Mixed Integer Linear Programming Formulations for Probabilistic Constraints

J. P. Vielma[a,*], S. Ahmed[b], G. Nemhauser[b]

[a]Department of Industrial Engineering, University of Pittsburgh 1048 Benedum Hall, Pittsburgh, PA 15261
[b]H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, 765 Ferst Drive NW, Atlanta, GA 30332-0205

Abstract

We introduce two new formulations for probabilistic constraints based on extended disjunctive formulations. Their strength results from considering multiple rows of the probabilistic constraints together. The properties of the first can be used to construct hierarchies of relaxations for probabilistic constraints, while the second provides computational advantages over other formulations.

Keywords: Mixed Integer Linear Programming, Formulations, Probabilistic Constraints, Chance Constraints

[*]Corresponding author
Email addresses: jvielma@pitt.edu (J. P. Vielma), sahmed@isye.gatech.edu (S. Ahmed), gnemhaus@isye.gatech.edu (G. Nemhauser)

1. Introduction

We consider Mixed Integer Linear Programming (MILP) formulations of joint probabilistic or chance constraints for finitely distributed random variables. For arbitrary distributions such constraints have been extensively studied and have many applications (see for example [18, 22] and the references within). The discrete distribution case has been studied in [3, 7, 11, 13, 15, 21] and used in applications in [3, 4, 12, 16, 20]. Finite distributions also appear naturally in Sample Average Approximations (SAA) of general probabilistic constraints [14].

We concentrate on the probabilistically constrained set $Q := \{x \in \mathbb{R}^d \,:\, \mathbb{P}\big(x \geq \xi\big) \geq 1-\delta\}$ where $\delta \in (0,1)$ and ξ is a d-dimensional random vector with finite support $\{\xi^1,\ldots,\xi^S\} \subset \mathbb{R}^d_+$ and with $\mathbb{P}(\xi = \xi^s) = 1/S$ for each $s \in \{1,\ldots,S\}$. A standard MILP formulation for Q was introduced in [19] and is given by

$$\sum_{s=1}^{S} z_s \leq k, \tag{1a}$$

$$z_s \in \{0,1\} \qquad \forall s \in \{1,\ldots,S\} \tag{1b}$$

$$x \geq (1-z_s)\xi^s \quad \forall s \in \{1,\ldots,S\} \tag{1c}$$

where $k := \lfloor \delta S \rfloor$. This formulation uses binary variables $z \in \{0,1\}^S$ such that $z_s = 1$ if $x \ngeq \xi^s$ and restricts the number of violated $x \geq \xi^s$ inequalities through the cardinality constraint (1a). The Linear Programming (LP) relaxation of formulation (1) can be very weak, so valid inequalities for it have been developed in [11, 15]. In addition, a strengthened version of (1) was introduced in [15]. Because of the use of big-M type constraints based on lower bounds on x we denote the original and strengthened versions of (1) as *Big-M* and *Strong Big-M* formulations respectively.

Alternative MILP formulations for Q can be constructed using standard disjunctive programming arguments. Unfortunately, the sizes of the resulting formulations are exponential in S for fixed δ. Although this size can be significantly reduced by using so-called $(1-\delta)$-efficient points [3, 7, 18, 21], the resulting sizes usually remain exponential (e.g. see section 5.3 of [24] for an example in which the reduced formulation size is exponential in d). Hence, it is not practical to use these disjunctive formulations directly and they are mostly used as a base for specialized algorithms [3, 6, 7, 8] or to construct valid inequalities [19, 21, 23].

Our main contribution is to introduce two new MILP formulations for Q. The first formulation can be used to construct a hierarchy of relaxations for Q and the second one provides a computational advantage over other formulations. In addition, both formulations can consider more than one row of (1c) at a time. To the best of our knowledge, no other existing formulation can do this without assuming a special structure for $\{\xi^1,\ldots,\xi^S\}$.

The rest of this paper is organized as follows. In Section 2 we introduce the new formulations and their theoretical properties and in Section 3 we present results of computational experiments that illustrate the strength and effectiveness of existing and new formulations.

2. New Formulations

Our new formulations rely on the following disjunctive characterization of the feasible region of (1) when considering only a subset D of the rows of (1c).

For $D \subset \{1,\ldots,d\}$ let $x_D = (x_i)_{i\in D}$, $\xi^s_D = (\xi^s_i)_{i \in D}$ and define $Q^D := \Big\{(x,z) \in \mathbb{R}^D \times \{0,1\}^S \,:\, \sum_{s=1}^S z_s \leq k,\, x_D \geq (1-z_s)\xi^s_D \quad \forall s \in \{1,\ldots,S\}\Big\}$. Additionally for $D \subset \{1,\ldots,d\}$ and $g \in \mathbb{R}^D$ let $v_D(g) = \{s \in \{1,\ldots,S\} \,:\, g \ngeq \xi^s_D\}$ be the set of scenarios for which g violates constraint $g \geq \xi^s_D$ and define $Q^D_g := \Big\{(x,z) \in \mathbb{R}^D \times \{0,1\}^S \,:\, x_D \geq g, \quad z_s = 1 \quad \forall s \in v_D(g), \quad \sum_{s \notin v_D(g)} z_s \leq (k - |v_D(g)|)\Big\}$. We then have the following two lemmas whose proofs are straightforward.

Proposition 1. *Let $\mathcal{D} \subset 2^{\{1,\ldots,d\}}$ be such that $\bigcup_{D\in\mathcal{D}} D = \{1,\ldots,d\}$, then (x,z) satisfies (1) if and only if $(x_D, z) \in Q^D$ for all $D \in \mathcal{D}$.*

Proposition 2. *Let* $G_D := \left\{ g \in \prod_{j\in D} \{\xi_j^s\}_{s=1}^S \,:\, |v_D(g)| \le k \right\}$ *where* $\prod$ *denotes the cartesian product. Then* $Q^D = \bigcup_{g\in G_D} Q_g^D$.

Using these two propositions we can use standard MILP formulations for disjunctive constraints to obtain reformulations of (1). However, before we give them, we refine the characterization of Q^D in Proposition 2 by reducing the number of points in G_D as follows.

Proposition 3. *Let* $\tilde{G}_D := \bigcup_{l=0}^k \left\{ g \in \prod_{j\in D} \{\xi_j^s\}_{s=1}^S \,:\, |v_D(g)| \le l \text{ and } |v_D(g-q)| > l \quad \forall q \in \mathbb{R}_+^D \setminus \{\mathbf{0}\}. \right\}$ *then*

$$Q^D = \bigcup_{g\in\tilde{G}_D} Q_g^D. \tag{2}$$

Proof. For the first inclusion let $(x^0, z^0) \in Q^D$ and for each $j \in D$ let $s_j := \arg\max_{s=1}^S \{\xi_j^s \,:\, x_j^0 \ge \xi_j^s\}$. Let $g \in \mathbb{R}^D$ be such that $g_j := \xi_j^{s_j}$ for each j. Then $g \in \tilde{G}_D$ and $(x^0, z^0) \in Q_g^D$. The reverse inclusion is direct. □

When D is a singleton $G_D = \tilde{G}_D$, but when $|D| > 1$ $\tilde{G}_D$ can be significantly smaller that G_D. However, $\tilde{G}_D$ is usually a strict superset of the $(1-\delta)$-efficient points associated with Q^D. These facts are illustrated in the following example.

Example 1. *Let* $d = 2$, $S = 4$, $\xi^1 = (0, 20)$, $\xi^2 = (10, 10)$, $\xi^3 = (20, 0)$, $\xi^4 = (30, 30)$ *and* $k = 2$. *For this data the projection onto the x space of* (1) *is given by the shaded region in Figure 1. This depicts the choice of* $D = \{1, 2\}$ *for which* $\xi^i = \xi^i_{\{1,2\}}$ *for* $i \in \{1, \ldots, 4\}$. *The figure also shows the projections onto the* x_1 *and* x_2 *variables which corresponds to the choices of* $D = \{1\}$ *and* $D = \{2\}$ *respectively. In these two last cases we can see that* $G_D = \tilde{G}_D$ *(points surrounded by triangles for* $D = \{1\}$ *and by squares for* $D = \{2\}$*). For instance, for* $D = \{2\}$ *we have that* $\left|v_{\{2\}}\left(\xi^2_{\{2\}}\right)\right| = 2$ *and* $\left|v_{\{2\}}\left(\xi^2_{\{2\}} + \lambda w\right)\right| \ge 3$ *for any* $\lambda > 0$. *In contrast, for* $D = \{1, 2\}$ *we have* $\tilde{G}_D = \{(10, 20), (20, 10), (20, 20), (30, 30)\}$ *(points surrounded by circles) while* $G_D = \{(10, 20), (20, 10), (20, 20), (30, 30), (10, 30), (20, 30), (30, 10), (30, 20)\}$ *(points surrounded by circles and diamonds). In particular,* $(20, 20)$ *is in* $\tilde{G}_D$ *because* $\left|v_{\{1,2\}}((20, 20))\right| = 1$ *and* $\left|v_{\{1,2\}}((20, 20) - q)\right| \ge 2$ *for any* $q \in \mathbb{R}_+^2$ *(such as for* $q = -u$ *and* $q = -v$*), while* $(30, 10)$ *is not in* $\tilde{G}_D$ *because* $\left|v_{\{1,2\}}((30, 10))\right| = 2$, *but* $\left|v_{\{1,2\}}((30, 10) + \lambda h)\right| = 2$ *for any sufficiently small* $\lambda > 0$. *Finally, note that* $\tilde{G}_D$ *strictly contains the set of* $(1-\delta)$*-efficient points which is given by* $\{(10, 20), (20, 10), (20, 20)\}$ *(points surrounded by hexagons).*

Using Propositions 1 and 3 we can construct the following two families of formulations for Q.

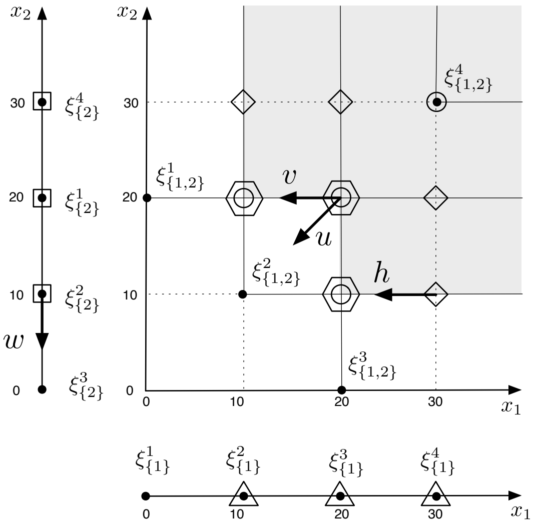

Figure 1: Example 1

Proposition 4. *Let $\mathcal{D} \subset 2^{\{1,\ldots,d\}}$ be such that $\bigcup_{D\in\mathcal{D}} D = \{1,\ldots,d\}$. Then*

$$x_j \geq \sum_{g\in\tilde{G}_D} y_g^D g_j \qquad \forall j \in D,\, D \in \mathcal{D} \tag{3a}$$

$$\sum_{g\in\tilde{G}_D} y_g^D = 1 \qquad \forall D \in \mathcal{D} \tag{3b}$$

$$y_g^D \in \{0,1\} \qquad \forall g \in \tilde{G}_D,\, D \in \mathcal{D} \tag{3c}$$

$$0 \leq z_s^{D,g} \leq y_g^D \qquad \forall g \in \tilde{G}_D,\, s \in \{1,\ldots,S\},\, D \in \mathcal{D} \tag{3d}$$

$$z_s^{D,g} \geq y_g^D \qquad \forall g \in \tilde{G}_D,\, s \in v_D(g),\, D \in \mathcal{D} \tag{3e}$$

$$\sum_{s\notin v_D(g)} z_s^{D,g} \leq y_g^D(k - |v_D(g)|) \qquad \forall g \in \tilde{G}_D,\, D \in \mathcal{D} \tag{3f}$$

$$z_s = \sum_{g\in\tilde{G}_D} z_s^{D,g} \qquad \forall s \in \{1,\ldots,S\},\, D \in \mathcal{D} \tag{3g}$$

$$z_s \in \{0,1\} \qquad \forall s \in \{1,\ldots,S\} \tag{3h}$$

is a valid formulation of Q. A smaller valid formulation is given by

$$x_j \geq \sum_{g\in\tilde{G}_D} y_g^D g_j \qquad \forall j \in D,\, D \in \mathcal{D} \tag{4a}$$

$$\sum_{g\in\tilde{G}_D} y_g^D = 1 \qquad \forall D \in \mathcal{D} \tag{4b}$$

$$y_g^D \in \{0,1\} \qquad \forall g \in \tilde{G}_D,\, D \in \mathcal{D} \tag{4c}$$

$$\sum_{s=1}^{S} z_s \leq k \tag{4d}$$

$$z_s \in \{0,1\} \qquad \forall s \in \{1,\ldots,S\} \tag{4e}$$

$$z_s \geq \sum_{g:s\in v_D(g)} y_g^D \qquad \forall s \in \{1,\ldots,S\},\, D \in \mathcal{D}. \tag{4f}$$

Proof. For $\mathcal{D} = \{D\}$ we have that (3) with (3a) replaced by

$$x_j^{D,g} \geq y_g^D g_j, \quad x_j = \sum_{g \in \tilde{G}_D} x_j^{D,g} \quad \forall g \in \tilde{G}_D,\, j \in D,\, D \in \mathcal{D} \tag{5}$$

is the standard MILP formulation of disjunctive characterization (2) (e.g. Corollary 2.1.2 of [2]). Then, for $\mathcal{D} = \{D\}$, we have that (3) is a valid formulation for Q^D as it is obtained from the standard disjunctive formulation by eliminating variables $x_j^{D,g}$ in a way that naturally preserves formulation validity (e.g. [1, 5, 10]). The result then follows from Proposition 1.

Formulation (4) is valid for Q because it is obtained by similarly eliminating variables $z_s^{D,g}$ and $x_j^{D,g}$ from the standard disjunctive formulation of (2) when using the equivalent definition of Q_g^D given by $Q_g^D := \Big\{(x,z) \in \mathbb{R}^D \times \{0,1\}^S : x_D \geq g, \quad z_s = 1 \quad \forall s \in v_D(g), \quad \sum_{s=1}^S z_s \leq k\Big\}$. □

The following two examples illustrate both formulations.

Example 2. *For the data in Example 1 with $\mathcal{D} = \{D\}$ for $D = \{1,2\}$ formulation* (3) *is*

$$10y_{(10,20)}^D + 20y_{(20,10)}^D + 20y_{(20,20)}^D + 30y_{(30,30)}^D \leq x_1, \qquad 20y_{(10,20)}^D + 10y_{(20,10)}^D + 20y_{(20,20)}^D + 30y_{(30,30)}^D \leq x_2$$
$$y_{(10,20)}^D + y_{(20,10)}^D + y_{(20,20)}^D + y_{(30,30)}^D = 1, \qquad y_g^D \in \{0,1\} \quad \forall g \in \{(10,20),(20,10),(20,20),(30,30)\}$$
$$0 \leq z_s^{D,g} \leq y_g^D \qquad \forall s \in \{1,\ldots,4\},\, g \in \{(10,20),(20,10),(20,20),(30,30)\}$$
$$z_3^{D,(10,20)} \geq y_{(10,20)}^D, \quad z_4^{D,(10,20)} \geq y_{(10,20)}^D, \quad z_1^{D,(20,10)} \geq y_{(20,10)}^D, \quad z_4^{D,(20,10)} \geq y_{(20,10)}^D, \quad z_4^{D,(20,20)} \geq y_{(20,20)}^D$$
$$z_1^{D,(10,20)} + z_2^{D,(10,20)} \leq 0, \quad z_2^{D,(20,10)} + z_3^{D,(20,10)} \leq 0, \qquad \sum_{s=1}^3 z_s^{D,(20,20)} \leq y_{(20,20)}^D, \quad \sum_{s=1}^4 z_s^{D,(30,30)} \leq 2y_{(30,30)}^D$$
$$z_s^{D,(10,20)} + z_s^{D,(20,10)} + z_s^{D,(20,20)} + z_s^{D,(30,30)} = z_s, \qquad z_s \in \{0,1\} \quad \forall s \in \{1,\ldots,4\}.$$

and formulation (4) *is*

$$10y_{(10,20)}^D + 20y_{(20,10)}^D + 20y_{(20,20)}^D + 30y_{(30,30)}^D \leq x_1, \qquad 20y_{(10,20)}^D + 10y_{(20,10)}^D + 20y_{(20,20)}^D + 30y_{(30,30)}^D \leq x_2$$
$$y_{(10,20)}^D + y_{(20,10)}^D + y_{(20,20)}^D + y_{(30,30)}^D = 1, \qquad y_g^D \in \{0,1\} \quad \forall g \in \{(10,20),(20,10),(20,20),(30,30)\}$$
$$\sum_{s=1}^4 z_s \leq 2, \quad z_s \in \{0,1\}, \quad \forall s \in \{1,\ldots,4\}, \quad z_4 \geq y_{(10,20)}^D + y_{(20,10)}^D + y_{(20,20)}^D, \quad z_1 \geq y_{(20,10)}^D, \quad z_3 \geq y_{(10,20)}^D.$$

Example 3. *For the data in Example 1 with $\mathcal{D} = \{\{1\},\{2\}\}$ formulation* (3) *is*

$$10y_{10}^{\{j\}} + 20y_{20}^{\{j\}} + 30y_{30}^{\{j\}} \leq x_j \quad \forall j \in \{1,2\}$$
$$y_{10}^{\{j\}} + y_{20}^{\{j\}} + y_{30}^{\{j\}} = 1 \quad \forall j \in \{1,2\}, \qquad y_g^{\{j\}} \in \{0,1\} \quad \forall g \in \{10,20,30\},\, j \in \{1,2\}$$
$$0 \leq z_s^{\{j\},g} \leq y_g^{\{j\}} \qquad \forall g \in \{10,20,30\},\, s \in \{1,\ldots,4\},\, j \in \{1,2\}$$
$$z_3^{\{1\},10} \geq y_{10}^{\{1\}}, \quad z_4^{\{1\},10} \geq y_{10}^{\{1\}}, \quad z_4^{\{1\},20} \geq y_{20}^{\{1\}} \qquad z_1^{\{2\},10} \geq y_{10}^{\{2\}}, \quad z_4^{\{2\},10} \geq y_{10}^{\{2\}}, \quad z_4^{\{2\},20} \geq y_{20}^{\{2\}}$$
$$z_1^{\{1\},10} + z_2^{\{1\},10} \leq 0, \quad z_2^{\{2\},10} + z_3^{\{2\},10} \leq 0, \qquad \sum_{s=1}^3 z_s^{\{j\},20} \leq y_{20}^{\{j\}}, \sum_{s=1}^4 z_s^{\{j\},30} \leq 2y_{30}^{\{j\}} \quad \forall j \in \{1,2\}$$
$$z_s^{\{j\},10} + z_s^{\{j\},20} + z_s^{\{j\},30} = z_s \, \forall s \in \{1,\ldots,4\},\, j \in \{1,2\}, \qquad z_s \in \{0,1\} \quad \forall s \in \{1,\ldots,4\}.$$

and formulation (4) *is*

$$10y_{10}^{\{j\}} + 20y_{20}^{\{j\}} + 30y_{30}^{\{j\}} \leq x_j \quad \forall\, j \in \{1,2\}$$
$$y_{10}^{\{j\}} + y_{20}^{\{j\}} + y_{30}^{\{j\}} = 1 \quad \forall j \in \{1,2\}, \qquad y_g^{\{j\}} \in \{0,1\} \quad \forall g \in \{10,20,30\},\, j \in \{1,2\}$$
$$\sum_{s=1}^4 z_s \leq 2, \quad z_s \in \{0,1\} \quad \forall s \in \{1,\ldots,4\}, \qquad z_3 \geq y_{10}^{\{1\}}, \quad z_4 \geq y_{10}^{\{1\}} + y_{20}^{\{1\}} \quad z_1 \geq y_{10}^{\{2\}}, \quad z_4 \geq y_{10}^{\{2\}} + y_{20}^{\{2\}}.$$

As shown in Examples 2 and 3, different choices of $\mathcal{D}$ yield different formulations. For instance, if we take $\mathcal{D} = \{D_l\}_{l=1}^d$ for $D_l = \{l\}$ formulation (3) is equal to a known formulation that was first introduced in [11] and which can also be obtained by combining Corollary 1 of [15] and Proposition 1 of [17]. Because it is based on sets Q^D that only consider one row of (1c) at a time we refer to this formulation as the 1-row Extended Disjunctive Formulation and denote it by *Ext-1-row*. Similarly, if $\xi^1 \geq \xi^2 \geq \ldots \geq \xi^S$ and $\mathcal{D} = \{D\}$ for $D = \{1, \ldots, d\}$ then (3) reduces to the formulation in Theorem 9 of [11]. To the best of our knowledge these are the only two special cases of formulations (3) and (4) that have previously appeared in the literature. In particular, (4) with $\mathcal{D} = \{D_l\}_{l=1}^d$ for $D_l = \{l\}$ is a new formulation which we denote by *Proj-1-row* as it is obtained from *Ext-1-row* by a variable elimination procedure that is akin to projection. Similarly, for even d, we denote formulations (3) and (4) with $\mathcal{D} = \{D_l\}_{l=1}^{d/2}$ for $D_l = \{2l-1, 2l\}$ as *Ext-2-row* and *Proj-2-row* respectively. These two formulations are also new to this paper. In fact, to the best of our knowledge, these are the first two formulations that consider more than one row of (1c) at a time without assuming some special structure such as in Theorem 9 of [11].

We can of course use families $\mathcal{D}$ composed of sets of size greater than two. However, the sizes of (3) and (4) are $\Theta\left(\prod_{D\in\mathcal{D}} S * |\tilde{G}_D|\right)$ and $\Theta\left(\prod_{D\in\mathcal{D}}(|\tilde{G}_D| + S)\right)$ respectively and $|\tilde{G}_D|$ can be exponential in D (e.g. see section 5.3 of [24]). Formulations based on (3) and (4) are hence tractable only for specially structured $\{\xi^s\}_{s=1}^S$ (e.g. for the special case in Theorem 9 of [11] we get $|\tilde{G}_D| = k+1$) or for very small values of $\max_{D\in\mathcal{D}} |D|$. Even in those cases, the formulations can be tractable only in a theoretical sense. For instance, Ext-1-row is theoretically compact in the sense that it has a polynomial number of variables and constraints: $S + d(2 + k + S + kS)$ variables besides x and $d/2(6 + 4S + k(3 + k + 2S))$ constraints excluding variable bounds. However, as noted in [11], it is still too large to be of direct practical use. Fortunately, Proj-1-row has a much smaller size: $S + (k+1)d$ variables besides x and $(k+2)d + 1$ constraints. As we will see in Section 3, this smaller size allows Proj-1-row to be of direct computational use.

2.1. Formulation Strength

The variable eliminations used to construct (3) and (4) clearly preserve formulation validity, but do not always preserve formulation strength [1, 5, 10]. However, strength preservation is shown to hold for Ext-1-row in [11]. A direct extension of the proof in [11] shows that this holds for all versions of (3).

Proposition 5. *The projection of the LP relaxation of* (3) *onto the* (x, z) *variables is equal to* $\bigcap_{D\in\mathcal{D}} \operatorname{conv}(Q^D)$.

Proof. It suffices to show the result for $\mathcal{D} = \{D\}$, that is to show that (3) for a single D is equal to $\operatorname{conv}(Q^D)$. In this case we have that (3) with (3a) replaced by (5) is the standard disjunctive formulation of $\bigcup_{g\in\tilde{G}_D} Q_g^D$ and hence its LP relaxation is equal to $\operatorname{conv}\left(\bigcup_{g\in\tilde{G}_D} Q_g^D\right)$ (e.g. Corollary 2.1.2 of [2]). Because the extreme points of the LP relaxation of Q_g^D have integer z variables we additionally have that the projection onto the (x, z) variables of the LP relaxation of this disjunctive formulation is also equal to $\operatorname{conv}\left(Q^D\right)$. The result then follows by noting that any solution to the LP relaxation of (3) can be easily extended to a solution to the standard disjunctive formulation (e.g. pick $g^0 \in \tilde{G}_D$, let $x_j^g = y_g g_j$ for all $j \in D$, $g \in \tilde{G}_D \setminus \{g^0\}$ and $x_j^{g^0} = y_{g^0} g_j^0 + x_j - \sum_{g\in\tilde{G}_D} y_g g_j$). □

As a direct corollary we obtain the following hierarchies of relaxations of $\operatorname{conv}(Q)$.

Corollary 6. *Let* $\{\mathcal{D}_p\}_{p=1}^P$ *be a family of partitions of* $\{1, \ldots, d\}$ *such that* $\mathcal{D}_1 = \{D_j\}_{j=1}^d$ *for* $D_j = \{j\}$, $\mathcal{D}_P = \{D\}$ *for* $D = \{1, \ldots, d\}$ *and* $\mathcal{D}_p$ *is a refinement of* $\mathcal{D}_{p+1}$ *(That is* $\mathcal{D}_p$ *is obtained from* $\mathcal{D}_{p+1}$ *by partitioning some of its elements). For each* $p \in \{1, \ldots, P\}$ *let* H^p *be the projection onto the* (x, z) *variables of the LP relaxation of formulation* (3) *for* $\mathcal{D} = \mathcal{D}_p$. *Then* $H^p = \bigcap_{D\in\mathcal{D}_p} \operatorname{conv}\left(Q^D\right)$ *and hence* $H^1 \supset H^2 \supset \ldots \supset H^P = \operatorname{conv}(Q)$.

In general, the strength of the LP relaxation of (3) will be equivalent to $\operatorname{conv}(Q)$ only when $\mathcal{D}_P = \{D\}$ for $D = \{1, \ldots, d\}$. However, in Section 3.2 we will see that even Ext-2-row and Ext-1-row can be very strong formulations. Furthermore, Theorem 8 of [11] shows that Ext-1-row is always at least as strong as

Strong Big-M and, although the LP relaxation of Ext-1-row can yield bounds arbitrarily close to the ones provided by Strong Big-M (see Section 5.4.1 of [24]), it is usually a significantly stronger formulation (see Section 5.4.2 and 5.5 of [24]).

With regards to formulation (4) we have that the additional variable eliminations necessary to construct it do result in a loss of strength. However, in Section 3.2 we will see that the loss of strength can be quite small. Furthermore, we have that the strength of Proj-1-row is equivalent to that of another known strong formulation for Q. This formulation was introduced in [15] and is given by

$$\sum_{s=1}^{S} z_s \leq k \tag{6a}$$

$$z_s \in \{0,1\} \qquad \forall s \in \{1,\dots,k\} \tag{6b}$$

$$x_j \geq \xi_j^{[1]_j} + \sum_{s=1}^{k} \left(\xi_j^{[s+1]_j} - \xi_j^{[s]_j} \right) w_s^j \quad \forall j \in \{1,\dots,d\} \tag{6c}$$

$$w_s^j \geq w_{s+1}^j \qquad \forall s \in \{1,\dots,k-1\},\ j \in \{1,\dots,d\} \tag{6d}$$

$$z_{[s]_j} \geq w_s^j \qquad \forall s \in \{1,\dots,k\},\ j \in \{1,\dots,d\} \tag{6e}$$

$$w_s^j \in \{0,1\} \qquad \forall s \in \{1,\dots,k\},\ j \in \{1,\dots,d\} \tag{6f}$$

where $[\cdot]_j : \{1,\dots,S\} \to \{1,\dots,S\}$ is the one to one function such that $\xi_j^{[1]_j} \geq \xi_j^{[2]_j} \geq \ldots \geq \xi_j^{[S]_j}$. Because constraint (6c) forces $x_j \geq \xi_j^s$ in an incremental fashion we refer to this formulation as the Incremental Model and denote it by *Inc*. Besides x this formulation has $S + kd$ variables that are all binary and $2kd + 1$ constraints excluding variable bounds.

Proposition 7. *The projection onto the (x,z) variables of the LP relaxations of Proj-1-row and Inc are identical.*

Proof. For any (x,z,w) feasible to the LP relaxation of Inc we have that (x,z,y) is a solution to the LP relaxations of Proj-1-row for $y_{i,s} = w_{s-1}^j - w_s^j$ for $s \in \{2,\dots,k+1\}$ and $y_{i,1} = 1 - \sum_{t=2}^{k+1} y_{i,t}$. Conversely, for any (x,z,y) feasible to the LP relaxation of Proj-1-row we have that (x,z,w) is a solution to the LP relaxations of Inc for $w_s^j = \sum_{t=s+1}^{k+1} y_{j,t}$. □

3. Computational Results

In this section we present some computational results illustrating properties of the two new formulations introduced in the previous section. For this we use the probabilistically constrained fixed charge transportation problem given by

$$\min \quad \sum_{i=1}^{n} \sum_{j=1}^{d} (c_{i,j} w_{i,j} + f_{i,j} y_{i,j}) \tag{7a}$$

s.t.

$$\sum_{j=1}^{d} w_{i,j} \leq b_i \qquad \forall i \in \{1,\dots,n\} \tag{7b}$$

$$\sum_{i=1}^{n} w_{i,j} = x_j \qquad \forall j \in \{1,\dots,d\} \tag{7c}$$

$$0 \leq w_{i,j} \leq b_i y_{i,j} \quad \forall i \in \{1,\dots,n\},\ j \in \{1,\dots,d\} \tag{7d}$$

$$y_{i,j} \in \{0,1\} \quad \forall i \in \{1,\dots,n\},\ j \in \{1,\dots,d\} \tag{7e}$$

$$\mathbb{P}\big(x \geq \xi\big) \geq 1 - \delta \tag{7f}$$

where ξ is a d-dimensional random vector of uncertain demands with finite support on $\{\xi^1, \ldots, \xi^S\} \subset \mathbb{R}^d_+$ and with $\mathbb{P}(\xi = \xi^s) = 1/S$ for each $s \in \{1, \ldots, S\}$. We use instances with $n = 40$ supply nodes, $d = 10$ demand nodes and varying number of samples of the uncertain demands S. Each instance is generated as follows. $c_{i,j}$'s are independently chosen from the uniform distribution on $[1, 100]$ and ξ^s's are independently chosen from the uniform distribution on $[1, 50]^{10}$. To generate b we first independently chose d_i from the uniform distribution on $[1, 100]$ and then rescale it so the problem is feasible for each ξ^s. Finally, $f_{i,j}$'s are all identical to 0 or 100 depending on the type of instance. All models are generated using IBM ILOG Concert Technology and solved with IBM CPLEX v12 [9] on an 2.93GHz Xeon workstation. Our aim is to compare the models out of the box without the need to add specialized cutting planes to CPLEX. Because of this we are only able to test small and medium sized instances. Larger instances require the use of specialized techniques such as the ones introduced in [11, 13, 15].

3.1. Solve Times

We begin by comparing the solve times for the MILP problems obtained when using different formulations of (7f). In preliminary tests we discovered that the only formulations that could solve instances of non-trivial sizes were Inc and Proj-1-row. These formulations have similar sizes and Proposition 7 shows they have the same strength. In this section we illustrate how both formulations can provide a computational advantage under different circumstances.

In our first experiment we consider a transportation problem without fixed charges (i.e. $f_{i,j} = 0$ for all i, j). We can then remove the y variables and the only binary variables of the resulting MILP are the ones used to formulate (7f). We choose $\delta \in \{0.10, 0.20\}$ and $S \in \{2000, 3000\}$. Table 1(a) shows statistics for the solve times over ten instances for each choice of δ and S. Solve times include the time required to generate the formulations. In our second experiment we consider a transportation problem with fixed charges, so that we have a MILP problem even with a deterministic version of (7f). These problems are much harder so we now choose $S \in \{200, 300\}$. Table 1(b) shows statistics for the solve times over ten instances for each choice of δ and S.

		2000		3000	
		0.10	0.20	0.10	0.20
Inc	min	10	33	2	77
	avg	**2**	**57**	**4**	**208**
	max	5	94	7	429
	std	1	22	2	102
Proj-1-row	min	5	1800	10	1800
	avg	**34**	**1800**	**88**	**1800**
	max	104	1800	282	1800
	std	37	0	81	0

(a) Transportation Problems without Fixed Charges

		200		300	
		0.10	0.20	0.10	0.20
Inc	min	21	38	46	162
	avg	**479**	**578**	**591**	**1009**
	max	1800	1800	1800	1800
	std	538	656	549	626
Proj-1-row	min	9	25	12	53
	avg	**148**	**304**	**256**	**370**
	max	402	1007	1056	914
	std	141	337	310	281

(b) Transportation Problems with Fixed Charges.

Table 1: Solve Times [s].

We see that for the problems without fixed charges Inc provides a substantial advantage over Proj-1-row, while the reverse is true, although to a lesser extent, for problems with fixed charges. Tables 2 and 3 show some solve statistics that shed some light on this behavior.

For example, we observe that Proj-1-row's fewer rows seems to allow CPLEX's preprocessing and cuts to close more GAP at the root node for this formulation than for Inc. However, Inc's smaller number of non-zero coefficients seems to make the LP solve times smaller. We also observe that Inc seems to need fewer branch-and-bound nodes to solve the problem. This could be due to branching being more efficient for this model, which is a common property of incremental type formulations (see [25]). The relative effectiveness of

		2000		3000	
		0.10	0.20	0.10	0.20
Inc	min	9	197	13	221
	avg	**57**	**369**	**56**	**604**
	max	174	713	109	1249
	std	50	188	33	308
Proj-1-row	min	289	22403	327	12892
	avg	**1845**	**38538**	**3244**	**16695**
	max	7245	57176	16254	20069
	std	2468	14162	4897	2847

(a) Branch-and-bound nodes processed.

	2000		3000	
	0.10	0.20	0.10	0.20
min	0.08	2.76	0.07	2.79
avg	**0.35**	**4.26**	**0.43**	**4.41**
max	0.82	5.40	0.76	5.49
std	0.26	0.82	0.25	0.94
min	0.08	2.75	0.05	2.77
avg	**0.35**	**4.23**	**0.42**	**4.39**
max	0.82	5.38	0.75	5.47
std	0.26	0.81	0.25	0.94

(b) Root GAP. [%]

		2000		3000	
		0.10	0.20	0.10	0.20
Inc	min	160	750	340	1480
	avg	**188**	**856**	**387**	**2068**
	max	220	960	410	2540
	std	23	58	27	324
Proj-1-row	min	180	1160	480	3360
	avg	**246**	**1605**	**704**	**4321**
	max	330	1930	990	6650
	std	51	229	200	1062

(c) LP relaxation solve time. [ms]

		2000		3000	
		0.10	0.20	0.10	0.20
Inc	cols	4	6	6	9
	bin	4	6	6	9
	rows	4	8	6	12
	nzs	13	23	19	34
Proj-1-row	cols	4	6	6	9
	bin	4	6	6	9
	rows	2	4	3	6
	nzs	210	817	464	1824

(d) Formulation Size $[\times 10^3]$

Table 2: Statistics for Transportation Problems without Fixed Charges

		200		300	
		0.10	0.20	0.10	0.20
Inc	min	7	8	10	12
	avg	**77**	**57**	**80**	**67**
	max	246	175	181	112
	std	70	56	60	34
Proj-1-row	min	3	5	3	7
	avg	**38**	**50**	**48**	**47**
	max	100	175	154	108
	std	33	58	47	36

(a) Branch-and-bound nodes processed $[\times 10^3]$

	200		300	
	0.10	0.20	0.10	0.20
min	2.90	4.69	2.94	5.49
avg	**3.59**	**6.08**	**3.70**	**6.47**
max	4.43	8.26	4.53	8.02
std	0.47	1.14	0.56	0.83
min	2.65	3.63	2.38	3.80
avg	**3.63**	**5.15**	**3.60**	**5.49**
max	4.49	7.33	4.47	7.22
std	0.54	1.15	0.62	0.95

(b) Root GAP. [%]

		200		300	
		0.10	0.20	0.10	0.20
Inc	min	14.20	26.50	17.30	46.20
	avg	**18.08**	**36.97**	**25.94**	**59.48**
	max	20.50	42.00	29.70	64.40
	std	2.54	4.57	3.62	7.01
Proj-1-row	min	20.50	39.50	31.80	63.60
	avg	**25.56**	**49.22**	**38.42**	**84.31**
	max	29.00	56.30	42.40	97.50
	std	2.89	5.77	3.61	12.00

(c) LP relaxation solve time. [ms]

		200		300	
		0.10	0.20	0.10	0.20
Inc	cols	12	14	14	17
	binaries	8	10	10	13
	rows	8	12	10	16
	nzs	28	38	34	50
Proj-1-row	cols	12	14	14	17
	binaries	8	10	10	13
	rows	7	9	8	10
	nzs	45	112	75	220

(d) Formulation Size $[\times 10^2]$

Table 3: Statistics for Transportation Problems with Fixed Charges

the formulations will then depend on which of these advantages is stronger for a particular class of problems. For instance, it is possible that for the problems with fixed charges the efficient branching of Inc is diminished by the existence of additional binary variables, which results in Proj-1-row having a better performance.

3.2. LP GAPs for Transportation Problems

Our final experiment compares the LP GAPs for the different formulations calculated as 100 × $\frac{z_{IP}-z_{LP}}{z_{IP}}$ where z_{IP} is the optimal value of (7) and z_{LP} is the optimal value of its LP relaxation. We compare formulations Ext-k-row and Proj-k-row for $k \in \{1, 2\}$ and include Strong Big-M as a reference. Because the larger formulations are very hard to solve we consider a transportation problem without fixed charges and choose $\delta \in \{0.10, 0.20\}$ and $S \in \{200, 300\}$. The sets $\tilde{G}_D$ for $|D| = 2$ were generated by enumeration, which can be done relatively fast by first sorting the components of $\{\xi^s\}_{s=1}^S$. Table 4 shows statistics for the GAPs over ten instances for each choice of δ and S.

		200		300	
		0.10	0.20	0.10	0.20
Ext-1-row	min	0.00	0.32	0.01	0.30
	avg	**0.10**	**1.13**	**0.20**	**1.57**
	max	0.39	2.59	0.62	3.34
	std	0.12	0.77	0.17	1.01
Ext-2-rows	min	0.00	0.07	0.00	0.00
	avg	**0.08**	**0.59**	**0.12**	**0.71**
	max	0.32	1.52	0.30	1.72
	std	0.10	0.48	0.10	0.59
Proj-1-row	min	0.05	0.51	0.08	0.35
	avg	**0.18**	**1.29**	**0.29**	**1.64**
	max	0.47	2.59	0.70	3.37
	std	0.14	0.71	0.17	0.98
Proj-2-rows	min	0.05	0.12	0.08	0.02
	avg	**0.15**	**0.82**	**0.22**	**0.81**
	max	0.39	1.61	0.35	1.72
	std	0.10	0.49	0.10	0.56
Strong Big-M	min	4.21	11.71	5.28	12.55
	avg	**6.92**	**14.44**	**7.39**	**15.16**
	max	10.07	19.30	9.33	19.71
	std	1.84	2.52	1.55	2.40

Table 4: LP GAPs for Transportation Problems without Fixed Charges [%]

As expected the k-row formulations are significantly better than the Strong Big-M formulation, the extended formulations are better than the corresponding projected formulations and the 2-row formulations are better than the corresponding 1-row formulations. However, we see that the loss in strength from the extended to the projected formulations is quite small and can be smaller than the gain in strength from 1-row formulations to 2-row formulations. In particular, this allows Proj-2-row to be stronger on average than Ext-1-row for $\delta = 0.20$.

4. Future Work

Proposition 1 does not require that the sets in $\mathcal{D}$ be disjoint. Hence it is possible to strengthen the formulations by considering subsets of rows that have some overlap. For instance, we could consider $\mathcal{D} = \{D_l\}_{l=1}^{d-1}$ for $D_l = \{l, l+1\}$ which would yield a strengthening of the 2-row formulations. Preliminary results in smaller problems showed that the strengthened formulations can indeed provide slightly better LP bounds, but the increased size usually prevents them from performing better than the original 2-row formulations, which are already outperformed by the 1-row formulations. Still, careful selection of the overlaps (e.g. by considering correlations among the components of ξ) could provide a theoretical or computational advantage in some problems.

5. Acknowledgements

This research was supported by NSF grant CMMI-0758234 to the Georgia Institute of Technology. The authors would like to thank an anonymous referee for some very helpful comments.

References

[1] E. Balas. On the convex-hull of the union of certain polyhedra. *Oper. Res. Lett.*, 7:279–283, 1988.

[2] E. Balas. Disjunctive programming: Properties of the convex hull of feasible points. *Discrete Appl. Math.*, 89:3–44, 1998.

[3] P. Beraldi and A. Ruszczynski. A Branch and Bound Method for Stochastic Integer Problems Under Probabilistic Constraints. *Optim. Method. Softw.*, 17:359–382, 2002.

[4] P. Beraldi and A. Ruszczynski. The probabilistic set-covering problem. *Oper. Res.*, 50:956–967, 2002.

[5] C. Blair. Representation for multiple right-hand sides. *Math. Programming*, 49:1–5, 1990.

[6] D. Dentcheva, B. Lai, and A. Ruszczynski. Dual methods for probabilistic optimization problems. *Math. Met. Oper. Res.*, 60:331–346, 2004.

[7] D. Dentcheva, A. Prekopa, and A. Ruszczynski. Concavity and efficient points of discrete distributions in probabilistic programming. *Math. Programming*, 89:55–77, 2000.

[8] D Dentcheva, A Prekopa, and A Ruszczynski. On convex probabilistic programming with discrete distributions. *Nonlinear Anal. Theory Met. Appl.*, 47:1997–2009, 2001.

[9] IBM ILOG. CPLEX High-performance mathematical programming engine. `http://www.ibm.com/software/integration/optimization/cplex/`.

[10] R. G. Jeroslow. A simplification for some disjunctive formulations. *Eur. J. Oper. Res.*, 36:116–121, 1988.

[11] S. Kucukyavuz. On mixing sets arising in probabilistic programming. *Math. Programming (To appear)*, 2010. doi:10.1007/s10107-010-0385-3.

[12] M. A. Lejeune and A. Ruszczynski. An efficient trajectory method for probabilistic production-inventory-distribution problems. *Oper. Res.*, 55:378–394, 2007.

[13] J. Luedtke. An integer programming and decomposition approach to general chance-constrained mathematical programs. In F. Eisenbrand and F. B. Shepherd, editors, *IPCO*, volume 6080 of *Lecture Notes in Computer Science*, pages 271–284. Springer, 2010.

[14] J. Luedtke and S. Ahmed. A Sample Approximation Approach for Optimization with Probabilistic Constraints. *SIAM J. Optim.*, 19:674–699, 2008.

[15] J. Luedtke, S. Ahmed, and G. Nemhauser. An integer programming approach for linear programs with probabilistic constraints. *Math. Programming*, 122:247–272, 2010.

[16] G. Lulli and S. Sen. A branch-and-price algorithm for multistage stochastic integer programming with application to stochastic batch-sizing problems. *Manage. S.*, 50:786–796, 2004.

[17] R. K. Martin. Using separation algorithms to generate mixed integer model reformulations. *Oper. Res. Lett.*, 10:119–128, 1991.

[18] A. Prekopa. Probabilistic programming. In A. Shapiro and A. Ruszczynski, editors, *Stochastic Programming*, volume 10 of *Handbooks in Operations Research and Management Science*, pages 267–351. Elsevier, 2003.

[19] A. Ruszczynski. Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra. *Math. Programming*, 93:195–215, 2002.

[20] A. Saxena, V. Goyal, and M. A. Lejeune. MIP reformulations of the probabilistic set covering problem. *Math. Programming*, 121:1–31, 2010.

[21] S. Sen. Relaxations for probabilistically constrained programs with discrete random variables. *Oper. Res. Lett.*, 11:81–86, 1992.

[22] A. Shapiro, D. Dentcheva, and A. Ruszczynski. *Lectures on Stochastic Programming: Modeling and Theory.* SIAM, 2009.

[23] A. T. Tuncel, F. Preciado, R. L. Rardin, M. Langer, and J.-P. P. Richard. Strong valid inequalities for fluence map optimization problem under dose-volume restrictions. *Ann. Oper. Res. (To appear)*, 2010. doi:10.1007/s10479-010-0759-1.

[24] J. P. Vielma. *Mixed integer programming approaches for nonlinear and stochastic programming.* PhD thesis, Georgia Institute of Technology, 2009.

[25] J. P. Vielma, S. Ahmed, and G. L. Nemhauser. Mixed-integer models for nonseparable piecewise linear optimization: Unifying framework and extensions. *Oper. Res.*, 58:303–315, 2010.