Globally Converging Algorithm for Multistage Stochastic Mixed-Integer Programs via Enhanced Lagrangian Cuts

Abstract

This paper proposes a globally converging cutting-plane algorithm for solving *multistage stochastic mixed-integer programs* with general mixed-integer state variables. We demonstrate the generation process of Lagrangian cuts and show that Lagrangian cuts capture the convex envelope of value functions on a restricted region. To approximate nonconvex value functions to exactness, we propose to iteratively add binary state variables and generate Lagrangian cuts on the lifted state space. This lift-and-cut procedure converges to the global optimum with probability one. We further propose two types of *enhanced Lagrangian cuts*, Pareto-optimal Lagrangian cuts and square minimization cuts, to improve the computational performance of vanilla Lagrangian cuts. Extensive computational experiments on the generation expansion planning and stochastic unit commitment problems demonstrate the effectiveness and efficiency of our proposed methodology in solving large-scale, multistage stochastic mixed-integer programs.

1 Introduction

Multistage stochastic mixed-integer programming (MS-SMIP) models sequential decision-making under uncertainty with both continuous and integer decision variables. Decisions are made based on the realized values of random parameters and the system's current state while accounting for future uncertainty. MS-SMIP framework is widely applied in many practical problems, such as power system planning and operations (De Matos et al. 2017, Flatabø et al. 1998, Newham and Wood 2007, Rajagopalan et al. 1998, Takriti et al. 1996, Yang et al. 2024, Zou et al. 2018), transportation and logistics (Fan and Liu 2010, Listes and Dekker 2005, Wang and Jacquillat 2020), and healthcare scheduling (Gul et al. 2015, Guo et al. 2021).

We consider an MS-SMIP model (1) similar to those discussed in Romeijnders and van der Laan (2024), Zou et al. (2019). Decisions are made at each stage over a finite time horizon consisting of T stages, with uncertainty represented by a scenario tree $\mathcal{N}$. For each node $n \in \mathcal{N}$, we let $a(n)$ denote its unique ancestor node, and p_n be the probability of reaching node n. The root node of $\mathcal{N}$ is labeled as node 1, with initial condition $x_{a(1)} = 0$ and $p_1 = 1$. We let $\mathcal{N}_t$ represent the subset of nodes in stage $t = 1, \ldots, T$. At each node n, the local decision y_n is made based on its ancestor's state $x_{a(n)}$, and the state is updated to x_n for subsequent stages.

$$
\begin{aligned}
V^* = \min_{\{x_n, y_n\}_{n\in\mathcal{N}}} \quad & \sum_{n\in\mathcal{N}} p_n \left(c_n^\top x_n + g_n^\top y_n \right) && && \text{(1a)}\\
\text{s.t.} \quad & W_n x_n + B_n y_n = T_n x_{a(n)} + d_n && \forall\, n \in \mathcal{N} && \text{(1b)}\\
& x_n \in X_n, y_n \in Y_n && \forall\, n \in \mathcal{N}. && \text{(1c)}
\end{aligned}
$$

Model (1) minimizes the expected cost over the scenario tree $\mathcal{N}$. At each node n, the cost includes a linear term in the state variables x_n with cost vector c_n and a linear term in the decision variables

y_n with cost vector g_n. We use the subscript "j" to denote the j-th component of the vector x_n, written as $x_{n,j}$. The index set $\mathcal{J}_n$ represents the indices of x_n. The set X_n and Y_n model general mixed-integer constraints, which can include simple bounds or be extended to incorporate more complex linear constraints, i.e., rational MILP-representable.

We can also represent an MS-SMIP model in a recursive fashion with cost-to-go functions. The problem at the root node can be expressed as:

$$V^* = \min_{x_1, y_1} \quad c_1^\top x_1 + g_1^\top y_1 + \sum_{m \in \mathcal{C}(1)} q_{1m} Q_m(x_1) \tag{2a}$$

$$\text{s.t.} \quad W_1 x_1 + B_1 y_1 = d_1 \tag{2b}$$

$$x_1 \in X_1, y_1 \in Y_1, \tag{2c}$$

where $\mathcal{C}(n)$ represents the set of child nodes of some node $n \in \mathcal{N}$, $q_{nm} = p_m / p_n$ denotes the conditional probability of transitioning from node n to its child node $m \in \mathcal{C}(n)$, and $Q_m(\cdot)$ represents the optimal value function at node m. The recursive value function Q_n for any non-root node n is defined as:

$$Q_n(x_{a(n)}) = \min_{x_n, y_n} \quad c_n^\top x_n + g_n^\top y_n + \sum_{m \in \mathcal{C}(n)} q_{nm} Q_m(x_n) \tag{3a}$$

$$\text{s.t.} \quad W_n x_n + B_n y_n = T_n x_{a(n)} + d_n \tag{3b}$$

$$x_n \in X_n, y_n \in Y_n. \tag{3c}$$

For any leaf node $n \in \mathcal{N}_T$, we set $\mathcal{C}(n) = \emptyset$ and the cost-to-go function as zero.

1.1 Literature Review

Multi-stage stochastic programs are computationally challenging because the uncertainty model grows exponentially with the number of stages, and decisions at each stage depend on future uncertainties, forming a nested problem structure. To address those challenges, cutting-plane decomposition methods—such as stochastic nested decomposition (SND) (Birge 1985, Füllner and Rebennack 2022) and stochastic dual dynamic programming (SDDP) (Pereira and Pinto 1991)—are widely used for multi-stage stochastic convex programs. These paradigms generate linear cuts to approximate value functions, such that we can formulate decisions' dependence on future information as a piecewise linear convex function with a closed-form representation and obtain an optimal solution by solving a tractable convex program at each stage. The success of cutting-plane decomposition methods depends heavily on the convexity of each stage's problem.

However, once integer variables are introduced, such a convexity property no longer exists, as subproblems become nonconvex mixed-integer programs and value functions are now nonconvex and potentially discontinuous. Linear cutting planes may not be able to characterize value functions exactly. For problems with special structure, such as pure binary state variables, nonconvex value functions can be equivalently transformed to convex ones without introducing an optimality gap; e.g., integer L-shaped cuts (Angulo et al. 2016, Laporte and Louveaux 1993), combinatorial Benders cuts (Codato and Fischetti 2006), and logic-based Benders cuts (Guo et al. 2021, Hooker and Ottosson 2003). For general problem structures, previous work focuses on iterative convexification of the subproblem, including disjunctive programming (Sen and Higle 2005, Sen and Sherali 2006) and Fenchel cuts (Ntaimo 2013), which address pure integer state variables, and parametric Gomory cuts (Gade et al. 2014, Zhang and Küçükyavuz 2014), which incorporate state variables in the subproblem to tighten subproblem feasible regions. Scaled cuts (van der Laan and Romeijnders 2024) can achieve an asymptotic converging approximation for the expected value function via aggregating and transforming tight nonlinear cuts into linear ones.

Those convexification techniques often run into issues when extended to a multi-stage setting, because the nested problem structure may require complicated bookkeeping for all state variables across scenario paths (Romeijnders and van der Laan 2024). Again, for MS-SMIP with special problem structures, there are recent theoretical and algorithmic developments. The stochastic dual dynamic integer programming (SDDiP) algorithm (Zou et al. 2019) extends SDDP to cases with binary state variables by introducing local state variable copies and relaxing nonanticipativity constraints. With some monotonicity assumptions, the MIDAS approach (Philpott et al. 2016) uses step functions to approximate monotonic cost-to-go functions. Nonlinear cuts, such as augmented Lagrangian cuts (Ahmed et al. 2022, Füllner et al. 2024b, Rahmaniani et al. 2020) and generalized conjugacy cuts (Zhang and Sun 2022), provide theoretical convergence under specific Lipschitz continuity conditions. However, theory and methods are still not well-established for many large-scale practical problems under a general MS-SMIP setting.

That said, recent developments in Lagrangian cuts still bear much potential to capture combinatorial structures and tighten approximations, particularly for problems with pure binary state variables (Zou et al. 2019), which highlights the potential in handling general mixed-integer cases. Recent literature has Lagrangian cut implementations in three main areas: accelerating Lagrangian dual solution processes (Chen and Luedtke 2022, Deng and Xie 2024); selecting cuts with desirable properties (Füllner et al. 2024a,b); and improving cutting-plane frameworks by incorporating additional cut families (Ahmed et al. 2022, Deng and Xie 2024, Zhang and Sun 2022). For example, Chen and Luedtke (2022) combine feasibility and optimality cuts by employing techniques from Fischetti et al. (2010) while restricting the coefficients to the span of prior Benders cuts. This approach accelerates the cut-generation process but sacrifices optimality. Furthermore, Füllner et al. (2024a) extends the concepts of facet-defining, Pareto-optimal, and deepest cuts from Benders cuts to Lagrangian cuts through appropriate normalization.

Therefore, our work aims to develop an iterative approximation tightening framework for a general MS-SMIP setting, where we can still utilize linear Lagrangian cuts but simultaneously guarantee asymptotic convergence to a global optimum. In addition, the framework should be computationally tractable for large-scale practical applications, for which we develop cut enhancement techniques with nice theoretical properties such as Pareto-optimality.

1.2 Contributions

This paper introduces a globally converging algorithm, termed *Stochastic Dual Dynamic Programming with Lifting* (SDDP-L), to address general MS-SMIP problems. The proposed approach progressively lifts the state variables into a higher-dimensional space by introducing binary variables, enabling a tighter characterization of future cost functions via linear Lagrangian cuts on the lifted state variable space. We show that the lifting process helps the algorithm exactly characterize the nonconvex future value function. We further propose geometrically motivated enhancement techniques to overcome inefficiencies associated with vanilla Lagrangian cuts (Bansal and Küçükyavuz 2024, Zou et al. 2019), for which we demonstrate significant performance improvements in our numerical experiments. We summarize our contributions as follows:

(i) We demonstrate that Lagrangian cuts approximate the convex envelope of value functions over restricted regions of the state space. This concept of restricted region inspires the lifting process and partition refinement.

(ii) We present the SDDP-L algorithm, which partitions the original state variable space and generates Lagrangian cuts on a progressively lifted space. We prove asymptotic convergence to the global optimum for MS-SMIP with general mixed-integer state variables.

(iii) We propose two enhancement techniques, Pareto-optimal Lagrangian cuts (PLC) and square minimization cuts (SMC), to improve Lagrangian cuts' quality. We show the validity and tightness of those enhanced Lagrangian cuts.

(iv) Extensive computational experiments are conducted to compare the SDDP-L algorithm with existing methods on two large-scale, real-world MS-SMIP problems. The results show that SDDP-L consistently provides tighter lower bounds and runs faster than benchmarks.

The structure of this paper is as follows: Section 2 establishes the key geometric properties of Lagrangian cuts, demonstrating how these cuts characterize the convex envelope of value functions over restricted regions. Section 3 presents the SDDP-L algorithm and proves its asymptotic convergence to the global optimum for MS-SMIP. Section 4 proposes two enhancement techniques to improve Lagrangian cuts. Computational experiments are detailed in Section 5, highlighting the efficiency and effectiveness of our proposed methods. Finally, Section 6 summarizes our findings and suggests future directions.

1.3 Notation and Assumptions

We define the domain of value function Q_n as $\mathrm{dom}(Q_n) = \{x_{a(n)} \mid \exists (x_n, y_n)$ s.t. constraints (3b) and (3c) holds$\}$ for every node $n \in \mathcal{N}$ and state the necessary assumptions for our decision problem:

Assumption 1. *The following assumptions are satisfied by problems* (1)-(3) *for all nodes $n \in \mathcal{N}$:*

(A1) All coefficients in c_n, g_n, W_n, B_n, T_n, d_n, and p_n are rational.

(A2) The sets X_n and Y_n are rational and compact MILP-representable sets. Specifically, for each $j \in \mathcal{J}_n$, $\underline{x}_{n,j} \leq x_{n,j} \leq \overline{x}_{n,j}$. Overall we have $X_n \subseteq \mathbb{X}_n \equiv \prod_{j \in \mathcal{J}_n} [\underline{x}_{n,j}, \overline{x}_{n,j}]$.

(A3) We have relatively complete recourse, i.e., $X_{a(n)} \subseteq \mathrm{dom}(Q_n)$.

(A4) Every value function is lower-bounded, $Q_n(x_{a(n)}) \geq \underline{\theta}_n, \forall x_{a(n)} \in X_{a(n)}$.

The assumptions outlined above ensure that the value functions Q_n are proper, lower semi-continuous, and piecewise linear with finitely many pieces as discussed in Schultz (1993, 2003). The assumptions ($A3$) and ($A4$) for the root node $n = 1$ guarantee that model (2) is feasible and bounded from below by a finite $\underline{\theta}_1$.

We denote the convex hull and the closed convex hull of a set X by $\mathrm{conv}(X)$ and $\overline{\mathrm{conv}}(X)$, respectively. We define the epigraph and closed convex envelope of f over a subset of domain $\mathcal{Z} \subseteq \mathrm{dom}(f)$ as:

(i) The epigraph of f over $\mathcal{Z}$ is given by

$$\mathrm{epi}_{\mathcal{Z}}(f) = \{(x, \eta) \in \mathcal{Z} \times \mathbb{R} \mid \eta \geq f(x)\}.$$

(ii) The closed convex envelope of f over $\mathcal{Z}$ by $\overline{\mathrm{co}}_{\mathcal{Z}}(f) : \mathrm{conv}(\mathcal{Z}) \mapsto \mathbb{R}$:

$$\overline{\mathrm{co}}_{\mathcal{Z}}(f)(x) = \inf\{\eta \in \mathbb{R} \mid (x, \eta) \in \overline{\mathrm{conv}}(\mathrm{epi}_{\mathcal{Z}}(f))\}.$$

If $\mathrm{dom}(f)$ is compact and f is lower semi-continuous, then the closed convex envelope $\overline{\mathrm{co}}_{\mathcal{Z}}(f)(x)$ coincides with the convex envelope $\mathrm{co}_{\mathcal{Z}}(f)(x)$ for all $x \in \mathcal{Z}$:

$$\overline{\mathrm{co}}_{\mathcal{Z}}(f)(x) = \mathrm{co}_{\mathcal{Z}}(f)(x),$$

where $\mathrm{co}_{\mathcal{Z}}(f) : \mathrm{conv}(\mathcal{Z}) \mapsto \mathbb{R}$ is the supremum of convex functions majorized by f. We use them interchangeably in this paper.

2 Lagrangian Cuts and Convex Envelope of Value Functions

In this section, we introduce Lagrangian cuts and present their key properties. We also demonstrate how we generate Lagrangian cuts within a decomposition algorithm and what results such cuts can achieve. Specifically, we examine the relationship between value functions' convex envelopes and Lagrangian cuts, leading to a generalization of established results in the literature.

2.1 Lagrangian Cuts

We first describe the process of generating a Lagrangian cut. Suppose that Φ_n is a lower approximation for the future value functions of node n. We can formulate the following approximate value function at node n:

$$
\begin{aligned}
(P_n) \qquad & \underline{Q}_n(x_{a(n)}; \Phi_n) = \\
& \min \quad c_n^\top x_n + g_n^\top y_n + \sum_{m \in \mathcal{C}(n)} q_{nm} \phi_m(x_n) && \text{(4a)} \\
& \text{s.t.} \quad W_n x_n + B_n y_n = T_n z_n + d_n && \text{(4b)} \\
& \qquad\quad x_n \in X_n, y_n \in Y_n && \text{(4c)} \\
& \qquad\quad z_n = x_{a(n)}, && \text{(4d)}
\end{aligned}
$$

We explicitly write out the formulation for the root node:

$$
\begin{aligned}
(P_1) \qquad \underline{V} = \min \quad & c_1^\top x_1 + g_1^\top y_1 + \sum_{m \in \mathcal{C}(1)} q_{1m} \phi_m(x_1) && \text{(5a)} \\
\text{s.t.} \quad & W_1 x_1 + B_1 y_1 = d_1 && \text{(5b)} \\
& x_1 \in X_1, y_1 \in Y_1. && \text{(5c)}
\end{aligned}
$$

Here we introduce an auxiliary variable z_n and equate it to the parent node's state $x_{a(n)}$, which is commonly called the nonanticipativity constraint. If $\mathcal{C}(n) \neq \emptyset$, the lower approximation of the future value functions $\{Q_m\}_{m \in \mathcal{C}(n)}$ is $\Phi_n := \{\phi_m\}_{m \in \mathcal{C}(n)}$ with

$$
\phi_m(x_n) = \min \left\{ \theta_m \;\middle|\; \begin{array}{l} \theta_m \geq \underline{\theta}_m \\ \theta_m \geq v_m^\ell + {\pi_m^\ell}^\top x_n, \quad \forall \ell = 1, \ldots, L_m \end{array} \right\}. \tag{6}
$$

The approximation model (4) is a standard representation in cutting-plane algorithms, such as SND (Füllner and Rebennack 2022) and SDDiP (Zou et al. 2019).

Given a state variable $\hat{x}_{a(n)}$, the Lagrangian dual problem is defined as:

$$
(D_n) \qquad \max_{\pi_n} \quad \mathcal{L}_n\left(\pi_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right), \tag{7}
$$

where $\mathcal{L}_n(\pi_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}) =$

$$
\min \quad c_n^\top x_n + g_n^\top y_n + \sum_{m \in \mathcal{C}(n)} q_{nm} \theta_m + \pi_n^\top (\hat{x}_{a(n)} - z_n) \tag{8a}
$$

$$
\begin{aligned}
\text{s.t.}\quad & W_n x_n + B_n y_n = T_n z_n + d_n && && (8b)\\
& x_n \in X_n, y_n \in Y_n && && (8c)\\
& z_n \in \mathcal{Z}_{a(n)} && && (8d)\\
& \theta_m \geq \underline{\theta}_m && \forall m \in \mathcal{C}(n) && (8e)\\
& \theta_m \geq v_m^{\ell} + {\pi_m^{\ell}}^{\top} x_n && \forall m \in \mathcal{C}(n), \ell = 1, \ldots, L_m. && (8f)
\end{aligned}
$$

The auxiliary variable z_n lies within the restricted region $\mathcal{Z}_{a(n)}$. We assume $\mathcal{Z}_{a(n)} = X_{a(n)}$ for now, aligned with previous works on dual decomposition methods (Rahmaniani et al. 2020). We will show how different choices of $\mathcal{Z}_{a(n)}$ affect the generated Lagrangian cut in Section 2.2.

Let π_n^* be an optimal solution to the Lagrangian dual problem (7). We can generate a Lagrangian cut as follows:

$$
\theta_n \geq v_n^* + {\pi_n^*}^{\top} x_{a(n)}, \tag{9}
$$

where we let $v_n^* \coloneqq \mathcal{L}_n(\pi_n^*; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}) - {\pi_n^*}^{\top} \hat{x}_{a(n)}$ be the intercept. The collection of Lagrangian cuts (9) forms the lower approximation ϕ_n. When the state variable $x_{a(n)}$ is binary, Zou et al. (2019) show that *validity* and *tightness* of the cut with $\mathcal{Z}_{a(n)}$ defined as $[0,1]^{|\mathcal{J}_n|}$, i.e.,

$$
v_n^* + {\pi_n^*}^{\top} x_{a(n)} \leq \underline{Q}_n(x_{a(n)}; \Phi_n) \qquad \forall x_{a(n)} \in X_{a(n)}
$$

and

$$
v_n^* + {\pi_n^*}^{\top} \hat{x}_{a(n)} = \underline{Q}_n(\hat{x}_{a(n)}; \Phi_n).
$$

However, these properties may fail to hold when $x_{a(n)}$ includes general mixed-integer variables, as strong duality may not hold at every point $\hat{x}_{a(n)}$.

2.2 Lagrangian Cuts and Convex Envelope of $\underline{Q}_n$

In this section, we show the properties of Lagrangian cuts when applied to the case with general mixed-integer state variables. Here, Lagrangian cuts still provide a valid lower approximation for the value function $\underline{Q}_n$. In addition, Theorem 1 in Geoffrion (1974) provides the intuition that performing Lagrangian relaxation over a linear constraint is equivalent to applying the partial convex hull operator to the remaining constraints. Following a similar logic, we connect Lagrangian cuts to the convex envelope of $\underline{Q}_n$ over the restricted region $\mathcal{Z}_{a(n)}$.

Theorem 1. *For every node $n \in \mathcal{N}$, given Φ_n, $\mathcal{Z}_{a(n)}$, and an incumbent solution $\hat{x}_{a(n)} \in \mathcal{Z}_{a(n)}$, let π_n^* be an optimal solution to (D_n) in model (7). Then, the Lagrangian cut (9) defined by the cut coefficients (v_n^*, π_n^*) is valid and tight for the convex envelope of $\underline{Q}_n(\cdot; \Phi_n)$ over $\mathcal{Z}_{a(n)}$:*

(i) valid over $\mathcal{Z}_{a(n)}$:

$$
Q_n(x_{a(n)}) \geq \underline{Q}_n(x_{a(n)}; \Phi_n) \geq v_n^* + {\pi_n^*}^{\top} x_{a(n)}, \ \forall\, x_{a(n)} \in \mathcal{Z}_{a(n)}; \tag{10}
$$

(ii) tight at $\hat{x}_{a(n)}$ for the convex envelope of $\underline{Q}_n(\cdot;\Phi_n)$ over $\mathcal{Z}_{a(n)}$:

$$v_n^* + {\pi_n^*}^\top \hat{x}_{a(n)} = \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n\left(\cdot;\Phi_n\right)\right)\left(\hat{x}_{a(n)}\right). \tag{11}$$

Proof. For simplicity, let $f_n(x_n, y_n, \Theta_n) \coloneqq c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}\theta_m$ where Θ_n denotes the collection of value function variables $\{\theta_m\}_{m\in\mathcal{C}(n)}$, and

$$\mathcal{F}_n \coloneqq \{(z_n, x_n, y_n, \Theta_n) \mid \text{ constraints (8b)} - \text{(8f)}\}.$$

First, we show the validity of Lagrangian cuts by induction. For nodes in the last stage $n \in \mathcal{N}_T$, since $\mathcal{C}(n) = \emptyset$, there is no further value function approximation in Φ_n, and we have $Q_n(x_{a(n)}) = \underline{Q}_n(x_{a(n)};\Phi_n)$. For any $x_{a(n)} \in \mathcal{Z}_{a(n)}$, we have the following inequalities hold via taking Lagrangian relaxation with dual multiplier π_n^*,

$$\begin{aligned}
Q_n(x_{a(n)}) \geq & \max_{\pi_n}\left\{\min\ \left\{f_n(x_n, y_n, \Theta_n) + \pi_n^\top(x_{a(n)} - z_n) \mid (z_n, x_n, y_n, \Theta_n) \in \mathcal{F}_n\right\}\right\} \\
= & \max_{\pi_n}\left\{\pi_n^\top(x_{a(n)} - \hat{x}_{a(n)}) + \right. \\
& \qquad \left. \min\ \left\{f_n(x_n, y_n, \Theta_n) + \pi_n^\top(\hat{x}_{a(n)} - z_n) \mid (z_n, x_n, y_n, \Theta_n) \in \mathcal{F}_n\right\}\right\} \\
\geq & \mathcal{L}_n(\pi_n^*; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}) + {\pi_n^*}^\top(x_{a(n)} - \hat{x}_{a(n)}) \\
\geq & v_n^* + {\pi_n^*}^\top x_{a(n)}.
\end{aligned}$$

Suppose for a node $n \in \mathcal{N}\backslash\mathcal{N}_T$ and Lagrangian cuts defined by (v_m^ℓ, π_m^ℓ), $\forall m \in \mathcal{C}(n), \ell = 1, \ldots, L_m$ are all valid. For the induction step, we have,

$$\begin{aligned}
Q_n(x_{a(n)}) = & \min\ \{f_n(x_n, y_n, \Theta_n) \mid \text{constraints (8b)} - \text{(8d)}, z_n = x_{a(n)}, \\
& \qquad\qquad\qquad\qquad\qquad \theta_m \geq Q_m(x_n), \forall m \in \mathcal{C}(n)\} \\
\geq & \min\ \{f_n(x_n, y_n, \Theta_n) \mid (z_n, x_n, y_n, \Theta_n) \in \mathcal{F}_n, z_n = x_{a(n)}\} \\
= & \underline{Q}_n(x_{a(n)}; \Phi_n) \\
\geq & \max_{\pi_n}\{\min\ \{f_n(x_n, y_n, \Theta_n) + \pi_n^\top(x_{a(n)} - z_n) \mid (z_n, x_n, y_n, \Theta_n) \in \mathcal{F}_n\}\} \\
= & \max_{\pi_n}\{\pi_n^\top(x_{a(n)} - \hat{x}_{a(n)}) + \min\ \{f_n(x_n, y_n, \Theta_n) + \pi_n^\top(\hat{x}_{a(n)} - z_n) \mid \\
& \qquad\qquad (z_n, x_n, y_n, \Theta_n) \in \mathcal{F}_n\}\} \\
\geq & \mathcal{L}_n(\pi_n^*; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}) + {\pi_n^*}^\top(x_{a(n)} - \hat{x}_{a(n)}) \\
= & v_n^* + {\pi_n^*}^\top x_{a(n)}.
\end{aligned}$$

The steps above complete the argument that the Lagrangian cut defined by (π_n^*, v_n^*) is valid. Next, we prove the tightness result of the Lagrangian cut:

$$\begin{aligned}
& v_n^* + {\pi_n^*}^\top \hat{x}_{a(n)} \\
\overset{(a)}{=} & \min\left\{f_n(x_n, y_n, \Theta_n) \mid (z_n, x_n, y_n, \Theta_n) \in \mathrm{conv}\left(\mathcal{F}_n\right), z_n = \hat{x}_{a(n)}\right\} \\
= & \min\left\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \{(z_n, f_n(x_n, y_n, \Theta_n)) \mid (z_n, x_n, y_n, \Theta_n) \in \mathrm{conv}\left(\mathcal{F}_n\right)\}\right\} \\
\overset{(b)}{=} & \min\left\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \mathrm{conv}\left(\{(z_n, f_n(x_n, y_n, \Theta_n)) \mid (z_n, x_n, y_n, \Theta_n) \in \mathcal{F}_n\}\right)\right\} \\
= & \min\left\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \mathrm{conv}\left(\left\{(z_n, \eta') \mid \eta' \geq \underline{Q}_n(z_n; \Phi_n), z_n \in \mathcal{Z}_{a(n)}\right\}\right)\right\}
\end{aligned}$$

$$\overset{(c)}{=} \min\left\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \operatorname{conv}\left(\operatorname{epi}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n\left(\cdot; \Phi_n\right)\right)\right)\right\}$$
$$= \operatorname{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n\left(\cdot; \Phi_n\right)\right)\left(\hat{x}_{a(n)}\right),$$

where equality (a) results from Theorem 1 in Geoffrion (1974) and the definition of convex envelope for a lower semi-continuous function leads to equality (c). We provide a detailed proof of equality (b) in Lemma 11. Thus, we prove the tightness of Lagrangian cut at $\hat{x}_{a(n)}$ for the convex envelope of $\underline{Q}_n(\cdot; \Phi_n)$ over $\mathcal{Z}_{a(n)}$. □

The convex envelope, obtained via the Lagrangian cuts, depends on the selection of $\mathcal{Z}_{a(n)}$. The result of Theorem 1 can be extended to the case where $\mathcal{Z}_{a(n)}$ is a superset or a subset of $X_{a(n)}$. If $X_{a(n)} \subseteq \mathcal{Z}_{a(n)}$, the validity of approximation for Q_n and $\underline{Q}_n$ still holds. We use an example to illustrate the relationship between the convex envelope of the value function and the corresponding Lagrangian cuts over different restricted regions.

Example 1. *We consider the following problem with a scenario tree $\mathcal{N} = \{1, n\}$ with $a(n) = 1$ and $X_1 = \{0, 1, 2\}$:*

$$\min_{x_1 \in X_1} \quad Q_n(x_1),$$

where $Q_n(x_1) \coloneqq \min\{x_n \mid 1.5x_n \geq x_1, x_n \in \{0, 1, 2\}\}$. Let us define $\mathcal{Z}_1^1 = [0, 2]$ and $\mathcal{Z}_1^2 = \{0, 1, 2\}$. The three-piece black line represents the value function Q_n when we set its domain as $\mathcal{Z}_1^1$. The illustration of Q_n becomes three discrete dots if we set the domain as $\mathcal{Z}_1^2$. The choice of domain results in different convex envelopes $\operatorname{co}_{\mathcal{Z}_1^1}(Q_n)$ and $\operatorname{co}_{\mathcal{Z}_1^2}(Q_n)$, represented by the teal lines and the gray line, respectively. This comparison is illustrated in Fig. 1(a) and shows how different choices of $\mathcal{Z}_1$ influence the convex envelope of the value function. Furthermore, if we solve the following Lagrangian dual problem (D_n^1) with $\mathcal{Z}_1^1$:

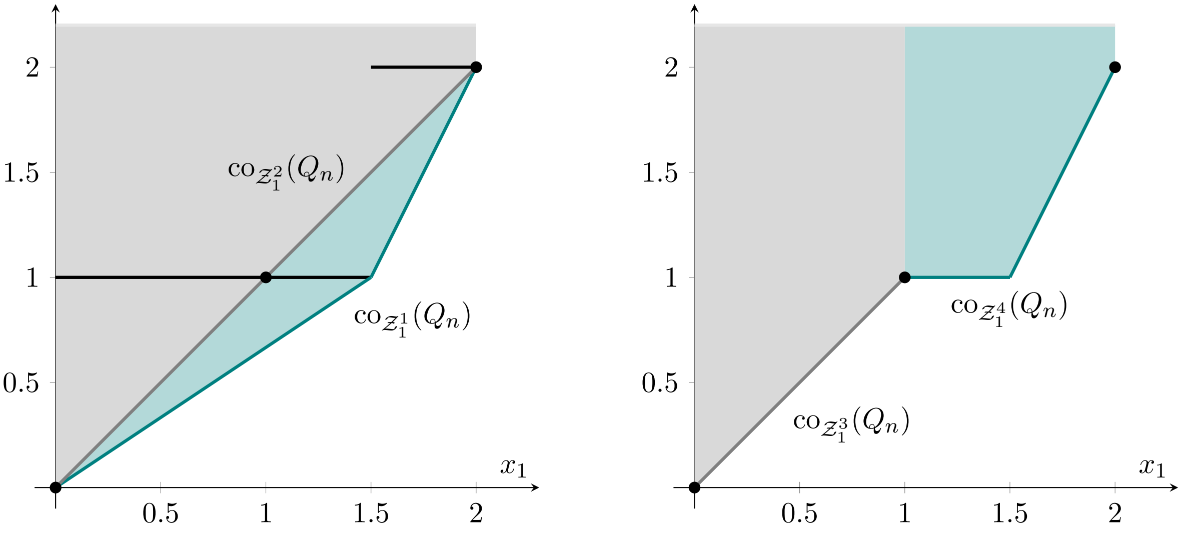
$$(D_n^1) \quad \max_{\pi} \; \min \quad x_n + \pi(\hat{x}_1 - z_n) \qquad\qquad (D_n^2) \quad \max_{\pi} \; \min \quad x_n + \pi(\hat{x}_1 - z_n)$$
$$\text{s.t.} \quad 1.5x_n \geq z_n \qquad\qquad \text{s.t.} \quad 1.5x_n \geq z_n$$
$$x_n \in \{0, 1, 2\} \qquad\qquad x_n \in \{0, 1, 2\}$$
$$z_n \in \mathcal{Z}_1^1 \qquad\qquad z_n \in \mathcal{Z}_1^2.$$

The Lagrangian cuts obtained at different $\hat{x}_1$ align with the teal lines and cannot be pushed further up. In this case, $\mathcal{Z}_1^1$ is a superset of X_1. However, if we substitute the restricted region of z_n as $\mathcal{Z}_1^2 = X_1 = \{0, 1, 2\}$, the Lagrangian cuts will align with the gray line. We can observe that when expanding the restricted region $\mathcal{Z}_1$, the convex envelope $\operatorname{co}_{\mathcal{Z}_1}(Q_n)$ becomes farther away from Q_n and $\operatorname{co}_{X_1}(Q_n)$.

If $\mathcal{Z}_1^1$ is partitioned into $\mathcal{Z}_1^3 = [0, 1]$ and $\mathcal{Z}_1^4 = [1, 2]$, the corresponding convex envelopes of Q_n over $\mathcal{Z}_1^3$ and $\mathcal{Z}_1^4$ are depicted in Fig. 1(b). Each of $\mathcal{Z}_1^3$ and $\mathcal{Z}_1^4$ is a subset of $\mathcal{Z}_1^1$. The Lagrangian *cuts generated to characterize the convex envelope* $\mathrm{co}_{\mathcal{Z}_1^3}(Q_n)$ *and* $\mathrm{co}_{\mathcal{Z}_1^4}(Q_n)$ *may not be valid for* $\mathrm{co}_{\mathcal{Z}_1^1}(Q_n)$*, nor for* $\mathrm{co}_{\mathcal{Z}_1^2}(Q_n)$*. However, the union of these convex epigraphs is a subset of the convex epigraph over* $\mathcal{Z}_1^1$*, i.e.,*

$$\mathrm{conv}(\mathrm{epi}_{\mathcal{Z}_1^3}(Q_n)) \cup \mathrm{conv}(\mathrm{epi}_{\mathcal{Z}_1^4}(Q_n)) \subset \mathrm{conv}(\mathrm{epi}_{\mathcal{Z}_1^1}(Q_n)).$$

The union of two convex envelope curves $\mathrm{co}_{\mathcal{Z}_1^3}(Q_n)$ *and* $\mathrm{co}_{\mathcal{Z}_1^4}(Q_n)$ *is closer to* Q_n *and* $\mathrm{co}_{X_1}(Q_n)$ *compared to* $\mathrm{co}_{\mathcal{Z}_1^1}(Q_n)$*. This inspires us with the idea of spatial partitioning in Section 3.* □



(a) Value function Q_n and its convex envelopes under $\mathcal{Z}_1^1$ (teal lines) and $\mathcal{Z}_1^2$ (gray line).

(b) Illustration of the union of convex envelopes of Q_n under $\mathcal{Z}_1^3$ (teal lines) and $\mathcal{Z}_1^4$ (gray line).

Figure 1: Illustration of different convex envelopes for Example 1.

We list natural choices for $\mathcal{Z}_{a(n)}$ and discuss their properties.

1. $\mathcal{Z}_{a(n)} = X_{a(n)}$. This is the most intuitive choice since $X_{a(n)} \subseteq \mathrm{dom}(Q_n)$. Here, $\mathcal{Z}_{a(n)} = X_{a(n)}$ ensures that the approximation of Q_n is valid at any feasible $x_{a(n)}$. Since X_n and Y_n are both mixed-integer sets, using $\mathcal{Z}_{a(n)} = X_{a(n)}$ in the Lagrangian dual model does not affect the fact that model (8) remains a mixed-integer program.

2. $\mathcal{Z}_{a(n)} = \mathrm{conv}(X_{a(n)})$. Although this choice may lead to loose lower approximations of Q_n on $X_{a(n)}$, as it extends the approximation over the convex hull of $X_{a(n)}$, it can be advantageous in certain contexts when the convex hull is easy to obtain; e.g., this choice was utilized in SDDiP (Zou et al. 2019) to replace the binary range with $[0, 1]$ interval.

3. $\mathcal{Z}_{a(n)} = X_{a(n)}^{LP}$. In this case, $\mathcal{Z}_{a(n)}$ represents the LP relaxation of $X_{a(n)}$. The benefit of this choice is that no additional integer variables are introduced in the Lagrangian dual function (8), simplifying the computational burden while retaining a valid lower approximation.

2.3 Lagrangian Cut-Based Cutting-Plane Algorithm for MS-SMIP

Cutting-plane algorithms, such as SND and SDDP, are widely used to solve multi-stage stochastic programs, as the key issue is to approximate the future value functions. It is natural to replace the Benders cut used in multi-stage stochastic convex programming by the Lagrangian cut to deal with the nonconvexity introduced by mixed-integer variables. In this section, we first outline details of the iterative cutting-plane procedure in Algorithm 1.

Algorithm 1: Lagrangian Cutting-plane Algorithm for MS-SMIP.

```
Initialize with $LB \leftarrow -\infty, UB \leftarrow +\infty, i \leftarrow 1$, sets $\mathcal{Z}_{a(n)} \leftarrow X_n$ and an lower approximation
 $\{\Phi_n^i\} \quad \forall n \in \mathcal{N}$;
while stopping criterion is not satisfied do
    /* Forward pass                                                          */
    Sample $M$ scenario paths $\{\mathcal{P}^{i,\omega}\}_{\omega=1}^{M}$ independently;
    for $\omega = 1, \dots, M$ do
        for $n \in \mathcal{P}^{i,\omega}$ do
            Solve forward problem $(P_n)$ in model (4) with $x_{a(n)}^i$ and $\Phi_n^i$;
            Collect the solution $(x_n^i,\, y_n^i)$;
        end
        $u^\omega \leftarrow \sum_{n \in \mathcal{P}^{i,\omega}} c_n^\top x_n^i + g_n^\top y_n^i$;
    end
    /* Update statisical upper bound                                         */
    Let $\hat{\mu} \leftarrow \frac{1}{M} \sum_{\omega=1}^{M} u^\omega$ and $\hat{\sigma}^2 \leftarrow \frac{1}{M-1} \sum_{\omega=1}^{M} (u^\omega - \hat{\mu})^2$;
    $UB \leftarrow \hat{\mu} + z_{\alpha/2} \hat{\sigma} / \sqrt{M}$;
    /* Backward pass                                                         */
    for $t = T-1, \dots, 1$ do
        for $n \in \mathcal{N}_t$ do
            if $n \in \mathcal{P}^{i,\omega}$ for some $\omega$ then
                for $m \in \mathcal{C}(n)$ do
                    Solve the Lagrangian dual problem $(D_m)$ in model (7) with $x_n^i$, $\Phi_m^{i+1}$, and $\mathcal{Z}_n$;
                    Collect $(D_m)$'s optimal solution $\pi_m^{*,i}$ and intercept $v_m^{*,i}$;
                    Add a Lagrangian cut defined by coefficients $(v_m^{*,i}, \pi_m^{*,i})$ to $\phi_m^i$;
                    $L_m \leftarrow L_m + 1$ and $\phi_m^{i+1} \leftarrow \phi_m^i$;
                end
                $\Phi_n^{i+1} \leftarrow \{\phi_m^{i+1}\}_{m \in \mathcal{C}(n)}$ ;
            else
                $\Phi_n^{i+1} \leftarrow \Phi_n^i$;
            end
        end
    end
    /* Update lower bound                                                    */
    Solve forward problem $(P_1)$, obtain the optimal value $\underline{V}^i$, and set $LB \leftarrow \underline{V}^i$;
    Update $i \leftarrow i + 1$;
end
```

Each iteration of Algorithm 1 can be divided into two parts: forward pass and backward pass. The superscript i represents the iteration index. In the i-th iteration, we first generate M scenario paths $\{\mathcal{P}^{i,\omega}\}_{\omega=1}^{M}$ in the forward pass. Each path $\mathcal{P}^{i,\omega}$ represents a sequence of nodes from the root to a leaf, where each node has its ancestor and descendants in the same path. The forward pass proceeds stage by stage from $t = 1$ to T, solving a forward nodal problem (P_n) for each sampled node $n \in \mathcal{P}^{i,\omega}$, given the optimal solution $x^i_{a(n)}$ from the parent node $a(n)$ and the current future value approximation Φ^i_n. The solution x^i_n obtained at each node n is then passed forward to solve the problems at its sampled child node $m \in \mathcal{C}(n)$. Note that $x^i_n = x^{i,\omega}_n$ for all ω such that $n \in \mathcal{P}^{i,\omega}$. Accordingly, we omit the superscript ω when recording solutions.

Once all forward problems on the sampled paths are solved in iteration i, the backward pass begins from the final stage T. The purpose of the backward pass is to update the approximate cost-to-go function $\Phi^i_n(\cdot)$ for each sampled node $n \in \mathcal{P}^{i,\omega}$. In particular, given a solution x^i_n for the sampled node $n \in \mathcal{N}_{T-1}$, for each child node $m \in \mathcal{C}(n)$, we solve the Lagrangian dual problem (D_m) to obtain the optimal dual multiplier $\pi^{*,i}_m$ and intercept $v^{*,i}_m$. Those coefficients form a Lagrangian cut underestimating the true value function Q_m. The generated cut is used to update ϕ^i_m to ϕ^{i+1}_m. Note that $\Phi^i_m \equiv \emptyset$ for all i because there is no cost-to-go function at the last-stage node m, as $\mathcal{C}(m) = \emptyset$. For a sampled node $n \in \mathcal{N}_{t-1}$, where $t \in \{1, \ldots, T-1\}$, and for each child node $m \in \mathcal{C}(n)$, the updated set Φ^{i+1}_m will be incorporated in model (D_m). We solve model (D_m) and add the obtained Lagrangian cut to ϕ^i_m, updating the lower approximation at node n from Φ^i_n to Φ^{i+1}_n. The backward process continues iteratively until the root node is reached.

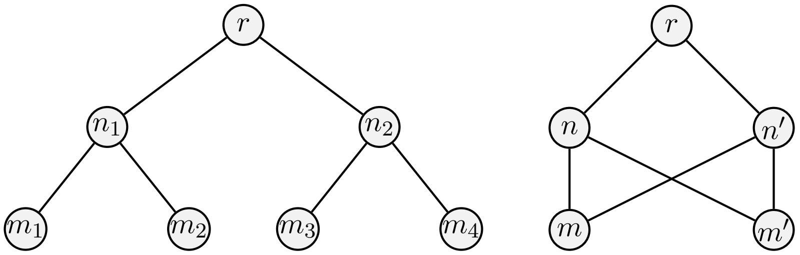
A potential termination criterion for Algorithm 1 can be that when the gap between UB and LB is small enough. Another commonly used termination criterion is to set a time limit or an iteration number limit (Dowson 2020).

Remark: Algorithm 1 is based on SND with a general scenario tree illustrated in Fig. 2(a), where each node may have a distinct value function. This paper assumes a general scenario tree structure for all the following algorithms. However, we can simplify the scenario tree structure when we assume stage-wise independence. We refer to such a scenario tree representing stage-wise independent uncertainty as an SDDP scenario tree, illustrated in Fig. 2(b). With stage-wise independence, for any two nodes n and n' in $\mathcal{N}_t$, their child nodes $\mathcal{C}(n)$ and $\mathcal{C}(n')$ are defined by identical data and conditional probabilities. Therefore, for any $m \in \mathcal{C}(n)$, the value function $Q_m(\cdot)$ used in model (3) for node n should be identical when it is used for node n'. In such case, cuts obtained for ϕ_m at a sampled x^i_n can be shared among all Φ_n of nodes within the same stage $n \in \mathcal{N}_t$.

We prove the convergence of Algorithm 1 by demonstrating that, with probability one, the approximate cost-to-go functions constructed using Lagrangian cuts form a convoluted convex envelope approximation of the value function. We recursively define the convoluted convex envelope approximation of the value function as follows:

Definition 1. *For $n \in \mathcal{N}_T$, we let*

$$\tilde{Q}_n(x_{a(n)}) = \min \quad c_n^\top x_n + g_n^\top y_n \tag{12a}$$



(a) General scenario tree used for SND.

(b) SDDP scenario tree.

Figure 2: Illustration of a general and SDDP scenario tree.

$$\text{s.t.} \quad W_n x_n + B_n y_n = T_n x_{a(n)} + d_n \tag{12b}$$

$$x_n \in X_n, y_n \in Y_n. \tag{12c}$$

Note that $\tilde{Q}_n(x_{a(n)}) = Q_n(x_{a(n)})$ for $n \in \mathcal{N}_T$. For any nodes that are neither a leaf node nor the root node, $n \in \mathcal{N} \backslash (\mathcal{N}_T \cup \{1\})$, we let

$$\tilde{Q}_n(x_{a(n)}) = \min \quad c_n^\top x_n + g_n^\top y_n + \sum_{m \in \mathcal{C}(n)} q_{nm} \mathrm{co}_{\mathcal{Z}_n}\left(\tilde{Q}_m\right)(x_n) \tag{13a}$$

$$\text{s.t.} \quad W_n x_n + B_n y_n = T_n x_{a(n)} + d_n \tag{13b}$$

$$x_n \in X_n, y_n \in Y_n. \tag{13c}$$

We call function $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)$ the convoluted convex envelope approximation of the value function at node $n \in \mathcal{N} \backslash \{1\}$. We assume $X_n \subseteq \mathcal{Z}_n$ to avoid the invalid approximation issue. And finally, for the root node 1, we define the optimal value for the convoluted convex envelope approximation as:

$$\tilde{V}^* = \min \quad c_1^\top x_1 + g_1^\top y_1 + \sum_{m \in \mathcal{C}(1)} q_{1m} \mathrm{co}_{\mathcal{Z}_1}\left(\tilde{Q}_m\right)(x_1) \tag{14a}$$

$$\text{s.t.} \quad W_1 x_1 + B_1 y_1 = d_1 \tag{14b}$$

$$x_1 \in X_1, y_1 \in Y_1. \tag{14c}$$

Theorem 2. *We assume the sampling process in Algorithm 1 satisfies the forward pass sampling property and backward pass sampling property in Philpott and Guan (2008) and the Lagrangian cut coefficients are finite $\|\pi_n^{*,i}\| < \infty$ for all $n \in \mathcal{N} \backslash \{1\}$. The following results hold:*

(i) For any $n \in \mathcal{N} \backslash \{1\}$, suppose we denote the optimal solution as $(x_{a(n)}^k, y_{a(n)}^k)$ when model (P_n) is solved for the k-th time in Algorithm 1. For every converging infinite sequence $\{(x_{a(n)}^k, y_{a(n)}^k)\}_{k \in K \subseteq \mathbb{N}}$ to $(x_{a(n)}^*, y_{a(n)}^*)$, there exists an infinite subsequence $J \subseteq K$ such that:

- *The sequence $\{(x_n^k, y_n^k)\}_{k \in J}$ converges to a point (x_n^*, y_n^*).*
- *The sequence $\left\{\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot; \Phi_n^{k+1})\right)(x_{a(n)}^k)\right\}_{k \in J}$ converges to $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^*)$ with probability one.*

- *The sequence* $\{\phi_n^{k+1}(x_{a(n)}^k)\}_{k\in J}$ *converges to* $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^*)$ *with probability one.*

(ii) The sequence $\{\underline{V}^i\}_{i\in\mathbb{N}}$ *converges to* $\tilde{V}^*$ *with probability one. Every accumulation point of the sequence of the solution to model* (P_1) *is an optimal solution to model* (14).

The proof of Theorem 2 follows the convergence proof of Theorems 1 and 2 in Yang and Nagarajan (2022). We present the details of this proof in Appendix B. The results demonstrate that the cut approximation ϕ_n^i can only converge to the convoluted convex envelope approximation $\mathrm{co}_{\mathcal{Z}_{a(n)}}(\tilde{Q}_n)$ rather than the true value function Q_n. Thus, Algorithm 1 only converges to a lower bound, $\tilde{V}^*$, rather than the true optimal value of MS-SMIP, V^*.

3 Lift-and-Cut Algorithm

As discussed in Section 2, Algorithm 1 with Lagrangian cuts characterizes a convoluted convex envelope approximation of value functions and only achieves a lower bound. In order to pursue the global optimum of MS-SMIP, we must generate cuts that can characterize value functions accurately.

According to Theorem 1, Lagrangian cuts form the convex envelope of $\underline{Q}_n$ on a specific region $\mathcal{Z}_{a(n)}$ given the future value function approximation Φ_n. The gap between the approximate value function $\underline{Q}_n$ and its convex envelope $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n\right)$ varies in different $\mathcal{Z}_{a(n)}$, and becomes zero uniformly over $\mathcal{Z}_{a(n)}$ if $\underline{Q}_n$ can be exactly characterized everywhere on $\mathcal{Z}_{a(n)}$ by a piecewise linear convex function, as shown in Example 1 where $\mathcal{Z}_{a(n)} = \{0,1,2\}$. In this section, we look into the method to partition the set $\mathcal{Z}_{a(n)}$ into smaller pieces, for which we can use binary indicator variables to represent. We can generate Lagrangian cuts on the lifted state space, including the original state variables and the binary indicators, so that we can reduce the aforementioned gap to zero and recover the true value function Q_n, eventually leading to global convergence of the cutting-plane algorithm.

3.1 Lifted Lagrangian Cuts

For the subproblem at node $n \in \mathcal{N}$, each component of the state variable $x_{n,j}, j \in \mathcal{J}_n$ is bounded in an interval $[\underline{x}_{n,j}, \overline{x}_{n,j}]$ according to Assumption 1. We first define a partition of such an interval $[\underline{x}_{n,j}, \overline{x}_{n,j}]$.

Definition 2. *A partition of interval* $[\underline{x}_{n,j}, \overline{x}_{n,j}]$ *is an ordered set of two-element tuples* $\mathcal{P}_{n,j} = \{[\underline{x}_{n,j}^\sigma, \overline{x}_{n,j}^\sigma], \sigma \in \mathcal{S}_{n,j}\}$ *with index set* $\mathcal{S}_{n,j} = \{1, 2, \ldots, S_{n,j}\}$ *and following properties:*

- $\underline{x}_{n,j}^1 = \underline{x}_{n,j}$;
- $\overline{x}_{n,j}^{S_{n,j}} = \overline{x}_{n,j}$;

- $\overline{x}^{\sigma}_{n,j} > \underline{x}^{\sigma}_{n,j} \qquad \forall \sigma \in \mathcal{S}_{n,j};$
- $\overline{x}^{\sigma-1}_{n,j} = \underline{x}^{\sigma}_{n,j} \qquad \forall \sigma \in \{2, \ldots, S_{n,j}\}.$

The partition is defined in a way to fit all types of variables, including continuous, integer, and binary variables. In the actual implementation of algorithms, it is possible to adjust the partition setup for integer or binary variables to only consider intervals feasible for them; e.g., a partition of an integer variable in $[1, 6]$ may consists of two non-adjacent intervals $[1, 3]$ and $[4, 6]$ since it is not possible for the integer variable to take any value in $(3, 4)$. With the interval partition defined in Definition 2, for each node $n \in \mathcal{N}$, we can represent the partition of the state variables' feasible region via

$$\mathbb{P}_n = \prod_{j \in \mathcal{J}_n} \mathcal{P}_{n,j}.$$

We focus on the setup where each element within this partition $\mathbb{P}_n$ is a hyperrectangle, which makes it straightforward to split elements within a partition for further bound-tightening derivation. The hyperrectangle setup can be extended to other geometric shapes, such as simplices, cones, and ellipsoids. For a discussion of other geometric shapes, we refer the reader to Chapter 5 of Locatelli and Schoen (2013).

We introduce binary variables $\lambda^{\sigma}_{n,j}$ and following logical constraints to indicate whether variable $x_{n,j}$ is within the interval σ of partition $\mathcal{P}_{n,j}$. The feasible region of such binary variables, collectively denoted as λ_n, can be represented by a set parametrized by the value of x_n and the partition $\mathbb{P}_n$:

$$\Lambda_n(x_n, \mathbb{P}_n) = \left\{ \lambda_n \;\middle|\; \begin{array}{l} \lambda^{\sigma}_{n,j} \in \{0,1\} \quad \forall \sigma \in \mathcal{S}_{n,j} \\ \sum_{\sigma \in \mathcal{S}_{n,j}} \lambda^{\sigma}_{n,j} = 1 \\ \sum_{\sigma \in \mathcal{S}_{n,j}} \underline{x}^{\sigma}_{n,j} \lambda^{\sigma}_{n,j} \le x_{n,j} \le \sum_{\sigma \in \mathcal{S}_{n,j}} \overline{x}^{\sigma}_{n,j} \lambda^{\sigma}_{n,j} \end{array} , \forall j \in \mathcal{J}_n \right\}. \tag{15}$$

With a given partition $\mathbb{P}_n$ and associated binary variables λ_n, we can add λ_n as state variables and reformulate the approximation problem on a lifted x-λ space:

$$\begin{aligned}
\left(P^L_n\right) \qquad & \underline{Q}^L_n(x_{a(n)}, \lambda_{a(n)}; \Phi^L_n, \mathbb{P}_n, \mathbb{P}_{a(n)}) = \\
& \min \qquad c_n^\top x_n + g_n^\top y_n + \sum_{m \in \mathcal{C}(n)} q_{nm} \phi^L_m(x_n, \lambda_n) && \text{(16a)} \\
& \text{s.t.} \qquad W_n x_n + B_n y_n = T_n z_n + d_n && \text{(16b)} \\
& \qquad\qquad x_n \in X_n, y_n \in Y_n, \lambda_n \in \Lambda_n(x_n, \mathbb{P}_n) && \text{(16c)} \\
& \qquad\qquad z_n = x_{a(n)}, \mu_n = \lambda_{a(n)}, && \text{(16d)}
\end{aligned}$$

while for the root node, we have

$$\begin{aligned}
\left(P^L_1\right) \qquad \underline{V} = \min \quad & c_1^\top x_1 + g_1^\top y_1 + \sum_{m \in \mathcal{C}(1)} q_{1m} \phi^L_m(x_1, \lambda_1) && \text{(17a)} \\
\text{s.t.} \quad & W_1 x_1 + B_1 y_1 = d_1 && \text{(17b)} \\
& x_1 \in X_1, y_1 \in Y_1, \lambda_1 \in \Lambda_1(x_1, \mathbb{P}_1). && \text{(17c)}
\end{aligned}$$

The cuts $\Phi_n^L = \{\phi_m^L\}_{m\in\mathcal{C}(n)}$ are obtained on a lifted space:

$$\phi_m^L(x_n,\lambda_n) = \min\left\{ \theta_m \;\middle|\; \begin{array}{l} \theta_m \geq \underline{\theta}_m \\ \theta_m \geq v_m^\ell + {\pi_m^\ell}^\top x_n + {\rho_m^\ell}^\top \lambda_n, \quad \forall \ell = 1,\dots,L_m \end{array} \right\}.$$

To generate a Lagrangian cut above, we also need to update the Lagrangian dual problem, given a solution $(\hat{x}_{a(n)}, \hat{\lambda}_{a(n)})$ in the lifted space:

$$\left(D_n^L\right) \qquad \max_{\pi_n,\rho_n} \quad \mathcal{L}_n^L(\pi_n,\rho_n;\hat{x}_{a(n)},\hat{\lambda}_{a(n)},\Phi_n^L,\mathcal{Z}_{a(n)},\mathbb{P}_n,\mathbb{P}_{a(n)}), \tag{18}$$

where $\mathcal{L}_n^L(\pi_n,\rho_n;\hat{x}_{a(n)},\hat{\lambda}_{a(n)},\Phi_n^L,\mathcal{Z}_{a(n)},\mathbb{P}_n,\mathbb{P}_{a(n)}) =$

$$\begin{aligned}
\min \quad & c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}\theta_m + \pi_n^\top(\hat{x}_{a(n)} - z_n) + \rho_n^\top(\hat{\lambda}_{a(n)} - \mu_n) && \text{(19a)}\\
\text{s.t.} \quad & W_n x_n + B_n y_n = T_n z_n + d_n && \text{(19b)}\\
& x_n \in X_n, y_n \in Y_n, \lambda_n \in \Lambda_n(x_n,\mathbb{P}_n) && \text{(19c)}\\
& z_n \in \mathcal{Z}_{a(n)}, \mu_n \in \Lambda_{a(n)}(z_n,\mathbb{P}_{a(n)}) && \text{(19d)}\\
& \theta_m \geq \underline{\theta}_m \qquad \forall m \in \mathcal{C}(n) && \text{(19e)}\\
& \theta_m \geq v_m^\ell + {\pi_m^\ell}^\top x_n + {\rho_m^\ell}^\top \lambda_n \qquad \forall \ell = 1,\dots,L_n, m\in\mathcal{C}(n). && \text{(19f)}
\end{aligned}$$

Constraint (19d) serves a similar purpose as constraint (8d) by restricting the lifted copy variables (z_n,μ_n) to the set

$$\mathfrak{Z}_n(\mathcal{Z}_{a(n)},\mathbb{P}_{a(n)}) = \{(z_n,\mu_n) \mid z_n \in \mathcal{Z}_{a(n)}, \mu_n \in \Lambda_{a(n)}(z_n,\mathbb{P}_{a(n)})\}.$$

Let the optimal solution to the Lagrangian dual problem (18) be (π_n^*,μ_n^*). We can determine the cut intercept

$$v_n^* = \mathcal{L}_n^L(\pi_n^*,\rho_n^*;\hat{x}_{a(n)},\hat{\lambda}_{a(n)},\Phi_n^L,\mathcal{Z}_{a(n)},\mathbb{P}_n,\mathbb{P}_{a(n)}) - {\pi_n^*}^\top\hat{x}_{a(n)} - {\rho_n^*}^\top\hat{\lambda}_{a(n)},$$

and generate a Lagrangian cut on the lifted space:

$$\theta_n \geq v_n^* + {\pi_n^*}^\top x_{a(n)} + {\rho_n^*}^\top \lambda_{a(n)}. \tag{20}$$

Like Section 2, we present the validity and tightness properties of lifted Lagrangian cuts here.

Theorem 3. *For every node $n \in \mathcal{N}$, given Φ_n^L, $\mathcal{Z}_{a(n)}$, current partitions $\mathbb{P}_n$ and $\mathbb{P}_{a(n)}$, and an incumbent solution $(\hat{x}_{a(n)},\hat{\lambda}_{a(n)}) \in \mathfrak{Z}_n(\mathcal{Z}_{a(n)},\mathbb{P}_{a(n)})$, let (π_n^*,ρ_n^*) be an optimal solution to (D_n^L) in model (18). Then, the Lagrangian cut (20) defined by the cut coefficients (v_n^*,π_n^*,ρ_n^*) is valid and tight in the following sense:*

(i) valid over $\mathfrak{Z}_n(\mathcal{Z}_{a(n)},\mathbb{P}_{a(n)})$:

$$\begin{aligned}
Q_n(x_{a(n)}) &\geq \underline{Q}_n^L(x_{a(n)},\lambda_{a(n)};\Phi_n^L,\mathbb{P}_n,\mathbb{P}_{a(n)})\\
&\geq v_n^* + {\pi_n^*}^\top x_{a(n)} + {\rho_n^*}^\top \lambda_{a(n)},\\
&\qquad \forall\,(x_{a(n)},\lambda_{a(n)}) \in \mathfrak{Z}_n(\mathcal{Z}_{a(n)},\mathbb{P}_{a(n)});
\end{aligned} \tag{21}$$

(ii) tight at $(\hat{x}_{a(n)}, \hat{\lambda}_{a(n)})$ for the convex envelope of $\underline{Q}^L_n(\cdot,\cdot;\Phi^L_n,\mathbb{P}_n,\mathbb{P}_{a(n)})$ over $\mathfrak{Z}_n(\mathcal{Z}_{a(n)},\mathbb{P}_{a(n)})$:

$$\begin{aligned} & v^*_n + {\pi^*_n}^\top \hat{x}_{a(n)} + {\rho^*_n}^\top \hat{\lambda}_{a(n)} = \\ & \qquad \mathrm{co}_{\mathfrak{Z}_n(\mathcal{Z}_{a(n)},\mathbb{P}_{a(n)})}\left(\underline{Q}^L_n(\cdot,\cdot;\Phi^L_n,\mathbb{P}_n,\mathbb{P}_{a(n)})\right)(\hat{x}_{a(n)},\hat{\lambda}_{a(n)}) \end{aligned} \tag{22}$$

(iii) tight at $\hat{x}_{a(n)}$ for the convex envelope of $\underline{Q}_n(\cdot;\Phi_n)$ over the active restricted region on the original state variable space:

$$v^*_n + {\pi^*_n}^\top \hat{x}_{a(n)} + {\rho^*_n}^\top \hat{\lambda}_{a(n)} = \mathrm{co}_{\hat{\mathcal{Z}}_{a(n)}}(\underline{Q}_n(\cdot;\Phi_n))(\hat{x}_{a(n)}), \tag{23}$$

where the active restricted region is defined as:

$$\hat{\mathcal{Z}}_{a(n)} = \mathcal{Z}_{a(n)} \cap \prod_{j\in\mathcal{J}_{a(n)}} \left[\sum_{\sigma\in\mathcal{S}_{a(n),j}} \hat{\lambda}^\sigma_{a(n),j}\underline{x}^\sigma_{n,j}, \sum_{\sigma\in\mathcal{S}_{a(n),j}} \hat{\lambda}^\sigma_{a(n),j}\overline{x}^\sigma_{n,j}\right], \tag{24}$$

and the lower approximation Φ_n is a projection of Φ^L_n onto the space of x_n as:

$$\begin{aligned} &\Phi_n = \{\phi_m(\cdot;\mathbb{P}_n)\}_{m\in\mathcal{C}(n)}, \\ &\phi_m(x_n;\mathbb{P}_n) = \min\left\{ \theta_m \;\middle|\; \begin{array}{l} \theta_m \ge \underline{\theta}_m \\ \theta_m \ge \min_{\lambda_n\in\Lambda_n(x_n,\mathbb{P}_n)} v^\ell_m + {\pi^\ell_m}^\top x_n + {\rho^\ell_m}^\top \lambda_n, \\ \qquad\qquad \forall \ell = 1,\ldots,L_m \end{array} \right\}. \end{aligned}$$

Proof. The first two parts of the proof follow the same logic of the proof for Theorem 1. Here we focus on proving part (iii). Similar to the proof of Theorem 1, we define

$$\begin{aligned} & f_n(x_n,\lambda_n,y_n,\Theta_n) \coloneqq c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}\theta_m \\ & \mathcal{F}_n \coloneqq \{(z_n,\mu_n,x_n,\lambda_n,y_n,\Theta_n) \mid \text{ constraints (19b)} - \text{(19f)}\}. \end{aligned}$$

We further define two auxiliary feasible sets

$$\begin{aligned} \mathcal{F}'_n &\coloneqq \left\{(z_n,\mu_n,x_n,\lambda_n,y_n,\Theta_n) \mid \text{ constraints (19b)} - \text{(19f)}, \mu_n = \hat{\lambda}_{a(n)}\right\} \\ \mathcal{F}''_n &\coloneqq \{(z_n,\mu_n,x_n,\lambda_n,y_n,\Theta_n) \mid \text{ constraints (19b), (19c), (19e), (19f)}, \\ &\qquad\qquad z_n \in \hat{\mathcal{Z}}_{a(n)}\right\}, \end{aligned}$$

and we can make the following derivation:

$$\begin{aligned} & v^*_n + {\pi^*_n}^\top \hat{x}_{a(n)} + {\rho^*_n}^\top \hat{\lambda}_{a(n)} \\ & \overset{(a)}{=} \min\{f_n(x_n,\lambda_n,y_n,\Theta_n) \mid (z_n,\mu_n,x_n,\lambda_n,y_n,\Theta_n) \in \mathrm{conv}(\mathcal{F}_n), \\ & \qquad\qquad z_n = \hat{x}_{a(n)}, \mu_n = \hat{\lambda}_{a(n)}\right\} \\ & \overset{(b)}{=} \min\left\{f_n(x_n,\lambda_n,y_n,\Theta_n) \mid (z_n,\mu_n,x_n,\lambda_n,y_n,\Theta_n) \in \mathrm{conv}(\mathcal{F}'_n), z_n = \hat{x}_{a(n)}\right\} \\ & = \min\left\{\eta \mid (\hat{x}_{a(n)},\eta) \in \{(z_n, f_n(x_n,\lambda_n,y_n,\Theta_n)) \mid (z_n,x_n,\lambda_n,y_n,\Theta_n) \in \mathrm{conv}(\mathcal{F}''_n)\}\right\} \end{aligned}$$

$$
\begin{aligned}
&\overset{(c)}{=} \min\left\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \operatorname{conv}(\{(z_n, f_n(x_n, \lambda_n, y_n, \Theta_n)) \mid (z_n, x_n, \lambda_n, y_n, \Theta_n) \in \mathcal{F}''_n\})\right\} \\
&\overset{(d)}{=} \min\Big\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \operatorname{conv}\Big(\Big\{(z_n, \eta') \mid \\
&\qquad\qquad\qquad\qquad \eta' \geq \min \; f_n(x_n, \lambda_n, y_n, \Theta_n), \\
&\qquad\qquad\qquad\qquad\qquad \text{s.t.} \quad \text{constraints (19b), (19e)}, \\
&\qquad\qquad\qquad\qquad\qquad\qquad x_n \in X_n, y_n \in Y_n, z_n \in \hat{\mathcal{Z}}_{a(n)} \\
&\qquad\qquad\qquad\qquad\qquad\qquad \theta_m \geq \min_{\lambda_n \in \Lambda_n(x_n, \mathbb{P}_n)} v^{\ell}_m + {\pi^{\ell}_m}^{\top} x_n + {\rho^{\ell}_m}^{\top} \lambda_n \Big\}\Big)\Big\} \\
&\overset{(e)}{=} \operatorname{co}_{\hat{\mathcal{Z}}_{a(n)}}\left(\underline{Q}_n\right)(\hat{x}_{a(n)}),
\end{aligned}
$$

where equality (a) results from Theorem 1 in Geoffrion (1974) and the definition of convex envelope in the original state variable space (23) leads to equality (e). Since the auxiliary variable λ_n is binary, the convex combination of a set of points in the set $\mathcal{F}'_n$ should satisfy $\mu_n = \hat{\lambda}_{a(n)}$, and thus we obtain equality (b); see the proof of Theorem 3 in Zou et al. (2019). The proof of equality (c) is similar to the one in Lemma 11 for equality (b) of Theorem 1. We can obtain equality (d) by combining constraints $\lambda_n \in \Lambda_n(x_n, \mathbb{P}_n)$ and the cut definition in (19f) to project out the variable λ. □

Theorem 3 shows that adding Lagrangian cuts in the lifted space is equivalent to generating convex envelopes on each element of the feasible region partition and combining them together in the original state variable space. We illustrate the equivalence between Lagrangian cuts in the lifted space and the piecewise convex envelope on the original state variable space with the following example.

Example 2. *We consider a value function Q_n defined on scenario tree $\mathcal{N} = \{1, n\}$ with $a(n) = 1$ and $X_1 = [0, 4]$:*

$$
Q_n(x_1) = \begin{cases} -2x_1 + 2 & \textit{if } x_1 \in [0, 1] \\ 2 & \textit{if } x_1 \in (1, 3) \\ 2x_1 - 6 & \textit{if } x_1 \in [3, 4]. \end{cases}
$$

We introduce a binary variable λ_1 to represent the partition for the space of x_1, $\mathbb{P}_1$: $\lambda_1 = 0$ if $x_1 \in [0, 2]$ and $\lambda_1 = 1$ if $x_1 \in [2, 4]$. Since there is no future value function beyond node n, the lifted value function equals to its lower approximation as:

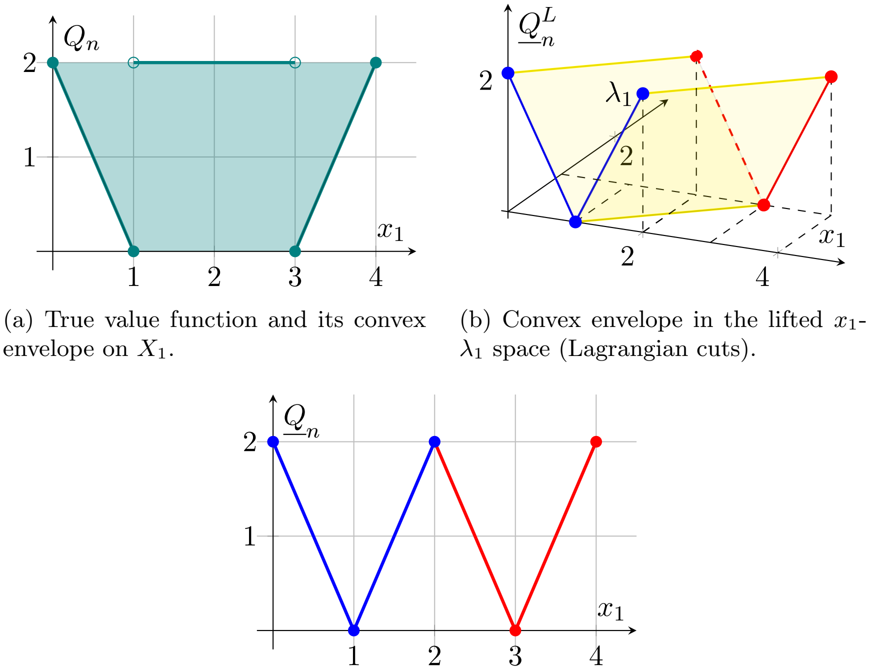
$$
\underline{Q}^{L}_n(x_1, \lambda_1; \mathbb{P}_1) = \begin{cases} -2x_1 + 2 & \textit{if } x_1 \in [0, 1] \textit{ and } \lambda_1 = 0 \\ 2 & \textit{if } x_1 \in (1, 2] \textit{ and } \lambda_1 = 0 \\ 2 & \textit{if } x_1 \in [2, 3) \textit{ and } \lambda_1 = 1 \\ 2x_1 - 6 & \textit{if } x_1 \in [3, 4] \textit{ and } \lambda_1 = 1. \end{cases}
$$

We can generate two Lagrangian cuts on the lifted space as follows:

$$
\theta_n \geq -2x_1 + 4\lambda_1 + 2
$$

$$\theta_n \geq 2x_1 - 4\lambda_1 - 2$$

Fig. 3(a) presents the original value function Q_n. Fig. 3(b) shows how the generated Lagrangian cuts characterize the convex envelope on the lifted space $\mathrm{co}_{\mathfrak{Z}_n(\mathcal{Z}_1,\mathbb{P}_1)}(\underline{Q}_n^L)$. Fig. 3(c) illustrates when we project those Lagrangian cuts back onto the original x_1 space, they form a piecewise convex envelope for the value function Q_n, with two pieces of convex envelopes on $x_1 \in [0,2]$ and $x_1 \in [2,4]$, respectively.



(a) True value function and its convex envelope on X_1.

(b) Convex envelope in the lifted x_1-λ_1 space (Lagrangian cuts).

(c) Piecewise convex envelope in the original x_1 space.

Figure 3: Geometric representation of the value function, lifted Lagrangian cuts, and piecewise convex envelopes.

□

3.2 SDDP-L Algorithm and Global Convergence Results

Theorem 3 shows that the obtained Lagrangian cuts in the lifted space correspond to the piecewise convex envelope in the original state variable space. We observe through Example 2 that when we further partition the state variable, the piecewise convex envelope (Fig. 3(c)) becomes a tighter lower approximation towards the nonconvex value function Q_n than the convex envelope (Fig. 3(a)) over the whole feasible region X_1. This improved tightness echoes with the result shown in Example 1. However, we need to generate the convex envelope on pieces $\mathcal{Z}_1^3$ and $\mathcal{Z}_1^4$ separately in Example 1, but with the added binary indicators λ_n, the convex envelope on the lifted space will incorporate the information for both pieces simultaneously in Example 2. Adding a polynomial number of variables λ_n achieves the equivalent effect of exploring an exponential number of partition elements as the restricted region.

We need to refine the partition of the feasible region to iteratively lift the state variable space by adding binary indicators and tighten the piecewise convex envelope approximation. We first define the *refinement* of a partition.

Definition 3. *For a node $n \in \mathcal{N}$, suppose we have two partitions $\mathbb{P}_n^1$ and $\mathbb{P}_n^2$ indexed by $\mathcal{S}_{n,j}^1$ and $\mathcal{S}_{n,j}^2$ for $j \in \mathcal{J}_n$ respectively. We say partition $\mathbb{P}_n^2$ is a refinement of $\mathbb{P}_n^1$ when*

$$\forall j \in \mathcal{J}_n, \forall \sigma^2 \in \mathcal{S}_{n,j}^2, \exists \sigma^1 \in \mathcal{S}_{n,j}^1 \quad \text{s.t.} \quad \overline{x}_{n,j}^{\sigma^1} \geq \overline{x}_{n,j}^{\sigma^2} \textit{ and } \underline{x}_{n,j}^{\sigma^1} \leq \underline{x}_{n,j}^{\sigma^2}.$$

We can refine a given partition by splitting a subset of its elements. Specifically, we list two natural ideas for refining a given partition within a cutting-plane algorithm framework. Suppose we obtain an incumbent solution $\hat{x}_n, \hat{\lambda}_n$ at a node $n \in \mathcal{N}$. For some $j \in \mathcal{J}_n$, let $\hat{\sigma}$ be the index where $\hat{\lambda}_{n,j}^{\hat{\sigma}} = 1$. We can split the interval $[\underline{x}_{n,j}^{\hat{\sigma}}, \overline{x}_{n,j}^{\hat{\sigma}}]$

- according to the incumbent solution: Let $[\underline{x}_{n,j}^{\hat{\sigma}}, \hat{x}_{n,j}]$ be the interval indexed by $\hat{\sigma}$ and $[\hat{x}_{n,j}, \overline{x}_{n,j}^{\hat{\sigma}}]$ be indexed by $\hat{\sigma}+1$ and we increase the index for all the subsequent intervals.
- according to bisection: Let $[\underline{x}_{n,j}^{\hat{\sigma}}, \frac{\underline{x}_{n,j}^{\hat{\sigma}}+\overline{x}_{n,j}^{\hat{\sigma}}}{2}]$ be the interval indexed by $\hat{\sigma}$ and $[\frac{\underline{x}_{n,j}^{\hat{\sigma}}+\overline{x}_{n,j}^{\hat{\sigma}}}{2}, \overline{x}_{n,j}^{\hat{\sigma}}]$ be indexed by $\hat{\sigma}+1$ and we increase the index for all the subsequent intervals.

With the updated partition, we also need to lift previously generated Lagrangian cuts to fit the current state variable space. We follow a similar scheme as Yang and Morton (2022) to update those inherited cuts. Suppose for some node $n \in \mathcal{N}$, we have a partition $\mathbb{P}_n$ and its refinement $\mathbb{P}'_n$, and for each $j \in \mathcal{J}_n$, their index sets are $\mathcal{S}_{n,j}$ and $\mathcal{S}'_{n,j}$. For an interval indexed by $\sigma \in \mathcal{S}_{n,j}$, we define the set of intervals it splits into as a *descendant set* $\Delta_{n,j}(\sigma) \subseteq \mathcal{S}'_{n,j}$. Then we can update the inherited cut

$$\theta_m \geq v_m + \pi_m^\top x_n + \sum_{j \in \mathcal{J}_n} \sum_{\sigma \in \mathcal{S}_{n,j}} \rho_{m,j}^\sigma \lambda_{n,j}^\sigma$$

to

$$\theta_m \geq v_m + \pi_m^\top x_n + \sum_{j \in \mathcal{J}_n} \sum_{\sigma \in \mathcal{S}_{n,j}} \rho_{m,j}^\sigma \left(\sum_{\sigma' \in \Delta_{n,j}(\sigma)} \lambda_{n,j}^{\sigma'} \right).$$

Algorithm 2 outlines the lifted Lagrangian cutting-plane algorithm, which enhances the approximation of value functions by incorporating lifted Lagrangian cuts on a dynamically lifted state variable space in each iteration. The superscript i represents the iteration index, same as the setup in Algorithm 1.

Algorithm 2 follows the logic of Algorithm 1 but adds the lifting and inherited cut update operations. In iteration i, we need to additionally record the partition information $\mathbb{P}_n^i$ for every

Algorithm 2: Lifted Lagrangian Cutting-plane Algorithm for MS-SMIP.

```
Initialize with $LB \leftarrow -\infty, UB \leftarrow +\infty, i \leftarrow 1$, sets $\mathcal{Z}_{a(n)} \leftarrow X_n$, partition $\mathbb{P}_n^i$ and an lower
  approximation $\{\Phi_n^{L,i}\} \quad \forall n \in \mathcal{N}$;
while stopping criterion is not satisfied do
    /* Forward pass                                                        */
    Sample $M$ scenario paths $\{\mathcal{P}^{i,\omega}\}_{\omega=1}^{M}$ independently;
    for $\omega = 1, \dots, M$ do
        for $n \in \mathcal{P}^{i,\omega}$ do
            Solve forward problem $(P_n^L)$ in model (16) with $x_{a(n)}^i, \lambda_{a(n)}^i, \mathbb{P}_n^i, \mathbb{P}_{a(n)}^i$ and $\Phi_n^{L,i}$. Collect
              the solution $(x_n^i, \lambda_n^i, y_n^i)$;
        end
        $u^\omega \leftarrow \sum_{n \in \mathcal{P}^{i,\omega}} c_n^\top x_n^i + g_n^\top y_n^i$;
    end
    /* Update statisical upper bound                                       */
    Let $\hat{\mu} \leftarrow \frac{1}{M} \sum_{\omega=1}^{M} u^\omega$ and $\hat{\sigma}^2 \leftarrow \frac{1}{M-1} \sum_{\omega=1}^{M} (u^\omega - \hat{\mu})^2$;
    $UB \leftarrow \hat{\mu} + z_{\alpha/2} \hat{\sigma} / \sqrt{M}$;
    /* Backward pass                                                       */
    for $t = T-1, \dots, 1$ do
        for $n \in \mathcal{N}_t$ do
            if $n \in \mathcal{P}^{i,\omega}$ for some $\omega$ then
                Refine the partition to obtain $\mathbb{P}_n^{i+1}$ and update the inherited cuts;
                for $m \in \mathcal{C}(n)$ do
                    Solve the Lagrangian dual problem $(D_m^L)$ in model (18) with
                      $x_n^i, \lambda_n^i, \Phi_m^{L,i+1}, \mathcal{Z}_n, \mathbb{P}_n^{i+1}$, and $\mathbb{P}_m^{i+1}$;
                    Collect $(D_m^L)$'s optimal solution $\pi_m^{*,i}, \rho_m^{*,i}$ and intercept $v_m^{*,i}$;
                    Add a Lagrangian cut defined by coefficients $(v_m^{*,i}, \pi_m^{*,i}, \rho_m^{*,i})$ to $\phi_m^{L,i}$;
                    $L_m \leftarrow L_m + 1$ and $\phi_m^{L,i+1} \leftarrow \phi_m^{L,i}$;
                end
                $\Phi_n^{L,i+1} \leftarrow \{\phi_m^{L,i+1}\}_{m \in \mathcal{C}(n)}$ ;
            else
                $\mathbb{P}_n^{i+1} \leftarrow \mathbb{P}_n^i$ and $\Phi_n^{L,i+1} \leftarrow \Phi_n^{L,i}$;
            end
        end
    end
    /* Update lower bound                                                  */
    Solve forward problem $(P_1^L)$, obtain the optimal value $\underline{V}^i$, and set $LB \leftarrow \underline{V}^i$;
    Update $i \leftarrow i + 1$;
end
```

node $n \in \mathcal{N}$. During the preparation phase of our paper, we noticed there is a new proposal of using ReLU duality to generate tight Lagrangian cuts (Deng and Xie 2024). Our lifted Lagrangian cuts can be considered as a generalization of ReLU Lagrangian cuts. New binary auxiliary variables are introduced in equation (8) of Deng and Xie (2024) to represent whether the decision variable is less than the incumbent solution or not, which can be considered the same as refining the partition to two intervals along one dimension according to the incumbent solution in our paper.

A key question regarding Algorithm 2 is whether it converges to the optimal value V^* and eventually obtains an optimal solution. We first show that Algorithm 2 may run an infinite number of iterations, with the true function value at the solution of each iteration never converging to the optimal value.

Example 3. *We consider the following instance defined on a scenario tree given in Fig. 4(a):*

$$V^* = \min_{x_1 \in \mathbb{R}^2} \sum_{n \in \mathcal{C}(1)} q_{1n} Q_n(x_1) \tag{25a}$$

$$\text{s.t.} \quad x_{1,1} \in [-1, 1], x_{1,2} \in [-1, 1]. \tag{25b}$$

where we set the scenario probability $q_{12} = q_{13} = \frac{1}{2}$. *The value function* Q_n *is defined as:*

$$Q_2(x_1) = \begin{cases} 1, & x_{1,1} < 0 \\ x_{1,2}, & x_{1,1} \geq 0 \end{cases}, \text{ and } Q_3(x_1) = \begin{cases} 1, & x_{1,2} < 0 \\ x_{1,1}, & x_{1,2} \geq 0 \end{cases}.$$

The optimal value is $V^* = 0$ *and there are infinite number of optimal solutions to model* (25)*: (i)* $x^*_{1,1} = x^*_{1,2} = 0$*, (ii) any points in* $\{x_1 \mid 0 \leq x^*_{1,1} \leq 1, x^*_{1,2} = -1\}$*, and (iii) any points in* $\{x_1 \mid x^*_{1,1} = -1, 0 \leq x^*_{1,2} \leq 1\}$*. We implement Algorithm 2 to solve model* (25)*, and its iterations go as follows. For simplicity, we project all obtained Lagrangian cuts to the space of the original state variable space* $[-1, 1] \times [-1, 1]$.

- *Initial Approximation* $(i = 1)$*: Suppose we start with an initial solution* $x_1^1 = [-1\ -1]^\top$*. At this point, we can generate two Lagrangian cuts for each scenario* $n \in \{1, 2\}$*:*

$$\phi_2^1(x_1) = \min\left\{ \theta_2 \;\middle|\; \begin{array}{l} \theta_2 \geq x_{1,2} \\ \theta_2 \geq -2x_{1,1} - 1 \end{array} \right\}, \quad \phi_3^1(x_1) = \min\left\{ \theta_3 \;\middle|\; \begin{array}{l} \theta_3 \geq x_{1,1} \\ \theta_3 \geq -2x_{1,2} - 1 \end{array} \right\}$$

- *First Refinement* $(i = 2)$*: Solving the forward problem yields* $x_1^2 = [-\frac{1}{3}, -\frac{1}{3}]^\top$ *with optimal value* $\underline{V}^2 = -\frac{1}{3}$*. We partition the region according to the incumbent solution* x_1^2*. After the partition refinement, we observe that the optimal solution to the lower approximation problem* (P_1) *can only come from the top right element of the partition, e.g.,* $[-\frac{1}{3}, 1] \times [-\frac{1}{3}, 1]$ *for this iteration. From now on, we only present the projected Lagrangian cuts on the top right element of the partition:*

$$\phi_2^2(x_1) = \min\left\{ \theta_2 \;\middle|\; \begin{array}{l} \theta_2 \geq x_{1,2} \\ \theta_2 \geq -2x_{1,1} - 1 \\ \theta_2 \geq -4x_{1,1} - \frac{1}{3} \end{array} \right\}, \quad \phi_3^2(x_1) = \min\left\{ \theta_3 \;\middle|\; \begin{array}{l} \theta_3 \geq x_{1,1} \\ \theta_3 \geq -2x_{1,2} - 1 \\ \theta_3 \geq -4x_{1,2} - \frac{1}{3} \end{array} \right\}.$$

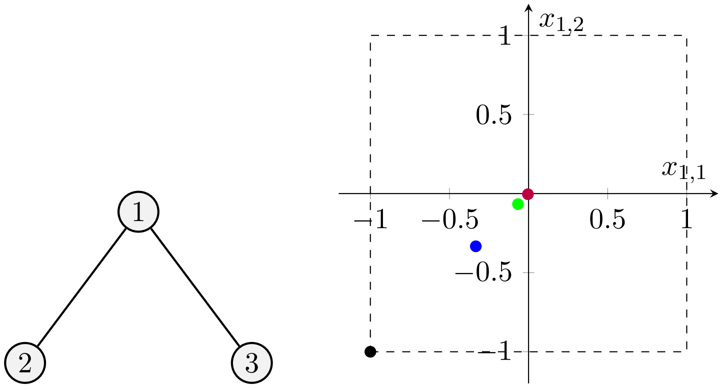
- *Second Refinement* $(i = 3)$*: Solving the forward problem yields* $x_1^3 = [-\frac{1}{15}\ -\frac{1}{15}]^\top$ *with optimal value* $\underline{V}^3 = -\frac{1}{15}$*. We partition the region according to the incumbent solution* x_1^3 *and obtain the projected Lagrangian cuts on the top right element of the partition* $[-\frac{1}{15}, 1] \times [-\frac{1}{15}, 1]$*:*

$$\phi_2^3(x_1) = \min\left\{ \theta_2 \;\middle|\; \begin{array}{l} \theta_2 \geq x_{1,2} \\ \theta_2 \geq -2x_{1,1} - 1 \\ \theta_2 \geq -4x_{1,1} - \frac{1}{3} \\ \theta_2 \geq -16x_{1,1} - \frac{1}{15} \end{array} \right\}, \quad \phi_3^3(x_1) = \min\left\{ \theta_3 \;\middle|\; \begin{array}{l} \theta_3 \geq x_{1,1} \\ \theta_3 \geq -2x_{1,2} - 1 \\ \theta_3 \geq -4x_{1,2} - \frac{1}{3} \\ \theta_3 \geq -16x_{1,2} - \frac{1}{15} \end{array} \right\}.$$

Solving the forward problem results in $x_1^4 = [-\frac{1}{255}\ -\frac{1}{255}]^\top$.

By iteratively applying this process, we observe that the sequence $\{x_1^i\}_{i\in\mathbb{N}}$ *lies on the ray* $\{x_1 \mid x_{1,1} = x_{1,2}, x_{1,1} < 0, x_{1,2} < 0\}$. *It can only approach but never reach the optimal solution* $[0\ 0]^\top$; *see Fig. 4(b) for an illustration.*

Note that for the solution in every iteration x_1^i, *the upper bound it yields* $\sum_{n\in\{2,3\}} \frac{1}{2} Q_n(x_1^i) = 1$, *but the lower bound* $\underline{V}^i$, *obtained from evaluating the piecewise convex envelope at* x_1^i, *is always negative. The gap between each upper bound and the corresponding lower bound is always greater than 1, and the gap between each upper bound and the optimal value is always 1. That is, although the sequence of solutions converges to the optimal solution, each solution is suboptimal, and the optimality gap does not converge to zero.* □



(a) The scenario tree.

(b) Sequence of solutions.

Figure 4: The left figure illustrates the scenario tree and the right figure shows the region $[-1,1]\times[-1,1]$ and the first four solutions in $\{x_1^i\}_{i\in\mathbb{N}}$: $[-1\ -1]^\top$ $[-\frac{1}{3}\ -\frac{1}{3}]^\top$, $[-\frac{1}{15}\ -\frac{1}{15}]^\top$, and $[-\frac{1}{255}\ -\frac{1}{255}]^\top$.

We observe that the sequence of lower bounds indeed converges to the true optimal value and the accumulation point of the solutions converges to the optimal solution in Example 3, despite the constant gap between upper bounds and the optimal value. With the aid of additional assumptions, we can formally state the global convergence result.

Assumption 2. *We assume the following properties when we refine the partition and generate lifted Lagrangian cuts. For every node* $n \in \mathcal{N}$:

- *The size of every element in partition* $\mathbb{P}_n^i$ *diminishes to* 0, *i.e.,* $(\overline{x}_{n,j}^{\sigma,i} - \underline{x}_{n,j}^{\sigma,i}) \to 0, \forall j \in \mathcal{J}_n, \sigma \in \mathcal{S}_{n,j}^i$, *as* $i \to \infty$.
- *The lower approximation* $\phi_n^{L,i}$ *consists of Lagrangian cuts generated at every solution in* $\mathfrak{Z}_n(\mathcal{Z}_{a(n)}, \mathbb{P}_{a(n)}^i)$.

Lemma 4. *In every iteration* $i \in \mathbb{N}$ *with Assumption 2, the projection of lifted lower approximation* $\phi_n^{L,i}(x_{a(n)}, \lambda_{a(n)})$ *to the original state variable space characterizes the piecewise convex envelope, i.e.,* *for every iteration i,*

$$\phi^i_n(x^i_{a(n)};\mathbb{P}^i_n) = \mathrm{co}_{\hat{\mathcal{Z}}^i_{a(n)}}(\underline{Q}_n(\cdot;\Phi^i_n))(x^i_{a(n)}) \;\; \forall (x^i_{a(n)}, \lambda^i_{a(n)}) \in \mathfrak{Z}^i_n(\mathcal{Z}_{a(n)}, \mathbb{P}^i_{a(n)}),$$

where $\hat{\mathcal{Z}}^i_{a(n)}$ is the active region corresponding to $(x^i_{a(n)}, \lambda^i_{a(n)})$.

Lemma 4 comes immediately after part (iii) of Theorem 3. With Assumption 2, every iteration of Algorithm 2 generates a piecewise convex envelope of $\underline{Q}_n$ for all nodes $n \in \mathcal{N}$ in the original state variable space. Next, we show a series of convergence property results of lower approximation functions $\underline{Q}^i_n$ and their piecewise convex envelopes ϕ^i_n. For clarity, we present all results in the original state variable space. We use "$\xrightarrow{e}$" and "$\xrightarrow{p}$" to denote epi-convergence and pointwise convergence. For nodes $n \in \mathcal{N}_T$ and $n = 1$, we might need minor modifications to the following results, but since they do not affect the correctness of the derivation, we present everything for all $n \in \mathcal{N}$ for simplicity.

Lemma 5. *Suppose $\phi^i_m \xrightarrow{e} Q_m$ and $\phi^i_m \xrightarrow{p} Q_m$ for any $n \in \mathcal{N}, m \in \mathcal{C}(n)$. We have that $\underline{Q}^i_n(\cdot;\Phi^i_n)$ pointwise converges to Q_n.*

Proof. By Theorem 7.46 in Rockafellar and Wets (2009), the objective function in $\underline{Q}_n(\cdot;\Phi^i_n)$ follows that

$$c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}\phi^i_m(x_n) \xrightarrow{e} c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}Q_m(x_n)$$

because we assume $\phi^i_m \xrightarrow{e} Q_m$ and $\phi^i_m \xrightarrow{p} Q_m$ for any $n \in \mathcal{N}, m \in \mathcal{C}(n)$. This epi-convergence of the objective function leads to the following pointwise convergence by Theorem 7.33 in Rockafellar and Wets (2009):

$$\lim_{i\to\infty} \underline{Q}_n(x_{a(n)};\Phi^i_n) = Q_n(x_{a(n)}) \qquad \forall x_{a(n)} \in \mathcal{Z}_{a(n)}. \tag{27}$$

This is equivalent to $\underline{Q}_n(\cdot;\Phi^i_n) \xrightarrow{p} Q_n(\cdot)$. □

Proposition 6. *For every iteration $i \in \mathbb{N}$ at every node $n \in \mathcal{N}$, suppose $\underline{Q}_n(\cdot;\Phi^i_n) \xrightarrow{p} Q_n(\cdot)$. The sequence $\{\underline{Q}_n(x_{a(n)};\Phi^i_n)\}$ is asymptotically equi-lower semi-continuous (equi-lsc) everywhere over $x_{a(n)} \in \mathcal{Z}_{a(n)}$.*

Proof. We prove this proposition by contradiction. Suppose there exist $\epsilon > 0$ and $\bar{x}_{a(n)} \in \mathcal{Z}_{a(n)}$ such that for every $\delta > 0$, there is an infinite subsequence of iteration index $N_{a(n)}$ that, for each $i \in N_{a(n)}$, there exists a ball $B(\bar{x}_{a(n)}, \delta)$ with a center $\bar{x}_{a(n)}$ and a radius δ. For any $x_{a(n)} \in B(\bar{x}_{a(n)}, \delta)$ we have

$$\underline{Q}_n(x_{a(n)};\Phi^i_n) < \underline{Q}_n(\bar{x}_{a(n)};\Phi^i_n) - \epsilon. \tag{28}$$

This inequality states that the sequence $\{\underline{Q}_n(x_{a(n)};\Phi_n^i)\}_{i\in\mathbb{N}}$ is not asymptotically equi-lsc at some point $\bar{x}_{a(n)}$ according to the definition in Rockafellar and Wets (2009). Since Q_n is lower semi-continuous, there exists $\delta' > 0$ such that

$$Q_n(x_{a(n)}) \geq Q_n(\bar{x}_{a(n)}) - \epsilon/2 \qquad \forall x_{a(n)} \in B(\bar{x}_{a(n)}, \delta') \tag{29}$$

For each $i \in N_{a(n)}$, we can find a point $\tilde{x}^i_{a(n)} \in B(\bar{x}_{a(n)}, \delta')$ satisfying inequality (28). Let $\tilde{x}_{a(n)} \in B(\bar{x}_{a(n)}, \delta')$ be a limit point of $\tilde{x}^i_{a(n)}$, and define $N'_{a(n)} \subseteq N_{a(n)}$ as the index set such that $\tilde{x}^i_{a(n)} \to \tilde{x}_{a(n)}$ with $i \in N'_{a(n)}$. For some natural number $k > 0$, we have

$$\underline{Q}_n(\tilde{x}^{i+k}_{a(n)};\Phi^i_n) \leq \underline{Q}_n(\tilde{x}^{i+k}_{a(n)};\Phi^{i+k}_n) < \underline{Q}_n(\bar{x}_{a(n)};\Phi^{i+k}_n) - \epsilon \leq Q_n(\bar{x}_{a(n)}) - \epsilon.$$

We obtain the first inequality because of the monotonicity of $\underline{Q}_n$ as we generate more cuts. The second inequality comes from the assumption (28). The last inequality results from $\underline{Q}_n(\cdot;\Phi^{i+k}_n)$ being a lower approximation of Q_n. Since $\underline{Q}_n$ is lower semi-continuous, we have

$$\underline{Q}_n(\tilde{x}_{a(n)};\Phi^i_n) \leq \liminf_{k\to\infty, i+k\in N'_{a(n)}} \underline{Q}_n(\tilde{x}^{i+k}_{a(n)};\Phi^i_n) \leq Q_n(\bar{x}_{a(n)}) - \epsilon. \tag{30}$$

By pointwise convergence of $\underline{Q}_n(\cdot;\Phi^i_n)$, we have that:

$$\lim_{i\in N'_{a(n)}} \underline{Q}_n(\tilde{x}_{a(n)};\Phi^i_n) = Q_n(\tilde{x}_{a(n)}).$$

Therefore, by taking the limit on inequalities (30), we obtain $Q_n(\tilde{x}_{a(n)}) < Q_n(\bar{x}_{a(n)}) - \epsilon$, which contradicts with inequality (29). Thus, the sequence $\{\underline{Q}_n(x_{a(n)};\Phi^i_n)\}_{i\in\mathbb{N}}$ is asymptotically equi-lsc everywhere over $x_{a(n)} \in \mathcal{Z}_{a(n)}$. □

Proposition 7. *For node $n \in \mathcal{N}_T$, $\phi^i_n \xrightarrow{p} Q_n$ and $\phi^i_n \xrightarrow{e} Q_n$.*

Proof. For any $n \in \mathcal{N}_T$, we let $\bar{x}_{a(n)}$ be any point on $\mathcal{Z}_{a(n)}$. We can always find an $x^i_{a(n)}$ in the same partition element as $\bar{x}_{a(n)}$ for every iteration i. Then we can form a convergent sequence $\{x^i_{a(n)}\}_{i\in N_{a(n)}}$ with $\lim_{i\in N_{a(n)}} x^i_{a(n)} = \bar{x}_{a(n)}$ because the diameter of each element in the partition diminishes to zero. One of the following two statements will hold for this sequence:

(i) There exists an infinite subsequence $K \subseteq N_{a(n)}$ such that

$$\phi^i_n(x^i_{a(n)}) \leq Q_n(\bar{x}_{a(n)}) \qquad \forall i \in K.$$

(ii) There exists an index N such that

$$\phi^i_n(x^i_{a(n)}) > Q_n(\bar{x}_{a(n)}) \qquad \forall i \in N_{a(n)}, i > N$$

For case (i), we define $\tilde{x}^i_{a(n)} \in \arg\min_{x_{a(n)} \in \hat{\mathcal{Z}}^i_{a(n)}} \phi^i_n(x_{a(n)})$. As $\tilde{x}^i_{a(n)}$ is the minimizer of the convex envelope of Q_n over $\hat{\mathcal{Z}}^i_{a(n)}$, we have

$$Q_n(\tilde{x}^i_{a(n)}) = \phi^i_n(\tilde{x}^i_{a(n)}) \leq \phi^i_n(x^i_{a(n)}) \leq Q_n(\bar{x}_{a(n)}). \tag{31}$$

Since the sequence $\{x^i_{a(n)}\}_{i \in N_{a(n)}}$ converges to $\bar{x}_{a(n)}$ and the diameter of any element in the partition diminishes to 0, we obtain

$$\|\tilde{x}^i_{a(n)} - \bar{x}_{a(n)}\| \leq \|\tilde{x}^i_{a(n)} - x^i_{a(n)}\| + \|x^i_{a(n)} - \bar{x}_{a(n)}\| \to 0 \quad \text{as } i \to \infty.$$

Taking the limit inferior on each term of (31), we obtain $\liminf_{i\in K} \phi^i_n(x^i_{a(n)}) = Q_n(\bar{x}_{a(n)})$ because $Q_n(\bar{x}_{a(n)}) \leq \liminf_{i\in K} Q_n(\tilde{x}^i_{a(n)}) \leq \liminf_{i\in K} \phi^i_n(x^i_{a(n)}) \leq Q_n(\bar{x}_{a(n)})$. For indices $i \notin K$, we know that $\phi^i_n(x^i_{a(n)}) > Q_n(\bar{x}_{a(n)})$. Together they imply $\liminf_{i\in N_{a(n)}} \phi^i_n(x^i_{a(n)}) = Q_n(\bar{x}_{a(n)})$.

For case (ii), by assumption,

$$\begin{aligned} &\phi^i_n(x^i_{a(n)}) > Q_n(\bar{x}_{a(n)}) \qquad \forall i \in N_{a(n)}, i > N, \qquad \text{and} \\ &\liminf_{i \in N_{a(n)}} \phi^i_n(x^i_{a(n)}) \geq Q_n(\bar{x}_n). \end{aligned}$$

Combining both cases, we conclude that

$$\liminf_{i \in N_{a(n)}} \phi^i_n(x^i_{a(n)}) \geq Q_n(\bar{x}_{a(n)}) \text{ for all } n \in \mathcal{N}_T.$$

Since this holds for any $\bar{x}_{a(n)} \in \mathcal{Z}_{a(n)}$, by Proposition 7.2 in Rockafellar and Wets (2009), we establish the epi-convergence: $\phi^i_n \xrightarrow{e} Q_n, \forall n \in \mathcal{N}_T$.

Next, we prove that $\phi^i_n \xrightarrow{p} Q_n$ for all $n \in \mathcal{N}_T$. Given any $\hat{x}_{a(n)} \in \mathcal{Z}_{a(n)}$, we define a new $\tilde{x}^i_{a(n)}$ as $\tilde{x}^i_{a(n)} \in \arg\min_{x_{a(n)} \in \hat{\mathcal{Z}}_{a(n)}} \phi^i_n(x_{a(n)})$, where $\hat{\mathcal{Z}}_{a(n)}$ is the active restricted region corresponding to $\hat{x}_{a(n)}$. By definition of $\tilde{x}^i_{a(n)}$, we have

$$Q_n(\hat{x}_{a(n)}) \geq \phi^i_n(\hat{x}_{a(n)}) \geq \phi^i_n(\tilde{x}^i_{a(n)}) = Q_n(\tilde{x}^i_{a(n)}).$$

Taking the limit inferior for every term in the above inequalities, we obtain

$$Q_n(\hat{x}_{a(n)}) \geq \lim_{i\to\infty} \phi^i_n(\hat{x}_{a(n)}) \geq \liminf_{i\to\infty} \phi^i_n(\tilde{x}^i_{a(n)}) \geq \liminf_{i\to\infty} Q_n(\tilde{x}^i_{a(n)}).$$

Since function Q_n is lower semi-continuous and $\tilde{x}^i_{a(n)} \to \hat{x}_{a(n)}$ as $i \to \infty$, we have $\liminf_{i\to\infty} Q_n(\tilde{x}^i_{a(n)}) \geq Q_n(\hat{x}_{a(n)})$, and by combining the inequalities above, we obtain

$$Q_n(\hat{x}_{a(n)}) \geq \lim_{i\to\infty} \phi^i_n(\hat{x}_{a(n)}) \geq Q_n(\hat{x}_{a(n)}) \qquad \forall \hat{x}_{a(n)} \in \mathcal{Z}_{a(n)},$$

i.e., the pointwise convergence result $\phi^i_n \xrightarrow{p} Q_n, \forall n \in \mathcal{N}_T$. □

Lemma 5, Proposition 6 and 7 provide required convergence results for the lower approximation functions obtained in our algorithm. Now we formally prove the convergence of Algorithm 2.

Theorem 8. *With Assumption 2, Algorithm 2 generates a sequence of optimal solution* $\{(x_1^i, y_1^i)\}_{i\in\mathbb{N}}$ *to the root node problem (17), and with probability one we have*

$$V^* = \lim_{i\to\infty} \underline{V}^i = \lim_{i\to\infty} \left(c_1^\top x_1^i + g_1^\top y_1^i + \sum_{m\in\mathcal{C}(1)} \underline{Q}_m\left(x_1^i; \Phi_m^i\right) \right),$$

and every accumulation point of the sequence $\{(x_1^i, y_1^i)\}_{i\in\mathbb{N}}$*,* (x_1^*, y_1^*)*, is an optimal solution to model (2).*

Proof. We first prove that $\phi_n^i \xrightarrow{p} Q_n$ and $\phi_n^i \xrightarrow{e} Q_n$ for every $n \in \mathcal{N}\backslash\{1\}$ via backward induction. The convergence results for node $n \in \mathcal{N}_T$ have been shown in Proposition 7.

We assume that $\phi_n^i \xrightarrow{p} Q_n$ and $\phi_n^i \xrightarrow{e} Q_n$ for all nodes beyond stage t. We aim to show that $\phi_n^i \xrightarrow{p} Q_n$ and $\phi_n^i \xrightarrow{e} Q_n$ for nodes $n \in \mathcal{N}_{t-1}$. Under this assumption, Lemma 5 leads to the pointwise convergence result that $\underline{Q}_n(\cdot; \Phi_n^i) \xrightarrow{p} Q_n(\cdot)$ and Proposition 6 shows that the sequence $\{\underline{Q}_n(x_{a(n)}; \Phi_n^i)\}_{i\in\mathbb{N}}$ is asymptotically equi-lsc everywhere over $x_{a(n)} \in \mathcal{Z}_{a(n)}$. Now we aim to use those results about $\underline{Q}_n(\cdot; \Phi_n^i)$ as a bridge to show that $\phi_n^i \xrightarrow{p} Q_n$ and $\phi_n^i \xrightarrow{e} Q_n$ for nodes $n \in \mathcal{N}_{t-1}$.

For any $\hat{x}_{a(n)} \in \mathcal{Z}_{a(n)}$ and $\epsilon > 0$, there exists a $\delta_n(\epsilon, \hat{x}_{a(n)}) > 0$ such that, for all $x_{a(n)} \in B(\hat{x}_{a(n)}, \delta_n(\epsilon, \hat{x}_{a(n)}))$, there exists an infinite sequence $N_{a(n)}$ for which by lower semi-continuity of $\underline{Q}_n$, we have

$$\underline{Q}_n(x_{a(n)}; \Phi_n^i) \geq \underline{Q}_n(\hat{x}_{a(n)}; \Phi_n^i) - \epsilon \qquad \forall i \in N_{a(n)}. \tag{32}$$

Suppose we define the active restricted region for $\hat{x}_{a(n)}$ as $\hat{\mathcal{Z}}_{a(n)}$ defined in (24). Note that here we perform our analysis on the original state variable space so $\hat{x}_{a(n)}$ may lie on the boundary of an element within the partition $\mathbb{P}_n^i$, resulting in different $\hat{\lambda}_{a(n)}^i$. However, it actually expands the selection of $\hat{\mathcal{Z}}_{a(n)}$, which does not affect the following result. For a sufficiently large $i \in N_{a(n)}$, we can find an active restricted region $\hat{\mathcal{Z}}_{a(n)} \subseteq B(\hat{x}_{a(n)}, \delta_n(\epsilon, \hat{x}_{a(n)}))$ such that

$$\phi_n^i(\hat{x}_{a(n)}) \geq \min_{x_{a(n)}\in\hat{\mathcal{Z}}_{a(n)}} \phi_n^i(x_{a(n)}) = \min_{x_{a(n)}\in\hat{\mathcal{Z}}_{a(n)}} \underline{Q}_n(x_{a(n)}; \Phi_n^i) \geq \underline{Q}_n(\hat{x}_{a(n)}; \Phi_n^i) - \epsilon. \tag{33}$$

We obtain the first inequality because $\hat{x}_{a(n)}$ is a feasible solution to the minimization problem in the second term. The equality holds since the minimum of a lower semi-continuous function equals to the minimum of its convex envelope on a compact set and we obtain such a convex envelope on $\hat{\mathcal{Z}}_{a(n)}$ by Lemma 4. The last inequality comes from inequality (32).

Taking limits for all terms in (33) with $i \in N_{a(n)}, i \to \infty$ and $\epsilon \to 0$, we yield the point convergence result $\phi_n^i \xrightarrow{p} Q_n$ because $\underline{Q}_n(\cdot; \Phi_n^i) \xrightarrow{p} Q_n(\cdot)$ in Lemma 5. Since $\underline{Q}_n(\hat{x}_{a(n)}; \Phi_n^i) \geq \phi_n^i(\hat{x}_{a(n)})$ for any $i \in \mathbb{N}$, by the asymptotically equi-lsc property of $\{\underline{Q}_n(x_{a(n)}; \Phi_n^i)\}_{i\in\mathbb{N}}$ everywhere over $x_{a(n)} \in \mathcal{Z}_{a(n)}$, we have

$$\min_{x_{a(n)}\in B(\hat{x}_{a(n)},\delta)} \underline{Q}_n(x_{a(n)}; \Phi_n^i) \geq \underline{Q}_n(\hat{x}_{a(n)}; \Phi_n^i) - \epsilon \geq \phi_n^i(\hat{x}_{a(n)}) - \epsilon. \tag{34}$$

On each element of the partition $\mathbb{P}_{a(n)}^i$, function ϕ_n^i characterizes the convex envelope of $\underline{Q}_n(\cdot; \Phi_n^i)$. We let $\tilde{\mathbb{P}}_{a(n)}^i \subseteq \mathbb{P}_{a(n)}^i$ where each element in $\tilde{\mathbb{P}}_{a(n)}^i$ has a non-empty intersection with $\hat{\mathcal{Z}}_{a(n)}$. We can obtain the following equality

$$\begin{aligned}
\min_{x_{a(n)}\in B(\hat{x}_{a(n)},\delta)} \underline{Q}_n(x_{a(n)};\Phi^i_n) &= \min_{\mathfrak{p}\in\tilde{\mathbb{P}}^i_{a(n)}} \min_{x_{a(n)}\in\mathfrak{p}\cap B(\hat{x}_{a(n)},\delta)} \underline{Q}_n(x_{a(n)};\Phi^i_n) && (35a)\\
&= \min_{\mathfrak{p}\in\tilde{\mathbb{P}}^i_{a(n)}} \min_{x_{a(n)}\in\mathfrak{p}\cap B(\hat{x}_{a(n)},\delta)} \phi^i_n(x_{a(n)}) && (35b)\\
&= \min_{x_{a(n)}\in B(\hat{x}_{a(n)},\delta)} \phi^i_n(x_{a(n)}). && (35c)
\end{aligned}$$

Therefore, $\min_{x_{a(n)}\in B(\hat{x}_{a(n)},\delta)} \phi^i_n(x_{a(n)}) \geq \phi^i_n(\hat{x}_{a(n)}) - \epsilon$ for any given $\hat{x}_{a(n)} \in \mathcal{Z}_{a(n)}$, and the sequence $\{\phi^i_n(x_{a(n)})\}$ is asymptotically equi-lsc everywhere over $x_{a(n)} \in \mathcal{Z}_{a(n)}$. By Theorem 7.10 in Rockafellar and Wets (2009), we have shown $\phi^i_n \xrightarrow{e} Q_n$.

By induction, we prove that $\phi^i_n \xrightarrow{p} Q_n$ and $\phi^i_n \xrightarrow{e} Q_n$ for all $n \in \mathcal{N}\backslash\{1\}$. Theorem 7.33 in Rockafellar and Wets (2009) holds for the first-stage problem because we have $\phi^i_m \xrightarrow{e} Q_m$ for all $m \in \mathcal{C}(1)$ and both ϕ^i_m and Q_m are bounded and lower semi-continuous, which leads to $V^* = \lim_{i\to\infty} \underline{V}^i$.

For the optimal solution convergence, we can apply the same logic used in the proof of Theorem 2 in Appendix B to show that the objective value at (x^*_1, y^*_1) is bounded by V^* both from below and from above. □

3.3 Sparse Lifted Lagrangian Cuts

In every iteration of Algorithm 2, we add binary variables to lift the state variable space. The number of binary variables added depends on how we refine the partition. For example, if we refine the partition at node $n \in \mathcal{N}$ according to the incumbent solution, we may add as many as $|\mathcal{J}_n|$ binary variables in one iteration. This may result in a high-dimensional lifted space, on which we need to obtain optimal dual multipliers and generate cutting planes. In order to generate a lifted Lagrangian cut, the convex optimization solution methods, such as the level method (Lemaréchal et al. 1995) and the subgradient method (Fisher 2004), are widely deployed to solve the Lagrangian dual model (18). The key component in those convex optimization methods is to search for an optimal dual multiplier, and a high-dimensional search space significantly affects its efficiency.

Therefore, we propose to generate *sparse Lagrangian cuts* on a space with the original state variables, x_n, and only the binary variables representing the active element within the current partition. To generate a sparse Lagrangian cut, we update the objective function in (19a) and solve an updated Lagrangian dual with $\mathcal{L}^L_n$ defined as follows. Suppose we let $\hat{\sigma}_{a(n),j}$ denote the index representing the active element in $\mathcal{S}_{a(n),j}$, where the current $\hat{\lambda}^{\hat{\sigma}_{a(n),j}}_{a(n),j} = 1$ for every $n \in \mathcal{N}, j \in \mathcal{J}_n$.

$$\begin{aligned}
&\mathcal{L}^L_n(\pi_n, \rho_n; \hat{x}_{a(n)}, \hat{\lambda}_{a(n)}, \Phi^L_n, \mathcal{Z}_{a(n)}, \mathbb{P}_n, \mathbb{P}_{a(n)}) = \\
&\min \quad c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}\theta_m + \pi_n^\top(\hat{x}_{a(n)} - z_n) +
\end{aligned}$$

$$\sum_{j\in\mathcal{J}_n} \rho_{n,j}^{\hat{\sigma}_{a(n),j}} (\hat{\lambda}_{a(n),j}^{\hat{\sigma}_{a(n),j}} - \mu_{n,j}^{\hat{\sigma}_{a(n),j}}) \tag{36}$$

$$\text{s.t.} \quad \text{constraints (19b)} - \text{(19f)}.$$

Solving the updated Lagrangian dual yields the optimal solution π_n^*, ρ_n^*, and intercept v_n^*, which can further generate a Lagrangian cut in the same form of (20). We name this improved cut "sparse" because there is only one non-zero element for each $j \in \mathcal{J}_n$ within ρ_n^*. It is equivalent to padding the value of all ρ coefficients for inactive elements with zero. The updated formulation accelerates the convex optimization method to solve the Lagrangian dual problem, since we only need to search on a significantly lower-dimensional space of (π, ρ) now.

The sparse Lagrangian cut retains the validity and tightness property of a regular Lagrangian cut since the dual multiplier to generate a sparse lifted Lagrangian cut is a feasible and optimal solution to the Lagrangian dual problem. Therefore, generating sparse Lagrangian cuts in the framework of Algorithm 2 does not affect the asymptotic convergence to the global optimum. We present the theoretical results of sparse lifted Lagrangian cuts in Appendix C.

4 Enhanced Lagrangian Cut

In this section, we explore geometric properties that may affect the strength of Lagrangian cuts. We further utilize them to propose enhancement techniques for the cut generation process so that we can achieve better computational efficiency. Those techniques can be applied in a unified problem framework regardless of whether we introduce lifting variables. We first present such a unified problem framework for notational clarity.

Throughout this section, for the node $n \in \mathcal{N}$, we let x_n be the state variable, y_n be the local variable, and z_n be the auxiliary variable introduced to model the nonanticipativity constraint. The set $\mathcal{Z}_{a(n)}$ specifies the region which the input state variable $x_{a(n)}$, or the auxiliary variable z_n, should lie within. Note that this is the same setup as in Section 2. Therefore, the forward pass problem here has the same formulation as (P_n) in model (4) and the backward pass problem is the same as (D_n) in model (7). We repeat those model formulations here to help readers find a quick reference.

$$(P_n) \qquad \underline{Q}_n(x_{a(n)}; \Phi_n) =$$

$$\min \quad c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}\phi_m(x_n) \tag{37a}$$

$$\text{s.t.} \quad W_n x_n + B_n y_n = T_n z_n + d_n \tag{37b}$$

$$x_n \in X_n, y_n \in Y_n \tag{37c}$$

$$z_n = x_{a(n)}. \tag{37d}$$

$$(D_n) \quad \max_{\pi_n} \quad \mathcal{L}_n\left(\pi_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right), \tag{38}$$

where

$$\begin{aligned}
& \mathcal{L}_n\left(\pi_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) = \\
& \min_{x_n, y_n, z_n} \quad c_n^\top x_n + g_n^\top y_n + \sum_{m \in \mathcal{C}(n)} q_{nm} \theta_m + \pi_n^\top (\hat{x}_{a(n)} - z_n) && \text{(39a)} \\
& \quad \text{s.t.} \quad W_n x_n + B_n y_n = T_n z_n + d_n && \text{(39b)} \\
& \qquad x_n \in X_n, y_n \in Y_n, z_n \in \mathcal{Z}_{a(n)} && \text{(39c)} \\
& \qquad \theta_m \geq \underline{\theta}_m \qquad \forall m \in \mathcal{C}(n) && \text{(39d)} \\
& \qquad \theta_m \geq v_m^\ell + {\pi_m^\ell}^\top x_n \quad \forall m \in \mathcal{C}(n), \ell = 1, \ldots, L_m. && \text{(39e)}
\end{aligned}$$

This framework can easily adapt to models with lifting variables in Section 3, as the unified state variable x_n include previous x_n and λ_n, the unified auxiliary variables z_n include previous z_n and μ_n, the unified feasible region X_n is characterized by previous X_n and $\Lambda_n(x_n, \mathbb{P}_n)$, and the unified restricted region should correspond to the set $\mathfrak{Z}_n(\mathcal{Z}_{a(n)}, \mathbb{P}_{a(n)})$.

4.1 Extreme Points and Multiple Optimal Dual Multipliers

When all state variables are binary, the solution to (P_n) in every iteration is at an extreme point of the convex hull of the restricted region for the auxiliary variable, which makes Lagrangian cuts a tight lower approximation for value function $\underline{Q}_n$ (Zou et al. 2019). When the state variables are mixed-integer, we lose this property and can only achieve tightness against the convex envelope of $\underline{Q}_n$ over the restricted region $\mathcal{Z}_{a(n)}$. That said, we can obtain a similar tightness result when we have a solution $\hat{x}_{a(n)}$ located at an extreme point of $\operatorname{conv}(\mathcal{Z}_{a(n)})$:

Proposition 9. *If $\hat{x}_{a(n)}$ is an extreme point of* $\operatorname{conv}(\mathcal{Z}_{a(n)})$*, we have*

$$v_n^* + {\pi_n^*}^\top \hat{x}_{a(n)} = \underline{Q}_n(\hat{x}_{a(n)}; \Phi_n). \tag{40}$$

We show the proof of this tightness result in Appendix D. The lifting operations create binary variables and smaller hyperrectangles within the partition, both of which make it more likely to obtain a solution to (P_n) at an extreme point of $\operatorname{conv}(\mathcal{Z}_{a(n)})$. This intuition explains why Lagrangian cuts, together with the lifting scheme, can quickly close out the optimality gap. However, when we obtain an extreme point solution $\hat{x}_{a(n)}$, the optimal solutions to (D_n) are generally not unique. We can generate multiple Lagrangian cuts, all of which are tight at $\hat{x}_{a(n)}$. Let $\Pi_n(\hat{x}_{a(n)})$ be the set of optimal solutions to (D_n) in (38). The following example demonstrates that different Lagrangian cuts can have significantly different approximation effects.

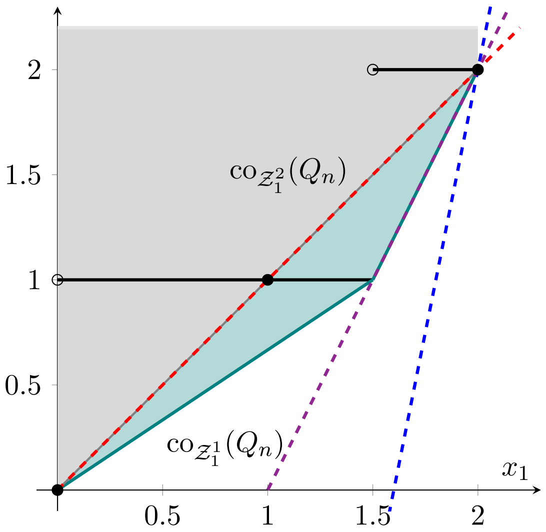
Example 4. *We consider the same problem as Example 1 and let $\hat{x}_1 = 2$. Fig. 5(a) shows the form of Lagrangian cuts and the range of optimal dual solutions, $\Pi_n(\hat{x}_1)$, for two restricted regions, $\mathcal{Z}_1^1$ and $\mathcal{Z}_1^2$. In Fig. 5(b), we visualize the Lagrangian cuts, with each cut color-coded for clarity to distinguish their properties and approximation effects.*

The purple cut represents a facet-defining Lagrangian cut generated under $\mathcal{Z}_1^1$. Similarly, the red cut is the facet-defining Lagrangian cut generated under $\mathcal{Z}_1^2$. Not only are these cuts tight at $\hat{x}_1$, but they can also characterize the tight lower approximation elsewhere within the restricted region. In contrast, the blue cut is valid and tight at $\hat{x}_1$, but is loose elsewhere. Generating loose cuts may significantly increase the number of iterations for the cutting-plane algorithm and lead to computational intractability; e.g., the integer L-shaped cuts (Laporte and Louveaux 1993). Therefore, we aim to generate strong cuts by selecting the right dual multiplier from the set $\Pi_n(\hat{x}_{a(n)})$. □

For $\pi_n \in \Pi_n(\hat{x}_1)$, a Lagrangian cut is given by: $\theta_n \geq \pi_n x_1 + 2(1 - \pi_n)$.

	$\mathcal{Z}_1^1$	$\mathcal{Z}_1^2$
$\Pi_n(\hat{x}_1)$	$[2, +\infty)$	$[1, +\infty)$

(a) Lagrangian cut coefficient information at $\hat{x}_1 = 2$.



(b) Illustration of different valid Lagrangian cuts.

Figure 5: Multiple optimal dual multipliers to problem (D_n) may result in different valid Lagrangian cuts in Example 4.

4.2 Pareto-optimal Lagrangian Cut (PLC)

To distinguish the strength of different cuts, Magnanti and Wong (1981) proposed a method to generate Pareto-optimal cuts for Benders decomposition. No cut is stronger than the Pareto-optimal cut over all points in the feasible region. Thus, Pareto-optimal cuts help generate a good lower approximation and accelerate the cutting-plane algorithm. The method in Magnanti and Wong (1981) aims to maximize the cut value at a core point, which is any point contained in the relative interior of the convex hull of the feasible region.

We propose a two-phase methodology to generate such *Pareto-optimal Lagrangian cuts (PLC)*.

Phase I ensures Lagrangian cuts' tightness at $\hat{x}_{a(n)}$ for the convex envelope $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n)\right)$, while Phase II maximizes the cut value at a core point $\tilde{x}_{a(n)} \in \mathrm{relint}(\mathrm{conv}(\mathcal{Z}_{a(n)}))$, the relative interior of the convex hull of the restricted region $\mathcal{Z}_{a(n)}$.

To ensure the tightness guarantee, we must select the dual multiplier from the set of optimal solutions to the Lagrangian dual problem $\Pi_n(\hat{x}_{a(n)})$. In addition, we need to represent the cut value at the core point, with which we replace the objective function of the Lagrangian dual problem. Combining those two factors, we propose the following *cut-generation convex program (CGCP)* to replace (D_n) in model (38) to obtain the cut coefficients:

$$
\begin{aligned}
(D_n) \quad \max_{\pi_n} \quad & \mathcal{L}_n\left(\pi_n;\hat{x}_{a(n)},\Phi_n,\mathcal{Z}_{a(n)}\right) + \pi_n^\top(\tilde{x}_{a(n)} - \hat{x}_{a(n)}) && \text{(41a)}\\
\text{s.t.} \quad & \mathcal{L}_n\left(\pi_n;\hat{x}_{a(n)},\Phi_n,\mathcal{Z}_{a(n)}\right) \geq \mathcal{L}_n\left(\hat{\pi}_n;\hat{x}_{a(n)},\Phi_n,\mathcal{Z}_{a(n)}\right) - \epsilon, && \text{(41b)}
\end{aligned}
$$

where $\epsilon > 0$ is a tolerance parameter and $\hat{\pi}_n$ is any optimal solution in the set $\Pi_n(\hat{x}_{a(n)})$. The objective function is the sum of a piecewise linear concave function and a linear function of π_n, and thus concave in π_n. Constraint (41b) formulates a convex set as the left-hand side of the inequality is a piecewise linear concave function of π_n. Characterizing the set $\Pi_n(\hat{x}_{a(n)})$ presents significant challenges, but we only need the optimal value of max $\mathcal{L}_n\left(\pi_n;\hat{x}_{a(n)},\Phi_n,\mathcal{Z}_{a(n)}\right)$ without specifying the exact $\hat{\pi}_n$ in CGCP. Therefore, in Phase I, we can solve the Lagrangian dual model (38) to obtain the objective value. Then we can plug this optimal value in the constraint (41b) and solve model (41) with convex optimization methods. We show that the Pareto-optimal Lagrangian cut can simultaneously achieve validity, tightness, and Pareto-optimality.

Theorem 10. *With $\epsilon = 0$, the Lagrangian cut generated by solving CGCP (41) is valid and tight at $\hat{x}_{a(n)}$ for $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n)\right)$. Moreover, it is Pareto-optimal with respect to $\mathrm{conv}(\mathcal{Z}_{a(n)})$.*

We show the proof of Theorem 10 in Appendix D. Note that we need to solve two convex optimization models to generate a PLC, one to obtain the value of $\mathcal{L}_n\left(\hat{\pi}_n;\hat{x}_{a(n)},\Phi_n,\mathcal{Z}_{a(n)}\right)$ and the other to solve model (41). This is consistent with Magnanti and Wong (1981) where the subproblem needs to be solved twice. However, if we use specific refinement schemes, e.g., splitting the active element in the partition along every dimension $j \in \mathcal{J}_n$ according to the incumbent solution, the incumbent solution becomes an extreme point of $\mathrm{conv}(\mathcal{Z}_{a(n)})$ in the next iteration, and we can use the result of Proposition 9 to replace $\mathcal{L}_n\left(\pi_n;\hat{x}_{a(n)},\Phi_n,\mathcal{Z}_{a(n)}\right)$ by $\underline{Q}_n(\hat{x}_{a(n)};\Phi_n)$. We discuss the details of computational techniques to generate a PLC in Section 5.

4.3 Square Minimization Cut (SMC)

Generating a PLC requires identifying a core point within the relative interior of the convex hull of the restricted region $\mathcal{Z}_{a(n)}$. The selection of core points will directly affect the geometry of cuts, and there is no rule to consistently guarantee the efficacy of cuts. Here, we propose an approach that generates an enhanced Lagrangian cut without identifying a core point.

In Example 4, we observe that among all tight cuts obtained at an extreme point, we would prefer "flat" cuts (purple line) over "steep" cuts (blue line). This observation motivates us to find a measure of "flatness" to differentiate cuts. Naturally, we use the two-norm of dual multiplier π_n^* as such a measure. We can visualize this phenomenon with Example 4: The quantity $\frac{1}{\sqrt{{\pi_n}^\top \pi_n + 1}}$ reflects the cosine of the acute angle formed by the cut and the horizontal axis. To generate a "flat" cut, we would like to minimize the $L2$-norm of π_n so that such an acute angle is small. The idea can be extended to high-dimensional cases, and we can formulate the following CGCP:

$$
\begin{aligned}
(D_n) \quad \min_{\pi_n} \quad & \|\pi_n\|_2 && \text{(42a)}\\
\text{s.t.} \quad & \mathcal{L}_n\left(\pi_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) \geq \mathcal{L}_n\left(\hat{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) - \epsilon, && \text{(42b)}
\end{aligned}
$$

Model (42) remains a convex optimization problem since the objective function is minimizing a convex function of π_n, and constraint (42b) remains the same to formulate a convex set. We name the cut *square minimization cut (SMC)*, with cut coefficients obtained from solving model (42). SMC can achieve validity and tightness but may fail to be Pareto-optimal, for which we show an example in Appendix D. Yet the computational results in Section 5 show its computational efficiency. We close this section with the following example that compares the vanilla Lagrangian cut (LC), PLC, and SMC.

Example 5. *We consider the same problem as Example 1 and let $\hat{x}_1 = 1.5$ and $X_1 = [0, 2] = \mathcal{Z}_1$. Fig. 6(a) shows the information of vanilla and enhanced Lagrangian cuts generated at $\hat{x}_1 = 1.5$. These cuts are visualized and color-coded accordingly in Fig. 6(b). We can see that the PLC and SMC generated for this case are facet-defining, while the LC may not be.* □

5 Numerical Experiments

We conduct numerical experiments to test the performance of proposed algorithms and enhancements to Lagrangian cuts for MS-SMIP problems (1). We build two large-scale test cases based on real-world applications: (i) a power generation expansion planning problem with general integer state variables, and (ii) a stochastic unit commitment problem with binary and continuous state variables. These experiments aim to address the following key questions:

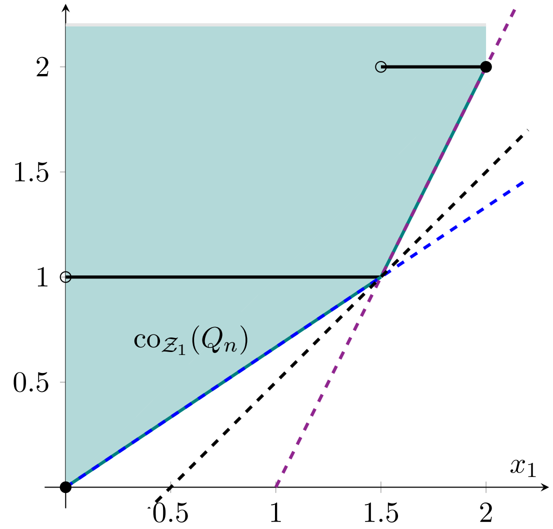
(i) How effective is our SDDP-L algorithm (Algorithm 2) compared to SDDP with Lagrangian cuts (Algorithm 1) and SDDiP in Zou et al. (2019) with a binary representation for all state variables, for MS-SMIP with general mixed-integer state variables?

(ii) To what extent do the enhancement techniques proposed in Section 4 improve the performance of the Lagrangian cut?

Type	$\tilde{x}_1$	Cut Formulation
LC	-	$\theta_n \geq \pi_n x_1 - \frac{3}{2}\pi_n + 1$
PLC	1.0	$\theta_n \geq \frac{2}{3}x_1$
	1.8	$\theta_n \geq 2x_1 - 2$
SMC	-	$\theta_n \geq \frac{2}{3}x_1$

For LC, $\pi_n \in [\frac{2}{3}, 2]$.

(a) Vanilla and enhanced Lagrangian cut coefficient information at $\hat{x}_1 = 1.5$.



(b) Illustration of vanilla and enhanced Lagrangian cuts, where we set $\pi_n = 1$ for LC.

Figure 6: Comparison of LC, PLC, and SMC.

(iii) What insights can be drawn about the features and mechanisms of different cuts and algorithms across the test cases?

All optimization models are formulated using JuMP (Dunning et al. 2017) in Julia v1.11 (Bezanson et al. 2017) and solved with Gurobi v11.1 (Gurobi Optimization, Inc. 2014). The experiments utilize 5 threads on a workstation with an M3 MAX CPU and 48 GB of memory. We set $M = 500$, but only generate cuts for the solution obtained from the first sample path. The parameter ϵ in Section 4 is set to 10^{-2}. We return the obtained cut coefficients from the iteration before we encounter numerical issues, as any dual multiplier π_n, together with $\mathcal{L}_n(\pi_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)})$, results in a valid cut. We let all test instances run for $3,600$ seconds, recording upper and lower bounds and the time and iteration number when the relative gap between the statistical upper bound and the lower bound reaches 1%. For PLC, the core point is fixed at $\tilde{x}_{n,j} = \frac{\overline{x}_{n,j} + \underline{x}_{n,j}}{2}$ for all $n \in \mathcal{N}, j \in \mathcal{J}_n$.

The Lagrangian dual problems (D_n) are solved using the level method in Lemaréchal et al. (1995). We set the convergence tolerance for the level method as 10^{-2}, with a maximum of 200 iterations. As mentioned in Section 4, generating a Lagrangian cut requires solving the Lagrangian dual problem twice, which may be time-consuming. In the experiment, we use the optimal value $\underline{Q}_n(\hat{x}_{a(n)}; \Phi_n)$ (or $\underline{Q}_n^L(\hat{x}_{a(n)}; \Phi_n)$) to replace $\mathcal{L}_n(\bar{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)})$ so we do not have to solve the Lagrangian dual problem to obtain $\mathcal{L}_n(\bar{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)})$. Since $\mathcal{L}_n(\bar{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)})$ only provides a lower approximation of $\underline{Q}_n$ (or $\underline{Q}_n^L$), CGCPs may run into infeasibility. When we encounter such infeasibility, we terminate the level method and output the cut coefficients from the previous iteration. This technique results in a valid Lagrangian cut, following the same reasoning when encountering numerical issues.

5.1 Generation Expansion Planning

We consider a generation expansion planning problem (GEP) with b types of expansion technologies. At a node of the scenario tree $n \in \mathcal{N}$, we decide the number of each type of generators built, w_n, total electricity generation by each type of generator, y_n, and the outsourced demand, s_n, penalized by a rate p. We let the state variable x_n represent the cumulative number of different types of generators built until node n. We present the extensive formulation of this multi-stage GEP as follows:

$$
\begin{aligned}
\min \quad & \sum_{n \in \mathcal{N}} p_n \left(c_n^\top w_n + g_n^\top y_n + p s_n \right) & & \\
\text{s.t.} \quad & x_n \leq \bar{u} & \forall n \in \mathcal{N} \qquad & \text{(43a)} \\
& \mathbf{1}^\top y_n + s_n \geq d_n & \forall n \in \mathcal{N} \qquad & \text{(43b)} \\
& y_n \leq N \left(x_n + w_0 \right) & \forall n \in \mathcal{N} \qquad & \text{(43c)} \\
& N \left(x_n + w_0 \right) \leq \beta N \left(x_n + w_0 \right) & \forall n \in \mathcal{N} \qquad & \text{(43d)} \\
& x_n = w_n + x_{a(n)} & \forall n \in \mathcal{N} \qquad & \text{(43e)} \\
& w_n, x_n \in \mathbb{Z}_+^b,\ y_n \in \mathbb{R}_+^b,\ s_n \in \mathbb{R}_+ & \forall n \in \mathcal{N}. \qquad & \text{(43f)}
\end{aligned}
$$

For a node $n \in \mathcal{N}$, the investment cost per generator is c_n, and the unit generation cost, including fuel and operational expenses, is g_n. We assume one generator of each technology has a fixed capacity, represented by a diagonal matrix N. We let $\bar{u}$ be the maximum number of generators for expansion, d_n be the electricity demand, and w_0 be the number of initially installed generators.

Constraint (43a) imposes an upper bound $\bar{u}$ on the total number of generators constructed. Constraint (43b) ensures that the total electricity produced and outsourced must satisfy the electricity demand. Constraint (43c) ensures that the total electricity production does not exceed the capacity of the installed generators. Additionally, we impose upper bounds in constraint (43d) on the installed capacity share of certain generation types, with a matrix β representing share upper bound parameters. Constraint (43e) models the transition between stages; i.e., it updates the number of total installed generators.

The detailed parametric setups are based on Jin et al. (2011). We consider a series of stage-wise independent uncertain $\{d_n\}_{n \in \mathcal{N}}$. We assume a deterministic initial demand d_1. At each subsequent stage t, the demand is independently generated by multiplying d_1 with a growth factor 1.05^t and a random fluctuation factor sampled from a uniform distribution $Unif(1.0, 1.2)$. The cumulative installed capacities of nuclear and renewable energy units are each restricted to be no more than 20% of the total installed generation capacity. In each iteration, we refine the partition according to the incumbent solution.

We compare the computational performance of SDDP-L with SDDiP and SDDP in Tables 1–3. We test three different Lagrangian cut types: LC, PLC, and SMC. There are R scenarios for each stage. We record the best lower bound (LB) and optimality gap (Gap) between statistical upper and lower bounds. The tables also include the number of iterations in the time limit (Iter), average time per iteration (Time/Iter.) with the standard deviation in the parenthesis, the time and the number of iterations to reach a 1% optimality gap (Time$_{1\%}$ and Iter$_{1\%}$).

SDDP-L Algorithm (Algorithm 2)

Cut	T	R	LB	Gap(%)	Iter.	Time/Iter.	Time$_{1\%}$	Iter$_{1\%}$
LC	10	5	16098.0	65.5	206	17.6 (8.7)	-	-
		10	15830.7	66.4	179	20.2 (9.9)	-	-
	15	5	15757.3	87.3	144	25.2 (10.9)	-	-
		10	15830.7	86.2	150	24.2 (11.4)	-	-
PLC	10	5	33397.3	0.8	629	5.7 (2.1)	13.4	3
		10	33643.6	0.3	426	8.5 (1.5)	18.4	4
	15	5	72723.5	0.6	166	22.0 (3.5)	181.7	12
		10	70833.8	1.0	124	29.3 (5.0)	876.6	33
SMC	10	5	33399.6	0.6	816	4.4 (1.4)	25.4	10
		10	33555.8	0.4	453	8.0 (1.5)	63.7	11
	15	5	72817.7	0.9	216	16.8 (2.5)	557.3	41
		10	71548.6	0.1	196	18.5 (4.9)	274.1	26

Table 1: Computational performance of SDDP-L algorithm for GEP.

Table 1 presents the performance of the SDDP-L algorithm with different types of cuts. With enhanced cuts PLC and SMC, the SDDP-L algorithm achieves an optimality gap below 1% for all instances, and we observe comparable computational performance between PLC and SMC. In contrast, LC does not close the optimality gap within the time limit.

Table 2 reports the result when we utilize the SDDiP algorithm in Zou et al. (2019) with general integer state variables equivalently represented by binary variables. The original state variable contains six integer elements, and the binary representation has 21 binary state variables. This dimensional growth imposes a heavier computational load on the SDDiP algorithm, which matches the nature of cutting-plane algorithms. As we compare Tables 1 and 2, the binary representation leads to a longer average time per iteration with larger run-time variability. Enhanced cuts exhibit a pronounced increase in per-iteration time, especially for PLC, because more iterations are required for the level algorithm to search the higher-dimensional dual space to generate a cut. In addition, a higher-dimensional state space also requires more cuts to approximate the value function to the same tolerance level.

Table 3 shows the result using the SDDP algorithm with Lagrangian cuts (Algorithm 1). Algorithm 1 demonstrates the best iteration efficiency as we do not need to include additional binary variables in the state space as in SDDP-L and SDDiP. The gap result shows that the convoluted

SDDiP Algorithm with State Variables Binary Representation

Cut	T	R	LB	Gap(%)	Iter.	Time/Iter.	Time$_{1\%}$	Iter$_{1\%}$
LC	10	5	27476.0	19.5	293	12.4 (7.1)	-	-
		10	26882.2	71.4	261	14.0 (6.6)	-	-
	15	5	26381.5	82.3	240	15.2 (7.0)	-	-
		10	25795.0	85.2	215	16.9 (7.8)	-	-
PLC	10	5	33393.9	0.9	70	52.7 (12.1)	50.4	2
		10	33639.5	0.0	37	101.2 (15.8)	431.1	6
	15	5	72834.1	0.4	29	133.0 (45.1)	958.0	12
		10	71561.2	0.7	19	204.1 (73.4)	1784.9	11
SMC	10	5	33089.0	2.7	242	15.0 (5.3)	-	-
		10	33273.4	2.0	167	21.8 (5.6)	-	-
	15	5	72663.8	1.0	126	29.0 (9.2)	1002.3	50
		10	71128.1	1.0	88	41.9 (9.3)	2676.8	68

Table 2: Computational performance of SDDiP algorithm for GEP.

convex envelope approximation is close to the original value function for GEP. This result echoes with Dowson and Kapelevich (2021), demonstrating that the two are often close in many practical problems. However, the SDDP-L algorithm is more reliable in guaranteeing a high approximation accuracy for a general problem setting. Additionally, for this low-dimensional problem without many added binary variables, LC also achieves convergence, but we observe that enhanced Lagrangian cuts achieve better lower bounds within the time limit.

SDDP with Lagrangian Cuts (Algorithm 1)

Cut	T	R	LB	Gap(%)	Iter.	Time/Iter.
LC	10	5	32866.6	2.4	1046	3.4 (1.2)
		10	33140.3	2.5	1102	3.3 (0.5)
	15	5	71678.0	2.0	694	5.2 (1.1)
		10	70398.5	2.1	681	5.3 (0.9)
PLC	10	5	32973.7	2.6	756	4.8 (0.6)
		10	33232.0	2.1	447	8.1 (1.3)
	15	5	71956.5	2.3	340	10.6 (1.5)
		10	70751.8	2.5	313	11.6 (3.8)
SMC	10	5	32899.0	1.4	1450	2.5 (0.5)
		10	33248.5	1.2	1417	2.5 (0.5)
	15	5	71929.6	2.1	654	5.5 (2.1)
		10	70776.0	2.4	327	11.0 (6.3)

Table 3: Computational performance of SDDP with Lagrangian cuts for GEP.

5.2 Multi-period Stochastic Unit Commitment

We consider a multi-period stochastic unit commitment (MSUC) problem, where we decide generators' on/off status, generation levels, and power flows. We formulate our MSUC problem based on similar models in Bienstock et al. (2024) and Zou et al. (2018). The electricity network consists of a set of buses $i \in \mathcal{B}$, loads $d \in \mathcal{D}$, and generators $g \in \mathcal{G}$. We use $\mathcal{D}_i$ and $\mathcal{G}_i$ to represent the subset of loads and generators connected to bus i. At a scenario tree node $n \in \mathcal{N}$, the state variables include binary indicators for generator commitment y_{gn}, startup action v_{gn}, and shutdown action w_{gn}, and continuous power generation levels P^G_{gn} for each $g \in \mathcal{G}$. We represent the proportion of load shed by local decisions, $(1 - x_{dn})$. The total cost includes a piecewise linear function of power output costs, denoted by h_{gn} with each piece $o \in \mathcal{O}$ a linear function parametrized by intercept c^0_{og} and slope c^1_{og}, startup and shutdown costs c^S_g and c^D_g, and load-shedding costs with a penalty rate of w_d. Constraint (44c) models DC power flow equations. Constraints (44d) and (44e) specify bounds for power flow on transmission lines and generation levels. Constraint (44f) represents the power flow balance. Constraint (44g) links the on/off status with startup and shutdown decisions. Constraints (44h) and (44i) enforce minimum up and down time constraints for the generators, where the set $\mathcal{N}(n,t)$ contains the path of nodes from the t-th ancestor of n to n. Finally, constraints (44j) and (44k) enforce the ramping limits with ramping bound parameters M_g.

$$\min \quad \sum_{n\in\mathcal{N}} p_n \left[\sum_{g\in\mathcal{G}} \left(h_{gn} + c^S_g v_{gn} + c^D_g w_{gn} \right) + \sum_{d\in\mathcal{D}} w_d (1 - x_{dn}) \right] \tag{44a}$$

$$\text{s.t.} \quad h_{gn} \geq c^1_{og} P^G_{gn} + c^0_{og} y_{gn} \qquad \forall o \in \mathcal{O}, g \in \mathcal{G}, n \in \mathcal{N} \tag{44b}$$

$$P^L_{ijt} = B_{ij} \left(\theta_{it} - \theta_{jt} \right) \qquad \forall (i,j) \in \mathcal{L}, n \in \mathcal{N} \tag{44c}$$

$$-W_{ij} \leq P^L_{ijn} \leq W_{ij} \qquad \forall (i,j) \in \mathcal{L}, n \in \mathcal{N} \tag{44d}$$

$$\underline{P}^G_g y_{gn} \leq P^G_{gn} \leq \overline{P}^G_g y_{gn} \qquad \forall g \in \mathcal{G}, n \in \mathcal{N} \tag{44e}$$

$$\sum_{g\in\mathcal{G}_i} P^G_{gn} + \sum_{(i,j)\in\mathcal{L}} P^L_{ijt} = \sum_{d\in\mathcal{D}_i} D_{dn} x_{dn} \qquad \forall i \in \mathcal{B}, n \in \mathcal{N} \tag{44f}$$

$$v_{gn} - w_{gn} = y_{gn} - y_{g,a(n)} \qquad \forall g \in \mathcal{G}, n \in \mathcal{N} \tag{44g}$$

$$\sum_{m\in\mathcal{N}(n,UT-1)} v_{gm} \leq y_{gn} \qquad \forall g \in \mathcal{G}, n \in \mathcal{N} \tag{44h}$$

$$\sum_{m\in\mathcal{N}(n,UD-1)} w_{gm} \leq 1 - y_{gn} \qquad \forall g \in \mathcal{G}, n \in \mathcal{N} \tag{44i}$$

$$P^G_{gn} - P^G_{g,a(n)} \leq M_g (y_{g,a(n)} + v_{gn}) \qquad \forall g \in \mathcal{G}, n \in \mathcal{N} \tag{44j}$$

$$P^G_{g,a(n)} - P^G_{gn} \leq M_g (y_{gn} + w_{gn}) \qquad \forall g \in \mathcal{G}, n \in \mathcal{N} \tag{44k}$$

$$v_{gn}, y_{gn}, w_{gn} \in \{0,1\} \qquad \forall g \in \mathcal{G}, n \in \mathcal{N} \tag{44l}$$

$$x_{dn} \in [0,1] \qquad \forall d \in \mathcal{D}, n \in \mathcal{N} \tag{44m}$$

We test algorithms on a `IEEE 30-bus` test case (Shahidehpour and Wang 2003), across 6, 8, or

12 periods, and each period corresponds to a stage. The uncertain demand series D_{dn} is modeled as follows: We create a series of nominal demands with values for the first six stages equal to 90% of the static demand and the values for the seventh stage and beyond equal to 130% of the static demand, while each stage bears an additional ±10% fluctuation distributed uniformly. We assume stage-wise independence, each stage having 5 or 10 realizations.

SDDP-L Algorithm with Partitions Refined according to Incumbent Solutions

Cut	T	R	LB	Gap(%)	Iter.	Time/Iter.	$\text{Time}_{1\%}$	$\text{Iter}_{1\%}$
LC	6	5	1629.5	61.3	52	73.4 (54.1)	-	-
		10	1296.2	66.4	51	74.5 (55.9)	-	-
	8	5	1662.8	74.1	44	88.3 (68.4)	-	-
		10	1662.8	74.2	43	88.8 (70.1)	-	-
	12	5	1662.8	84.5	33	112.6 (94.0)	-	-
		10	1999.7	83.2	45	85.4 (44.9)	-	-
PLC	6	5	4203.6	0.1	56	66.3 (55.0)	24.3	4
		10	3849.0	0.2	48	79.2 (57.0)	33.1	4
	8	5	6405.7	0.3	42	88.0 (70.9)	85.7	8
		10	6386.8	0.6	39	96.1 (62.5)	290.1	10
	12	5	10563.1	1.0	32	121.1 (94.3)	1709.9	24
		10	11574.1	2.5	30	133.4 (89.1)	-	-
SMC	6	5	4206.3	0.1	57	66.0 (41.4)	16.6	4
		10	3853.4	0.0	53	71.5 (49.4)	17.3	4
	8	5	6413.1	0.1	44	87.9 (60.8)	46.4	6
		10	6399.3	0.0	43	88.8 (63.6)	71.5	7
	12	5	10627.8	0.1	34	114.9 (84.8)	161.7	9
		10	11749.2	0.5	36	104.8 (80.4)	308.3	12

Table 4: Computational performance of SDDP-L algorithm with partitions refined according to incumbent solutions for MSUC.

Table 4 presents the performance of SDDP-L applied to MSUC, with the partitions refined according to the incumbent solution. LC still fails to provide meaningful lower-bound improvements within the time limit, while PLC and SMC quickly improve the lower bounds, closing optimality gaps to 1% tolerance in most cases, except for PLC with $T = 12$ and $R = 10$. SMC requires fewer iterations and less time than PLC to achieve a gap below 1%.

We test SDDP-L applied to MSUC with the partitions refined according to bisection and show the computational results in Table 5, so that we can further investigate the impact of partition refinement strategies. The results suggest that both partitioning schemes with enhanced Lagrangian cuts can quickly close the optimality gap. However, refining partitions according to bisection significantly improves computational performance for LC, especially for smaller cases. We believe the

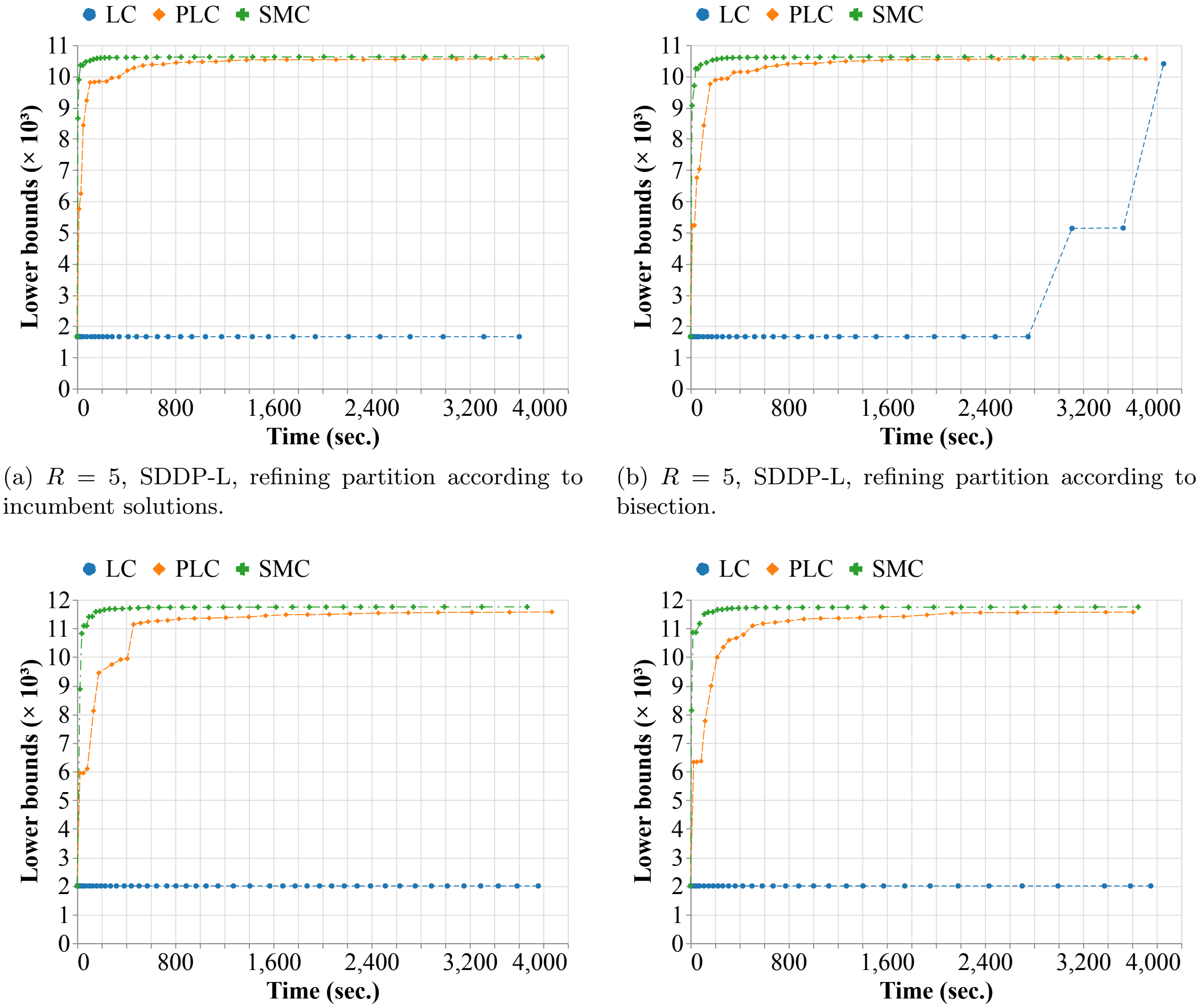
SDDP-L Algorithm with Partitions Refined according to Bisection

Cut	T	R	LB	Gap(%)	Iter.	Time/Iter.	Time$_{1\%}$	Iter$_{1\%}$
LC	6	5	4137.5	1.7	50	77.2 (60.6)	-	-
		10	3788.2	1.7	48	77.5 (60.5)	-	-
	8	5	6342.2	0.9	42	91.8 (80.9)	2082.2	35
		10	4004.9	37.4	41	94.5 (81.5)	-	-
	12	5	10402.3	10.8	32	124.4 (112.7)	-	-
		10	1999.7	83.0	32	121.0 (94.0)	-	-
PLC	6	5	4203.4	0.1	49	76.2 (55.4)	30.7	4
		10	3847.6	0.2	46	83.2 (55.8)	30.2	4
	8	5	6408.8	0.4	41	92.1 (76.3)	190.6	10
		10	6389.5	0.2	39	101.5 (78.0)	146.0	8
	12	5	10560.3	1.0	31	123.7 (97.8)	2796.1	28
		10	11568.1	1.9	28	133.6 (99.6)	-	-
SMC	6	5	4206.4	0.1	49	75.0 (64.8)	18.0	4
		10	3853.1	0.1	50	74.3 (59.2)	18.5	4
	8	5	6413.7	0.3	42	91.6 (75.1)	73.9	7
		10	6399.2	0.5	41	91.2 (78.7)	129.1	9
	12	5	10627.1	0.0	33	113.4 (92.1)	254.9	10
		10	11746.0	0.8	32	117.7 (96.2)	256.7	10

Table 5: Computational performance of SDDP-L algorithm with partitions refined according to bisection for MSUC.

incumbent solutions often lie in the relative interior of an element of the partition when we use the bisection refinement. Therefore, LC will encounter multiple optimal dual multipliers less frequently, and the cuts generated will provide a good approximation for the value function elsewhere. On the other hand, it is easier to identify an optimal dual multiplier to the Lagrangian dual problem when we have multiple optimal solutions. This may explain why the average iteration time is slightly longer across all test cases when we refine partitions according to bisection.

Fig. 7 illustrates the progression of the lower bound when $T = 12$. We observe that the lower bound. PLC and SMC substantially improve the lower bound within a small number of iterations, and SMC consistently performs better than PLC in both time and the number of iterations. In contrast, LC exhibits slower progress in the early stages. In the case with $R = 5$, LC undergoes many iterations without lower-bound improvement, followed by a sharp lower-bound increase only in the final iterations. Although LC is tight at the incumbent solution that is also an extreme point of $\mathcal{Z}_{a(n)}$, it is often too steep to approximate the value function elsewhere. Thus, the algorithm needs to explore many points on the boundary before exploring the points in the neighborhood of the optimal solution. We also observe a similar trend for $T = 6$ and $T = 8$. However, for the most challenging instance with $T = 12$ and $R = 10$, it takes too many iterations to enumerate boundary points and close the gap within the given time limit, resulting in a flat lower-bound trajectory.



(a) $R = 5$, SDDP-L, refining partition according to incumbent solutions.

(b) $R = 5$, SDDP-L, refining partition according to bisection.

(c) $R = 10$, SDDP-L, refining partition according to incumbent solutions.

(d) $R = 10$, SDDP-L, refining partition according to bisection.

Figure 7: Progression of lower bounds for the MSUC instance with $T = 12$. Each row of panels compares the effectiveness of two partitioning strategies in SDDP-L.

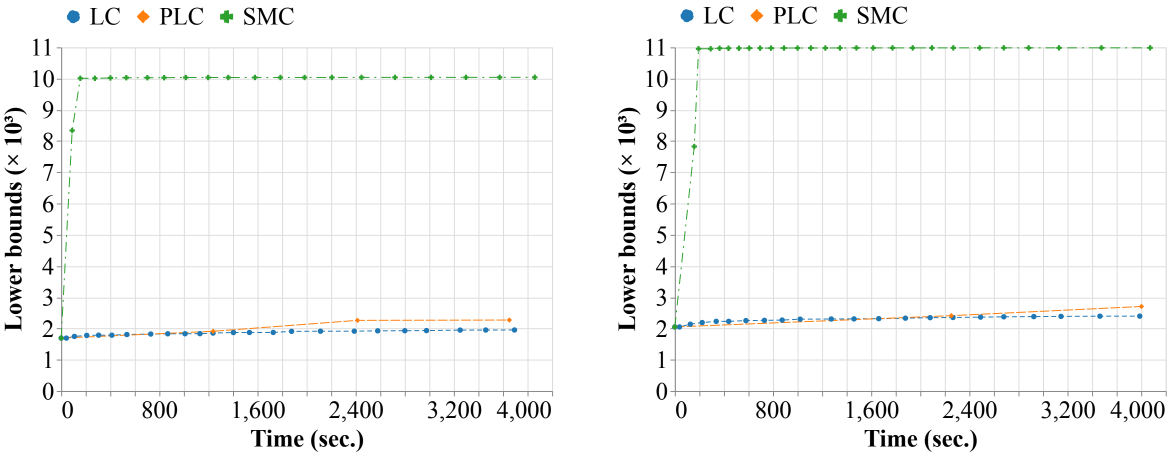
Table 6 reports the computational performance of the SDDiP algorithm with mixed-integer state variables represented by binary variables for the MSUC problem. We also present the progression of the lower bounds for instances with $T = 12$ in Fig. 8. We approximate each continuous variable with a precision of 2^{-8}. Binary representation increases the dimension of the state variable space from 24 to 63, leading to a longer average iteration time and fewer iterations completed within the time limit. Only SMC can obtain optimality gaps below 1% for $T = 6$ cases. For larger cases ($T = 8$ and 12), neither PLC nor SMC achieves a decent gap result within the time limit due to the substantially increased complexity. Like GEP, LC fails to provide meaningful lower bounds in all test cases because of the steepness issue, and SMC achieves the best gap result because of its computational efficiency. However, we observe a different trend for PLC as it exhibits the longest per-iteration time among the three, resulting in few total cuts generated and poor lower bounds.

SDDiP Algorithm with State Variables Binary Representation of Precision 2^{-8}								
Cut	T	R	LB	Gap(%)	Iter.	Time/Iter.	$\text{Time}_{1\%}$	$\text{Iter}_{1\%}$
LC	6	5	2037.9	54.6	30	129.5 (67.3)	-	-
		10	1554.4	61.7	32	123.3 (77.0)	-	-
	8	5	2003.7	69.5	29	138.0 (72.2)	-	-
		10	2003.4	69.8	25	159.0 (90.1)	-	-
	12	5	1952.8	82.5	24	160.6 (63.2)	-	-
		10	2395.9	80.4	22	180.4 (72.9)	-	-
PLC	6	5	2832.3	35.0	7	704.5 (114.2)	-	-
		10	2391.1	41.5	5	983.4 (125.7)	-	-
	8	5	2483.1	64.4	6	866.2 (127.0)	-	-
		10	2275.5	65.6	4	1569.0 (322.4)	-	-
	12	5	2271.1	79.4	4	1217.3 (35.9)	-	-
		10	2705.5	78.3	3	1900.9 (497.8)	-	-
SMC	6	5	4266.1	0.1	60	62.5 (29.0)	120.4	5
		10	3887.9	0.2	30	127.2 (48.9)	349.6	5
	8	5	5354.9	18.2	23	171.2 (55.7)	-	-
		10	5437.8	16.6	23	173.4 (76.3)	-	-
	12	5	10041.1	8.4	21	193.0 (68.7)	-	-
		10	10985.2	8.5	27	149.0 (82.6)	-	-

Table 6: Computational performance of SDDiP algorithm for MSUC.

We test the performance of SDDP with Lagrangian cuts (Algorithm 1) for the MSUC problem and present the results in Table 7 and Fig. 9. The results show that PLC and SMC quickly improve lower bounds and reach a plateau. For larger instances ($T = 8$ and 12), the optimality gaps are approximately 2%, indicating a gap between the convex envelope and the original value function. LC performs poorly as it does not improve the lower bound after many iterations. This is consistent with observations from SDDP-L and SDDiP tests.

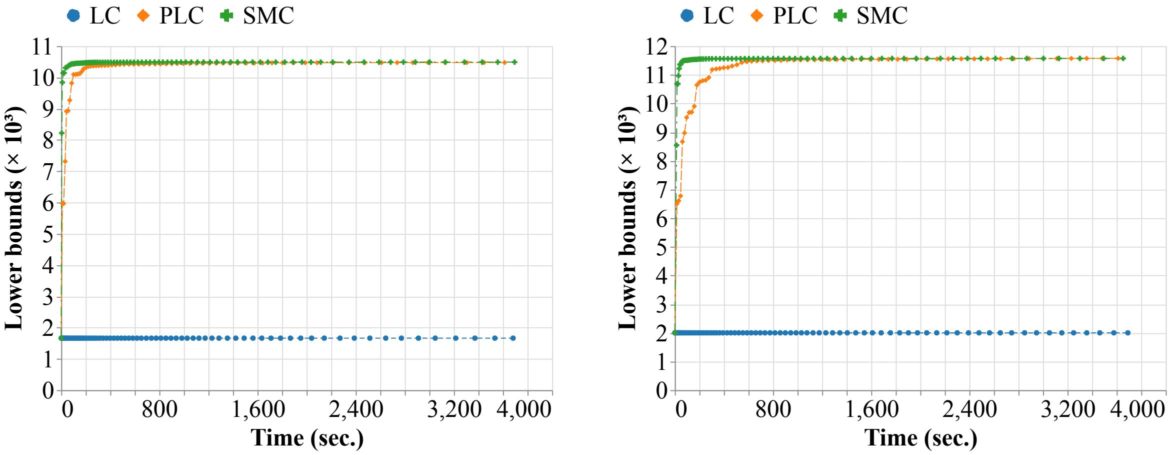
Fig. 10 lists statistics of the average iteration time and the average number of iterations required to solve CGCPs for generating Lagrangian cuts for the MSUC instance with $T = 12$ and $R = 10$. Fig. 10(a) shows the average number of iterations in the level method to generate a Lagrangian cut, while Fig. 10(b) illustrates the average time per iteration to generate a Lagrangian cut. PLC requires the longest average iteration time and the highest number of iterations across all cut types, while SMC achieves good performance without requiring as much computational effort.



(a) $R = 5$, SDDiP.

(b) $R = 10$, SDDiP.

Figure 8: Progression of lower bounds for the MSUC instance with $T = 12$. The results compare the performance of SDDiP with $R = 5$ and $R = 10$.



(a) $R = 5$, SDDP with Lagrangian cuts.

(b) $R = 10$, SDDP with Lagrangian cuts.

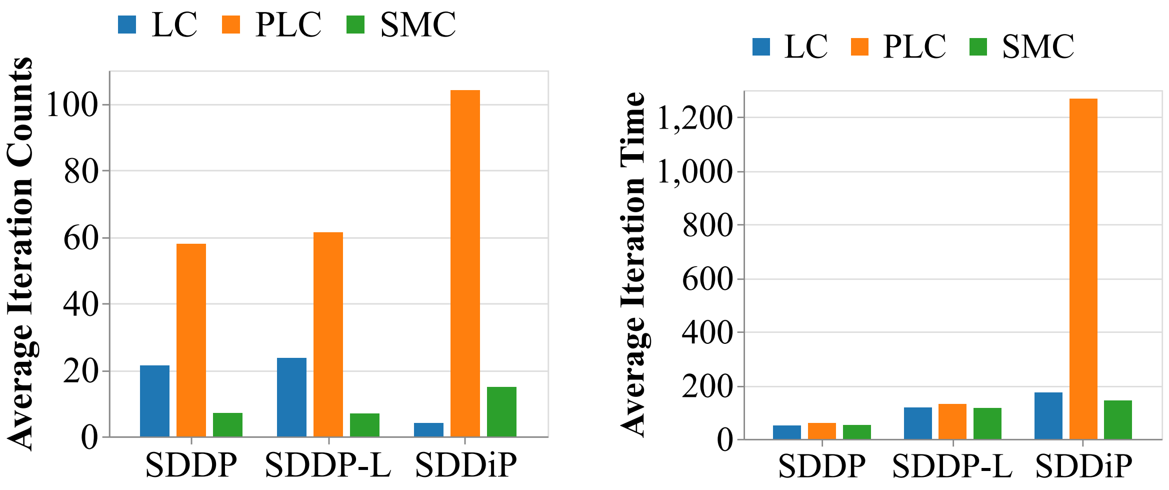
Figure 9: Progression of lower bounds for the MSUC instance with $T = 12$. The results compare the performance of SDDP with Lagrangian cuts with $R = 5$ and $R = 10$.

SDDP with Lagrangian Cuts (Algorithm 1)						
Cut	T	R	LB	Gap(%)	Iter.	Time/Iter.
LC	6	5	1629.5	61.3	116	31.4 (24.3)
		10	1296.2	66.4	108	34.3 (21.5)
	8	5	1662.8	74.1	94	39.9 (30.0)
		10	1662.8	74.1	97	37.5 (26.0)
	12	5	1662.8	84.4	72	51.9 (44.6)
		10	1999.7	83.1	76	49.2 (32.1)
PLC	6	5	4169.3	0.9	108	34.4 (25.2)
		10	3819.0	0.9	89	41.1 (22.7)
	8	5	6327.1	1.7	89	41.6 (28.9)
		10	6277.8	2.1	82	45.6 (28.2)
	12	5	10476.0	2.0	65	56.5 (44.0)
		10	11574.4	1.5	62	59.2 (34.6)
SMC	6	5	4172.7	0.9	131	27.8 (21.2)
		10	3822.6	0.8	104	35.8 (29.8)
	8	5	6330.3	1.6	104	35.0 (25.3)
		10	6265.6	2.0	94	39.2 (27.9)
	12	5	10490.3	1.7	80	46.8 (39.0)
		10	11571.9	1.9	70	52.9 (45.3)

Table 7: Computational performance of SDDP algorithm with Lagrangian cuts for MSUC.

6 Conclusion

In this paper, we demonstrate the geometric structure of Lagrangian cuts and show how they characterize the convex envelope of the value function over a restricted region. We integrate Lagrangian cuts into the SDDP framework, propose Algorithm 1, and provide theoretical guarantees for the convergence of Algorithm 1 to a convoluted convex envelope approximation of the value function. To recover the global optimum with exactness for MS-SMIP, we progressively refine the partition of state variables and represent it with additional binary state variables (lifting) in the SDDP-L algorithm. We prove the tightness of lifted Lagrangian cuts and establish the asymptotic convergence of the SDDP-L algorithm. The lifting process may lead to multiple candidate Lagrangian cuts, and there is no consistent rule of cut selection to guarantee computational efficiency. We propose two enhancements for Lagrangian cuts, PLC and SMC, and we show their validity, tightness, and Pareto-optimality properties. These properties help significantly accelerate convergence and improve computational efficiency. Our numerical experiments validate the effectiveness and efficiency of the proposed methods. In particular, the combination of progressive lifting and enhanced Lagrangian cuts in the SDDP-L algorithm consistently improves lower bounds and ensures quick



(a) Average number of iterations in the level method required to generate a Lagrangian cut.

(b) Average time per iteration of the level method to generate a Lagrangian cut.

Figure 10: Statistics on the cut generation process for the MSUC instance with $T = 12$ and $R = 10$.

convergence to the optimal solution for MS-SMIP with general mixed-integer state variables.

Our methods provide a fundamental algorithmic framework to solve general MS-SMIP to exactness. In the future, we plan to explore the use of restricted Lagrangian multipliers for cut generation, as suggested in Chen and Luedtke (2022), and investigate the integration of regularized Lagrangian cuts within the SDDP-L framework to enhance algorithmic stability and performance further. Another future direction involves utilizing the power of machine learning to quickly generate Lagrangian cuts of good quality and establish partition rules to reduce the number of iterations. All the future directions should serve the ultimate goal of building easily deployable tools to fully solve general MS-SMIP to exactness.

References

S. Ahmed, F. G. Cabral, and B. Freitas Paulo da Costa. Stochastic Lipschitz dynamic programming. *Mathematical Programming*, 191(2):755–793, 2022.

G. Angulo, S. Ahmed, and S. S. Dey. Improving the integer L-shaped method. *INFORMS Journal on Computing*, 28(3):483–499, 2016.

A. Bansal and S. Küçükyavuz. A computational study of cutting-plane methods for multi-stage stochastic integer programs. *arXiv preprint arXiv:2405.02533*, 2024. URL https://arxiv.org/abs/2405.02533.

J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

D. Bienstock, Y. Dvorkin, C. Guo, R. Mieth, and J. Wang. Risk-aware security-constrained unit commitment: Taming the curse of real-time volatility and consumer exposure. *IEEE Transactions on Energy Markets, Policy and Regulation*, 2024.

J. R. Birge. Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research*, 33(5):989–1007, 1985.

R. Brandenberg and P. Stursberg. Refined cut selection for Benders decomposition: applied to network capacity expansion problems. *Mathematical Methods of Operations Research*, 94: 1–30, 12 2021.

R. Chen and J. Luedtke. On generating Lagrangian cuts for two-stage stochastic integer programs. *INFORMS Journal on Computing*, 34(4), 2022.

G. Codato and M. Fischetti. Combinatorial Benders' cuts for mixed-integer linear programming. *Operations Research*, 54(4):756–766, 2006.

V. L. De Matos, D. P. Morton, and E. C. Finardi. Assessing policy quality in a multistage stochastic program for long-term hydrothermal scheduling. *Annals of Operations Research*, 253:713–731, 2017.

H. Deng and W. Xie. On the ReLU Lagrangian cuts for stochastic mixed integer programming. *arXiv preprint arXiv:2411.01229*, 2024. URL https://arxiv.org/abs/2411.01229.

O. Dowson. The policy graph decomposition of multistage stochastic programming problems. *Networks*, 76(1):3–23, 2020.

O. Dowson and L. Kapelevich. SDDP.jl: a Julia package for stochastic dual dynamic programming. *INFORMS Journal on Computing*, 33(1):27–33, 2021.

I. Dunning, J. Huchette, and M. Lubin. JuMP: A modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320, 2017.

Y. Fan and C. Liu. Solving stochastic transportation network protection problems using the progressive hedging-based method. *Networks and Spatial Economics*, 10(2):193–208, 2010.

M. Fischetti, D. Salvagnin, and A. Zanette. A note on the selection of Benders' cuts. *Mathematical Programming*, 124:175–182, 07 2010.

M. L. Fisher. The Lagrangian relaxation method for solving integer programming problems. *Management Science*, 50(12_supplement):1861–1871, 2004.

N. Flatabø, A. Haugstad, B. Mo, and O. B. Fosso. Short-term and medium-term generation scheduling in the Norwegian hydro system under a competitive power market structure. In *International Conference on Electrical Power System Operation and Management*, 1998.

C. Füllner and S. Rebennack. Non-convex nested Benders decomposition. *Mathematical Programming*, 196(1):987–1024, 2022.

C. Füllner, X. A. Sun, and S. Rebennack. A new framework to generate Lagrangian cuts in multistage stochastic mixed-integer programming. *Available at Optimization Online*, 2024a. URL https://optimization-online.org/?p=27312.

C. Füllner, X. A. Sun, and S. Rebennack. On Lipschitz regularization and Lagrangian cuts in multistage stochastic mixed-integer linear programming. *Available at Optimization Online*, 2024b. URL https://optimization-online.org/?p=27295.

D. Gade, S. Küçükyavuz, and S. Sen. Decomposition algorithms with parametric Gomory cuts for two-stage stochastic integer programs. *Mathematical Programming*, 144(1-2):39–64, 2014.

A. Geoffrion. Lagrangian relaxation and its uses in integer programming. *Mathematical Programming*, 2, 01 1974.

P. Girardeau, V. Leclere, and A. B. Philpott. On the convergence of decomposition methods for multistage stochastic convex programs. *Mathematics of Operations Research*, 40(1):130–145, 2015.

S. Gul, B. T. Denton, and J. W. Fowler. A progressive hedging approach for surgery planning under uncertainty. *INFORMS Journal on Computing*, 27(4):755–772, 2015.

C. Guo, M. Bodur, D. M. Aleman, and D. R. Urbach. Logic-based Benders decomposition and binary decision diagram based approaches for stochastic distributed operating room scheduling. *INFORMS Journal on Computing*, 33(4):1551–1569, 2021.

Gurobi Optimization, Inc. *Gurobi optimizer reference manual*, 2014. URL https://www.gurobi.com.

J. N. Hooker and G. Ottosson. Logic-based Benders decomposition. *Mathematical Programming*, 96(1):33–60, 2003.

T. Ichiishi. 1 - introduction to convex analysis. In T. Ichiishi, editor, *Game Theory for Economic Analysis*, Economic Theory, Econometrics, and Mathematical Economics, pages 7–29. Academic Press, San Diego, 1983.

S. Jin, S. M. Ryan, J.-P. Watson, and D. L. Woodruff. Modeling and solving a large-scale generation expansion planning problem under uncertainty. *Energy Systems*, 2:209–242, 2011.

G. Laporte and F. V. Louveaux. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, 13(3):133–142, 1993.

C. Lemaréchal, A. Nemirovskii, and Y. Nesterov. New variants of bundle methods. *Mathematical Programming*, 69(1):111–147, 1995.

O. Listes and R. Dekker. A scenario aggregation–based approach for determining a robust airline fleet composition for dynamic capacity allocation. *Transportation Science*, 39(3):367–382, 2005.

M. Locatelli and F. Schoen. *Global optimization: Theory, algorithms, and applications.* SIAM, 2013.

T. L. Magnanti and R. T. Wong. Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, 29(3):464–484, 1981.

N. Newham and A. Wood. Transmission investment planning using SDDP. In *Australasian Universities Power Engineering Conference*, pages 1–5, 2007.

L. Ntaimo. Fenchel decomposition for stochastic mixed-integer programming. *Journal of Global Optimization*, 55:141–163, 2013.

M. V. Pereira and L. M. Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52(1-3):359–375, 1991.

A. Philpott, F. Wahid, and J. Bonnans. MIDAS: A mixed integer dynamic approximation scheme. *Mathematical Programming*, pages 1–32, 2016.

A. B. Philpott and Z. Guan. On the convergence of stochastic dual dynamic programming and related methods. *Operations Research Letters*, 36(4):450–455, 2008.

R. Rahmaniani, S. Ahmed, T. G. Crainic, M. Gendreau, and W. Rei. The Benders dual decomposition method. *Operations Research*, 68(3):878–895, 2020.

S. Rajagopalan, M. R. Singh, and T. E. Morton. Capacity expansion and replacement in growing markets with uncertain technological breakthroughs. *Management Science*, 44(1):12–30, 1998.

R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

W. Romeijnders and N. van der Laan. Benders decomposition with scaled cuts for multistage stochastic mixed-integer programs. *Available at Optimization Online*, 2024. URL https://optimization-online.org/?p=26876.

R. Schultz. Continuity properties of expectation functions in stochastic integer programming. *Mathematics of Operations Research*, 18(3):578–589, 1993.

R. Schultz. Mixed-integer value functions in stochastic programming. In M. Jünger, G. Reinelt, and G. Rinaldi, editors, *Combinatorial Optimization — Eureka, You Shrink!: Papers Dedicated to Jack Edmonds 5th International Workshop Aussois, France, March 5–9, 2001 Revised Papers*, pages 171–184. Springer Berlin Heidelberg, 2003.

S. Sen and J. L. Higle. The C^3 theorem and a D^2 algorithm for large scale stochastic mixed-integer programming: set convexification. *Mathematical Programming*, 104:1–20, 2005.

S. Sen and H. Sherali. Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming. *Mathematical Programming*, 106:203–223, 2006.

M. Shahidehpour and Y. Wang. Appendix C: IEEE30 bus system data. In *Communication and Control in Electric Power Systems: Applications of Parallel and Distributed Processing*, pages 493–495. 2003.

S. Takriti, J. R. Birge, and E. Long. A stochastic model for the unit commitment problem. *IEEE Transactions on Power Systems*, 11(3):1497–1508, 1996.

M. Tawarmalani and N. Sahinidis. Convex extensions and envelopes of lower semi-continuous functions. *Mathematical Programming*, 93:247–263, 12 2002.

N. van der Laan and W. Romeijnders. A converging Benders' decomposition algorithm for two-stage mixed-integer recourse models. *Operations Research*, 72(5):2190–2214, 2024.

K. Wang and A. Jacquillat. A stochastic integer programming approach to air traffic scheduling and operations. *Operations Research*, 68(5):1375–1402, 2020.

H. Yang and D. P. Morton. Optimal crashing of an activity network with disruptions. *Mathematical Programming*, 194(1):1113–1162, 2022.

H. Yang and H. Nagarajan. Optimal power flow in distribution networks under N-1 disruptions: A multistage stochastic programming approach. *INFORMS Journal on Computing*, 34(2): 690–709, 2022.

H. Yang, H. Yang, N. Rhodes, L. Roald, and L. Ntaimo. Multistage stochastic program for mitigating power system risks under wildfire disruptions. *Electric Power Systems Research*, 234: 110773, 2024.

M. Zhang and S. Küçükyavuz. Finitely convergent decomposition algorithms for two-stage stochastic pure integer programs. *SIAM Journal on Optimization*, 24:1933–1951, 2014.

S. Zhang and X. A. Sun. Stochastic dual dynamic programming for multistage stochastic mixed-integer nonlinear optimization. *Mathematical Programming*, 196(1):935–985, 2022.

J. Zou, S. Ahmed, and X. A. Sun. Multistage stochastic unit commitment using stochastic dual dynamic integer programming. *IEEE Transactions on Power Systems*, 34(3):1814–1823, 2018.

J. Zou, S. Ahmed, and X. A. Sun. Stochastic dual dynamic integer programming. *Mathematical Programming*, 175(1-2):461–502, 2019.

Appendix

A Proof of Equality (b) in Theorem 1

Lemma 11. *Suppose we have the same setting as in Theorem 1. We let* $f_n(x_n, y_n, \Theta_n) := c_n^\top x_n + g_n^\top y_n + \sum_{m\in\mathcal{C}(n)} q_{nm}\theta_m$ *where* Θ_n *denotes the collection of value function variables* $\{\theta_m\}_{m\in\mathcal{C}(n)}$*, and*

$$\mathcal{F}_n := \{(z_n, x_n, y_n, \Theta_n) \mid \text{ constraints } (8b) - (8f)\}.$$

Then we have

$$\min\left\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \{(z_n, f_n(x_n, y_n, \Theta_n)) \mid (z_n, x_n, y_n, \Theta_n) \in \operatorname{conv}(\mathcal{F}_n)\}\right\} \tag{45}$$

$$= \min\left\{\eta \mid (\hat{x}_{a(n)}, \eta) \in \operatorname{conv}\left(\{(z_n, f_n(x_n, y_n, \Theta_n)) \mid (z_n, x_n, y_n, \Theta_n) \in \mathcal{F}_n\}\right)\right\}, \tag{46}$$

where the incumbent solution $\hat{x}_{a(n)} \in \mathcal{Z}_{a(n)}$*.*

Proof. We can rewrite the model in (45) as:

$$\begin{aligned}
\nu^1 = \min_{\eta, z_n, x_n, y_n, \Theta_n} \quad & \eta && \text{(47a)}\\
\text{s.t.} \quad & (z_n, x_n, y_n, \Theta_n) \in \operatorname{conv}(\mathcal{F}_n) && \text{(47b)}\\
& z_n = \hat{x}_{a(n)} && \text{(47c)}\\
& \eta = f_n(x_n, y_n, \Theta_n). && \text{(47d)}
\end{aligned}$$

We can enumerate all combinations of discrete variables within $\mathcal{F}_n$. For each element within this combination set fixed, we can enumerate all the extreme points of the continuous part and collect a finite set of points within $\mathcal{F}_n$, because the set $\mathcal{F}_n$ is compact and there must exist a set of extreme points. We denote the indices of this finite set as $\tau \in \mathfrak{T}$ and each point can be represented as $(z_n^\tau, x_n^\tau, y_n^\tau, \Theta_n^\tau)$. Then the model in (46) can be rewritten as the following model:

$$\begin{aligned}
\nu^2 = \min_{\alpha, \eta} \quad & \eta && \text{(48a)}\\
\text{s.t.} \quad & (z_n^\tau, x_n^\tau, y_n^\tau, \Theta_n^\tau) \in \mathcal{F}_n && \text{(48b)}\\
& \hat{x}_{a(n)} = \sum_{\tau\in\mathfrak{T}} \alpha_\tau z_n^\tau && \text{(48c)}\\
& \eta = \sum_{\tau\in\mathfrak{T}} \alpha_\tau f_n(x_n^\tau, y_n^\tau, \Theta_n^\tau) && \text{(48d)}\\
& \sum_{\tau\in\mathfrak{T}} \alpha_\tau = 1 && \text{(48e)}\\
& \alpha_\tau \geq 0 \qquad\qquad \forall \tau \in \mathfrak{T}. && \text{(48f)}
\end{aligned}$$

We can show that $\nu^1 \geq \nu^2$ by transforming the optimal solution to model (47) as a feasible solution to model (48). Suppose the optimal solution to model (47) is $(z_n^*, x_n^*, y_n^*, \Theta_n^*)$ and $z_n^* =$

$\hat{x}_{a(n)}, \eta^* = f_n(x_n^*, y_n^*, \Theta_n^*)$. Since $(z_n^*, x_n^*, y_n^*, \Theta_n^*) \in \operatorname{conv}(\mathcal{F}_n)$, we can find convex combination coefficients $\alpha_\tau \geq 0, \tau \in \mathfrak{T}$ where $z_n^* = \sum_{\tau \in \mathfrak{T}} \alpha_\tau z_n^\tau = \hat{x}_{a(n)}$, $x_n^* = \sum_{\tau \in \mathfrak{T}} \alpha_\tau x_n^\tau$, $y_n^* = \sum_{\tau \in \mathfrak{T}} \alpha_\tau y_n^\tau$, $\Theta_n^* = \sum_{\tau \in \mathfrak{T}} \alpha_\tau \Theta_n^\tau$, $\sum_{\tau \in \mathfrak{T}} \alpha_\tau = 1$. Since the function f is a linear function in x_n, y_n and Θ_n, we have $f(x_n^*, y_n^*, \Theta_n^*) = f(\sum_{\tau \in \mathfrak{T}} \alpha_\tau x_n^\tau, \sum_{\tau \in \mathfrak{T}} \alpha_\tau y_n^\tau, \sum_{\tau \in \mathfrak{T}} \alpha_\tau \Theta_n^\tau) = \sum_{\tau \in \mathfrak{T}} \alpha_\tau f_n(x_n^\tau, y_n^\tau, \Theta_n^\tau)$. We can directly see this solution $(\{\alpha_\tau\}_{\tau \in \mathfrak{T}}, \eta)$ is a feasible solution to model (48), and thus $\nu^1 \geq \nu^2$.

On the other hand, we can reach the conclusion that $\nu^2 \geq \nu^1$ by transforming the optimal solution $(\{\alpha_\tau^*\}_{\tau \in \mathfrak{T}}, \eta^*)$ to model (47) as a feasible solution to model (48). We can directly construct a solution $z_n = \sum_{\tau \in \mathfrak{T}} \alpha_\tau^* z_n^\tau = \hat{x}_{a(n)}$, $x_n = \sum_{\tau \in \mathfrak{T}} \alpha_\tau^* x_n^\tau$, $y_n = \sum_{\tau \in \mathfrak{T}} \alpha_\tau^* y_n^\tau$, $\Theta_n = \sum_{\tau \in \mathfrak{T}} \alpha_\tau^* \Theta_n^\tau$. By the linearity of function f, this solution satisfies all constraints in model (47) so it is a feasible solution to model (47). Therefore, we show that $\nu^2 \geq \nu^1$.

Combining both directions, we prove $\nu^1 = \nu^2$, which is equivalent to the statement that (45) equals to (46). □

B Proof of Theorem 2

Before proving Theorem 2, we show the validity result of Lagrangian cuts for $\operatorname{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n(\cdot)\right)$.

Lemma 12. *For every node $n \in \mathcal{N}$, given Φ_n and $\mathcal{Z}_{a(n)}$. Then the lower approximation comprising any collection of Lagrangian cuts $\phi_n(x_{a(n)}) \leq \operatorname{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)$ at any $x_{a(n)} \in \mathcal{Z}_{a(n)}$.*

Proof. Each Lagrangian cut is a linear function of $x_{a(n)}$, and thus $\phi_n(x_{a(n)})$ is convex in $x_{a(n)}$. By the definition of convex envelope and the validity of Lagrangian cuts for $\underline{Q}_n(\cdot;\Phi_n)$, for any $n \in \mathcal{N}$ we have

$$\operatorname{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n)\right)(x_{a(n)}) \geq \phi_n(x_{a(n)}) \qquad \forall x_{a(n)} \in \mathcal{Z}_{a(n)}.$$

We can use backward induction to prove for any node $n \in \mathcal{N}$, at any $x_{a(n)} \in \mathcal{Z}_{a(n)}$,

$$\operatorname{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n)\right)(x_{a(n)}) \leq \operatorname{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}) \tag{49}$$

For all $n \in \mathcal{N}_T$, we have $\underline{Q}_n(x_{a(n)};\Phi_n) = \tilde{Q}_n(x_{a(n)})$ at any given $x_{a(n)} \in \mathcal{Z}_{a(n)}$ since $\mathcal{C}(n) = \emptyset$, and the inequality (49) automatically holds. Suppose for a node $n \in \mathcal{N} \backslash \mathcal{N}_T$ with the inequality (49) true for all $m \in \mathcal{C}(n)$. The feasible regions of (x_n, y_n) for $\tilde{Q}_n$ and $\underline{Q}_n(\cdot;\Phi_n)$ are the same, and for every feasible solution (x_n, y_n), we have $\operatorname{co}_{\mathcal{Z}_n}\left(\tilde{Q}_m\right)(x_n) \geq \phi_m(x_n)$. Therefore, $\tilde{Q}_n(x_{a(n)}) \geq \underline{Q}_n(x_{a(n);\Phi_n})$ for any $x_{a(n)} \in \mathcal{Z}_{a(n)}$, and the same inequality should hold for their convex envelopes as well. □

Lemma 13 (Lemma A.1. in Girardeau et al. (2015))**.** *Suppose f is convex and X is compact, and suppose for any integer κ, the sequence of α-Lipschitz convex functions $f^k, k \in \mathbb{N}$ satisfies*

$$f^{k-\kappa}(x) \leq f^k(x) \leq f(x), \text{ for all } x \in X.$$

Then, for any infinite sequence $x^k \in X$

$$\lim_{k\to\infty} (f(x^k) - f^k(x^k)) = 0 \iff \lim_{k\to\infty} (f(x^k) - f^{k-\kappa}(x^k)) = 0$$

With the validity result and technical lemma above, we now prove Theorem 2.

Proof of Theorem 2

Proof. We prove part (i) by backward induction. For all $n \in \mathcal{N}_T$, since the feasible region X_n is compact, within the sequence K, there must exist an infinite subsequence J such that $\{(x_n^k, y_n^k)\}_{k\in J}$ converges to (x_n^*, y_n^*). Suppose we rearrange the indices for K so that it aligns with the natural numbers.

Since $\mathcal{C}(n) = \emptyset$, $\underline{Q}_n(x_{a(n)}^k; \Phi_n^{k+1}) = \tilde{Q}_n(x_{a(n)}^k)$ for all $k \in K$, and so their convex envelopes over $\mathcal{Z}_{a(n)}$ equal to each other:

$$\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\underline{Q}_n(\cdot; \Phi_n^{k+1})\right)(x_{a(n)}^k) = \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k)$$

By the continuity of convex envelope $\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(\cdot)$ and the convergence of the sequence $\{(x_{a(n)}^k, y_{a(n)}^k)\}_{k\in K}$, the sequence $\left\{\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\underline{Q}_n(\cdot; \Phi_n^k)\right)(x_{a(n)}^k)\right\}_{k\in J}$ converges to $\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^*)$ with probability one.

We can derive that for $k \in K$,

$$\begin{aligned}
\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k) &\geq \phi_n^{k+1}(x_{a(n)}^k) && \text{(50a)}\\
&\geq \mathcal{L}_n(\pi_n^{k-1}; x_{a(n)}^{k-1}, \Phi_n^k, \mathcal{Z}_{a(n)}) + {\pi_n^{k-1}}^\top (x_{a(n)}^k - x_{a(n)}^{k-1}) && \text{(50b)}\\
&= \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\underline{Q}_n(\cdot; \Phi_n^k)\right)(x_{a(n)}^{k-1}) + {\pi_n^{k-1}}^\top (x_{a(n)}^k - x_{a(n)}^{k-1}) && \text{(50c)}\\
&= \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) + {\pi_n^{k-1}}^\top (x_{a(n)}^k - x_{a(n)}^{k-1}). && \text{(50d)}
\end{aligned}$$

We obtain the first inequality by Lemma 12, the second inequality by the definition of the $k-1$-th Lagrangian cut, and the third equality by the tightness result in Theorem 1. Therefore, if we subtract $\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k)$ from every part in inequalities (50), we obtain:

$$\begin{aligned}
0 &\geq \phi_n^{k+1}(x_{a(n)}^k) - \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k) && \text{(51a)}\\
&\geq \left(\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k)\right) + {\pi_n^{k-1}}^\top (x_{a(n)}^k - x_{a(n)}^{k-1}). && \text{(51b)}
\end{aligned}$$

We can take the absolute value of inequalities (51) to find a bound of the difference:

$$\begin{aligned}
\left|\phi_n^{k+1}(x_{a(n)}^k) - \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k)\right| \leq \left|\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k)\right| +&\\
\left\|\pi_n^{k-1}\right\| \left\|x_{a(n)}^k - x_{a(n)}^{k-1}\right\|.& && \text{(52)}
\end{aligned}$$

We have $\left|\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)(x_{a(n)}^k)\right| \to 0$ and $\left\|x_{a(n)}^k - x_{a(n)}^{k-1}\right\| \to 0$ with probability one, due to the continuity of $\mathrm{co}_{\mathcal{Z}_{a(n)}} \left(\tilde{Q}_n\right)$ and the convergence of $\{(x_n^k, y_n^k)\}_{k\in K}$. As we assume

$\left\|\pi_n^{k-1}\right\| < +\infty$, the second term converges to zero with probability one too. Then for the third statement in part (i), we can derive:

$$\left|\phi_n^{k+1}(x_{a(n)}^k) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^*)\right| \leq \left|\phi_n^{k+1}(x_{a(n)}^k) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^k)\right| + \\ \left|\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^k) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^*)\right| \tag{53}$$

The first term in the right-hand side converges to zero with probability one by inequality (52) and the second term converges to zero due to the continuity of $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)$ and the convergence of $\{(x_n^k, y_n^k)\}_{k\in K}$, and we prove part (i) for $n \in \mathcal{N}_T$.

Suppose for all $m \in \mathcal{C}(n)$ the three statements in part (i) are true. We can use a similar proof technique to show part (i) holds for $n \in \mathcal{N}\backslash\mathcal{N}_T$. The feasible region X_n is compact, so within the sequence K, there must exist an infinite subsequence J such that $\{(x_n^k, y_n^k)\}_{k\in J}$ converges to (x_n^*, y_n^*), given $\{(x_{a(n)}^k, y_{a(n)}^k)\}_{k\in K} \to (x_{a(n)}^*, y_{a(n)}^*)$.

We first show that $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n^k)\right)(x_{a(n)}^{k-1})$ converges to $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1})$ with probability one. Here we assume that a Lagrangian cut is only added to ϕ_n when both nodes $a(n)$ and n are within a sample path. This assumption does not affect the convergence of the cutting-plane algorithm because we assume every node's problem will be solved an infinite number of times, by the forward pass sampling property when we obtain a sampled solution and the backward pass sampling property when we generate a cut.

$$\begin{aligned}
0 &\geq \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n^k)\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) && \text{(54a)}\\
&\geq \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n^{k-1})\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) && \text{(54b)}\\
&= c_n^\top x_n^{k-1} + g_n^\top y_n^{k-1} + \sum_{m\in\mathcal{C}(n)} q_{nm}\phi_m^{k-1}(x_n^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) && \text{(54c)}\\
&\geq c_n^\top x_n^{k-1} + g_n^\top y_n^{k-1} + \sum_{m\in\mathcal{C}(n)} q_{nm}\phi_m^{k-1}(x_n^{k-1}) - \\
&\qquad \left(c_n^\top x_n^{k-1} + g_n^\top y_n^{k-1} + \sum_{m\in\mathcal{C}(n)} q_{nm}\mathrm{co}_{\mathcal{Z}_n}\left(\tilde{Q}_m\right)(x_n^{k-1})\right) && \text{(54d)}\\
&= \sum_{m\in\mathcal{C}(n)} q_{nm}\left(\phi_m^{k-1}(x_n^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_n^{k-1})\right). && \text{(54e)}
\end{aligned}$$

We can obtain inequality (54a) by Lemma 12, inequality (54b) because we are adding additional Lagrangian cuts to Φ_n^{k-1} to obtain Φ_n^k, equality (54c) because (x_n^{k-1}, y_n^{k-1}) is the optimal solution to the model (P_n) for the $k-1$-th time, and inequality (54d) by plugging in (x_n^{k-1}, y_n^{k-1}) as a feasible solution to $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1})$. Finally, the lower approximation function ϕ_n^k is α-Lipschitz continuous because $\|\pi\|$ is bounded by a finite number by our assumption. Therefore, Lemma 13 holds. Combined with the induction assumption $\phi_m^k(x_n^{k-1})$ converges to $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_n^{k-1})$, we can show the right-hand side of (54e), $\lim_{k\to\infty}\left(\phi_m^{k-1}(x_n^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_n^{k-1})\right) = 0$, when we set $\kappa = 1$.

For the third statement, we can use a similar argument as the one in the last stage. We need to replace the term $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1})$ by $\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n^k)\right)(x_{a(n)}^{k-1})$ in inequality (52), but we have

$$\left|\phi_n^{k+1}(x_{a(n)}^k) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^k)\right| \tag{55}$$

$$\leq \left|\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n^k)\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^k)\right| + \left\|\pi_n^{k-1}\right\|\left\|x_{a(n)}^k - x_{a(n)}^{k-1}\right\|. \tag{56}$$

$$\leq \left|\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot;\Phi_n^k)\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1})\right| + \left|\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^{k-1}) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_n\right)(x_{a(n)}^k)\right| + \left\|\pi_n^{k-1}\right\|\left\|x_{a(n)}^k - x_{a(n)}^{k-1}\right\|. \tag{57}$$

The right-hand side still converges to zero and the rest of the proof follows. There we prove three statements hold for any $n \in \mathcal{N}\backslash\mathcal{N}_T$, and thus finish part (i) of the theorem.

For part (ii) of the theorem, by part (i), we have with probability one

$$\lim_{i\to\infty}\Bigg((c_1^\top x_1^i + g_1^\top y_1^i + \sum_{m\in\mathcal{C}(1)} q_{1m}\phi_m^{i+1}(x_1^i)) - (c_1^\top x_1^* + g_1^\top y_1^* + \sum_{m\in\mathcal{C}(1)} q_{1m}\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_1^*))\Bigg) = 0. \tag{58a}$$

By Lemma (13), equality (58) is equivalent to

$$\lim_{i\to\infty}\Bigg((c_1^\top x_1^i + g_1^\top y_1^i + \sum_{m\in\mathcal{C}(1)} q_{1m}\phi_m^{i}(x_1^i)) - \tag{59a}$$

$$(c_1^\top x_1^* + g_1^\top y_1^* + \sum_{m\in\mathcal{C}(1)} q_{1m}\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_1^*))\Bigg) = 0 \tag{59b}$$

with probability one.

The accumulation point (x_1^*, y_1^*) is also a feasible solution to the model (14) because the feasible region X_1 and Y_1 are compact sets. Therefore,

$$c_1^\top x_1^* + g_1^\top y_1^* + \sum_{m\in\mathcal{C}(1)} q_{1m}\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_1^*) \tag{60a}$$

$$\geq \tilde{V}^* \tag{60b}$$

$$\geq c_1^\top x_1^i + g_1^\top y_1^i + \sum_{m\in\mathcal{C}(1)} q_{1m}\phi_m^{i+1}(x_1^i) \tag{60c}$$

$$= c_1^\top x_1^i + g_1^\top y_1^i + \sum_{m\in\mathcal{C}(1)} q_{1m}\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_1^i) + \sum_{m\in\mathcal{C}(1)} q_{1m}\left(\phi_m^{i+1}(x_1^i) - \mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_1^*)\right). \tag{60d}$$

Taking the limit of the right-hand side of inequality (60d) yields $c_1^\top x_1^* + g_1^\top y_1^* + \sum_{m\in\mathcal{C}(1)} q_{1m}\mathrm{co}_{\mathcal{Z}_{a(n)}}\left(\tilde{Q}_m\right)(x_1^*)$ as the limit of the last term is zero. Therefore, we prove the sequence $\{\underline{V}^i\}_{i\in\mathbb{N}}$ converges to $\tilde{V}^*$ with probability one, and every accumulation point of the sequence of the solution to model (P_1) is an optimal solution to model (14). This is the part (ii) of Theorem 2. □

C Properties of Sparse Lagrangian Cuts and Proof

Theorem 14. *For every node $n \in \mathcal{N}$, given Φ_n^L, $\mathcal{Z}_{a(n)}$, current partitions $\mathbb{P}_n$ and $\mathbb{P}_{a(n)}$, and an incumbent solution $(\hat{x}_{a(n)}, \hat{\lambda}_{a(n)}) \in \mathfrak{Z}_n(\mathcal{Z}_{a(n)}, \mathbb{P}_{a(n)})$. Let $(\pi_n^*, \rho_n^*, v_n^*)$ be the sparse cut coefficients obtained by solving (D_n^L) in model (18) with an updated Lagrangian relaxation function $\mathcal{L}_n^L$ in model (36). Then, the sparse Lagrangian cut is valid and tight in the following sense:*

(i) valid over $\mathfrak{Z}_n(\mathcal{Z}_{a(n)}, \mathbb{P}_{a(n)})$:

$$
\begin{aligned}
Q_n(x_{a(n)}) &\geq \underline{Q}_n^L(x_{a(n)}, \lambda_{a(n)}; \Phi_n^L, \mathbb{P}_n, \mathbb{P}_{a(n)}) \\
&\geq v_n^* + {\pi_n^*}^\top x_{a(n)} + {\rho_n^*}^\top \lambda_{a(n)}, \\
&\qquad \forall\, (x_{a(n)}, \lambda_{a(n)}) \in \mathfrak{Z}_n(\mathcal{Z}_{a(n)}, \mathbb{P}_{a(n)});
\end{aligned} \tag{61}
$$

(ii) tight at $(\hat{x}_{a(n)}, \hat{\lambda}_{a(n)})$ for the convex envelope of $\underline{Q}_n^L(\cdot, \cdot; \Phi_n^L, \mathbb{P}_n, \mathbb{P}_{a(n)})$ over $\mathfrak{Z}_n(\mathcal{Z}_{a(n)}, \mathbb{P}_{a(n)})$:

$$
\begin{aligned}
&v_n^* + {\pi_n^*}^\top \hat{x}_{a(n)} + {\rho_n^*}^\top \hat{\lambda}_{a(n)} = \\
&\qquad \mathrm{co}_{\mathfrak{Z}_n(\mathcal{Z}_{a(n)}, \mathbb{P}_{a(n)})}(\underline{Q}_n^L(\cdot, \cdot; \Phi_n^L, \mathbb{P}_n, \mathbb{P}_{a(n)}))(\hat{x}_{a(n)}, \hat{\lambda}_{a(n)})
\end{aligned} \tag{62}
$$

Proof. The proof for the validity in part (i) directly follows the proof in Theorem 1. Therefore, we focus on proving part (ii) here. We follow the proof of Theorem 3 to define $\mathcal{F}_n := \{(z_n, \mu_n, x_n, \lambda_n, y_n, \Theta_n) \mid \text{ constraints (19b)} - \text{(19f)}\}$.

Suppose we only dualize $\mu_{n,j}^{\hat{\sigma}_{a(n),j}} = \hat{\lambda}_{a(n),j}^{\hat{\sigma}_{a(n),j}}$ for $j \in \mathcal{J}_{a(n)}, \hat{\sigma}_{a(n),j} \in \mathcal{S}_{a(n),j}$ where $\hat{\lambda}_{a(n),j}^{\hat{\sigma}_{a(n),j}} = 1$. We follow Theorem 1 in Geoffrion (1974) and obtain

$$
\begin{aligned}
&v_n^* + {\pi_n^*}^\top \hat{x}_{a(n)} + \sum_{j \in \mathcal{J}_{a(n)}} \rho_{n,j}^{*,\hat{\sigma}_{a(n),j}} \hat{\lambda}_{a(n),j}^{\hat{\sigma}_{a(n),j}} \\
&= \min \Big\{ f_n(x_n, \lambda_n, y_n, \Theta_n) \mid (z_n, \mu_n, x_n, \lambda_n, y_n, \Theta_n) \in \mathrm{conv}(\mathcal{F}_n), z_n = \hat{x}_{a(n)}, \\
&\qquad\qquad \mu_{n,j}^{\hat{\sigma}_{a(n),j}} = 1 \quad \forall j \in \mathcal{J}_{a(n)} \Big\}.
\end{aligned}
$$

Since constraint (19d) specifies that $\mu_n \in \Lambda_{a(n)}(z_n, \mathbb{P}_{a(n)})$, we can find that the right-hand side is equivalent to

$$
\begin{aligned}
\min \Big\{ & f_n(x_n, \lambda_n, y_n, \Theta_n) \mid (z_n, \mu_n, x_n, \lambda_n, y_n, \Theta_n) \in \mathrm{conv}(\mathcal{F}_n), z_n = \hat{x}_{a(n)}, \\
& \mu_n = \hat{\lambda}_{a(n)} \quad \forall j \in \mathcal{J}_{a(n)} \Big\},
\end{aligned}
$$

because the condition $\mu_{n,j}^{\hat{\sigma}_{a(n),j}} = 1 \quad \forall j \in \mathcal{J}_{a(n)}$, together with $\mu_n \in \Lambda_{a(n)}(z_n, \mathbb{P}_{a(n)})$, implies the rest of elements within μ_n is all zero. The rest of the proof follows the same principle as the proof of Theorem 3. $\square$

D Theoretical Results in Section 4

D.1 Proof of Proposition 9

Proof. The existence of extreme points of $\mathcal{Z}_{a(n)}$ is given in Ichiishi (1983). Note that the value function approximation $\underline{Q}_n(\cdot; \Phi_n)$ is lower semi-continuous and piecewise polyhedral (Schultz 2003). By definition, $\hat{x}_{a(n)}$ is an extreme point of $\text{conv}(\text{dom}(\underline{Q}_n) \cap \mathcal{Z}_{a(n)})$. Therefore, according to Corollary 3 of Tawarmalani and Sahinidis (2002), the convex envelope of $\underline{Q}_n$ over $\text{dom}(\underline{Q}_n) \cap \mathcal{Z}_{a(n)}$ coincides with $\underline{Q}_n(\cdot; \Phi_n)$ at the extreme points of $\text{conv}(\text{dom}(\underline{Q}_n) \cap \mathcal{Z}_{a(n)})$. $\square$

D.2 Proof of Theorem 10

Proof. The Lagrangian cut supports $\text{co}_{\mathcal{Z}_{a(n)}}\left(\underline{Q}_n(\cdot; \Phi_n)\right)$ at $\hat{x}_{a(n)}$ because of its tightness. If $\hat{x}_{a(n)} \in \text{relint}(\text{conv}(\mathcal{Z}_{a(n)}))$, $\hat{x}_{a(n)}$ is a core point, which indicates Pareto-optimality by Theorem 3.4 in Brandenberg and Stursberg (2021). On the other hand, suppose that $\hat{x}_{a(n)}$ is on the boundary $\text{conv}(\mathcal{Z}_{a(n)})$ and the cut is not Pareto-optimal. That is, there exists $\bar{\pi}_n \in \Pi_n(\hat{x}_{a(n)})$ such that for any $x_{a(n)} \in \mathcal{Z}_{a(n)}$, the following inequality holds:

$$\begin{aligned}
&\mathcal{L}_n\left(\bar{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + {\bar{\pi}_n}^\top (x_{a(n)} - \hat{x}_{a(n)}) \\
\geq &\mathcal{L}_n\left(\pi_n^*; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + {\pi_n^*}^\top (x_{a(n)} - \hat{x}_{a(n)}).
\end{aligned} \tag{63}$$

Moreover, inequality (63) extends to all $x_{a(n)} \in \text{conv}(\mathcal{Z}_{a(n)})$.

By evaluating inequality (63) at $x_{a(n)} = \hat{x}_{a(n)}$, we deduce that $\bar{\pi}$ must be an optimal solution to CGCP model (41). Therefore, we have:

$$\begin{aligned}
&\mathcal{L}_n\left(\bar{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + {\bar{\pi}_n}^\top (x_{a(n)} - \tilde{x}_{a(n)}) \\
= &\mathcal{L}_n\left(\pi_n^*; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + {\pi_n^*}^\top (x_{a(n)} - \tilde{x}_{a(n)}).
\end{aligned} \tag{64}$$

If the Lagrangian cut defined by $\bar{\pi}_n$ dominates the Lagrangian cut defined by π_n^*, then there exists at least one point $\bar{x}_{a(n)} \in \mathcal{Z}_{a(n)}$ such that:

$$\begin{aligned}
&\mathcal{L}_n\left(\bar{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + {\bar{\pi}_n}^\top (\bar{x}_{a(n)} - \tilde{x}_{a(n)}) \\
> &\mathcal{L}_n\left(\pi_n^*; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + {\pi_n^*}^\top (\bar{x}_{a(n)} - \tilde{x}_{a(n)}).
\end{aligned} \tag{65}$$

Since $\tilde{x}_{a(n)} \in \text{relint}(\text{conv}(\mathcal{Z}_{a(n)}))$, there exists a scalar $\alpha > 1$ such that $x^0_{a(n)} = \alpha \tilde{x}_{a(n)} + (1 - \alpha)\bar{x}_{a(n)}$ and $x^0_{a(n)} \in \text{conv}(\mathcal{Z}_{a(n)})$. Multiplying equality (64) by α and inequality (65) by $1 - \alpha$, and adding the results yields:

$$\begin{aligned}&\mathcal{L}_n\left(\bar{\pi}_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + \bar{\pi}_n{}^\top (x^0_{a(n)} - \tilde{x}_{a(n)})\\ <&\mathcal{L}_n\left(\pi^*_n; \hat{x}_{a(n)}, \Phi_n, \mathcal{Z}_{a(n)}\right) + \pi^*_n{}^\top (x^0_{a(n)} - \tilde{x}_{a(n)}),\end{aligned}$$

which contradicts inequality (63) for all $x_{a(n)} \in \operatorname{conv}(\mathcal{Z}_{a(n)})$. Therefore, we conclude that the Lagrangian cut defined by π^*_n is Pareto-optimal. □

D.3 Example of SMC not Achieving Pareto-Optimality

Example 6. *We consider a value function Q_n defined on the scenario tree $\mathcal{N} = \{1, n\}$ with $a(n) = 1$ and $X_1 = \mathcal{Z}_1 = \{0, 1\}^2$:*

$$\begin{aligned} Q_n(x_1) = \min \quad & y_n \\ s.t. \quad & y_n = 2\left(x_{1,1} + x_{1,2}\right) - 1 \\ & y_n \in \mathbb{Z}. \end{aligned}$$

Suppose we obtain an incumbent solution $\hat{x}_1 = [1\ 0]^\top$. The Lagrangian dual problem for Q_n given $\hat{x}_1$ is

$$\max_{\pi_n} \quad \mathcal{L}_n(\pi_n; \hat{x}_1, \mathcal{Z}_1), \tag{66}$$

where

$$\begin{aligned} \mathcal{L}_n(\pi_n; \hat{x}_1, \mathcal{Z}_1) = \min \quad & y_n + \pi_n^\top (\hat{x}_1 - z_n) \\ s.t. \quad & y_n = 2\left(z_{n,1} + z_{n,2}\right) - 1 \\ & y_n \in \mathbb{Z},\ z_n \in \mathcal{Z}_1. \end{aligned}$$

We find that the closed form of function $\mathcal{L}_n(\pi_n; \hat{x}_1, \mathcal{Z}_1)$ can be represented by a piecewise linear concave function as $\mathcal{L}_n(\pi_n; \hat{x}_1, \mathcal{Z}_1) = \min\left\{\pi_{n,1} - 1, 1, 1 + \pi_{n,1} - \pi_{n,2}, 3 - \pi_{n,2}\right\}$ and the set of optimal solutions to (66) is $\Pi_n(\hat{x}_1) = \{\pi_n \mid \pi_{n,1} \geq 2, \pi_{n,2} \leq 2\}$. Therefore, any $\pi_n \in \Pi_n(\hat{x}_1)$ can be used to generate a tight Lagrangian cut at $\hat{x}_1$, which takes a form of $\theta_n \geq \pi_n^\top x_1 + (1 - \pi_n^\top \hat{x}_1)$.

If we solve a CGCP according to model (42) to generate an SMC, the optimal solution to model (42) is $\pi_n = [2\ 0]^\top$ and we can generate a Lagrangian cut

$$\theta_n \geq 2x_{1,1} - 1.$$

However, this cut is dominated at every point on $\mathcal{Z}_1$ by the following Pareto-optimal cut,

$$\theta_n \geq 2x_{1,1} + 2x_{1,2} - 1,$$

where the corresponding coefficients are $\pi_n = [2\ 2]^\top$. □