

Decomposition Algorithms for Two-Stage Distributionally Robust Mixed Binary Programs

Manish Bansal

Grado Department of Industrial and Systems Engineering, Virginia Tech
Email: bansal@vt.edu

Kuo-Ling Huang and Sanjay Mehrotra

Department of Industrial Engineering and Management Sciences, Northwestern University
Emails: {jupiters1117, mehrotra}@northwestern.edu

February 7, 2017

Abstract. In this paper, we introduce and study a two-stage distributionally robust mixed binary problem (TSDR-MBP) where the random parameters follow the worst-case distribution belonging to an uncertainty set of probability distributions. We present a decomposition algorithm, which utilizes distribution separation procedure and parametric cuts within Benders' algorithm or L-shaped method, to solve TSDR-MBPs with binary variables in the first stage and mixed binary programs in the second stage. We refer to this algorithm as distributionally robust integer (DRI) L-shaped algorithm. Using similar decomposition framework, we provide another algorithm to solve TSDR linear problem where both stages have only continuous variables. We investigate conditions and the families of ambiguity set for which our algorithms are finitely convergent. We present two examples of ambiguity set, defined using moment matching, or Kantorovich-Rubinstein distance (Wasserstein metric), which satisfy the foregoing conditions. We also present a cutting surface algorithm to solve TSDR-MBPs. We computationally evaluate the performance of the DRI L-shaped algorithm and the cutting surface algorithm in solving distributionally robust versions of a few instances from the Stochastic Integer Programming Library, in particular stochastic server location and stochastic multiple binary knapsack problem instances. We also discuss the usefulness of incorporating partial distribution information in two-stage stochastic optimization problems.

1. Introduction

Stochastic programming and robust optimization are well established optimization models used for making decisions under parametric uncertainty. In stochastic programs, it is assumed that the uncertain parameters follow a specified probability distribution and an objective function is specified by taking expectation over these parameters. On the other hand, robust optimization handles the uncertainty by solving a minimax problem, where the max is taken over a set of possible values of uncertain parameters. These two competing ideas in decision making under uncertainty can be unified in the framework of distributionally-robust optimization (DRO). In DRO we seek a solution that optimizes the expected value of the objective function for the worst case probability distribution within a prescribed (ambiguity) set of distributions that may be followed by the uncertain parameters. Note that in DRO, the exact distribution followed by the uncertain parameters is unknown.

In this paper, we consider a unification of DRO with two-stage stochastic mixed binary programs (TSS-MBPs), thereby leading to a two-stage distributionally robust optimization (TSDRO) framework with binary variables in both stages. More specifically, we study the following two-stage

distributionally robust mixed binary program (TSDR-MBP):

$$\min \left\{ c^T x + \max_{P \in \mathfrak{P}} \mathbb{E}_{\xi_P} [\mathcal{Q}_\omega(x)] \mid Ax \geq b, x \in \{0, 1\}^p \right\}. \quad (1)$$

where ξ_P is a random vector defined by probability distribution P with support Ω and for any scenario ω of Ω ,

$$\mathcal{Q}_\omega(x) := \min g_\omega^T y_\omega \quad (2a)$$

$$\text{s.t. } W_\omega y_\omega \geq r_\omega - T_\omega x \quad (2b)$$

$$y_\omega \in \{0, 1\}^{q_1} \times \mathbb{R}^{q-q_1}. \quad (2c)$$

We refer to the set of distributions \mathfrak{P} as the *ambiguity set*. In (1), the parameters $c \in \mathbb{Q}^p$, $A \in \mathbb{Q}^{m_1 \times p}$, $b \in \mathbb{Q}^{m_1}$, and for each $\omega \in \Omega$, $g_\omega \in \mathbb{Q}^q$, *recourse matrix* $W_\omega \in \mathbb{Q}^{m_2 \times q}$, *technology matrix* $T_\omega \in \mathbb{Q}^{m_2 \times p}$, and $r_\omega \in \mathbb{Q}^{m_2}$. The formulation defined by (2) and the function $\mathcal{Q}_\omega(x)$ are referred to as the second-stage subproblem and recourse function, respectively.

We assume that

1. $X := \{x : Ax \geq b, x \in \{0, 1\}^p\}$ is non-empty,
2. $\mathcal{K}_\omega(x) := \{y_\omega : (2b)-(2c) \text{ hold}\}$ is non-empty for all $x \in X$ and $\omega \in \Omega$,
3. $\mathcal{Q}_\omega(x) < \infty$ for all $x \in X$ and $\omega \in \Omega$ (relatively complete recourse),
4. Each probability distribution $P \in \mathfrak{P}$ has finite support Ω , i.e. $|\Omega|$ is finite.

Without loss of generality, we also assume that all elements of the following vectors or matrices are integer valued (which can be obtained by multiplying these parameters with appropriate multipliers): c , A , b , and for $\omega \in \Omega$, g_ω , W_ω , T_ω , and r_ω . Furthermore, we assume that there exists an oracle that provides a probability distribution $P \in \mathfrak{P}$, i.e., $\{p_\omega\}_{\omega \in \Omega}$ where p_ω is the probability of occurrence of scenario $\omega \in \Omega$, by solving the optimization problem:

$$\mathcal{Q}(x) := \max_{P \in \mathfrak{P}} \mathbb{E}_{\xi_P} [\mathcal{Q}_\omega(x)] \quad (3)$$

for a given $x \in X$. We refer to this optimization problem as distribution separation problem corresponding to an ambiguity set, and the algorithm to solve this problem is referred to as the *distribution separation algorithm*. In Section 1.7, we provide two examples of ambiguity set and the distribution separation problems associated to them. To the best of our knowledge, TSDR-MBPs have not been studied before.

1.1 Contributions of this paper

We present a decomposition algorithm, which utilizes distribution separation procedure and parametric cuts within Benders' algorithm [7], to solve TSDR-MBPs. We refer to this algorithm as distributionally robust integer (DRI) L-shaped algorithm because it generalizes the well-known integer L-shaped algorithm [19] developed for a special case of TSDR-MBP where \mathfrak{P} is singleton. Using similar decomposition framework, we develop a decomposition algorithm for two-stage distributionally robust linear program, i.e. TSDR-MIP with no binary restrictions on both first and second stage variables. Moreover, we provide conditions and the families of ambiguity set \mathfrak{P} for

which our algorithms are finitely convergent. Furthermore, we present a cutting surface algorithm to solve TSDR-MBPs and computationally evaluate the performance of DRI L-shaped algorithm and the cutting surface algorithm in solving DR versions of stochastic server location problem and stochastic multiple binary knapsack problem instances from Stochastic Integer Programming Library (SIPLIB) [1]. We observe that our DRI L-shaped algorithm solves all these problem instances to optimality in less than an hour. It is important to note that the TSDR-MBP generalizes various classes of optimization problems studied in literature. Below are few of them:

- *Two-Stage Stochastic Mixed Binary Program* (TSS-MBP): TSDR-MBP (1) with a singleton $\mathfrak{P} = \{P_0\}$ defines TSS-MBP [15, 19, 29].
- *Two-Stage Robust Mixed Binary Program*(TSR-MBP): TSDR-MBP (1) with a set \mathfrak{P} that consists of all probability distributions supported on Ω is equivalent to TSR-MBP.
- *Distributionally Robust Optimization* (DRO): In literature, the general DRO problem [12, 21, 26, 27, 28] is defined as follows:

$$\min_{x \in \bar{X}} \max_{P \in \mathfrak{P}} \mathbb{E}_{\xi_P} [H_\omega(x)] \quad (4)$$

where H_ω is a random cost or disutility function corresponding to scenario ω . Assuming that $\bar{X} = X$ and H_ω is defined by a mixed binary program, Problem (4) is a special case of TSDR-MBP (i.e., Problem (1)) where c is a vector of zeros.

- *Two-Stage Distributionally Robust Linear Program* (TSDR-LP): The TSDR-LP [9, 16, 20] is a relaxation of TSDR-MBP where the both stages have only continuous variables.

The remaining section provides significance of TSDRO framework (Section 1.2), literature review of TSS-MBP, DRO framework and TSDR-LP, and ambiguity sets (Section 1.3, 1.4, and 1.5, respectively), and organization of this paper (Section 1.6).

1.2 Significance of TSDRO framework

The ability to allow incomplete information on the probability distribution is a major advantage of the TSDRO approach to model formulation. In a two-stage decision framework where future predictions are generated using a data driven approach, no assumptions on the knowledge of prediction error distribution is required. The prediction errors can be empirically generated, and the TSDRO framework can be used to robustify around the empirical error distribution. Alternatively, when the uncertain parameters in the model are specified by uncertainty quantification (UQ) techniques, this framework allows one to model errors in UQ without requiring the errors to follow a specified (e.g., normal) distribution. In addition, TSDRO framework provides decision sensitivity analysis with respect to the reference probability distribution.

One can also view the TSDR-MBP not only as a common generalization of TSR-MBP and TSS-MBP, but also as an optimization model with an adjustable level of risk-aversion. To see this, consider a nested sequence of sets of probability distributions $\mathfrak{P}_0 \supseteq \mathfrak{P}_1 \supseteq \dots$, where \mathfrak{P}_0 is the set of all probability distributions supported on Ω , and $\mathfrak{P}_\infty \stackrel{\text{def}}{=} \bigcap_{i=0}^{\infty} \mathfrak{P}_i$ is a singleton set. In the corresponding sequence of problems (1), the first one ($\mathfrak{P} = \mathfrak{P}_0$) is the TSR-MBP, which is the most conservative (risk-averse) of all, and the last one is the TSS-MBP, where the optimization is against a fixed distribution. At the intermediate levels the models correspond to decreasing levels of risk-aversion.

1.3 Literature review of TSS-MBPs

Laporte and Louveaux [19] provide the integer L-shaped algorithm for TSS-MBP by assuming that the second stage problems are solved to optimality at each iteration. Sherali and Fraticelli [29] consider single scenario TSS-MBPs, i.e. $\Omega := \{\omega_1\}$. They develop globally valid cuts in (x, y_{ω_1}) space using the reformulation-linearization technique. These cuts are then used while solving the second stage mixed binary programs for a given $x \in X$ and hence the cuts in y_{ω} space are referred to as the “parametric” cuts. Likewise, Gade et. al. [15] utilize the parametric Gomory fractional cuts within Benders’ decomposition algorithm for solving TSS-MIPs with only binary variables in the first stage and non-negative integer variables in the second stage. Note that in the aforementioned papers, the parametric cuts are added in succession. In contrast, Bansal et al. [3] provide conditions under which the second stage mixed integer programs of two-stage stochastic mixed integer program (TSS-MIP) can be convexified by adding parametric inequalities a priori. They provide examples of TSS-MIPs which satisfy these conditions by considering parametrized versions of some structured mixed integer set such as special cases of continuous multi-mixing set [5, 4], and convex objective integer programs in the second stage.

1.4 Literature review of DRO framework and TSDR-LPs

Scarf [26] introduced the concept of DRO by considering a news vendor problem. Thereafter, this framework has been used to model varieties of problems [13, 14, 23]. On algorithmic front for solving DRO problems, Shapiro and Kleywegt [28] and Shapiro and Ahmed [27] provide approaches to derive an equivalent stochastic program with a certain distribution. Delage and Ye [12] give general conditions for polynomial time solvability of DRO where the ambiguity set is defined by setting constraints on first and second moments. Using ellipsoidal method, they [12] also show that under certain conditions on the disutility functions H_{ω} and the ambiguity set \mathfrak{B} , Problem (4) is solvable in polynomial time. Mehrotra and Zhang [22] extend this result by providing polynomial time methods for distributionally robust least squares problems, using semidefinite programming. Recently, Mehrotra and Papp [21] develop a central cutting surface algorithm for Problem (4) where the ambiguity set is defined using constraints on first to fourth moments.

Lately researchers have been considering two-stage stochastic linear programs with ambiguity sets, in particular TSDR-LP [9, 20, 16]. More specifically, Bertsimas et al. [9] consider TSDR-LP where the ambiguity set is defined using multivariate distributions with known first and second moments and risk is incorporated in the model using a convex nondecreasing piecewise linear function on the second stage costs. They show that the corresponding problem has semidefinite programming reformulations. Jiang and Guan [18] present sample average approximation algorithm to solve a special case of TSDR-MBP with binary variables only in the first stage where the ambiguity set is defined using l_1 -norm on the space of all (continuous and discrete) probability distributions. Recently, Love and Bayraksan [20] develop a decomposition algorithm for solving TSDR-LP where the ambiguity set is defined using ϕ -divergence. Whereas Hanasusanto and Kuhn [16] provide a conic programming reformulations for TSDR-LP where the ambiguity set comprises of a 2-Wasserstein ball centered at a discrete distribution. As per our knowledge, no work has been done to solve TSDR-LP with binary variables in both stages, i.e. TSDR-MBP. In this paper, we develop new decomposition algorithms for TSDR-LP and TSDR-MBP with a general family of ambiguity sets, and provide conditions and families of ambiguity sets for which these algorithms are finitely convergent.

1.5 Literature review of ambiguity sets

Though in this paper we consider a general family of ambiguity sets, in literature there exists different ways to construct the ambiguity set of distributions; refer to [17] and references therein for more details. To begin with, Scarf [26] define the ambiguity set using linear constraints on the first two moments of the distribution. Similar ambiguity set is also considered in [8, 13, 25]; whereas Bertsimas et al. [9] and Delage and Ye [12] use conic constraints to describe the set of distributions with moments, and a more general model allowing bounds on higher order moments has been recently studied in [21]. Other definitions of the ambiguity sets considered in the literature include the usage of the measure bounds and general moment constraints [22, 27], Kantorovich distance or Wasserstein metric [22, 24, 23, 31], ζ -structure metrics [34], ϕ -divergences such as χ^2 distance and Kullback-Leibler divergence [6, 10, 18, 20, 30, 32], and Prokhorov metrics [14]. We give two examples of families of ambiguity set in Section 1.7 and also provide distribution separation problem associated to each of the ambiguity sets.

1.6 Organization of this paper

In Section 2, we present a decomposition algorithm to solve TSDR-LP by embedding distribution separation algorithm (associated to the ambiguity set) within L-shaped method. We refer to our algorithm as the distributionally robust L-shaped algorithm and provide families of ambiguity set for which it is finitely convergent. In Section 3, we further extend our algorithm to solve TSDR-MBPs using parametric cuts, thereby generalizing the integer L-shaped method. We refer to this generalized algorithm as the *distributionally robust integer (DRI) L-shaped algorithm*. In Section 3.3, we provide conditions and the families of ambiguity set for which the DRI L-shaped algorithm is finitely convergent. Interestingly, the two examples of ambiguity set discussed in Section 1.7 satisfy the aforementioned conditions. In Section 3.4, we present a cutting surface algorithm where we solve the subproblems to optimality using branch-and-cut approach. Thereafter, in Section 4 we computationally evaluate the performance of the DRI L-shaped algorithm and the cutting surface algorithm in solving distributionally robust versions of problem instances taken from the Stochastic Integer Programming Library (SIPLIB) [1]. In particular, we consider instances of the stochastic server location problem and stochastic multiple binary knapsack problem. Finally, we give concluding remarks in Section 5.

1.7 Examples of ambiguity set and distribution separation procedures

In this section, we provide two examples of ambiguity sets and the distribution separation algorithm associated to these sets. These sets, referred to as the *moment matching set* (5) and *Kantorovich set* (7), are defined by polytopes with finite number of extreme points.

Moment matching set. We define the set \mathfrak{P} via bounds on some (not necessarily polynomial) moments. For a finite sample space $\Omega := \{\omega^1, \dots, \omega^{|\Omega|}\}$, let $v := (v_1, \dots, v_{|\Omega|})$ be the corresponding probability measure. Given continuous *basis functions* f_1, \dots, f_N defined for $(\Omega, X) \mapsto \mathbb{R}$, the moment matching set is given by:

$$\mathfrak{P}_M := \left\{ v \in \mathbb{R}_+^{|\Omega|} : \underline{u} \leq \sum_{l=1}^{|\Omega|} v_l f(\omega^l) \leq \bar{u} \right\}, \quad (5)$$

where \underline{u} and \bar{u} are lower and upper bound vectors, respectively, on the corresponding moments. In addition, the optimization (or distribution separation) problem (3) associated with \mathfrak{P}_M is a linear program:

$$\max_{v \in \mathbb{R}^{|\Omega|}} \left\{ \sum_{l=1}^{|\Omega|} v_l \mathcal{Q}_{\omega^l}(x) \mid \underline{u} \leq \sum_{l=1}^{|\Omega|} v_l f(\omega^l) \leq \bar{u}, v \geq 0 \right\}, \quad (6)$$

where decision variables are the weights v_l that the distribution P assigns to each point $\omega^l \in \Omega$ for $l = 1, \dots, |\Omega|$.

Kantorovich set. The use of Kantorovich-Rubinstein (KR) distance (or Wasserstein metric) is another important choice in specifying ambiguity in a distribution. Assume that \mathbb{P}^* is a known reference probability measure and $\epsilon > 0$ is given. Again, if $\Omega := \{\omega^1, \dots, \omega^{|\Omega|}\}$ is a finite set, $v := (v_1, \dots, v_{|\Omega|})$ is the corresponding probability measure, and $v_j^*, j = 1, \dots, |\Omega|$, are given probabilities corresponding to a reference distribution defined on the sample space, then the Kantorovich set is given by:

$$\begin{aligned} \mathfrak{P}_K := \left\{ v \in \mathbb{R}^{|\Omega|} : \sum_{i=1}^{|\Omega|} \sum_{j=1}^{|\Omega|} \|\omega^i - \omega^j\|_1 k_{i,j} \leq \epsilon, \right. \\ \sum_{j=1}^{|\Omega|} k_{i,j} = v_i, \quad i = 1, \dots, |\Omega| \\ \sum_{i=1}^{|\Omega|} k_{i,j} = v_j^*, \quad j = 1, \dots, |\Omega| \\ \sum_{i=1}^{|\Omega|} v_i = 1 \\ v_i \geq 0, \quad i = 1, \dots, |\Omega| \\ \left. k_{i,j} \geq 0, \quad i = 1, \dots, |\Omega|, j = 1, \dots, |\Omega| \right\}, \quad (7) \end{aligned}$$

and the associated distribution separation problem is given by the following linear program:

$$\max \left\{ \sum_{l=1}^{|\Omega|} v_l \mathcal{Q}_{\omega^l}(x) : v \in \mathfrak{P}_K \right\}. \quad (8)$$

In (8), the decision variables are $k_{i,j}$ and v_i for $i, j \in \{1, \dots, |\Omega|\}$, and the constraints are similar to those in a standard transportation problem with an additional inequality constraint.

2. Distributionally robust L-shaped algorithm for TSDR-LP

In this section, we develop a decomposition algorithm which utilizes distribution separation algorithm within Benders' method for solving TSDR-LP where the ambiguity set \mathfrak{P} is defined by a polytope with a finite number of extreme points (for example, moment matching set \mathfrak{P}_M and Kantorovich set \mathfrak{P}_K). Recall that TSDR-LP is a relaxation of TSDR-MBP, i.e. (1), where the both stages have only continuous variables. We refer to our algorithm as the distributionally robust

L-shaped algorithm. The pseudocode of this algorithm is given by Algorithm 1. Now, let LB and UB be the lower bound and upper bound, respectively, on the optimal solution value of a given TSDR-LP. We define subproblem $\mathcal{S}_\omega(x)$, for $x \in X_{LP} := \{x \in \mathbb{R}^p : Ax \geq b\}$ and $\omega \in \Omega$, as follows:

$$\mathcal{Q}_\omega^s(x) := \min g_\omega^T y_\omega \quad (9a)$$

$$s.t. \quad W_\omega y_\omega \geq r_\omega - T_\omega x \quad (9b)$$

$$y_\omega \in \mathbb{R}^q. \quad (9c)$$

Let $\pi_{\omega,0}^*(x) \in \mathbb{R}^{m_2}$ be the optimal dual multipliers corresponding to constraints (9b) which are obtained by solving $\mathcal{S}_\omega(x)$ for a given $x \in X_{LP}$ and $\omega \in \Omega$. We derive a lower bounding approximation of the linear programming relaxation of the first stage problem (1) using the following *optimality cut*, $\text{OCS}(\pi_{\omega,0}^*(x), \{p_\omega\}_{\omega \in \Omega})$:

$$\sum_{\omega \in \Omega} p_\omega \left\{ \pi_{\omega,0}^*(x)^T (r_\omega - T_\omega x) \right\} \leq \theta, \quad (10)$$

where $\{p_\omega\}_{\omega \in \Omega}$ is obtained by solving the distribution separation problem associated to the ambiguity set \mathfrak{P} . We refer to the lower bound approximation of the first stage problem (1) as the *master problem* which is defined by

$$\min \{c^T x + \theta : x \in X_{LP} \text{ and } \text{OCS}(\pi_{\omega,0}^*(x^k), \{p_\omega^k\}_{\omega \in \Omega}) \text{ holds, for } k = 1, \dots, l\} \quad (11)$$

where $x^k \in X_{LP}$ for $k = 1 \dots, l$, $\pi_{\omega,0}^*(x^k)$ is the set of optimal dual multiplier obtained by solving $\mathcal{S}_\omega(x^k)$, $\omega \in \Omega$, and $\{p_\omega^k\}_{\omega \in \Omega}$ is the set of probabilities for scenarios in sample space Ω obtained by solving distribution separation problem for $x^k \in X_{LP}$, $k \in \{1, \dots, l\}$. We denote this problem by \mathcal{M}_l for $l \in \mathbb{Z}_+$. Note that \mathcal{M}_0 is the master problem without any optimality cut.

Now, we initialize Algorithm 1 by setting lower bound LB to negative infinity, upper bound UB to positive infinity, iteration counter l to 1, and by solving \mathcal{M}_0 to get a first stage feasible solution $x^1 \in X_{LP}$ (Line 1). At each iteration $l \geq 1$, we solve linear programs $\mathcal{S}_\omega(x^l)$ for all $\omega \in \Omega$ and store the corresponding optimal solution $y_\omega^*(x^l)$ and the optimal solution value $\mathcal{Q}_\omega^s(x^l) := g_\omega^T y_\omega^*(x^l)$ for each $\omega \in \Omega$ (Lines 3-5). Next, we solve the distribution separation problem associated to the ambiguity set \mathfrak{P} and obtain the optimal solution, i.e. $\{p_\omega^l\}_{\omega \in \Omega}$ (Line 7). Since $y_\omega^*(x^l) \in \mathcal{K}_\omega(x^l)$ for all $\omega \in \Omega$, we have a feasible solution $(x^l, y_{\omega_1}^*(x^l), \dots, y_{\omega_{|\Omega|}}^*(x^l))$ for the original problem. Therefore, using $\{p_\omega^l\}_{\omega \in \Omega}$, we update UB if the solution value corresponding to thus obtained feasible solution is smaller than the existing upper bound (Lines 8-9). We also utilize the stored information and optimal dual multipliers (Line 14) to derive optimality cut $\text{OCS}(\pi_{\omega,0}^*(x^l), \{p_\omega^l\}_{\omega \in \Omega})$ using (10) and add this cut to the master problem \mathcal{M}_{l-1} to get an augmented master problem \mathcal{M}_l (Lines 15-16). We solve the master problem \mathcal{M}_l (a linear program) and use thus obtained optimal solution value to update lower bound LB (Lines 17-18). Let (x^l, θ^l) be the optimal solution of \mathcal{M}_l . It is important to note that the lower bound LB is a non-increasing with respect to the iterations. This is because \mathcal{M}_{l-1} is a relaxation of \mathcal{M}_l for each $l \geq 1$. Therefore, after every iteration the difference between the bounds, $UB - LB$, either decreases or remains same as in the previous iteration. We terminate our algorithm when this difference becomes zero, i.e., $UB = LB$, (Line 2 or Lines 17-19), and return the optimal solution $(x^l, \{y(\omega, x^l)\}_{\omega \in \Omega})$ and the optimal solution value UB (Line 21).

Algorithm 1 Distributionally Robust L-shaped Method for TSDR-LP

```
1: Initialization:  $l \leftarrow 1$ ,  $LB \leftarrow -\infty$ ,  $UB \leftarrow \infty$ . Solve  $\mathcal{M}_0$  to get  $x^1 \in X_{LP}$ .
2: while  $UB > LB$  do  $\triangleright \epsilon$  is a pre-specified tolerance
3:   for  $\omega \in \Omega$  do
4:     Solve linear program  $\mathcal{S}_\omega(x^l)$ ;
5:      $y_\omega^*(x^l) \leftarrow$  optimal solution;  $Q_\omega^s(x^l) \leftarrow$  optimal solution value;
6:   end for
7:   Solve distribution separation problem using  $Q_\omega^s(x^l)$ ,  $\omega \in \Omega$ , to get  $\{p_\omega^l\}_{\omega \in \Omega}$ ;
8:   if  $UB > c^T x^l + \sum_{\omega \in \Omega} p_\omega^l Q_\omega^s(x^l)$  then
9:      $UB \leftarrow c^T x^l + \sum_{\omega \in \Omega} p_\omega^l Q_\omega^s(x^l)$ ;
10:    if  $UB \leq LB$  then
11:      Go to Line 21;
12:    end if
13:  end if
14:   $\pi_{\omega,0}^*(x^l) \leftarrow$  optimal dual multipliers obtained by solving  $\mathcal{S}_\omega(x^l)$  for all  $\omega \in \Omega$ ;
15:  Derive optimality cut  $\text{OCS}(\pi_{\omega,0}^*(x^l), \{p_\omega^l\}_{\omega \in \Omega})$  using (10);
16:  Add  $\text{OCS}(\pi_{\omega,0}^*(x^l), \{p_\omega^l\}_{\omega \in \Omega})$  to  $\mathcal{M}_{l-1}$  to get  $\mathcal{M}_l$ ;
17:  Solve master problem  $\mathcal{M}_l$  (a linear program);
18:   $(x^{l+1}, \theta^{l+1}) \leftarrow$  optimal solution of  $\mathcal{M}_l$ ;  $LB \leftarrow$  optimal solution value of  $\mathcal{M}_l$ ;
19:  Set  $l \leftarrow l + 1$ ;
20: end while
21: return  $(x^l, \{y_\omega^*(x^l)\}_{\omega \in \Omega})$ ,  $UB$ 
```

2.1 Finite convergence

We present conditions under which Algorithm 1 (DR L-shaped algorithm) solves TSDR-LP in finitely many iterations.

Theorem 1 (DR L-shaped Algorithm). *Algorithm 1 solves the TSDR-LP to optimality in finitely many iterations if assumptions (1)-(3) defined in Section 1 are satisfied and the ambiguity set \mathfrak{P} is defined by a polytope with a finite number of extreme points.*

Proof. In Algorithm 1, for a given $x^l \in X_{LP}$, we solve $|\Omega|$ number of linear programs, i.e. $\mathcal{S}_\omega(x^l)$ for all $\omega \in \Omega$, distribution separation problem associated with the ambiguity set \mathfrak{P} , and a master problem \mathcal{M}_l which is also a linear program. Assuming that the ambiguity set \mathfrak{P} is defined with a polytope with a finite number of extreme points which means that the associated distribution separation algorithm is finitely convergent, it is clear that Lines 3-19 in Algorithm 1 are performed in finite iterations. Now we have to ensure that the “while” loop in Line 2 terminates after finite iterations and provide the optimal solution.

Assuming that for each $x \in X_{LP}$ and $\omega \in \Omega$ there exist finite solutions to the second stage programs (Assumptions 1-3 defined in Section 1), we first prove that the optimality cuts (10) are supporting hyperplanes of $Q^s(x) := \max_{P \in \mathfrak{P}} \mathbb{E}_{\xi_P}[Q_\omega^s(x)]$ for all $x \in X_{LP}$. Notice that the original

problem TSDR-LP is equivalent to

$$\min (c^T x + \theta) \tag{12}$$

$$s.t. \ x \in X_{LP} \tag{13}$$

$$\max_{P \in \mathfrak{P}} \left\{ \sum_{\omega \in \Omega} p_{\omega} \mathcal{Q}_{\omega}^s(x) \right\} \leq \theta. \tag{14}$$

Furthermore, for each $x^l \in X_{LP}$,

$$\mathcal{Q}_{\omega}^s(x^l) = \pi_{\omega,0}^*(x^l)^T (r_{\omega} - T_{\omega} x^l),$$

and after solving the distribution separation problem for x^l , we get

$$\mathcal{Q}^s(x^l) = \max_{P \in \mathfrak{P}} \left\{ \sum_{\omega \in \Omega} p_{\omega} \mathcal{Q}_{\omega}^s(x^l) \right\} = \sum_{\omega \in \Omega} p_{\omega}^l \mathcal{Q}_{\omega}^s(x^l) \tag{15}$$

$$= \sum_{\omega \in \Omega} p_{\omega}^l \left\{ \pi_{\omega,0}^*(x^l)^T (r_{\omega} - T_{\omega} x^l) \right\}. \tag{16}$$

Since $\mathcal{Q}_{\omega}^s(x)$ is convex in x , $\mathbb{E}_{\xi_P}[\mathcal{Q}_{\omega}(x)] = \sum_{\omega \in \Omega} p_{\omega} \mathcal{Q}_{\omega}^s(x)$ is convex for a given probability distribution $P \in \mathfrak{P}$ or $\{p_{\omega}\}_{\omega \in \Omega}$. This implies $\mathcal{Q}^s(x)$ is also a convex function of x because maximum over an arbitrary collection of convex functions is convex. Therefore from the subgradient inequality,

$$\mathcal{Q}^s(x) = \max_{P \in \mathfrak{P}} \left\{ \sum_{\omega \in \Omega} p_{\omega} \mathcal{Q}_{\omega}^s(x) \right\} \geq \sum_{\omega \in \Omega} p_{\omega}^l \left\{ \pi_{\omega,0}^*(x^l)^T (r_{\omega} - T_{\omega} x) \right\},$$

and hence from (14), it is clear that

$$\sum_{\omega \in \Omega} p_{\omega}^l \left\{ \pi_{\omega,0}^*(x^l)^T (r_{\omega} - T_{\omega} x) \right\} \leq \theta. \tag{17}$$

Inequalities (17) are same as the optimality cuts $\text{OCS}(\pi_{\omega,0}^*(x^l), \{p_{\omega}^l\}_{\omega \in \Omega})$ and are the supporting hyperplanes for $\mathcal{Q}^s(x)$. Also, in (12)-(14), θ is unrestricted except for $\theta \geq \mathcal{Q}^s(x)$.

Let (x^{l+1}, θ^{l+1}) be the optimal solution obtained after solving the master problem \mathcal{M}_l (11) in Step 17 of Algorithm 1. Then, either of the following two cases will occur:

Case I [$\theta^{l+1} \geq \mathcal{Q}^s(x^{l+1})$]: Observe that (x^{l+1}, θ^{l+1}) is a feasible solution to the problem defined by (12)-(14) because $x^{l+1} \in X_{LP}$ and $\theta^{l+1} \geq \mathcal{Q}^s(x^{l+1})$. Interestingly, it is also the optimal solution of the problem because if there exists a solution $(x^*, \theta^*) \neq (x^{l+1}, \theta^{l+1})$ such that $c^T x^* + \theta^* < c^T x^{l+1} + \theta^{l+1}$ then (x^*, θ^*) must be the optimal solution to the master problem \mathcal{M}_l . Also,

$$LB = c^T x^{l+1} + \theta^{l+1} \geq c^T x^{l+1} + \sum_{\omega \in \Omega} p_{\omega}^{l+1} \mathcal{Q}_{\omega}^s(x^{l+1}) = UB.$$

The last inequality satisfies the termination condition in Line 10 and hence, Algorithm 1 terminates whenever this case occurs and returns the optimal solution.

Case II [$\theta^{l+1} < \mathcal{Q}^s(x^{l+1})$]: Clearly (x^{l+1}, θ^{l+1}) is not a feasible solution to the problem (12)-(14) because constraint (14) is violated by it. Also,

$$LB = c^T x^{l+1} + \theta^{l+1} < c^T x^{l+1} + \sum_{\omega \in \Omega} p_{\omega}^{l+1} \mathcal{Q}_{\omega}^s(x^{l+1}) = UB.$$

Since the termination condition in Line 10 is not satisfied, we derive optimality cut in Line 15 to cut-off the point (x^{l+1}, θ^{l+1}) .

Next we show that Case II will occur finite number of times in Algorithm 1, which implies that the "while" loop will terminate after finite iterations and will return the optimal solution. In Case II, $\theta^{l+1} < \mathcal{Q}^s(x^{l+1})$; it means that none of the previously derived optimality cuts adequately impose $\mathcal{Q}^s(x) \leq \theta$ at (x^{l+1}, θ^{l+1}) . Therefore probability distribution $\{p_\omega^{l+1}\}_{\omega \in \Omega}$ and a new set of dual multipliers $\{\pi_{\omega,0}^*(x^{l+1})\}_{\omega \in \Omega}$ are obtained in Lines 7 and 14, respectively, to derive an appropriate optimality cut $\text{OCS}(\pi_{\omega,0}^*(x^{l+1}), \{p_\omega^{l+1}\}_{\omega \in \Omega})$ which cut-off the point (x^{l+1}, θ^{l+1}) . It is important to note that adding the optimality cut to \mathcal{M}_l forces $\theta \geq \mathcal{Q}^s(x^{l+1})$. Since each dual multiplier $\pi_{\omega,0}^*(x)$ corresponds to one of the finitely many different basis, there are finitely number of the set of dual multiplier. In addition, because of the assumption that the ambiguity set \mathfrak{B} is defined by a polytope with finite number of extreme points, there are finite number of possible solutions $\{p_\omega^{l+1}\}_{\omega \in \Omega}$ to the distribution separation algorithm for x^{l+1} . Hence, there are finite number of optimality cuts. Therefore, after finite iterations with Case II, Case I occurs and Algorithm 1 terminates. This completes the proof. \square

3. Distributionally robust integer L-shaped algorithm for TSDR-MBPs

In this section, we further generalize the distributionally robust L-shaped algorithm (Algorithm 1) for solving TSDR-MBP using parametric cuts. We refer to this generalized algorithm as the distributionally robust integer L-shaped algorithm. The pseudocode of our algorithm is given by Algorithm 2. Because of the presence of the binary variables in both stages, we re-define subproblem $\mathcal{S}_\omega(x^l)$, master problem \mathcal{M}_l , and optimality cuts for Algorithm 2.

First, we define subproblem $\mathcal{S}_\omega(x)$, for $x \in X$ and $\omega \in \Omega$, as follows:

$$\mathcal{Q}_\omega^s(x) := \min g_\omega^T y_\omega \quad (18a)$$

$$s.t. \quad W_\omega y_\omega \geq r_\omega - T_\omega x \quad (18b)$$

$$\alpha_\omega^t y_\omega \geq \beta_\omega^t - \psi_\omega^t x, \quad t = 1, \dots, \tau_\omega \quad (18c)$$

$$y_\omega \in \mathbb{R}_+^q, \quad (18d)$$

where $\alpha_\omega^t \in \mathbb{Q}^q$, $\psi_\omega^t \in \mathbb{Q}^p$, and $\beta_\omega^t \in \mathbb{Q}$ are the coefficients of y_ω , coefficients of x , and the constant term in the right hand side, respectively, of a parametric inequality. We will discuss how these parametric inequalities (more specifically, parametric lift-and-project cuts) are developed in succession for mixed binary second stage problems in Section 3.1. Also, for a given $x \in X$ and $\omega \in \Omega$, the optimal dual multipliers obtained by solving $\mathcal{S}_\omega(x)$ are defined by $\pi_\omega^*(x) = (\pi_{\omega,0}^*(x), \pi_{\omega,1}^*(x), \dots, \pi_{\omega,\tau_\omega}^*(x))^T$ where $\pi_{\omega,0}^*(x) \in \mathbb{R}^{m_2}$ corresponds to constraints (18b) and $\pi_{\omega,t}^*(x) \in \mathbb{R}$ corresponds to constraint (18c) for $t = 1, \dots, \tau(\omega)$. In contrast to the previous section, we derive a lower bounding approximation of the first stage problem (1), referred as the master problem \mathcal{M}_l :

$$\min \{c^T x + \theta : x \in X \text{ and } \text{OCS}(\pi_\omega^*(x^k), \{p_\omega^k\}_{\omega \in \Omega}) \text{ holds, for } k = 1, \dots, l\}, \quad (19)$$

using the following *optimality cut*, $\text{OCS}(\pi_\omega^*(x), \{p_\omega\}_{\omega \in \Omega})$:

$$\sum_{\omega \in \Omega} p_\omega \left\{ \pi_{\omega,0}^*(x)^T (r_\omega - T_\omega x) + \sum_{t=1}^{\tau_\omega} \pi_{\omega,t}^*(x) (\beta_\omega^t - \psi_\omega^t x) \right\} \leq \theta. \quad (20)$$

Algorithm 2 Distributionally Robust Integer L-shaped Method for TSDR-MBP

```

1: Initialization:  $l \leftarrow 1$ ,  $LB \leftarrow -\infty$ ,  $UB \leftarrow \infty$ ,  $\tau_\omega \leftarrow 0$  for all  $\omega \in \Omega$ . Assume  $x^1 \in X$ .
2: while  $UB - LB > \epsilon$  do ▷  $\epsilon$  is a pre-specified tolerance
3:   for  $\omega \in \Omega$  do
4:     Solve linear program  $\mathcal{S}_\omega(x^l)$ ;
5:      $y_\omega^*(x^l) \leftarrow$  optimal solution;  $\mathcal{Q}_\omega^{s*}(x^l) \leftarrow$  optimal solution value;
6:   end for
7:   if  $y_\omega^*(x^l) \notin \mathcal{K}_\omega(x^l)$  for some  $\omega \in \Omega$  then
8:     for  $\omega \in \Omega$  where  $y_\omega^*(x^l) \notin \mathcal{K}_\omega(x^l)$  do ▷ Add parametric inequalities
9:       Add the parametric cut to  $\mathcal{S}_\omega(x)$  as explained in Section 3.1;
10:      Set  $\tau_\omega \leftarrow \tau_\omega + 1$  and solve linear program  $\mathcal{S}_\omega(x^l)$ ;
11:       $y_\omega^*(x^l) \leftarrow$  optimal solution;  $\mathcal{Q}_\omega^{s*}(x^l) \leftarrow$  optimal solution value;
12:    end for
13:   end if
14:   Solve distribution separation problem using  $\mathcal{Q}_\omega^{s*}(x^l)$ ,  $\omega \in \Omega$ , to get  $\{p_\omega^l\}_{\omega \in \Omega}$ ;
15:   if  $y_\omega^*(x^l) \in \mathcal{K}_\omega(x^l)$  for all  $\omega \in \Omega$  and  $UB > c^T x^l + \sum_{\omega \in \Omega} p_\omega^l \mathcal{Q}_\omega^{s*}(x^l)$  then
16:      $UB \leftarrow c^T x^l + \sum_{\omega \in \Omega} p_\omega^l \mathcal{Q}_\omega^{s*}(x^l)$ ;
17:     if  $UB \leq LB + \epsilon$  then
18:       Go to Line 21;
19:     end if
20:   end if
21:    $\pi_\omega^*(x^l) \leftarrow$  optimal dual multipliers obtained by solving  $\mathcal{S}_\omega(x^l)$  for all  $\omega \in \Omega$ ;
22:   Derive optimality cut  $\text{OCS}(\pi_\omega^*(x^l), \{p_\omega^l\}_{\omega \in \Omega})$  using (10);
23:   Add  $\text{OCS}(\pi_\omega^*(x^l), \{p_\omega^l\}_{\omega \in \Omega})$  to  $\mathcal{M}_{l-1}$  to get  $\mathcal{M}_l$ ;
24:   Solve master problem  $\mathcal{M}_l$  as explained in Section 3.1;
25:    $(x^l, \theta^l) \leftarrow$  optimal solution of  $\mathcal{M}_l$ ;  $LB \leftarrow$  optimal solution value of  $\mathcal{M}_l$ ;
26:   Set  $l \leftarrow l + 1$ ;
27: end while
28: return  $(x^l, \{y_\omega^*(x^l)\}_{\omega \in \Omega})$ ,  $UB$ 

```

Notice that in Algorithm 2, some steps are similar to the steps of Algorithm 1, except Lines 1, 7-13, 15, and 24. However for the sake of readers' convenience and the completeness of this section, we explain all the steps of Algorithm 2 which works as follows: First, we initialize Algorithm 2 by setting lower bound LB to negative infinity, upper bound UB to positive infinity, iteration counter l to 1, number of parametric inequalities τ_ω for all $\omega \in \Omega$ to zero, and by selecting a first stage feasible solution $x^1 \in X$ (Line 1). At each iteration $l \geq 1$, we solve linear programs $\mathcal{S}_\omega(x^l)$ for all $\omega \in \Omega$ and store the corresponding optimal solution $y_\omega^*(x^l)$ and the optimal solution value $\mathcal{Q}_\omega^s(x^l) := g_\omega^T y_\omega^*(x^l)$ for each $\omega \in \Omega$ (Lines 3-5). Now, for each $\omega \in \Omega$ with $y_\omega^*(x^l) \notin \mathcal{K}_\omega(x^l)$, we develop parametric lift-and-project cut for mixed binary second stage programs (explained in Section 3.2), add it to $\mathcal{S}_\omega(x)$, resolve the updated subproblem $\mathcal{S}_\omega(x)$ by fixing $x = x^l$, and obtain its optimal solution $y_\omega^*(x^l)$ along with optimal solution value (Lines 8-12). Next, we solve the distribution separation problem associated to the ambiguity set \mathfrak{P} and obtain the optimal solution, i.e. $\{p_\omega^l\}_{\omega \in \Omega}$ (Line 14). Interestingly, in case $y_\omega^*(x^l) \in \mathcal{K}_\omega(x^l)$ for all $\omega \in \Omega$, we have a feasible solution $(x^l, y_{\omega_1}^*(x^l), \dots, y_{\omega_{|\Omega|}}^*(x^l))$ for the original problem. Therefore, using $\{p_\omega^l\}_{\omega \in \Omega}$, we update UB if the solution value corresponding to thus obtained feasible solution is smaller than the existing upper bound (Lines 15-16). We also utilize the stored information and optimal dual multipliers (Line 21) to derive optimality cut $\text{OCS}(\pi_\omega^*(x^l), \{p_\omega^l\}_{\omega \in \Omega})$ using (10) and add this cut to the master

problem \mathcal{M}_{l-1} to get an augmented master problem \mathcal{M}_l (Lines 22-23). We solve the master problem \mathcal{M}_l as explained in Section 3.1 and use thus obtained optimal solution value to update lower bound LB (Lines 24-25) as it is lower bounding approximation of (1). Let (x^l, θ^l) be the optimal solution of \mathcal{M}_l . It is important to note that the lower bound LB is a non-increasing with respect to the iterations. This is because \mathcal{M}_{l-1} is a relaxation of \mathcal{M}_l for each $l \geq 1$. Therefore, after every iteration the difference between the bounds, $UB - LB$, either decreases or remains same as in the previous iteration. We terminate our algorithm when this difference becomes zero, i.e., $UB = LB$, or reaches a pre-specified tolerance ϵ (Line 2 or Lines 17-19), and return the optimal solution $(x^l, \{y(\omega, x^l)\}_{\omega \in \Omega})$ and the optimal solution value UB (Line 28).

In the following sections, we customize well-known cutting plane algorithms to solve master problem at iteration $l \geq 1$, i.e. \mathcal{M}_l , and subproblems for a given $x \in X$, i.e. $\mathcal{S}_\omega(x)$ for all $\omega \in \Omega$. Also in Section 3.3, we investigate the conditions under which Algorithm 2 solves TSDR-MBP in finitely many iterations.

3.1 Solving master problem using cutting planes

Notice that \mathcal{M}_l is a mixed binary program for TSDR-MBP where $\theta \in \mathbb{R}$ is the continuous variable. Balas et al. [2] develop a specialized lift-and-project algorithm to solve mixed binary programs which terminates after a finite number of iterations (see Page 227 of [11]). Therefore, we use their algorithm to solve master problem associated of TSDR-MBP.

3.2 Solving subproblems using parametric cuts

Next, we showcase how to develop “parametric cuts” using simplex tableau for solving the subproblems. More specifically, we develop parametric lift-and-project cuts for TSDR-MBP. It is important to note that given $(\bar{x}, \bar{\omega}) \in (X, \Omega)$, a parametric cut for $\mathcal{S}_{\bar{\omega}}(\bar{x})$ is developed by first deriving a valid inequality for \mathcal{E} which has the form $\sum_{i=1}^p \bar{\psi}_i x_i + \sum_{j=1}^q \bar{\alpha}_{\bar{\omega}, j} y_{\bar{\omega}, j} \geq \bar{\beta}_{\bar{\omega}}$ where $\bar{\psi} \in \mathbb{Q}^p$, $\bar{\alpha}_{\bar{\omega}} \in \mathbb{Q}^q$, and $\bar{\beta}_{\bar{\omega}} \in \mathbb{Q}$, and then projecting this inequality on to $y_{\bar{\omega}}$ space by setting $x = \bar{x}$. As a result, the same valid inequality for \mathcal{E} can be used to derive cuts for $\mathcal{S}_{\bar{\omega}}(x)$ for all values of parameter $x \in X$. To do so, we utilize lift-and-project cuts of Balas et al. [2]. Assume that the slack variables of both stages are also incorporated in the matrices A , W_ω , and T_ω . Now let the basis matrix associated to the optimal solution of the linear programming relaxation of first stage program be denoted by $A_B := (a_{B_1}, \dots, a_{B_m})$ where a_j is the j th column of A . Likewise, we define $T_{\omega, B} := (t_{\omega, B_1}, \dots, t_{\omega, B_m})$ where $t_{\omega, j}$ is the j th column of T_ω . Also, for given $x^l \in X$ and $\omega \in \Omega$, let $W_{\omega, \bar{B}}$ be the basis matrix associated to the optimal solution of the linear programming relaxation of the second stage programs with $x = x^l$. Then the lift-and-project cuts generated from any row of

$$G \begin{bmatrix} A \\ T_{\omega_1} \\ T_{\omega_2} \\ \vdots \\ T_{\omega_{|\Omega|}} \end{bmatrix} x + G \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ W_{\omega_1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & W_{\omega_2} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & W_{\omega_{|\Omega|}} \end{bmatrix} \begin{bmatrix} y_{\omega_1} \\ y_{\omega_2} \\ y_{\omega_3} \\ \vdots \\ y_{\omega_{|\Omega|}} \end{bmatrix} = G \begin{bmatrix} b \\ r_{\omega_1} \\ r_{\omega_2} \\ \vdots \\ r_{\omega_{|\Omega|}} \end{bmatrix} \quad (21)$$

where

$$G := \begin{bmatrix} A_B^{-1} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ -W_{\omega_1, \bar{B}}^{-1} T_{\omega_1, B} A_B^{-1} & W_{\omega_1} & \mathbf{0} & \dots & \mathbf{0} \\ -W_{\omega_2, \bar{B}}^{-1} T_{\omega_2, B} A_B^{-1} & \mathbf{0} & W_{\omega_2} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -W_{\omega_{|\Omega|}, \bar{B}}^{-1} T_{\omega_{|\Omega|}, B} A_B^{-1} & \mathbf{0} & \mathbf{0} & \dots & W_{\omega_{|\Omega|}} \end{bmatrix},$$

is valid for the following extensive formulation TSDR-MBP (refer to [33] for details):

$$\min \quad c^T x + \max_{P \in \mathfrak{P}} \{ \mathbb{E}_{\xi_P} [g_\omega^T y_\omega] \} \quad (22a)$$

$$\text{s.t.} \quad Ax \geq b \quad (22b)$$

$$W_\omega y_\omega \geq r_\omega - T_\omega x \quad \omega \in \Omega \quad (22c)$$

$$x \in \{0, 1\}^p \quad (22d)$$

$$y_\omega \in \{0, 1\}^{q_1} \times \mathbb{R}^{q-q_1} \quad \omega \in \Omega. \quad (22e)$$

We suggest readers to refer to Chapter 5 of [11] for the procedure to generate the aforementioned cut.

3.3 Finite convergence

We present conditions under which Algorithm 2 (DRI L-shaped algorithm) solves TSDR-MBP in finitely many iterations.

Theorem 2 (DRI L-shaped Algorithm). *Algorithm 2 solves the TSDR-MBP to optimality in finitely many iterations if assumptions (1)-(3) defined in Section 1 are satisfied and the distribution separation algorithm associated to the ambiguity set \mathfrak{P} is finitely convergent.*

Proof. Since all the variables in the first stage of TSDR-MBP are binary, the number of first stage feasible solutions $|X|$ is finite. In Algorithm 2, for a given $x^l \in X$, we solve $|\Omega|$ number of linear programs, i.e. $\mathcal{S}_\omega(x^l)$ for all $\omega \in \Omega$, distribution separation problem associated with the ambiguity set \mathfrak{P} , and a master problem \mathcal{M}_l (after adding an optimality cut which requires a linear program to be solved). Notice that the master problem is a mixed binary program for TSDR-MBP and can be solve in finite iterations using specialized lift-and-project algorithm of Balas et al. [2]. Therefore, Lines 3-26 in Algorithm 2 are performed in finite iterations because we assume that the distribution separation algorithm is finitely convergent.

Now we have to ensure that the “while” loop in Line 2 terminates after finite iterations and provide the optimal solution. Notice that if $(x^{l+1}, \theta^{l+1}) \neq (x^l, \theta^l)$, then (x^l, θ^l) will not be visited again in future iterations because the optimality cut generated in Line 15 cuts (x^l, θ^l) . Therefore, in case addition of every optimality cut $\text{OCS}(\pi_\omega^*(x^l), \{p_\omega^l\}_{\omega \in \Omega})$ to \mathcal{M}_l cuts (x^l, θ^l) , Algorithm 1 terminates after finite iterations. This is because $|X|$ is finite and if $x^{l+1} = x^l = x^{l-1}$ and $\theta^l \neq \theta^{l-1}$ then $\theta^{l+1} = \theta^l$ which implies $(x^{l+1}, \theta^{l+1}) = (x^l, \theta^l)$. The second part of the last statement contradicts that the optimality cut cuts off (x^l, θ^l) as considered in this case.

Another possibility is $(x^{l+1}, \theta^{l+1}) = (x^l, \theta^l)$ which can further be divided into two cases. In first case, we assume that $y_\omega^*(x^l) \in \mathcal{K}_\omega(x^l)$ for all $\omega \in \Omega$, therefore the lower bound $LB = c^T x^l + \sum_{\omega \in \Omega} p_\omega^l g_\omega^T y_\omega^*(x^l)$ which is equal to UB as $(x^l, \{y_\omega^*(x^l)\}_{\omega \in \Omega}) \in \mathcal{P}$ is a feasible solution and $\{p_\omega^l\}_{\omega \in \Omega}$

is the optimal solution of the distribution separation problem associated with \mathfrak{B} . This implies that $(x^l, \{y_\omega^*(x^l)\}_{\omega \in \Omega})$ is the optimal solution and hence, in this case the algorithm terminates after returning the optimal solution and optimal objective value UB . For the second case, we assume that for some $\bar{\omega} \in \Omega$, $y_{\bar{\omega}}^*(x^l) \notin \mathcal{K}_{\bar{\omega}}(x^l)$ or more specifically, $y_{\bar{\omega}}^*(x^l) \notin \{0, 1\}^q$. Notice that $x^l \in \text{ver}(X)$ where $\text{ver}(X)$ is the set of vertices of $\text{conv}(X)$ because X is defined by binary variables only. Hence, $x = x^l$ defines a face of $\text{conv}(\mathcal{E})$ and therefore, each extreme point of $\text{conv}(\mathcal{E}) \cap \{x = x^l\}$ have $y_\omega \in \{0, 1\}^q$ for all $\omega \in \Omega$. Using the arguments in the proof of Theorem 3.1 of Balas et al. [2], we can obtain $y_{\bar{\omega}}^*(x^l) \in \{0, 1\}^q$, i.e. $y_{\bar{\omega}}^*(x^l) \in \mathcal{K}_{\bar{\omega}}(x^l)$, by adding a finite number of parametric lift-and-projects cuts (Line 9) to $\mathcal{S}_\omega(x)$. This step can be repeated until $y_\omega^*(x^l) \in \mathcal{K}_\omega(x^l)$ for all $\omega \in \Omega$. As explained above, under such situation, our algorithm terminates and returns the optimal solution after finite number of iterations as $|\Omega|$ is finite. This completes the proof. \square

Instead of solving the master problem (11) to optimality at each iteration, a branch-and-cut approach can also be adopted for a practical implementation. In this approach, similar to the integer L-shaped method [19], a linear programming relaxation of the master problem is solved. The solution thus obtained is used to either generate a feasibility cut (if this current solution violates any of the relaxed constraints), or create new branches/nodes following the usual branch-and-cut procedure. The (globally valid) optimality cut, $\text{OCS}(\pi_\omega^*(x), \{p_\omega\}_{\omega \in \Omega})$, is derived at a node whenever the current solution is also feasible for the original master problem. Interestingly, because of the finiteness of the branch-and-bound approach, it is easy to prove the finite convergence of this algorithm under the conditions mentioned in Theorem 1.

Remark 1. *The distribution separation algorithm associated with the moment matching set \mathfrak{P}_M and Kantorovich set \mathfrak{P}_K are finitely convergent.*

3.4 A cutting surface algorithm for TSDR-MBPs

In this section, we present a cutting surface algorithm to solve TSDR-MBP where the ambiguity set is a set of finite number distributions (denoted by \mathfrak{P}_F) which are not known beforehand. This algorithm utilizes branch-and-cut approach to solve master problems, similar to the integer L-shaped algorithm [19]. Unlike Algorithm 2, we solve the subproblems to optimality using branch-and-cut method instead of sequentially adding parametric lift-and-project cuts. Furthermore, instead of considering the whole set of distributions \mathfrak{P}_F at once, we solve multiple TSDR-MBPs for an increasing sequence of known ambiguity sets, $\mathfrak{P}_F^0, \mathfrak{P}_F^1, \dots$, such that $\mathfrak{P}_F^0 \subset \mathfrak{P}_F^1 \subset \dots \subset \mathfrak{P}_F$, until we reach a subset of \mathfrak{P}_F which contains the optimal probability distribution. Assume that the distributions in the set \mathfrak{P}_F^i are known at step i of our implementation. Then, since each probability distribution is equivalent to a set of probabilities of occurrence of finite number of scenarios, i.e. $\{p_\omega\}_{\omega \in \Omega}$, the distribution separation problem (3) associated to the set \mathfrak{P}_F^i can be solved in finite iterations by explicitly enumerating the distributions in the set.

The cutting surface algorithm works as follows. We start with a known distribution P_0 belonging to \mathfrak{P}_F and set $\mathfrak{P}_F^0 := \{P_0\}$ or obtain it by solving the distribution separation algorithm corresponding to \mathfrak{P} for a first stage feasible solution. Then at step $i \geq 0$, given a known ambiguity set $\mathfrak{P}_F^i \subseteq \mathfrak{P}_F$, similar to the integer L-shaped method, we solve a linear programming relaxation of the master problem. The solution thus obtained is used to either generate a feasibility cut (if this current solution violates any of the relaxed constraints), or create new branches/nodes following the usual branch-and-cut procedure. Whenever the current solution is also feasible for the original master problem, we solve the distribution separation problem associated to the set

\mathfrak{P}_F^i (as mentioned above) to get optimal $\{p_\omega\}_{\omega \in \Omega}$ corresponding to the current first stage feasible solution. Thereafter, we derive the globally valid optimality cut, $\text{OCS}(\cdot, \{p_\omega\}_{\omega \in \Omega})$ and add it to the master problem. We continue the exploration of the nodes until we solve the TSDR-MBP for the ambiguity set \mathfrak{P}_F^i . Then we utilize thus obtained optimal solution to solve the distribution separation problem (3) associated to \mathfrak{P}_F^i and obtain the optimal probability distribution. In case this distribution already belongs to \mathfrak{P}_F^i , we terminate our implementation; otherwise we add it to \mathfrak{P}_F^i to get \mathfrak{P}_F^{i+1} and resolve the TSDR-MBP for the ambiguity set \mathfrak{P}_F^{i+1} using the aforementioned procedure.

Table 1: Details of DRSLP and DRMKP Instances

Instance	Stage I			Stage II				
	#Cons	#BinVar	#ContVar	#Cons	#BinVar	#ContVar	$ \Omega $	RandParam
DRSLP.5.25.50	1	5	0	30	125	5	50	RHS
DRSLP.5.25.100	1	5	0	30	125	5	100	RHS
DRSLP.10.50.50	1	10	0	60	500	10	50	RHS
DRSLP.10.50.100	1	10	0	60	500	10	100	RHS
DRSLP.10.50.500	1	10	0	60	500	10	500	RHS
DRSLP.15.45.5	1	15	0	60	675	15	5	RHS
DRSLP.15.45.10	1	15	0	60	675	15	10	RHS
DRSLP.15.45.15	1	15	0	60	675	15	15	RHS
DRMKP.1	50	240	0	5	120	0	20	OBJ
DRMKP.2	50	240	0	5	120	0	20	OBJ
DRMKP.3	50	240	0	5	120	0	20	OBJ
DRMKP.4	50	240	0	5	120	0	20	OBJ
DRMKP.5	50	240	0	5	120	0	20	OBJ
DRMKP.6	50	240	0	5	120	0	20	OBJ
DRMKP.7	50	240	0	5	120	0	20	OBJ
DRMKP.8	50	240	0	5	120	0	20	OBJ
DRMKP.9	50	240	0	5	120	0	20	OBJ
DRMKP.10	50	240	0	5	120	0	20	OBJ

4. Computational experiments

In this section, we evaluate the performance of the DRI L-shaped algorithm (Algorithm 2) and the cutting surface algorithm (discussed in the previous section) by solving DR versions of stochastic server location problem (SSLP) and stochastic multiple binary knapsack problem (SMKP). The SSLP and SMKP instances are part of the Stochastic Integer Programming Library (SIPLIB) [1] and have only binary variables in the first stage and mixed binary programs in the second stage. Also, the probability of occurrence of each scenario is same in these instances, i.e. $p_\omega = 1/|\Omega|$ for all $\omega \in \Omega$. On the other hand, in our DR version of SSLP and SMKP, referred to as the distributionally robust server location problem (DRSLP) and distributionally robust multiple binary knapsack problem (DRMKP), respectively, the probability distribution belongs to the moment matching set \mathfrak{P}_M (5) or Kantorovich set \mathfrak{P}_K (7). Recall that the distribution separation problem associated

to \mathfrak{P}_M and \mathfrak{P}_K are linear programs and both \mathfrak{P}_M and \mathfrak{P}_K are polytopes with finite number of extreme points. This implies that in these two cases, the ambiguity set is a set of finite number of distributions corresponding to the extreme points which are not known beforehand and the associated distribution separation algorithms are finitely convergent.

4.1 Instance generation

In Table 1, we provide details of the DRSLP and DRMKP instances used for our experiments. In particular, #Cons, #BinVar, and #ContVar denote the number of constraints, binary variables, and continuous variables, respectively, in Stage I and Stage II of the problems. The number of scenarios is given by the column labeled as $|\Omega|$. Also, only the right hand side, i.e. r_ω , are uncertain in DRSLP instances and only the coefficients in the objective function, i.e. g_ω , are uncertain in DRMKP instances. For the sake of uniformity, we are using similar nomenclature for DRSLP as used for SSLP in SIPLIB [1]. Notice that instance DRSLP. $\alpha.\beta.c$ has α number of binary variables in the first stage, $\alpha \times \beta$ binary variables and α non-zero continuous variables in the second stage, and c number of scenarios.

The results of our computational experiments for DRSLP and DRMKP instances where the ambiguity set is defined using Kantorovich-Rubinstein distance (or Wasserstein metric), first two moments, and first three moments are given in Tables 2, 4, and 6 for DRI L-shaped algorithm (or Tables 3, 5, and 7 for the cutting surface algorithm), respectively. In all these tables, we report the optimal solution value z_{opt} , the total number of times the associated distribution separation problem is solved until the termination of our implementation (denoted by #DCs), and the total time taken in seconds (denoted by $T(s)$) to solve the problem instance to optimality, denoted by T . Note that #DCs also provides the number of distribution cuts added until the termination of the cutting surface algorithm.

4.2 Computational results for instances with Kantorovich set as the ambiguity set

We solve DRSLP and DRMKP instances with Kantorovich set (7) as the ambiguity set and present our computational results in Tables 2 and 3. We consider two different values of ϵ , i.e. 5.0 and 10.0, and observe that for each problem instance, z_{opt} remains same for both values of ϵ . It is important to note that the average number of times the distribution separation problem (8) is solved is only 9. Interestingly, the DRI L-shaped algorithm solves all the problem instances to optimality; whereas the cutting surface algorithm failed to solve DRMKP.4, DRMKP.9, and DRMKP.10 to optimality within a time limit of 3 hours. Moreover, the DRI L-shaped algorithm is on average two times faster than the cutting surface algorithm.

4.3 Computational results for instances with moment matching set as the ambiguity set

In this section, we discuss about our computational results for solving DRSLP and DRMKP instances with moment matching set (5) as the ambiguity set. We consider bounds on the first and second moments in Tables 4 and 5, and bounds on the first, second, and third moments in Tables 6 and 7. Furthermore, in each of these tables we consider three different values of confidence interval

(CI), i.e. 80%, 90%, and 95%. Observe that for each problem instance in each table, z_{opt} increases with increase in CI as the size of the ambiguity set increases with increase in CI. In other words, the decision maker becomes conservative as the size of the ambiguity set increases. Because of the similar argument, it is evident that the z_{opt} decreases when additional bound constraints are added in the moment matching set. Comparing z_{opt} for each problem instance with same CI in Tables 4 and 6 (or Tables 5 and 7), we notice that z_{opt} decreases because of the presence of bounds on the third moment.

Next we evaluate the performance of DRI L-shaped algorithm and the cutting surface algorithm in solving DRSLP and DRMKP instances with moment matching set. Interestingly, DRI L-shaped algorithm solves all problem instances to optimality, whereas for instance DRSLP.10.50.500 in Table 7, the cutting surface algorithm failed to terminate within the time limit of 3 hours. On average DRI L-shaped algorithm is 1.4 times faster than the cutting surface algorithm. However, for some instances the latter is faster than the former. In addition, these two algorithms differ in the number of the times distribution separation problems is solved. As expected #DCs is more for DRI L-shaped algorithm in comparison to the cutting surface algorithm.

5. Concluding remarks

We developed decomposition algorithms to solve two-stage distributionally robust mixed binary programs (TSDR-MBPs) and TSDR linear programs (TSDR-LPs) where the random parameters follow the worst-case distribution belonging to a general ambiguity set. As per our knowledge, no work has been done to solve TSDR-MBPs. More specifically, we utilized distribution separation procedure within Bender’s algorithm to solve TSDR-LPs and TSDR-MBPs, and referred to these algorithms as distributionally robust L-shaped algorithm and distributionally robust integer L-shaped algorithm, respectively. Moreover, we provided conditions and the families of ambiguity set for which the foregoing algorithms are finitely convergent. We also presented a cutting surface algorithm to solve TSDR-MBPs where ambiguity set is a set of finite number of distributions which are not known beforehand. Finally, we computationally evaluated the performance of the DRI L-shaped algorithm and the cutting surface algorithm to solve distributionally robust server location problem (DRSLP) and distributionally robust multiple binary knapsack problem (DRMKP) where the ambiguity set is either Kantorovich set or moment matching set (defined by bounds on first two moments or bounds on first three moments). We observed that DRI L-shaped algorithm solved all problems instances to optimality within less than one hour; whereas for 5 DRMKP instances, the cutting surface algorithm did not terminate within a time limit of 3 hours. Also, on average DRI L-shaped algorithm is faster than the cutting surface algorithm.

Table 2: Results for the DRI L-shaped algorithm: Ambiguity set is defined using KR distance

Instance	$\epsilon = 5.0$			$\epsilon = 10.0$		
	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$
DRSLP.5.25.50	14.0	7	2.6	14.0	7	2.5
DRSLP.5.25.100	-40.0	10	7.3	-40.0	10	7.1
DRSLP.10.50.50	-200.0	5	240.0	-200.0	5	236.6
DRSLP.10.50.100	-237.0	16	656.1	-237.0	16	657.2
DRSLP.10.50.500	-159.0	7	1151.6	-159.0	7	1148.9
DRSLP.15.45.5	-252.0	5	288.5	-252.0	5	287.8
DRSLP.15.45.10	-220.0	7	518.4	-220.0	7	515.2
DRSLP.15.45.15	-208.0	11	1203.2	-208.0	11	1202.3
DRMKP.1	9686.0	10	285.4	9686.0	6	101.7
DRMKP.2	9388.0	9	906.0	9388.0	5	591.2
DRMKP.3	8844.0	10	1462.2	8844.0	6	900.5
DRMKP.4	9237.0	23	2695.1	9237.0	14	2485.2
DRMKP.5	10024.0	9	1656.9	10024.0	5	1199.0
DRMKP.6	9515.0	9	257.3	9515.0	5	185.9
DRMKP.7	10003.0	9	434.4	10003.0	5	316.2
DRMKP.8	9427.0	28	4554.3	9427.0	18	3503.7
DRMKP.9	10038.0	10	1090.0	10038.0	6	697.3
DRMKP.10	9082.2	13	4870.0	9082.0	7	3072.8

Table 3: Results for the cutting surface algorithm: Kantorovich set as the ambiguity set

Instance	$\epsilon = 5.0$			$\epsilon = 10.0$		
	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$
DRSLP.5.25.50	14.0	5	4.1	14.0	5	4.6
DRSLP.5.25.100	-40.0	7	21.2	-40.0	7	23.4
DRSLP.10.50.50	-200.0	3	136.2	-200.0	3	134.3
DRSLP.10.50.100	-237.0	7	712.9	-237.0	7	710.7
DRSLP.10.50.500	-159.0	3	611.9	-159.0	3	614.0
DRSLP.15.45.5	-252.0	5	182.2	-252.0	5	181.0
DRSLP.15.45.10	-220.0	5	772.5	-220.0	5	773.7
DRSLP.15.45.15	-208.0	4	584.0	-208.0	4	584.7
DRMKP.1	9686.0	10	243.2	9686.0	9	780.0
DRMKP.2	9388.0	10	878.4	9388.0	6	589.6
DRMKP.3	8844.0	11	1345.1	8844.0	11	5685.0
DRMKP.4	9237.0	14	10800.0	9237.0	10	10800.0
DRMKP.5	10024.0	11	3732.5	10024.0	6	1053.8
DRMKP.6	9515.0	10	225.0	9515.0	6	162.6
DRMKP.7	10003.0	10	386.7	10003.0	6	261.9
DRMKP.8	9426.8	18	2910.4	9426.9	11	2301.2
DRMKP.9	10038.0	18	10800.0	10038.0	10	3739.4
DRMKP.10	9084.0	10	10800.0	9084.0	10	10800.0

Table 4: Results for the DRI L-shaped algorithm: Ambiguity set is defined using bounds on first and second moments

Instance	$CI = 80\%$			$CI = 90\%$			$CI = 95\%$		
	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$
DRSLP.5.25.50	-91.06	109	3.8	-84.92	89	3.3	-78.28	89	3.2
DRSLP.5.25.100	-107.17	97	5.9	-103.07	96	6.4	-98.53	104	8.2
DRSLP.10.50.50	-322.93	388	144.2	-314.41	381	130.9	-304.98	394	138.4
DRSLP.10.50.100	-323.75	465	278.9	-317.70	500	352.6	-310.98	425	302.3
DRSLP.10.50.500	-318.33	500	1603.1	-313.91	484	1482.3	-309.12	489	1395.6
DRSLP.15.45.5	-255.03	32	282.8	-253.88	21	76.7	-253.63	24	144.4
DRSLP.15.45.10	-242.70	99	245.8	-239.47	106	283.2	-236.31	101	272.3
DRSLP.15.45.15	-236.72	569	569.8	-233.54	361	562.8	-230.09	605	1079.7
DRMKP.1	9428.27	13	66.8	9444.49	15	60.4	9462.15	11	45.1
DRMKP.2	9105.96	13	89.0	9125.44	11	94.2	9146.55	9	107.0
DRMKP.3	8631.19	14	73.5	8644.78	16	72.3	8659.79	15	79.6
DRMKP.4	9003.54	27	476.1	9020.71	27	472.3	9039.02	17	564.6
DRMKP.5	9518.85	14	432.0	9538.01	17	247.1	9559.43	11	261.5
DRMKP.6	9214.35	23	440.4	9228.87	18	390.4	9244.70	15	415.1
DRMKP.7	9719.42	17	317.4	9734.60	16	321.0	9750.01	12	312.2
DRMKP.8	9207.61	39	901.1	9224.81	33	1228.9	9242.88	24	662.8
DRMKP.9	9841.14	47	1166.9	9857.03	48	1026.1	9874.73	48	1179.6
DRMKP.10	8871.75	40	3242.8	8886.56	29	2549.9	8901.99	30	2894.4

References

- [1] Shabbir Ahmed, Renan Garcia, Nan Kong, Lewis Ntaimo, Gyana Parija, F. Qiu, and Suvrajeet Sen. Siplib: A stochastic integer programming test problem library. 2015. Available at <http://www.isye.gatech.edu/~sahmed/siplib>.
- [2] Egon Balas, Sebastián Ceria, and Gérard Cornuéjols. A lift-and-project cutting plane algorithm for mixed 0-1 programs. *Mathematical Programming*, 58(1-3):295–324, January 1993.
- [3] Manish Bansal, Kuo-Ling Huang, and Sanjay Mehrotra. Tight second-stage formulations in two-stage stochastic mixed integer programs. under review, 2015. Available at http://www.optimization-online.org/DB_HTML/2015/08/5057.html.
- [4] Manish Bansal and Kiavash Kianfar. Facets for continuous multi-mixing set with general coefficients and bounded integer variables. under review, 2014. Available at http://www.optimization-online.org/DB_HTML/2014/10/4610.html.
- [5] Manish Bansal and Kiavash Kianfar. n -step cycle inequalities: facets for continuous multi-mixing set and strong cuts for multi-module capacitated lot-sizing problem. *Mathematical Programming*, pages 1–32, 2015.

Table 5: Results for the cutting surface algorithm: Ambiguity set is defined using bounds on first and second moments

Instance	$CI = 80\%$			$CI = 90\%$			$CI = 95\%$		
	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$
DRSLP.5.25.50	-91.06	11	21.6	-84.92	12	27.2	-78.28	10	16.5
DRSLP.5.25.100	-107.16	11	29.5	-103.07	13	45.4	-98.53	12	35.6
DRSLP.10.50.50	-322.93	7	305.9	-314.41	8	338.6	-304.98	9	363.0
DRSLP.10.50.100	-323.75	9	559.1	-317.70	12	907.7	-310.98	13	974.5
DRSLP.10.50.500	-318.32	19	5808.5	-313.90	13	3719.5	-309.11	14	4074.8
DRSLP.15.45.5	-255.02	6	120.5	-253.88	6	106.2	-253.63	6	110.5
DRSLP.15.45.10	-242.69	6	257.9	-239.47	6	183.8	-236.30	6	225.9
DRSLP.15.45.15	-236.72	8	351.7	-233.53	6	408.9	-230.08	7	1000.9
DRMKP.1	9428.26	7	188.3	9444.48	7	117.1	9462.14	7	130.0
DRMKP.2	9105.96	6	255.5	9125.44	7	267.2	9146.55	6	261.3
DRMKP.3	8631.18	6	221.6	8644.77	6	277.0	8659.79	5	75.9
DRMKP.4	9003.53	5	232.8	9020.70	5	242.6	9039.01	6	712.8
DRMKP.5	9518.84	8	1036.4	9538.00	7	840.1	9559.43	6	508.8
DRMKP.6	9214.34	6	410.2	9228.87	5	541.4	9244.70	7	516.1
DRMKP.7	9719.41	5	849.1	9734.59	5	813.2	9750.00	6	805.2
DRMKP.8	9207.61	7	1253.8	9224.81	6	1383.5	9242.87	7	1166.7
DRMKP.9	9841.14	6	1144.6	9857.03	6	1197.1	9874.73	6	1243.9
DRMKP.10	8871.74	6	1212.1	8886.55	7	2773.5	8901.99	7	2516.1

- [6] Aharon Ben-Tal, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust Solutions of Optimization Problems Affected by Uncertain Probabilities. *Management Science*, 59(2):341–357, November 2012.
- [7] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, December 1962.
- [8] D. Bertsimas and I. Popescu. Optimal inequalities in probability theory: A convex optimization approach. *SIAM Journal on Optimization*, 15(3):780–804, 2005.
- [9] Dimitris Bertsimas, Xuan Vinh Doan, Karthik Natarajan, and Chung-Piaw Teo. Models for minimax stochastic linear optimization problems with risk aversion. *Mathematics of Operations Research*, 35(3):580–602, 2010.
- [10] G. Calafiore. Ambiguous Risk Measures and Optimal Robust Portfolios. *SIAM Journal on Optimization*, 18(3):853–877, January 2007.
- [11] Michele Conforti, Gérard Cornuéjols, and Giacomo Zambelli. *Integer Programming*, volume 271 of *Graduate Texts in Mathematics*. Springer International Publishing, Cham, 2014.
- [12] Erick Delage and Yinyu Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58:595–612, 2010.
- [13] J. Dupacová. The minimax approach to stochastic programming and an illustrative application. *Stochastics*, 20:73–88, 1987.

Table 6: Results for the DRI L-shaped algorithm: Ambiguity set is defined using bounds on first, second, and third moments

Instance	$CI = 80\%$			$CI = 90\%$			$CI = 95\%$		
	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$
DRSLP.5.25.50	-93.39	99	4.2	-87.62	98	3.9	-81.27	93	4.1
DRSLP.5.25.100	-107.73	108	10.1	-103.71	116	11.3	-99.20	104	9.8
DRSLP.10.50.50	-332.79	385	162.3	-326.26	450	187.5	-318.92	438	157.4
DRSLP.10.50.100	-325.03	472	413.6	-319.05	450	377.2	-312.33	437	382.6
DRSLP.10.50.500	-325.03	499	7234.2	-320.25	533	7653.6	-314.98	482	6060.0
DRSLP.15.45.5	-255.03	30	272.6	-253.88	22	190.0	-253.63	22	121.2
DRSLP.15.45.10	-242.70	119	743.7	-239.47	98	241.1	-236.34	86	2818.2
DRSLP.15.45.15	-237.05	347	650.1	-233.93	377	622.9	-230.48	518	1306.3
DRMKP.1	9418.71	13	289.3	9434.83	13	253.6	9451.86	13	333.3
DRMKP.2	9093.42	15	319.1	9111.53	10	339.3	9129.61	15	407.3
DRMKP.3	8619.42	19	389.4	8630.87	18	341.8	8643.47	16	357.3
DRMKP.4	8990.40	31	661.9	9005.47	30	570.7	9022.28	30	656.2
DRMKP.5	9503.67	15	510.1	9519.97	14	581.6	9537.57	13	539.2
DRMKP.6	9204.78	18	486.0	9217.42	24	536.1	9231.59	17	414.2
DRMKP.7	9709.79	13	684.0	9722.50	18	588.2	9734.91	18	717.1
DRMKP.8	9199.72	40	1555.2	9215.69	38	1226.4	9232.25	37	1389.8
DRMKP.9	9830.45	50	1298.0	9844.10	46	1569.0	9859.03	35	1437.0
DRMKP.10	8864.22	49	4547.7	8878.22	28	3107.9	8892.18	33	3588.3

- [14] E. Erdogan and G. Iyengar. Ambiguous chance constrained problems and robust optimization. *Mathematical Programming*, 107(1-2):37–61, 2006.
- [15] Dinakar Gade, Simge Küçükyavuz, and Suvrajeet Sen. Decomposition algorithms with parametric gomory cuts for two-stage stochastic integer programs. *Mathematical Programming*, pages 1–26, 2012.
- [16] Grani A. Hanasusanto and Daniel Kuhn. Conic programming reformulations of two-stage distributionally robust linear programs over wasserstein balls. under review, 2016. Available at <https://arxiv.org/abs/1609.07505>.
- [17] Grani A. Hanasusanto, Vladimir Roitch, Daniel Kuhn, and Wolfram Wiesemann. A distributionally robust perspective on uncertainty quantification and chance constrained programming. *Mathematical Programming*, 151(1):35–62, 2015.
- [18] Ruiwei Jiang and Yongpei Guan. Risk-averse two-stage stochastic program with distributional ambiguity. 2015. Available at http://www.optimization-online.org/DB_FILE/2015/05/4908.pdf.
- [19] Gilbert Laporte and François V. Louveaux. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, 13(3):133–142, April 1993.
- [20] David K. Love and Guzin Bayraksan. Phi-divergence constrained ambiguous stochastic

Table 7: Results for the cutting surface algorithm: Ambiguity set is defined using bounds on first, second, and third moments

Instance	$CI = 80\%$			$CI = 90\%$			$CI = 95\%$		
	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$	z_{opt}	#DCs	$T(s)$
DRSLP.5.25.50	-93.39	12	24.7	-87.62	12	22.9	-81.27	11	20.6
DRSLP.5.25.100	-107.73	11	32.8	-103.71	7	11.6	-99.20	12	42.1
DRSLP.10.50.50	-332.79	7	248.4	-326.26	9	362.7	-318.92	13	537.2
DRSLP.10.50.100	-325.03	8	466.8	-319.05	8	512.1	-312.33	12	920.2
DRSLP.10.50.500	-325.03	23	10191	-320.25	23	8533	-314.98	20	7341
DRSLP.15.45.5	-255.03	6	63.2	-253.88	6	106.8	-253.63	6	110.9
DRSLP.15.45.10	-242.70	6	175.4	-239.47	6	195.9	-236.34	6	482.2
DRSLP.15.45.15	-237.05	8	588.7	-233.93	7	533.7	-230.48	6	600.4
DRMKP.1	9418.71	5	292.5	9434.83	7	294.7	9451.86	7	413.5
DRMKP.2	9093.42	6	622.5	9111.53	5	316.2	9129.61	8	697.2
DRMKP.3	8619.42	6	513.6	8630.87	7	487.3	8643.47	5	476.3
DRMKP.4	8990.40	6	724.0	9005.47	7	872.2	9022.28	5	786.1
DRMKP.5	9503.67	8	1188.1	9519.97	5	1073.0	9537.57	6	1106.6
DRMKP.6	9204.78	7	1129.1	9217.42	7	753.5	9231.59	6	723.6
DRMKP.7	9709.79	6	1397.0	9722.50	7	1059.1	9734.91	7	1125.3
DRMKP.8	9199.72	6	1610.7	9215.69	6	1368.7	9232.25	7	2007.8
DRMKP.9	9830.45	6	1621.4	9844.10	6	2050.7	9859.03	6	2146.0
DRMKP.10	8864.22	6	3165.9	8878.22	6	3345.6	8892.18	6	1640.9

programs for data-driven optimization. under review, 2016. Available at http://www.optimization-online.org/DB_HTML/2016/03/5350.html.

- [21] Sanjay Mehrotra and Dávid Papp. A cutting surface algorithm for semi-infinite convex programming with an application to moment robust optimization. *SIAM J. on Optimization*, 24(4):1670–1697, 2014.
- [22] Sanjay Mehrotra and He Zhang. Models and algorithms for distributionally robust least squares problems. *Mathematical Programming*, 148(1–2):123–141, 2014.
- [23] G. Pflug and D. Wozabal. Ambiguity in portfolio selection. *Quantitative Finance*, 7(4):435–442,, 2007.
- [24] Georg. Pflug, Alois. Pichlera, and David. Wozabalb. The 1/n investment strategy is optimal under high model ambiguity. *Journal of Banking and Finance*, 36(2):410–417, 2012.
- [25] A. Prékopa. *Stochastic Programming*. Kluwer Academic, 1995.
- [26] H. Scarf. A min-max solution of an inventory problem. In *Studies in the Mathematical Theory of Inventory and Production*, chapter 12, pages 201–209. 1958.
- [27] Alexander Shapiro and Shabbir Ahmed. On a class of minimax stochastic programs. *SIAM Journal on Optimization*, 14(4):1237–1249, 2004.

- [28] Alexander Shapiro and Anton Kleywegt. Minimax analysis of stochastic problems. *Optimization Methods and Software*, 17(3):523–542, 2002.
- [29] Hanif D Sherali and Barbara MP Fraticelli. A modification of benders’ decomposition algorithm for discrete subproblems: An approach for stochastic programs with integer recourse. *Journal of Global Optimization*, 22(1-4):319–342, 2002.
- [30] Zizhuo Wang, Peter W. Glynn, and Yinyu Ye. Likelihood robust optimization for data-driven problems. *Computational Management Science*, 13(2):241–261, April 2016.
- [31] David. Wozabal. A framework for optimization under ambiguity. *Annals of Operations Research*, 193(1):21–47, 2012.
- [32] İhsan Yanıkoğlu and Dick den Hertog. Safe Approximations of Ambiguous Chance Constraints Using Historical Data. *INFORMS Journal on Computing*, 25(4):666–681, November 2012.
- [33] M. Zhang and S. Kücükyavuz. Finitely convergent decomposition algorithms for two-stage stochastic pure integer programs. *SIAM Journal on Optimization*, 24(4):1933–1951, January 2014.
- [34] Chaoyue Zhao and Yongpei Guan. Data-driven risk-averse two-stage stochastic program with ζ -structure probability metrics. under review, 2015. Available at http://www.optimization-online.org/DB_HTML/2015/07/5014.html.