A Polyhedral Study on Chance Constrained Program with Random Right-Hand Side

Ming Zhao Kai Huang Bo Zeng

April 25, 2016

Abstract

The essential structure of the mixed–integer programming formulation for chance–constrained program (CCP) with stochastic right–hand side is the intersection of multiple mixing sets with a $0-1$ knapsack. To improve our computational capacity on CCP, an underlying substructure, the (single) mixing set with a $0-1$ knapsack, has received substantial attentions recently. In this study, we first present a family of strong inequalities that subsumes known facet-defining ones for that single mixing set. Due to the flexibility of our generalized inequalities, we develop a new separation heuristic that has a complexity much less than existing one and guarantees generated cutting planes are facet–defining for the polyhedron of CCP. Then, we study lifting and superadditive lifting on knapsack cover inequalities, and provide an implementable procedure on deriving another family of strong inequalities for the single mixing set. Finally, different from the traditional approach that aggregates original constraints to investigate polyhedral implications due to their interactions, we propose a novel blending procedure that produces strong valid inequalities for CCP by integrating those derived from individual mixing sets. We show that, under certain conditions, they are the first type of facet-defining inequalities describing intersection of multiple mixing sets, and design an efficient separation heuristic for implementation. In the computational experiments, we perform a systematic study and illustrate the efficacy of the proposed inequalities on solving chance constrained static probabilistic lot-sizing problems.

Keywords: Mixed–integer programming, Chance constraints, Mixing set, Knapsack, Lifting, Blending

1 Introduction

Chance constraints appear in optimization formulations of many important applications that model service levels, risk measures, or reliability requirements. When the randomness occurs only at the right–hand side vector, the chance–constrained program with joint probabilistic constraints (CCP) can be formulated as follows

$$\begin{aligned} \min \quad & \mathbf{c}^T\mathbf{x} \\ s.t. \quad & y = \mathbf{A}\mathbf{x} \\ & \mathbb{P}\{y \geq \mathbf{h}(\omega)\} \geq 1-\tau \\ & \mathbf{x} \in \mathbf{X} \subseteq \mathbb{R}^{m_1} \times \mathbb{Z}^{m_2}, \mathbf{y} \in \mathbb{R}^d_+ \end{aligned}$$

where $\mathbf{X}$ is a polyhedron, d and m are positive integers with $m = m_1 + m_2$ for other two positive integers m_1 and m_2, ω is a random scenario in the probability space Ω, $\mathbf{x}$ is an m–dimensional decision variable, $\mathbf{A}$ is a $d \times m$ matrix, $\mathbf{h}(\omega)$ is a d–dimensional column random vector, $\mathbf{c}$ is an m–dimensional cost vector, and τ is a threshold probability with $0 \leq \tau \leq 1$. The chance–constrained program with joint probabilistic constraints was first studied by Miller and Wagner [20] for independent random variables. A disjunctive programming reformulation for CCP with discrete distributions was studied in [6, 8, 25] by using the concept of $1-\epsilon$ efficient points [19]. The chance–constrained program has many applications, such as probabilistic lot-sizing [6, 32], health care [5, 26], probabilistic set covering [7, 24]. Recently, many studies extended the model to multi-stage chance–constrained programs. They derived valid inequalities for the deterministic equivalent formulation [32] or strong feasibility and optimality cuts for decomposition algorithms, see [16, 17, 28, 31].

In this paper, we consider a mixed–integer programming (MIP) reformulation of chance–constrained program. Suppose Ω has finitely many realizations, i.e., $\Omega = \{\omega_1, \omega_2, \ldots, \omega_n\}$ and π_i is the probability associated with ω_i, $\forall i \in \{1, \ldots, n\}$. Let h_{ri} be the r-th component of $\mathbf{h}(\omega_i)$. As described in [15, 18], we can assume $h_{ri} \geq 0$ without loss of generality. Throughout, we denote $[i, j] \equiv \{r \in \mathbb{Z} : i \leq r \leq j\}$. A deterministic equivalent formulation of the chance–constrained program is (see also [1, 15, 18])

$$\begin{aligned} \min \quad & \mathbf{c}^T\mathbf{x} \\ s.t. \quad & \mathbf{y} = \mathbf{A}\mathbf{x} \\ & y_r \geq h_{ri}(1-z_i) \qquad \forall r \in [1,d], i \in [1,n] && (1) \\ & \sum_{i=1}^{n} \pi_i z_i \leq \tau && (2) \\ & \mathbf{x} \in \mathbf{X} \subseteq \mathbb{R}^{m_1} \times \mathbb{Z}^{m_2} \\ & \mathbf{z} \in \{0,1\}^n, \mathbf{y} \in \mathbb{R}^d_+ \end{aligned}$$

where $z_i = 0$ indicates that $y \geq h(\omega)$ is satisfied when $\omega = \omega_i$ and $z_i = 1$ otherwise. The constraints (1) and (2) define the key substructure of this MIP reformulation, i.e., the polyhedron of CCP

$$\mathcal{Q} = \left\{ (\mathbf{y}, \mathbf{z}) \in \mathbb{R}^d_+ \times \{0,1\}^n : \sum_{i=1}^{n} \pi_i z_i \leq \tau,\ y_r \geq h_{ri}(1-z_i),\ r \in [1,d], i \in [1,n] \right\}.$$

For $r \in [1, d]$, we have a mixing set with $0-1$ knapsack

$$\mathcal{Q}_r = \left\{ (y_r, \mathbf{z}) \in \mathbb{R}_+ \times \{0,1\}^n : \sum_{i=1}^{n} \pi_i z_i \le \tau, y_r \ge h_{ri}(1 - z_i),\ i \in [1, n] \right\}.$$

By dropping the index r, we redefine the mixing set with $0-1$ knapsack as

$$\mathcal{K} = \left\{ (y, \mathbf{z}) \in \mathbb{R}_+ \times \{0,1\}^n : \sum_{i=1}^{n} \pi_i z_i \le \tau,\ y \ge h_i(1 - z_i),\ i \in [1, n] \right\}.$$

Importantly, the study of the polyhedron of CCP is fundamental for solving chance–constrained programs efficiently. Most literature focuses on its substructure the set $\mathcal{K}$. Since our contributions include studies on the set $\mathcal{Q}$ and $\mathcal{K}$, we give literature review on both sets in following subsections.

1.1 Mixing set with $0-1$ knapsack

Observe that the set $\mathcal{K}$ consists of a mixing set, introduced by Günlük and Pochet [13] on general integer variables. The mixing set was extensively studied in varying degrees of generality by many authors in [2, 12, 23, 21, 33, 34] and its convex hull can be described by the so–called star inequalities in [2].

Without loss of generality, we can assume $h_1 \ge h_2 \ge \cdots \ge h_n \ge 0$ in the set $\mathcal{K}$. As in [1, 15, 18], we introduce two parameters ν and p. The parameter ν is defined such that

$$\sum_{i=1}^{\nu} \pi_i \le \tau \text{ and } \sum_{i=1}^{\nu+1} \pi_i > \tau.$$

As noted by [18], we have $y \ge h_{\nu+1}$ and the set $\mathcal{K}$ can be strengthened as follows

$$\left\{ (y, \mathbf{z}) \in \mathbb{R}_+ \times \{0,1\}^\nu : \sum_{i=1}^{n} \pi_i z_i \le \tau,\ y + (h_i - h_{\nu+1}) z_i \ge h_i,\ i \in [1, \nu] \right\}. \quad (3)$$

Indeed, by using $y + (h_i - h_{\nu+1}) z_i \ge h_i$ in (3) to replace (1), we obtain a new formulation of CCP with a tighter LP relaxation and less constraints. Let $\{\langle 1 \rangle, \langle 2 \rangle, \ldots, \langle n \rangle\}$ be a permutation of set $[1, n]$ with

$$\pi_{\langle 1 \rangle} \le \pi_{\langle 2 \rangle} \le \cdots \le \pi_{\langle n \rangle}.$$

The parameter p is defined such that

$$\sum_{i=1}^{p} \pi_{\langle i \rangle} \le \tau \text{ and } \sum_{i=1}^{p+1} \pi_{\langle i \rangle} > \tau.$$

Note that in the case of equal probabilities, i.e., $\pi_i = 1/n\ \forall i \in [1, n]$, the knapsack constraint reduces to the following cardinality constraint

$$\sum_{i=1}^{n} z_i \le p$$

and $p = \nu$. Luedtke et al. [18] applied the star inequality in [2] to the *strengthened star inequality* (which is stated as Theorem 1 in [15])

$$y + \sum_{j=1}^{a} (h_{t_j} - h_{t_{j+1}}) z_{t_j} \geq h_{t_1} \quad \forall T = \{t_1, \ldots, t_a\} \subseteq [1, \nu] \tag{4}$$

where $t_1 < \cdots < t_a$ and $h_{t_{a+1}} = h_{v+1}$, and showed that it is facet–defining for $\mathcal{K}$ when $t_1 = 1$. This result was generalized in [15] and [1] where more facet–defining inequalities were introduced for mixing set with either cardinality constraint or general knapsack. Luedtke et al. [18] and Küçükyavuz [15] also performed numerical studies for their proposed inequalities to evaluate the computational impact on solving lot-sizing based CCP instances.

1.2 Polyhedron of CCP

The polyhedron of CCP, i.e., $\mathcal{Q}$, was initially studied in [15]. When the $0-1$ knapsack in the set $\mathcal{Q}$ is just a cardinality constraint, the author in [15] developed so–called TL inequalities and showed that they are facet–defining for both $\mathcal{K}$ and $\mathcal{Q}$. We actually find that this result could be greatly generalized to any facet–defining inequalities as follows.

Proposition 1.1 (i) If an inequality is valid and facet-defining for $\mathcal{Q}_r$ for some $r \in [1, d]$, then the inequality is valid and facet-defining for $\mathcal{Q}$; moreover, (ii) if an inequality is valid and facet-defining for $\cap_{r \in \mathcal{D}} \mathcal{Q}_r$ for a set $\mathcal{D} \subseteq [1, d]$, then the inequality is valid and facet-defining for $\mathcal{Q}$.

Proof We omit the proof since it is the same as the last paragraph of the proof for Theorem 4 in [15]. □

Clearly, this proposition implies that the study of the single set $\mathcal{K}$ (i.e., a single $\mathcal{Q}_r$) provides a crucial polyhedral description to the set $\mathcal{Q}$. However, it can only bring us inequalities with at most one nonzero coefficient of y_r for some $r \in [1, d]$. As illustrated by a numerical example in [15], non-trivial inequalities for the polyhedron of CCP with $d = 2$, which are not obtainable from its single mixing subset $\mathcal{Q}_1$ or $\mathcal{Q}_2$, can be obtained by studying an aggregation of $\mathcal{Q}_1$ and $\mathcal{Q}_2$. Specifically, it involves selecting two scalars β_1, β_2, setting $y = \beta_1 y_1 + \beta_2 y_2$, and obtaining a new set

$$\mathcal{Q}' = \left\{ (y, \mathbf{z}) \in \mathbb{R}_+ \times \{0, 1\}^n : \sum_{i=1}^{n} \pi_i z_i \leq \tau, y \geq (\beta_1 h_{1i} + \beta_2 h_{2i})(1 - z_i) \ \forall i \in [1, n] \right\}.$$

Then, all the results on the set $\mathcal{K}$ can be readily applied to $\mathcal{Q}'$. Nevertheless, no analytical study has been done and the effectiveness of this combining approach is either theoretically or computationally unknown. For example, we do not know if this approach will derive any facet–defining inequalities of $\mathcal{Q}$.

1.3 Main contributions and outline

Our main contributions are in both theory and computation, and presented in each of the following sections in details. Here we summarize our contributions as follows,

- In Section 2, we derive a new family of inequalities for the mixing set with general $0-1$ knapsack, which subsumes all known facet-defining inequalities proposed in [1, 15, 18] as special cases. Due to the flexibility of our generalized inequalities, we are able to develop a new separation heuristic different from those in [1, 15], which has a much less complexity and is guaranteed to identify the strongest (i.e., facet–defining) cutting planes for the polyhedron of CCP.
- In Section 3, we present another large family of inequalities by performing lifting and superadditive lifting procedures on cover inequalities of $0-1$ knapsack. Noting that all existing valid inequalities (including our results in Section 2) are based on star inequalities for the basic mixing set, our results are the first type of valid inequalities with a different structure. Unlike a characterization on parameters of valid inequalities derived in [1] for $\mathcal{K}$, our inequalities are explicit and constructive, which provide an implementable procedure to identify strong inequalities to strengthen the polyhedral description and to improve our solution capacity of CCP.
- In Section 4, we introduce a novel blending approach to study $\mathcal{Q}$ such that, instead of combining original formulation, we are blending facet–defining inequalities of $\mathcal{Q}_r$ for $r \in [1, d]$. Especially, we are able to show when the resulting inequalities are facet–defining for $\mathcal{Q}$. To the best of our knowledge, they are the first group of facet–defining inequalities capturing interactions among multiple mixing sets. We also develop a very effective separation heuristic that can find violated facet–defining blending inequalities.
- In Section 5, we fully test the *strengthened star inequality*, TL inequalities (as described in [15]) and our three families of inequalities. We show that our inequalities outperform other known inequalities in the literature and substantially improve a commercial solver's ability to solve large static probabilistic lot-sizing problems.

The last section concludes the paper.

2 Strong inequalities derived from mixing set

On the top of the classical mixing inequality, a family of strong inequalities was developed in [18] for the set $\mathcal{K}$ when the knapsack constraint was replaced by a cardinality constraint. Then, Küçükyavuz [15] generalized that result for the cardinality constrained mixing set and extended those inequalities as valid ones for $\mathcal{K}$. Recently, Abdi et al. [1] provided a characterization of valid inequalities for $\mathcal{K}$, and explicitly developed a set of facet-defining inequalities for $\mathcal{K}$ under some special conditions. In the following, we present a large family of strong inequalities for $\mathcal{K}$, and derive sufficient conditions under which our proposed inequalities are facet-defining. In order to understand the connections to existing research on $\mathcal{K}$, we provide a detailed analysis and numerical examples to show that explicit strong inequalities or facet-defining inequalities developed in [1, 15] are either dominated or subsumed by ours. Due to the flexibility of our inequalities, we can further present a separation heuristic which can identify potential facet–defining inequalities with a computational complexity less than those of [1, 15].

Theorem 2.1 For $m \in [1, \nu]$ and $q \in [0, p-m]$, we define

- a set $T = \{t_1, \ldots, t_a\} \subseteq [1, m]$ with $t_1 < \ldots < t_a$;
- a set L with a permutation $\Pi_L = \{l_1, \ldots, l_q\}$;
- a sequence of integers $s_j \in [0, \nu - m + 1]$ such that $0 \leq s_1 \leq \cdots \leq s_q \leq s_{q+1} = \nu - m + 1$.

If $L \subseteq [m + s_1 + 1, n]$ with $l_j \geq m + \min\{1 + s_j, s_{j+1}\}$ and

$$\sum_{i=1}^{m+s_j} \pi_i + \sum_{i=j}^{q} \pi_{k_i} > \tau \quad \forall j \in [1, q]$$

where $\{k_1, \ldots, k_q\}$ is a permutation of L with $\pi_{k_1} \geq \cdots \geq \pi_{k_q}$, we have the following inequality

$$y + \sum_{j=1}^{a} (h_{t_j} - h_{t_{j+1}}) z_{t_j} + \sum_{j=1}^{q} \delta_j (1 - z_{l_j}) \geq h_{t_1} \tag{5}$$

that is valid for $\mathcal{K}$, where $t_{a+1} = m + s_1$ and

$$\delta_j = \begin{cases} h_{m+s_1} - h_{m+s_2} & j = 1 \\ \max\left\{ \delta_{j-1}, h_{m+s_1} - h_{m+s_{j+1}} - \displaystyle\sum_{i \in [1, j-1] \text{ and } l_i \geq m + \min\{1+s_j, s_{j+1}\}} \delta_i \right\} & j \in [2, q]. \end{cases} \tag{6}$$

Proof It is clear that if $y \geq h_{t_1}$, the inequality (5) is trivially satisfied. If $y \geq h_{t_i}$ for some $i = 2, \ldots, a+1$ and $y < h_{t_j}$ for all $j \in [1, i-1]$, then we must have $z_{t_j} = 1$ for all $j \in [1, i-1]$. Thus,

$$\begin{aligned} y + \sum_{j=1}^{a} (h_{t_j} - h_{t_{j+1}}) z_{t_j} &\geq h_{t_i} + \sum_{j=1}^{i-1} (h_{t_j} - h_{t_{j+1}}) \\ &= h_{t_1} \geq h_{t_1} - \sum_{j=1}^{q} \delta_j (1 - z_{l_j}) \end{aligned}$$

and inequality (5) is satisfied when $y \geq h_{t_{a+1}} = h_{m+s_1}$. Therefore, we assume that $y < h_{m+s_1}$, which implies $z_{t_j} = 1 \ \forall j = 1, \ldots, a$ and

$$\sum_{j=1}^{a} (h_{t_j} - h_{t_{j+1}}) z_{t_j} = h_{t_1} - h_{m+s_1}$$

in the rest of proof.

If $q = 0$, we have $m + s_1 = m + \nu - m + 1 = \nu + 1$. Such case is trivial because $y \geq h_{\nu+1}$ and the resulting inequality is the strengthened star inequality. It is sufficient to consider $q \geq 1$. Because $y \geq h_{\nu+1}$ and $s_{q+1} = \nu - m + 1$, we must have $h_{m+s_{i'}} > y \geq h_{m+s_{i'+1}}$ for some $i' = 1, \ldots, q$. Without loss of generality, we assume that $s_{i'+1} \geq s_{i'} + 1$, i.e., $m + \min\{1 + s_{i'}, s_{i'+1}\} = m + 1 + s_{i'}$. Thus, $z_j = 1$ for all $j = 1, \ldots, m + s_{i'}$, which implies

$$\sum_{i=1}^{m+s_{i'}} \pi_i + \sum_{i=m+s_{i'}+1}^{n} \pi_i z_i \leq \tau \tag{7}$$

from the knapsack inequality, and we have

$$
\begin{aligned}
&\sum_{j=1}^{q} \delta_j(1-z_{l_j}) = \sum_{j\in[1,q] \text{ and } l_j \geq m+s_{i'}+1} \delta_j(1-z_{l_j}) \\
= &\sum_{j=i'+1}^{q} \delta_j(1-z_{l_j}) + \sum_{j\in[1,i'] \text{ and } l_j \geq m+s_{i'}+1} \delta_j(1-z_{l_j}) \\
= &\sum_{j=i'+1}^{q} \delta_j + \sum_{j\in[1,i'] \text{ and } l_j \geq m+s_{i'}+1} \delta_j - \sum_{j=i'+1}^{q} \delta_j z_{l_j} - \sum_{j\in[1,i'] \text{ and } l_j \geq m+s_{i'}+1} \delta_i z_{l_j}. \qquad (8)
\end{aligned}
$$

The first equality holds because $z_j = 1\ \forall j \in [1, m+s_{i'}]$. The second equality holds because

$$
l_j \geq \begin{cases} m+s_j+1 \geq m+s_{i'}+1, \\ m+s_{j+1} \geq m+s_{i'+1} \geq m+s_{i'}+1 \end{cases} \quad \forall j \in [i'+1, q].
$$

Next, we show that

$$
\sum_{j=1}^{q} z_{l_j} \leq q - i' \qquad (9)
$$

by introducing a contradiction. Suppose $\sum_{j=1}^{q} z_{l_j} \geq q-i'+1$. We have

$$
\begin{aligned}
\tau \quad &\geq \quad \sum_{i=1}^{m+s_{i'}} \pi_i + \sum_{i=m+s_{i'}+1}^{n} \pi_i z_i \qquad (10) \\
&\geq \quad \sum_{i=1}^{m+s_{i'}} \pi_i + \sum_{j\in[1,q] \text{ and } l_j \geq m+s_{i'}+1} \pi_{l_j} z_{l_j} \\
&\geq \quad \sum_{i=1}^{m+s_{i'}} \pi_i + \sum_{j=i'}^{q} \pi_{l_j} z_{l_j} + \sum_{j\in[1,i'] \text{ and } l_j \geq m+s_{i'}+1} \pi_{l_j} z_{l_j} \qquad (11) \\
&\geq \quad \sum_{i=1}^{m+s_{i'}} \pi_i + \sum_{j=i'}^{q} \pi_{k_j} > \tau \qquad (12)
\end{aligned}
$$

where inequality (10) is just (7). Inequality (11) holds because

$$
l_j \geq \begin{cases} m+s_j+1 \geq m+s_{i'}+1, \\ m+s_{j+1} \geq m+s_{i'+1} \geq m+s_{i'}+1 \end{cases} \quad \forall j \in [i', q].
$$

Inequality (12) holds because $\sum_{j=1}^{q} z_{l_j} \geq q-i'+1$ and $\pi_{k_1} \geq \cdots \geq \pi_{k_q}$.

Given the contradiction introduced by (10)–(12), we have that (9) holds. Note that $\delta_1 \leq \cdots \leq \delta_{q+1}$ is monotonic. With (9), we get

$$
\sum_{j=i'+1}^{q} \delta_j \geq \sum_{j=i'+1}^{q} \delta_j z_{l_j} + \sum_{j\in[1,i'] \text{ and } l_j \geq m+s_{i'}+1} \delta_j z_{l_j}.
$$

Then from (8), we have

$$\begin{aligned}\sum_{j=1}^{q}\delta_j(1-z_{l_j}) &= \sum_{j=i'+1}^{q}\delta_j + \sum_{j\in[1,i'] \text{ and } l_j\geq m+s_{i'}+1}\delta_j - \sum_{j=i'+1}^{q}\delta_j z_{l_j} - \sum_{j\in[1,i'] \text{ and } l_j\geq m+s_{i'}+1}\delta_j z_{l_j}\\ &\geq \sum_{j\in[1,i'] \text{ and } l_j\geq m+s_{i'}+1}\delta_j\\ &\geq \delta_{i'} + \sum_{j\in[1,i'-1] \text{ and } l_j\geq m+s_{i'}+1}\delta_j \geq h_{m+s_1} - h_{m+s_{i'+1}}.\end{aligned}$$

The last inequality holds because of the definition of $\delta_{i'}$. Therefore, we have

$$\begin{aligned}& y + \sum_{j=1}^{a}(h_{t_j} - h_{t_{j+1}})z_{t_j} + \sum_{j=1}^{q}\delta_j(1-z_{l_j})\\ \geq\ & h_{m+s_{i'+1}} + h_{t_1} - h_{m+s_1} + h_{m+s_1} - h_{m+s_{i'+1}} \geq h_{t_1}.\end{aligned}$$

□

Next, we give necessary condition that (5) is facet–defining for $\mathcal{K}$.

Proposition 2.1 If (5) is facet–defining for $\mathcal{K}$, then we have $t_1 = 1$ and

$$\sum_{i=1}^{m+s_j-1}\pi_i + \sum_{i=j}^{q}\pi_{k_i} \leq \tau \quad \forall j \in [1,q] \text{ with } s_j \geq 1. \tag{13}$$

Proof Note that the inequality (5) is uniquely determined by a triple $(T, \Pi_L, \mathbf{s})$. We will denote (5) as the triple $(T, \Pi_L, \mathbf{s})$ in the proof. First, we will prove the necessary condition that $t_1 = 1$. Given a $(T, \Pi_L, \mathbf{s})$ with $t_1 > 1$. We can have $(T', \Pi_L, \mathbf{s})$ with $T' = T \cup \{1\}$, i.e.,

$$y + (h_1 - h_{t_1})z_1 + \sum_{j=1}^{a}(h_{t_j} - h_{t_{j+1}})z_{t_j} + \sum_{j=1}^{q}\delta_j(1-z_{l_j}) \geq h_1$$

which implies

$$(h_1 - h_{t_1})(z_1 - 1) + y + \sum_{j=1}^{a}(h_{t_j} - h_{t_{j+1}})z_{t_j} + \sum_{j=1}^{q}\delta_j(1-z_{l_j}) \geq h_{t_1}.$$

As $(h_1 - h_{t_1})(z_1 - 1) \leq 0$, The $(T', \Pi_L, \mathbf{s})$ is at least as strong as the $(T, \Pi_L, \mathbf{s})$ inequality.

Suppose the condition (13) does not hold for a $(T, \Pi_L, \mathbf{s})$ with coefficient δ_j $\forall j \in [1,q]$ and $i' \in [1,q]$ is the first index that the condition (13) does not hold. Thus, we have

$$\sum_{i=1}^{m+s_{i'}-1}\pi_i + \sum_{i=i'}^{q}\pi_{k_i} > \tau \quad \text{but} \quad \sum_{i=1}^{m+s_{i'-1}-1}\pi_i + \sum_{i=i'}^{q}\pi_{k_i} \leq \tau$$

which implies that $s_{i'-1} < s_{i'}$. We can define a triple $(T, \Pi_L, \mathbf{s}')$ such that $s'_j = s_j$ $\forall j \in [1,q] - \{i\}$ and $s'_{i'} = s_{i'} - 1$. Note that s' is still a monotonic sequence as s. It is clear that we have $\delta'_j \leq \delta_j$ $\forall j \in [1,q]$ where δ'_j is the coefficient for $(T, \Pi_L, \mathbf{s}')$. So $(T, \Pi_L, \mathbf{s}')$ inequality is at least as strong as the $(T, \Pi_L, \mathbf{s})$ inequality. □

Corollary 2.1.1 Let $q = p - m$ and $s_j = \min\{j, \nu - m + 1\}$ $\forall j \in [1, q]$, we have valid inequalities

$$y + \sum_{j=1}^{a} (h_{t_j} - h_{t_{j+1}}) z_{t_j} + \sum_{j=1}^{p-m} \delta_j (1 - z_{l_j}) \geq h_{t_1} \tag{14}$$

where

$$\delta_j = \begin{cases} h_{m+1} - h_{m+\min\{\nu-m+1,2\}} & j = 1 \\ \max\left\{ \delta_{j-1}, h_{m+1} - h_{m+\min\{\nu-m+1,j+1\}} - \sum_{i\in[1,j-1] \text{ and } l_i \geq m+1+\min\{\nu-m+1,j\}} \delta_i \right\} & j \in [2, q]. \end{cases}$$

Proof Because the definition of p in Section 1 implies that the summation of any $p + 1$ many π_i's for $i \in [1, n]$ is strictly greater than τ, we have the following result for any permutation $\{k_1, \ldots, k_q\}$ of L

$$\sum_{i=1}^{m+s_j} \pi_i + \sum_{i=j}^{q} \pi_{k_i} \geq \begin{cases} \sum_{i=1}^{m+j} \pi_i + \sum_{i=j}^{p-m} \pi_{k_i} \\ \sum_{i=1}^{m+(\nu-m+1)} \pi_i + \sum_{i=j}^{p-m} \pi_{k_i} \end{cases} > \tau \quad \forall j \in [1, q].$$

Apparently $s_1 = \min\{1, \nu - m + 1\} = 1$ since $m \in [1, \nu]$. Therefore, the expected result follows. □

Remark 1 (i) Corollary 2.1.1 is equivalent to Theorem 3 in [15] when the knapsack constraint is reduced to a cardinality constraint. In such case, we have $p = \nu$ and $s_j = \min\{j, p - m + 1\} = j$ $\forall j \in [1, p - m]$.
(ii) Corollary 2.1.1 improves Theorem 6 in [15], which is the main result in [15] for general $\mathcal{K}$. Note that

$$\{i \in [1, j - 1] : l_i \geq m + 1 + \min\{\nu - m + 1, j\}\} \supseteq \{i \in [1, j - 1] : l_i \geq m + 1 + j\}$$

and δ_js are all positive. Hence, inequality (14) is at least as strong as the one in [15].
(iii) We note that the choice of s_j in Corollary 2.1.1 does not need to satisfy (13). If it is the case, (14) could be dominated by (5), which is demonstrated in an example adopted from [15].

Example 1 (Example 1 in [15]) Let $h = (40, 38, 34, 31, 26, 16, 8, 4, 2, 1)$ for $n = 10$, and $\pi_1 = \cdots = \pi_4 = \tau/4$ and $\pi_5 = \cdots = \pi_{10} = \tau/6$ with $\tau = 0.5$. It is easy to check that $\nu = 4$ and $p = 6$. As showed in [15], inequality (14) with $m = 1$, $t_1 = 1$ and $\Pi_L = \{4, 6, 7, 8, 9\}$ gives

$$\begin{aligned} y + (h_1 - h_2) z_1 \;&+\; (h_2 - h_3)(1 - z_4) + (h_2 - h_3)(1 - z_6) + (h_2 - h_5 - \delta_2)(1 - z_7) \\ &+\; (h_2 - h_5 - \delta_2)(1 - z_8) + (h_2 - h_5 - \delta_2)(1 - z_9) \geq h_1 \end{aligned}$$

or specifically,

$$y + 2z_1 + 4(1 - z_4) + 4(1 - z_6) + 8(1 - z_7) + 8(1 - z_8) + 8(1 - z_9) \geq 40 \tag{15}$$

where $\delta_2 = h_2 - h_3$ is the coefficient for term $(1 - z_6)$. This inequality, nevertheless, is not facet–defining as it is dominated by

$$y + (h_1 - h_2)z_1 + (h_2 - h_3)(1 - z_4) + (h_2 - h_3)(1 - z_7) + (h_2 - h_5 - \delta_2)(1 - z_8) \geq h_1 \tag{16}$$

or specifically,

$$y + 2z_1 + 4(1 - z_4) + 4(1 - z_7) + 8(1 - z_8) \geq 40$$

which is valid and facet-defining. According to Theorem 1, inequality (16) can be generated by letting $m = 1$, $\Pi_L = \{4, 7, 8\}$ with $q = 3$. So $\{k_1, k_2, k_3\} = \{4, 7, 8\}$ as well since $\pi_4 \geq \pi_7 \geq \pi_8$. It is easy to see that we can choose $(s_1, s_2, s_3) = (1, 2, 3)$ because, for $j \in [1, q] = [1, 3]$, we have

$$\sum_{i=1}^{m+s_j} \pi_i + \sum_{i=j}^{q} \pi_{k_i} = \begin{cases} \frac{2}{4}\tau + \frac{1}{4}\tau + \frac{2}{6}\tau = \frac{13}{12}\tau & \text{when } j = 1 \\ \frac{3}{4}\tau + \frac{2}{6}\tau = \frac{13}{12}\tau & \text{when } j = 2 \\ \tau + \frac{1}{6}\tau = \frac{7}{6}\tau & \text{when } j = 3 \end{cases} > \tau.$$

Because of Proposition 2.1, we make the following assumption to have stronger inequalities (5).

Assumption 1 In Theorem 2.1, we always choose s_j $\forall j \in [1, q+1]$ such that

$$\sum_{i=1}^{m+s_j-1} \pi_i + \sum_{i=j}^{q} \pi_{k_i} \leq \tau \quad \forall j \in [1, q].$$

Next, we provide sufficient conditions that guarantee (5) to be facet-defining for $\mathcal{K}$.

Theorem 2.2 The inequality (5) is facet–defining for $\mathcal{K}$ if $t_1 = 1$, $\pi_{l_1} \geq \cdots \geq \pi_{l_q}$, and

$$\sum_{i=1}^{q} \pi_{l_i} + \pi_j \leq \tau \quad \forall j \notin T \cup L. \tag{17}$$

Proof The proof is similar to that of Theorem 4 in [15]. However, since our inequality (5) is more general, we give a self–contained proof. First, let $y^0 = h_1$, vector $\mathbf{z}^0$ with $\mathbf{z}^0_j = 1$ if $j \in L$ and $\mathbf{z}^0_j = 0$ otherwise. Next, for each $j \notin (T \cup L)$, we have point $(y^j, \mathbf{z}^j) = (y^0, \mathbf{z}^0 + e_j)$, where e_j is an n dimensional unit vector with jth component equal to 1. The point is feasible because of (17).

For each $j \in [1, a]$, let $y^{t_j} = h_{t_{j+1}}$, $\mathbf{z}^{t_j}_i = 1$ if $i \in [1, t_{j+1} - 1] \cup L$ and $\mathbf{z}^{t_j}_i = 0$ otherwise. The point is feasible because of the condition

$$\sum_{i=1}^{t_{j+1}-1} \pi_i + \sum_{i=1}^{q} \pi_{l_i} \leq \sum_{i=1}^{m+s_1-1} \pi_i + \sum_{i=1}^{q} \pi_{l_i} \leq \tau.$$

For $j \in [1, q]$, first we let $y^{l_1} = h_{m+s_2}$ and

$$\mathbf{z}^{l_j}_i = 1 \ \forall i \in [1, m + s_2 - 1] \cup \{l_2, \ldots, l_{q+1}\} \text{ and } \mathbf{z}^{l_j}_i = 0 \text{ otherwise.}$$

The point is feasible because of Assumption 1. Then, for $j \in [2, q + 1]$, if

$$\delta_j = h_{m+s_1} - h_{m+s_j} - \sum_{i \in [1, j-1] \text{ and } l_i \geq m+1+s_j} \delta_i,$$

let $y^{l_j} = h_{m+s_{j+1}}$ and

$$\mathbf{z}_i^{l_j} = 1 \ \forall i \in [1, m+s_{j+1}-1] \cup \{l_{j+1}, \ldots, l_{q+1}\} \text{ and } \mathbf{z}_i^{l_j} = 0 \text{ otherwise.}$$

If $\delta_j = \delta_{j-1}$, we let

$$(y^{l_j}, \mathbf{z}^{l_j}) = (y^{l_{j-1}}, \mathbf{z}^{l_{j-1}} + e_{l_{j-1}} - e_{l_j}).$$

Note that $\pi_{l_j} \geq \pi_{l_{j-1}}$. In either case, the point is feasible because of Assumption 1. These $n+1$ points on the face defined by inequality (5) are affinely independent. □

In the following, we consider an implementation strategy of this type of strong inequalities in our numerical study.

Corollary 2.2.1 When $|L| = 1$, the inequality (5) in Theorem 2.1 is facet-defining if $t_1 = 1$, and

$$\pi_{l_1} + \pi_j \leq \tau \quad \forall j \in [1, n] - \{l_1\}.$$

Next we will provide a separation algorithm for implementing Corollary 2.2.1.

Separation Heuristic We always set $t_1 = 1$. Let z^* be a fraction solution and we keep an ordered list of the elements in $\{i \in [\nu+1, n] : z_i^* = 1, \pi_i + \pi_j \leq \tau, \ \forall j \in [1, n] - \{i\}\}$, denoted as Ω, in decreasing order of π_i. We choose the first element in the list to be in the set L, i.e., $L = \{l_1\}$ where $l_1 = \text{argmax}_i\{\pi_i : i \in \Omega\}$. We let

$$m + s_1 = \min\left\{j : \sum_{i=1}^{j} \pi_i + \pi_{l_1} > \tau, j \in [1, \nu]\right\}$$

which can be found in $O(\nu)$. Then, the best choice of the set T in the inequality (5) can be found by solving a shortest path problem from the source 1 to the sink $m+s_1$ on a directed acyclic graph with vertices $\{1, \ldots, m+s_1\}$ (see Section 3.1 in [15]) and the complexity is $O(\nu^2)$. The coefficient $\delta_1 = h_{m+s_1} - h_{\nu+1}$. Note that $z_{l_1}^* = 1$ and shortest path algorithm is used to find the best choice of the set T. Therefore, our separation algorithm can find the most violated inequality in the form of Corollary 2.2.1, which is facet–defining for $\mathcal{Q}$, in $O(\nu^2)$.

Note that the separation algorithm in [15] is $O(p^4)$ without guarantee of finding facet–defining inequalities of $\mathcal{Q}$. Abdi et al. [1] mentioned that the same separation algorithm can be applied in their case without providing computational evaluation.

As mentioned, a set of facet–defining inequalities for $\mathcal{K}$ was developed in Theorem 14 of [1]. In the following, we analyze the connection between those facet–defining inequalities and results in Theorem 2.1 and 2.2. In particular, we show that they are subsumed as special cases of (5).

Corollary 2.2.2 Given a positive integer M such that $M\pi_i = a_i \ \forall i \in [1, n]$ and $M\tau = \mu$, the inequality (5) in Theorem 2.1 is valid for $\mathcal{K}$ if

- $a_i = 1 \ \forall i \in L$ and $q = \mu - \sum_{i=1}^{m} a_i$ is an integer;
- $s_1 = 1$ and $m + s_j = \mathfrak{m}(j-1) + 1 \ \forall j \in [2, q]$;
- $l_j > \mathfrak{m}(j) \ \forall j \in [1, q]$

where $\mathfrak{m}(j) = max\left\{k : j \geq \sum_{i=1}^{k} a_i - \sum_{i=1}^{m} a_i\right\}$. The inequality is facet-defining if $t_1 = 1$ and $a_j \leq \sum_{i=1}^{m} a_i \ \forall j \in [1, n] - L$.

It is not difficult to verify that Corollary 2.2.2 is equivalent to Theorem 14 in [1]. So, before providing a proof to this result, we make a few remarks on the applicability of Corollary 2.2.2 (equivalently, Theorem 14 of [1]).

Remark 2 *(i)* The condition that $a_i = 1 \ \forall i \in L$ requires that $\pi_i \ \forall i \in L$ are equal, which is a very special case of the knapsack constraint of $\mathcal{K}$. For example, Corollary 2.2.2 cannot explain (15) (derived in [15]) or (16) because $\pi_4 \neq \pi_6$. Note however that Corollary 2.1.1 or Theorem 6 in [15] is able to generate explicit valid inequalities for a general $\mathcal{K}$. Hence, different from the understanding made in [1], we cannot conclude that Corollary 2.2.2 subsumes Corollary 2.1.1 or Theorem 6 in [15].

(ii) The condition that $q = \mu - \sum_{i=1}^{m} a_i$ is an integer is very restrictive. Actually, Corollary 2.2.2 cannot describe any inequalities in Example 1 with $m = 1$. Since we need to have $a_i = 1 \ \forall i \in L$, we have either $L \subseteq \{1, \ldots, 4\}$ with $M = 8$ or $L \subseteq \{5, \ldots, 10\}$ with $M = 12$. If $L \subseteq \{1, \ldots, 4\}$, $q = M(\tau - \pi_1) = 3$. So, we must have $L = \{2, 3, 4\}$. However, $l_j > \mathfrak{m}(j) = j+1$ implies that the choice of L is not viable. If $L \subseteq \{5, \ldots, 10\}$, $q = M(\tau - \pi_1) = 4.5$, which is not an integer. So, there is no valid L satisfying those conditions.

(iii) Corollary 2.2.2 could fail to provide any inequalities other than *strengthened star inequalities*. Consider an example with $n = 8$, $\pi_1 = \ldots = \pi_3 = \tau/3$, and $\pi_4 = \ldots = \pi_8 = \tau/5$ where $\tau = 0.5$. Since we need to have $a_i = 1 \ \forall i \in L$, we have either $L \subseteq \{1, \ldots, 3\}$ with $M = 6$ or $L \subseteq \{4, \ldots, 8\}$ with $M = 10$. Suppose $m < \nu = 3$. To have that $q = \mu - \sum_{i=1}^{m} a_i$ is an integer, we can only set $L \subseteq \{1, \ldots, 3\}$ with $M = 6$. If $m = 1$ or 2, it is easy to check that $\mathfrak{m}(j) = j + m$, which implies that a viable choice of L with $l_j > \mathfrak{m}(j)$ does not exist. So, $m = \nu$, i.e., the only set of inequalities Corollary 2.2.2 implies is the *strengthened star inequalities*.

(iv) Comparing to Corollary 2.2.2, we mention that Theorem 2.1 has no such limitations but includes it as a special case. Therefore, Theorem 2.1 generalizes one of the main theorems in [1]. Note also that Corollary 2.2.2 does not imply Corollary 2.2.1 because $q = \frac{1}{\pi_{l_1}}(\tau - \sum_{i=1}^{m} \pi_i)$ might not be an integer.

Proof Note that $\forall j \in [1, q]$, $l_j > \mathfrak{m}(j)$ implies $l_j \geq m + s_{j+1}$, and

$$s_j = \mathfrak{m}(j-1) - m + 1 = \text{argmax}_s \left\{ \sum_{i=1}^{m+s} a_i - \sum_{i=1}^{m} a_i \leq j - 1 \right\} + 1 = \text{argmin}_s \left\{ \sum_{i=1}^{m+s} a_i - \sum_{i=1}^{m} a_i > j - 1 \right\}.$$

For any $j \in [1, q]$, we have

$$\begin{aligned}
\sum_{i=1}^{m+s_j} \pi_i + \sum_{i=j}^{q} \pi_{k_i} &= \frac{1}{M}\left(\sum_{i=1}^{m+s_j} a_i + \sum_{i=j}^{q} a_{k_i}\right) \\
&= \frac{1}{M}\left(\sum_{i=1}^{m+s_j} a_i + q - j + 1\right) \\
&= \frac{1}{M}\left(\sum_{i=1}^{m+s_j} a_i - \sum_{i=1}^{m} a_i + \mu - j + 1\right) \\
&> \frac{1}{M}(M\tau) > \tau
\end{aligned} \tag{18}$$

where (18) holds because $a_i = 1 \ \forall i \in L$. By Theorem 2.1, the inequality (2.1) is valid for $\mathcal{K}$. Since

$$\mu = q + \sum_{i=1}^{m} a_i = \sum_{i \in L} a_i + \sum_{i=1}^{m} a_i \geq \sum_{i \in L} a_i + a_j \ \forall j \in [1, n]/L,$$

(17) holds if we divided it by M. Thus, Theorem 2.2 implies (2.1) is facet–defining. □

3 Strong inequalities derived from lifting

In Section 2, although a large number facet-defining inequalities subsuming those identified in [1, 15, 18] are proposed for $\mathcal{K}$, they are not sufficient to define its convex hull. For instance, we observe that the following inequalities are facet-defining for the set in Example 1, which, nevertheless, cannot be derived based on Theorem 2.1.

$$\begin{aligned}
\frac{1}{3}z_1 - \frac{1}{3}z_2 + z_3 + z_4 + z_5 + z_7 + z_8 + z_9 &\leq 4 + \frac{1}{3}(y - 40) + \frac{8}{3}(1 - z_{10}) \\
\frac{1}{3}z_1 - \frac{1}{3}z_2 + z_3 + z_4 + z_5 + z_6 + z_7 + z_9 &\leq 4 + \frac{1}{3}(y - 40) + \frac{8}{3}(1 - z_{10}) \\
\frac{1}{3}z_1 - \frac{1}{3}z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9 &\leq 4 + \frac{1}{3}(y - 40) + \frac{8}{3}(1 - z_5).
\end{aligned} \tag{19}$$

Indeed, as pointed out by Abdi and Fukasawa [1], set $\mathcal{K}$ has abundant polyhedral structure and many of its valid inequalities are related to $0-1$ knapsack polyhedron. They further presented a characterization of all valid inequalities of $\mathcal{K}$. Yet, from our understanding, such result does not have explicit representation for implementation, unless a general purpose solver is called for separation. To advance our understanding on $\mathcal{K}$ as well as our solution capacity for chance–constrained problem, we directly make use of the cover inequality of $0-1$ knapsack polyhedron and develop lifting techniques with consideration of mixing inequalities to derive valid inequalities. As demonstrated at the end of this section, inequalities in (19) can be simply obtained using this approach. We mention that, the family of inequalities obtained through this approach is, to the best of our knowledge, the first type that has a different structure from those of all star inequality based ones, including those in [1, 15, 18], as well as ours presented in Section 2.

For $m \in [1,\nu]$, let $N_0 = [1, m-1]$ and $N_1 = \{l_1, \dots, l_q\} \subseteq [\nu+2, n]$ with cardinality q such that

$$\sum_{i=1}^{m} \pi_i + \sum_{i \in N_1} \pi_i \le \tau \text{ and } \sum_{i=1}^{m+1} \pi_i + \sum_{i \in N_1} \pi_i > \tau. \tag{20}$$

We consider a restricted $0-1$ knapsack polytope $\mathcal{S}(N_0, N_1)$, where

$$\mathcal{S}(N_0, N_1) = \left\{ \mathbf{z} \in \{0,1\}^n : \sum_{i=1}^{n} \pi_i z_i \le \tau, z_i = 0 \ \forall i \in N_0 \text{ and } z_i = 1 \ \forall i \in N_1 \right\}.$$

Let $\tau' = \tau - \sum_{i \in N_1} \pi_i$. Then, for set $\hat{N} = [1,n] - N_0 - N_1$, we assume that a cover $\mathbb{C}$ is available with respect to τ' and a lifted cover inequality (LCI) is derived as the following

$$\sum_{i \in E} \alpha_i z_i \le |\mathbb{C}| - 1 \tag{21}$$

with $\alpha_i > 0 \ \forall i \in E \subseteq \hat{N}$ and $\alpha_i = 0 \ \forall i \in \hat{N} - E$.

We then define the following function with set $W \subseteq E$ and scalar $\beta \le \sum_{i=1}^{m} \pi_i$.

$$\begin{aligned} G(W, \beta) = \max \quad & \sum_{i \in W} \alpha_i z_i \\ \text{s.t.} \quad & \sum_{i \in W} \pi_i z_i \le \tau' - \beta, \ \ z_i \in \{0,1\} \ \forall i \in W. \end{aligned}$$

The function G is well defined because of (20). Note that the lifting function of (21) can be defined as $|\mathbb{C}| - 1 - G(E, \beta)$. Let $\bar{\rho} = G\left(E, \sum_{i=1}^{m-1} \pi_i\right) - G\left(E - \{m\}, \sum_{i=1}^{m} \pi_i\right) - \alpha_m$, which is nonnegative noting that an optimal solution of $G\left(E - \{m\}, \sum_{i=1}^{m} \pi_i\right)$, together with $z_m = 1$, is just feasible to $G\left(E, \sum_{i=1}^{m-1} \pi_i\right)$.

Theorem 3.1 Let $m \in [1,\nu]$ and $N_1 \subseteq [\nu+2, n]$. Given a superadditive function $\Phi(\beta)$ such that $0 \le \Phi(\beta) \le |\mathbb{C}| - 1 - G(E, \beta)$. If $h_m > h_{m+1}$ and $0 \le \rho \le \bar{\rho}$, then the inequality

$$\sum_{i=1}^{m-1} \phi_i z_i + \sum_{i \in E} \alpha_i z_i \le |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}} (y - h_1) \tag{22}$$

is valid for the set $\mathcal{K}(N_1) = \mathcal{K} \cap \{\mathbf{z} \in \{0,1\}^n : z_i = 1 \ \forall i \in N_1\}$, where

$$\phi_i = \Phi(\pi_i) + \frac{\rho}{h_m - h_{m+1}} (h_{i+1} - h_i) \quad \forall i = 1, \dots, m-1. \tag{23}$$

Proof By the definition of $\mathcal{K}(N_1)$, we can fix $z_i = 1 \ \forall i \in N_1$ for the rest of proof. Note that $y \ge h_{m+1}$ in the set $\mathcal{K}(N_1)$, because of condition (20). Next, we consider all three situations based on the value of y.

First, if $y \ge h_1$, we assume that $z_i = 1 \ \forall i \in Q \subseteq [1, m-1]$ and $z_i = 0 \ \forall i \in [1, m-1] - Q$ for a set Q. Then the knapsack constraint in $\mathcal{K}(N_1)$ becomes

$$\sum_{i \in E} \pi_i z_i \le \tau' - \sum_{i \in Q} \pi_i$$

and we have

$$\begin{aligned}\sum_{i=1}^{m-1}\phi_i z_i+\sum_{i\in E}\alpha_i z_i &\leq \sum_{i\in Q}\Phi(\pi_i)+\sum_{i\in E}\alpha_i z_i\\ &\leq \Phi\left(\sum_{i\in Q}\pi_i\right)+\sum_{i\in E}\alpha_i z_i\\ &\leq \Phi\left(\sum_{i\in Q}\pi_i\right)+G\left(E,\sum_{i\in Q}\pi_i\right)\leq |\mathbb{C}|-1\\ &\leq |\mathbb{C}|-1+\frac{\rho}{h_m-h_{m+1}}(y-h_1)\end{aligned}$$

where the first inequality holds because $\phi_i \leq \Phi(\pi_i)$.

Second, if $h_{t-1} > y \geq h_t$ for some $t = 2, \ldots, m$. We must have $z_i = 1\ \forall i \in [1, t-1]$, and also assume that $z_i = 1\ \forall i \in Q \subseteq [t, m-1]$ and $z_i = 0\ \forall i \in [1, m-1] - Q$ for a set Q. Then the knapsack constraint in $\mathcal{K}(N_1)$ becomes

$$\sum_{i\in E}\pi_i z_i \leq \tau' - \sum_{i=1}^{t-1}\pi_i - \sum_{i\in Q}\pi_i$$

and we get

$$\begin{aligned}&\sum_{i=1}^{m-1}\phi_i z_i+\sum_{i\in E}\alpha_i z_i = \sum_{i=1}^{t-1}\phi_i+\sum_{i=t}^{m-1}\phi_i z_i+\sum_{i\in E}\alpha_i z_i\\ \leq\ &\sum_{i=1}^{t-1}\Phi(\pi_i)+\frac{\rho}{h_m-h_{m+1}}(h_t-h_1)+\sum_{i\in Q}\Phi(\pi_i)+\sum_{i\in E}\alpha_i z_i\\ \leq\ &\Phi\left(\sum_{i=1}^{t-1}\pi_i+\sum_{i\in Q}\pi_i\right)+\frac{\rho}{h_m-h_{m+1}}(h_t-h_1)+\sum_{i\in E}\alpha_i z_i\\ \leq\ &\Phi\left(\sum_{i=1}^{t-1}\pi_i+\sum_{i\in Q}\pi_i\right)+\frac{\rho}{h_m-h_{m+1}}(h_t-h_1)+G\left(E,\sum_{i=1}^{t-1}\pi_i+\sum_{i\in Q}\pi_i\right)\\ \leq\ &|\mathbb{C}|-1+\frac{\rho}{h_m-h_{m+1}}(y-h_1).\end{aligned}$$

Otherwise, $h_m > y \geq h_{m+1}$ and $z_i = 1 \; \forall i \in [1, m]$. So, we get

$$\begin{aligned}
\sum_{i=1}^{m-1} \phi_i z_i + \sum_{i \in E} \alpha_i z_i &= \sum_{i=1}^{m-1} \phi_i + \alpha_m + \sum_{i \in E-\{m\}} \alpha_i z_i \\
&\leq \sum_{i=1}^{m-1} \Phi(\pi_i) + \frac{\rho}{h_m - h_{m+1}}(h_m - h_1) + \alpha_m + \sum_{i \in E-\{m\}} \alpha_i z_i \\
&= \Phi\left(\sum_{i=1}^{m-1} \pi_i\right) + \frac{\rho}{h_m - h_{m+1}}(h_{m+1} - h_1) + \rho + \alpha_m + \sum_{i \in E-\{m\}} \alpha_i z_i \\
&\leq \Phi\left(\sum_{i=1}^{m-1} \pi_i\right) + \frac{\rho}{h_m - h_{m+1}}(h_{m+1} - h_1) + G\left(E, \sum_{i=1}^{m-1} \pi_i\right) \qquad (24) \\
&\leq |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}}(h_{m+1} - h_1) \\
&\leq |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}}(y - h_1)
\end{aligned}$$

where the inequality (24) holds because of the definition of ρ.

With analysis on all three cases, it can be seen that the inequality (22) is valid for the set $\mathcal{K}(N_1)$. □

Remark 3 (i) LCI in (21) can be easily obtained through the sequential lifting algorithm in [30]. Note that α_i are positive integers for $i \in E$, and $G(E, \beta)$ (also the lifting function $|\mathbb{C}| - 1 - G(E, \beta)$) will be a staircase function [9]. Then, a dynamic program algorithm can be used to analytically describe the lifting function, whose superadditive approximation $\Phi(\beta)$ can be constructed using the approximation method developed in [11].
(ii) To support our example at the end of this section and to make this paper self–contained, we next provide a valid superadditive approximation for the lifting function $|\mathbb{C}| - 1 - G(E, \beta)$. Our intention here is for demonstration, rather than deriving the strongest but complicated ones, i.e., maximal and non-dominated approximations, which certainly is a future research direction. Assume step lengths of lifting function are $p_0, \ldots, p_{|\mathbb{C}|-1}$. We sort them from large to small to obtain a permutation $(p_{a_0}, \ldots, p_{a_{|\mathbb{C}|-1}})$. Then, by [11], the following function is a valid superadditive approximation.

$$\Phi(\beta) = \begin{cases} 0 & \text{if } 0 \leq \beta < p_{a_0} \\ \mathfrak{h} & \text{if } \sum_{k=0}^{\mathfrak{h}-1} p_{a_k} \leq \beta < \sum_{k=0}^{\mathfrak{h}} p_{a_k}, \; \mathfrak{h} = 1, \ldots, |\mathbb{C}| - 1. \end{cases}$$

Note that when $\bar{\rho} = \rho = 0$, (22) reduces to an LCI of $0-1$ knapsack set $\mathcal{S}(\emptyset, N_1)$. Then performing exact lifting with respect to variables fixed at 1 provides us a valid inequality for both $\mathcal{S}(\emptyset, \emptyset)$ and $\mathcal{K}$. It actually corresponds the observation made in [1] that facet-defining inequalities of $\mathcal{S}(\emptyset, \emptyset)$ are also facet-defining for $\mathcal{K}$. To study more interesting interactions between mixing set and $0-1$ knapsack set, we next limit ourselves to the case that $\rho > 0$ in Theorem 3.1 and investigate lifting (22) sequentially with respect to variables $z_i \; \forall i \in N_1$ in the order of $\{l_1, \ldots, l_q\}$. Suppose we already have lifting coefficients for $z_{l_1}, \ldots, z_{l_{r-1}}$ such that

$$\sum_{i=1}^{m-1} \alpha_i z_i + \sum_{i \in E} \alpha_i z_i \leq |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}}(y - h_1) + \sum_{j=1}^{r-1} \delta_{l_j}(1 - z_{l_j}) \qquad (25)$$

is valid for the set $\mathcal{K}\cap\{\mathbf{z}\in\{0,1\}^n : z_i = 1\ \forall i\in\{l_r,\ldots,l_q\}\}$, and we are lifting inequality (25) with respect to variable z_{l_r}. By denoting the lifting coefficient as δ_{l_r}, we have

Proposition 3.1 The lifting coefficient δ_{l_r} can be defined as

$$\delta_{l_r} = \sum_{j=1}^{r-1}\delta_{l_j} + \frac{\rho}{h_m - h_{m+1}}h_1 - |\mathbb{C}| - 1 + \max_{t\in[1,\nu_r+1]}\left\{\Delta_{t,r} - \frac{\rho}{h_m - h_{m+1}}h_t\right\} \tag{26}$$

where

$$\eta_r = \max\left\{j : \sum_{i=1}^{j}\pi_i \le \tau' + \sum_{i=1}^{r}\pi_{l_i}\right\}$$

and for $t\in[1,\eta_r+1]$ we have

$$\begin{aligned}\Delta_{t,r} \ge \sum_{i=1}^{t-1}\alpha_i \quad + \quad \max & \sum_{i\in E\cup[t,m-1]}\alpha_i z_i + \sum_{j=1}^{r-1}\delta_{l_j}z_{l_j} \\ \text{s.t.} & \sum_{i\in[t,n]-N_1}\pi_i z_i + \sum_{j=1}^{r-1}\pi_{l_j}z_{l_j} \le \tau' + \sum_{j=1}^{r}\pi_{l_j} - \sum_{i=1}^{t-1}\pi_i \\ & z_i\in\{0,1\}\quad \forall i\in[t,n]-\{l_r,\ldots,l_q\}.\end{aligned} \tag{27}$$

Proof It is sufficient to show that

$$\sum_{i=1}^{m-1}\alpha_i z_i + \sum_{i\in E}\alpha_i z_i \le |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}}(y - h_1) + \sum_{j=1}^{r}\delta_{l_j}(1 - z_{l_j}) \tag{28}$$

is valid for the set $\mathcal{K}\cap\{\mathbf{z}\in\{0,1\}^n : z_i = 1\ \forall i\in\{l_{r+1},\ldots,l_q\}$ and $z_{l_r}=0\}$.

Suppose $h_{t-1} > y \ge h_t$ for some $t\in[1,\nu+1]$, where we denote h_0 as $+\infty$. It implies that $z_j = 1$ $\forall j\in[1,t-1]$ ($[1,t-1]=\emptyset$ when $t=1$). Then we note that the knapsack constraint in $\mathcal{K}$ becomes

$$\sum_{i\in[t,n]-N_1}\pi_i z_i + \sum_{j=1}^{r-1}\pi_{l_j}z_{l_j} \le \tau' + \sum_{j=1}^{r}\pi_{l_j} - \sum_{i=1}^{t-1}\pi_i$$

and we have

$$\sum_{i=1}^{m-1}\alpha_i z_i + \sum_{i\in E}\alpha_i z_i + \sum_{j=1}^{r}\delta_{l_j}z_{l_j} = \sum_{i=1}^{t-1}\alpha_i + \sum_{i\in E\cup[t,m-1]}\alpha_i z_i + \sum_{j=1}^{r-1}\delta_{l_j}z_{l_j} \le \Delta_{t,r}.$$

The definition of δ_{l_r} in (26) implies that

$$\begin{aligned}&\delta_{l_r} \ge \sum_{j=1}^{r-1}\delta_{l_j} + \frac{\rho}{h_m - h_{m+1}}h_1 - |\mathbb{C}| - 1 + \Delta_{t,r} - \frac{\rho}{h_m - h_{m+1}}h_t \\ \Rightarrow\quad & \Delta_{t,r} \le |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}}(h_t - h_1) + \sum_{j=1}^{r-1}\delta_{l_j} \\ \Rightarrow\quad & \Delta_{t,r} \le |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}}(y - h_1) + \sum_{j=1}^{r-1}\delta_{l_j}.\end{aligned}$$

So the inequality (28) holds. □

Therefore, by summarizing Theorem 3.1 together with Proposition 3.1, we get

Theorem 3.2 If $h_m > h_{m+1}$, then, for any $\rho \leq \overline{\rho}$, the inequality

$$\sum_{i=1}^{m-1} \phi_i z_i + \sum_{i \in E} \alpha_i z_i \leq |\mathbb{C}| - 1 + \frac{\rho}{h_m - h_{m+1}}(y - h_1) + \sum_{j=1}^{q} \delta_{l_j}(1 - z_{l_j}) \tag{29}$$

is valid for $\mathcal{K}$, where ϕ_i $\forall i \in N_0$ are defined in (23) and δ_i $\forall i \in N_1$ are given by lifting procedure in (26).

Remark 4 Given that knapsack cover inequalities have been intensively implemented in commercial solvers, we point it out that they can be readily strengthened by Theorem 3.2. Although the computation of (26) is a little bit involved, we can address this issue by computing LP relaxations of (27) with a simple greedy algorithm.

Example *1* (cont.) We suppose general probabilities as shown previously, where $\nu = 4$. Let $m = 3$, $N_0 = \{1, 2\}$ and $N_1 = \{10\}$. Note that the condition (20) holds. So, we have set $\mathcal{S}(N_0, N_1)$ that includes a knapsack as follows,

$$\frac{\tau}{4}z_3 + \frac{\tau}{4}z_4 + \frac{\tau}{6}z_5 + \frac{\tau}{6}z_6 + \frac{\tau}{6}z_7 + \frac{\tau}{6}z_8 + \frac{\tau}{6}z_9 \leq \tau - \frac{\tau}{6}.$$

The set $\{3, 4, 5, 6, 8\}$ gives a minimal cover for this $0-1$ knapsack and we have the minimal cover inequality

$$z_3 + z_4 + z_5 + z_6 + z_8 \leq 4.$$

By lifting the cover inequality with respect to variable z_9, we have

$$z_3 + z_4 + z_5 + z_6 + z_8 + z_9 \leq 4.$$

For its lifting function, we note that all step lengths are $\frac{\tau}{6}$. According to the superadditive function presented in Remark 3, that lifting function is naturally superadditive. Therefore, Theorem 3.1 gives the following inequality

$$\begin{aligned}\left(1 + \frac{1}{h_3 - h_4}(h_2 - h_1)\right) z_1 \quad &+ \quad \left(1 - \frac{1}{h_3 - h_4}(h_3 - h_2)\right) z_2 + z_3 \\ &+ \quad z_4 + z_5 + z_6 + z_8 + z_9 \leq 4 + \frac{1}{h_3 - h_4}(y - h_1),\end{aligned}$$

or specifically

$$\frac{1}{3}z_1 - \frac{1}{3}z_2 + z_3 + z_4 + z_5 + z_6 + z_8 + z_9 \leq 4 + \frac{1}{3}(y - h_1), \tag{30}$$

which is valid for the set $\mathcal{K} \cap \left\{\mathbf{z} \in \{0,1\}^{10} : z_{10} = 1\right\}$. Then, by lifting inequality (30) with respect to z_{10}, we get

$$\Delta_{t1} = \begin{cases} 1 & \text{when } t = 1 \\ 1 & \text{when } t = 2 \\ 1 & \text{when } t = 3 \\ 1 & \text{when } t = 4 \\ \frac{8}{3} & \text{when } t = 5. \end{cases}$$

So $\delta_1 = 8/3$ and we have the valid inequality

$$\frac{1}{3}z_1 - \frac{1}{3}z_2 + z_3 + z_4 + z_5 + z_6 + z_8 + z_9 \leq 4 + \frac{1}{3}(y - 40) + \frac{8}{3}(1 - z_{10})$$

for $\mathcal{K}$, which is the first inequality in the list (19). Similar procedures can produce other inequalities in the list.

Implementation Because we cannot access the cover inequality generating procedure in commercial solvers, we added (29) into the initial formulation in our numerical study. Specifically, we let $N_1 = \emptyset$, i.e., $m = \nu$, cover $\mathbb{C}$ be obtained by Algorithm 1 in the following and E be an extended cover of $\mathbb{C}$. Since $\alpha_i = 1\ \forall i \in E$, the value of $\bar{\rho}$ can be obtained easily and we set $\rho = \bar{\rho}$. Then the inequalities (29) can be derived from Theorem 3.1 for $\mathcal{Q}_r\ \forall r \in [1, d]$. Overall, we could include up to d lifting inequalities into the initial formulation.

Algorithm 1 Find cover $\mathbb{C}$

Sort $\pi_i\ \forall i \in 1, \ldots, n$ such that $\pi_{\langle 1 \rangle} \leq \cdots \leq \pi_{\langle n \rangle}$
$\mathbb{C} \leftarrow \emptyset$
for $j = 1$ to n **do**
    if $\langle j \rangle \geq \nu$ **then**
        $\mathbb{C} \leftarrow \mathbb{C} \cup \{\langle j \rangle\}$
    if $\sum_{i \in \mathbb{C}} \pi_i > \tau$ **then** break
for $j = 1$ to $|\mathbb{C}|$ **do**
    if $\sum_{i \in \mathbb{C} - \{j\}} \pi_i > \tau$ **then**
        $\mathbb{C} \leftarrow \mathbb{C} - \{j\}$
    else
        break

4 Intersection of multiple mixing sets with knapsack

Up to now, we focus on deriving strong valid inequalities for a single mixing set with a 0-1 knapsack constraint, i.e., $\mathcal{K}$, which, according to Proposition 1.1, are crucial to understand the polyhedron $\mathcal{Q}$ of CCP. It is clear that such types of inequalities are not sufficient to describe $\mathcal{Q}$. Hence, in this section, we investigate the intersection of multiple mixing sets with knapsack constraint by developing valid inequalities that explicitly capture their interactions, i.e., valid inequalities with nonzero coefficients of multiple y_rs, where $r \in [1, d]$.

Given existing algebraic derivations of valid inequalities for $\mathcal{K}$, one direct approach to study $\mathcal{Q}$ is to aggregate multiple mixing sets (with assigned weights) into a single mixing set and derive valid inequalities from that aggregation, which definitely are valid for $\mathcal{Q}$. Using a numerical example, [15] illustrated that such strategy could lead to a new valid inequality that cannot be obtained from each individual mixing set. Nevertheless, observing that there is no systematical study on this approach, it remains unknown that: (i) how to select appropriate weights to generate a significant aggregation carrying rich interactive information among mixing sets; and (ii) what is the strength of inequalities obtained from that aggregation with respect to $\mathcal{Q}$. In this section, instead of aggregating original constraints, we propose a novel blending procedure that allows us to aggregate derived strong inequalities from individual mixing sets into a strong one for the polyhedron $\mathcal{Q}$. Moreover, a set of sufficient conditions is derived, which ensures the resulting *blending inequality* is facet-defining. It is worth mentioning that it is the first type of facet-defining inequalities for the intersection of multiple mixing sets.

Apparently, in the study of $\mathcal{Q}$, it is not valid to assume $h_{r1} \geq \cdots \geq h_{rn}$ for all $r \in [1, d]$ without loss of generality. So, we keep subscript r and for each $r \in [1, d]$, we define a $1-1$ mapping $\langle \cdot \rangle_r$ on $[1, n]$ such that

$$h_{r\langle 1\rangle_r} \geq h_{r\langle 2\rangle_r} \geq \cdots \geq h_{r\langle n\rangle_r}.$$

We also use the notation that $\langle X\rangle_r = \{\langle i\rangle_r : \forall i \in X\}$ for any set $X \subseteq [1, n]$. For each $\mathcal{Q}_r$, we define ν_r such that

$$\sum_{i=1}^{\nu_r} \pi_{\langle i\rangle_r} \leq \tau \text{ but } \sum_{i=1}^{\nu_r+1} \pi_{\langle i\rangle_r} > \tau.$$

Note that the value p is independent of index r, since it is based on the monotonic order of all π_i's.

Definition 1 Given $\theta \in [1, n]$, for each $r \in [1, d]$, let $m_r \in [1, \nu_r]$ and $q_r \in [0, p - m_r]$, we define

- a set $T_r = \{t_{r1}, t_{r2}, \ldots, t_{ra_r}\} \subseteq \{1, \ldots, m_r\}$ with $t_{r1} < t_{r2} < \cdots < t_{ra_r}$;
- a set L_r with a permutation $\Pi_{L_r} = \{l_{r1}, l_{r2}, \ldots, l_{r,q_r}\}$ and $l_{r1} = \theta$;
- a sequence of integers $s_{rj} \in [0, \nu_r - m_r + 1]$ $\forall j \in [1, q + 1]$ such that
  - $L_r \subseteq \{m_r + s_{r1} + 1, \ldots, n\}$ with $l_{rj} \geq m_r + \min\{s_{rj} + 1, s_{r,j+1}\}$;
  - $0 \leq s_{r1} \leq \cdots \leq s_{q_r+1} = \nu_r - m_r + 1$; and
  - $$\sum_{i=1}^{m_r+s_{rj}} \pi_{\langle i\rangle_r} + \sum_{i=j}^{q_r} \pi_{\langle k_{ri}\rangle_r} > \tau \;\; \forall j \tag{31}$$

  where $\{k_{r1}, \ldots, k_{rq_r}\}$ is a permutation of set L_r with $\pi_{\langle k_{r1}\rangle_r} \geq \cdots \geq \pi_{\langle k_{rq_r}\rangle_r}$;
- an expression

  $$\mathcal{I}_r = y_r + \sum_{j=1}^{a_r} (h_{r\langle t_{rj}\rangle_r} - h_{r\langle t_{r,j+1}\rangle_r}) z_{\langle t_{rj}\rangle_r} + \sum_{j=1}^{q_r} \delta_{rj}(1 - z_{\langle l_{rj}\rangle_r})$$

  where

$$\delta_{rj} = \begin{cases} h_{r,\langle m_r+s_{r1}\rangle_r} - h_{r,\langle m_r+s_{r2}\rangle_r} & j = 1 \\ \max\left\{\delta_{r,j-1}, h_{r,\langle m_r+s_{r1}\rangle_r} - h_{r,\langle m_r+s_{r,j+1}\rangle_r} - \displaystyle\sum_{i:i<j \text{ and } l_{ri} \geq m_r+\min\{1+s_{rj}, s_{r,j+1}\}} \delta_{ri}\right\} & j \in [2, q_r]. \end{cases}$$

When the sets $\langle L_r\rangle_r - \{\theta\}$ $\forall r \in [1, d]$ are mutually disjoint, we define the *blending inequality* as

$$\sum_{r\in[1,d]} \frac{1}{\delta_{r1}} \mathcal{I}_r - (1 - z_\theta) \geq \sum_{r\in[1,d]} \frac{1}{\delta_{r1}} h_{r\langle t_{r1}\rangle_r}. \tag{32}$$

Next, we give a necessary condition that the *blending inequality* is valid for $\mathcal{Q}$.

Theorem 4.1 The blending inequality is valid for $\mathcal{Q}$ if

$$\sum_{i \in \bigcup_{r \in [1,d]} \langle [1, m_r + s_{r1}] \rangle_r} \pi_i > \tau. \tag{33}$$

Proof Consider a single mixing set $\mathcal{Q}_r$ for a given $r \in [1, d]$. We can assume $\langle i \rangle_r = i \ \forall i \in [1, n]$ without loss of generality. Thus, the next inequality is exactly the inequality (5) for $\mathcal{Q}_r$

$$\mathcal{I}_r \geq h_{r \langle t_{r1} \rangle_r}. \tag{34}$$

Since $\mathcal{Q}_r \supseteq \mathcal{Q}$, the inequality (34) is valid for $\mathcal{Q}$. Next, we show that for some $u \in [1, d]$, $\mathcal{I}_u - \delta_{u1}(1 - z_\theta) \geq h_{u \langle t_{u1} \rangle_u}$ is valid for $\mathcal{Q}_u$.

Because of the condition (33), we claim that there exists some $u \in [1, d]$ such that $y_u \geq h_{u, m_u + s_{u1}}$. Suppose the claim is not true. Then we have $y_r < h_{r, m_r + s_{r1}}$ for all $r \in [1, d]$. It implies that $z_i = 1 \ \forall i \in \langle [1, m_r + s_{r1}] \rangle_r, r \in [1, d]$, which violates the knapsack constraint because of (33).

Then, for that particular u, we assume $y_u \geq h_{u, m_u + s_{u1}}$. Because we are considering single mixing set, without loss of generality, we can assume $\langle i \rangle_u = i \ \forall i \in [1, n]$ and write the inequality $\mathcal{I}_u - \delta_{u1}(1 - z_\theta) \geq h_{u \langle t_{u1} \rangle_u}$ explicitly as follows,

$$y_u + \sum_{j=1}^{a_u} (h_{ut_{uj}} - h_{ut_{u,j+1}}) z_{t_{uj}} + \sum_{j=2}^{q_u} \delta_{uj}(1 - z_{l_{uj}}) \geq h_{ut_{u1}}.$$

To simplify the notation, we can drop subscript u and have inequality

$$y + \sum_{j=1}^{a} (h_{t_j} - h_{t_{j+1}}) z_{t_j} + \sum_{j=2}^{q} \delta_j (1 - z_{l_j}) \geq h_{t_1}.$$

Because of the assumption that $y_u \geq h_{u, m_u + s_{u1}}$, we need to show that the above inequality is valid when $y \geq h_{m + s_1}$, which is already proved in the first paragraph of the proof for Theorem 2.1. Now, we have $\mathcal{I}_u - \delta_{u1}(1 - z_\theta) \geq h_{u \langle t_{u1} \rangle_u}$ is valid for $\mathcal{Q}_u$, i.e., valid for $\mathcal{Q}$.

Together with (34) for any $r \in [1, d] - \{u\}$, we get

$$\begin{aligned}
\sum_{r \in [1,d]} \frac{1}{\delta_{r1}} \mathcal{I}_r - (1 - z_\theta) &= \sum_{r \in [1,d] - \{u\}} \frac{1}{\delta_{r1}} \mathcal{I}_r + \frac{1}{\delta_{u1}} \mathcal{I}_u - (1 - z_\theta) \\
&\geq \sum_{r \in [1,d] - \{u\}} \frac{1}{\delta_{r1}} h_{rt_{r1}} + \frac{1}{\delta_{u1}} h_{ut_{u1}} = \sum_{r \in [1,d]} \frac{1}{\delta_{r1}} h_{rt_{r1}}.
\end{aligned}$$

Therefore, the blending inequality is valid for $\mathcal{Q}$. □

When all the scenarios have equal probabilities, the condition that the *blending inequality* is valid follows immediately after Theorem 4.1.

Corollary 4.1.1 Suppose all the scenarios have equal probabilities, then the blending inequality is valid for $\mathcal{Q}$ if

$$\left| \bigcup_{r \in [1,d]} \langle [1, m_r + 1] \rangle_r \right| > p. \tag{35}$$

In the next theorem, we show that the *blending inequality* is facet–defining under certain conditions.

Theorem 4.2 Suppose all scenarios have equal probabilities, $d = 2$ and (35) is satisfied. The *blending inequality* is facet-defining for $\mathcal{Q}$ if, $\forall r \in \{1, 2\}$, the sets $\langle T_r\rangle_r$, $\langle L_r\rangle_r - \{\theta\}$ are mutually disjoint, $t_{r1} = 1$ and we have

1. $\langle 1\rangle_1 \notin \langle[1, m_2]\rangle_2 \bigcup \langle L_2\rangle_2$, $\langle 1\rangle_2 \notin \langle[1, m_1]\rangle_1 \bigcup \langle L_1\rangle_1$;
2. $\langle L_1\rangle_1 - \{\theta\} \subsetneq \langle[1, m_2]\rangle_2$, $\langle L_2\rangle_2 - \{\theta\} \subsetneq \langle[1, m_1]\rangle_1$.

Proof To show that the blending inequality is facet–defining for $conv(\mathcal{Q})$, we give $n + 2$ affinely independent points. The idea is that we can always set $y_r = h_{r\langle 1\rangle_r}$ and enumerate extreme points as in the proof of Theorem 2.2 for $y_{\bar{r}}$ because of condition 1.

First, we let $y_1^0 = h_{1\langle 1\rangle_1}, y_2^0 = h_{2\langle 1\rangle_2}$, and $\mathbf{z}_j^0 = 1$ if $j \in \langle L_1\rangle_1 \bigcup \langle L_2\rangle_2$. Note that

$$\langle L_1\rangle_1 \cup \langle L_2\rangle_2 \subsetneq \langle L_1\rangle_1 \cup \langle[1, m_1]\rangle_1 \cup \{\theta\} = \langle L_1\rangle_1 \cup \langle[1, m_1]\rangle_1. \qquad (36)$$

So, $|\langle L_1\rangle_1 \bigcup \langle L_2\rangle_2| \le |\langle L_1\rangle_1 \cup \langle[1, m_1]\rangle_1| = p$ and the point is feasible.

Next, for each $j \notin \langle T_1 \cup L_1\rangle_1 \bigcup \langle T_2 \cup L_2\rangle_2$, we consider the point $(\mathbf{y}^j, \mathbf{z}^j) = (\mathbf{y}^0, \mathbf{z}^0 + e_j)$. For each $t_{1j} \in \{t_{11}, \dots, t_{1a_1}\}$, we let $y_2^{t_{2j}} = h_{2\langle 1\rangle_2}$, $y_1^{t_{1j}} = h_{1t_{1,j+1}}$, $\mathbf{z}_i^{t_{1j}} = 1$ if $i \in \langle[1, t_{1,j+1} - 1]\rangle_1 \cup \langle L_1\rangle_1 \cup \langle L_2\rangle_2$ and 0 otherwise. The point is feasible because (36) implies

$$|\langle[1, t_{1,j+1} - 1]\rangle_1 \cup \langle L_1\rangle_1 \cup \langle L_2\rangle_2| \le |\langle L_1\rangle_1 \cup \langle[1, m_1]\rangle_1| = p.$$

Due to the symmetry, we have similar way to get points for each $t_{2j} \in \{t_{21}, \dots, t_{2a_2}\}$ by letting $y_1^{t_{1j}} = h_{1\langle 1\rangle_1}$.

Let $y_1^{l_{11}} = h_{1\langle 1\rangle_1}$, $y_2^{l_{21}} = h_{m_2+2}$, $\mathbf{z}_i^{l_{21}} = 1$ if $i \in \langle[1, m_2 + 1]\rangle_2$ and $\mathbf{z}_{l_{2i}}^{l_{21}} = 1$ for $i > 1$, and 0 otherwise. The point is feasible because of condition 2. For each $j \in [2, p - m_2]$ if $\delta_{2j} = \delta_{2,j-1}$, we have

$$(\mathbf{y}^{l_{2j}}, \mathbf{z}^{l_{2j}}) = (\mathbf{y}^{l_{2,j-1}}, \mathbf{z}^{l_{2,j-1}} + e_{l_{2,j-1}} - e_{l_{2j}}),$$

otherwise we have that $y_1^{l_{1j}} = h_{1\langle 1\rangle_1}$, $y_2^{l_{2j}} = h_{m_2+1+j}$, $\mathbf{z}_i^{l_{2j}} = 1$ if $i \in \langle[1, m + j]\rangle_2$ and $\mathbf{z}_{l_{2i}}^{l_{2j}} = 1$ for $i > j$, and 0 otherwise. Thus, we get points for each $l_{2j} \in \{l_{21}, \dots, l_{2,p-m_2}\}$. Due to the symmetry, we can also get points for each $l_{1j} \in \{l_{11}, \dots, l_{1,p-m_1}\}$.

Note that $\langle T_1\rangle_1$, $\langle T_2\rangle_1$, $\langle L_1\rangle_1$, $\langle L_2\rangle_2$ are mutually disjoint except $\langle L_1\rangle_1 \cap \langle L_2\rangle_2 = \{\theta\}$. So, we get totally $n + 2$ points on the face defined by blending inequality and they are affinely independent. $\square$

We next give a numerical example to illustrate Theorem 4.2.

Example 2 Let $n = 6$ and $p = 3$. Suppose we have equal probabilities for all scenarios and

$$(h_{11}, \dots, h_{16}) = (28, 25, 15, 8, 5, 3) \text{ and } (h_{21}, \dots, h_{26}) = (2, 5, 6, 8, 17, 10)$$

The inequality (5) implies that

$$\begin{aligned} y_1 + 3z_1 + 10(1 - z_3) + 17(1 - z_6) \ge 28 \quad & \text{is valid for } \mathcal{Q}_1 \text{ with } m_1 = 1 \\ y_2 + 9z_5 + 2(1 - z_3) \ge 17 \quad & \text{is valid for } \mathcal{Q}_2 \text{ with } m_2 = 2 \end{aligned}$$

Let $\theta = 3$. We have

- $\langle 1\rangle_1 = 1 \notin \{5,6\}\bigcup\{3\} = \langle[1,m_2]\rangle_2\bigcup\langle L_2\rangle_2$;
- $\langle 1\rangle_2 = 5 \notin \{1\}\bigcup\{3,6\} = \langle[1,m_1]\rangle_1\bigcup\langle L_1\rangle_1$;
- $\langle L_1\rangle_1 - \{\theta\} = \{3,6\} - \{3\} \subseteq \{5,6\} = \langle[1,m_2]\rangle_2$;
- $\langle L_2\rangle_2 - \{\theta\} = \{3\} - \{3\} \subseteq \{1\} = \langle[1,m_1]\rangle_1$.

Thus, the conditions in Theorem 4.2 are satisfied and we have a blending inequality

$$y_1 + 5y_2 + 3z_1 + 45z_5 + 10(1-z_3) + 17(1-z_6) \geq 113$$

which is valid and facet–defining for $\mathcal{Q}$.

Next, we give a special case of Theorem 4.2 and show how to use it as our separation algorithm.

Corollary 4.2.1 Suppose all scenarios have equal probabilities and $d = 2$. If $L_1 = L_2 = \{\theta\}$ for some $\theta \in [1,n]$, then the *blending inequality* is valid and facet-defining for $\mathcal{Q}$ if following conditions hold

- $\theta \in \langle[p+1,n]\rangle_1\bigcap\langle[p+1,n]\rangle_2$;
- $\langle T_1\rangle_1 \cap \langle T_2\rangle_2 = \emptyset$;
- $\langle 1\rangle_1 \notin \langle[1,p]\rangle_2\bigcup\{\theta\}$; and
- $\langle 1\rangle_2 \notin \langle[1,p]\rangle_1\bigcup\{\theta\}$.

Proof Because $\langle 1\rangle_1 \notin \langle[1,p]\rangle_2\bigcup\{\theta\}$, condition (35) holds, which implies that the *blending inequality* is valid. It is easy to check that all conditions in Theorem 4.2 are satisfied. Therefore, the *blending inequality* is valid and facet–defining for $\mathcal{Q}$. □

Separation Heuristic First, we collect a set R consisting of pairs (r_1, r_2) where $r_1 \neq r_2 \in [1,d]$, $\langle 1\rangle_{r_1} \notin \langle[1,p]\rangle_{r_2}$ and $\langle 1\rangle_{r_2} \notin \langle[1,p]\rangle_{r_1}$. Suppose z^* is a fraction solution. For each pair $(r_1,r_2) \in R$, let $\theta = \text{argmax}_i\{\pi_i : z_i^* = 1, i \in \langle[p+1,n]\rangle_{r_1}\bigcap\langle[p+1,n]\rangle_{r_2}\}$. Next, we find the set T_1 by solving a shortest path problem from the source $\langle 1\rangle_{r_1}$ to the sink $\langle p-1\rangle_{r_1}$ on a directed acyclic graph with vertices $\langle\{1,\ldots,p-1\}\rangle_{r_1}$. Finally, the set T_2 is obtained by solving another shortest path problem from the source $\langle 1\rangle_{r_2}$ to the sink $\langle p-1\rangle_{r_2}$ on a directed acyclic graph with vertices $\langle\{1,\ldots,p-1\}\rangle_{r_2} - \langle T_1\rangle_{r_1}$. Following Corollary 4.2.1, all violated inequalities found by this separation algorithm are facet–defining for $\mathcal{Q}$.

At the end of this section, we adopt an example from [15] to show that our *blending inequalities* are stronger than those generated from a single mixing set, which is obtained by combining original formulation.

Example 3 (Example 2 in [15]) We have the chance–constrained program

Table 1: Joint probability density function

Scenario	1	2	3	4	5	6	7	8	9
ω_1	0.75	0.5	0.5	0.25	0.25	0.25	0	0	0
ω_2	1.25	1.5	1.25	1.75	1.5	1.25	2	1.5	1.25
Probability	0.2	0.14	0.06	0.06	0.06	0.3	0.04	0.04	0.1

$$\begin{aligned} \min \quad & x_1 + x_2 \\ \text{s.t.} \quad & P\left\{\begin{aligned} 2x_1 - x_2 &\geq \omega_1 \\ x_1 + 2x_2 &\geq \omega_2 \end{aligned}\right\} \geq 0.6 = 1 - \tau \\ & x_1, x_2 \geq 0 \end{aligned}$$

where ω_1 and ω_2 are dependent random variables with joint probability density function given in Table 1. The optimal solution is $(x, y) = (0.55, 0.35, 0.75, 1.25)$ with objective value 0.9.

For this example, we have $\tau = 0.4$, $d = 2$, $p = 6$, $\nu_1 = 3$ and $\nu_2 = 5$. Let $y_1 = 2x_1 - x_2$ and $y_2 = x_1 + 2x_2$. Then the mixing set reformulation is

$$\begin{array}{rr} y_1 + 0.75z_1 \geq 0.75 & y_2 + 2.00z_7 \geq 2.00 \\ y_1 + 0.50z_2 \geq 0.50 & y_2 + 1.75z_4 \geq 1.75 \\ y_1 + 0.50z_3 \geq 0.50 & y_2 + 1.50z_2 \geq 1.50 \\ y_1 + 0.25z_4 \geq 0.25 & y_2 + 1.50z_5 \geq 1.50 \\ y_1 + 0.25z_5 \geq 0.25 & y_2 + 1.50z_8 \geq 1.50 \\ y_1 + 0.25z_6 \geq 0.25 & y_2 + 1.25z_1 \geq 1.25 \\ \vdots & \vdots \\ \sum_{i=1}^{9} \pi_i z_i \leq & 0.4 = \tau \end{array}$$

and the tighter formulation (3) in [18] is

$$\begin{array}{rr} y_1 + 0.50z_1 \geq 0.75 & y_2 + 0.75z_7 \geq 2.00 \\ y_1 + 0.25z_2 \geq 0.50 & y_2 + 0.50z_4 \geq 1.75 \\ y_1 + 0.25z_3 \geq 0.50 & y_2 + 0.25z_2 \geq 1.50 \\ y_1 \geq 0.25 & y_2 + 0.25z_5 \geq 1.50 \\ & y_2 + 0.25z_8 \geq 1.50 \\ \vdots & y_2 \geq 1.25 \\ & \vdots \\ \sum_{i=1}^{9} \pi_i z_i \leq & 0.4 = \tau. \end{array}$$

The initial linear programming (LP) relaxation solution by using tight formulation (3) is

$$(x, y) = (0.49, 0.38, 0.60, 1.25).$$

In [15], the author proposed 3 inequalities in the form of (14), which were generated for y_1 and y_2, respectively. Then the author combined two mixing sets with y_1 and y_2 (with equal weights), and derived a valid inequality $y_1 + y_2 \geq 2$ from that aggregation. Augmented with those inequalities, the updated LP relaxation has a solution with optimal (x, y).

For this example, based on our previously presented theorems, we are able to develop several new facet-defining inequalities. First, we have facet–defining inequality

$$y_2 + 0.25z_2 + 0.25z_4 + 0.25z_7 \geq 2 \tag{37}$$

which is a *strengthened star inequality* in [18] and also a special case of inequality (5). Then, we have another facet–defining inequality

$$z_2 + z_4 + z_6 \leq 2 + 4(y_1 - 0.75) + (1 - z_7) \tag{38}$$

which is derived from lifting by fixing $z_7 = 1$ and letting $m = 2$. Note that, with these choices, we have a minimal cover inequality

$$z_2 + z_4 + z_6 \leq 2.$$

From Theorem 3.1, we can easily calculate $\rho = 1$ and $\Phi(\pi_1) = \Phi(0.2) = 1$. So we have

$$z_2 + z_4 + z_6 \leq 2 + 4(y_1 - 0.75)$$

which is valid when $z_7 = 1$. By performing lifting procedure on variable z_7, we can derive inequality (38). Indeed, by summing (37) and 0.25×(38), we have a valid inequality $y_1 + y_2 \geq 2 + 0.25z_6$. Because it dominates the inequality $y_1 + y_2 \geq 2$, the latter one clearly is not facet–defining. Instead, the *blending inequality* proposed in this section gives a facet-defining inequality

$$y_1 + y_2 + 0.25z_1 + 0.5z_7 + 0.25(1 - z_9) \geq 2.75 \tag{39}$$

by combining

$$y_1 + 0.25z_1 + 0.25(1 - z_9) \geq 0.75 \tag{40}$$

$$y_2 + 0.5z_7 + 0.25(1 - z_9) \geq 2 \tag{41}$$

where (40) and (41) are facet–defining inequalities for two individual mixing sets respectively, and can be derived by (5).

5 Computational study

In order to investigate the computational advantages of the proposed strong inequalities, we implement them as cutting planes within professional solver CPLEX's branch-and-bound process, i.e., an implementation of of branch-and-cut (B&C) algorithm. In particular, the CALLABLE libraries of CPLEX 12.6.1 are called for adding user defined cuts and a single thread with traditional branch-and-bound search method is adopted with one hour solution time limit. Because it is often the case that a very large number of user defined cuts will be generated, we activate the purging option (which is the default option as well) in CPLEX to allow ineffective ones to be removed by CPLEX. The complete computational study is carried out on the Texas Tech High Performance Computing Center's node based system, where each node contains two Westmere 2.8 GHz 6-core processors with 24 GB main memory [14].

Our numerical study data sets consist of a set of difficult and large instances of the static probabilistic lot-sizing (SPLS) model as described in [32] with d periods, n scenarios and service level $1-\tau$. Let x_t be the decision variable of ordering quantity in period t, I_{it} be the inventory level at the end of period t under scenario i, and w_t be binary variables to indicate order setup. The *natural* deterministic equivalent of SPLS model is

$$
\begin{aligned}
\max \quad & \sum_{t=1}^{d}\sum_{i=1}^{n}\pi_i(c_t x_t + h_{it}I_{it} + g_t w_t) & \\
\text{s.t.} \quad & y_t = \sum_{j=1}^{t} x_j & t\in\{1,\dots,d\} \\
& y_t \geq D_{it}(1-z_i) & t\in\{1,\dots,d\}, i\in\{1,\dots,n\} \\
& I_{it} \geq y_t - D_{it} & t\in\{1,\dots,d\}, i\in\{1,\dots,n\} \\
& 0 \leq x_t \leq M_t w_t & t\in\{1,\dots,d\} \\
& \sum_{i=1}^{n}\pi_i z_i \leq \tau & \\
& I_{i,t} \geq 0, z_i, w_t \in \{0,1\} & t\in\{1,\dots,d\}, i\in\{1,\dots,n\}
\end{aligned}
$$

where D_{it} is the cumulative demand until period t under scenario i, c_t and g_t are the variable and fixed costs of ordering, h_{it} is the variable holding cost in period t under scenario i, and M_t is the order capacity in period t. As mentioned in Section 1, this natural formulation can be easily strengthened by using (3), which has a tighter LP relaxation and less constraints. Hence, the reformulation (3) is applied on those instances to build our testing platform for both CPLEX and B&C algorithms.

We assume that the demands are i.i.d. across periods. In any time period, the demand follows a normal distribution with mean 100 and standard deviation 40, i.e., Normal(100,40), and we set demand to 0 if a negative number is generated. The variable production costs are generated from a uniform distribution between 1 and 5 and inventory holding costs are generated from an uniform distribution between 1 and 2. The fixed costs follow a discrete uniform distribution between 900 and 1000. We set the order capacity $M_t = 500$.

5.1 Computational results of instances with general probabilities

To generate instances with scenarios of general probabilities, we first assign every scenario a random number from a discrete uniform distribution between 1 and 99, and then normalize those numbers by dividing their total summation to obtain probabilities for all scenarios. By varying values of d, n and τ, 40 instances with different sizes are produced. Those random instances are tested by our implementations of the following computing methods. Observing that adding cuts at every node of branch–and–bound tree is rather ineffective, our implementations mainly focus on cut generation at the root node.

- **CPX** indicates the default CPLEX with traditional B&C in single thread mode;
- **Mix** indicates a B&C algorithm by adding *strengthened star inequalities* [2, 18] at every node of branch–and–bound tree;
- **MixR** indicates a B&C algorithm by adding *strengthened star inequalities* [2, 18] at the root node only;
- **TL** indicates a B&C algorithm by adding TL inequalities (using the separation algorithm described in [15]) at every node of the B&C tree;
- **TLR** indicates a B&C algorithm by adding TL inequalities (using the separation algorithm described in [15]) at the root node only;
- **LF** indicates applying the algorithm MixR on an updated formulation with lifting cover inequalities (see Section 3 for implementation details);
- **GMixR** indicates a B&C algorithm by adding new inequalities (5) (using the separation algorithm presented in Section 2) at the root node only.

The detailed computational results of each algorithm are presented in two tables, i.e., Table 4–5, due to space limitation. To evaluate the overall performance of those algorithms and gain a general understanding, a summary is presented in Table 2, where we report the number of instances solved to the optimality (column Solved), the number of unsolved instances (column Unsolved), the average gap before termination among unsolved instances (column Avg. Gap (Unsolved)), and the average number of user defined cuts added before termination but after CPLEX purging (column Avg. Cuts). Given that B&C algorithms have different behaviors and almost all of them (excluding TL) are actually able to solve instances that are solved by CPLEX, we benchmark their performances against CPLEX on those instances to have a fair comparison basis. Hence, in Table 2, we also report algorithms' data on those instances, specifically, the average CPU seconds (column Avg. Time (CPX Solved)), the ratio of the average CPU seconds between CPLEX and a B&C algorithm (column $\frac{\text{CPX Time}}{\text{B\&C Time}}$), the average number of nodes explored (column Avg. Nodes (CPX Solved)), and the ratio of the average number of nodes explored between CPLEX and a B&C algorithm (column $\frac{\text{CPX Nodes}}{\text{B\&C Nodes}}$).

Table 2: Summarized Results of Computing Lot-Sizing Instances with General Probabilities

	Solved	Unsolved	Avg. Gap (Unsolved)	Avg. Cuts	Avg. Time (CPX Solved)	$\frac{\text{CPX Time}}{\text{B\&C Time}}$	Avg. Nodes (CPX Solved)	$\frac{\text{CPX Nodes}}{\text{B\&C Nodes}}$
CPX	17	23	1.66%	0.0	1180.0	1.0	83204.2	1.0
Mix	29	11	2.14%	12792.1	925.8	1.3	4707.8	17.7
MixR	33	7	1.60%	240.6	568.6	2.1	24965.9	3.3
TL	1	39	5.42%	33989.6	NA	NA	NA	NA
TLR	17	23	1.90%	41.6	1510.4	0.8	80846.3	1.0
LF	34	6	1.58%	231.4	441.9	2.7	19878.6	4.2
GMixR	35	5	1.12%	237.7	324.1	3.6	16650.9	5.0

Based on Table 2, a few non-trivial observations can be made. First, as mentioned, adding cuts at every node of the whole branch–and–bound tree is not an effective strategy. As demonstrated in **Mix** and **TL**, an extremely large number of cuts are generated, which drastically increase the complexity of SPLS formulation. Such situation is especially severe in **TL**. Note that SPLS only has $d+1$ non-sparse constraints, i.e., the first set of constraints and the 0–1 knapsack. Nevertheless, for our instances with $d = 30$ or 40, the average number of generated TL inequalities, which are non-sparse, is almost thirty–four thousands. Comparing to the performances of their two corresponding variants, i.e., **MixR** and **TLR**, we can clearly see the benefits of adding cuts in a less frequent fashion.

Second, among all implemented computing methods, our **GMixR** and **LF** provide the most significant computational improvements and perform much better than CPLEX. Note that only 17 of 40 instances can be solved to optimality by CPLEX, while **LF** and **GMixR** can solve 34 and 35 instances, respectively. In addition, **GMixR** and **LF**, on instances solved by CPLEX, have the best computation speed (are faster by 2.7 and 3.6 times, respectively) with the smaller numbers of branch-and-bound nodes (roughly 23.8% and 20% of CPLEX nodes, respectively). Also, for those that cannot be solved to optimality, the average gaps before termination in **GMixR** and **LF** are the least ones. For the efficacy of **GMixR**, we believe that it is mainly due to the fact that the separation algorithm based on Corollary 2.2.1 leads to facet–defining cuts for the polyhedron of CCP. As the *strengthened star inequalities* are also facet–defining cuts for the polyhedron of CCP, results of **GMixR** and **MixR** show that they are all practically useful and the former has more computationally advantages. For **LF**, we think the lifted inequalities largely capture the non-trivial interactions between mixing set and the 0–1 knapsack. Their strong performance suggests that a more sophisticated separation algorithm is worth further exploring.

Third, although TL inequalities are theoretically valid and strong, they are less effective than the *strengthened star inequalities* in computation. One explanation is that, as shown in Corollary 2.1.1 and Example 1, TL inequalities are usually weak for CCP instances with general probabilities. Another interesting observation is that **TL** has much more user defined cuts than **Mix**, while the root node implementation variant **TLR** has much less cuts than the corresponding **MixR**. One possibility is that we allow CPLEX to purge user defined cuts if they are deemed weak, CPLEX may decide to purge a high proportion of TL cuts and start branching very soon.

5.2 Computational results of instances with equal probabilities

In Section 4, we develop a new technique to blend strong inequalities (5) derived from individual mixing sets into a strong one for CCP. In particular, we show that blending inequality could be facet–defining when scenarios have equal probabilities. In order to computationally evaluate this new type of inequalities, we generate 40 instances with equal probabilities (by simply setting scenario probabilities as $\frac{1}{n}$) and compute them by the following computing methods.

- **CPX** indicates the default CPLEX with transitional branch–and–bound in single thread mode;
- **Mix** indicates a branch–and–cut algorithm by adding *strengthened star inequalities* at every node of branch–and–bound tree;
- **Mix**100indicates a branch–and–cut algorithm by adding *strengthened star inequalities* at every 100 nodes of branch–and–bound tree;

- **BL** indicates a branch–and–cut algorithm by adding *blending inequalities* at the root node and *strengthened star inequalities* at every 100 nodes of the branch–and–bound tree.

The **Mix**100 is implemented due to the observation made from Table 2 that adding cuts at every node of the whole branch-and-bound tree is not effective. With some preliminary tests, we find that adding *strengthened star inequalities* at every 100 nodes of branch–and–bound tree gives the best result in general. Then, on top of the **Mix**100, we develop **BL** algorithm by adding blending inequalities at the root node. Similar to the organization of Section 5.1, the detailed computational results are presented in Table 6 and a summary is presented in Table 3.

Table 3: Summarized Results of Computing Lot-Sizing Instances with Equal Probabilities

	Solved	Unsolved	Avg. Gap (Unsolved)	Avg. Cuts	Avg. Time (CPX Solved)	CPX Time / B&C Time	Avg. Nodes (CPX Solved)	CPX Nodes / B&C Nodes
CPX	20	20	1.98%	0.0	1727.5	1.0	103145.2	1
Mix	29	11	2.84%	12359.4	921.7	1.8	5434.6	18.9
Mix100	32	8	1.72%	2242.3	531.7	3.2	8445.2	12.2
BL	39	1	0.60%	1946.2	334.9	5.2	6530.8	15.7

Based on the performances of CPLEX and **Mix** in Table 2 and Table 3, we first note that the instances with equal probabilities are not necessarily easier than the instances with general probabilities. It is different from the observation made in [18], but reasonable since $p \geq \nu$ (or $p \geq \nu_r \ \forall r \in [1, d]$) and hence the instances with equal probabilities usually have much larger problem size in the reformulation (3).

Regarding the effectiveness of B&C methods, we observe that **BL** completely dominates other algorithms by optimally computing almost all instances (39 out of 40 instances) and demonstrating much fast computational speed (as 5.2×CPLEX). We believe that such strong performance shows that the generated blending cuts, which capture the interactive information among mixing sets and are facet–defining for CCP, have a great computational advantage over existing strong inequalities derived from single mixing sets. The ratio of the average number of nodes between CPLEX and **BL** supports our understanding. Note that **BL** generally has 13% less user defined cuts than **Mix100** while it explores 23% less nodes in the branch-and-bound trees than **Mix100**.

6 Conclusion and future research

In this paper, we study the polyhedral structure of chance–constrained program with stochastic right–hand side, which is computationally very challenging. We develop three families of strong inequalities from different perspectives. Following the tradition that considers a single mixing set with a $0-1$ knapsack, our first family of inequalities of that set dominates or subsumes all explicit inequalities described in [1, 15, 18]. Our second family of inequalities builds a direct link between a single mixing set and $0-1$ knapsack through lifting and superadditive lifting with respect to cover inequality. In order to analyzing the interactions among multiple mixing sets, we design a novel technique to integrate facet–defining inequalities derived from individual mixing sets, which leads to our third family of inequalities, i.e., *blending inequalities*. Finally, we implement all three families of strong inequalities and test them on random instances of static probabilistic lot–sizing problem. Through benchmarking with a professional MIP solver and existing cutting plane methods, we observe significant computational improvements can be achieved by all those proposed inequalities, especially by the newly developed *blending inequalities*.

There are three directions, we believe, that deserve further studies. The first one is to develop more general and effective separation algorithms. Note that cutting planes generated in our current numerical study is just a small proportion of our first family of strong inequalities. So, more powerful separation algorithms will allow us to make good use of that large family of strong inequalities. Another one is to develop stronger superadditive lifting functions and better lifting techniques. For example, our current superadditive approximation scheme is adopted from existing research, which probably is not the best one. A deeper study on the lifting function should help us develop tighter valid approximations for stronger cutting planes. Finally, given the outstanding computational performance of *blending inequalities*, this technique should be fully investigated and we will extend it for other similar models.

References

[1] Abdi, A., Fukasawa, R.: On the mixing set with a knapsack constraint. Math. Program. **Online First Articles.** 2016.

[2] Atamtürk, A., Nemhauser, G.L., Savelsbergh, M.W.P.: The mixed vertex packing problem. Math. Program. 89(1), 35–53. 2000.

[3] Balas, E.: Facets of the knapsack polytope. Math. Program. 8(1), 146–164. 1975.

[4] Balas, E., Ceria, S., Cornuéjols, G.: A lift-and-project cutting plane algorithm for mixed 0–1 programs. Math. Program. 58(1), 295–324. 1993.

[5] Beraldi, P., Bruni, M.E., Conforti, D.: Designing robust emergency medical service via stochastic programming. Eur. J. Oper. Res. 158(1), 183–193. 2004.

[6] Beraldi, P., Ruszczyński, A.: A branch and bound method for stochastic integer programs under probabilistic constraints. Optim. Methods & Softw. 17(3), 359–382. 2002.

[7] Beraldi, P., Ruszczyński, A.: The probabilistic set covering problem. Oper. Res. 50(6), 956–967. 2002.

[8] Dentcheva, D., Prékopa, A., Ruszczyński, A.: Concavity and efficient points of discrete distributions in probabilistic programming. Math. Program. 89(1), 55–77. 2000.

[9] Gu, Z., Nemhauser, G.L., Savelsbergh, M.W.: Lifted cover inequalities for 0–1 integer programs: Computation. INFORMS J. on Comp. 10(4), 427–437. 1998.

[10] Gu, Z., Nemhauser, G.L., Savelsbergh, M.W.: Lifted cover inequalities for 0–1 integer programs: Complexity. INFORMS J. on Comp. 11(1), 117–123. 1999.

[11] Gu, Z., Nemhauser, G.L., Savelsbergh, M.W.: Sequence independent lifting in mixed integer programming. J. of Combin. Optim. 4(1), 109–129. 2000.

[12] Guan, Y., Ahmed, S., Nemhauser, G.L.: Sequential pairing of mixed integer inequalities. Discrete Optim. 4(1), 21-39. 2007.

[13] Günlük, O., Pochet, Y.: . Math. Program. 90(3), 429-457. 2001.

[14] High Performance Computing Center. http://www.hpcc.ttu.edu/. Accessed April, 2016.

[15] Küçükyavuz, S.: On mixing sets arising in chance–constrained programming. Math. Program. 132(1), 31–56. 2012.

[16] Liu, X., Küçükyavuz, S., Luedtke, J.: Decomposition algorithms for two–stage chance–constrained programs. Math. Program., 1–25. 2014.

[17] Luedtke, J.: A branch-and-cut decomposition algorithm for solving chance–constrained mathematical programs with finite support. Math. Program. 146(1), 219–244. 2014.

[18] Luedtke, J., Ahmed, S., Nemhauser, G.: An integer programming approach for linear programs with probabilistic constraints. Math. Program. 122(2), 247–272. 2010.

[19] Prékopa, A.: Dual method for the solution of a one-stage stochastic programming problem with random RHS obeying a discrete probability distribution. Zeitschrift für Oper. Res. 34(6), 441–461. 1990.

[20] Miller, B.L., Wagner, H.M.: Chance constrained programming with joint constraints. Oper. Res. 13(6), 930-965. 1965.

[21] Miller, A.J., Wolsey, L.A.: Tight formulations for some simple mixed integer programs and convex objective integer programs. Math. Program. 98(1), 73-88. 2003.

[22] Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. Wiley, New York. 1988.

[23] Qiu, F., Shabbir, A., Dey S.D., Wolsey, L.A.: Covering linear programming with violations. INFORMS J. on Comp. 26(3), 531–546. 2014.

[24] Saxena, A., Goyal, V., Lejeune, M.A.: MIP reformulations of the probabilistic set covering problem. Math. Program. 121(1), 1–31. 2010.

[25] Sen, S.: Relaxations for probabilistically constrained programs with discrete random variables. Oper. Res. Lett. 11(2), 81–86. 1992.

[26] Tanner, N.W., Ntaimo, L.: IIS branch–and–cut for joint chance–constrained stochastic programs and application to optimal vaccine allocation. Eur. J. Oper. Res. 207(1), 290-296. 2010.

[27] Van Vyve, M.: The continuous mixing polyhedron. Math. Oper. Res. 30(2), 441–452. 2005.

[28] Wang, J., Shen, S.: Risk and energy consumption tradeoffs in cloud computing service via stochastic optimization models. In: Proceedings of the 5th IEEE/ACM International Conference on Utility and Cloud Computing (UCC 2012). Chicago, Illinois. 2012.

[29] Wolsey, L.A.: Faces for a linear inequality in 0–1 Variables. Math. Program. 8(1), 165-178. 1975.

[30] Zemel, E.: Easily computable facets of the knapsack polytope. Math. O.R. 14(4), 760-764. 1989.

[31] Zeng, B., An, Y., Kuznia, L.: Chance constrained mixed integer program: Bilinear and linear formulations, and Benders decomposition. **Optimization Online.** 2014.

[32] Zhang, M., Küçükyavuz, S., Goel, S.: A branch-and-cut method for dynamic decision making under joint chance constraints. Manage. Sci. 60(5), 1317–1333. 2014.

[33] Zhao, M., de Farias Jr, I.R.: The mixing-MIR set with divisible capacities. Math. Program. 115(1), 73-103. 2008.

[34] Zhao, M., de Farias Jr, I.R.: A note on the continuous mixing set. Oper. Res. Letters 36(6), 726–733. 2008.

A Computational Tables for Instances with General Probabilities

The column $d \times n \times \tau$ indicates that the instance has d periods and n scenarios with service level $1 - \tau$. For a given combination of d, n, τ, we solve 5 instances. In all tables, we compare the percentage of root node gap (column **%RootGap**), the number of branch–and–cut tree nodes explored (column **Nodes**), and CPU time in second on solving the instance to the optimality (column **Time (%Endgap)**). We indicate the instance that could not be solved within one hour with T(*gap*), where *gap*, in parenthesis, is the percentage gap between the best lower bound and the best integer solution found in the search tree when the time limit is reached. We also report the number of user cuts added (column **Cuts**) after purging. Due to the page size restriction, we partition the results into two tables as in Table 4 and 5.

Table 4: Static probabilistic lot–sizing problem with general probabilities (cuts on every node)

$d \times n \times \tau$	%RootGap			Nodes			Time (%Endgap)			Cuts	
	CPX	**Mix**	**TL**	**CPX**	**Mix**	**TL**	**CPX**	**Mix**	**TL**	**Mix**	**TL**
$30 \times 300 \times 0.2$	12.89	12.97	11.90	42667	1472	1885	493	323	T(5.87)	9566	44235
	12.64	12.91	11.67	42295	2635	13914	573	619	T(4.41)	14758	40042
	13.52	14.79	15.58	129237	6537	8803	2446	1253	T(5.42)	13104	40881
	13.28	10.67	13.74	108929	3265	5103	1433	915	T(5.56)	15950	40929
	10.06	13.30	13.26	20672	1295	6153	226	233	T(3.44)	7926	39937
$30 \times 300 \times 0.25$	16.58	16.52	16.61	119828	4494	27855	2022	1663	T(4.26)	18743	29973
	17.60	15.33	16.72	165600	5101	6022	T(0.47)	2027	T(6.5)	22006	28915
	12.74	16.04	14.70	144137	4788	5261	2253	2880	T(5.82)	28180	36505
	12.21	13.65	11.52	82373	7554	2808	930	1385	T(6.23)	12317	41126
	14.06	13.79	13.41	63613	3040	1899	1191	1129	T(6.85)	15587	42442
$30 \times 400 \times 0.2$	12.47	12.58	11.06	116158	3404	3589	1765	928	T(4.59)	13472	29113
	20.90	17.32	15.06	164404	3246	2371	T(0.66)	1343	T(8.44)	16116	43727
	14.74	13.27	13.30	213110	3197	11140	T(0.85)	1326	T(5.81)	18949	35557
	10.15	9.33	11.02	164548	1744	2327	1831	634	T(5.26)	12995	38621
	12.02	13.18	14.24	100968	2584	1987	1345	1097	T(6.74)	19420	37002
$30 \times 400 \times 0.25$	13.77	13.51	13.66	136077	1375	1887	T(0.57)	852	T(5.9)	12759	39507
	11.80	14.03	14.85	91600	3050	1337	T(0.61)	T(2.39)	T(7.5)	27890	36893
	11.43	12.03	11.79	117300	2588	1003	T(1.41)	T(2.31)	T(6.19)	26575	31546
	15.73	21.15	16.12	82502	3379	2985	T(3.08)	2153	T(8.06)	17688	42711
	19.55	19.11	16.42	103810	2846	2000	T(2.51)	T(2.15)	T(8.83)	27901	39520
$40 \times 400 \times 0.1$	17.66	13.96	9.33	81267	6172	23300	1206	507	T(2.7)	7591	31354
	9.74	10.71	8.56	36124	3625	45453	472	267	T(1.8)	5988	28524
	12.17	11.92	11.09	81150	10969	57600	855	794	T(0.97)	8106	16858
	10.40	13.70	13.09	40383	11830	23322	501	837	T(2.11)	7169	34188
	12.49	10.52	14.96	40123	4625	23565	518	275	962	6594	16093
$40 \times 400 \times 0.15$	14.68	16.66	11.77	87227	8766	21167	T(1.76)	1474	T(3.68)	12091	26988
	16.19	19.15	13.87	131435	8222	44299	T(1.83)	1584	T(3.63)	14904	24704
	12.35	13.51	11.50	118690	17492	21604	T(1.23)	2761	T(4.67)	17575	32079
	14.83	17.15	15.56	100350	13919	9946	T(1.42)	2184	T(5.46)	14086	27822
	12.49	21.96	14.87	93400	17402	10298	T(2.26)	T(1.57)	T(6.38)	16683	38790
$40 \times 500 \times 0.1$	15.33	11.31	14.86	150900	19504	11399	T(0.34)	T(0.86)	T(4.2)	16296	31822
	12.64	9.91	12.76	115200	15580	10953	T(1.41)	T(1.29)	T(5.7)	22261	39487
	13.25	16.15	13.31	137300	8474	21125	T(0.74)	1531	T(3.59)	15160	32950
	13.07	13.02	10.23	157900	14466	10593	T(1.96)	T(1.28)	T(4.71)	20375	32937
	19.03	14.00	12.18	153900	29846	25474	T(1.06)	3510	T(3.67)	14761	29732
$40 \times 500 \times 0.15$	18.00	18.14	19.68	84737	5080	2884	T(2.75)	T(2.38)	T(6.86)	20350	38996
	16.94	14.32	13.41	82100	5427	2950	T(2.43)	T(2.29)	T(6.78)	23158	33970
	17.29	14.91	12.49	90885	6665	6490	T(2.75)	T(2.08)	T(6.85)	22109	36647
	11.97	15.87	15.12	76884	6575	2900	T(4.45)	T(4.98)	T(8.73)	23662	37875
	11.57	11.93	14.56	114678	12517	2434	T(1.58)	3570	T(7.15)	17821	37627

Table 5: Static probabilistic lot–sizing problem with general probabilities (cuts on root node)

$d \times n \times \tau$	%RootGap					Nodes					Time (%Endgap)					Cuts			
	CPX	**MixR**	**TLR**	**LF**	**GMixR**	**CPX**	**MixR**	**TLR**	**LF**	**GMixR**	**CPX**	**MixR**	**TLR**	**LF**	**GMixR**	**MixR**	**TLR**	**LF**	**GMixR**
$30 \times 300 \times 0.2$	12.89	12.97	11.9	14.33	11.19	42667	14036	33440	13404	7960	493	224	696	203	119	209	101	213	212
	12.64	12.91	11.67	12.37	15.39	42295	18938	58888	15575	14206	573	389	899	299	241	250	25	176	198
	13.52	14.79	15.58	12.92	14.03	129237	30655	110865	21694	18130	2446	708	1843	459	374	273	47	299	269
	13.28	10.67	13.74	10.98	15.07	108929	22701	81320	23294	21661	1433	584	1997	494	340	274	29	256	265
	10.06	13.3	13.26	10.7	12.55	20672	9149	22667	8700	7645	226	172	311	171	94	172	36	186	158
$30 \times 300 \times 0.25$	16.58	16.52	16.61	16.04	15.75	119828	35109	136123	19645	21179	2022	811	2759	596	473	303	2	264	317
	17.6	15.33	16.72	15.99	16.86	165600	37590	92900	30974	35583	T(0.47)	1532	T(0.61)	1135	850	329	94	297	328
	12.74	16.04	18.54	13.12	11.73	144137	25706	108042	26458	18875	2253	759	2010	710	492	346	87	342	309
	12.21	13.65	11.52	11.89	13.87	82373	39008	86022	25600	19794	930	1085	1569	629	392	224	43	232	281
	14.06	13.79	13.41	14.87	12.4	63613	18399	108476	12197	11501	1191	540	2230	523	343	262	63	266	296
$30 \times 400 \times 0.2$	12.47	12.58	11.06	12.21	12.19	116158	34109	139224	33693	26111	1765	628	2092	513	360	298	107	273	299
	20.9	17.32	15.06	23.61	14.46	164404	30017	85300	27841	21779	T(0.66)	984	T(1.23)	768	679	391	36	369	428
	14.74	13.27	13.3	11.95	14.14	213110	75576	202768	37938	36273	T(0.85)	1807	T(0.57)	1325	946	379	27	415	505
	10.15	9.33	11.02	9.4	9.01	164548	33891	100680	21500	21447	1831	584	1622	347	337	249	33	253	229
	12.02	13.18	14.24	16.43	15.39	100968	23354	139159	23597	16037	1345	739	3249	633	519	362	91	311	358
$30 \times 400 \times 0.25$	13.77	13.51	14.27	13.56	15.27	136077	51254	95178	15051	29366	T(0.57)	1876	T(1.41)	557	1016	425	46	398	433
	11.8	14.03	14.38	14.39	15.51	91600	37872	61381	34696	28585	T(0.61)	2447	T(2.56)	1879	1517	349	136	405	442
	11.43	12.03	11.79	13.04	11.81	117300	70938	88109	60147	40863	T(1.41)	2258	T(1.41)	1937	945	352	152	350	422
	15.73	21.15	20.17	11.64	19.2	82502	44552	109623	35863	31453	T(3.08)	2752	T(1.78)	2052	1361	447	40	444	514
	19.55	19.11	21.81	14.38	14.77	103810	42420	83890	34127	52048	T(2.51)	3201	T(1.9)	1665	2212	508	70	483	541
$40 \times 400 \times 0.1$	17.66	13.96	9.33	10.75	9.91	81267	21202	43391	27634	18298	1206	523	1029	863	346	200	3	221	227
	9.74	10.71	8.56	9.22	7.88	36124	13419	46115	10341	10713	472	310	837	201	175	198	7	193	194
	12.17	11.92	11.09	10.75	10.72	81150	33767	101033	25414	14718	855	580	1591	415	235	141	9	152	143
	10.4	13.7	13.09	9.89	20.85	40383	32891	35208	15660	20757	501	700	599	255	465	223	15	197	191
	12.49	10.52	14.96	12.31	13.83	40123	18086	23734	13530	14034	518	330	344	202	205	107	9	99	95
$40 \times 400 \times 0.15$	14.68	16.66	11.77	17.28	16.01	87227	42816	81100	52020	32366	T(1.76)	1628	T(1.45)	1691	947	266	49	291	314
	16.19	19.15	13.87	13.32	13.08	131435	44291	100800	37365	28983	T(1.83)	1668	T(1.95)	1335	1153	372	10	339	324
	12.35	13.51	11.5	11.27	13.67	118690	57456	108000	27582	43054	T(1.23)	2331	T(1.66)	1162	1595	336	7	327	322
	14.83	17.15	14.83	16.27	16.12	100350	35132	72951	37760	36241	T(1.42)	1409	T(1.51)	2116	1816	307	85	309	379
	12.49	21.96	14.87	12.94	15.21	93400	57887	86360	56502	68337	T(2.26)	T(1.03)	T(2.75)	T(1.11)	T(0.68)	429	20	382	485
$40 \times 500 \times 0.1$	15.33	11.31	14.86	12.42	12.71	150900	87120	75095	75619	66276	T(0.34)	2152	T(1.78)	1848	1404	208	36	201	217
	12.64	9.91	12.76	14.09	16.5	115200	60753	96827	79870	48108	T(1.41)	3190	T(1.75)	3418	1670	280	24	283	282
	13.25	16.15	13.31	17.42	12.58	137300	30928	90777	25329	18469	T(0.74)	1372	T(0.81)	931	675	312	25	306	312
	13.07	13.02	10.23	13.89	13.43	157900	99400	94981	57914	63710	T(1.96)	T(0.3)	T(1.22)	1713	1708	308	38	318	307
	19.03	14	12.18	12.55	11.88	153900	73357	104783	83749	66947	T(1.06)	3438	T(1.61)	2115	1793	251	16	224	255
$40 \times 500 \times 0.15$	18	18.14	19.68	15.6	20.31	84737	37963	54111	33575	43560	T(2.75)	T(2.04)	T(2.76)	T(2.03)	T(0.46)	313	72	337	355
	16.94	14.32	13.41	11.58	13.77	82100	37417	50971	38153	42195	T(2.43)	T(1.37)	T(2.75)	T(0.99)	T(0.92)	369	79	321	407
	17.29	14.91	12.39	11.95	15.29	90885	42507	56673	43859	37639	T(2.75)	T(2.17)	T(2.87)	T(0.99)	T(1.33)	460	57	461	514
	11.97	15.87	15.12	17.78	17.67	76884	39392	57300	41900	47504	T(4.45)	T(4.19)	T(4.44)	T(3.25)	T(2.24)	386	57	391	467
	11.57	11.93	14.56	14.95	12.46	114678	63772	74900	55924	83968	T(1.58)	T(0.11)	T(2.86)	T(1.13)	2982	383	58	411	340

B Computational Tables for Instances with Equal Probabilities

We keep the same table format as in Appendix A. For a given combination of d, n, τ, we solve 5 instances.

Table 6: Static probabilistic lot–sizing problem with equal probabilities

$d \times n \times \tau$	%RootGap				Nodes				Time (%Endgap)				Cuts		
	CPX	**Mix**	**Mix100**	**BL**	**CPX**	**Mix**	**Mix100**	**BL**	**CPX**	**Mix**	**Mix100**	**BL**	**Mix**	**Mix100**	**BL**
$30 \times 300 \times 0.2$	12.34	11.51	11.51	9.31	33621	695	2452	1325	362	87	86	58	4619	1125	770
	12.56	12.06	12.06	11.99	65932	2402	4255	3172	827	506	230	122	10080	1944	1143
	17.89	17.5	17.5	25.35	77859	2101	5990	4101	871	679	475	221	13875	2212	1853
	14.09	18.03	18.03	18.2	81532	4023	4998	3666	1012	977	315	203	14790	2425	1368
	13.05	14.1	14.1	13.39	66282	3917	3998	3611	1352	976	253	183	15927	1871	1927
$30 \times 300 \times 0.25$	15.09	13.5	13.5	12.27	118859	3656	8230	5549	T(1.51)	1527	849	494	17157	3519	2406
	15.76	16.12	16.12	13.16	117627	2294	4242	3275	1498	1222	409	291	17088	2353	2454
	17.86	17.3	17.3	17.04	89386	2695	4986	2194	1402	1232	423	216	18575	2694	2209
	15.52	16.1	16.1	14.73	76442	2368	4815	3819	965	1111	552	325	17440	2199	1922
	13.84	15.12	15.12	24.63	148971	2676	3820	3782	3048	1320	360	306	16643	2617	2934
$30 \times 400 \times 0.2$	13.98	14.06	14.06	12.41	182865	3364	5507	4768	T(0.16)	1928	723	491	21185	3897	2689
	16.1	15.09	15.09	16.61	218740	7589	23316	10360	T(2.06)	T(1.35)	3591	1423	23274	7014	4759
	12.43	14.22	14.22	18.49	108940	5447	5969	4062	2259	2718	780	472	24391	3657	2997
	13.74	21.5	21.5	17.33	176128	4237	6513	4045	2629	T(1.54)	920	460	27310	4085	3195
	15.63	15.64	15.64	14.07	137539	5980	10183	5688	T(0.39)	2820	1069	752	17202	3377	3179
$30 \times 400 \times 0.25$	18.09	17.99	17.99	15.39	109445	1949	4874	4263	T(1.93)	T(3.73)	1384	818	25542	4631	3393
	22.75	21.64	21.64	16.83	119700	1419	7371	7158	T(1.9)	T(4.73)	T(1.6)	2195	31299	9066	6467
	18.72	16.69	16.69	20.68	61919	1553	10228	6130	T(2.67)	T(4.88)	T(1.1)	1907	31162	7961	4884
	13.42	13.43	13.43	19.66	140255	3757	9424	5700	T(0.6)	T(0.26)	1546	1005	22064	4784	5163
	18.77	14.27	14.27	20.49	122025	2855	13764	14914	T(3.16)	T(5.43)	T(1.82)	3492	26825	8435	6270
$40 \times 400 \times 0.1$	14.49	12.98	12.98	14.05	63612	6857	10716	7443	791	616	561	349	9113	2412	1862
	13.68	9.25	9.25	8.94	139619	10274	10846	9650	2266	589	464	321	7054	1405	1653
	13.1	13.21	13.21	11.72	242727	18461	30759	27243	2626	1302	1216	789	9255	2267	1917
	8.56	11.8	11.8	11.09	66765	7759	8434	6500	1072	432	325	185	6753	1379	1149
	11.97	9.41	9.41	11.11	53415	7130	10553	8610	735	593	455	297	9973	1724	1817
$40 \times 400 \times 0.15$	11.22	9.06	9.06	11.55	88000	8165	14667	11080	T(1.93)	2089	1646	1166	16865	4272	3569
	13.06	12.57	12.57	12.57	146900	9410	11368	10312	T(0.88)	1199	1150	919	8988	2664	1908
	10.01	10.94	10.94	11.42	75643	4491	6838	4542	T(0.7)	760	724	429	10359	2169	1892
	16.42	14.05	14.05	14.08	98085	9604	14618	9392	T(1.67)	2552	2062	1023	17229	4401	3430
	13.27	15.17	15.17	12.59	137367	3023	8870	5542	3571	401	790	365	6583	2299	1337
$40 \times 500 \times 0.1$	10.84	11.92	11.92	9.94	68922	4309	8625	6944	1241	435	572	441	8271	1956	2235
	12.1	16.97	16.97	16.18	123100	25255	33523	25734	T(2.46)	3548	3123	1837	15494	4916	4236
	17.32	16.39	16.39	14.21	103300	18722	21244	14675	T(1.5)	2128	1464	889	12094	3456	3052
	11.97	12.05	12.05	13.38	162388	5486	10629	7743	3516	992	760	458	12954	2715	2399
	11.1	15.58	15.58	13.93	158352	11341	15501	11404	3408	1324	1076	762	11444	3349	3031
$40 \times 500 \times 0.15$	13.7	11.13	11.13	13.53	59100	4686	15395	17573	T(3.3)	T(2.51)	T(0.77)	3370	26263	6646	5161
	11.01	15.68	15.68	13.6	85159	8200	15618	14867	T(2.16)	3380	T(1.49)	3256	20233	6277	6142
	14.78	13.54	13.54	14.05	86000	4148	11774	12999	T(4.26)	T(1.81)	T(2.71)	3322	23851	8537	6815
	14.84	12.68	12.68	14.47	61500	4830	13565	13311	T(2.85)	T(2.38)	T(2.5)	2212	25767	8242	6360
	16.12	14.41	14.41	14.28	74055	4759	14081	15076	T(3.57)	T(2.65)	T(1.83)	T(0.6)	25001	7574	7107