

Nonanticipative duality, relaxations, and formulations for chance-constrained stochastic programs

Shabbir Ahmed[†], James Luedtke[‡], Yongjia Song[§], Weijun Xie[†]

[†]Georgia Institute of Technology, Atlanta, GA

[‡]University of Wisconsin-Madison, Madison, WI

[§]Virginia Commonwealth University, Richmond, VA

January 23, 2016

Abstract

We propose two new Lagrangian dual problems for chance-constrained stochastic programs based on relaxing nonanticipativity constraints. We compare the strength of the proposed dual bounds and demonstrate that they are superior to the bound obtained from the continuous relaxation of a standard mixed-integer programming (MIP) formulation. For a given dual solution, the associated Lagrangian relaxation bounds can be calculated by solving a set of single scenario subproblems and then solving a single knapsack problem. We also derive two new primal MIP formulations and demonstrate that for chance-constrained linear programs, the continuous relaxations of these formulations yield bounds equal to the proposed dual bounds. We propose a new heuristic method and two new exact algorithms based on these duals and formulations. The first exact algorithm applies to chance-constrained binary programs, and uses either of the proposed dual bounds in concert with cuts that eliminate solutions found by the subproblems. The second exact method is a branch-and-cut algorithm for solving either of the primal formulations. Our computational results indicate that the proposed dual bounds and heuristic solutions can be obtained efficiently, and the gaps between the best dual bounds and the heuristic solutions are small.

1 Introduction

We consider chance-constrained stochastic programs (CCSPs) of the form

$$v^* = \min \{f(x) : x \in S, \mathbb{P}[x \in X(\xi)] \geq 1 - \epsilon\}. \quad (1)$$

We make the following assumptions throughout this paper: (i) the random vector ξ has a finite support, i.e., $\Xi = \{\xi^1, \dots, \xi^N\}$, where each $i \in \mathcal{N} := \{1, \dots, N\}$ is referred to as a scenario; (ii) $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a linear function; i.e., $f(x) = c^\top x$ (if $f(x)$ is nonlinear, we can introduce a new variable y , add a new constraint $f(x) \leq y$ into S and change the objective to minimize y); (iii) $X(\xi^i) \subseteq S$ for each $i \in \mathcal{N}$; otherwise, we can replace $X(\xi)$ by $X(\xi) \cap S$ which yields an equivalent problem; (iv) the feasible region is nonempty. We can then rewrite (1) as

$$v^* = \min \left\{ c^\top x : x \in S, \sum_{i \in \mathcal{N}} p_i \mathbb{I}(x \in X^i) \geq 1 - \epsilon \right\}, \quad (2)$$

where S is a mixed-integer set (i.e., $S \subseteq \mathbb{R}^{n-r} \times \mathbb{Z}^r$ with $0 \leq r \leq n$), $X^i := X(\xi^i)$, $\mathbb{I}(\cdot)$ is the indicator function and p_i is the probability mass associated with scenario i . Using a binary variable z_i to model the indicator function for each scenario i , we reformulate (1) as

$$v^* = \min \{c^\top x : x \in S, z_i = \mathbb{I}(x \in X^i), i \in \mathcal{N}, z \in Z\}, \quad (3)$$

where

$$Z := \left\{ z \in \{0, 1\}^N : \sum_{i \in \mathcal{N}} p_i z_i \geq 1 - \epsilon \right\}.$$

We assume throughout this paper that S , and therefore X^i , are compact sets for all $i \in \mathcal{N}$. Our results can be directly generalized to the unbounded case when the sets $\text{conv}(X^i)$ for all $i \in \mathcal{N}$ share the same recession cone.

CCSPs were first studied in [11, 12]. Subsequently, significant research has been carried out on structural and algorithmic issues for CCSPs under various distributional settings [15, 16, 20, 30, 39], as well as on sampled approximations [6, 7, 19, 26, 29]. A CCSP with a finite number of scenarios can be formulated as a large-scale mixed-integer program (MIP), by introducing a binary variable for each individual scenario and adding big-M inequalities into the formulation. However, the natural MIP formulation based on big-M inequalities often has a weak linear programming relaxation [8]. Motivated by this drawback, there has been significant recent works investigating the use of MIP techniques for solving CCSPs having a finite number of scenarios. In particular, the mixing structure of CCSPs has been studied in [21, 25, 27]. By exploring special combinatorial structures of CCSPs, [22, 37, 38] introduced problem formulations without the binary scenario variables. Chance-constrained formulations have also been proposed for two-stage [24] and multi-stage settings [41].

In this paper we introduce two new Lagrangian relaxation techniques for obtaining lower bounds for the CCSPs (3). Inspired by the associated Lagrangian dual problems, we also introduce new MIP formulations of (3) that yield stronger relaxations than existing formulations. The Lagrangian relaxations we construct are obtained by variable splitting: creating multiple copies of the variables x , which are constrained to be equal to each other, and then constructing a relaxation in which these “nonanticipativity constraints” are relaxed. In stochastic programming this technique is known as dual decomposition, and was firstly introduced by [34], and used in [9] for obtaining strong relaxations of two-stage stochastic integer programs. See also [17, 35] for more results on dual decomposition in the two-stage stochastic programming. Some Lagrangian relaxation approaches for CCSPs have been investigated recently. In [3] the original problem constraints defining X^i are relaxed within an augmented Lagrangian relaxation framework. In [16], auxiliary variables are introduced to separate the deterministic and probabilistic constraints, and the deterministic constraints are relaxed. The work in [39] exploits decomposable structure of the deterministic constraints for a specific class of CCSPs. Finally, in [40], both the nonanticipativity constraints and the knapsack constraint $\sum_{i \in \mathcal{N}} p_i z_i \geq 1 - \epsilon$ are relaxed. In contrast, we do not relax the knapsack constraint or the original problem constraints and directly work on the original formulation (3), leading to relaxation bounds that are better than existing alternatives. Somewhat surprisingly, even though the knapsack constraint – which links scenarios together – is not relaxed, the majority of the work required to solve our proposed Lagrangian relaxation problems can be still decomposed by scenarios.

The remainder of the paper is organized as follows. In Section 2, we discuss three valid lower bounds, which can be obtained by continuous relaxation, quantile bounding and scenario grouping. We then provide two new Lagrangian dual formulations based on relaxing the nonanticipativity constraints in Section 3. In Section 4, we compare these bounds with the basic lower bounds introduced in Section 2. We derive new primal formulations in Section 5 that are related to the dual formulations from Section 3. In Section 6 we present a heuristic and two new exact algorithms to solve CCSPs. Finally, we devote Section 7 to computational illustration of the lower bounds and performances of the proposed algorithms and make concluding remarks in Section 8.

2 Basic lower bounds

We first present three different lower bounds for the CCSPs (3). The first two are known results while the third bound presented in Section 2.3 is new.

2.1 Continuous relaxation

Assume that for each scenario i , X^i is described by an inequality system $G_i(x) \leq 0$ with a mapping $G_i : S \mapsto \mathbb{R}^{m_i}$. Problem (3) can then be modeled as the following MIP:

$$v^* = \min_{x,z} \{c^\top x : x \in S, G_i(x) \leq M_i(1 - z_i), \forall i \in \mathcal{N}, z \in Z\}, \quad (4)$$

where M_i is a vector of big- M parameters such that M_{ij} gives a valid upper bound of $G_{ij}(x)$ for all feasible x in (2), that is, $M_{ij} \geq \max\{G_{ij}(x) : x \in S, \sum_{i \in \mathcal{N}} p_i \mathbb{I}(x \in X^i) \geq 1 - \epsilon\}$ for each $j = 1, 2, \dots, m_i$. It is impractical to compute the tightest possible upper bound, since it involves solving another CCSP. Therefore, computationally tractable techniques for deriving upper bounds have been investigated. For example, one may begin by choosing $M_{ij} \geq \sup\{G_{ij}(x) : x \in S \cap X^i\}$ for all $j = 1, 2, \dots, m_i$. This simple bound could then be strengthened using a coefficient strengthening that considers more of the feasible region of (4) as in [31, 38] (see Section 7). We assume that whenever a strengthened big- M parameter $M'_{ij} < M_{ij}$ is obtained, we include the valid inequality $G_{ij}(x) \leq M'_{ij}$ in S , so that $S \subseteq \{x \in \mathbb{R}^n : G_i(x) \leq M'_i, \forall i \in \mathcal{N}\}$. We define $v^C(M)$ to be the optimal objective value of the relaxation of (4) in which the integrality constraints on the z variables are relaxed (but the integrality constraints on x , if any, are not relaxed). We denote by $\underline{v}^C(M)$ the relaxation obtained by also relaxing integrality constraints on x . Clearly, we have

$$\underline{v}^C(M) \leq v^C(M) \leq v^*.$$

2.2 Quantile bound

Another lower bound for problem (3) is the so-called quantile bound [38]. We first calculate for each $i \in \mathcal{N}$,

$$\eta_i = \min_x \{c^\top x : x \in X^i\}.$$

We then sort these values to obtain a permutation σ of \mathcal{N} such that $\eta_{\sigma_1} \geq \dots \geq \eta_{\sigma_N}$. The quantile bound is then defined as $v^Q = \eta_{\sigma_q}$, where $q = \min\{k \in \mathcal{N} : \sum_{i=1}^k p_{\sigma_i} > \epsilon\}$. Then clearly

$$v^Q \leq v^*$$

because at least one scenario in the scenario set $\{\sigma_1, \dots, \sigma_q\}$ must be satisfied in a feasible solution.

2.3 Scenario grouping based lower bound

We partition the scenarios \mathcal{N} into $K < N$ disjoint subsets $\mathcal{N}_j, j \in \mathcal{K} = \{1, \dots, K\}$ where $\sum_{j=1}^K |\mathcal{N}_j| = N$. For each $j \in \mathcal{K}$, we define $\tilde{z}_j = 1$ if $z_i = 1$ for all the scenarios $i \in \mathcal{N}_j$ and 0 otherwise (i.e., $\tilde{z}_j = \min\{z_i : i \in \mathcal{N}_j\}$). For each $j \in \mathcal{K}$, we define $q_j = \min_{i \in \mathcal{N}_j} p_i$. We then define the following scenario grouping model:

$$v^G = \min \left\{ c^\top x : x \in S, \tilde{z}_j = \mathbb{I}\left(x \in \bigcap_{i \in \mathcal{N}_j} X^i\right), j \in \mathcal{K}, \sum_{j \in \mathcal{K}} q_j \tilde{z}_j \geq \sum_{j \in \mathcal{K}} q_j - \epsilon \right\}. \quad (5)$$

Proposition 1 *The scenario grouping model (5) is a relaxation of the CCSP (3), i.e., $v^G \leq v^*$.*

Proof: Let (x, z) be any feasible solution to (3) and let $\tilde{z}_j = \min\{z_i : i \in \mathcal{N}_j\}$ for each $j \in \mathcal{K}$. We show that (\tilde{z}, x) is feasible to (5). By construction it holds that $\tilde{z}_j = \mathbb{I}(x \in \bigcap_{i \in \mathcal{N}_j} X^i)$ for each $j \in \mathcal{K}$. We first establish the following inequality for each $j \in \mathcal{K}$:

$$q_j \tilde{z}_j \geq \sum_{i \in \mathcal{N}_j} p_i z_i - \left(\sum_{i \in \mathcal{N}_j} p_i - q_j \right). \quad (6)$$

Indeed, if $\tilde{z}_j = 0$ this implies that $z_i = 0$ for some $i \in \mathcal{N}_j$, and thus $\sum_{i \in \mathcal{N}_j} p_i z_i \leq \sum_{i \in \mathcal{N}_j} p_i - q_j$ as required. On the other hand, if $\tilde{z}_j = 1$ this implies that $z_i = 1$ for all $i \in \mathcal{N}_j$, and (6) follows. Now, summing (6) over all $j \in \mathcal{K}$ yields

$$\sum_{j \in \mathcal{K}} q_j \tilde{z}_j \geq \sum_{j \in \mathcal{K}} \sum_{i \in \mathcal{N}_j} p_i z_i - \sum_{j \in \mathcal{K}} \sum_{i \in \mathcal{N}_j} p_i + \sum_{j \in \mathcal{K}} q_j \geq 1 - \epsilon - 1 + \sum_{j \in \mathcal{K}} q_j,$$

which establishes the result. \square

If we scale the knapsack inequality in (5) by $(\sum_{j \in \mathcal{K}} q_j)^{-1}$, this problem is again a CCSP, but with $K < N$ scenarios. Thus, any technique for obtaining a lower bound of a CCSP can also be applied to this relaxation. In particular, the quantile bound may be applied, and the resulting scenario grouping based quantile bound may be better than the original quantile bound (see Section 7 for an illustration). The dual bounds that we derive in the following sections may also be applied to a grouping-based relaxation.

3 Lagrangian dual bounds

We next introduce two Lagrangian dual problems associated with the CCSPs (3) obtained by relaxing nonanticipativity constraints. We use the following standard result (cf. [23, 28]) on a primal characterization of the Lagrangian dual.

Theorem 1 Consider a mathematical program $\min\{f(x) : H(x) \leq h, x \in X\}$, where f, H are convex functions and X is compact, and let

$$\mathcal{L}^* := \sup_{\lambda \geq 0} \left\{ \min_x \{f(x) + \lambda^\top (H(x) - h) : x \in X\} \right\}$$

be the Lagrangian dual value. Then

$$\mathcal{L}^* = \inf \{f(x) : x \in \text{conv}(X), H(x) \leq h\},$$

where $\text{conv}(X)$ denotes the convex hull of the set X .

3.1 Basic nonanticipative dual

By making copies of the decision variables x , the problem (3) can be reformulated as

$$v^* = \min_{x, z} \sum_{i \in \mathcal{N}} p_i c^\top x^i, \quad (7a)$$

$$\text{s.t. } \sum_{i \in \mathcal{N}} p_i H_i x^i = h, \quad (7b)$$

$$z_i = \mathbb{I}(x^i \in X^i), \quad i \in \mathcal{N}, \quad (7c)$$

$$z \in Z, \quad (7d)$$

$$x^i \in S, \quad i \in \mathcal{N}, \quad (7e)$$

where (7b) enforce the nonanticipativity constraints $x^1 = \dots = x^N$. The Lagrangian dual problem obtained by dualizing these nonanticipativity constraints with dual vector λ can be written as:

$$v_1^{LD} = \max_{\lambda} \{\mathcal{L}_1(\lambda) - \lambda^\top h\}, \quad (8)$$

where

$$\mathcal{L}_1(\lambda) := \min_{x, z} \left\{ \sum_{i \in \mathcal{N}} p_i (c^\top x^i + \lambda^\top H_i x^i) : (7c) - (7e) \right\}. \quad (9)$$

Next, observe that for a fixed $z \in Z$, problem (9) reduces to:

$$\psi(z) := \min_x \left\{ \sum_{i \in \mathcal{N}} p_i (c^\top x^i + \lambda^\top H_i x^i) : (7c), (7e) \right\}.$$

This problem decomposes by scenario. Let

$$\theta_i(\lambda) = \min_x \{ c^\top x + \lambda^\top H_i x : x \in S \} \quad (10)$$

and

$$\zeta_i(\lambda) = \min_x \{ c^\top x + \lambda^\top H_i x : x \in X^i \}. \quad (11)$$

Note that the feasible region of $\zeta_i(\lambda)$ is included in that of $\theta_i(\lambda)$, so we have that $\zeta_i(\lambda) \geq \theta_i(\lambda)$ for all $i \in \mathcal{N}$. By compactness of S , both θ and ζ are finite valued for all λ . Then, we have

$$\psi(z) = \sum_{i \in \mathcal{N}} p_i (\theta_i(\lambda)(1 - z_i) + \zeta_i(\lambda) z_i) = \sum_{i \in \mathcal{N}} p_i \theta_i(\lambda) + \sum_{i \in \mathcal{N}} p_i (\zeta_i(\lambda) - \theta_i(\lambda)) z_i$$

and so

$$\mathcal{L}_1(\lambda) = \sum_{i \in \mathcal{N}} p_i \theta_i(\lambda) + \min_z \left\{ \sum_{i \in \mathcal{N}} p_i (\zeta_i(\lambda) - \theta_i(\lambda)) z_i : z \in Z \right\}. \quad (12)$$

Thus, for a fixed λ , the Lagrangian relaxation value $\mathcal{L}_1(\lambda) - \lambda^\top h$ can be calculated by first calculating the values $\theta_i(\lambda)$ and $\zeta_i(\lambda)$ by solving (10) and (11) separately for each $i \in \mathcal{N}$, and then solving a single-row knapsack problem.

We close this subsection by noting that the dual problem (8) can be interpreted as a stochastic program with a mean-risk objective function. Let $\Delta_i(\lambda) = \zeta_i(\lambda) - \theta_i(\lambda)$ for all $i \in \mathcal{N}$ and let $F_{\Delta(\lambda)}(\cdot)$ denote the cumulative distribution function of $\{\Delta_i(\lambda)\}_{i \in \mathcal{N}}$ while the point mass function is $\mathbb{P}(\Delta(\lambda) = \Delta_i(\lambda)) = p_i$ for all $i \in \mathcal{N}$ and $F_{\Delta(\lambda)}^{-1}(t) := \inf\{s \in \mathbb{R} : F_{\Delta(\lambda)}(s) \geq t\}$. Then $\text{VaR}_{1-\epsilon}(\lambda) := F_{\Delta(\lambda)}^{-1}(1 - \epsilon)$ is the $(1 - \epsilon)$ -value at risk of $\Delta(\lambda)$, and the $(1 - \epsilon)$ -conditional value at risk of $\Delta(\lambda)$ is defined as [32]:

$$\text{CVaR}_{1-\epsilon}(\lambda) := \text{VaR}_{1-\epsilon}(\lambda) + \frac{1}{\epsilon} \mathbb{E}[\Delta(\lambda) - \text{VaR}_{1-\epsilon}(\lambda)]_+,$$

where $[\cdot]_+ = \max\{\cdot, 0\}$

Proposition 2 *If all of the scenarios are equally likely, i.e., $p_i = \frac{1}{N}$ for all $i \in \mathcal{N}$, and ϵN is an integer, then the dual problem (8) is equivalent to*

$$v_1^{LD} = \max_\lambda \{ \mathbb{E}[\zeta(\lambda)] - \epsilon \text{CVaR}_{1-\epsilon}(\lambda) - \lambda^\top h \}$$

Proof: When the scenarios are equally likely, the knapsack problem (12) can be solved by sorting the values $\Delta_i(\lambda)$, $i \in \mathcal{N}$ to find $\text{VaR}_{1-\epsilon}(\lambda)$. That is, we claim that $z_i^*(\lambda) = \mathbb{I}(\Delta_i(\lambda) \leq \text{VaR}_{1-\epsilon}(\lambda))$ for all $i \in \mathcal{N}$ solves (12). Indeed, $\Delta_i(\lambda) = \zeta_i(\lambda) - \theta_i(\lambda) \geq 0$ for all $i \in \mathcal{N}$. Thus, to solve (12) with $p_i = \frac{1}{N}$ for all $i \in \mathcal{N}$, we can simply choose the smallest values of $\Delta_i(\lambda)$ until the cardinality constraint $\sum_{i \in \mathcal{N}} z_i \geq (1 - \epsilon)N$ is satisfied. Therefore, the optimal solution sets $z_i^*(\lambda) = \mathbb{I}(\Delta_i(\lambda) \leq \text{VaR}_{1-\epsilon}(\lambda))$ for all $i \in \mathcal{N}$. If $N\epsilon$ is an integer, $\mathcal{L}_1(\lambda)$ can be further simplified as

$$\begin{aligned} \mathcal{L}_1(\lambda) &= \frac{1}{N} \sum_{i \in \mathcal{N}} \zeta_i(\lambda) - \epsilon \left(\text{VaR}_{1-\epsilon}(\lambda) + \frac{1}{N\epsilon} \sum_{i \in \mathcal{N}} [\Delta_i(\lambda) - \text{VaR}_{1-\epsilon}(\lambda)]_+ \right) \\ &= \mathbb{E}[\zeta(\lambda)] - \epsilon \text{CVaR}_{1-\epsilon}(\lambda). \end{aligned}$$

□

Proposition 2 can be extended to more general distributions by appropriately adjusting the definitions of $\text{VaR}_{1-\epsilon}$ and $\text{CVaR}_{1-\epsilon}$ as in [33].

3.2 Quantile based Lagrangian dual

The quantile bound in Section 2.2 can be interpreted as a relaxation obtained by creating a copy x^i of the variables x for each $i \in \mathcal{N}$, as in the reformulation (7), but then instead of using the *weighted average* of the objective values of these copies, the *maximum* objective function value among the enforced scenarios is used. This motivates the following alternative reformulation of (3):

$$v^* = \min_{x,y,z} y, \quad (13a)$$

$$\text{s.t. } c^\top x^i \leq y, \quad i \in \mathcal{N}, \quad (13b)$$

$$\sum_{i \in \mathcal{N}} p_i H_i x^i = h, \quad (13c)$$

$$z_i = \mathbb{I}(x^i \in X^i), i \in \mathcal{N}, \quad (13d)$$

$$z \in Z, \quad (13e)$$

$$x^i \in S, \quad i \in \mathcal{N}, \quad (13f)$$

where (13c) – (13f) are just a restatement of (7b) - (7e). For a fixed $y \in \mathbb{R}$, we further define the problem:

$$g(y) := y + \min_{x,z} \{0 : (13b) - (13f)\}. \quad (14)$$

Clearly, $g(y) = y$ if (13b) - (13f) is feasible for this fixed y value, otherwise (14) is infeasible, and we use the convention $g(y) = +\infty$ in this case. Then (13) can be formulated as:

$$v^* = \min_y \{g(y) : y \in \mathbb{R}\}.$$

Next, for a fixed $y \in \mathbb{R}$, let

$$R(y) = \{\{x^i, z_i\}_{i \in \mathcal{N}} : (13b), (13d) - (13f)\}$$

be the set of feasible solutions to (13) in which the nonanticipativity constraints (13c) are relaxed, and the variable y is fixed. Also, define

$$\mathcal{L}_2(\lambda, y) = \min_{x,z} \left\{ \sum_{i \in \mathcal{N}} p_i \lambda^\top H_i x^i : \{x^i, z_i\}_{i \in \mathcal{N}} \in R(y) \right\},$$

and finally

$$\omega_2(y) = y + \max_\lambda \{\mathcal{L}_2(\lambda, y) - \lambda^\top h\}. \quad (15)$$

We use the notation $\omega_2(y) = +\infty$ to indicate that the maximization problem in (15) is unbounded. In fact, as the following proposition shows, the maximization problem either has an optimal objective value that equals zero or is unbounded.

Proposition 3 *There exists $\bar{y} \in \mathbb{R}$ such that*

$$\omega_2(y) = \begin{cases} y, & \text{if } y \geq \bar{y}, \\ \infty, & \text{if } y < \bar{y}. \end{cases} \quad (16)$$

Proof: By Theorem 1, $\omega_2(y) = y + \min\{0 : \{x^i, z_i\}_{i \in \mathcal{N}} \in T(y)\}$ where $T(y) = \{\{x^i, z_i\}_{i \in \mathcal{N}} : (13c), \{x^i, z_i\}_{i \in \mathcal{N}} \in \text{conv}(R(y))\}$. Thus, $\omega_2(y) = y$ if $T(y) \neq \emptyset$ and $\omega_2(y) = \infty$, otherwise. Next, for y large enough, any feasible solution to (3) can be used to construct a feasible point in $T(y)$ (just set all x^i equal to x), and so for y large enough $\omega_2(y) = y$. In addition, since the set S is compact, it follows that for y small enough the set $R(y)$ is empty, and hence $T(y)$ is empty. The result then follows because $T(y_1) \subseteq T(y_2)$ whenever $y_1 \leq y_2$. \square

We now define the *quantile-based* Lagrangian dual problem as:

$$v_2^{LD} = \min_y \{\omega_2(y) : y \in \mathbb{R}\} = \min_y \{y : \omega_2(y) = y\}. \quad (17)$$

Theorem 2 *The quantile-based Lagrangian dual problem (17) is a relaxation of the CCSP (3), i.e., $v_2^{LD} \leq v^*$.*

Proof: This follows because $\omega_2(y) \leq g(y)$ for all $y \in \mathbb{R}$. \square

We next discuss the calculation of v_2^{LD} . First, for a given λ and y , $\mathcal{L}_2(\lambda, y)$ can be calculated by solving for each $i \in \mathcal{N}$,

$$\bar{\theta}_i(\lambda, y) := \min_x \left\{ \lambda^\top H_i x : c^\top x \leq y, x \in S \right\}$$

and

$$\bar{\zeta}_i(\lambda, y) := \min_x \left\{ \lambda^\top H_i x : c^\top x \leq y, x \in X^i \right\}.$$

Then,

$$\mathcal{L}_2(\lambda, y) = \sum_{i \in \mathcal{N}} p_i \bar{\theta}_i(\lambda, y) + \min_z \left\{ \sum_{i \in \mathcal{N}} p_i (\bar{\zeta}_i(\lambda, y) - \bar{\theta}_i(\lambda, y)) z_i : z \in Z \right\}.$$

The above characterization leads to a bisection procedure to obtain a lower bound on v_2^{LD} . It takes as input an upper bound, U , on the optimal objective value v^* , which can be obtained by any feasible solution to (3), and a lower bound L (we show in Section 4 that $L = v^Q$ is valid). At each iteration, we fix the candidate value $y = (U + L)/2$, and solve the nonsmooth problem (15) with a specified finite termination condition (e.g., an iteration limit) to obtain a lower bound $\underline{\omega}_2(y)$ on $\omega_2(y)$. The nonsmooth problem (15) could be solved using convex programming techniques, e.g., the subgradient method or its regularized variants such as the bundle method. If $\underline{\omega}_2(y) > y$, then we can update $L = y$, otherwise we update $U = y$. The bisection procedure terminates when the difference between the upper and lower bounds is less than a given tolerance. At any step of the algorithm, L is a valid lower bound on v_2^{LD} .

4 Strength of Lagrangian dual bounds

In this section, we compare the Lagrangian dual bounds developed in Section 3 and the basic lower bounds in Section 2.

4.1 Comparing v_1^{LD} and $v^C(M)$

We first show that v_1^{LD} is no smaller than $v^C(M)$. Let

$$C_M = \left\{ (x, z) : x \in S, G_i(x) \leq M_i(1 - z_i), z_i \in [0, 1], \forall i \in \mathcal{N}, \sum_{i \in \mathcal{N}} p_i z_i \geq 1 - \epsilon \right\}$$

be the feasible region of the continuous relaxation of (4) in which the variables z are relaxed to be continuous.

Theorem 3 *Assume that problem (4) satisfies Slater's condition, i.e., there exists $(\hat{x}, \hat{z}) \in \text{int}(\text{conv}(C_M))$, where $\text{int}(\cdot)$ denotes the interior of a set. Then,*

$$v_1^{LD} \geq v^C(M).$$

Proof: First observe that the continuous relaxation of (4) is a convex program with a linear objective function over the convex hull of the set C_M , which is assumed to satisfy Slater's condition [36]. Therefore, by strong duality, the Lagrangian dual of this convex program in which the nonanticipativity constraints $\sum_{i \in \mathcal{N}} p_i H_i x^i = h$ are relaxed has the optimal value equal to $v^C(M)$. But the Lagrangian relaxation problem used in this Lagrangian dual is identical to that in (8) except that the z variables are relaxed to be continuous. The conclusion follows. \square

Next we establish a set of sufficient conditions under which $v^C(M)$ is equal to v_1^{LD} .

Proposition 4 *Suppose that $S = \mathbb{R}_+^n$ and for each $i \in \mathcal{N}$, $p_i = \frac{1}{N}$ and $G_i(x) = \bar{G}_i(x) + M_i$, where $\bar{G}_i(tx) \leq t\bar{G}_i(x)$ for all $t \geq 1$ and $\bar{G}_i(0) \leq 0$. Then $v^C(M) = v_1^{LD}$.*

Proof: We only need to show that $v^C(M) \geq v_1^{LD}$. Recall that

$$\begin{aligned} v^C(M) &= \min_{x,z} \left\{ c^\top x : x \in S, G_i(x) \leq M_i(1 - z_i), z_i \in [0, 1], \forall i \in \mathcal{N}, \right. \\ &\quad \sum_{i \in \mathcal{N}} z_i \geq \lceil N(1 - \epsilon) \rceil \} \\ &= \min_{x,z} \left\{ c^\top x : x \in S, G_i(x) \leq M_i(1 - z_i), \forall i \in \mathcal{N}, z \in \text{conv}(Z) \right\}. \end{aligned}$$

From Theorem 1 and equation (8) we know that

$$\begin{aligned} v_1^{LD} &= \min_{x,z} \left\{ \sum_{i=1}^N p_i c^\top x^i : \{(x^i, z_i)\}_{i \in \mathcal{N}} \in \text{conv}(\{(x^i, z_i)\}_{i \in \mathcal{N}} : x^i \in S, \right. \\ &\quad \left. G_i(x^i) \leq M_i(1 - z_i), \forall i \in \mathcal{N}, z \in Z\}) , \sum_{i \in \mathcal{N}} p_i H_i x^i = h \right\}. \end{aligned}$$

Let $(\hat{\mathbf{x}}, \hat{\mathbf{z}})$, where $\hat{\mathbf{z}} := \{\hat{z}_i\}_{i \in \mathcal{N}}$, be an optimal solution of the continuous relaxation of (4). Since $\hat{\mathbf{z}} \in \text{conv}(Z)$, there exists a set of points $\{\mathbf{z}_k\} \in Z$ such that $\hat{\mathbf{z}} = \sum_k \lambda_k \mathbf{z}_k$ with $\sum_k \lambda_k = 1$ and $\lambda_k > 0$. Construct $\mathbf{x}_k^i = \frac{\hat{\mathbf{x}}}{\hat{z}_i} z_{ik}$ for all k and for all $i \in \mathcal{N}$. Note that the operations are well defined since, for each i , $\hat{z}_i = 1$ (or 0) implies $z_{ik} = 1$ (or 0) for all k , and we assume that $0 \cdot \infty = 0$. It follows that

$$\begin{aligned} \mathbf{x}_k^i &\in S, \mathbf{z}_k \in Z, \\ G_i(\mathbf{x}_k^i) &= \bar{G}_i(\mathbf{x}_k^i) + M_i \leq \frac{1}{\hat{z}_i} \bar{G}_i(\hat{\mathbf{x}}) z_{ik} + M_i \leq M_i(1 - z_{ik}), \forall i \in \mathcal{N}, \end{aligned}$$

where the first equality is the definition of $G_i(\cdot)$, the second inequality follows because if $z_{ik} = 0$, then $\mathbf{x}_k^i = 0$, and $\bar{G}_i(0) \leq 0$; otherwise, $\mathbf{x}_k^i = \frac{\hat{\mathbf{x}}}{\hat{z}_i}$ and $\bar{G}_i(t\mathbf{x}) \leq t\bar{G}_i(\mathbf{x})$ for all $t \geq 1$; while the last inequality follows since $G_i(\hat{\mathbf{x}}) \leq M_i(1 - \hat{z}_i)$ or equivalently, $\frac{1}{\hat{z}_i} \bar{G}_i(\hat{\mathbf{x}}) \leq -M_i$. Now define $(\mathbf{x}^i, \hat{\mathbf{z}}) = \sum_k \lambda_k (\mathbf{x}_k^i, \mathbf{z}_k)$, then we have $\mathbf{x}^i = \hat{\mathbf{x}}$ for all $i \in \mathcal{N}$. Hence,

$$\{(\mathbf{x}^i, \hat{z}_i)\}_{i \in \mathcal{N}} \in \text{conv}(\{(x^i, z_i)\}_{i \in \mathcal{N}} : x^i \in S, G_i(x^i) \leq M_i(1 - z_i), \forall i \in \mathcal{N}, z \in Z\})$$

and $\{\mathbf{x}^i\}_{i \in \mathcal{N}}$ also satisfies the nonanticipativity constraints. Thus $(\hat{\mathbf{x}}, \hat{\mathbf{z}})$ is also feasible to (8) implying $v^C(M) \geq v_1^{LD}$. \square

A large class of problems that satisfy the conditions of Proposition 4 are chance-constrained covering linear programs with equiprobable scenarios [31]:

$$\min_{x,z} \left\{ c^\top x : A^i x \geq b^i z_i, \forall i \in \mathcal{N}, \sum_{i \in \mathcal{N}} z_i \geq (1 - \epsilon)N, x \geq 0, z_i \in \{0, 1\}, \forall i \in \mathcal{N} \right\}.$$

where $A_i \in \mathbb{R}_+^{m_i \times n}$, $b_i \in \mathbb{R}_+^{m_i}$ for all $i \in \mathcal{N}$. Recasting the above problem in the form of (4) we note that $S = \mathbb{R}_+^n$, $\bar{G}_i(x) = -A^i x$, $M_i = b_i$. Indeed we have $\bar{G}_i(tx) = -tA^i x = t\bar{G}_i(x)$ for all $t \geq 1$, and $\bar{G}_i = 0$.

Remark 1 For chance-constrained covering linear programs with equiprobable scenarios [31], the basic nonanticipative dual bound v_1^{LD} is equal to the relaxation bound of the big-M formulation (4) obtained by relaxing the integrality constraints on z , i.e., $v_1^{LD} = v^C(M)$. For other chance-constrained linear programs, v_1^{LD} could be strictly better than $v^C(M)$ (see Table 1 in Section 7).

4.2 Comparing v_1^{LD} and v_2^{LD}

The following theorem compares the strengths of the two different Lagrangian dual bounds v_1^{LD} and v_2^{LD} .

Theorem 4 *The quantile-based Lagrangian dual is at least as strong as the basic nonanticipative Lagrangian dual, i.e., $v_1^{LD} \leq v_2^{LD}$.*

Proof: Let $Q_i := \{(x^i, z_i) : (7c), (7e)\} = \{(x^i, z_i) : (13d), (13f)\}$ for all $i \in \mathcal{N}$. The claim follows since

$$\begin{aligned} v_1^{LD} &= \max_{\lambda} \min_{x,y,z} \left\{ y + \sum_{i \in \mathcal{N}} p_i \lambda^\top H_i x^i - \lambda^\top h : (x^i, z_i) \in Q_i, \forall i \in \mathcal{N}, \right. \\ &\quad \left. z \in Z, y \geq \sum_{i \in \mathcal{N}} p_i c^\top x^i \right\} \\ &\leq \max_{\lambda} \min_{x,y,z} \left\{ y + \sum_{i \in \mathcal{N}} p_i \lambda^\top H_i x^i - \lambda^\top h : (x^i, z_i) \in Q_i, \forall i \in \mathcal{N}, \right. \\ &\quad \left. z \in Z, y \geq c^\top x^i, \forall i \in \mathcal{N} \right\} \\ &\leq \min_y \max_{\lambda} \min_{x,z} \left\{ y + \sum_{i \in \mathcal{N}} p_i \lambda^\top H_i x^i - \lambda^\top h : (x^i, z_i) \in Q_i, \forall i \in \mathcal{N}, \right. \\ &\quad \left. z \in Z, y \geq c^\top x^i, \forall i \in \mathcal{N} \right\} \\ &= v_2^{LD}, \end{aligned}$$

where the first equality follows from the definition of v_1^{LD} ; the second inequality follows since $y \geq \sum_{i \in \mathcal{N}} p_i c^\top x^i$ is an aggregation of the constraints $y \geq c^\top x^i$ for each $i \in \mathcal{N}$; the third inequality follows from the max-min inequality; and the final equality is from the definition of v_2^{LD} . \square

4.3 Comparisons with v^Q

Theorem 5 *The quantile-based Lagrangian dual bound is at least as strong as the quantile bound, i.e., $v_2^{LD} \geq v^Q$.*

Proof: First define

$$v_2^R(y) = \begin{cases} y & R(y) \neq \emptyset, \\ +\infty & \text{otherwise.} \end{cases}$$

Observe that $v_2^R(y) = y + \mathcal{L}_2(0, y)$, and so $v_2^R(y) \leq \omega_2(y)$ for all $y \in \mathbb{R}$ because it is obtained by using $\lambda = 0$ in (15). Thus, it follows that $v^R := \min\{v_2^R(y) : y \in \mathbb{R}\} \leq v_2^{LD}$. We show that this first bound is identical to the quantile bound, i.e., $v^R = v^Q$.

Recall that

$$R(y) = \{\{x^i, z_i\}_{i \in \mathcal{N}} : (13b), (13d) - (13f)\}.$$

We first show $R(v^Q) \neq \emptyset$, which implies $v_2^R(v^Q) = v^Q$ and thus $v^R \leq v^Q$. Recall that $\eta_i := \min\{c^\top x : x \in X^i\}$. Let $I^Q = \{\sigma_q, \dots, \sigma_N\}$ be the set of scenarios i that $\eta_i \leq \eta_{\sigma_q} = v^Q$ for all $i \in I^Q$. By the definition of v^Q , $\sum_{i \in I^Q} p_i \geq 1 - \epsilon$. Also, for each $i \in I^Q$, there exists $\bar{x}^i \in X^i$ with $c^\top \bar{x}^i = \eta_i \leq v^Q$. Next, let $\hat{x} \in \arg \min\{c^\top x : x \in S\}$ and observe that $c^\top \hat{x} \leq \eta_i \leq v^Q$ for all $i \in I^Q$. Then, a feasible point of $R(v^Q)$ is obtained by setting $x^i = \bar{x}^i$ for $i \in I^Q$, $x^i = \hat{x}$ for $i \in \mathcal{N} \setminus I^Q$ and setting $z_i = 1$ for $i \in I^Q$ and $z_i = 0$ otherwise.

Now let $y < v^Q$ and let $I(y) := \{i \in \mathcal{N} : \eta_i \leq y\}$. For each scenario $i \in \mathcal{N} \setminus I(y)$ there is no $x^i \in X^i$ with $c^\top x^i \leq y$. By definition of v^Q , it holds that $\sum_{i \in I(y)} p_i < 1 - \epsilon$. Thus, $R(y) = \emptyset$ and $v_2^R(y) = +\infty$. Thus $v^R > y$. As $y < v^Q$ is arbitrary, we conclude that $v^R \geq v^Q$. \square

Neither of $v_1^{LD}, v^C(M)$ has a general bound relationship with v^Q . The computational results in Section 7 provide examples where the quantile bound v^Q is stronger than v_1^{LD} or $v^C(M)$, while the following example shows that v_1^{LD} or $v^C(M)$ can be stronger than v^Q .

Example 1 Consider a three-scenario instance as follows: $X^1 = \{x \in \mathbb{R}_+^2 : 0.5x_1 + 2x_2 \geq 1\}$, $X^2 = \{x \in \mathbb{R}_+^2 : 2x_1 + 0.5x_2 \geq 1\}$, $X^3 = \{x \in \mathbb{R}_+^2 : x_1 + x_2 \geq 1\}$, and $S = \mathbb{R}_+^2$. Each scenario happens with probability $1/3$, and we let $\epsilon = 1/3$, $M = 1$. The objective is to minimize $x_1 + x_2$. For this instance, the quantile bound $v^Q = 0.5$, and $v_1^{LD} = v^C(M) = 4/7$, therefore, $v_1^{LD}, v^C(M)$ are stronger lower bounds.

4.4 Bound comparison summary

We close this section by noting a set of sufficient conditions under which there is no duality gap.

Proposition 5 Suppose $G_i : S \rightarrow \mathbb{R}_{-}^{m_i} \cup \mathcal{R}_M^i$ for all $i \in \mathcal{N}$, where $\mathcal{R}_M^i = \{s \in \mathbb{R}^{m_i} : \|s\|_{\infty} = M\}$ for all $i \in \mathcal{N}$ and $M \in \mathbb{R}_+$. Then we have $v^C(M) = v_1^{LD} = v_2^{LD} = v^*$.

Proof: From Theorem 2 and Theorems 3 and 4, it is sufficient to show that $v^C(M) \geq v^*$.

Suppose that (\hat{x}, \hat{z}) is an optimal solution of the continuous relaxation of (4), where $\hat{z} := \{\hat{z}_i\}_{i \in \mathcal{N}}$. We show that $(\hat{x}, [\hat{z}])$ is another optimal solution. Indeed, if \hat{z} is integral, then we are done. Otherwise, suppose that there is an i' such that $\hat{z}_{i'} \in (0, 1)$, then by the definition of $G_{i'}(\cdot)$, we must have $G_{i'}(\hat{x}) \leq 0$. Thus, $(\hat{x}, [\hat{z}])$ is feasible to the continuous relaxation of (4) with the optimal value $v^C(M)$. Since $[\hat{z}] \in Z$, $(\hat{x}, [\hat{z}])$ is also feasible to (4). Thus, $v^C(M) \geq v^*$. \square

A large class of problems that satisfy the conditions of Proposition 5 are chance-constrained set covering problems [2, 5]:

$$\min_{x,z} \left\{ c^\top x : A^i x \geq b^i z_i, \forall i \in \mathcal{N}, x \in \{0,1\}^n, z \in Z \right\}, \quad (18)$$

where $A^i \in \{0,1\}^{m_i \times n}$, $b^i \in \{0,1\}^{m_i}$ and $\|b^i\|_{\infty} = 1$ for all $i \in \mathcal{N}$. Here $G_i(x) = -A^i x + b^i$ and so $G_i(x) : \{0,1\}^n \rightarrow \mathbb{Z}_{-}^{m_i} \cup \mathcal{R}_1^i$ with $\mathcal{R}_1^i = \{s \in \{0,1\}^{m_i} : \|s\|_{\infty} = 1\}$, for all $i \in \mathcal{N}$ and hence $v^C(M) = v_1^{LD} = v_2^{LD} = v^*$.

The relationships between the basic lower bounds of Section 2 and the Lagrangian dual bounds of Section 3 are summarized in Figure 1.

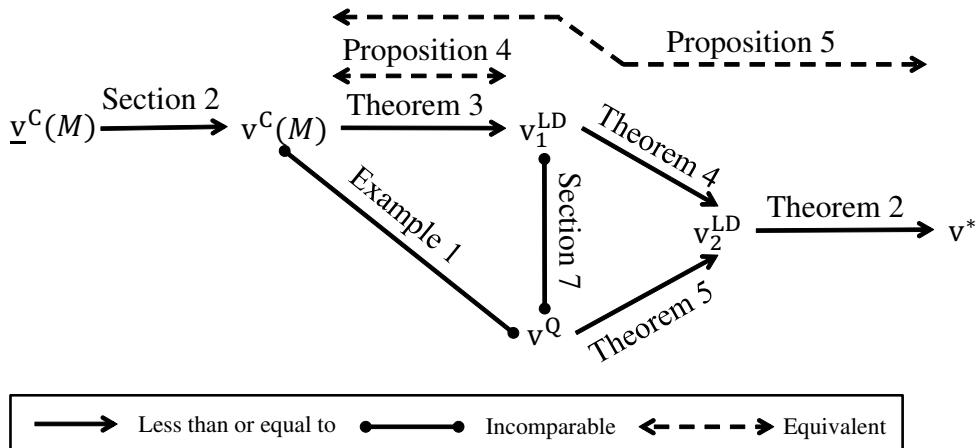


Figure 1: Bound comparison summary

5 Primal formulations for chance-constrained mixed-integer linear programs

In this section we consider chance-constrained mixed-integer linear programs (MILP), i.e., problem (3) where $S = \{x \in \mathbb{R}^{n-r} \times \mathbb{Z}^r : Dx \leq d\}$ and $X^i = \{x \in \mathbb{R}^{n-r} \times \mathbb{Z}^r : A^i x \leq b^i\}$ for each $i \in \mathcal{N}$. Recall

our assumption that, for all $i \in \mathcal{N}$, $X^i \subseteq S$ and so we may assume the constraints $Dx \leq d$ are included in the constraints $A^i x \leq b^i$. We derive two new formulations for such problems that are inspired by the two Lagrangian dual problems proposed in the previous sections. In particular, under certain conditions, these relaxations are primal formulations of the Lagrangian duals. The constructions here can be extended to the case where S , X^i and f are defined by convex functions using the perspective function approach in [10].

5.1 Primal formulation corresponding to v_1^{LD}

Note that replacing $z_i = \mathbb{I}(x^i \in X^i)$ in (8) by $z_i \leq \mathbb{I}(x^i \in X^i)$ for each $i \in \mathcal{N}$ yields an equivalent formulation. Recall that $Z = \{z \in \{0, 1\}^N : \sum_{i=1}^N p_i z_i \geq 1 - \epsilon\}$ and let

$$T_1 := \left\{ \{(x^i, z_i)\}_{i \in \mathcal{N}} : D x^i \leq d, z_i \leq \mathbb{I}(A^i x^i \leq b^i), \right. \\ \left. x^i \in \mathbb{R}^{n-r} \times \mathbb{Z}^r, \forall i \in \mathcal{N}, z \in Z \right\}.$$

From Theorem 1 and equation (8) we know that

$$v_1^{LD} = \min_{x, z} \left\{ \sum_{i=1}^N p_i c^\top x^i : \{(x^i, z_i)\}_{i \in \mathcal{N}} \in \text{conv}(T_1), \sum_{i \in \mathcal{N}} p_i H_i x^i = h \right\}. \quad (19)$$

Next we use an extended formulation of $\text{conv}(T_1)$ to derive a linear programming relaxation of (19) in the following form

$$z_1^{LP} := \min_{x, z, u, w} c^\top x \quad (20a)$$

$$\text{s.t. } u^i + w^i = x, \quad \forall i \in \mathcal{N}, \quad (20b)$$

$$A^i u^i \leq b^i z_i, \quad \forall i \in \mathcal{N}, \quad (20c)$$

$$Dw^i \leq d(1 - z_i), \quad \forall i \in \mathcal{N}, \quad (20d)$$

$$z \in \text{conv}(Z). \quad (20e)$$

We let $P_S := \{x \in \mathbb{R}^n : Dx \leq d\}$ and $P^i := \{x \in \mathbb{R}^n : A^i x \leq b^i\}, i \in \mathcal{N}$ be the continuous relaxations of S, X^i for each $i \in \mathcal{N}$ respectively (the sets are identical in the case $r = 0$). The next theorem shows the relationship between v_1^{LD} and z_1^{LP} .

Theorem 6 *The basic Lagrangian dual bound v_1^{LD} and the primal bound z_1^{LP} defined in formulation (20) satisfy: $v_1^{LD} \geq z_1^{LP}$, where the equality holds if $P_S = \text{conv}(S)$ and $P^i = \text{conv}(X^i)$ for all $i \in \mathcal{N}$.*

Proof: We just need to show that when $P_S = \text{conv}(S)$ and $P^i = \text{conv}(X^i)$ for all $i \in \mathcal{N}$, $v_1^{LD} = z_1^{LP}$. In the following, for the sake of notational simplicity, we use $(\mathbf{x}, \mathbf{z}, \mathbf{u}, \mathbf{w}) = \{(x^i, z_i, u^i, w^i)\}_{i \in \mathcal{N}}$ and the operations on these vectors will be assumed to be scenario-wise, e.g., $\mathbf{x} \cdot \mathbf{z} := \{x^i z_i\}_{i \in \mathcal{N}}$ and $\mathbf{x}/\mathbf{z} := \{x^i/z_i\}_{i \in \mathcal{N}}$ (here, if $z_i = 0$ the corresponding element is defined to be zero).

(i) Define

$$\bar{T}_1 := \left\{ \{(x^i, z_i)\}_{i \in \mathcal{N}} : D x^i \leq d, z_i \leq \mathbb{I}(A^i x^i \leq b^i), \forall i \in \mathcal{N}, z \in Z \right\}.$$

We show that $\text{conv}(T_1) = \text{conv}(\bar{T}_1)$. Clearly, $\text{conv}(T_1) \subseteq \text{conv}(\bar{T}_1)$ as $T_1 \subseteq \bar{T}_1$. Hence, we only need to show that $\text{conv}(T_1) \supseteq \text{conv}(\bar{T}_1)$ or equivalently, $\text{conv}(T_1) \supseteq \bar{T}_1$.

Let $(\mathbf{x}, \mathbf{z}) \in \bar{T}_1$. Then for each $i \in \mathcal{N}$, we have $x^i \in P^i$ if $z_i = 1$; $x^i \in P_S$, otherwise. Thus, there exists a finite set of vectors $\{x_{k^i}^i\}_{k^i \in K^i}$ and nonnegative weights $\{\lambda_k^i\}_{k \in K^i}$ such that $\sum_{k \in K^i} \lambda_k^i = 1$, $x^i = \sum_{k^i \in K^i} \lambda_{k^i}^i x_{k^i}^i$ and $D x_{k^i}^i \leq d$, $z_i \leq \mathbb{I}(A^i x_{k^i}^i \leq b^i)$, $x_{k^i}^i \in \mathbb{R}^{n-r} \times \mathbb{Z}^r$ for each $k^i \in K^i$. Hence, for each $\{k^i\}_{i \in \mathcal{N}} \in \prod_{i \in \mathcal{N}} K^i$, we have $\{(x_{k^i}^i, z_i)\}_{i \in \mathcal{N}} \in T_1$, and $(\mathbf{x}, \mathbf{z}) \in \text{conv}(\{(x_{k^i}^i, z_i)\}_{i \in \mathcal{N}}, \forall \{k^i\}_{i \in \mathcal{N}} \in \prod_{i \in \mathcal{N}} K^i) \subseteq \text{conv}(\bar{T}_1)$.

(ii) Next, we define the polyhedron:

$$W_1 := \{(x^i, z_i)\}_{i \in \mathcal{N}} : \exists (u^i, w^i), i \in \mathcal{N}, \text{ s.t. (20c) - (20e), } u^i + w^i = x^i, \forall i \in \mathcal{N}\}.$$

Then, because the constraints $\sum_{i \in \mathcal{N}} p_i H_i x^i = h$ enforce that all vectors x^i are equal to the same vector, say x , (20) can be reformulated as:

$$z_1^{LP} = \min_{x, z} \left\{ \sum_{i=1}^N p_i c^\top x^i : \{(x^i, z_i)\}_{i \in \mathcal{N}} \in W_1, \sum_{i \in \mathcal{N}} p_i H_i x^i = h \right\}.$$

Therefore, it is sufficient to show that $\text{conv}(T_1) = \text{conv}(\bar{T}_1) = W_1$. We claim that $\bar{T}_1 \subseteq W_1$, since for each $(x, z) \in \bar{T}_1$, we can find a (u^i, w^i) pair for each $i \in \mathcal{N}$ that satisfies all the constraints in W_1 by setting $u^i = x^i, w^i = 0$ if $z_i = 1$, and $u^i = 0, w^i = x^i$ otherwise. Therefore, $\text{conv}(T_1) = \text{conv}(\bar{T}_1) \subseteq W_1$. Thus, we only need to show $W_1 \subseteq \text{conv}(\bar{T}_1)$.

Let $(\mathbf{x}, \mathbf{z}, \mathbf{u}, \mathbf{w}) \in W_1$. As $\mathbf{z} \in \text{conv}(Z)$, there exists a finite set of vectors $\{\bar{\mathbf{z}}_k\}_{k \in K}$ and nonnegative weights $\{\lambda_k\}_{k \in K}$ such that $\mathbf{z} = \sum_{k \in K} \lambda_k \bar{\mathbf{z}}_k$. Now, for each $k \in K$, define vector $\bar{\mathbf{x}}_k = \bar{\mathbf{z}}_k \cdot (\mathbf{u}/\mathbf{z}) + (1 - \bar{\mathbf{z}}_k) \cdot \mathbf{w}/(1 - \mathbf{z})$. Then, a simple calculation would show that $\sum_{k \in K} \lambda_k \bar{\mathbf{x}}_k = \mathbf{x}$.

The vector $(\bar{\mathbf{x}}_k, \bar{\mathbf{z}}_k)$ satisfies $\bar{\mathbf{z}}_k \in Z$, and for $i \in \mathcal{N}$, if $\bar{\mathbf{z}}_{ik} = 0$ then $\bar{\mathbf{x}}_k^i = w^i/(1 - z_i) \in \text{conv}(S) = P_S$ from (20d) and if $\bar{\mathbf{z}}_{ik} = 1$, then $\bar{\mathbf{x}}_k^i = u^i/z_i \in \text{conv}(X^i) = P^i$ from (20c). Thus, $(\bar{\mathbf{x}}_k, \bar{\mathbf{z}}_k) \in \bar{T}_1$ for each $k \in K$, which directly implies that $(\mathbf{x}, \mathbf{z}) \in \text{conv}(\bar{T}_1)$. \square

It follows from Theorem 6 that $v_1^{LD} = z_1^{LP}$ for chance-constrained linear programs (i.e., when $r = 0$).

When $p_i = 1/N$ for all $i \in \mathcal{N}$ then $\text{conv}(Z) = \{z : \sum_{i \in \mathcal{N}} z_i \geq \lceil (1 - \epsilon)N \rceil, z_i \in [0, 1], i \in \mathcal{N}\}$. For general p_i values a description of $\text{conv}(Z)$ in (20e) would require the convex hull of the corresponding knapsack set. Since this is in general intractable, we may replace constraint (20e) with a suitable polyhedral relaxation, at the expense of weakening the LP relaxation bound.

Inspired by the above primal formulation, we obtain the following “big- M ” free formulation for chance-constrained MILP.

Proposition 6

$$v^* = \min_{x, z, u, w} \{c^\top x : (20b) - (20d), z \in Z, x \in \mathbb{R}^{n-r} \times \mathbb{Z}^r\}, \quad (21)$$

is a valid MILP formulation of (3).

5.2 Primal formulation corresponding to v_2^{LD}

We next derive a primal formulation for v_2^{LD} under certain conditions. From Theorem 1 and equation (17), we have

$$v_2^{LD} = \min_{x, y, z} \left\{ y : \exists \{(x^i, z_i)\}_{i \in \mathcal{N}} \in \text{conv}(R(y)), \sum_{i \in \mathcal{N}} p_i H_i x^i = h \right\}, \quad (22)$$

where $R(y) = \{\{x^i, z_i\}_{i \in \mathcal{N}} : c^\top x^i \leq y, D x^i \leq d, z_i \leq \mathbb{I}(A^i x^i \leq b^i), x^i \in \mathbb{R}^{n-r} \times \mathbb{Z}^r, \forall i \in \mathcal{N}, z \in Z\}$. Next we use an extended formulation of $\text{conv}(R(y))$ to derive the following nonlinear programming formulation of (22):

$$z_2^{NLP} := \min_{x, y, z, u, w} y \quad (23a)$$

$$\text{s.t. } c^\top u^i \leq y z_i, \quad \forall i \in \mathcal{N}, \quad (23b)$$

$$c^\top w^i \leq y(1 - z_i), \quad \forall i \in \mathcal{N}, \quad (23c)$$

and (20b) - (20e).

We define $P_S(y) := \{x \in \mathbb{R}^n : c^\top x \leq y, D x \leq d\}$ and $P^i(y) := \{x \in \mathbb{R}^n : c^\top x \leq y, A^i x \leq b^i\}, i \in \mathcal{N}$. The next theorem shows the relationship between v_2^{LD} and z_2^{NLP} .

Theorem 7 *The quantile-based Lagrangian dual bound v_2^{LD} and the primal bound z_2^{NLP} defined in formulation (23) satisfy: $v_2^{LD} \geq z_2^{NLP}$, where the equality holds if $P_S(y) = \text{conv}(S \cap \{x : c^\top x \leq y\})$ and $P^i(y) = \text{conv}(X^i \cap \{x : c^\top x \leq y\})$ for all $i \in \mathcal{N}$ and for all $y \in \mathbb{R}$.*

Proof. We only need to show that when $P_S(y) = \text{conv}(S \cap \{x : c^\top x \leq y\})$ and $P^i(y) = \text{conv}(X^i \cap \{x : c^\top x \leq y\})$ for all $i \in \mathcal{N}$ and y , then $v_2^{LD} = z_2^{NLP}$. This directly implies that $v_2^{LD} \geq z_2^{NLP}$ as $P_S(y) \supseteq \text{conv}(S \cap \{x : c^\top x \leq y\})$ and $P^i(y) \supseteq \text{conv}(X^i \cap \{x : c^\top x \leq y\})$.

The remainder of the proof is almost identical to that of Theorem 6, so we provide a sketch.

- (i) Let us define $\bar{R}(y) := \{(x^i, z_i)\}_{i \in \mathcal{N}} : c^\top x^i \leq y, Dx^i \leq d, z_i \leq \mathbb{I}(A^i x^i \leq b^i), \forall i \in \mathcal{N}, z \in Z\}$. Using an approach identical to that in part (i) of the proof of Theorem 6 it can be shown that $\text{conv}(R(y)) = \text{conv}(\bar{R}(y))$ for a given y .
- (ii) Next, we define a set $W_2(y) = \{(x^i, z_i)\}_{i \in \mathcal{N}} : \exists(u^i, w^i), i \in \mathcal{N}, \text{s.t. (20c)} - \text{(20e)}, \text{(23b)}, \text{(23c)}, u^i + w^i = x^i, \forall i \in \mathcal{N}\}$ with a given y . Then, because the constraints $\sum_{i \in \mathcal{N}} p_i H_i x^i = h$ enforce that all vectors x^i are equal to the same vector, say x , (23) can be reformulated as:

$$z_2^{NLP} = \min_{x,y,z} \left\{ y : \{(x^i, z_i)\}_{i \in \mathcal{N}} \in W_2(y), \sum_{i \in \mathcal{N}} p_i H_i x^i = h \right\}.$$

Therefore, it is sufficient to show that $\text{conv}(R(y)) = \text{conv}(\bar{R}(y)) = W_2(y)$. The proof of this is similar to part (ii) of the proof of Theorem 6. \square

It follows from Theorem 7 that $v_2^{LD} = z_2^{NLP}$ for chance-constrained linear programs (i.e., when $r = 0$).

Although (23) is a nonconvex nonlinear program, it can be solved by bisection on the value of y by observing that the feasible region of (23) is nonincreasing over y . Thus, z_2^{NLP} can be calculated by finding the minimum value of y for which the feasible regions of (23) is nonempty. Such a minimum exists, since the feasible region of (23) is a closed set, and its objective is bounded from below. For any fixed y , the feasibility problem of (23) is a linear program. The disadvantage of solving (23) by bisection is that it may be difficult to incorporate such a procedure within a standard linear programming based branch-and-cut algorithm. We therefore propose an iterative scheme that solves a sequence of linear programs that generate progressively better lower bounds for z_2^{NLP} , and eventually converges to z_2^{NLP} .

A linear programming based approach for z_2^{NLP}

Let ℓ be a lower bound for z_2^{NLP} (e.g., one can use $\underline{v}^C(M)$). Given such a lower bound ℓ , the nonconvex constraints (23b) and (23c) can be reformulated with linear constraints, leading to the following formulation:

$$z_2^{LP}(\ell) = \min_{x,y,z,u,w} y, \quad (24a)$$

$$\text{s.t. } y \geq c^\top u^i + \ell(1 - z_i), \forall i \in \mathcal{N}, \quad (24b)$$

$$y \geq c^\top w^i + \ell z_i, \forall i \in \mathcal{N}, \quad (24c)$$

and (20b) – (20e).

Observe that $z_2^{LP}(\ell)$ is an increasing function of ℓ , and if $\ell \leq z_2^{NLP}$ then $z_2^{LP}(\ell) \leq z_2^{NLP}$. Therefore, if we solve (24) iteratively and update ℓ using the optimal objective value, eventually we will converge to some $z_2^{LP}(\bar{\ell}) = \bar{\ell} \leq z_2^{NLP}$. In fact, the value $\bar{\ell}$ will be the same as z_2^{NLP} , since if we replace y and ℓ in (24b) and (24c) by $\bar{\ell}$, we get the same structure as (23b) and (23c). We formalize these assertions in the next two results.

Proposition 7 *Let $z_2^{NLP} = \ell^*$, then $z_2^{LP}(\ell^*) = \ell^*$.*

Proof: First, $y = \ell^*$ satisfies all the constraints in (24), so $z_2^{LP}(\ell^*) \leq \ell^*$. Suppose $z_2^{LP}(\ell^*) < \ell^*$, then there exists $y^* < \ell^*$ and $\{(u^i, w^i, z_i)\}_{i \in \mathcal{N}}$, which is feasible to (20b) - (20e) and (24b) and (24c) being:

$$\begin{aligned} y^* &\geq c^\top u^i + \ell^*(1 - z_i), \quad \forall i \in \mathcal{N}, \\ y^* &\geq c^\top w^i + \ell^* z_i, \quad \forall i \in \mathcal{N}. \end{aligned}$$

Thus $y^* \geq c^\top u^i + y^*(1 - z_i)$, $y^* \geq c^\top w^i + y^* z_i$ since $z_i \in [0, 1]$ for each $i \in \mathcal{N}$. So this solution is feasible to (23), which is a contradiction. \square

Proposition 8 Suppose $\ell_0 \leq z_2^{NLP}$, and let $\ell_k = z_2^{LP}(\ell_{k-1})$ for $k \geq 1$. Then $\{\ell_k, k \geq 1\}$ converges to z_2^{NLP} .

Proof: As $\{\ell_k\}$ is bounded above by z_2^{NLP} , the sequence converges to some $\bar{\ell} \leq z_2^{NLP}$ by the monotone convergence theorem. On the other hand, as $z_2^{LP}(\bar{\ell}) = \bar{\ell}$, there exists $\{(u^i, w^i, z_i)\}_{i \in \mathcal{N}}$ such that this solution with $y = \bar{\ell}$ is feasible to (24b) and (24c) with $\ell = \bar{\ell}$, and (20b) - (20e). But this implies that this solution is also feasible to (23), and hence $z_2^{NLP} \leq \bar{\ell}$. \square

Similar to the primal formulation (21), enforcing integrality constraints on z and any integer constrained x in (24) yields an alternative “big- M ” free formulation for a general chance-constrained MILP.

Proposition 9

$$v^* = \min \{y : x \in \mathbb{R}^{n-r} \times \mathbb{Z}^r, z \in Z, (24b), (24c), \text{and } (20b) - (20d)\}, \quad (25)$$

is a valid MILP formulation of (3).

Recall that the constraints (24b) and (24c) depend on a given lower bound ℓ . In our arguments above we required $\ell \leq z_2^{NLP}$ in order to argue that the iterative solution of the linear programming relaxation will converge to z_2^{NLP} . However, any $\ell \leq v^*$ can be used for validity of the formulation (25). As examples, one may choose to use the quantile bound v^Q , or z_2^{NLP} obtained by iteratively solving (24). In Section 6, we develop branch-and-cut decomposition algorithms based on MILP formulations (21) and (25).

A second-order cone programming based approach for z_2^{NLP}

Inspired by the nonlinear program (23), we consider the following second order cone programming (SOCP) problem

$$z_2^{SOC}(\ell) := \min_{x, \beta, z, u, w, s, t} \beta, \quad (26a)$$

$$\text{s.t. } c^\top u^i - \ell z_i \leq s_i, \quad (26b)$$

$$c^\top w^i - \ell(1 - z_i) \leq t_i, \quad (26c)$$

$$(s_i)^2 \leq \beta z_i, \quad \forall i \in \mathcal{N}, \quad (26d)$$

$$(t_i)^2 \leq \beta(1 - z_i), \quad \forall i \in \mathcal{N}, \quad (26e)$$

$$s_i, t_i \in \mathbb{R}_+, \quad \forall i \in \mathcal{N}, \quad (26f)$$

and (20b) - (20e),

where ℓ is a lower bound on z_2^{NLP} . To see that that (26) is indeed an SOCP problem note that $\beta z_i = \frac{1}{4}(\beta + z_i)^2 - \frac{1}{4}(\beta - z_i)^2$ and $\beta(1 - z_i) = \frac{1}{4}(\beta + (1 - z_i))^2 - \frac{1}{4}(\beta - (1 - z_i))^2$. Note that, since β is nonnegative by (26d), so is $z_2^{SOC}(\ell)$. We next relate the values $z_2^{SOC}(\ell)$, z_2^{NLP} , and $z_2^{LP}(\ell)$.

Proposition 10 $z_2^{NLP} \geq \sqrt{z_2^{SOC}(\ell)} + \ell \geq z_2^{LP}(\ell)$ for all $\ell \leq z_2^{NLP}$.

Proof: We first show that $z_2^{NLP} \geq \sqrt{z_2^{SOC}(\ell)} + \ell$. Let $(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{u}, \mathbf{w})$ be an optimal solution of (23). Consider $\beta' = (\mathbf{y} - \ell)^2$, $s_i = \max\{c^\top u^i - \ell z_i, 0\}$ and $t_i = \max\{c^\top w^i - \ell(1 - z_i), 0\}$. It is clear that $(\mathbf{x}, \beta', \mathbf{z}, \mathbf{u}, \mathbf{w}, \mathbf{s}, \mathbf{t})$

satisfies (20b) - (20e) and (26b), (26c), (26f). From (23b), (23c), and the fact that ℓ is a lower bound on z_2^{NLP} , we have

$$\begin{aligned} 0 &\leq \mathbf{y} - \ell, \\ c^\top u^i - \ell z_i &\leq (\mathbf{y} - \ell)z_i, \forall i \in \mathcal{N}, \\ c^\top w^i - \ell(1 - z_i) &\leq (\mathbf{y} - \ell)(1 - z_i), \forall i \in \mathcal{N}. \end{aligned}$$

Hence, since $z_i^2 \leq z_i$ for all $i \in \mathcal{N}$,

$$\begin{aligned} (s_i)^2 &\leq (\mathbf{y} - \ell)^2 z_i^2 \leq \beta' z_i, \forall i \in \mathcal{N}, \\ (t_i)^2 &\leq (\mathbf{y} - \ell)^2 (1 - z_i)^2 \leq \beta' (1 - z_i), \forall i \in \mathcal{N}. \end{aligned}$$

Thus, $(\mathbf{x}, \beta', \mathbf{z}, \mathbf{u}, \mathbf{w}, \mathbf{s}, \mathbf{t})$ also satisfies (26d) and (26e). Hence $z_2^{NLP} \geq \sqrt{z_2^{SOC}(\ell)} + \ell$.

Now we show that $\sqrt{z_2^{SOC}(\ell)} + \ell \geq z_2^{LP}(\ell)$. Let $(\mathbf{x}, \beta, \mathbf{z}, \mathbf{u}, \mathbf{w}, \mathbf{s}, \mathbf{t})$ be an optimal solution of (26). Consider $\mathbf{y}' = \sqrt{\beta} + \ell$. It is clear that $(\mathbf{x}, \mathbf{y}', \mathbf{z}, \mathbf{u}, \mathbf{w})$ satisfies (20b) - (20e). From (26b)-(26f), the fact that ℓ is a lower bound on z_2^{NLP} , and $z_i \leq 1$ for all $i \in \mathcal{N}$, we have

$$\begin{aligned} (\max\{c^\top u^i - \ell z_i, 0\})^2 &\leq (\mathbf{y}' - \ell)^2 z_i \leq (\mathbf{y}' - \ell)^2, \forall i \in \mathcal{N}, \\ (\max\{c^\top w^i - \ell(1 - z_i), 0\})^2 &\leq (\mathbf{y}' - \ell)^2 (1 - z_i) \leq (\mathbf{y}' - \ell)^2, \forall i \in \mathcal{N}. \end{aligned}$$

Taking square roots of the above inequalities we see that $(\mathbf{x}, \mathbf{y}', \mathbf{z}, \mathbf{u}, \mathbf{w})$ satisfies (24b) and (24c). Hence $\sqrt{z_2^{SOC}(\ell)} + \ell \geq z_2^{LP}(\ell)$. \square

Based on the above result we can extend the successive linear programming based approach established in Propositions 7 and 8 to one involving solving successive solutions of the SOCP (26). Also similar to (25), the SOCP (26) after enforcing integrality constraints on z and any integer constrained x variables leads to a “big-M free” mixed-integer SOCP (MISOCP) formulation for a general chance-constrained MILP.

Proposition 11

$$(v^* - \ell)^2 = \min \{ \beta : x \in \mathbb{R}^{n-r} \times \mathbb{Z}^r, z \in Z, (26b) - (26f), \text{and } (20b) - (20d) \}, \quad (27)$$

is a valid MISOCP formulation of (3).

6 Decomposition algorithms

In this section, we introduce a heuristic algorithm inspired by the bisection procedure for calculating the Lagrangian dual v_2^{LD} and also present two exact algorithms for solving CCSPs (3).

6.1 A heuristic scheme

The idea of our proposed heuristic algorithm is to use bisection to search for a value ℓ so that fixing $y = \ell$ in (13) may yield a feasible solution. Let $X^i = \{x \in \mathbb{R}^{n-r} \times \mathbb{Z}^r : G_i(x) \leq 0\}$ for all $i \in \mathcal{N}$ and let L and U be known initial lower and upper bounds on the optimal value of (13). For a fixed $y \in [L, U]$, say $y = (L + U)/2$, we consider the following optimization problem that minimizes the sum of infeasibilities for each scenario:

$$\min_{s, x} \sum_{i \in \mathcal{N}} p_i s_i \quad (28a)$$

$$\text{s.t. } G_i(x) \leq s_i \mathbf{e}, \quad i \in \mathcal{N}, \quad (28b)$$

$$c^\top x \leq y, \quad x \in S, \quad (28c)$$

$$x \in \mathbb{R}^{n-r} \times \mathbb{Z}^r, \quad s \in \mathbb{R}_+^N, \quad (28d)$$

where \mathbf{e} is a vector of all 1's. This problem is of the form of a two stage stochastic program with simple recourse and can benefit from specialized decomposition schemes for such problems. Given an optimal solution $(\hat{\mathbf{x}}, \hat{\mathbf{s}})$ of (28), we check if it is feasible to the original problem (13). We set $\hat{z}_i = \mathbb{I}(\hat{s}_i = 0)$ for all $i \in \mathcal{N}$. If $\sum_{i \in \mathcal{N}} p_i \hat{z}_i \geq 1 - \epsilon$, then $\hat{\mathbf{x}}$ is feasible to (13), and therefore y is a valid upper bound for (13). Then we can set $U = y$ and repeat the above steps to find a better feasible solution and hence a better upper bound. On the other hand, if $\sum_{i \in \mathcal{N}} p_i \hat{z}_i < 1 - \epsilon$, we set $L = y$ and repeat the above steps to try to find a feasible solution. Notice that L might not be a valid lower bound on v^* when the algorithm terminates. The detailed procedure is described in Algorithm 1.

Algorithm 1 A bisection-based heuristic.

```

1: Let  $L > -\infty$  and  $U < \infty$  be known lower and upper bounds for (13), let  $\delta > 0$  be the stopping tolerance
   parameter.
2: while  $U - L > \delta$  do
3:    $y \leftarrow (L + U)/2$ .
4:   Let  $(\hat{\mathbf{x}}, \hat{\mathbf{s}})$  be an optimal solution of (28) and set  $\hat{z}_i = \mathbb{I}(\hat{s}_i = 0)$  for all  $i \in \mathcal{N}$ .
5:   if  $\sum_{i=1}^N p_i \hat{z}_i \geq 1 - \epsilon$  then
6:      $U \leftarrow y$ .
7:   else
8:      $L \leftarrow y$ .
9:   end if
10: end while

```

Let v^H denote the solution given by Algorithm 1. We next show that $0 \leq v^H - v^* \leq \delta$ under certain conditions.

Proposition 12 Suppose $G_i(x) : S \rightarrow \mathbb{R}_{-}^{m_i} \cup \mathcal{R}_M^i, \forall i \in \mathcal{N}$, where $\mathcal{R}_M^i = \{s \in \mathbb{R}^{m_i} : \|s\|_{\infty} = M\}$ for all $i \in \mathcal{N}$, $M \in \mathbb{R}_{+}$. Then Algorithm 1 returns a feasible solution with $v^H - v^* \leq \delta$, where $\delta > 0$ is the chosen stopping tolerance parameter.

Proof:

First of all, we observe that for any optimal solution $(\mathbf{x}^*, \mathbf{s}^*)$ of (28) with a given y , s_i^* is either 0 or M for each $i \in \mathcal{N}$. Indeed, if $G_i(\mathbf{x}^*) \leq 0$, then $s_i^* = 0$; otherwise, the smallest s_i^* that can be chosen is M since $G_i(\mathbf{x}^*) \in \mathcal{R}_M^i$.

Suppose that $(\hat{\mathbf{x}}, \hat{\mathbf{s}})$ is an optimal solution of (3). Let $\hat{\mathbf{s}} = M\mathbf{e} - M\hat{\mathbf{z}}$, and $(\hat{\mathbf{x}}, \hat{\mathbf{s}})$ is a feasible solution to (28) with any $y \geq v^*$ and $\sum_{i \in \mathcal{N}} p_i \hat{s}_i / M \leq \epsilon$. Thus, $v^H \leq v^* + \delta$. □

The conditions of Proposition 12 are identical to those in Proposition 5, and the chance-constrained set covering problems (18) satisfy these conditions. Note also that for this problem class with an integer cost vector we can choose $\delta < 1$ and recover an exact optimal solution.

6.2 A scenario decomposition algorithm for chance-constrained 0-1 programs

For two-stage stochastic programs in which the first-stage variables are all binary [1] presented a scenario decomposition algorithm that uses the nonanticipative Lagrangian dual of such problems. In this approach, feasible solutions from the scenario subproblems are used to update the upper bound. We describe a simple extension of this method to solve chance-constrained 0-1 programs, which can take advantage of the new Lagrangian dual problems proposed in Section 3. Exactly solving the Lagrangian dual problems (8) and (17) may be challenging in computation. However, the scenario decomposition algorithm remains valid even if the Lagrangian dual multipliers are not optimal. In a practical implementation, we may settle with a lower bound of v_1^{LD} , or v_2^{LD} . For example, we may simply use the quantile bound v^Q , or even a valid lower bound from the scenario grouping based relaxation (5).

Algorithm 2 provides a description of the scenario decomposition approach. Finite convergence of Algorithm 2 is an immediate consequence of the following three facts: the lower bound is nondecreasing; a

Algorithm 2 Scenario decomposition algorithm

```

1: Let  $UB$  be a known upper bound, let  $LB \leftarrow -\infty$ ,  $E = \emptyset$ , and let  $\delta > 0$  be the stopping tolerance
   parameter.
2: while  $UB - LB > \delta$  do
3:   Calculate a lower bound  $\underline{v}$  for the Lagrangian dual  $v_1^{LD}$  or the quantile-based Lagrangian dual  $v_2^{LD}$ ;
4:   Collect the optimal solutions  $\hat{x}^i, i \in \mathcal{N}$  that correspond to  $\underline{v}$ , let  $E = \cup_{i \in \mathcal{N}} \hat{x}^i$ .
5:   Update  $LB \leftarrow \max \{LB, \underline{v}\}$ .
6:   for  $x \in E$  do
7:     if  $x$  is feasible and  $c^\top x < UB$  then
8:        $UB \leftarrow c^\top x$ .
9:     end if
10:    end for
11:    Let  $X^i = X^i \setminus E, \forall i \in \mathcal{N}$ .
12: end while

```

feasible solution is never excluded if it has not been evaluated; and there are finitely many feasible solutions since for each scenario $i \in \mathcal{N}$, X^i is a finite set. Implementation of the update of the set X^i in line 11 can be accomplished with “no good” cut based on the assumption that all the x variables are binary; see [1] for details.

6.3 Branch-and-cut algorithms based on primal MILP formulations

Section 5 provided two MILP formulations (21) and (25) for (3) when the objective function and all constraints are linear. These formulations have a set of variables and constraints for each scenario $i \in \mathcal{N}$, so solving them directly may be time-consuming. We therefore apply a branch-and-cut framework to solve these two MILPs. Specifically, a Benders decomposition algorithm is applied in which the master problem consists of the decision variables x and the scenario decision variables z , and in the case of (25) the variable y . The decision variables u^i and w^i for $i \in \mathcal{N}$ are projected out, and Benders feasibility cuts are added to iteratively build the projected feasible region. The generation of Benders feasibility cuts decomposes into a separate subproblem for each scenario. The detailed development of the cuts and the resulting branch-and-cut algorithms are presented in the Appendix.

7 Numerical illustration

In this section we evaluate the proposed bounds and algorithms on two sets of multi-dimensional knapsack instances, *mk-20-10* and *mk-39-5*, from [38] (these instances are named *1-4-multi* and *1-6-multi*, respectively, in [38]). Instance name *mk-n-m* indicates the instance has n decision variables and m constraints in each scenario. For both sets of instances, we consider four different scenario sizes: $N \in \{100, 500, 1000, 3000\}$. Under each scenario size (e.g., $N = 100$), we perform five different replications. Since the results among different replications are similar, we report averages over the five replications. We consider two different types of variables x : continuous and binary. The deterministic feasible set is $S = [0, 1]^n$ for the instances with continuous x variables and $S = \{0, 1\}^n$ for those with binary x variables. For each scenario $i \in \mathcal{N}$, the feasible set is $X^i := \{x \in S : A^i x \leq b^i\}$, where $A^i \in \mathbb{R}_+^{m \times n}$, $b^i \in \mathbb{R}_+^m$ is the i th realization of random data ξ , and the objective is to minimize $-c^\top x$, where the coefficient vector $c \in \mathbb{R}_+^n$. In all our experiments, we set the number of threads to one, and we use the default optimality gap tolerance in CPLEX for solving MIP models, which is 0.01%.

7.1 Illustration of bound quality on instances with continuous x

We first present a numerical illustration on the strength of the Lagrangian dual bounds on instances with continuous x , i.e., when $S = [0, 1]^n$. We compare the proposed Lagrangian bounds v_1^{LD} and v_2^{LD} with the LP relaxation bound $v^C(M)$ ($\underline{v}^C(M)$ and $\bar{v}^C(M)$ are identical in this case), quantile bound v^Q , and the optimal

Table 1: Bound comparison for multi-dimensional continuous knapsack instances.

Instance	ϵ	N	No big-M Str.			With big-M Str.			v^Q	v^H
			$v^C(M)$	v_1^{LD}	v_2^{LD}	$v^C(M)$	v_1^{LD}	v_2^{LD}		
mk-20-10	0.1	100	10.1%	7.3%	1.3%	2.3%	2.3%	1.2%	2.4%	0.4%
		500	10.0%	7.0%	1.4%	2.4%	2.4%	1.2%	2.1%	0.2%
		1000	10.0%	7.3%	1.6%	2.5%	2.5%	1.4%	2.5%	0.3%
		3000	9.8%	7.2%	1.7%	2.6%	2.6%	1.5%	2.5%	0.2%
	0.2	100	14.5%	10.5%	1.3%	3.0%	3.0%	1.1%	2.0%	0.6%
		500	14.7%	10.3%	1.4%	3.0%	2.9%	1.3%	2.1%	0.3%
		1000	14.8%	10.7%	1.7%	3.2%	3.2%	1.5%	2.5%	0.3%
		3000	14.4%	10.5%	1.8%	3.2%	3.2%	1.6%	2.6%	0.1%
mk-39-5	0.1	100	8.0%	7.4%	2.0%	2.0%	2.0%	1.5%	3.2%	0.6%
		500	8.9%	8.3%	2.5%	2.6%	2.5%	2.0%	3.8%	0.3%
		1000	8.8%	8.3%	2.6%	2.6%	2.6%	2.1%	3.9%	0.4%
		3000	8.7%	8.3%	2.9%	2.8%	2.8%	2.3%	4.3%	0.1%
	0.2	100	11.4%	10.7%	2.1%	2.7%	2.7%	1.8%	3.3%	0.4%
		500	12.4%	11.7%	2.6%	3.4%	3.3%	2.2%	3.5%	0.3%
		1000	12.4%	11.7%	2.9%	3.5%	3.4%	2.4%	4.0%	0.2%
		3000	12.1%	11.6%	3.0%	3.5%	3.5%	2.5%	4.2%	0.1%

objective value v^* . Since x is continuous, from Theorems 6 and 7, the Lagrangian dual bounds v_1^{LD}, v_2^{LD} are equal to z_1^{LP}, z_2^{NLP} , respectively. Therefore, we compute v_1^{LD} using (20), and when computing $v_2^{LD}(z_2^{NLP})$, we start with the quantile bound v^Q , then solve the primal LP (24) iteratively using Benders decomposition.

For v_1^{LD}, v_2^{LD} and $v^C(M)$, we report bounds that are obtained with and without big- M strengthening, a coefficient strengthening procedure introduced in [38] for CCSPs. Without any big- M strengthening, a *naive big- M* parameter for formulation (4) is calculated by: $M_{ij} = \max\{A_j^i x - b_j^i : x \in S\}$, for all $i \in \mathcal{N}, j = 1, 2, \dots, m_i$. At the other extreme, one could obtain a strengthened big- M parameter M_{kl} for scenario $k \in \mathcal{N}$, constraint l by solving an MIP that has the same structure as formulation (4) for the original CCSP: $M_{kl} = \max\{A_l^k x - b_l^k : A^i x \leq b^i + M_i(1 - z_i), \forall i \in \mathcal{N}, x \in S, z \in Z\}$, which would be too time-consuming. The idea of the big- M strengthening is to obtain a relaxation bound of this MIP using the quantile bound presented in Section 2.2. The strengthened big- M parameters M_i 's are useful for formulations other than (4) as well, where the scenario-independent feasible set S is strengthened by valid inequalities $A^i x \leq M_i + b^i$ for all $i \in \mathcal{N}$.

In Table 1, we show the optimality gaps of $v^C(M), v^Q, v_1^{LD}$ and v_2^{LD} in the columns labeled accordingly, where optimality gap for a given lower bound LB is defined as $(v^* - LB)/|v^*|$. Under each lower bound, the columns under label ‘With big- M Str.’ provides the bounds obtained if the big- M coefficients have been strengthened, and the columns under label ‘No big- M Str.’ provides the bounds obtained without strengthening big- M coefficients. We also show the optimality gap of the upper bound UB given by the heuristic algorithm described in Section 6.1 in column labeled v^H , which is calculated as $(UB - v^*)/|v^*|$. In Table 2, we present the average computation time, in seconds, for obtaining each of these bounds. In addition, the column ‘M-T’ displays the time spent on the big- M strengthening procedure, which is a pre-processing step required for calculating the bounds under the ‘With big- M Str.’ columns. (Thus, the total time for calculating such a bound is the sum of the ‘M-T’ and the bound time.)

We can see from Table 1 that strengthening big- M parameters significantly improves the bounds given by v_1^{LD} and $v^C(M)$. Without strengthening big- M parameters, bounds given by v_1^{LD} and $v^C(M)$ are rather weak, especially when a higher risk tolerance parameter $\epsilon = 0.2$ is used. With strengthened big- M parameters, the difference between v_1^{LD} and $v^C(M)$ is very small. On the other hand, the bound improvement by strengthening big- M parameters is modest for v_2^{LD} , since v_2^{LD} already gives a tight bound even without strengthening big- M parameters. Overall, the best bounds are obtained by using strengthened big- M parameters and v_2^{LD} . We also find that the heuristic scheme yields a very small optimality gap. For large instances where the exact optimal solution may be challenging to find, one may accept the heuristic solu-

Table 2: Computational times for computing bounds for multi-dimensional continuous knapsack instances.

Instance	ϵ	N	No big- M Str.			With big- M Str.			M-T	v^H
			$v^C(M)$	v_1^{LD}	v_2^{LD}	$v^C(M)$	v_1^{LD}	v_2^{LD}		
mk-20-10	0.1	100	0.0	0.8	2.6	0.0	0.4	0.8	0.1	0.0
		500	0.2	10.5	26.3	0.0	3.4	6.2	3.7	0.2
		1000	0.5	34.6	83.7	0.2	11.8	21.5	14.6	0.9
		3000	4.1	259.2	683.1	2.0	84.4	151.9	132.2	9.5
	0.2	100	0.0	1.0	3.2	0.0	0.5	1.3	0.1	0.0
		500	0.2	10.9	42.0	0.1	3.6	12.5	3.7	0.3
		1000	0.5	39.8	143.4	0.2	13.7	39.0	14.6	1.2
		3000	4.5	301.8	1338.4	2.3	93.5	292.0	132.2	11.9
mk-39-5	0.1	100	0.0	1.0	5.4	0.0	0.5	1.2	0.1	0.0
		500	0.1	8.3	40.5	0.1	3.9	9.1	3.5	0.3
		1000	0.5	31.4	127.1	0.2	12.9	29.7	13.9	0.8
		3000	5.4	222.6	961.2	1.1	84.8	187.5	125.5	7.4
	0.2	100	0.0	1.1	8.5	0.0	0.7	2.2	0.1	0.0
		500	0.1	8.7	62.1	0.1	5.3	18.7	3.5	0.3
		1000	0.5	36.3	224.8	0.3	18.0	55.0	13.9	1.0
		3000	4.7	242.7	1636.0	2.4	122.6	382.6	125.2	9.3

tion v^H , when the gap between the lower bound given by v_2^{LD} and the upper bound given by v^H is small enough. From Table 2 we see that we can obtain these strong bounds in a small amount of time. Interestingly, we see that formulations with naive big- M parameters take longer to solve than the ones with strengthened big- M parameters, even after including the time spent on strengthening the big- M parameters. Thus, for these instances, big- M strengthening yields improvements in both computation time and bound.

7.2 Illustration of the branch-and-cut algorithm on instances with continuous x

In this section we describe computational experiments using the branch-and-cut approach described in Section 6.3 for solving formulation (25) on instances with continuous x . For all experiments in our test, we use a time limit of 3600 seconds. We turn off the CPLEX presolve for the branch-and-cut algorithm. We use the heuristic solution obtained by the heuristic algorithm in Section 6.1 as an MIP start solution. Further implementation details of the branch-and-cut algorithm are given in the Appendix.

In Table 3 we compare the performances of three computational options: the MIP (4) using strengthened big- M parameters (MIP-(4)), the branch-and-cut algorithm with strengthened big- M parameters (Benders With big- M), and the branch-and-cut algorithm without strengthened big- M parameters (Benders without big- M). For instances that are not solved to optimality within the time limit, we show in parentheses the number of instances out of five replications that are solved to optimality, and report the average optimality gap. For these instances, we use the number of nodes that have been processed up to the time limit to calculate the average number of nodes.

We observe from Table 3 that the performance of the branch-and-cut algorithm is improved by using strengthened big- M parameters. This is consistent with what has been shown in Table 1. Also, we have seen from Table 1 that the root relaxation bound for the branch-and-cut algorithm is tighter than the MIP formulation (4). However, this advantage at the root node does not lead to a significant improvement in terms of the total computation time for solving these instances to optimality. We observe that for those instances that could be solved within the time limit, the computational times for the big- M formulation and the branch-and-cut algorithm are similar, although for instances with the largest number of scenarios in our experiments, the branch-and-cut algorithm yields much smaller optimality gaps. We also notice that both methods seem having more difficulty in solving the instances with a higher risk tolerance ϵ . It appears that branching in the Benders formulation is less effective than that in the MIP formulation (4) and thus

Table 3: Computational results for MIP formulation (4) and branch-and-cut algorithm on multi-dimensional continuous knapsack instances.

Instance	ϵ	N	MIP-(4)		Benders w/o big- M		Benders With big- M	
			AvgT	AvgN	AvgT	AvgN	AvgT	AvgN
mk-20-10	0.1	100	0.2	159	2.1	444	0.9	266
		500	52.5	19k	376.6	93k	36.8	35k
		1000	2004.9	356k	0.8%(0)	>210k	0.2%(4)	>788k
		3000	1.8%(0)	>205k	1.6%(0)	>111k	1.3%(0)	>546k
	0.2	100	0.7	664	8.8	1161	1.7	473
		500	452.8	166k	0.5%(1)	>297k	760.4	459k
		1000	0.6%(0)	>610k	1.7%(0)	>145k	0.6%(0)	>1007k
		3000	2.5%(0)	>180k	2.8%(0)	>65k	1.9%(0)	>344k
mk-39-5	0.1	100	0.3	350	5.4	2231	1.4	917
		500	781.1	393k	0.2%(1)	>383k	339.3	263k
		1000	0.5%(0)	>905k	1.3%(0)	>195k	0.6%(0)	>1237k
		3000	2.1%(0)	>250k	1.7%(0)	>96k	1.4%(0)	>412k
	0.2	100	1.2	2294	31.8	9022	4.5	3294
		500	0.3%(0)	>1729k	1.4%(0)	>229k	0.4%(0)	>1441k
		1000	1.4%(0)	>893k	2.4%(0)	>168k	1.7%(0)	>832k
		3000	2.8%(0)	>215k	3.0%(0)	>61k	2.3%(0)	>257k

more nodes are explored. This motivates further study on effective ways to take advantage of the strong relaxation bound v_2^{LD} for solving CCSPs to optimality.

7.3 Performance on instances with binary x

We next consider the binary instances, i.e., with $S = \{0, 1\}^n$. We compare the proposed dual bounds and also illustrate the effectiveness of the heuristic algorithm (Algorithm 1), the scenario decomposition algorithm (Algorithm 2) with and without scenario grouping, and the MIP formulation (4) with strengthened big- M coefficients. For the scenario decomposition algorithm, the lower bounds are obtained by v^Q . For the scenario grouping, the number of groups K is chosen as the smallest divisor of N that is larger than $\lceil \epsilon N \rceil$ and the scenarios are divided into K groups with the same size.

Table 4 summarizes the optimality gaps of $\underline{v}^C(M)$, z_1^{LP} , z_2^{NLP} , v^Q , v^{QG} and v^H , where v^{QG} denotes the results of the grouping quantile bound. For these instances, we report only gaps obtained with strengthened big- M coefficients. As we do not have the optimal solutions for most of the mk-39-5 instances, we use the best known upper bound and lower bound to estimate the lower bound and heuristic gaps, respectively. Table 5 displays the time to compute these bounds. (Note that, the times for computing $\underline{v}^C(M)$, z_1^{LP} , and z_2^{NLP} are the times for $v^C(M)$, v_1^{LD} , and v_2^{LD} from Table 2 for the continuous x case, since these are equivalent.)

In Table 4, we see that the best Lagrangian dual bounds corresponding to continuous x still have at most 4% optimality gap, which demonstrates the effectiveness of these bounds. In addition, the quantile bound, which is obtained by solving binary IP subproblems, is somewhat stronger than any of the bounds $\underline{v}^C(M)$, z_1^{LP} , and z_2^{NLP} . On the other hand, we see from Table 2 that the quantile bound v^Q takes longer to calculate. However, when we apply the quantile bound to the scenario grouping relaxation, the resulting bounds v^{QG} are comparable with the quantile bound obtained without grouping, but take much shorter time to compute (see Table 5).

We observe in Tables 4 and 5 that for the mk-20-10 instances the heuristic performs extremely well in terms of both quality (zero optimality gap) and solution time. For the larger instances, the optimality gaps are not exact since we cannot obtain the optimal objective values within the time limit, but we can see that the bounds obtained from the heuristic methods are still quite close to the optimal ones. Thus, the solution from the heuristic method could be treated as a good starting point for other algorithms.

Table 4: Bound comparison for multi-dimensional binary knapsack instances.

Instances	ϵ	N	$\underline{v}^C(M)$	z_1^{LP}	z_2^{NLP}	v^Q	v^{QG}	v^H
mk-20-10	0.1	100	3.5%	3.5%	2.3%	1.6%	1.1%	0.0%
		500	3.8%	3.8%	2.6%	1.8%	1.5%	0.0%
		1000	3.8%	3.8%	2.7%	2.0%	1.8%	0.0%
		3000	3.8%	3.8%	2.7%	2.0%	1.7%	0.0%
	0.2	100	4.8%	4.7%	2.8%	2.3%	2.4%	0.0%
		500	3.9%	3.9%	2.2%	1.5%	1.6%	0.0%
		1000	4.4%	4.4%	2.7%	2.2%	2.2%	0.0%
		3000	4.4%	4.4%	2.7%	2.1%	2.0%	0.0%
mk-39-5	0.1	100	3.2%	3.2%	2.7%	3.3%	2.2%	0.6%
		500	$\leq 3.9\%^*$	$\leq 3.9\%$	$\leq 3.3\%$	$\leq 3.9\%$	$\leq 3.1\%$	$\leq 0.3\%$
		1000	$\leq 4.0\%$	$\leq 4.0\%$	$\leq 3.5\%$	$\leq 4.1\%$	$\leq 3.4\%$	$\leq 1.6\%$
		3000	$\leq 4.0\%$	$\leq 4.0\%$	$\leq 3.5\%$	$\leq 4.2\%$	$\leq 3.3\%$	$\leq 2.7\%$
	0.2	100	3.9%	3.9%	2.9%	3.4%	3.0%	0.4%
		500	$\leq 4.2\%$	$\leq 4.2\%$	$\leq 3.0\%$	$\leq 3.3\%$	$\leq 3.6\%$	$\leq 1.1\%$
		1000	$\leq 4.4\%$	$\leq 4.3\%$	$\leq 3.3\%$	$\leq 3.8\%$	$\leq 4.0\%$	$\leq 2.3\%$
		3000	$\leq 4.5\%$	$\leq 4.5\%$	$\leq 3.5\%$	$\leq 4.0\%$	$\leq 3.8\%$	$\leq 3.2\%$

* A “ \leq ” indicates instances for which the optimal value is not known, and the associated number represents an upper bound on the true optimality gap.

Table 5: Computational times for computing bounds for multi-dimensional binary knapsack instances.

Instance	ϵ	N	$\underline{v}^C(M)$	z_1^{LP}	z_2^{NLP}	v^Q	v^{QG}	v^H
mk-20-10	0.1	100	0.0	0.4	0.8	22.8	5.8	1.9
		500	0.0	3.4	6.2	136.0	31.7	12.6
		1000	0.2	11.8	21.5	282.8	62.5	47.6
		3000	2.0	84.4	151.9	770.4	134.8	285.8
	0.2	100	0.0	0.5	1.3	22.8	5.8	1.9
		500	0.1	3.6	12.5	136.0	31.7	11.7
		1000	0.2	13.7	39.0	282.8	62.5	48.1
		3000	2.3	93.5	292.0	770.4	134.8	340.6
mk-39-5	0.1	100	0.0	0.5	1.2	31.8	6.7	2.7
		500	0.1	3.9	9.1	157.7	35.2	12.8
		1000	0.2	12.9	29.7	327.0	73.5	39.2
		3000	1.1	84.8	187.5	992.0	191.7	212.3
	0.2	100	0.0	0.7	2.2	31.8	6.7	2.4
		500	0.1	5.3	18.7	157.7	35.2	12.0
		1000	0.3	18.0	55.0	327.0	73.5	39.6
		3000	2.4	122.6	382.6	992.0	191.7	220.6

Table 6: Performance of scenario decomposition and MIP on the multi-dimensional binary knapsack instances.

Instances			Scenario decomposition				MIP (4) With Big- M Str.			
Instance	ϵ	N	Grouping		Non-grouping		M	Time	Tot. Time	Gap
			Time	Gap	Time	Gap				
mk-20-10	0.1	100	23.3	0.0%(5)	111.4	0.0%(5)	2.4	3.4	3.4	0.0%(5)
		500	82.6	0.0%(5)	371.9	0.0%(5)	47.5	54.8	54.8	0.0%(5)
		1000	119.9	0.0%(5)	972.0	0.0%(5)	185.6	207.1	207.1	0.0%(5)
		3000	358.8	0.0%(5)	2253.1	0.0%(5)	1656.8	1837.4	1837.4	0.0%(5)
	0.2	100	52.9	0.0%(5)	173.0	0.0%(5)	2.9	3.7	3.7	0.0%(5)
		500	122.9	0.0%(5)	357.1	0.0%(5)	47.4	56.8	56.8	0.0%(5)
		1000	232.3	0.0%(5)	1271.8	0.0%(5)	185.7	224.4	224.4	0.0%(5)
		3000	719.0	0.0%(5)	2017.3	0.0%(5)	1681.1	3281.4	3281.4	1.4%(1)
mk-39-5	0.1	100	3600.0	2.0%(0)	3600.0	3.6%(0)	1.0	4.7	4.7	0.0%(5)
		500	3600.0	2.4%(0)	3600.0	3.9%(0)	24.8	2619.9	2619.9	0.1%(3)
		1000	3600.0	2.6%(0)	3600.0	4.0%(0)	98.8	3600.0	3600.0	1.5%(0)
		3000	3600.0	3.0%(0)	3600.0	3.2%(0)	878.8	3600.0	3600.0	2.7%(0)
	0.2	100	3600.0	2.6%(0)	3600.0	3.4%(0)	1.0	15.1	15.1	0.0%(5)
		500	3600.0	2.8%(0)	3600.0	3.2%(0)	24.7	3600.0	3600.0	1.0%(0)
		1000	3600.0	3.6%(0)	3600.0	3.8%(0)	97.3	3600.0	3600.0	2.3%(0)
		3000	3600.0	3.4%(0)	3600.0	3.7%(0)	878.9	3600.0	3600.0	3.3%(0)

Table 6 presents the results of solving these instances to optimality using the scenario decomposition algorithm (Algorithm 2) with and without scenario grouping, and the MIP formulation (4) with strengthened big- M coefficients. We find that the scenario grouping based decomposition method significantly outperforms the one without grouping in terms of computational time, for the instances solved within the time limit, and ending optimality gaps, for the remaining instances. From Table 6, we further observe that when the number of scenarios is small (e.g., not larger than 1000), the MIP formulation (4) with strengthened big- M parameters gives the best performance among these three methods. However, when the number of scenarios is larger, the MIP formulation (4) could not close the optimality gap within the time limit, while the scenario decomposition method with grouping can still solve all of the mk-20-10 instances within 15 minutes. We also observe that neither method is able to solve the majority of the mk-39-5 instances. This negative result for the scenario decomposition approach on problems with a larger x space is somewhat expected, since only the simple “no good cuts” are employed to drive the convergence. This motivates future work to find better ways to incorporate the strong Lagrangian dual bound into the scenario decomposition approach.

8 Conclusion

We studied two new Lagrangian dual problems for chance-constrained stochastic programs and developed their associated primal formulations which can be used to exactly compute these dual bounds for chance-constrained linear programs, or a lower bound on them for chance-constrained mixed integer programs. We also proposed a new heuristic method and two new exact algorithms that make use of these new bounds to solve these problems to optimality. In our numerical study, we find that for all of our instances, the dual bounds can be quickly computed and demonstrate that heuristic solutions are within 4% of optimal. Our exact algorithms are able to solve more than half of the instances to optimality, although

there remain some challenging unsolved instances. Thus, we see it as a direction for future work to investigate alternative means for using these strong bounds to solve these problems to optimality and also seek better ways of excluding solutions as well as optimally grouping scenarios for the scenario decomposition approach. In addition, for chance-constrained mixed integer programs, as opposed to using the (weaker) primal formulations as we have done in this work, it would be interesting to explore efficient techniques for directly computing the proposed dual bounds using, e.g., the recently proposed inexact bundle methods with on-demand accuracy in [13, 14].

References

- [1] S. Ahmed. A scenario decomposition algorithm for 0-1 stochastic programs. *Oper. Res. Lett.*, 41:555–569, 2013.
- [2] S. Ahmed and D. J. Papageorgiou. Probabilistic set covering with correlations. *Oper. Res.*, 61(2):438–452, 2013.
- [3] X. Bai, J. Sun, X. Sun, and X. Zheng. An alternating direction method for chance-constrained optimization problems with discrete distributions. Available at http://www.optimization-online.org/DB_FILE/2012/04/3448.pdf, 2012.
- [4] E. Balas. A modified lift-and-project procedure. *Math. Program.*, 79:19–31, 1997.
- [5] P. Beraldi and A. Ruszcynski. The probabilistic set-covering problem. *Oper. Res.*, 50(6):956–967, 2002.
- [6] G. Calafiore and M. Campi. Uncertain convex programs: randomized solutions and confidence levels. *Math. Program.*, 102:25–46, 2005.
- [7] G. Calafiore and M. Campi. The scenario approach to robust control design. *IEEE Trans. Automat. Control*, 51:742–753, 2006.
- [8] J. D. Camm, A. S. Raturi, and S. Tsubakitani. Cutting big m down to size. *Interfaces*, 20(5):61–66, 1990.
- [9] C. C. Carøe and R. Schultz. Dual decomposition in stochastic integer programming. *Oper. Res. Lett.*, pages 37–45, 1999.
- [10] S. Ceria and J. Soares. Convex programming for disjunctive convex optimization. *Math. Program.*, 86(3):595–614, 1999.
- [11] A. Charnes and W. W. Cooper. Deterministic equivalents for optimizing and satisficing under chance constraints. *Oper. Res.*, 11:18–39, 1963.
- [12] A. Charnes, W. W. Cooper, and G. H. Symonds. Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil. *Manage. Sci.*, 4:235–263, 1958.
- [13] W. de Oliveira and C. Sagastizábal. Level bundle methods for oracles with on demand accuracy. *Optimization Methods and Software*, 29(6):1180–1209, 2014.
- [14] W. de Oliveira, C. Sagastizábal, and C. Lemaréchal. Convex proximal bundle methods in depth: a unified analysis for inexact oracles. *Math. Program.*, 148:241–277, 2014.
- [15] D. Dentcheva. Optimisation models with probabilistic constraints, chapter 4 in [25], mps-siam series on optimization. *SIAM and MPS, Philadelphia*, 2009.
- [16] D. Dentcheva and G. Martinez. Regularization methods for optimization problems with probabilistic constraints. *Math. Program.*, 138(1-2):223–251, 2013.
- [17] D. Dentcheva and W. Römisch. Duality gaps in nonconvex stochastic optimization. *Math. Program.*, 101:515–535, 2004.

- [18] M. Fischetti, A. Lodi, and A. Tramontani. On the separation of disjunctive cuts. *Math. Program.*, 128:205–230, 2011.
- [19] R. Henrion. A critical note on empirical (sample average, monte carlo) approximation of solutions to chance constrained programs. In *System Modelling and Optimization*, pages 25–37. Springer, 2011.
- [20] R. Henrion and A. Möller. A gradient formula for linear chance constraints under gaussian distribution. *Math. Oper. Res.*, 37(3):475–488, 2012.
- [21] S. Küçükyavuz. On mixing sets arising in chance-constrained programming. *Math. Program.*, 132(1):31–56, 2012.
- [22] M. A. Lejeune. Pattern-based modeling and solution of probabilistically constrained optimization problems. *Oper. Res.*, 60(6):1356–1372, 2012.
- [23] C. Lemaréchal and A. Renaud. A geometric study of duality gaps, with applications. *Math. Program.*, 90(3):399–427, 2001.
- [24] X. Liu, S. Küçükyavuz, and J. Luedtke. Decomposition algorithms for two-stage chance-constrained programs. *Math. Program.*, 2014.
- [25] J. Luedtke. A branch-and-cut decomposition algorithm for solving chance-constrained mathematical programs with finite support. *Math. Program.*, 2013. DOI 10.1007/s10107-013-0684-6.
- [26] J. Luedtke and S. Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM J. Optim.*, 19:674–699, 2008.
- [27] J. Luedtke, S. Ahmed, and G. L. Nemhauser. An integer programming approach for linear programs with probabilistic constraints. *Math. Program.*, 12:247–272, 2010.
- [28] T. Magnanti, J. Shapiro, and M. Wagner. Generalized linear programming solves the dual. *Management science*, 22(11):1195–1203, 1976.
- [29] B. Pagnoncelli, S. Ahmed, and A. Shapiro. The sample average approximation method for chance constrained programming:theory and applications. *J. Optim. Theory Appl.*, 142:399–416, 2009.
- [30] A. Prékopa. *Stochastic programming*. Kluwer, Dordrecht, 1995.
- [31] F. Qiu, S. Ahmed, S. S. Dey, and L. A. Wolsey. Covering linear programming with violations. *INFORMS Journal on Computing*, 2014.
- [32] R. T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.
- [33] R. T. Rockafellar and S. Uryasev. Conditional value-at-risk for general loss distributions. *Journal of Banking & Finance*, 26(7):1443–1471, 2002.
- [34] R. T. Rockafellar and R.-B. Wets. Nonanticipativity and \mathcal{L}^1 -martingales in stochastic optimization problems. In *Stochastic Systems: Modeling, Identification and Optimization, II*, pages 170–187. Springer, 1976.
- [35] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, Philadelphia, 2009.
- [36] M. Slater. Lagrange multipliers revisited. Technical report, Cowles Foundation for Research in Economics, Yale University, 1959.
- [37] Y. Song and J. Luedtke. Branch-and-cut approaches for chance-constrained formulations of reliable network design problems. *Math. Program. Comput.*, 5(4):397–432, 2013.
- [38] Y. Song, J. Luedtke, and S. Küçükyavuz. Chance-constrained binary packing problems. *INFORMS J. Comput.*, 2014.

- [39] W. van Ackooij. Decomposition approaches for block-structured chance-constrained programs with application to hydro-thermal unit commitment. *Math. Meth. Oper. Res.*, 80(3):227–253, 2014.
- [40] J. P. Watson, R. J.-B. Wets, and D. L. Woodruff. Scalable heuristics for a class of chance-constrained stochastic programs. *INFORMS J. Comput.*, 22:543–554, 2010.
- [41] M. Zhang, S. Küçükyavuz, and S. Goel. A branch-and-cut method for dynamic decision under joint chance constraints. *Manage. Sci.*, 2014.

Appendix: Implementation details of the branch-and-cut algorithms for solving the primal MILP formulations (21) and (25)

Section 5 provided two MILP formulations (21) and (25) for (3) when the objective function and all constraints are linear. These formulations have a set of variables and constraints for each scenario $i \in \mathcal{N}$, so solving them directly may be time-consuming. We therefore apply a branch-and-cut framework to solve these two MILPs.

We first propose a branch-and-cut algorithm for solving (21). Given a solution $(\bar{x}, \bar{z}) \in \mathbb{R}^m \times [0, 1]^N$, checking its feasibility to the LP relaxation of (21) is equivalent to checking the existence of $\{(u^i, w^i)\}_{i \in \mathcal{N}}$ that satisfies (20b) - (20d) with (\bar{x}, \bar{z}) . This can be done by solving the following feasibility problem *separately* for each scenario $i \in \mathcal{N}$, where the dual variables associated with each set of constraints are denoted in parenthesis:

$$f^1(i) = \min_{u, w, \rho} \rho \quad (29a)$$

$$\text{s.t. } A^i u^i - \rho \mathbf{e} \leq b^i \bar{z}_i, \quad (\gamma) \quad (29b)$$

$$D u^i - \rho \mathbf{e} \leq d \bar{z}_i, \quad (\pi_1) \quad (29c)$$

$$D w^i - \rho \mathbf{e} \leq d(1 - \bar{z}_i), \quad (\pi_2) \quad (29d)$$

$$u^i + w^i = \bar{x}, \quad (\alpha) \quad (29e)$$

$$\rho \geq 0. \quad (29f)$$

The form of the feasibility problem (29) is chosen to impose a particular normalization on the generated cut, inspired by work on cut-generating linear programs for split cuts in MIP [4, 18].

If $f^1(i) = 0$, then no cut is generated for scenario i . Otherwise, let $(\bar{\gamma}, \bar{\pi}_1, \bar{\pi}_2, \bar{\alpha})$ be an optimal dual solution of (29). Then $(\bar{x}, \bar{y}, \bar{z})$ is cut off by the following Benders feasibility cut:

$$\bar{\alpha}^\top x + (\bar{\gamma}^\top b_i + \bar{\pi}_1^\top d) z_i + \bar{\pi}_2^\top d(1 - z_i) \leq 0. \quad (30)$$

This motivates a branch-and-cut algorithm for solving (21) based on Benders decomposition, by treating x and z variables as the first-stage variables, and $\{(u^i, w^i)\}_{i \in \mathcal{N}}$ as the second-stage auxiliary variables. The LP relaxation of the master program is given by:

$$\min_{x, z} c^\top x \quad (31a)$$

$$\text{s.t. } \sum_{i=1}^N p_i z_i \geq 1 - \epsilon, \quad (31b)$$

$$\pi^k z + \alpha^k x \geq \beta^k, \quad k = 1, \dots, K \quad (31c)$$

$$D x \leq d, \quad (31d)$$

$$z \in [0, 1]^N, \quad (31e)$$

where (31c) are Benders feasibility cuts that have been added in the process of solving the LP relaxation (or throughout the branch-and-cut tree). In each of the Benders cuts (31c), obtained from the projection from (29), only one scenario variable z_i has a nonzero coefficient. The constraints (31d) are actually redundant

because the cuts (31c) will eventually enforce these. However, including them may help improve convergence of the cutting plane algorithm for solving the relaxation, and also enables generating additional cuts in the case when there are some integer restrictions on x .

Similarly, we develop a branch-and-cut algorithm for solving the MILP (25). Recall that defining (25) requires specifying a lower bound ℓ on v^* . We can, for example, use the quantile bound v^Q , or z_2^{NLP} obtained by iteratively solving (24). A better lower bound ℓ will yield a better LP relaxation bound. In our implementation, we use the quantile bound as the initial ℓ , and solve (24) iteratively using Benders decomposition to get the quantile-based Lagrangian dual bound v_2^{LD} as the lower bound ℓ in (33). When the quantile-based Lagrangian dual bound v_2^{LD} is obtained, we also obtain a set of Benders cuts. Among these, we include in the master problem (32) the cuts that are binding with respect to the relaxation solution in the last iteration. We also update this bound by the new optimal objective value of the master problem (32) at each iteration. Given a solution $(\bar{x}, \bar{y}, \bar{z}) \in \mathbb{R}^m \times \mathbb{R} \times [0, 1]^N$, checking its feasibility to the LP relaxation of (25) is again decomposable for each scenario. Treating x, y, z as the first-stage variables, and $\{(u^i, w^i)\}_{i \in \mathcal{N}}$ as the second-stage auxiliary variables, the LP relaxation of the master program can be written as:

$$\min_{x, y, z} y \quad (32a)$$

$$\text{s.t. } \sum_{i=1}^N p_i z_i \geq 1 - \epsilon, \quad (32b)$$

$$\lambda^k y \geq \beta^k + \rho^k z + \alpha^k x, \quad k = 1, \dots, K \quad (32c)$$

$$Dx \leq d \quad (32d)$$

$$z \in [0, 1]^N. \quad (32e)$$

Let $(\bar{x}, \bar{y}, \bar{z})$ be an optimal solution of (32), we check its feasibility to the LP relaxation of (25) by solving the following feasibility problem for each scenario $i \in \mathcal{N}$, where the dual variables associated with each set of constraints are denoted in parenthesis.

$$f^2(i) = \min_{u, w, \rho} \rho \quad (33a)$$

$$\text{s.t. } c^\top w^i - \rho \leq \bar{y} - \ell \bar{z}_i, \quad (\lambda_1) \quad (33b)$$

$$c^\top u^i - \rho \leq \bar{y} - \ell(1 - \bar{z}_i), \quad (\lambda_2) \quad (33c)$$

$$A^i u^i - \rho \mathbf{e} \leq b^i \bar{z}_i, \quad (\gamma_1) \quad (33d)$$

$$Du^i - \rho \mathbf{e} \leq d \bar{z}_i, \quad (\pi_1) \quad (33e)$$

$$Dw^i - \rho \mathbf{e} \leq d(1 - \bar{z}_i), \quad (\pi_2) \quad (33f)$$

$$u^i + w^i = \bar{x}, \quad (\alpha) \quad (33g)$$

$$\rho \geq 0. \quad (33h)$$

If $f^2(i) = 0$, then no cut is generated for scenario i . Otherwise, let $(\bar{\lambda}_1, \bar{\lambda}_2, \bar{\gamma}_1, \bar{\pi}_1, \bar{\pi}_2, \bar{\alpha})$ be an optimal dual solution. Then, the following Benders feasibility cut will cut off $(\bar{x}, \bar{y}, \bar{z})$:

$$(\bar{\lambda}_1 + \bar{\lambda}_2)y \leq -\bar{\alpha}^\top x + [\ell \bar{\lambda}_1^\top - \bar{\gamma}_1^\top b^i - \bar{\pi}_1^\top d]z_i + [\ell \bar{\lambda}_2^\top - \bar{\pi}_2^\top d](1 - z_i). \quad (34)$$

The lower bound ℓ can be updated in the tree based on the best known global lower bound obtained from the branch-and-cut tree. Once ℓ is updated, one could also update the previously generated cuts of the form (34) with this new bound, provided that the data required to calculate the coefficients in (34), $\bar{\lambda}_1, \bar{\lambda}_2, \bar{\gamma}_1^\top b^i + \bar{\pi}_1^\top d$, and $\bar{\pi}_2^\top d$, is saved. However, it is challenging to update the coefficients of Benders cuts that have been added to the master problem within CPLEX, and we choose not to do so in our implementation.

When implementing the branch-and-cut algorithm for solving (21) and (25), it is not necessary to attempt to generate a cut for every scenario in every round of cut generation. To the contrary, it is possible that a small subset of the scenarios may be most important for generating cuts, and so it makes sense to prioritize generating cuts for scenarios that have yielded cuts in previous rounds.

At the root node, we check feasibility for all scenarios and we add Benders cuts whose cut violations are larger than 10^{-5} . At non-root nodes, we maintain this cut generation effort for integer relaxation solutions

to guarantee that integer infeasible solutions are excluded. For fractional relaxation solutions, we limit the number of rounds for cut generation to be one at each node, and we apply the following strategy for generating Benders cuts according to the depth of the current branch-and-bound node: For nodes whose depths are no more than four, we check feasibility for all scenarios, but we only add up to n most violated Benders cuts, where n is the number of x variables in the model; for nodes whose depths are more than four, we only check feasibility for scenarios that have $\hat{z}_i > 1 - 10^{-3}$, and we only add the most violated Benders cut; for nodes whose depths are more than 10, we only try to generate a Benders cut for a fractional relaxation solution if the depth of the node is divisible by 15.

We update the lower bound ℓ by the best relaxation bound obtained so far, and generate Benders cut using this ℓ by solving (33). However, we do not update the coefficients of Benders cuts that have been added to the master problem using this new bound ℓ .

This decomposition scheme can be incorporated within a branch-and-cut algorithm in which some of the x variables are integer constrained. In this case, branching would be done on both integer constrained x variables and the z variables. Cuts of the form (31c) or (32c) must be separated if possible any time a solution that satisfies the integrality constraints for both x and z is found, via a lazy cut callback. These cuts can also optionally be added throughout the branch-and-cut tree as user cuts, using some rule for balancing the effort generating the cuts with the bound reduction benefit provided by the cuts.