On Solving General Two-Stage Stochastic Programs

Manish Bansal and Sanjay Mehrotra
Department of Industrial Engineering and Management Sciences, Northwestern University
Emails: {manish.bansal; mehrotra}@northwestern.edu

October 15, 2015

Abstract. We study general two-stage stochastic programs and present conditions under which the second stage programs can be convexified. This allows us to relax the restrictions, such as integrality, binary, semi-continuity, and many others, on the second stage variables in certain situations. Next, we introduce two-stage stochastic disjunctive programs (TSS-DPs) and extend Balas’s linear programming equivalent for deterministic disjunctive programs to TSS-DPs. In addition, we develop a finitely convergent algorithm, which utilizes the sequential convexification approach of Balas within L-shaped method, to solve various classes of TSS-DPs. We formulate a semi-continuous program (SCP) as a DP and use our results for TSS-DPs to solve two-stage stochastic SCPs (TSS-SCPs). In particular, we provide linear programming equivalent for the second stage of the TSS-SCPs and showcase how our convexification approach can be utilized to solve a TSS-SCP having semi-continuous inflow set in the second stage.

1. Introduction

In this paper, we consider the following general two-stage stochastic program (TSSP):

\[
\begin{align*}
\min & \quad c^T x + E_\omega[Q(\omega, x)] \\
\text{s.t.} & \quad Ax \geq b \\
& \quad x \in \mathcal{X},
\end{align*}
\]

where \( \omega \) is a random vector with finite support \( \Omega \) and for any scenario \( \omega \) of \( \Omega \),

\[
Q(\omega, x) := \min \ g(\omega)^T y(\omega) \\
\text{s.t.} & \quad W(\omega)y(\omega) \geq r(\omega) - T(\omega)x \\
& \quad y(\omega) \in \mathcal{Y}(\omega, x).
\]

Here, \( c \in \mathbb{R}^p \), \( A \in \mathbb{R}^{m_1 \times p} \), \( b \in \mathbb{R}^{m_1} \), and for each \( \omega \in \Omega \), \( g(\omega) \in \mathbb{R}^q \), recourse matrix \( W(\omega) \in \mathbb{R}^{m_2 \times q} \), technology matrix \( T(\omega) \in \mathbb{R}^{m_2 \times p} \), and \( r(\omega) \in \mathbb{R}^{m_2} \). Let \( \Pi(x, y) := c^T x + \sum_{\omega \in \Omega} g(\omega)^T y(\omega) \).

The formulation defined by (2)-(4) and the function \( Q(\omega, x) \) are referred to as the second-stage subproblem and recourse function, respectively. We assume that

(i) \( T(\omega) \in \mathbb{Z}^{m_2 \times p} \),
(ii) \( X := \{ x : Ax \geq b, x \in \mathcal{X} \} \) is non-empty,
(iii) \( \mathcal{K}(\omega, x) := \{ y(\omega) : (3)-(4) \text{ hold} \} \) is non-empty for all \( (\omega, x) \in \Omega \times X \),
(iv) \( Q(\omega, x) < \infty \) for all \( (\omega, x) \in \Omega \times X \) (relatively complete recourse).

We consider general sets \( \mathcal{X} \subseteq \mathbb{R}^p \) and \( \mathcal{Y}(\omega, x) \subseteq \mathbb{R}^q \), thereby subsuming restrictions such as integrality, semi-continuity, binary, etc. on the variables. In this paper, we provide conditions under
which these restrictions on the second stage non-continuous variables of TSSPs (1) can be relaxed. This means that we can obtain linear programming equivalent for the second stage programs of the TSSP in certain conditions. Readers are referred to [14] for general introduction on stochastic programming. Note that the TSSPs with \( X := \mathbb{Z}^{p_1} \times \mathbb{R}^{p-p_1} \) and \( Y(\omega, x) := \mathbb{Z}^{q_1} \times \mathbb{R}^{q-q_1} \) are referred to as the two-stage stochastic mixed integer programs (TSS-MIPs). In the literature on TSS-MIPs, algorithms have been developed for TSS-MIPs with second stage having pure integer variables [1, 25, 31, 40], or binary and continuous variables [15, 21, 26, 28, 29, 33, 37], or mixed integer variables [33, 34, 38]. For survey on TSS-MIPs, we refer the reader to [27, 30, 32].

One of the major challenges in solving the general TSSPs is to optimize NP-hard second-stage problems for a given first stage decision and a particular realization (scenario) of uncertain parameters. Therefore, it is important to evaluate the conditions under which the second stage programs can be convexified. This will allow us to relax the restrictions, if any, on the second stage non-continuous variables and obtain linear programming equivalent for the second stage program. In this direction, Bansal et al. [9] provide conditions under which the second stage formulation of general two-stage MIPs in Benders’ form or TSS-MIPs with \( |\Omega| = 1 \) (single scenario) can be convexified. They also present conditions under which the convexification of the second stage of a TSS-MIP can be used to relax integrality constraints of the corresponding second stage integer variables in the extensive formulation of the problem. Bansal et al. [9] extends the research of Kim and Mehrotra [24] in which they consider an integrated staffing and scheduling problem (ISSP) under demand uncertainty as a TSS-MIP with pure integer variables in the first stage and convexify the second stage mixed integer program by adding (a priori) parametric mixed integer rounding inequalities. Both Kim and Mehrotra [24] and Bansal et al. [9] show that the convexification approach is computationally very effective in solving the ISSP and four variants of two-stage capacitated lot-sizing problem, respectively, which are intractable otherwise. In this paper, we further generalize the results of Bansal et al. [9] for a general two-stage mathematical optimization program (in Bender’s form) and TSSPs. More specifically, we show that under suitable conditions, the general second stage programs can be convexified, thereby obtaining linear programming equivalent for the second stage programs.

We also introduce two-stage stochastic disjunctive program (TSS-DP) with disjunctive constraints in both first and the second stage of TSSP. It is important to note that the class of TSS-DPs subsumes TSS-MIPs, two-stage stochastic semi-continuous programs (TSS-SCPs), and many classes of TSSPs where the first (or/and second) stage is a non-convex program (such as general quadratic programs, separable non-linear programs, etc.). As per our knowledge, the general TSS-DP has not been studied before. Disjunctive programming (DP) is a well-known area in the field of optimization where a linear programming problem has disjunctive constraints, i.e. linear constraints with the “or” (\( \lor \), disjunctive) operations. More specifically, DP is optimization over a union \( \mathcal{R} \) of polyhedra \( \mathcal{R}_i = \{ z \in \mathbb{R}^n : E_i^t z \geq f_i \} \), denoted by \( \mathcal{R} := \bigcup_i \mathcal{R}_i = \{ z \in \mathbb{R}^n : \lor_i (E_i^t z \geq f_i) \} \). Refer to Chapter 10 of [23] and see Section 3 for details. Balas [3, 4] provides a tight extended formulation for \( \mathcal{R} \) and the convex hull description of \( \mathcal{R} \) in the original space. In this paper, we amalgamate these results with our convexification approach for TSSPs and provide conditions to get a linear programming equivalent for the second stage of TSS-DPs, i.e TSSPs with \( X := \{ x \in \mathbb{R}^p : \lor_{s \in S} (D_s x \geq d_s) \} \) and \( Y(\omega, x) := \{ y(\omega) \in \mathbb{R}^q : \lor_{h \in H} (D_h^1(\omega)y(\omega) \geq d_h^0(\omega) - D_h^2(\omega)x) \} \) for all \( (\omega, x) \in (\Omega, X) \). Similar to the Benders’ decomposition [13] and L-shaped method [39], we present a decomposition algorithm utilizing our convexification approach to solve TSS-DPs.

Balas [3, 4] exhibits a property according to which the convex hull of a set of points satisfying multiple disjunctive constraints, where each disjunction contains exactly one inequality, can be de-
rived by sequentially generating the convex hull of points satisfying only one disjunctive constraint. This property is referred to as the \textit{sequential convexification}; see Section 3 for details. A subclass of DPs for which the sequential convexification property holds is called sequentially convexifiable DPs. In \cite{3, 4}, Balas shows that the so-called \textit{facial} DPs are sequentially convexifiable. Interestingly, all pure binary and mixed 0-1 programming problems are facial DPs, while general mixed (or pure) integer programs are not \cite{4, 5, 6}. Later, Balas et al. \cite{7} extend the sequential convexification property for a general non-convex set with multiple constraints. They provide the necessary and sufficient conditions under which reverse convex programs (DPs with infinitely many terms) are sequentially convexifiable and present classes of problems, in addition to facial DPs, which always satisfy the sequential convexification property. In this paper, we present a decomposition algorithm akin to the Benders’ decomposition \cite{13} and L-shaped method \cite{39} to solve TSS-DPs with sequentially convexifiable DPs in the second stage. We harness the benefits of sequential convexification property for this algorithm and present conditions under which it is finitely convergent. These results generalize the results developed in \cite{37, 38} for TSSPs with mixed 0-1 programs in the second stage, utilizing the reformulation-linearization technique (RLT) of Sherali and Adams \cite{35, 36} and lift-and-project cuts of Balas et al. \cite{5}.

Furthermore, we write a DP equivalent of a semi-continuous program (SCP) which is a linear program with semi-continuity restrictions on some continuous variables, i.e. a variable belongs to a set of the form $[0, \ell] \cup [\ell, \bar{u}]$ where $0 \leq \ell \leq \bar{l} \leq \bar{u}$. Note that by setting $\ell = \bar{l}$, the semi-continuous variable becomes continuous and by setting $\ell = 0$ and $\bar{l} = \bar{u} = 1$, the semi-continuous variable becomes binary. Therefore, mixed 0-1 programs are special cases of SCPS. In this paper, we study two-stage stochastic semi-continuous programs (TSS-SCPs), a special case of TSS-DPs. More specifically, we present a linear programming equivalent for the second stage of the TSS-SCP and showcase how our convexification approach for TSSPs can be utilized to solve a TSS-SCP having semi-continuous inflow set (studied by Angulo et al. \cite{2}) in the second stage.

\textbf{Organization of this paper:} In Section 2, we provide conditions under which the second stage programs of general two-stage mathematical optimization programs (MOPs) in Benders’ form (or TSSPs with $|\Omega| = 1$) and TSSPs in general can be linearized, thereby relaxing the restrictions, if any, on the corresponding second stage variables in these problems. After briefly reviewing the results developed in \cite{3, 4, 5} for the disjunctive programming problems in Section 3, we introduce TSS-DPs in Section 4, extend linear programming equivalent for DPs \cite{3, 4} to TSS-DPs, and provide a decomposition algorithm to solve general TSS-DPs using our convexification approach. We also present another decomposition algorithm, which utilizes the sequential convexification approach within L-shaped method, to solve TSS-DPs with facial DPs in the second stage in finite iterations and TSS-DPs with sequentially convexifiable programs in the second stage. In Section 5, we study TSS-SCPs and showcase how our convexification approach (discussed in Section 2) can be utilized to solve TSS-SCPs, in particular a relaxation of two-stage stochastic semi-continuous network flow problem. In addition, present a linear programming equivalent for the second stage of TSS-SCPs by formulating TSS-SCP as a TSS-DP. Finally, we provide concluding remarks in Section 6.

2. Convexifying second stage programs

In this section, we present conditions sufficient to convexify the second stage programs of general two-stage mathematical optimization programs (MOPs) in Benders’ form (or TSSPs with $|\Omega| = 1$) and TSSPs in general, thereby relaxing the restrictions, if any, on the corresponding second stage
variables in these problems. Later in this paper, we demonstrate the significance of these results in solving TSSPs with disjunctive constraints or semi-continuous variables in the second stage, and TSSP with semi-continuous inflow set in the second stage.

### 2.1 Sufficient conditions to convexify second stage of MOPs in Benders’ form

We consider the following two-stage MOP in Benders’ form which is same as the TSSPs with \(|\Omega| = 1\) (single scenario):

\[
\min \{c^T x + g^T y : x \in X, y \in K(x) \} \tag{5}
\]

where \(K(x) = \{y \in Y(x) : W y \geq r - T x\}\), for \(x \in X\), is the feasible region of the second stage subproblem. We denote the feasible region of the extensive formulation for (5) by \(P := \{(x, y) \in X \times Y(x) : T x + W y \geq r\}\). Also, let \(P_{\text{hull}}\) be a formulation such that \(\text{conv}(P_{\text{hull}}) = \text{conv}(P)\) and \(\Pi(x, y) = c^T x + g^T y\). In the next theorem, we provide conditions sufficient to relax \(Y(x)\) in formulation (5) and \(P\) to \(\mathbb{R}^q\). Bansal et al. (Theorem 4) \([9]\) provide an alternate proof for Theorem 1 stated for two-stage MIP in Benders’ form, a special case of (5) where \(X = \mathbb{Z}^{p_1} \times \mathbb{R}^{p_1}\) and \(Y(x) = \mathbb{Z}^q \times \mathbb{R}^{q_1}\) for all \(x \in X\).

**Theorem 1.** Define

\[
K_{\text{tight}}(x) := \{(y, u) \in \mathbb{R}^q \times \mathbb{R}^{q_2} : W y \geq r - T x, \\
N_2 y + N_3 u \geq d - N_1 x\},
\]

where \(N_1, N_2, N_3,\) and \(d\) are matrices (or vectors) of appropriate sizes. If \(\text{conv}(K(x)) = \text{Proj}_y(K_{\text{tight}}(x))\) for all \(x \in X\), then

\[
P_{\text{tight}} := \{(x, y, u) \in X \times \mathbb{R}^q \times \mathbb{R}^{q_2} : T x + W y \geq r, \\
N_1 x + N_2 y + N_3 u \geq d\},
\]

is a tight extended formulation of \(P\), i.e. \(P_{\text{hull}} = \text{Proj}_{x,y}(P_{\text{tight}})\).

**Proof.** Let \((x^*, y^*)\), \((\hat{x}, \hat{y}, \hat{u})\), and \(\check{y}\) be the optimal solutions of \(\min \{\Pi(x, y) : (x, y) \in P\}\), \(\min \{\Pi(x, y) : (x, y, u) \in P_{\text{tight}}\}\), and \(\min \{\Pi(\hat{x}, y) : y \in K(\hat{x})\}\), respectively. Since \(K(x^*) = \text{Proj}_{x,y}(P)\), \(y^* \in K(x^*)\). Now, for all \(x \in X\), if \(\text{conv}(K(x)) = \text{Proj}_y(K_{\text{tight}}(x))\), then there exists a vector \(u^* \in \mathbb{R}^{q_2}\) such that \((y^*, u^*) \in K_{\text{tight}}(x^*)\). Hence, \((x^*, y^*, u^*) \in P_{\text{tight}}\) as \(K_{\text{tight}}(x^*) = \text{Proj}_{x,y}(P_{\text{tight}})\), implying that \(P_{\text{tight}}\) is an extended formulation of \(P\), i.e., \(P_{\text{hull}} \subseteq \text{Proj}_{x,y}(P_{\text{tight}})\) or \(\Pi(\hat{x}, \hat{y}) \leq \Pi(x^*, y^*)\). Also, since \((\hat{x}, \hat{y}) \in P\), \(\Pi(x^*, y^*) \leq \Pi(\hat{x}, \hat{y})\) because \((x^*, y^*)\) is the optimal solution. Therefore, for each \(c \in \mathbb{R}^p\) and \(g \in \mathbb{R}^q\), we have

\[
\Pi(\hat{x}, \hat{y}) \leq \Pi(x^*, y^*) \leq \Pi(\hat{x}, \hat{y}) \tag{6}
\]

Notice that since \(K_{\text{tight}}(\hat{x}) = \text{Proj}_{x,y}(P_{\text{tight}})\), \((\hat{y}, \hat{u}) \in K_{\text{tight}}(\hat{x})\) or \(\check{y} \in \text{conv}(K(\hat{x}))\). Let \(\check{y} = \sum_i \lambda_i \hat{y}_i\) such that \(\sum_i \lambda_i = 1\), \(\lambda_i \geq 0\), and \(\hat{y}_i \in K(\hat{x})\), i.e. \(\hat{y}_i\) are the vertices of \(\text{conv}(K(\hat{x}))\). This implies that \(g^T \hat{y} = \sum_i \lambda_i (g^T \hat{y}_i)\) which means either \(g^T \hat{y} = (g^T \hat{y}_i)\) for all \(i\) such that \(\lambda_i > 0\) or \(g^T \hat{y} > (g^T \hat{y}_i)\) for some \(i\) with \(\lambda_i > 0\). The latter case is not possible because it contradicts
the optimality of \((\hat{x}, \hat{y})\). Furthermore, \(g^T \hat{y} \leq g^T \hat{y}^i\) for all \(i\) because of optimality, and hence \(\Pi(\hat{x}, \hat{y}) \leq \Pi(\hat{x}, \hat{y})^i\). Using inequalities (6), we get \(\Pi(\hat{x}, \hat{y}) = \Pi(x^*, y^*) = \Pi(\hat{x}, \hat{y})\) for each \(c \in \mathbb{R}^p\) and \(g \in \mathbb{R}^q\), which implies that \(\mathcal{P}_{\text{hull}} = \text{Proj}_{\hat{x},\hat{y}}(\mathcal{P}_{\text{tight}})\). This completes the proof.  

\[\text{Theorem 2.}\] If \(\mathcal{P}_{\text{hull}} = \text{Proj}_{\hat{x},\hat{y}}(\mathcal{P}_{\text{tight}})\) then \(\text{conv}(\mathcal{K}(x)) \subseteq \text{Proj}_y(\mathcal{K}_{\text{tight}}(x))\) for all \(x \in X\) and \(\text{conv}(\mathcal{K}(x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(x))\) for all \(x \in \text{ver}(X)\) where \(\text{ver}(X)\) is the set of vertices of \(\text{conv}(X)\).

**Proof.** We assume that \(\mathcal{P}_{\text{hull}} = \text{Proj}_{\hat{x},\hat{y}}(\mathcal{P}_{\text{tight}})\). First, let \(\hat{y} \in \mathcal{K}(\hat{x})\) for \(\hat{x} \in X\). Since \(\mathcal{K}(\hat{x}) = \text{Proj}_{\hat{x},\hat{y}}(\mathcal{P})\), \((\hat{x}, \hat{y}) \in \mathcal{P}\) and because of our assumption, there exist \(\hat{u} \in \mathbb{R}^q\) such that \((\hat{x}, \hat{y}, \hat{u}) \in \text{Proj}_{\hat{x},\hat{y}}(\mathcal{P}_{\text{tight}})\). Hence, \((\hat{y}, \hat{u}) \in \mathcal{K}_{\text{tight}}(\hat{x})\) because \(\mathcal{K}_{\text{tight}}(\hat{x}) = \text{Proj}_{\hat{x},\hat{y}}(\mathcal{P}_{\text{tight}})\). This implies that for all \(\hat{x} \in X\), \(\text{conv}(\mathcal{K}(\hat{x})) \subseteq \text{Proj}_y(\mathcal{K}_{\text{tight}}(\hat{x}))\). Next, we know that \(\text{conv}(\mathcal{K}(\hat{x})) = \{y : (\hat{x}, y) \in \mathcal{P}_{\text{hull}}\}\) and \(\mathcal{K}_{\text{tight}}(\hat{x}) = \{(y, u) : (\hat{x}, y, u) \in \mathcal{P}_{\text{tight}}\}\) for \(\hat{x} \in X\). However, for \(\hat{x} \in \text{ver}(X)\) where \(\text{ver}(X)\) is the set of vertices of \(\text{conv}(X)\), \(x = \hat{x}\) defines a face of \(\text{conv}(\mathcal{P})\). Therefore, each extreme point of \(\mathcal{P}_{\text{hull}} \cap \{x = \hat{x}\}\) have \(y \in \mathcal{Y}(x)\), and hence \(\text{conv}(\mathcal{K}(\hat{x})) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\hat{x}))\) for \(\hat{x} \in \text{ver}(X)\). This completes the proof.

Bansal et al. [9] provide two examples of MOP in Benders’ form, where \(\mathcal{X} := \mathbb{Z}^p\) and \(\mathcal{X} := \mathbb{R}^p\), for which the second stage mixed integer programs can and cannot, respectively, be convexified by adding a priori parametric cuts. These cuts are obtained from the convexification for the extensive formulation. Next, we use the below presented example of two-stage MIP in Benders’ form with pure integer program in the first stage, i.e. \(\mathcal{X} := \mathbb{Z}^p\), and mixed integer programs in the second stage, i.e. \(\mathcal{Y}(x) := \mathbb{Z}^{q_1} \times \mathbb{R}^{q_2}\), to show that in general, if \(\mathcal{P}_{\text{hull}} = \text{Proj}_{\hat{x},\hat{y}}(\mathcal{P}_{\text{tight}})\) then it is not necessary that \(\text{conv}(\mathcal{K}(x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(x))\) for each \(x \in X\). In other words, for some cases it is possible that there exists an \(\hat{x} \in X := \{x \in \mathbb{Z}^p : Ax \geq b\}\) such that the second stage mixed integer program, \(\mathcal{K}(\hat{x})\), cannot be convexified by adding all parametric inequalities either a priori (like the ones in [9, 24]) or in succession (like the ones in [38, 40]). As a result, for all two-stage MIPs in Benders’ form, a set of all parametric inequalities is not sufficient to provide the second stage optimal solution \(y \in \mathcal{K}(x)\) for each feasible and integral first stage solution.

**Example 1.** Consider the following two-stage MIP in Benders’ form:

\[
\begin{align*}
\min & \quad cx + gy \\
\text{s.t.} & \quad x - y \geq -4 \tag{7} \\
& \quad x + 3y \leq 16 \tag{8} \\
& \quad 3x - 2y \leq 4 \tag{9} \\
& \quad x + 2y \geq 4 \tag{10} \\
& \quad x \leq 4 \tag{11} \\
& \quad x \in \mathbb{Z}_+, y \in \mathbb{Z}_+. \tag{12}
\end{align*}
\]

Here, we have \(\mathcal{X} = \mathbb{Z}_+, X = \{x \in \mathcal{X} : x \leq 4\} = \{0, 1, 2, 3, 4\}\), \(\text{ver}(X) = \{0, 4\}\), \(\mathcal{P} = \{(x, y) : (8) - (13) \text{ hold}\}\), and for \(x \in X\),

\[
\mathcal{K}(x) = \{y \in \mathcal{Y}(x) : y \leq 4 + x, \quad 3y \leq 16 - x, \\
2y \geq -4 + 3x, \quad 2y \geq 4 - x\}
\]

where \(\mathcal{Y}(x) = \mathbb{Z}_+\).
In Figure 1, the light gray region (including boundaries) represents the convex hull of $P$. Observe that $\text{conv}(P) = \mathcal{P}_{\text{hull}} = \mathcal{P}_{\text{tight}} := \{(x, y) \in \mathbb{R}_+ \times \mathbb{R}_+ : (8) - (12) \text{ hold}\}$ and $\mathcal{P}_{\text{tight}}$ is in the $(x, y)$ space, i.e. $\text{Proj}_{x,y}(\mathcal{P}_{\text{tight}}) = \mathcal{P}_{\text{tight}}$. Recall that for $\hat{x} \in X$, $\mathcal{K}(\hat{x}) = \text{Proj}_{x=\hat{x},y}(P)$ and $\mathcal{K}_{\text{tight}}(\hat{x}) = \text{Proj}_{x=\hat{x},y}(\mathcal{P}_{\text{tight}})$. Therefore,

\[
\begin{align*}
\mathcal{K}(0) &:= \{y \in \mathbb{Z}_+ : 2 \leq y \leq 4\}, & \mathcal{K}_{\text{tight}}(0) &:= \{y \in \mathbb{R}_+ : 2 \leq y \leq 4\}, \\
\mathcal{K}(1) &:= \{y \in \mathbb{Z}_+ : 1.5 \leq y \leq 5\}, & \mathcal{K}_{\text{tight}}(1) &:= \{y \in \mathbb{R}_+ : 1.5 \leq y \leq 5\}, \\
\mathcal{K}(2) &:= \{y \in \mathbb{Z}_+ : 1 \leq y \leq 14/3\}, & \mathcal{K}_{\text{tight}}(2) &:= \{y \in \mathbb{R}_+ : 1 \leq y \leq 14/3\}, \\
\mathcal{K}(3) &:= \{y \in \mathbb{Z}_+ : 5/2 \leq y \leq 13/3\}, & \mathcal{K}_{\text{tight}}(3) &:= \{y \in \mathbb{R}_+ : 5/2 \leq y \leq 13/3\}, \\
\mathcal{K}(4) &:= \{y \in \mathbb{Z}_+ : y = 4\}, & \mathcal{K}_{\text{tight}}(4) &:= \{y \in \mathbb{R}_+ : y = 4\}.
\end{align*}
\]

Note that in Figure 1, the red vertical line segments (or point) represent $\mathcal{K}_{\text{tight}}(x)$ for $x = 0, 1, 2, 3, 4$, and in this example, $q_2 = 0$. Now, we make following observations:

i) $\text{conv}(\mathcal{K}(x)) \subseteq \mathcal{K}_{\text{tight}}(x)$ for all $x \in X$, as proved in Theorem 2,

ii) $\text{conv}(\mathcal{K}(x)) = \mathcal{K}_{\text{tight}}(x)$ for all $x \in \text{ver}(X)$, as proved in Theorem 2,

iii) $\text{conv}(\mathcal{K}(x)) \subset \mathcal{K}_{\text{tight}}(x)$ for all $x \in \{1, 2, 3\}$.

This implies that even if $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}})$, it is not necessary that $\text{conv}(\mathcal{K}(x)) = \text{Proj}_{y}(\mathcal{K}_{\text{tight}}(x))$ for each $x \in X$, unless $X = \text{ver}(X)$.

Remark 1. If $\mathcal{X} = \mathbb{Z}^p$ then $\text{ver}(X) \subseteq X$; and if $\mathcal{X} = \{0, 1\}^p$ then $X = \text{ver}(X)$.  

Figure 1: Pictorial representation of $\text{conv}(P)$ and $\mathcal{K}_{\text{tight}}(x), x \in X$, for Example 1
The above example raises critical concerns regarding optimally and finitely (in finite iterations) solving a TSSP where \( \text{ver}(X) \subset X \) using parametric inequalities within a decomposition algorithm similar to the Benders’ decomposition [13] and L-shaped method [39]. The cause of the following concerns is the fact that in general, tightening (or convexifying) the extensive formulation \( P \) similar to the Benders’ decomposition [13] and L-shaped method [39]. The cause of the following concerns is the fact that in general, tightening (or convexifying) the extensive formulation \( P \) using parametric inequalities does not ensure that for \( x \in X \setminus \text{ver}(X) \), \( \text{conv}(K(x)) = \text{Proj}_y(K_{\text{tight}}(x)) \).

I. Finite Convergence. Given an \( \hat{x} \in X \), there may not exist an integer \( I(\hat{x}) < +\infty \) such that the second stage optimal solution \( y \in K(\hat{x}) \) can be found in at most \( I(\hat{x}) \) iterations by using parametric inequalities (added either a priori or in succession) within L-shaped like decomposition algorithm. Therefore, even after assuming \( |X| \) is finite, can we claim that such algorithm will terminate or converge after finite iterations, similar to the finitely convergent Benders’ decomposition algorithm?

II. Optimality Concern. Let \( (x^*, y^*) \) be the optimal solution of a TSSP with \( |\Omega| = 1 \) where \( x^* \in X \setminus \text{ver}(X) \). The Benders’ decomposition algorithm solves the second stage problem, min\{gy : y ∈ K(x^*)\}, and gives a feasible solution \( (x^*, y^*) \) for (7)-(13). However, since \( x^* \in X \setminus \text{ver}(X) \), it is not clear whether solving the relaxed second stage linear program, min\{gy : y ∈ \text{Proj}_y(K_{\text{tight}}(x^*))\} will provide the optimal solution \( (x^*, y^*) \) or not. Therefore, can we claim that using parametric inequalities (added either a priori or in succession) within Benders’ decomposition algorithm will provide the optimal solution of (5)?

Remark 2. As per our knowledge, the concerns related to providing an exact and finitely convergent algorithm for TSSPs have not been explicitly stated in literature except in [16] where these concerns have been briefly mentioned. Nevertheless, Cen [16] provides a heuristic based on dynamic programming to approximately solve the TSSP with only one integer variable in first stage. He also presents an approach using Fenchel cutting planes to obtain upper and lower bounds on the optimal value, and a sub-optimal integer solution. In addition, this approach is forced to terminate by setting an upper bound on the number of iterations.

In this paper, for the first time, we resolve the aforementioned concerns by providing exact and finitely convergent Algorithms 1 and 2 (in Section 4) to solve a class of TSSPs where \( \text{ver}(X) \subset X \), which subsumes TSSPs with pure integer (or binary) variables in the first stage. In Algorithms 1 and 2, we add parametric inequalities a priori and in succession, respectively, within a decomposition algorithm. Using proposition 1, we show that these algorithms provide optimal solution and and in Theorems 10 and 11, we provide conditions under which they are finitely convergent.

Proposition 1. Given \( c \in \mathbb{R}^p \) and \( g \in \mathbb{R}^q \), let \( Q(x) := \min\{g^T y : y \in K(x)\} \) and \( Q_{LP}(x) := \min\{g^T y : y \in \text{Proj}_y(K_{\text{tight}}(x))\} \). If \( P_{\text{hull}} = \text{Proj}_{x,y}(P_{\text{tight}}) \) then \( Q(x) = Q_{LP}(x) \) for \( x \in \text{ver}(X) \cup \{x^*\} \) where \( \text{ver}(X) \) is the set of vertices of \( \text{conv}(X) \) and \( (x^*, y^*) \) is the optimal solution of (5).

Proof. Assume that \( \text{conv}(P) = P_{\text{hull}} = \text{Proj}_{x,y}(P_{\text{tight}}) \). First let \( \hat{x} \in \text{ver}(X) \) where \( \text{ver}(X) \) is the set of vertices of \( \text{conv}(X) \). According to Theorem 2, \( \text{conv}(K(\hat{x})) = \text{Proj}_y(K_{\text{tight}}(\hat{x})) \). Therefore, \( Q(\hat{x}) = Q_{LP}(\hat{x}) \) for \( \hat{x} \in \text{ver}(X) \). Now, let \( (x^*, y^*) \) be the optimal solution of (5). Because of our assumption, \( (x^*, y^*) \) should also be the optimal solution of \( \min\{\Pi(x, y) : (x, y) \in \text{Proj}_{x,y}(P_{\text{tight}})\} \) and therefore, \( y^* \in \text{Proj}_y(K_{\text{tight}}(x^*)) \). Assume that \( \hat{y} \neq y^* \) is the optimal solution of \( Q_{LP}(x^*) \). This implies that \( g^T \hat{y} < g^T y^* \) or \( \Pi(x^*, \hat{y}) < \Pi(x^*, y^*) \), which is a contradiction as \( (x^*, \hat{y}) \in \text{Proj}_{x,y}(P_{\text{tight}}) \). Hence, \( \hat{y} = y^* \) and \( Q(x^*) = Q_{LP}(x^*) \). This completes the proof.

7
2.2 Convexifying second stage MP in two-stage stochastic programs

We now present conditions under which the convexification of the second stage of a TSSP can be used to relax the constraints, if any, on the corresponding second stage variables in the problem. By assuming that $\Omega$ is a finite set and the probability of occurrence of scenario $\omega \in \Omega$ is $p_w$, we can re-write (1) as a large-scale MOP:

\[
\begin{align*}
\min & \quad c^T x + \sum_{\omega \in \Omega} p_\omega g(\omega)^T y(\omega) \\
\text{s.t.} & \quad Ax \geq b \\
& \quad T(\omega)x + W(\omega)y(\omega) \geq r(\omega), \quad \omega \in \Omega \\
& \quad x \in X, y(\omega) \in Y(\omega, x), \quad \omega \in \Omega.
\end{align*}
\]

(14) (15) (16) (17)

We denote the feasible region of this formulation, which is also referred to as the extensive formulation of the TSSP, by $\mathcal{P} := \{(x, y) : (15) - (17) \text{ hold}\}$ where $y = \{y(\omega), \omega \in \Omega\}$. Also, let $\mathcal{P}_{\text{hull}}$ be a formulation such that $\text{conv}(\mathcal{P}_{\text{hull}}) = \text{conv}(\mathcal{P})$ and $\Pi(x, y) = c^T x + \sum_{\omega \in \Omega} p_\omega g(\omega)^T y(\omega)$. For $\omega \in \Omega$ and $x \in X$, assume that for known $N_1(\omega)$, $N_2(\omega)$, $N_3(\omega)$, and $d(\omega)$ matrices (or vectors), we have

\[
\mathcal{K}_{\text{tight}}(\omega, x) := \{(y(\omega), u(\omega)) \in \mathbb{R}^q \times \mathbb{R}^p : W(\omega)y(\omega) \geq r(\omega) - T(\omega)x, \quad N_2(\omega)y(\omega) + N_3(\omega)u(\omega) \geq d(\omega) - N_1(\omega)x\}.
\]

Lemma 1. Let $\omega_1 \in \Omega$. If $\text{conv}(\mathcal{K}(\omega_1, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega_1, x))$ for all $x \in X$, then $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}})$ where

\[
\mathcal{P}_{\text{tight}} := \left\{T(\omega)x + W(\omega)y(\omega) \geq r(\omega), \quad \omega \in \Omega, \quad N_1(\omega_1)x + N_2(\omega_1)y(\omega_1) + N_3(\omega_1)u(\omega_1) \geq d(\omega_1) \right\}.
\]

(18)

Proof. Similar to the proof of Theorem 1, let $(x^*, y^*)$, $(\hat{x}, \hat{y}, \hat{u})$, and $\tilde{y}$ be the optimal solutions of $\min \{\Pi(x, y) : (x, y) \in \mathcal{P}\}$, $\min \{\Pi(x, y) : (x, y, u) \in \mathcal{P}_{\text{tight}}\}$, and $\min \{\Pi(\hat{x}, y(\omega_1), \hat{y}(\omega_2), \ldots, \hat{y}(\omega_{|\Omega|})) : y(\omega_1) \in \mathcal{K}(\omega_1, \hat{x})\}$, respectively. Since $\mathcal{K}(\omega_1, x^*) = \text{Proj}_{x=x^*, y(\omega_1)}(\mathcal{P})$, $y^*(\omega_1) \in \mathcal{K}(\omega_1, x^*)$. Now, for all $x \in X$ and $\omega_1 \in \Omega$, if $\text{conv}(\mathcal{K}(\omega_1, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega_1, x))$ then there exists a vector $u^*(\omega_1) \in \mathbb{R}^p$ such that $(y^*(\omega_1), u^*(\omega_1)) \in \mathcal{K}_{\text{tight}}(\omega_1, x^*)$. Hence, $(x^*, y^*, u^*) \in \mathcal{P}_{\text{tight}}$ as $\mathcal{K}_{\text{tight}}(\omega_1, x^*) = \text{Proj}_{x=x^*, y(\omega_1)}(\mathcal{P}_{\text{tight}})$, implying that $\mathcal{P}_{\text{tight}}$ is an extended formulation of $\mathcal{P}$, i.e., $\mathcal{P}_{\text{hull}} \subseteq \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}})$ or $\Pi(\tilde{x}, \tilde{y}) \leq \Pi(x^*, y^*)$. Also, since $(\hat{x}, \hat{y}) \in \mathcal{P}$, $\Pi(x^*, y^*) \leq \Pi(\hat{x}, \hat{y})$ because $(x^*, y^*)$ is the optimal solution. Therefore, for each $c \in \mathbb{R}^p$ and $g(\omega) \in \mathbb{R}^q$, $\omega \in \Omega$, we have

\[
\Pi(\tilde{x}, \tilde{y}) \leq \Pi(x^*, y^*) \leq \Pi(\hat{x}, \hat{y}).
\]
for some $i$ with $\lambda_i > 0$. The latter case is not possible because it contradicts the optimality of $(\hat{x}, \hat{y})$. Furthermore, $g(\omega_1)^{T} \hat{y}(\omega_1) \leq g(\omega_1)^{T} \tilde{y}(\omega_1)$ for all $i$ because of optimality, and hence $\Pi(\hat{x}, \hat{y}) \leq \Pi(\hat{x}, \tilde{y})$. Using inequalities (18), we get $\Pi(\hat{x}, \hat{y}) = \Pi(x^*, y^*) = \Pi(\hat{x}, \tilde{y})$ for each $c \in \mathbb{R}^p$ and $g(\omega) \in \mathbb{R}^q$, $\omega \in \Omega$, which implies that $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}^1_{\text{tight}})$.

**Remark 3.** Bansal et al. (Lemma 1) [9] provide an alternate proof for the above result for TSS-MIPs, a special case of TSSP where $\mathcal{X} = \mathbb{Z}^p \times \mathbb{R}^{p-p}$ and $\mathcal{Y}(\omega, x) = \mathbb{Z}^q \times \mathbb{R}^{q-q}$ for all $(\omega, x) \in (\Omega, X)$.

**Theorem 3.** If $\text{conv}(\mathcal{K}(\omega, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega, x))$ for all $(x, \omega) \in (X, \Omega)$ then $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}})$, where

$$
\mathcal{P}_{\text{tight}} := \{T(\omega)x + W(\omega)y(\omega) \geq r(\omega), \quad \omega \in \Omega, \\
N_1(\omega)x + N_2(\omega)y(\omega) + N_3(\omega)u(\omega) \geq d(\omega), \quad \omega \in \Omega, \\
x \in X, y(\omega) \in \mathbb{R}^q, u(\omega) \in \mathbb{R}^{q_2} \quad \omega \in \Omega}. 
$$

**Remark 4.** The proof of this theorem is same as the proof given by Bansal et al. (Theorem 5) [9] for TSS-MIPs; but for the sake of completeness, we re-write the proof.

**Proof.** The proof follows Lemma 1 and an induction over $\omega \in \Omega = \{\omega_1, \ldots, \omega_{|\Omega|}\}$. According to Lemma 1, if $\text{conv}(\mathcal{K}(\omega_1, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega_1, x))$ for all $x \in X$, then $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}^1_{\text{tight}})$. Now, assume that for all $x \in X$, $\text{conv}(\mathcal{K}(\omega_2, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega_2, x))$. Then by considering $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}^1_{\text{tight}})$ and using the arguments similar to the proof of Lemma 1, we can prove that $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}^2_{\text{tight}})$, where

$$
\mathcal{P}^2_{\text{tight}} := \{T(\omega)x + W(\omega)y(\omega) \geq r(\omega), \quad \omega \in \Omega, \\
N_1(\omega_1)x + N_2(\omega_1)y(\omega_1) + N_3(\omega_1)u(\omega_1) \geq d(\omega_1) \\
N_1(\omega_2)x + N_2(\omega_2)y(\omega_2) + N_3(\omega_2)u(\omega_2) \geq d(\omega_2) \\
x \in X, y(\omega_1), y(\omega_2) \in \mathbb{R}^q, u(\omega_1), u(\omega_2) \in \mathbb{R}^{q_2}, \\
y(\omega) \in \mathcal{Y}(\omega, x), \omega \in \Omega \setminus \{\omega_1, \omega_2\}. 
$$

Next, we apply above discussed steps one by one for $\omega_i, \ i = 3, \ldots, |\Omega|$, by considering $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}^{i-1}_{\text{tight}})$ and assuming that $\text{conv}(\mathcal{K}(\omega_i, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega_i, x))$ for all $x \in X$. This gives $\mathcal{P}_{\text{hull}} = \text{Proj}_{x,y}(\mathcal{P}^{|\Omega|}_{\text{tight}}) = \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}})$ and completes the proof.

**Corollary 1.** If $\text{conv}(\mathcal{K}(\omega, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega, x))$ for all $(x, \omega) \in (X, \Omega)$, then the function $\mathbb{E}_\omega[Q(\omega, x)]$ is piecewise linear convex in $x$.

**Corollary 2.** If $\text{conv}(\mathcal{K}(\omega, x)) = \text{Proj}_y(\mathcal{K}_{\text{tight}}(\omega, x))$ for a given $(\omega, x) \in \Omega \times X$, then the recourse function $Q(\omega, x)$ is underestimated by

$$
Q(\omega, x) \geq (r(\omega)^T \pi^*_1(\omega) + d(\omega)^T \pi^*_2(\omega)) - (T(\omega)^T \pi^*_1(\omega) + N_1(\omega)^T \pi^*_2(\omega)) x 
$$

where $(\pi^*_1(\omega), \pi^*_2(\omega))$ is the dual optimal solution of the linear programming problem: $\min\{g(\omega)^T y(\omega) : (y(\omega), u(\omega)) \in \mathcal{K}_{\text{tight}}(\omega, x)\}$. 

9
We now show how a tight (extended) formulation of a substructure of the extensive formulation of TSSP, defined by
\[ \mathcal{P}(\omega) := \{(x, y(\omega)) \in X \times \mathcal{Y}(\omega, x) : T(\omega)x + W(\omega)y(\omega) \geq r(\omega)\}, \]
for \( \omega \in \Omega \), can be used to get valid parametric inequalities for the second stage of the problem. Later, we will present examples of TSSPs for which these inequalities are sufficient to convexify the second stage programs with disjunctive constraints or semi-continuous variables. Let \( \mathcal{P}_{\text{hull}}(\omega) \) be a formulation such that \( \text{conv}(\mathcal{P}_{\text{hull}}(\omega)) = \text{conv}(\mathcal{P}(\omega)) \).

**Corollary 3.** For \( \omega \in \Omega \), let
\[ \mathcal{P}_{\text{tight}}(\omega) := \{(x, y(\omega), u(\omega)) \in X \times \mathbb{R}^q \times \mathbb{R}^q : T(\omega)x + W(\omega)y(\omega) \geq r(\omega) \]
\[ N_1(\omega)x + N_2(\omega)y(\omega) + N_3(\omega)u(\omega) \geq d(\omega)\}. \]
If \( \mathcal{P}_{\text{hull}}(\omega) = \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}}(\omega)) \) then \( \text{conv}(\mathcal{K}(\omega, x)) \subseteq \text{Proj}_{y}(\mathcal{K}_{\text{tight}}(\omega, x)) \) for all \( x \in X \) and \( \text{conv}(\mathcal{K}(\omega, x)) = \text{Proj}_{y}(\mathcal{K}_{\text{tight}}(\omega, x)) \) for all \( x \in \text{ver}(X) \) where \( \text{ver}(X) \) is the set of vertices of \( \text{conv}(X) \).

**Corollary 4.** For \( \omega \in \Omega \), \( \mathcal{P}_{\text{hull}} = \text{conv}(\bigcap_{\omega \in \Omega \setminus \{\omega\}} \mathcal{P}(\omega)) \cap \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}}(\omega)) \) if \( \text{conv}(\mathcal{K}(\omega, x)) = \text{Proj}_{y}(\mathcal{K}_{\text{tight}}(\omega, x)) \) for all \( x \in X \).

**Corollary 5.** If \( \text{conv}(\mathcal{K}(\omega, x)) = \text{Proj}_{y}(\mathcal{K}_{\text{tight}}(\omega, x)) \) for all \( (x, \omega) \in (X, \Omega) \) then \( \mathcal{P}_{\text{hull}} \) is given by \( \bigcap_{\omega \in \Omega} \text{Proj}_{x,y}(\mathcal{P}_{\text{tight}}(\omega)) \).

Bansal et al. (Theorems 6 and 7) [9] demonstrated the significance of the above results by considering TSS-MIPs having special cases of the parametrized continuous multi-mixing set [8, 10, 11, 12] in the second stage.

### 3. Necessary background on disjunctive programming

As mentioned before, a disjunctive program is a linear program with disjunctive constraints, i.e. linear inequalities connected by \( \lor \) ("or", disjunction) logical operations. Given non-empty polyhedra \( \mathcal{R}_i := \{z \in \mathbb{R}^n : E^i z \geq f^i\}, i \in L \), the disjunctive normal form representation for the set \( \mathcal{R} = \bigcup_{i \in L} \mathcal{R}_i \) is \( \bigvee_{i \in L} \big(E^i z \geq f^i\big) \), where
\[ E^i = \begin{pmatrix} E^i_1 \\ E^i_2 \end{pmatrix} \quad \text{and} \quad f^i = \begin{pmatrix} f^i_1 \\ f^i_2 \end{pmatrix}, \quad i \in L. \]
Let \( \mathcal{R}_0 := \{z \in \mathbb{R}^n : E_1 z \geq f_1\} \) be the linear programming relaxation of \( \mathcal{R} \). In this section, we briefly review the results developed in [3, 4, 5] for disjunctive programming problems to provide the necessary background for the results in the following sections. Theorem 4 provides a tight extended formulation for the convex hull of the points satisfying disjunctive constraints. Theorems 5 and 6 provide the convex hull description of the union of the polyhedra, \( \bigcup_{i \in L} \mathcal{R}_i \), in the original z-space.
Theorem 4 ([3, 4]). The closed convex hull of $\bigcup_{i \in L} R_i$ is the projection of the following extended formulation (19), denoted by $R_{TEF}$, onto the $z$-space:

$$z = \sum_{i \in L} \zeta_i$$
$$E^i \zeta_i \geq f^i \zeta_0^i, \quad i \in L$$
$$\sum_{i \in L} \zeta_i = 1$$
$$\langle \zeta_i, \zeta_0 \rangle \geq 0, \quad i \in L.$$ 

(19)

Theorem 5 ([3, 4]). The projection of $R_{TEF}$ onto the $z$-space is given by:

$$\text{Proj}_z(R_{TEF}) = \{ z \in \mathbb{R}^n : \alpha z \geq \beta \text{ for all } (\alpha, \beta) \in C_0 \}$$

where $C_0 := \{ (\alpha, \beta) \in \mathbb{R}^{n+1} : \alpha = \sigma^i E^i, \beta = \sigma^i f^i \text{ for some } \sigma^i \geq 0, i \in L \}$. Let $R_{hull} = \text{conv}(\bigcup_{i \in L} R_i)$. Then, the cone of all valid inequalities for $R_{hull}$, denoted by $R_{hull}^*$, is same as the polyhedral cone $C_0$, i.e.

$$R_{hull}^* := \{ (\alpha, \beta) \in \mathbb{R}^{n+1} : \alpha z \geq \beta \text{ for all } z \in R_{hull} \}$$
$$= \{ (\alpha, \beta) \in \mathbb{R}^{n+1} : \alpha = \sigma^i E^i, \beta = \sigma^i f^i \text{ for some } \sigma^i \geq 0, i \in L \}.$$ 

Theorem 6 ([3, 4]). Assuming $R_{hull}$ is full dimensional. The inequality $\alpha z \geq \beta$ defines a facet of $R_{hull}$ if and only if $(\alpha, \beta)$ is an extreme ray of the cone $R_{hull}^*$.

Since the logical operations $\lor$ (disjunction) and $\land$ (“and”, conjunction) obey the distributive law, i.e., $(a_1 \land a_2) \lor (b_1 \land b_2) = (a_1 \lor b_1) \land (a_1 \lor b_2) \land (a_2 \lor b_1) \land (a_2 \lor b_2)$, the set $R$ can also be written in the conjunctive normal form as

$$\left\{ z \in \mathbb{R}^n : E_1 z \geq f_1, \bigwedge_{j=1}^m \left( \bigvee_{i \in L_j} d_i z \geq d_0^i \right) \right\}$$

where $|L_j| = |L|$ for $j = 1, \ldots, m$ and each disjunction $j$ contains exactly one inequality from the system $E_i z \geq f_i^j, i \in L$. The disjunctive set $R$ is called facial if each inequality $d_i z \geq d_0^i, i \in L_j, j = 1, \ldots, m$, defines a face of $R_0$. In this paper, we introduce a new concept of “super-facial” disjunctive set, which is defined as follows.

Definition 1. A disjunctive set $R$ is called super-facial if it is a facial set and for each $\hat{z} \in R$, $R_0 \cap \{ z = \hat{z} \}$ is also a face of $R_0$, i.e., $\hat{z} \in R$ is a vertex of $\text{conv}(R)$.

Remark 5. The set of feasible solutions of a mixed 0-1 program is facial, but not super-facial; whereas the set of feasible solutions of a pure 0-1 program is super-facial.

Theorem 7 ([3, 4]). If $R$ is facial then $\Pi_m = \text{conv}(R)$, where $\Pi_0 := R_0$,

$$\Pi_j := \text{conv} \left( \Pi_{j-1} \cap \left\{ z : \bigvee_{i \in L_j} d_i z \geq d_0^i \right\} \right),$$

for $j = 1, \ldots, m$. 

11
According to Theorem 7, the convex hull of the facial disjunctive set $\mathcal{R}$ can be obtained in a sequence of $m$ steps, where at each step the convex hull of points satisfying only one disjunctive constraints is generated. This property is referred to as the sequential convexification, and a subclass of DPs which satisfy this property is called sequentially convexification DPs. Balas et al. [7] extend the sequential convexification property for a general non-convex set with multiple constraints. They provide the necessary and sufficient conditions under which reverse convex programs (DPs with infinitely many terms) are sequentially convexifiable, and present classes of problems, in addition to facial DPs, which always satisfy the sequential convexification property.

4. Two-stage stochastic disjunctive programs (TSS-DPs)

In this section, we introduce two-stage stochastic disjunctive programs (TSS-DPs) which are TSSPs with disjunctive constraints in both first and second stages. We explicitly write a TSS-DP as follows:

$$\begin{align*}
\min & \quad c^T x + E_\omega [Q(\omega, x)] \\
\text{s.t.} & \quad Ax \geq b \\
& \quad x \in \bigvee_{s \in S} (\bar{D}_s x \geq \bar{d}_s) \\
& \quad x \in \mathbb{R}^p
\end{align*}$$

(20)

where $S, \Omega$ are finite sets, and for any scenario $\omega$ of $\Omega$ and a finite set $H$,

$$Q(\omega, x) := \min g(\omega)^T y(\omega)$$

(21)

$$\text{s.t.} \quad W(\omega)y(\omega) \geq r(\omega) - T(\omega)x$$

(22)

$$\bigvee_{h \in H} \left( D^h_1(\omega)y(\omega) \geq d^h_0(\omega) - D^h_2(\omega)x \right)$$

(23)

$$y(\omega) \in \mathbb{R}^q.$$  

(24)

We re-write the disjunctive constraint in the first stage, i.e. $\bigvee_{s \in S} (\bar{D}_s x \geq \bar{d}_s)$, and constraint (23) in the conjunctive normal form (based on the above mentioned definition) as

$$\bigvee_{j=1}^{m_1} \left( \bigvee_{i \in S_j} \mu^i x \geq \mu^i_0 \right) \quad \text{and} \quad \bigvee_{j=1}^{m_2} \left( \bigvee_{i \in H_j} \eta^i(\omega)y(\omega) \geq \eta^i_0(\omega) - \eta^i_2(\omega)x \right),$$

respectively. Here each disjunction $j$ contains exactly one inequality from each system of inequalities in the corresponding disjunctive constraint, i.e. $|S_j| = |S|$ and $|H_j| = |H|$. We use the notations $X, \mathcal{P}, \mathcal{P}_{hull}, \mathcal{P}(\omega)$ and $\mathcal{P}_{tight}(\omega)$ for $\omega \in \Omega$, and $\mathcal{K}(\omega, x)$ and $\mathcal{K}_{tight}(\omega, x)$ for $(\omega, x) \in \Omega \times X$, same as defined for TSSPs in the previous sections, except that $\mathcal{Y}(\omega) = \{ y(\omega) \in \mathbb{R}^q : \bigvee_{h \in H} \left( D^h_1(\omega)y(\omega) \geq d^h_0(\omega) - D^h_2(\omega)x \right) \}$ and $\mathcal{X} := \{ x \in \mathbb{R}^p : \bigvee_{s \in S} (\bar{D}_s x \geq \bar{d}_s) \}$. Letting

$$W^h(\omega) := \left( \begin{array}{c} W(\omega) \\ D^h_1(\omega) \end{array} \right), \quad T^h(\omega) := \left( \begin{array}{c} T(\omega) \\ D^h_2(\omega) \end{array} \right), \quad r^h(\omega) := \left( \begin{array}{c} r(\omega) \\ d^h_0(\omega) \end{array} \right),$$

and $\mathcal{K}^h(\omega, x) := \{ y(\omega) \in \mathbb{R}^q_+ : W^h(\omega)y(\omega) \geq r^h(\omega) - T^h(\omega)x \} \neq \emptyset$ for $(\omega, x, h) \in (\Omega, X, H)$, we get

$$\mathcal{K}(\omega, x) = \bigcup_{h \in H} \mathcal{K}^h(\omega, x) = \left\{ \bigvee_{h \in H} \left( W^h(\omega)y(\omega) \geq r^h(\omega) - T^h(\omega)x \right) \right\}.$$
where $K^h$, for each $h \in H$, is a polyhedral set, and
\[
P(\omega) = \left\{ (x, y(\omega)) : x \in X, \bigvee_{h \in H} \left( T^h(\omega)x + W^h(\omega)y(\omega) \geq r^h(\omega) \right) \right\}.
\]

Also, let $X_{LP} := \{x \in \mathbb{R}^p : Ax \geq b\}$ and $K_{LP}(\omega, x) := \{y(\omega) \in \mathbb{R}^q : (22) \text{ hold}\}$ for $(\omega, x) \in (\Omega, X)$. In the following theorems, we extend the result of Balas [3, 4] for DP (Theorem 4) by providing conditions to get a linear programming equivalent for the second stage of TSS-DPs, i.e. $K(\omega, x)$.

**Theorem 8.** If the disjunctive set $X$ is super-facial (according to Definition 1) then $\text{conv}(K(\omega, x)) = \text{Proj}_{y(\omega)}(K_{\text{tight}}(\omega, x))$ for all $(\omega, x) \in (X, \Omega)$ where
\[
K_{\text{tight}}(\omega, x) := \left\{ \sum_{h \in H} \xi_1^h(\omega) - y(\omega) = 0, \sum_{h \in H} \xi_2^h(\omega) = x, W^h(\omega)\xi_1^h(\omega) + T^h(\omega)\xi_2^h(\omega) \geq r^h(\omega)\xi_0^h(\omega), h \in H, \sum_{h \in H} \xi_0^h(\omega) = 1, y(\omega) \in \mathbb{R}^q, \xi_1^h(\omega) \in \mathbb{R}^q_+, \xi_2^h(\omega) \in \mathbb{R}^p, \xi_0^h(\omega) \in \mathbb{R}_+, h \in H \right\}.
\]

**Proof.** Using Theorem 4, we derive a tight extended formulation for $P(\omega), \omega \in \Omega$, which is given by
\[
P_{\text{tight}}(\omega) := \left\{ y(\omega) = \sum_{h \in H} \xi_1^h(\omega), x = \sum_{h \in H} \xi_2^h(\omega), W^h(\omega)\xi_1^h(\omega) + T^h(\omega)\xi_2^h(\omega) \geq r^h(\omega)\xi_0^h(\omega), h \in H, \sum_{h \in H} \xi_0^h(\omega) = 1, x \in X, y(\omega) \in \mathbb{R}^q, \xi_1^h(\omega) \in \mathbb{R}^q_+, \xi_2^h(\omega) \in \mathbb{R}^p, \xi_0^h(\omega) \in \mathbb{R}_+, h \in H \right\}.
\]

This means $\text{conv}(P(\omega)) = \text{Proj}_{x,y(\omega)}(P_{\text{tight}}(\omega))$ for $\omega \in \Omega$. Let $\hat{x} \in X = \{x \in \mathbb{R}^p : Ax \geq b, \bigwedge_{j=1}^{m_1} \left( \bigvee_{i \in S_j} \mu^i x \geq \mu_0^i \right) \}$. Therefore, for each $j \in \{1, \ldots, m_1\}$, there exist at least one inequality $\mu^i x \geq \mu_0^i, i \in S_j$, such that $\mu^i \hat{x} \geq \mu_0^i$. Let the disjunctive set $X$ be super-facial, i.e. each $X_i := X_{LP} \cap \{x \in \mathbb{R}^p : \mu^i x \geq \mu_0^i \}$ is a face of $X_{LP}$, for all $i \in S_j$, and $X \cap \{x = \hat{x}\}$ is also a face of $X_{LP}$. As a result, $\hat{x} \in \text{ver}(X)$ where $\text{ver}(X)$ is the set of vertices of $\text{conv}(X)$. Now using the arguments similar to the ones used in the proof of Theorem 2, we can prove that $\text{conv}(K(\omega, \hat{x})) = \text{Proj}_{y(\omega)}(K_{\text{tight}}(\omega, \hat{x})) = \text{Proj}_{x=\hat{x},y(\omega)}(P_{\text{tight}}(\omega))$ for all $(\omega, \hat{x}) \in (\Omega, X)$. \hfill \Box
Theorem 9. If $X$ is super-facial then $P_{halt} = \text{Proj}_{x,y}(P_{tight})$, where

$$ P_{tight} := \left\{ y(\omega) = \sum_{h \in H} \xi^h_1(\omega), \quad \omega \in \Omega \right\} \quad \left\{ \begin{array}{l} x = \sum_{h \in H} \xi^h_2(\omega), \quad \omega \in \Omega \\ W^h(\omega)\xi^h_1(\omega) + T^h(\omega)\xi^h_2(\omega) \geq r^h(\omega)\xi^h_0(\omega), \quad \omega \in \Omega, h \in H \\ \sum_{h \in H} \xi^h_0(\omega) = 1 \quad \omega \in \Omega \\ x \in X, y(\omega) \in \mathbb{R}^q, \quad \omega \in \Omega \\ \xi^h_1(\omega) \in \mathbb{R}_+, \xi^h_2(\omega) \in \mathbb{R}_+, \xi^h_0(\omega) \in \mathbb{R}_+, \quad \omega \in \Omega, h \in H \end{array} \right\}. $$

Proof. We assume that the disjunctive set $X$ is super-facial. Therefore, according to Theorem 8, $\text{conv}(K(\omega, x)) = \text{Proj}_{y(\omega)}(K_{tight}(\omega, x))$ for all $(x, \omega) \in (X, \Omega)$. Now, because of Theorem 3, we can deduce that $P_{halt} = \text{Proj}_{x,y}(P_{tight})$ where $P_{tight} = \cap_{\omega \in \Omega} P_{tight}(\omega)$ such that $K_{tight}(\omega, x) = \text{Proj}_{x,y}(\omega)(P_{tight}(\omega))$ for all $(x, \omega) \in (X, \Omega)$. \hfill $\blacksquare$

Next, we present two decomposition algorithms similar to the Benders’ decomposition [13] and L-shaped method [39] to solve: (i) General TSS-DPs (20) using convexification result in Theorem 8 and (ii) TSS-DPs (20) where disjunctive constraints (23) in the second stage are sequentially convexifiable. We also provide conditions under which these algorithms are finitely convergent.

4.1 Decomposition algorithm for general TSS-DPs

The pseudocode of a decomposition algorithm which utilizes our convexification approach (in particular, Theorem 8) to solve general TSS-DPs (20) is given by Algorithm 1. Let $LB$ and $UB$ be the lower bound and upper bound, respectively, on the optimal solution value of a given TSS-DP. We denote the following strengthened linear programming relaxation of the second stage disjunctive program (21)-(24) by SLP($\omega, x$), $(\omega, x) \in (\Omega, X)$:

$$ Q_{LP}(\omega, x) := \min\{ g(\omega)^T y(\omega) : y(\omega) \in \text{Proj}_{y(\omega)}(K_{tight}(\omega, x)) \}, \quad (27) $$

where $K_{tight}(\omega, x)$ is defined by (25). Also, let $\pi^*(\omega, x)$ be the optimal dual multipliers obtained by solving SLP($\omega, x$) for a given $(\omega, x) \in (\Omega, X)$, and $K_{tight}(\omega, x)$ be written in a compact form as

$$ \left\{ N_2(\omega)y(\omega) + \sum_{h \in H} \{ N^h_3(\omega)\xi^h_1(\omega) + N^h_4(\omega)\xi^h_2(\omega) + N^h_5(\omega)\xi^h_0(\omega) \} \geq \Delta(\omega) - N_1(\omega)x \right\} $$

$$ y(\omega) \in \mathbb{R}^q, \xi^h_1(\omega) \in \mathbb{R}_+, \xi^h_2(\omega) \in \mathbb{R}_+, \xi^h_0(\omega) \in \mathbb{R}_+, h \in H \right\}. $$

Then, the corresponding optimality cut, OC($x$), is

$$ \sum_{\omega \in \Omega} p(\omega)\{ \pi^*(\omega, x)^T \Delta(\omega) - N_1(\omega)x \} \geq \theta. \quad (28) $$
Algorithm 1: Decomposition Algorithm for General TSS-DPs (20)

1: Initialization: $l \leftarrow 1$, $LB \leftarrow -\infty$, $UB \leftarrow \infty$. Assume $x^1 \in X$.

2: while $UB - LB > \epsilon$ do $\triangleright$ $\epsilon$ is a pre-specified tolerance

3: for $\omega \in \Omega$ do

4: Solve linear program $SLP(\omega, x^l)$;

5: $y^*(\omega, x^l) \leftarrow$ optimal solution; $Q_{LP}(\omega, x^l) \leftarrow$ optimal solution value;

6: $\pi^*(\omega, x^l) \leftarrow$ optimal dual multipliers;

7: end for

8: if $y^*(\omega, x^l) \in K(\omega, x^l)$ for all $\omega \in \Omega$ and $UB > c^T x^l + \sum_{\omega \in \Omega} p_\omega Q_{LP}(\omega, x^l)$ then

9: $UB \leftarrow c^T x^l + \sum_{\omega \in \Omega} p_\omega Q_{LP}(\omega, x^l)$;

10: if $UB \leq LB + \epsilon$ then

11: Go to Line 27;

12: end if

13: end if

14: Derive optimality cut $OC(x^l)$ using (28);

15: Add $OC(x^l)$ to $\mathcal{M}_{l-1}$ to get $\mathcal{M}_l$;

16: Solve master problem $\mathcal{M}_l$;

17: $x^{l+1} \leftarrow$ optimal solution; $LB \leftarrow$ optimal solution value;

18: if $x^{l+1} = x^l \triangleright$ Alternate optimal solutions

19: $\hat{X}_l \leftarrow$ set of $x$ components of all alternative optimal solutions of $\mathcal{M}_l$;

20: for all $\hat{x} \in \hat{X}_l \setminus \{x^{l+1}\}$ do

21: Repeat Lines 3-13 where $x^l = \hat{x}$;

22: end for

23: end if

24: $l \leftarrow l + 1$;

25: end while

26: return $(x^l, \{y(\omega, x^l)\}_{\omega \in \Omega}), UB$

These cuts help in deriving a lower bounding approximation of the first stage problem (20), defined by

$$
\min \ c^T x + \theta \\
\text{s.t. } x \in X \\
\sum_{\omega \in \Omega} p_\omega (\pi^*(\omega, x^k)^T (\Delta(\omega) - N(\omega)x)) \geq \theta, \text{ for } k = 1, \ldots, l,
$$

(29)

where $x^k \in X$ for $k = 1 \ldots, l$. We denote problem (29) by $\mathcal{M}_l$ for $l \in \mathbb{Z}_+$ and refer to it as the master problem at iteration $l$. Note that $\mathcal{M}_0$ is the master problem without any optimality cut.

Now we initialize Algorithm 1 by setting $LB$ to negative infinity, $UB$ to positive infinity, iteration counter $l$ to 1, and selecting a first stage feasible solution $x^1 \in X$ (Line 1). At each iteration $l \geq 1$, we solve linear programs $SLP(\omega, x^l)$ for all $\omega \in \Omega$ and store the corresponding optimal solution $y^*(\omega, x^l)$, the optimal objective value $Q_{LP}(\omega, x^l) := g(\omega)^T y^*(\omega, x^l)$, and the optimal dual multipliers $\pi^*(\omega, x^l)$, for each $\omega \in \Omega$ (Lines 3-7). Interestingly, in case $y^*(\omega, x^l) \in K(\omega, x^l)$ for all $\omega \in \Omega$, we have a feasible solution $(x^l, y^*(\omega_1, x^l), \ldots, y^*(\omega_{|\Omega|}, x^l))$ for the original problem. Therefore, we update $UB$ if the solution value corresponding to thus obtained feasible solution is smaller than the existing best known upper bound (Lines 8-9). We also utilize this stored information to derive optimality cut $OC(x^l)$ using (28) and add this cut to the master problem $\mathcal{M}_{l-1}$ to get an augmented master
problem \( M_l \) (Lines 14-15). We solve \( M_l \) and since it is a lower bounding approximation of \((20)\), we use the optimal solution value associated with \( M_l \) to update \( LB \). It is important to note that the lower bound \( LB \) is a non-increasing function with respect to the iterations. This is because \( M_{l-1} \) is a relaxation of \( M_l \) for each \( l \geq 1 \). Therefore, after every iteration the difference between the bounds, \( UB - LB \), either decreases or remains same as in previous iteration. We terminate our algorithm when this difference becomes zero, i.e., \( UB = LB \), or reaches a pre-specified tolerance \( \epsilon \) (Line 2 or Lines 10-12), and return the optimal solution \((x^l, \{y(\omega, x^l)\}_{\omega \in \Omega}) \) and the optimal objective value \( UB \).

While solving a TSS-DP \((20)\) with alternative optimal solutions using Algorithm 1, it is possible that at some iteration \( l \), the optimal solution, \( x^{l+1} \), obtained after solving \( M_l \) is same as \( x^l \) and there exists an \( \omega \in \Omega \) such that the second stage optimal solution \( y^*(\omega, x^l) \) obtained by solving SLP\((\omega, x^l)\) does not belongs to \( K(\omega, x^l) \). This results in a non-terminating loop, referred to as cycling. Therefore, to avoid such cycling, we add a preventive measure in Algorithm 1, i.e., Lines 18-23. Notice that whenever \( x^{l+1} = x^l \), \( LB \) or \( c^T x^l + \sum_{\omega \in \Omega} p_\omega Q_{LP}(\omega, x^l) \) gives the optimal objective value for the TSS-DP. Although \((x^l, \{y^*(\omega, x^l)\}_{\omega \in \Omega}) \) does not belong to \( \mathcal{P} \), there exist feasible solutions \((\tilde{x}^1, \{\tilde{y}^1(\omega)\}_{\omega \in \Omega}) \in \mathcal{P} \) and \((\tilde{x}^2, \{\tilde{y}^2(\omega)\}_{\omega \in \Omega}) \in \mathcal{P} \) such that \((x^l, \{y^*(\omega, x^l)\}_{\omega \in \Omega}) = \lambda (\tilde{x}^1, \{\tilde{y}^1(\omega)\}_{\omega \in \Omega}) + (1 - \lambda)(\tilde{x}^2, \{\tilde{y}^2(\omega)\}_{\omega \in \Omega}) \) for \( \lambda \in (0, 1) \). Let \((x^l, \theta^l = \sum_{\omega \in \Omega} p_\omega g(\omega)^T y^*(\omega, x^l)) \) be the optimal solution of \( M_l \). Since \( M_l \) is a lower bound approximation, both \((\tilde{x}^1, \tilde{\theta}^1 = \sum_{\omega \in \Omega} p_\omega g(\omega)^T \tilde{y}^1(\omega)) \) and \((\tilde{x}^2, \tilde{\theta}^2 = \sum_{\omega \in \Omega} p_\omega g(\omega)^T \tilde{y}^2(\omega)) \) are feasible solutions of \( M_l \). Observe that \((x^l, \theta^l) = \lambda (\tilde{x}^1, \tilde{\theta}^1) + (1 - \lambda)(\tilde{x}^2, \tilde{\theta}^2) \) for \( \lambda \in (0, 1) \). Therefore, \((\tilde{x}^1, \tilde{\theta}^1) \) and \((\tilde{x}^2, \tilde{\theta}^2) \) also belong to the set of alternative optimal solutions of \( M_l \). We denote the set of \( x \) components of all alternative optimal solutions of \( M_l \) by \( X^l \subseteq X \). Now, for each \( \hat{x} \in X \setminus \{x^{l+1}\} \), we solve subproblems SLP\((\omega, \hat{x}) \), \( \omega \in \Omega \), and find a feasible solution which is also the optimal solution for \( (20) \) (Lines 19-22).

**Corollary 6.** Given \((\omega, \hat{x}) \in (\Omega, X)\), \( Q_{LP}(\omega, \hat{x}) \leq Q(\omega, \hat{x}) \). If \( \hat{x} \in \text{ver}(X) \cup \{x^*\} \) where \( \text{ver}(X) \) is the set of vertices of \( X \) and \((x^*, y^*) \) is the optimal solution of \((20)\), then there exists a solution \( \hat{y}(\omega) \in K(\omega, \hat{x}) \) such that \( \hat{y}(\omega) \) is the optimal solution for \( Q_{LP}(\omega, \hat{x}) \) and \( Q_{LP}(\omega, \hat{x}) = Q(\omega, \hat{x}) \).

**Proof.** Let \((\omega, \hat{x}) \in (\Omega, X)\). Observe that in SLP\((\omega, \hat{x})\), the set of feasible solutions \( K_{\text{tight}}(\omega, \hat{x}) \) is equivalent to the projection of \( P_{\text{tight}}(\omega) \), defined by \((26)\), on \((x = \hat{x}, y(\omega)) \) space. Since \( \text{conv}(P(\omega)) = \text{Proj}_{x,y(\omega)}(P_{\text{tight}}(\omega)) \), using Theorem 2, we know that \( \text{conv}(K(\omega, \hat{x})) \subseteq K_{\text{tight}}(\omega, \hat{x}) \). This implies that \( Q_{LP}(\omega, \hat{x}) \leq Q(\omega, \hat{x}) \). Now, if \( \hat{x} \) is a vertex of \( X \) then again according to Theorem 2, \( \text{conv}(K(\omega, \hat{x})) = K_{\text{tight}}(\omega, \hat{x}) \). This means that there exists a solution \( \hat{y}(\omega) \in K(\omega, \hat{x}) \) such that \( \hat{y}(\omega) \) is the optimal solution for \( Q_{LP}(\omega, \hat{x}) \) and \( Q_{LP}(\omega, \hat{x}) = Q(\omega, \hat{x}) \); and this last statement is also true for \( \hat{x} = x^* \) because of Proposition 1.

**Theorem 10** (Convergence Result). Algorithm 1 solves the TSS-DP \((20)\) to optimality in finitely many iterations if assumptions (iii)-(iv), defined in Section 1, are satisfied and \(|X|\) is finite.

**Proof.** We assume that \(|X|\) is finite and therefore, the number of first stage feasible solutions \( X \) is also finite. In Algorithm 1, for a given \( x^l \in X \), we solve \(|\Omega|\) number of linear programs, i.e. SLP\((\omega, x^l)\) for all \( \omega \in \Omega \), and if needed, a master problem \( M_l \) (after adding an optimality cut). Notice that the master problem has disjunctive constraints where the set \( S \) is finite (see \((20)\)), and also \( \Omega \) is finite. Therefore, Lines 3-17 in Algorithm 1 are performed in finite iterations. Based on Corollary 6, it is clear that if \((x^*, y^*) \) is the optimal solution of the TSS-DP \((20)\), then for \( x^l = x^* \), Algorithm 1 returns the optimal solution for the problem in finite iterations as \(|X|\) is
finite. Furthermore, in case of alternative optimal solutions, because of Lines 18-23, cycling does not occur in this algorithm and it terminates after finite iterations with the optimal solution as \(|\bar{X}^t|\) is also finite.

In case the disjunctive set \(X\) is super-facial then for all \(x^l \in X\), the convex hull of \(\mathcal{K}(\omega, x^l)\), \(\omega \in \Omega\), is given by \(\text{Proj}_y(\omega)(\mathcal{K}_{\text{tight}}(\omega, x^l))\). Therefore, solving \(\text{SP}(\omega, x^l)\) in Line 4 of Algorithm 1 gives the optimal solution \(y^*(\omega, x^l) \in \mathcal{K}(\omega, x^l)\). As a result, cycling does not occur is such cases. As mentioned before, first stage program with pure 0-1 variables is one such example. Hence, the algorithms developed by Gade et al. [21] and Sherali and Fraticelli [37] for TSSP with \(X := \{0, 1\}^P\), where \(P(\omega)\) is convexified after finite iterations using parametric Gomory cuts and reformulation-linearization technique, respectively, do not face the concerns of cycling. But the algorithms developed for TSSPs with \(X := \mathbb{Z}^P\) (or \(\mathbb{Z} \times \mathbb{Z}_{+}^{p-1}\)), where parametric inequalities are added either a priori or in succession, may not converge after finite iterations until the measures to avoid cycling are incorporated. We explain this claim using Example 1 where \(|\Omega| = 1\) and \(P(= P(\omega))\) is integral.

**Example 1** (continued). Let \(c = -3\) and \(g = 2\). Then the optimal solution for problem (7)-(13) is \((2,1)\) or \((4,4)\) with optimal objective value equal to \(-4\). While solving this problem using a decomposition algorithm similar to Algorithm 1, assume that \(x^l = 3\) at some iteration \(l\). Since \(P\) is integral, there is no need to add any parametric inequality, or in other words, addition of any type of parametric inequality will be redundant. Now, solving the second stage problem for \(x^l = 3\) and the feasible region \(\mathcal{K}_{\text{tight}}(x^l) = \{y \in \mathbb{R}_+ : 5/2 \leq y \leq 13/3\}\) gives the optimal solution \(y^*(x^l) = 5/2\) and optimal objective value \(c x^l + g y^*(x^l) = -4\). Since lower bound cannot be further improved, addition of Benders’ cut to the master problem \(M_l\) does not cut \(x^l\) and hence, resolving the updated masters problem \(M_{l+1}\) gives \(x^{l+1} = x^l = 3\). This gives rise to a non-terminating loop, i.e. cycling in the absence of any preventive measure. However, using Lines 18-23 in Algorithm 1, we can prevent the cycling as follows. Let \(\bar{X}^l\) be the set of \(x\) components of all alternative optimal solutions of \(M_l\), i.e. \(\bar{X}^l := \{2, 3, 4\}\). Now, we solve the second stage problem for \(x^l = 2\) and the feasible region \(\mathcal{K}_{\text{tight}}(2) = \{y \in \mathbb{R}_+ : 1 \leq y \leq 14/3\}\) and get the optimal solution \(y^*(x^l) = 1\). Since \((2,1) \in \mathcal{P}\), Algorithm 1 updates the UB to \(-4\) and the algorithm terminates because UB have become equal to LB.

**Remark 6.** Zhang and Kücükyavuz [40] extend the algorithm of Gade et al. [21] for solving TSS-MIPs with pure integer variables in the both stages. In their algorithms, the parametric inequalities are added in succession and the master problems are solved by adding finite number of Gomory cut and then solving the linear program using the lexicographic dual simplex method. As a result, at each iteration \(l\), \(x^l\) is the extreme point of the associated polyhedron and in case of alternative optimal solutions, their algorithms consider lexicographically smallest solution. This prevents cycling in their algorithms and hence, provide finitely convergent algorithms.

**Remark 7.** It is important to note that instead of solving the master problem (29) to optimality at each iteration, a branch-and-cut approach can also be adopted for a practical implementation. In this approach, similar to the integer L-shaped method [26], a linear programming relaxation of the master problem is solved. The solution thus obtained is used to either generate a feasibility cut (if this current solution violates any of the relaxed constraints), or create new branches/nodes following the usual branch-and-cut procedure. The (globally valid) optimality cut, \(OC(x)\), is derived at a node whenever the current solution is also feasible for the original master problem. Interestingly, because of the finiteness of the branch-and-bound approach, it is easy to prove the finite convergence
of this algorithm under the conditions mentioned in Theorem 10.

4.2 Decomposition algorithm for TSS-DP with sequentially convexifiable DPs in the second stage

We present a decomposition algorithm to solve TSS-DPs with sequentially convexifiable DPs in the second stage, which we refer to as TSS-SC-DPs, by harnessing the benefits of sequential convexification property within L-shaped method. As mentioned before, Balas [3, 4] introduced this property of a subclass of DPs, sequentially convexifiable DPs, according to which the convex hull of a set of points satisfying multiple disjunctive constraints, where each disjunction contains exactly one inequality, can be derived by sequentially generating the convex hull of points satisfying only one disjunctive constraint. In [3, 4], Balas shows that the facial DPs are sequentially convexifiable and later, Balas et al. [7] extend the sequential convexification property for a general non-convex set with multiple constraints. They provide the necessary and sufficient conditions under which reverse convex programs (DPs with infinitely many terms) are sequentially convexifiable and present classes of problems, in addition to facial DPs, which always satisfy the sequential convexification property. In the light of this discussion, it is clear that our algorithm for TSS-SC-DPs is capable to solve various classes of TSSPs.

The pseudocode of our decomposition algorithm is presented in Algorithm 2. In contrast to Algorithm 1 where we used a priori convexification approach, in Algorithm 2 we add “parametric” cuts in a successive fashion using sequential convexification approach of Balas [3, 4]. Similar to the definitions in Section 4.1, we denote the lower bound and upper bound on the optimal objective value of a given TSS-SC-DP by $LB$ and $UB$, respectively. We define subproblem $SP(\omega, x)$, $(\omega, x) \in (\Omega, X)$ as follows:

$$Q_{SP}(\omega, x) := \min g(\omega)^T y(\omega)$$

$$s.t. \ W(\omega)y(\omega) \geq r(\omega) - T(\omega)x$$

$$\alpha_t(\omega)y(\omega) \geq \beta_t(\omega) - \psi_t(\omega)x, \quad t = 1, \ldots, \tau(\omega)$$

$$y(\omega) \in \mathbb{R}^q,$$

where $\alpha_t(\omega) \in \mathbb{R}^q$, $\psi_t(\omega) \in \mathbb{R}^p$, and $\beta_t(\omega) \in \mathbb{R}$ are the coefficients of variables $y(\omega)$, coefficients of variables $x$, and the right hand side, respectively, of a parametric inequality, referred to as the parametric lift-and-project cut. In Remark 8, we discuss how these parametric inequalities are developed in succession using sequential convexification approach of Balas [3, 4]. Now, let $\pi^*(\omega, x) = (\pi^*_0(\omega, x), \pi^*_1(\omega, x), \ldots, \pi^*_\tau(\omega)(\omega, x))^T$ be the optimal dual multipliers obtained by solving $SP(\omega, x)$ for a given $(\omega, x) \in (\Omega, X)$. Then, the corresponding optimality cut, $OCS(x)$, is

$$\sum_{\omega \in \Omega} p_\omega \left\{ \pi^*_0(\omega, x)^T (r(\omega) - T(\omega)x) + \sum_{t=1}^{\tau(\omega)} \pi^*_t(\omega, x)(\beta_t(\omega) - \psi_t(\omega)x) \right\} \geq \theta.$$  \hspace{1cm} (31)

These cuts help in deriving a lower bounding approximation of the first stage problem (20), defined by $\min\{c^T x + \theta : x \in X$ and $OCS(x^k)$ holds, for $k = 1, \ldots, l\}$ where $x^k \in X$ for $k = 1, \ldots, l$. We denote this problem by $M_l$ for $l \in \mathbb{Z}_+$ and refer to it as the master problem. Note that $M_0$ is the master problem without any optimality cut. For the sake of convenience, we assume that the TSS-SC-DP solved using Algorithm 2 does not have alternative optimal solutions, but in Remark 10, we provide a preventive measure to avoid cycling which can happen while solving TSS-SC-DPs with alternative optimal solutions.
Algorithm 2 Algorithm for TSS-SC-DPs using Lift-and-Project Cuts

1: Initialization: $l \leftarrow 1$, $LB \leftarrow -\infty$, $UB \leftarrow \infty$, $\tau(\omega) \leftarrow 0$ for all $\omega \in \Omega$. Assume $x^1 \in X$.
2: while $UB - LB > \epsilon$ do \hfill $\triangleright \epsilon$ is a pre-specified tolerance
3:     for $\omega \in \Omega$ do
4:         Solve linear program $SP(\omega, x^l)$;
5:         $y^*(\omega, x^l) \leftarrow$ optimal solution; $Q_{SP}(\omega, x^l) \leftarrow$ optimal solution value;
6:     end for
7:     if $y^*(\omega, x^l) \notin K(\omega, x^l)$ for some $\omega \in \Omega$ then
8:         for $\omega \in \Omega$ where $y^*(\omega, x^l) \notin K(\omega, x^l)$ do \hfill $\triangleright$ Add parametric inequalities
9:             Add the lift-and-project cut to $SP(\omega, x)$ as explained in Remark 8;
10:        end if
11:  end if
12:  if $y^*(\omega, x^l) \in K(\omega, x^l)$ for all $\omega \in \Omega$ and $UB > c^T x^l + \sum_{\omega \in \Omega} p_\omega Q_{SP}(\omega, x^l)$ then
13:      $UB \leftarrow c^T x^l + \sum_{\omega \in \Omega} p_\omega Q_{SP}(\omega, x^l)$;
14:      if $UB \leq LB + \epsilon$ then
15:          Go to Line 27;
16:  end if
17: end if
18: end if
19: $\pi^*(\omega, x^l) \leftarrow$ optimal dual multipliers obtained by solving $SP(\omega, x^l)$ for all $\omega \in \Omega$;
20: Derive optimality cut $OCS(x^l)$ using (31);
21: Add $OCS(x^l)$ to $M_{l-1}$ to get $M_l$;
22: Solve master problem $M_l$;
23: $x^{l+1} \leftarrow$ optimal solution; $LB \leftarrow$ optimal solution value;
24: $l \leftarrow l + 1$;
25: end while
26: return $\{x^1, \{y(\omega, x^l)\}_{\omega \in \Omega}\}, UB$

Similar to Algorithm 1, we initialize Algorithm 2 by setting lower bound $LB$ to negative infinity, upper bound $UB$ to positive infinity, iteration counter $l$ to 1, number of parametric inequalities $\tau(\omega)$ for all $\omega \in \Omega$ to zero, and selecting a first stage feasible solution $x^1 \in X$ (Line 1). At each iteration $l \geq 1$, we solve linear programs $SP(\omega, x^l)$ for all $\omega \in \Omega$ and store the corresponding optimal solution $y^*(\omega, x^l)$ and the optimal solution value $Q_{SP}(\omega, x^l) := g(\omega)^T y^*(\omega, x^l)$ for each $\omega \in \Omega$ (Lines 3-6). Now, for each $\omega \in \Omega$ with $y^*(\omega, x^l) \notin K(\omega, x^l)$, we develop parametric lift-and-project cut (as explained in Remark 8), add it to $SP(\omega, x)$, resolve the updated subproblem $SP(\omega, x)$ by fixing $x = x^l$, and obtain its optimal solution $y^*(\omega, x^l)$ along with optimal solution value (Lines 8-12). Interestingly, in case $y^*(\omega, x^l) \in K(\omega, x^l)$ for all $\omega \in \Omega$, we have a feasible solution $(x^l, y^*(\omega, x^l), \ldots, y^*(\omega_{|\Omega|}, x^l))$ for the original problem. Therefore, we update $UB$ if the solution value corresponding to thus obtained feasible solution is smaller than the existing upper bound (Lines 14-15). We also utilize the stored information and optimal dual multipliers (Line 20) to derive optimality cut $OCS(x^l)$ using (31) and add this cut to the master problem $M_{l-1}$ to get an augmented master problem $M_l$ (Lines 21-22). We solve $M_l$ and since it is lower bounding approximation of (20), we use the optimal solution value associated with $M_l$ to update $LB$. It is important to note that the lower bound $LB$ is a non-increasing function with respect to the iterations. This is because $M_{l-1}$ is a relaxation of $M_l$ for each $l \geq 1$. Therefore, after every iteration the difference between the bounds, $UB - LB$, either decreases or remains same as in
previous iteration. We terminate our algorithm when this difference becomes zero, i.e., $UB = LB$, or reaches a pre-specified tolerance $\epsilon$ (Line 2 or Lines 16-18), and return the optimal solution $(x^l, \{y(\omega, x^l)\}_{\omega \in \Omega})$ and the optimal solution value $UB$.

**Remark 8.** Here we present how a "parametric" lift-and-project cut of the form $\alpha_t(\omega)y(\omega) \geq \beta_t(\omega) - \psi_t(\omega)x$ where $t = \tau(\omega) + 1$, is generated in Algorithm 2 (Line 9). Given a first stage feasible solution $x^l$ at iteration $l$, assume that there exists an $\tilde{\omega} \in \Omega$ such that the optimal solution of $SP(\tilde{\omega}, x^l)$, i.e. $y^*(\tilde{\omega}, x^l)$, does not belong to $\mathcal{K}(\tilde{\omega}, x^l)$. This implies that there exists a disjunctive constraint, $\bigvee_{i \in H_j} (\eta_i(\tilde{\omega})y(\tilde{\omega}) \geq \eta_0(\tilde{\omega}) + \xi(\tilde{\omega})x)$, $j \in \{1, \ldots, m\}$, which is not satisfied by the point $(x^l, \{y^*(\omega, x^l)\}_{\omega \in \Omega})$. In order to generate an inequality which cuts this point, we first use Theorem 4 to get a tight extended formulation for the closed convex hull of

\begin{align}
W(\tilde{\omega})y(\tilde{\omega}) + T(\tilde{\omega})x & \geq r(\tilde{\omega}) \quad \text{(32)} \\
\alpha_t(\tilde{\omega})y(\tilde{\omega}) + \psi_t(\tilde{\omega})x & \geq \beta_t(\tilde{\omega}), \ t = 1, \ldots, \tau(\tilde{\omega}) \quad \text{(33)} \\
\bigvee_{i \in H_j} (\eta_i(\tilde{\omega})y(\tilde{\omega}) + \eta_0(\tilde{\omega})x \geq \eta_0(\tilde{\omega})) & \quad \text{(34)} \\
x \in X, y(\tilde{\omega}) & \in \mathbb{R}^q \quad \text{(35)}
\end{align}

where $|H_j| = |H|$. Then, we project this tight extended formulation in the lifted space to the $(x, y(\tilde{\omega}))$ space using Theorems 5 and 6. Let $\mathcal{P}_j(\tilde{\omega}) := \{(32)-(35)\}$ and its linear programming equivalent in the lifted space be given by

$$
\mathcal{P}_{\text{tight}}^j(\tilde{\omega}) := \left\{ \sum_{i \in H_j} \xi_1^i(\tilde{\omega}) - y(\tilde{\omega}) = 0, \right. \\
\left. \sum_{i \in H_j} \xi_2^i(\tilde{\omega}) - x = 0, \right. \\
W(\tilde{\omega})\xi_1^i(\tilde{\omega}) + T(\tilde{\omega})\xi_2^i(\tilde{\omega}) \geq r(\tilde{\omega})\xi_0^i(\tilde{\omega}), \ i \in H_j, \\
\alpha_t(\tilde{\omega})\xi_1^i(\tilde{\omega}) + \psi_t(\tilde{\omega})\xi_2^i(\tilde{\omega}) \geq \beta_t(\tilde{\omega})\xi_0^i(\tilde{\omega}), \ i \in H_j, t = 1, \ldots, \tau(\tilde{\omega}), \\
\eta_i(\tilde{\omega})\xi_1^i(\tilde{\omega}) + \eta_0(\tilde{\omega})\xi_2^i(\tilde{\omega}) \geq \eta_0(\tilde{\omega})\xi_0^i(\tilde{\omega}), \ i \in H_j, \\
\sum_{i \in H_j} \xi_0^i(\tilde{\omega}) = 1, \\
x \in X, y(\tilde{\omega}) \in \mathbb{R}^q, \\
\xi_1^i(\tilde{\omega}) \in \mathbb{R}_+, \xi_2^i(\tilde{\omega}) \in \mathbb{R}_+, \xi_0^i(\tilde{\omega}) \in \mathbb{R}_+, \ i \in H_j \right\}.
$$

Using Theorem 5, we derive the projection of $\mathcal{P}_{\text{tight}}^j(\tilde{\omega})$ onto the $(x, y(\tilde{\omega}))$ space, i.e. $\Proj_{x,y(\tilde{\omega})}(\mathcal{P}_{\text{tight}}^j(\tilde{\omega}))$, which is given by

$$
\{(x, y(\tilde{\omega})) \in \mathbb{R}^p \times \mathbb{R}^q : \alpha y(\tilde{\omega}) + \psi x \geq \beta \text{ for all } (\alpha, \psi, \beta) \in \mathcal{C}_j(\tilde{\omega})\} \quad \text{(36)}
$$
where

\[ C_j(\omega) := \left\{ (\alpha, \psi, \beta) \in \mathbb{R}^p \times \mathbb{R}^q \times \mathbb{R}^r : \right. \]

\[ \alpha = \sigma^i \left( W(\omega) \right), \quad \alpha = \sigma^i \alpha_t(\omega), \quad i \in H_j, t = 1, \ldots, \tau(\omega) \]

\[ \psi = \sigma^i \left( T(\omega) \right), \quad \psi = \sigma^i \psi_t(\omega), \quad i \in H_j, t = 1, \ldots, \tau(\omega) \]

\[ \beta = \sigma^i \left( r(\omega) \right), \quad \beta = \sigma^i \beta_t(\omega), \quad i \in H_j, t = 1, \ldots, \tau(\omega) \]

\[ \text{for some } (\sigma^i, \sigma^i_c) \in \mathbb{R}^{m+1}_+ \times \mathbb{R}^{\tau(\omega)}_+, \quad i \in H_j \right\}. \]

Next, we solve the following cut-generating linear program (CGLP) to find the most violated parametric lift-and-project cut among the defining inequalities of (36) for \((x^l, y^*(\omega, x^l))\):

\[
\max \{ \beta - \alpha y^*(\omega, x^l) - \psi x^l : (\alpha, \psi, \beta) \in C_j(\omega) \cap \mathcal{N}(\omega) \},
\]  \hspace{1cm} (37)

where \(\mathcal{N}(\omega)\) is a normalization set (defined by one or more constraints) which truncate the cone \(C_j(\omega)\). Let \((\alpha^*, \psi^*, \beta^*)\) be the optimal solution for (37). Then, for \(t = \tau(\omega) + 1\), we set \(\alpha_t(\omega) = \alpha^*\), \(\psi_t(\omega) = \psi^*\), and \(\beta_t(\omega) = \beta^*\) to get the required parametric lift-and-project cut in Line 9 of Algorithm 2.

**Theorem 11** (Convergence result). Algorithm 2 solves TSS-DP (20) with facial DPs in the second stage programs to optimality in finitely many iterations if assumptions (iii)-(iv), defined in Section 1, are satisfied and \(|X|\) is finite.

**Proof.** We assume that \(|X|\) is finite and therefore, the number of first stage feasible solutions \(X\) is also finite. In Algorithm 2, for a given \(x^l \in X\), we solve \(|\Omega|\) number of linear programs, i.e. SP(\(\omega, x^l\)) for all \(\omega \in \Omega\), and if needed, a master problem \(M_t\) (after adding an optimality cut which requires a linear program to be solved). Notice that the master problem has disjunctive constraints where the set \(S\) is finite (see (20)), and also \(\Omega\) is finite. Therefore, Lines 3-25 in Algorithm 2 are performed in finite iterations. Now we have to ensure that the “while” loop in Line 2 terminates after finite iterations and provide the optimal solution. Notice that if \(x^{l+1} \neq x^l\), then \(x^l\) will not be visited again in future iterations because the optimality cut generated in Line 21 cuts \(x^l\).

Another possibility is \(x^{l+1} = x^l\) which can further be divided into two cases. In first case, we assume that \(y^*(\omega, x^l) \in \mathcal{K}(\omega, x^l)\) for all \(\omega \in \Omega\), therefore the lower bound \(LB = c^T x^l + \sum_{\omega \in \Omega} p_\omega g^T(\omega)y^*(\omega, x^l)\) which is equal to \(UB\) as \((x^l, \{y^*(\omega, x^l)\}_{\omega \in \Omega}) \in \mathcal{P}\) is a feasible solution. This implies that \((x^l, \{y^*(\omega, x^l)\}_{\omega \in \Omega})\) is the optimal solution and hence, in this case the algorithm terminates after returning the optimal solution and optimal objective value \(UB\). For the second case, we assume that for some \(\omega \in \Omega\), \(y^*(\omega, x^l) \notin \mathcal{K}(\omega, x^l)\). Based on the results of Jeroslow [22], we know that the addition of finite number of parametric lift-and-projects cuts (Line 9 or Remark 8) can provide \(\mathcal{P}_{\text{half}}(\omega)\). Because of Proposition 1, it is clear that if \((x^*, y^*)\) is the optimal solution of TSS-SC-DP, then for \(x^l = x^*\), Algorithm 2 returns the optimal solution for the problem in finite iterations. This completes the proof. ☐
Remark 9. Algorithm 2 generalizes the algorithm developed in [37, 38] for TSSPs with pure binary variables in first stage and mixed 0-1 programs in the second stage, which utilizes the reformulation-linearization technique (RLT) of Sherali and Adams [35, 36] and lift-and-project cuts of Balas et al. [5].

Remark 10. It is important to note that while solving a TSS-SC-DP with alternative optimal solution using Algorithm 2, it is possible that for a given first stage feasible solution $x^l$ at iteration $l$, no parametric lift-and-project cut is added to SP($\omega, x$) in Line 9 and solving $M_l$ gives $x^{l+1} = x^l$ in Line 24. This will result in a non-terminating loop or cycling. In order to avoid cycling, we provide a preventive measure by incorporating the following modifications in Algorithm 2: Similar to Algorithm 1, in such situation we first store the $x$ components of all alternative optimal solutions of $M_l$, denoted by $\bar{X}_l \subseteq X$. Then, for each $\hat{x} \in X \backslash \{x^{l+1}\}$, we solve subproblems SP($\omega, \hat{x}$), $\omega \in \Omega$, and find a feasible solution belonging to $P$ which is also the optimal solution for the TSS-SC-DP.

5. Two-stage stochastic semi-continuous programs (TSS-SCPs)

In this section, we study two-stage stochastic semi-continuous programs (TSS-SCPs) where second stage has semi-continuous variables, i.e. TSSP (1) where

$$Q(\omega, x) := \min g(\omega)^T y(\omega)$$

s.t. $W(\omega) y(\omega) \geq r(\omega) - T(\omega) x$

$$y_i(\omega) \in [0, \bar{l}_i(\omega)] \cup [\bar{l}_i(\omega), u_i(\omega)], i = 1, \ldots, q_1,$$

$$y_i(\omega) \geq 0, i = q_1 + 1, \ldots, q,$$

such that $0 \leq \bar{l}_i(\omega) < \bar{l}_i(\omega) \leq u_i(\omega)$ for $i = 1, \ldots, q_1$ and $\omega \in \Omega$. Note that by setting $\bar{l}_i(\omega) = \bar{l}_i(\omega)$ for all $i$ and $\omega$, the semi-continuous variables become continuous, and by setting $\bar{l}_i(\omega) = 0$ and $\bar{l}_i(\omega) = u_i(\omega) = 1$, the semi-continuous variables become binary. Therefore, the TSSPs with mixed 0-1 programs in the second stage are special cases of TSS-SCPs.

5.1 Example of convexifiable TSS-SCP

First, we demonstrate the application of our convexification approach (discussed in Section 2) to solve TSS-SCPs, in particular a relaxation of two-stage stochastic semi-continuous network flow problem. In [2], Angulo et al. study the semi-continuous inflow set, defined by

$$S(t, h) := \{(y, z) \in \mathbb{R}^n \times \mathbb{R}^n : \sum_{i \in N} y_i \geq d$$

$$t_i + z_i \geq y_i \quad i \in N,$$

$$y_i \in \{0\} \cup [h_i, \infty) \quad i \in N,$$

$$z_i \in \{0\} \cup [l_i, \infty) \quad i \in N\},$$
where \( N := \{1, \ldots, n\} \), and provide a tight and compact extended formulation for \( S(0, h) \), given as follows:

\[
S_{\text{tight}} := \{(y, z, u) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{|L|} : \]
\[
\sum_{i \in N \setminus L} \frac{y_i}{\max\{d, h_i\}} + \sum_{i \in L} u_i \geq 1
\]
\[
\frac{z_i}{l_i} \geq u_i \quad i \in L
\]
\[
\frac{y_i}{\max\{d, h_i\}} \geq u_i \quad i \in L
\]
\[
y_i \geq 0 \quad i \in N
\]
\[
z_i \geq y_i \quad i \in N
\]

where \( L := \{i \in N : \max\{d, h_i\} < l_i\} \). They show that \( S(t, h) \) arises as substructure in general semi-continuous network flow problem and semi-continuous transportation problem. It is important to remember that the approach of adding auxiliary binary variables and constraints to reformulate a SCP can result in a large scale MIP. In [17, 18, 19, 20], attempts have been made to overcome the difficulties with the auxiliary binary variables. Here, we consider the following TSS-SCP with semi-continuous inflow set in the second stage:

\[
\min c^T x + \mathbb{E}_\omega[Q(\omega, x)]
\]
\[
s.t. \ Ax \geq b
\]
\[
x \in \mathcal{X}
\]

where for any scenario \( \omega \in \Omega \),

\[
Q(\omega, x) := \min g_y(\omega)y(\omega) + g_z(\omega)z(\omega)
\]

\[
s.t. \ \sum_{i \in N} y_i(\omega) \geq d(\omega)
\]
\[
y_i(\omega) - z_i(\omega) \leq 0 \quad i \in N
\]
\[
x_i \leq d(\omega) \quad i \in N
\]
\[
y_i(\omega) \in \{0\} \cup [x_i, \infty) \quad i \in N
\]
\[
z_i(\omega) \in \{0\} \cup [l_i(\omega), \infty) \quad i \in N
\]

Here, \( g_y(\omega) \in \mathbb{R}^n, g_z(\omega) \in \mathbb{R}^n, \) and \( l(\omega) \in \mathbb{R}^n_+ \). Similar to TSSP, we also assume that \( \mathcal{X} \subseteq \mathbb{R}^p \) is a general set, \( X := \{x : Ax \geq b, x \in \mathcal{X}\} \) and \( K(\omega, x) := \{(y(\omega), z(\omega)) : (44)-(48) \) hold\} such that the assumptions (ii)-(iv), defined in Section 1, hold.
Theorem 12. For each \((x, \omega) \in (X, \Omega)\),

\[
\mathcal{K}_{\text{tight}}(\omega, x) := \{ (y(\omega), z(\omega), u(\omega)) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{|L(\omega)|} : \\
\frac{z_i(\omega)}{l_i(\omega)} \geq u_i(\omega) \quad i \in L(\omega) \\
\frac{y_i(\omega)}{d(\omega)} \geq u_i(\omega) \quad i \in L(\omega) \\
x_i \leq d(\omega) \quad i \in N \\
y_i(\omega) \geq 0 \quad i \in N \\
z_i(\omega) \geq y_i(\omega) \quad i \in N \\
\sum_{i \in N \setminus L(\omega)} \frac{y_i(\omega)}{d(\omega)} + \sum_{i \in L(\omega)} u_i(\omega) \geq 1 \},
\]

where \(L(\omega) := \{ i \in N : d(\omega) < l_i(\omega) \}\), is a tight extended formulation of \(\mathcal{K}(\omega, x)\), i.e. \(\text{conv}(\mathcal{K}(\omega, x)) = \text{Proj}_{y(\omega), z(\omega)}(\mathcal{K}_{\text{tight}}(\omega, x))\).

5.2 Linear programming equivalent for the second stage of TSS-SCPs

We re-write the semi-continuity constraints (40) as disjunctive constraints,

\[
\bigwedge_{i=1}^{q_1} \left\{ (0 \leq y_i(\omega) \leq l_i(\omega)) \lor (\bar{l}_i(\omega) \leq y_i(\omega) \leq \bar{u}_i(\omega)) \right\}, \quad (49)
\]

thereby, showing that the class of TSS-DPs subsumes the TSS-SCPs. Next, in order to convexify \(\mathcal{K}(\omega, x)\), \((\omega, x) \in \Omega \times X\), for the TSS-SCP, we assume that \(q_1 = 2\) (for the sake of convenience) and hence, the constraints (40) or (49) in the disjunctive normal form is given by:

\[
\begin{align*}
&\left(0 \leq y_1(\omega) \leq l_1(\omega)\right) \lor \left(0 \leq y_1(\omega) \leq l_1(\omega)\right) \\
&\lor \left(\bar{l}_1(\omega) \leq y_1(\omega) \leq \bar{u}_1(\omega)\right) \lor \left(\bar{l}_1(\omega) \leq y_1(\omega) \leq \bar{u}_1(\omega)\right)
\end{align*}
\]

\[
\lor \left(0 \leq y_2(\omega) \leq l_2(\omega)\right) \lor \left(0 \leq y_2(\omega) \leq l_2(\omega)\right) \\
\lor \left(\bar{l}_2(\omega) \leq y_2(\omega) \leq \bar{u}_2(\omega)\right) \lor \left(\bar{l}_2(\omega) \leq y_2(\omega) \leq \bar{u}_2(\omega)\right).
\]

(50)

Corollary 7. Assuming \(q_1 = 2\), if \(X\) is a super-facial disjunctive set then a tight extended formulation for \(\mathcal{K}(\omega, x) := \{ (y(\omega) : (39)-(41) \text{ hold}) \}, (\omega, x) \in (\Omega, X)\), is given by
\[ K_{tight}(\omega, x) := \left\{ \begin{array}{l}
\sum_{h \in H} \xi_{1,1}^h(\omega) - y_1(\omega) = 0 \\
\sum_{h \in H} \xi_{1,2}^h(\omega) - y_2(\omega) = 0 \\
\sum_{h \in H} \xi_{2}^h(\omega) = x \\
W(\omega)(\xi_{1,1}^h(\omega), \xi_{1,2}^h(\omega)) \geq r(\omega)\xi_0^h(\omega) - T(\omega)\xi_2^h(\omega), \quad h \in H \\
D_{1,1}^h\xi_{1,1}^h(\omega) + D_{1,2}^h\xi_{1,2}^h(\omega) \leq d_0^h(\omega)\xi_0^h(\omega), \quad h \in H \\
\sum_{h \in H} \xi_0^h(\omega) = 1 \\
y(\omega) \in \mathbb{R}^q, \xi_{1,1}^h(\omega) \in \mathbb{R}_+, \xi_{1,2}^h(\omega) \in \mathbb{R}_+, \xi_0^h(\omega) \in \mathbb{R}_+, \ h \in H \right\}. \]

where \( H := \{1, 2, 3, 4\} \), \( D_1^{h,1} = [-1 \ 0 \ 0 \ 1]^T \), \( D_1^{h,2} = [0 \ 0 \ -1 \ 1]^T \) for all \( h \in H \), and

\[
\begin{bmatrix}
    l_1(\omega) \\
    l_2(\omega)
\end{bmatrix},
\begin{bmatrix}
    l_1(\omega) \\
    l_2(\omega)
\end{bmatrix},
\begin{bmatrix}
    \bar{l}_1(\omega) \\
    \bar{l}_2(\omega)
\end{bmatrix},
\begin{bmatrix}
    \bar{u}_1(\omega) \\
    \bar{u}_2(\omega)
\end{bmatrix},
\begin{bmatrix}
    \bar{u}_1(\omega) \\
    \bar{u}_2(\omega)
\end{bmatrix}.
\]

6. Conclusion

We considered general two-stage stochastic programs and presented sufficient conditions under which the second stage programs can be convexified. This approach allowed us to relax the restrictions, such as integrality, binary, semi-continuity, and many others, on the second stage non-continuous variables in certain situations. We generalized the results of Bansal et al. [9] for two-stage stochastic mixed integer programs. We also introduced two-stage stochastic disjunctive programs (TSS-DPs) and extended the results of Balas [3, 4], developed for deterministic disjunctive programs, for TSS-DPs. More specifically, we provided linear programming equivalent for the second stage of TSS-DPs under certain conditions and developed a decomposition algorithm to solve general TSS-DPs using our convexification approach. By utilizing the sequential convexification approach of Balas [3] within L-shaped method, we developed another decomposition algorithm to solve TSS-DPs where second stage programs are facial DPs (in finite iterations), and sequentially convexifiable DPs (which include some non-convex programs such as general quadratic programs, separable non-linear programs, etc.). Furthermore, we showcased the significance of our convexification approach by solving two-stage stochastic semi-continuous programs (TSS-SCPs), in particular a TSS-SCP with semi-continuous inflow set in the second stage. We also presented a linear programming equivalent for the second stage of TSS-SCPs by formulating TSS-SCP as TSS-DP.

Acknowledgements. This work was supported by the grant ONR N00014-15-1-2226, which is gratefully acknowledged. We would like to thank Dr. Kuo-Ling Huang at Northwestern University for sharing his insight about the integer L-shaped method. Also, we sincerely thank Professor Minjiao Zhang at University of Alabama and Professor Simge Kıcıkyavuz at Ohio State University for discussing Remark 6 and mentioning about the usage of lexicographic dual simplex method in their algorithms in [40].
References


