}
\mathbf{MP_S}:\ \mathcal{Q}(\hat{y}) &= \max\ \theta \\
\theta &\leq \mathbf{g}\mathbf{z}^r + \mathbf{d}\mathbf{x}^r,\ \forall 1 \leq r \leq k \\
\mathbf{E}\mathbf{x}^r &\geq \mathbf{f} - \mathbf{R}\mathbf{u} - \mathbf{D}\hat{\mathbf{y}} - \mathbf{G}\mathbf{z}^r,\ \forall 1 \leq r \leq k \\
\mathbf{E}^t\pi^r &\leq \mathbf{d}^t,\ \forall 1 \leq r \leq k \\
\mathbf{x}^r(\mathbf{d}^t - \mathbf{E}^t\pi^r) &= 0,\ \forall 1 \leq r \leq k \\
\pi^r(\mathbf{E}\mathbf{x}^r - \mathbf{f} + \mathbf{R}\mathbf{u} + \mathbf{D}\hat{\mathbf{y}} + \mathbf{G}\mathbf{z}^r) &= 0,\ \forall 1 \leq r \leq k \\
\mathbf{u} \in \mathbb{U},\ \mathbf{x}^r \in \mathbb{R}^p_+,\ \pi^r &\in \mathbb{R}^{p'}_+,\ \forall 1 \leq r \leq k.
\end{aligned} \tag{21}$$

Obtain an optimal $\mathbf{u}$, denoted it by $\mathbf{u}^*_{k+1}$, and update $UB = \mathcal{Q}(\hat{y})$. If $UB - LB \leq \epsilon$, return $\mathbf{u}^*_{k+1}$ and terminate.

3. Solve the subproblem of **SP** $\quad \mathbf{SP_S}: \min_{\mathbf{z},\mathbf{x}\in\mathbb{F}(\hat{\mathbf{y}},\mathbf{u}^*_{k+1})} \mathbf{d}\mathbf{x} + \mathbf{g}\mathbf{z}.$

Obtain an optimal solution $(\mathbf{z}^{k+1*}, \mathbf{x}^{k+1*})$ and update $LB = \max\{LB, \mathbf{g}\mathbf{z}^{k+1*} + \mathbf{d}\mathbf{x}^{k+1*}\}$. If $UB - LB \leq \epsilon$, return $\mathbf{u}^*_{k+1}$ and terminate.

4. *Create* variables $(\mathbf{x}^{k+1}, \pi^{k+1})$ and *add* the following constraints

$$\begin{aligned}
\mathbf{E}\mathbf{x}^{k+1} &\geq \mathbf{f} - \mathbf{R}\mathbf{u} - \mathbf{D}\hat{\mathbf{y}} - \mathbf{G}\mathbf{z}^{k+1} \\
\mathbf{E}^t\pi^{k+1} &\leq \mathbf{d}^t \\
\mathbf{x}^{k+1}(\mathbf{d}^t - \mathbf{E}^t\pi^{k+1}) &= 0 \\
\pi^{k+1}(\mathbf{E}\mathbf{x}^{k+1} - \mathbf{f} + \mathbf{R}\mathbf{u} + \mathbf{D}\hat{\mathbf{y}} + \mathbf{G}\mathbf{z}^{k+1}) &= 0 \\
\mathbf{x}^{k+1} \in \mathbb{R}^p_+,\ \pi^{k+1} &\in \mathbb{R}^{p'}_+
\end{aligned}$$

to $\mathbf{MP_S}$. Update $k = k + 1$ and go to Step 2. □

Next, we show that the inner-level column-and-constraint procedure converges finitely.

Proposition 4. *For any given $\hat{\mathbf{y}} \in \mathbb{Y}$, the inner-level column-and-constraint generation procedure is finitely convergent, and the number of iterations is bounded by $\mathcal{R}$.*

Proof. Claim: Any repeated z^* in this procedure implies the optimality, i.e., $LB = UB$. Then, the proposition follows immediately due to the fact that Φ is a finite set.

Suppose at iteration k a worst-case uncertainty $\mathbf{u}^*$ is obtained by solving $\mathbf{MP_S}$, and it leads to optimal $\mathbf{z}^*$ and $\mathbf{x}^*$ to $\mathbf{SP_S}$. It follows that $UB \geq LB \geq \mathbf{g}\mathbf{z}^* + \mathbf{d}\mathbf{x}^*$. If $\mathbf{z}^*$ has been identified in a previous iteration k' with $1 \leq k' \leq k - 1$, then $\mathbf{u}^*$ will be an optimal solution to $\mathbf{MP_S}$ at iteration $k + 1$ as $\mathbf{MP_S}$ does not change from iteration k to $k + 1$. Given that $\mathbf{x}^*$ is optimal when $\mathbf{u} = \mathbf{u}^*$ and $\mathbf{z} = \mathbf{z}^*$, it follows from the first constraint in (20) (also the first constraint in (14)) that $UB = \mathcal{Q}(\hat{y}) \leq \mathbf{g}\mathbf{z}^* + \mathbf{d}\mathbf{x}^*$ in iteration $k + 1$. Therefore, we have $LB = UB$. □

We mention that the strong duality property of a linear program can also be used to build an alternative tri-level formulation of **SP**, which allows us to develop another monolithic equivalent form and a column-and-constraint generation variant. Specifically, by dualizing the innermost minimization problem in (13), **SP** is equivalent to the following tri-level problem:

$$\max_{\mathbf{u}\in\mathbb{U}} \min_{\mathbf{z}\in\Phi} \mathbf{g}\mathbf{z} + \max_{\pi\in\Pi} (\mathbf{f} - \mathbf{R}\mathbf{u} - \mathbf{D}\mathbf{y} - \mathbf{G}\mathbf{z})^t\pi, \tag{22}$$

where $\Pi = \{\pi \in \mathbb{R}^{p'}_+ : \mathbf{E}^t\pi \leq \mathbf{d}^t\}$. By enumerating all possible assignments of $\mathbf{z}$, we can obtain its monolithic equivalent form as follows:

$$\begin{aligned}
\mathbf{Bi/Tri{-}Equivalent\ II}:\ \mathcal{Q}(\hat{\mathbf{y}}) &= \max\ \theta \\
\theta &\leq \mathbf{g}\mathbf{z}^r + (\mathbf{f} - \mathbf{R}\mathbf{u} - \mathbf{D}\hat{\mathbf{y}} - \mathbf{G}\mathbf{z}^r)^t\pi^r, \qquad r = 1, \ldots, \mathcal{R} \\
\mathbf{E}^t\pi^r &\leq \mathbf{d}^t, \qquad r = 1, \ldots, \mathcal{R} \\
\mathbf{u} \in \mathbb{U},\ \pi^r &\geq 0, r = 1, \ldots, \mathcal{R}.
\end{aligned} \tag{23}$$

Clearly, a column-and-constraint generation procedure, which generates only dual variables π and constraints in (23), can be used to solve the tri-level formulation in (22). Nevertheless, as the first constraints in (23) are quadratic constraints, an efficient algorithm for a quadratically-constrained quadratic program (QCQP) is necessary. However, when $\mathbb{U}$ can be represented as a binary set, such as cardinality constrained uncertainty set (Bertsimas and Sim 2003), constraints in (23) can be linearized by *big-M* method. Then, an off-the-shelf MIP solver is sufficient to support this column-and-constraint generation procedure. As can easily be developed in the same spirit as the aforementioned, details of this variant are omitted in this paper.

Remark 2. *(i) If the* extended relatively complete recourse *assumption does not hold, the KKT condition based reformulation in (20) is not valid, and the corresponding column-and-constraint generation algorithm is not applicable. However, the strong duality based reformulation in* (23) *is valid. Note that* Π *is independent of* $\mathbf{y}, \mathbf{u}$*, and* $\mathbf{z}$*. Given that two-stage RO (MIP) has a feasible solution (in* $\mathbf{y}$*), it follows that* Π *is not empty (otherwise, it contradicts the existence of a feasible solution). Therefore, the corresponding column-and-constraint generation variant is applicable.*

(ii) If for any fixed $\mathbf{u}$ *and* $\mathbf{z}$*, the remaining linear programming problem has a unique solution, both KKT condition and strong duality based column-and-constraint generation procedures perform the same number of iterations.*

(iii) To the best of our knowledge, the developed column-and-constraint generation procedure(s) for **SP** *is a novel algorithm to solve general mixed integer bi-level programs. It has been successfully applied to solve a power grid interdiction problem with line switching decisions in the lower level, for which it significantly outperforms existing (heuristic) procedures (Zhao and Zeng 2011).*

3.3 The Nested Column-and-constraint Generation Algorithm

With the aforementioned outer and inner level column-and-constraint generation algorithms, we then integrate them into a unified solution procedure to solve two-stage RO with an MIP recourse problem. As the column-and-constraint generation method is implemented in both the outer and inner levels, i.e., in a nested fashion, we call the whole procedure the *nested column-and-constraint generation algorithm* to solve two-stage RO with an MIP recourse problem.

Corollary 3. *The nested column-and-constraint generation algorithm converges to an optimal solution of two-stage RO (MIP) with a finite number of column-and-constraint generation iterations.* □

Remark 3. *The nested column-and-constraint generation algorithm actually is also a solution procedure for the four-level* $\min - \max - \min - \max$ *program if the last maximization problem is a linear program. Using KKT condition, the last* $\min - \max$ *pair can be converted into a mixed integer minimization program. Hence, we obtain an optimization problem in the* $\min - \max - \min$ *form, which renders itself suitable for our nested column-and-constraint generation algorithm where the inner level one can be directly applied to solve the* $\max - \min - \max$ *part. We believe that implementing the column-and-constraint generation method in a nested fashion could be a solution strategy to solve a broad range of multi-level programs.*

4 A Numerical Study: Two-Stage Robust Rostering Problem

In this section, the solution capability of the nested column-and-constraint generation algorithm is demonstrated. Rather than focusing on developing a multifaceted and detailed model with many practical factors of real systems and providing comprehensive analysis, our intention is to describe the implementation procedure of this algorithm on a simple model and present its computational behavior. With this motivation, the study was performed on a simplified personnel scheduling (rostering) problem with uncertain demands. We first built a basic deterministic rostering model and present its two-stage robust optimization counterpart. Then, we describe specific forms of generated variables and constraints, using KKT condition and strong duality, respectively. Finally, we present the computational results on two-stage robust rostering model with two different uncertainty sets, i.e., a cardinality set and a polytope.

4.1 Two-stage Robust Rostering Problem

In the rostering problem, an organization such as a call center or a clinic needs to allocate its staff members to shifts to meet service demands with a minimized operation cost and satisfy governmental or industrial regulations/restrictions (Ernst et al. 2004). Sometimes, to deal with demand surges, overtime from its regular staff or part-time staff (or agency staff) will be called. In this paper, we consider the rostering problem in the latter situation where part-time staff will be used to deal with service demand fluctuation.

First, we formulated the rostering model in a deterministic environment such that the service demand d_t in shift t $(= 0, \dots, T-1)$ is known. Let i $(= 1, \dots, I)$ be the index of regular staff who work N hours, i.e. the length of the whole shift, and j $(= 1, \dots, J)$ be the index of part-time staff whose working hours are to be determined. The cost parameters are: in shift t, c_{it} is the wage cost of regular staff i, f_{jt} is the fixed cost and h_{jt} is hourly rate of part-time staff j, and M_t is the penalty cost for the unserved demand which will be lost.

$$\textbf{Rostering}_{MIP}: \ \min \sum_i \sum_t c_{it} x_{it} + \sum_j \sum_t (f_{jt} y_{jt} + h_{jt} z_{jt}) + \sum_t M_t w_t \tag{24}$$

$$\text{s.t. } x_{it} + x_{i,t+1} + x_{i,t+2} \leq 2, \ \forall i, \ t \leq T-3, \tag{25}$$

$$l_i \leq \sum_t x_{it} \leq u_i, \ \forall i \tag{26}$$

$$y_{jt} + y_{j,t+1} \leq 1, \ \forall j, \ t \leq T-2 \tag{27}$$

$$a_j \leq \sum_t y_{jt} \leq b_j, \ \forall j \tag{28}$$

$$z_{jt} \leq N y_{jt}, \ \forall j, t \tag{29}$$

$$N \sum_i x_{it} + \sum_j z_{jt} + w_t \geq d_t, \ \forall t \tag{30}$$

$$x_{it} \in \{0,1\}, \ \forall i, t; \ y_{jt} \in \{0,1\}, \ \forall j, t; \ z_{jt} \in \mathbb{R}_+, \ \forall j, t; \ w_t \in \mathbb{R}_+, \ \forall t. \tag{31}$$

The rostering problem is to minimize the total cost, including wage cost for regular staff, fixed and variable cost for part-time staff, and penalty cost from the unserved demand (24), subject to some constraints such as minimum/maximum workload and maximum consecutive working shifts (Cheang et al. 2003). Constraints in (25) ensure that regular staff cannot work through any three consecutive shifts. A similar requirement on part-time staff is represented in (27) where they cannot work through any two consecutive shifts. Constraints in (26) and (28) are lower and upper bounds on the total number of shifts for regular and part-time staff in the planning horizon. Constraints in (29) link binary activation and continuous working-hour decisions for part-time staff. Constraints in (30) guarantee that the coverage of demands and (31) are individual variable restrictions.

Once a large service demand is anticipated or observed, part-time staff are often called and scheduled to deal with demand fluctuation, which yields the recourse problem. Hence, the rostering problem with uncertain demand can be formulated by the following two-stage robust optimization model. Because the optimal value of the recourse problem is non-decreasing in $\mathbf{d}$, we directly present it in a $\min - \max - \min$ form. It is

$$\textbf{Rostering}_{RO}: \ \min_{\mathbf{x} \in \mathbb{X}} \sum_i \sum_t x_{it} c_{it} + \max_{\mathbf{d} \in \mathbb{D}} \ \min_{(\mathbf{y}, \mathbf{z}, \mathbf{w}) \in \mathbb{F}(\mathbf{x}, \mathbf{d})} \Big\{ \sum_t M_t w_t + \sum_j \sum_t (f_{jt} y_{jt} + h_{jt} z_{jt}) \Big\}, \tag{32}$$

where $\mathbb{D}$ is the demand uncertainty set and $\mathbb{F}(\mathbf{x}, \mathbf{d}) = \{(\mathbf{y}, \mathbf{z}, \mathbf{w}) : (27-30), y_{jt} \in \{0,1\}, z_{jt}, w_t \geq 0, \forall j, t\}$.

Note from (32) that scheduling of regular staff is made with consideration of the worst demand situations as well as the benefits from using part-time staff, a "right-and-now" decision. Then, after the demand is revealed, the recourse rostering decision of part-time staff will be made, a "wait-and-see" decision. Given that the recourse decision problem is a binary mixed integer program, the nested column-and-constraint generation method will be used to solve $\textbf{Rostering}_{RO}$.

4.2 Solving Rostering$_{RO}$ by Nested Column-and-Constraint Generation Algorithm

In this section, the forms of generated variables and constraints in typical iterations of outer and inner level algorithms are provided. The solution procedure with complete details can be obtained by modifying the outer and inner level algorithms in Sections 3.1 and 3.2 accordingly.

4.2.1 Generated Variables and Constraints in the Outer Level:

Assume that $\hat{\mathbf{d}}^k$ is a significant uncertainty scenario obtained in the k^{th} iteration. Then generated variables and constraints in the outer-level algorithm will be as follows.

$$
\begin{aligned}
&\eta \geq \sum_j \sum_t (f_{jt} y_{jt}^k + h_{jt} z_{jt}^k) + \sum_t M_t w_t^k \\
&y_{jt}^k + y_{j,t+1}^k \leq 1,\ \forall j, t \leq T-2 \\
&a_j \leq \sum_t y_{jt}^k \leq b_j,\ \forall j \\
&z_{jt}^k \leq N y_{jt}^k,\ \forall j, t \\
&N \sum_i x_{it} + \sum_j z_{jt}^k + w_t^k \geq \hat{d}_t^k,\ \forall t \\
&y_{jt}^k \in \{0,1\}, z_{jt}^k, w_t^k \geq 0,\ \forall j, t.
\end{aligned}
$$

4.2.2 Solving Mixed Integer Bi-level SP by Inner-Level Algorithm

To derive an optimal solution to bi-level **SP** for a given $\hat{\mathbf{x}}$, first, the binary decision variables are separated from remaining recourse variables, resulting in the following tri-level reformulation:

$$
\max_{\mathbf{d}\in\mathbb{D}} \min_{\mathbf{y}\in\mathbb{Y}} \sum_j \sum_t f_{jt} y_{jt} + \min_{(\mathbf{z},\mathbf{w})\in\mathbb{Q}(\hat{\mathbf{x}},\mathbf{d},\mathbf{y})} \sum_j \sum_t h_{jt} z_{jt} + \sum_t M_t w_t, \tag{33}
$$

where $\mathbb{Y} = \{\mathbf{y} : (27-28); y_{jt} \in \{0,1\}, \forall j, t\}$ and $\mathbb{Q}(\hat{\mathbf{x}}, \mathbf{d}, \mathbf{y}) = \{(\mathbf{z}, \mathbf{w}) : (29-30); z_{jt}, w_t \geq 0, \forall j, t\}$. Next, variables and constraints (and their linearized counterparts) generated by KKT condition and strong duality based inner-level column-and-constraint generation algorithms are presented, respectively.

(a) KKT condition based inner-level algorithm

Assume that $\hat{\mathbf{y}}^l$ is an optimal value by solving the recourse problem for a given $\hat{\mathbf{x}}, \hat{\mathbf{d}}$ in l^{th} inner-level iteration. Let (λ, π) be dual variables of the innermost minimization problem of (33). Then, the generated variables and constraints in the inner-level algorithm to solve (33) are as follows:

$$\theta \leq \sum_j \sum_t (f_{jt} \hat{y}_{jt}^l + h_{jt} z_{jt}^l) + \sum_t M_t w_t^l \tag{34}$$

$$z_{jt}^l \leq N \hat{y}_{jt}^l, \forall j, t \tag{35}$$

$$N \sum_i \hat{x}_{it} + \sum_j z_{jt}^l + w_t^l \geq d_t, \forall t \tag{36}$$

$$h_{jt} \geq -\lambda_{jt}^l + \pi_t^l, \forall j, t \tag{37}$$

$$M_t \geq \pi_t^l, \forall t \tag{38}$$

$$\lambda_{jt}^l (N \hat{y}_{jt}^l - z_{jt}^l) = 0, \forall j, t \tag{39}$$

$$\pi_t^l (N \sum_i \hat{x}_{it} + \sum_j z_{jt}^l + w_t^l - d_t) = 0, \forall t \tag{40}$$

$$z_{jt}^l (h_{jt} + \lambda_{jt}^l - \pi_t^l) = 0, \forall j, t \tag{41}$$

$$w_t^l (M_t - \pi_t^l) = 0, \forall t \tag{42}$$

$$z_{jt}^l \geq 0\ \forall j, t;\ w_t^l \geq 0\ \forall t;\ \lambda_{jt}^l \geq 0\ \forall j, t;\ \pi_t^l \geq 0\ \forall t. \tag{43}$$

Given that $\mathbb{D}$ is bounded mixed integer set, let $\bar{d}_t$ be an upper bound of all possible d_t for $t = 0, \ldots, T-1$. Then, for any fixed $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$, we have the following bounds on primal and dual variables and constraints, which are useful to linearize nonlinear complementary constraints in (39-42).

$$\begin{aligned}
&\lambda_{jt} \leq M_t;\ N\hat{y}_{jt} - z_{jt} \leq N,\ \forall j,t \\
&\pi_t \leq M_t;\ N\sum_i \hat{x}_{it} + \sum_j z_{jt} + w_t - d_t \leq I \times N,\ \forall t \\
&z_{jt} \leq N;\ h_{jt} + \lambda_{jt} - \pi_t \leq h_{jt} + M_t, \forall j,t \\
&w_t \leq \bar{d}_t;\ M_t - \pi_t \leq M_t,\ \forall t.
\end{aligned}$$

Consequently, the generated complementary constraints in (39-42) can be replaced by the following linear constraints with additional binary variables α, β and δ:

$$\begin{aligned}
&\lambda^l_{jt} \leq M_t(1 - \alpha^l_{jt}), \forall j,t \\
&N\hat{y}^l_{jt} - z^l_{jt} \leq N\alpha^l_{jt}, \forall j,t \\
&\pi^l_t \leq M_t(1 - \beta^l_t), \forall t \\
&N\sum_i \hat{x}_{it} + \sum_j z^l_{jt} + w^l_t - d_t \leq I \times N\beta^l_t, \forall t \\
&z^l_{jt} \leq N(1 - \gamma^l_{jt}), \forall j,t \\
&h_{jt} + \lambda^l_{jt} - \pi^l_t \leq (h_{jt} + M_t)\gamma^l_{jt}, \forall j,t \\
&w^l_t \leq \bar{d}_t(1 - \delta^l_t), \forall t \\
&M_t - \pi^l_t \leq M_t\delta^l_t, \forall t \\
&\alpha^l_{jt}, \gamma^l_{jt}\ \in \{0,1\}\ \forall j,t;\ \beta^l_t, \delta^l_t\ \in \{0,1\}\ \forall t.
\end{aligned}$$

(b) Strong duality based inner-level algorithm

By strong duality, (33) can be converted into the following max − min − max problem:

$$\max_{\mathbf{d}\in\mathbb{D}} \min_{\mathbf{y}\in\mathbb{Y}} \sum_j \sum_t f_{jt}y_{jt} + \max_{(\lambda,\pi)\in\mathbb{C}} \sum_t \pi_t(d_t - N\sum_i \hat{x}_{it}) - N\sum_j\sum_t \lambda_{jt}y_{jt},$$

where $\mathbb{Y} = \{\mathbf{y} : (27\text{–}28);\ y_{jt} \in \{0,1\}, \forall j,t\}$ and $\mathbb{C} = \{(\lambda,\pi) : -\lambda_{jt}+\pi_t \leq h_{jt}, \forall j,t;\ \pi_t\ \leq\ M_t, \forall t;\ \lambda_{jt}, \pi_t \geq 0, \forall j,t\}$. Then, for a fixed $\hat{\mathbf{y}}^l$, the generated variables and constraints are as follows:

$$\begin{aligned}
&\theta \leq \sum_j \sum_t f_{jt}\hat{y}^l_{jt} + \sum_t \pi^l_t(d_t - N\sum_i \hat{x}_{it}) - N\sum_j\sum_t \lambda^l_{jt}\hat{y}^l_{jt}, \\
&-\lambda^l_{jt} + \pi^l_t \leq h_{jt},\ \forall j,t; \\
&\pi^l_t \leq M_t\ \forall t; \\
&\lambda^l_{jt}, \pi^l_t \geq 0,\ \forall j,t.
\end{aligned}$$

When $\mathbb{D}$ is a binary set, the first nonlinear constraint can easily be linearized using the previously-mentioned upper bound information. Note that in both nonlinear and linearized forms, fewer variables and constraints are generated by a strong duality based inner algorithm compared to those generated by KKT condition based inner algorithm.

4.3 Experiment Setup

We consider a rostering instance of 12 full-time and 3 part-time employees, i.e., $I = 12$ and $J = 3$, with 8 hours per shift, 3 shifts per day, and 7 days as the planning horizon, i.e., $N = 8$ and $T = 21$. The parameters c_{it} are randomly generated within [5,15]. The fixed cost f_{jt} and hourly rate h_{jt} are randomly generated within [20,30] and [4,8], respectively. The penalty cost for unserved demand is randomly drawn from [40,50]. The lower and upper bounds on the number of working shifts for regular staff, i.e., l_i and u_i, are randomly selected from [4,8] and [8,14], respectively. Similarly, bound parameters for for part-time staff, i.e., a_j and b_j, are randomly generated within [2,4] and [4,6], respectively.

Table 1: Computational results for KKT-conditions based algorithm and $\mathbb{D}^1$

Γ	Time (s)	Outer Iteration	Inner Iteration
3	15	4	3,7,9,10
6	65	5	10,24,25,28,41
9	153	9	11,14,25,45,45,48,18,51,0
12	41	5	9,8,28,34,0
15	123	8	9,14,9,11,11,19,15,14
18	135	10	3,6,9,11,7,7,10,14,6,0

We consider two types of uncertainty sets. The first one is a cardinality-constrained binary set. Assume that d_t independently takes value from an interval $[\underline{d}_t, \underline{d}_t + \xi_t]$ for $t = 0, \ldots, T-1$. Because all of them rarely will take upper bounds in the planning horizon, an integer $\Gamma \in [0, T-1]$ can be introduced to control the number of upper bound values. Formally, we can describe the uncertainty set as

$$\mathbb{D}^1 = \{\mathbf{d} : d_t = \underline{d}_t + \xi_t g_t,\ \forall t;\ \sum_t g_t \leq \Gamma;\ g_t \in \{0,1\},\ \forall t\}.$$

Although the cardinality constrained binary set is simple, some structured polytope uncertainty sets can be reduced to it (Gabrel et al. 2011, Bertsimas and Sim 2003, Zhao and Zeng 2010). The second uncertainty set is a demonstrative general polytope with two overlapping budget constraints as follows. The general polytope set is particularly useful in describing correlated uncertainty factors.

$$\mathbb{D}^2 = \Big\{\mathbf{d} : d_t = \underline{d}_t + \xi_t g_t,\ \forall t; \sum_{t=0}^{T_1+1} g_t \leq \rho_1; \sum_{t=T_1-1}^{T-1} g_t \leq \rho_2;\ g_t \in [0,1],\ \forall t\Big\}.$$

In our numerical study, $\underline{d}_t$ is randomly selected from [30,80] (hour·person) for $t = 0, \ldots, T-1$. In both discrete and polytope uncertainty sets, ξ_t is set to be $0.05\underline{d}_t$.

Two initialization strategies were implemented in our nested algorithm: (i) a feasible point from the uncertainty set is supplied to the outer-level algorithm to generate the first-stage decision in the first iteration. Specifically, a random point with $\sum_t g_t = \Gamma$ is selected in $\mathbb{D}^1$, and a point solving $\max\{\sum_t d_t : d \in \mathbb{D}^2\}$ is selected in $\mathbb{D}^2$; (ii) instead of starting from scratch, the procedure of the inner-level column-and-constraint generation algorithm for **SP** starts with the worst-case uncertain point that solves the previous **SP**.

In our computational experiments, nested column-and-constraint generation variants were implemented in C++ with CPLEX 12.2 on a PC desktop with an Intel Core(TM) 2Duo 3.00GHz CPU and 3.25GB memory. The optimality tolerance is set to be $1e-3$ for both outer and inner algorithms. The whole procedure will be terminated after 600 seconds.

4.4 Computational Results

The computational results for cardinality constrained binary uncertainty sets using KKT condition and strong duality based nested algorithms are given in Table 1 and 2, respectively. Typical convergence behavior of outer- and inner-level algorithms is shown in Figure 1 and 2, respectively. Note from Step 3 of the outer-level algorithm in Section 3.1 that the whole nested algorithm could terminate in one iteration without performing the inner-level procedure. If this is the case, as shown in Table 1 and 2, the last outer-level iteration carries zero inner iteration. From this computational study, we observe that these two methods have comparable performance. Probably because the primal information is kept and the feasibility problem is easier, the KKT condition based nested algorithm computes solutions a little bit faster. For some instances, we observe that these two algorithms have a different number of iterations, in both outer- and lower-level procedures. Such different numbers of iterations most likely stem from the setup of the optimality tolerance, which is $1e-3$ for all problems. Indeed, the iteration numbers are more consistent if the tolerance is reduced to $1e-4$ with a larger computational time limit, which confirms our remarks on the number of iterations using those two different inner-level procedures.

Table 3 shows the computational results for the polyhedral uncertainty sets using the KKT condition based algorithm, which also suggests that two-stage RO with a polyhedral uncertainty set

Table 2: Computational results for strong duality based algorithm and $\mathbb{D}^1$

Γ	Time (s)	Outer Iteration	Inner Iteration
3	24	5	3,11,4,12,12
6	78	4	10,25,27,36
9	279	9	11,13,27,45,45,51,18,50,0
12	70	5	9,8,28,35,0
15	129	8	9,14,9,11,12,18,15,14
18	180	11	3,6,8,10,7,8,9,8,8,10,11

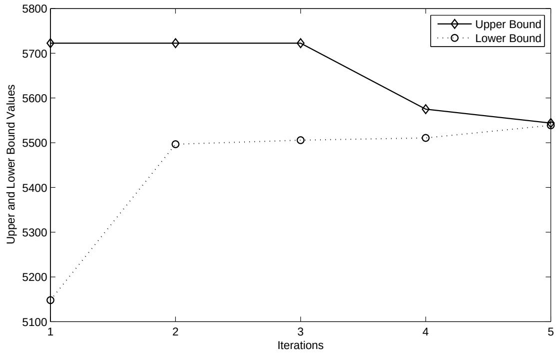

Figure 1: Performance of the outer-level algorithm (with KKT-based inner level) for $\mathbb{D}^1$ and $\Gamma = 12$.

is computationally more challenging than that with a discrete set. For example, the final gap can be reduced only to be 0.14% after 600 seconds running for the case of $(\rho_1, \rho_2) = (0.3, 0.2)$.

5 Conclusion

In this paper, we presented an exact algorithm, the nested column-and-constraint generation method, to solve the two-stage robust optimization problem with an MIP recourse problem. We also derived its convergence property and demonstrated its computational behavior on a simple two-stage robust rostering problem. To the best of our knowledge, because there is no exact algorithm available to solve this type of two-stage robust optimization problem, this algorithm is the first solution procedure to derive optimal solutions. As a result, the nested column-and-constraint generation method and the basic one presented in (Zeng and Zhao 2011) constitute an unified tool set to solve two-stage robust optimization models. In addition to solving robust models, the nested column-and-constraint generation method can be used to compute some four-level programs. Also, its subroutine, the inner-level column-and-constraint generation method, yields an effective approach to solve general mixed integer bi-level programs.

We would like to point out that the (nested) column-and-constraint generation method in its

Table 3: Computational results for KKT-conditions based algorithm and $\mathbb{D}^2$

(ρ_1, ρ_2)	Time (s)	Outer Iteration	Inner Iteration
(0.2,0.2)	279	12	7,13,16,15,14,10,17,14,9,14,17,0
(0.3,0.3)	262	9	6,8,20,14,19,13,11,19,0
(0.2,0.3)	501	12	7,15,8,14,10,15,15,11,13,18,13,0
(0.3,0.2)*	600*	12	6,12,11,12,18,15,11,6,14,17,14,12

Case (0.3,0.2) is terminated after 600 seconds with a final gap of 0.14%

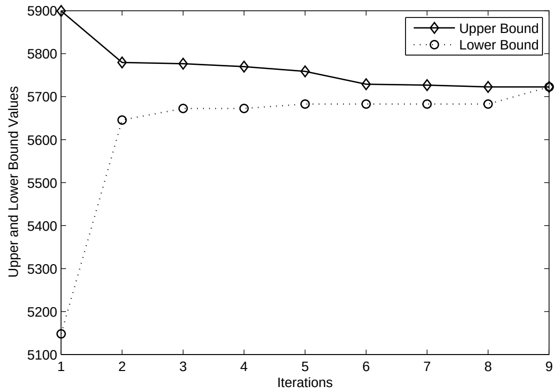

Figure 2: Performance of KKT-based inner algorithm within the first outer level iteration for $\mathbb{D}^1$ with $\Gamma = 12$.

current description provides only a basic scheme to solve general two-stage robust optimization problems. Clearly, more advanced and sophisticated techniques are needed to refine this method to deal with large-scale real problems in a reasonable time. We observe that two strategies can be studied to achieve this goal. One is to investigate efficient algorithms to solve master problems, i.e., $\mathbf{MP}$ and $\mathbf{MP_S}$ in outer and inner levels, respectively, which share a great similarity to classical scenario-based stochastic programming models. So, it would be interesting to develop algorithms based on Benders decomposition or L-shape method. Note also that $\mathbf{MP_S}$ has many complementarity constraints. Provided that $\mathbb{U}$ is a polytope, $\mathbf{MP_S}$ is a linear program with complementarity constraints. Then, instead of introducing binary variables and *big-M* to linearize those constraints, another improvement strategy is to solve $\mathbf{MP_S}$ by algorithms that are specific for linear programs with complementarity constraints. Another interesting direction is to investigate the hybrid strategy, combining the (nested) column-and-constraint generation and the (revised) affine rule methods, to develop algorithms that allow users to achieve an optimal trade-off between solution quality and computational time for complex applications.

References

S. Ahmed, M. Tawarmalani, and N.V. Sahinidis. A finite branch-and-bound algorithm for two-stage stochastic integer programs. *Mathematical Programming*, 100(2):355–377, 2004.

A. Atamturk and M. Zhang. Two-stage robust network flow and design under demand uncertainty. *Operations Research*, 55(4):662–673, 2007.

A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.

A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1):1–14, 1999.

A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88(3):411–424, 2000.

A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.

D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming*, 98(1):49–71, 2003.

D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

D. Bertsimas, D.B. Brown, and C. Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53:464–501, 2011a.

D. Bertsimas, E. Litvinov, X.A. Sun, J. Zhao, and T. Zheng. Adaptive robust optimization for the security constrained unit commitment problem. Technical report, submitted to *IEEE Transactions on Power Systems*, 2011b.

C.C. Carøe and J. Tind. L-shaped decomposition of two-stage stochastic programs with integer recourse. *Mathematical Programming*, 83(1):451–464, 1998.

B. Cheang, H. Li, A. Lim, and B. Rodrigues. Nurse rostering problems–a bibliographic survey. *European Journal of Operational Research*, 151(3):447–460, 2003.

L. El Ghaoui, F. Oustry, and H. Lebret. Robust solutions to uncertain semidefinite programs. *SIAM Journal of Optimization*, 9:33–52, 1998.

A.T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier. Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research*, 153(1):3–27, 2004.

V. Gabrel, M. Lacroix, C. Murat, and N. Remli. Robust location transportation problems under uncertain demands. Technical report, submitted to *Discrete Applied Mathematics*, Lamsade, Universite Paris-Dauphine, 2011.

T. Hoang. *Convex Analysis and Global Optimization*, volume 22. Springer, 1998.

R. Jiang, M. Zhang, G. Li, and Y. Guan. Benders decomposition for the two-stage security constrained robust unit commitment problem. Technical report, available in *optimization-online*, 2011.

G. Laporte and F.V. Louveaux. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, 13(3):133–142, 1993.

F. Ordonez and J. Zhao. Robust capacity expansion of network flows. *Networks*, 50(2):136–145, 2007.

S. Sen. Algorithms for stochastic mixed-integer programming models. *Handbook of Discrete Optimization*, pages 515–558, 2003.

S. Sen and H.D. Sherali. Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming. *Mathematical Programming*, 106(2):203–223, 2006.

K. Shimizu, Y. Ishizuka, and J.F. Bard. *Nondifferentiable and Two-level Mathematical Programming*. Kluwer Academic Pub, 1997.

A. Takeda, S. Taguchi, and RH Tutuncu. Adjustable robust optimization models for a nonlinear two-period system. *Journal of Optimization Theory and Applications*, 136(2):275–295, 2008.

A. Thiele, T. Terry, and M. Epelman. Robust linear optimization with recourse. Technical report, available in *optimization-online*, 2010.

B. Zeng and L. Zhao. Solving two-stage robust optimization problems using a column-and-constraint generation method. Technical report, under revision, available in *optimization-online*, University of South Florida, 2011.

L. Zhao and B. Zeng. Robust unit commitment problem with demand response and wind energy. Technical report, available in *optimization-online*, University of South Florida, 2010.

L. Zhao and B. Zeng. An exact algorithm for power grid interdiction problem with line switching. Technical report, submitted, available in *optimization-online*, University of South Florida, 2011.