FULL LENGTH PAPER

Series B

Stochastic dual dynamic programming for multistage stochastic mixed-integer nonlinear optimization

Shixuan Zhang[1] · Xu Andy Sun[2]

Abstract

In this paper, we study multistage stochastic mixed-integer nonlinear programs (MS-MINLP). This general class of problems encompasses, as important special cases, multistage stochastic convex optimization with *non-Lipschitzian* value functions and multistage stochastic mixed-integer linear optimization. We develop stochastic dual dynamic programming (SDDP) type algorithms with nested decomposition, deterministic sampling, and stochastic sampling. The key ingredient is a new type of cuts based on generalized conjugacy. Several interesting classes of MS-MINLP are identified, where the new algorithms are guaranteed to obtain the global optimum without the assumption of complete recourse. This significantly generalizes the classic SDDP algorithms. We also characterize the iteration complexity of the proposed algorithms. In particular, for a $(T+1)$-stage stochastic MINLP satisfying L-exact Lipschitz regularization with d-dimensional state spaces, to obtain an ε-optimal root node solution, we prove that the number of iterations of the proposed deterministic sampling algorithm is upper bounded by $\mathcal{O}((\frac{2LT}{\varepsilon})^d)$, and is lower bounded by $\mathcal{O}((\frac{LT}{4\varepsilon})^d)$ for the general case or by $\mathcal{O}((\frac{LT}{8\varepsilon})^{d/2-1})$ for the convex case. This shows that the obtained complexity bounds are rather sharp. It also reveals that the iteration complexity depends *polynomially* on the number of stages. We further show that the iteration complexity depends *linearly* on T, if all the state spaces are finite sets, or if we seek a $(T\varepsilon)$-optimal solution when the state spaces are infinite sets, i.e. allowing the optimality gap to scale with T. To the best of our knowledge, this is the first work that reports global optimization algorithms as well as iteration complexity results for solving such a large class of multistage stochastic programs. The iteration complexity study resolves a conjec-

✉ Xu Andy Sun
sunx@mit.edu

Shixuan Zhang
szhang483@gatech.edu

1 H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA

2 Sloan School of Management, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, USA

ture by the late Prof. Shabbir Ahmed in the general setting of multistage stochastic mixed-integer optimization.

Keywords Multistage stochastic programming · MINLP · SDDP · Complexity analysis

Mathematics Subject Classification 90C15 · 90C11 · 90C30 · 90C60 · 90C39 · 49N15

1 Introduction

A multistage stochastic mixed-integer nonlinear program (MS-MINLP) is a sequential decision making problem under uncertainty with both continuous and integer decisions and nonconvex nonlinear objective function and constraints. This provides an extremely powerful modeling framework. Special classes of MS-MINLP, such as multistage stochastic linear programming (MS-LP) and mixed-integer linear programming (MS-MILP), have already found a wide range of applications in diverse fields such as electric power system scheduling and expansion planning [3, 36, 37, 39], portfolio optimization under risk [8, 21, 26], and production and capacity planning problems [2, 4, 9, 12], just to name a few.

Significant progress has been made in the classic nested Benders decomposition (NBD) algorithms for solving MS-LP with general scenario trees, and an efficient random sampling variation of NBD, the stochastic dual dynamic programming (SDDP) algorithm, is developed for MS-LP with scenario trees having stagewise independent structures. In the past few years, these algorithms are extended to solve MS-MILP [31]. For example, SDDP is generalized to Stochastic Dual Dynamic integer Programming (SDDiP) algorithm for global optimization of MS-MILP with binary state variables [38, 39]. Despite the rapid development, key challenges remain in further extending SDDP to the most general problems in MS-MINLP: 1) There is no general cutting plane mechanism for generating exact under-approximation of nonconvex, discontinuous, or non-Lipschitzian value functions; 2) The computational complexity of SDDP-type algorithms is not well understood even for the most basic MS-LP setting, especially the interplay between iteration complexity of SDDP, optimality gap of obtained solution, number of stages, and dimension of the state spaces of the MS-MINLP.

This paper aims at developing new methodologies for the solution of these challenges. In particular, we develop a unified cutting plane mechanism in the SDDP framework for generating exact under-approximation of value functions of a large class of MS-MINLP, and develop sharp characterization of the iteration complexity of the proposed algorithms. In the remaining of this section, we first give an overview of the literature, then summarize more details of our contributions.

1.1 Literature review

Benders decomposition [6], Dantzig-Wolfe decomposition [11], and the L-shaped method [35] are standard algorithms for solving two-stage stochastic LPs. Nested

decomposition procedures for deterministic models are developed in [16, 20]. Louveaux [25] first generalized the two-stage L-shaped method to multistage quadratic problems. Nested Benders decomposition for MS-LP was first proposed in Birge [7] and Pereira and Pinto [28]. SDDP, the sampling variation of NBD, was first proposed in [29]. The largest consumer of SDDP by far is in the energy sector, see e.g. [14, 23, 29, 34].

Recently, SDDP has been extended to SDDiP [38]. It is observed that the cuts generated from Lagrangian relaxation of the nodal problems in an MS-MILP are always tight at the given parent node's state, as long as all the state variables only take binary values and have complete recourse. From this fact, the SDDiP algorithm is proved to find an exact optimal solution in finitely many iterations with probability one. In this way, the SDDiP algorithm makes it possible to solve nonconvex problems through binarization of the state variables [19, 39]. In addition, when the value functions of MS-MILP with general integer state variables are assumed to be Lipschitz continuous, which is a critical assumption, augmented Lagrangian cuts with an additional reverse norm term to the linear part obtained via augmented Lagrangian duality are proposed in [1].

The convergence analysis of the SDDP-type algorithms begins with the linear cases [10, 18, 24, 30, 33], where almost sure finite convergence is established based on the polyhedral nodal problem structures. For convex problems, if the value functions are Lipschitz continuous and the state space is compact, asymptotic convergence of the under-approximation of the value functions leads to asymptotic convergence of the optimal value and optimal solutions [15, 17]. By constructing over-approximations of value functions, an SDDP with a deterministic sampling method with asymptotic convergence is proposed for the convex case in [5]. Upon completion of this paper, we became aware of the recent work [22], which proves iteration complexity upper bounds for multistage convex programs under the assumption that all the value functions and their under-approximations are all Lipschitz continuous. It is shown that for discounted problems, the iteration complexity depends linearly on the number of stages. However, the following conjecture (suggested to us by the late Prof. Shabbir Ahmed) remains to be resolved, especially for the problems without convexity, Lipschitz continuity, or discounts:

Conjecture 1 The number of iterations needed for SDDP/SDDiP to find an optimal first-stage solution grows linearly in terms of the number of stages T, while it may depend nonlinearly on other parameters such as the diameter D and the dimension d of the state space.

Our study resolves this conjecture by giving a full picture of the iteration complexity of SDDP-type algorithms in a general setting of MS-MINLP problems that allow exact Lipschitz regularization (defined in Sect. 2.3). In the following, we summarize our contributions.

1.2 Contributions

1. To tackle the MS-MINLP problems without Lipschitz continuous value functions, which the existing SDDP algorithms and complexity analyses cannot handle, we propose *a regularization approach* to provide a surrogate of the original problem

such that the value functions become Lipschitz continuous. In many cases, the regularized problem preserves the set of optimal solutions.

2. We use the theory of generalized conjugacy to develop a cut generation scheme, referred to as *generalized conjugacy cuts*, that are valid for value functions of MS-MINLP. Moreover, generalized conjugacy cuts are shown to be tight to the regularized value functions. The generalized conjugacy cuts can be replaced by linear cuts without compromising such tightness when the problem is convex.

3. With the regularization and the generalized conjugacy cuts, we propose three algorithms for MS-MINLP, including nested decomposition for general scenario trees, SDDP algorithms with random sampling as well as deterministic sampling similar to [5] for stagewise independent scenario trees.

4. We obtain upper and lower bounds on the iteration complexity for the proposed SDDP with both sampling methods for MS-MINLP problems. The complexity bounds show that in general, Conjecture 1 holds if only we seek a $(T\varepsilon)$-optimal solution, instead of an ε-optimal first-stage solution for a $(T+1)$-stage problem, or when all the state spaces are finite sets.

In addition, this paper contains the following contributions compared with the recent independent work [22]: (1) We consider a much more general class of problems which are not necessarily convex. As a result, all the iteration complexity upper bounds of the algorithms are also valid for these nonconvex problems. (2) We use the technique of regularization to make the iteration complexity bounds independent of the subproblem oracles. This is particularly important for the conjecture, since the Lipschitz constants of the under-approximation of value functions may exceed those of the original value functions. (3) We propose matching lower bounds on the iteration complexity of the algorithms and characterize important cases for the conjecture to hold.

This paper is organized as follows. In Sect. 2 we introduce the problem formulation, regularization of the value functions, and the approximation scheme using generalized conjugacy. Sect. 3 proposes SDDP algorithms. Sect. 4 investigates upper bounds on the iteration complexity of the proposed algorithm, while Sect. 5 focuses on lower bounds, therefore completes the picture of iteration complexity analysis. We finally provide some concluding remarks in Sect. 6.

2 Problem formulations

In this section, we first present the extensive and recursive formulations of multistage optimization. Then we characterize the properties of the value functions, with examples to show that they may fail to be Lipschitz continuous. With this motivation in mind, we propose a penalty reformulation of the multistage problem through regularization of value functions and show that it is equivalent to the original formulation for a broad class of problems. Finally, we propose generalized conjugacy cuts for under-approximation of value functions.

2.1 Extensive and recursive formulation

For a multistage stochastic program, let $\mathcal{T} = (\mathcal{N}, \mathcal{E})$ be the scenario tree, where $\mathcal{N}$ is the set of nodes and $\mathcal{E}$ is the set of edges. For each node $n \in \mathcal{N}$, let $a(n)$ denote the

parent node of n, $\mathcal{C}(n)$ denote the set of child nodes of n, and $\mathcal{T}(n)$ denote the subtree starting from the node n. Given a node $n \in \mathcal{N}$, let $t(n)$ denote the stage that the node n is in and let $T := \max_{n \in \mathcal{N}} t(n)$ denote the last stage of the tree $\mathcal{T}$. A node in the last stage is called a leaf node, otherwise a non-leaf node. The set of nodes in stage t is denoted as $\mathcal{N}(t) := \{n \in \mathcal{N} : t(n) = t\}$. We use the convention that the root node of the tree is denoted as $r \in \mathcal{N}$ with $t(r) = 0$ so the total number of stages is $T + 1$. The parent node of the root node is denoted as $a(r)$, which is a dummy node for ease of notation.

For every node $n \in \mathcal{N}$, let $\mathcal{F}_n$ denote the feasibility set in some Euclidean space of decision variables (x_n, y_n) of the nodal problem at node n. We refer to x_n as the state variable and y_n as the internal variable of node n. Denote the image of the projection of $\mathcal{F}_n$ onto the subspace of the variable x_n as $\mathcal{X}_n$, which is referred to as the state space. Let $x_{a(r)} = 0$ serve as a dummy parameter and thus $\mathcal{X}_{a(r)} = \{0\}$. The nonnegative nodal cost function of the problem at node n is denoted as $f_n(x_{a(n)}, y_n, x_n)$ and is defined on the set $\{(z, y, x) : z \in \mathcal{X}_{a(n)}, (x, y) \in \mathcal{F}_n\}$. We allow f_n to take the value $+\infty$ so indicator functions can be modeled as part of the cost. Let $p_n > 0$ for all $n \in \mathcal{N}$ denote the probability that node n on the scenario tree is realized. For the root node, $p_r = 1$. The transition probability that node m is realized conditional on its parent node n being realized is given by $p_{nm} := p_m / p_n$ for all edges $(n, m) \in \mathcal{E}$.

The multistage stochastic program considered in this paper is defined in the following extensive form:

$$v^{\mathrm{prim}} := \min_{\substack{(x_n, y_n) \in \mathcal{F}_n, \\ \forall n \in \mathcal{N}}} \sum_{n \in \mathcal{N}} p_n f_n(x_{a(n)}, y_n, x_n). \tag{1}$$

The recursive formulation of the problem (1) is defined as

$$Q_n(x_{a(n)}) := \min_{(x_n, y_n) \in \mathcal{F}_n} \left\{ f_n(x_{a(n)}, y_n, x_n) + \sum_{m \in \mathcal{C}(n)} p_{nm} Q_m(x_n) \right\}, \tag{2}$$

where $n \in \mathcal{T}$ is a non-leaf node and $Q_n(x_{a(n)})$ is the *value function* of node n. At a leaf node, the sum in (2) reduces to zero, as there are no child nodes $\mathcal{C}(n) = \varnothing$. The problem on the right-hand side of (2) is called the *nodal problem* of node n. Its objective function consists of the nodal cost function f_n and the *expected cost-to-go function*, which is denoted as $\mathcal{Q}_n$ for future reference, i.e.

$$\mathcal{Q}_n(x_n) := \sum_{m \in \mathcal{C}(n)} p_{nm} Q_m(x_n). \tag{3}$$

To ensure that the minimum in problem (1) is well defined and finite, we make the following very general assumption on f_n and $\mathcal{F}_n$ throughout the paper.

Assumption 1 For every node $n \in \mathcal{N}$, the feasibility set $\mathcal{F}_n$ is compact, and the nodal cost function f_n is nonnegative and lower semicontinuous (l.s.c.). The sum $\sum_{n \in \mathcal{N}} p_n f_n$ is a proper function, i.e., there exists $(x_n, y_n) \in \mathcal{F}_n$ for all nodes $n \in \mathcal{N}$ such that the sum $\sum_{n \in \mathcal{N}} p_n f_n(x_{a(n)}, y_n, x_n) < +\infty$.

Note that the state variable $x_{a(n)}$ only appears in the objective function f_n of node n, not in the constraints. Perhaps the more common way is to allow $x_{a(n)}$ to appear in the constraints of node n. It is easy to see that any such constraint can be modeled by an indicator function of $(x_{a(n)}, x_n, y_n)$ in the objective f_n.

2.2 Continuity and convexity of value functions

The following proposition presents some basic properties of the value function Q_n under Assumption 1.

Proposition 1 *Under Assumption* 1, *the value function Q_n is lower semicontinuous (l.s.c.) for all $n \in \mathcal{N}$. Moreover, for any node $n \in \mathcal{N}$,*

1. *if $f_n(z, y, x)$ is Lipschitz continuous in the first variable z with constant l_n, i.e. $|f_n(z, y, x) - f_n(z', y, x)| \leq l_n \|z - z'\|$ for any $z, z' \in \mathcal{X}_{a(n)}$ and any $(x, y) \in \mathcal{F}_n$, then Q_n is also Lipschitz continuous with constant l_n;*
2. *if $\mathcal{X}_{a(n)}$ and $\mathcal{F}_n$ are convex sets, and f_n and $\mathcal{Q}_n$ are convex functions, then Q_n is also convex.*

The proof is given in Sect. A.1.1. When Q_m is l.s.c. for all $m \in \mathcal{C}(n)$, the sum $\sum_{m \in \mathcal{C}(n)} p_{nm} Q_m$ is l.s.c.. Therefore, the minimum in the definition (2) is well define, since $\mathcal{F}_n$ is assumed to be compact.

If the objective function $f_n(x_{a(n)}, y_n, x_n)$ is not Lipschitz, e.g., when it involves an indicator function of $x_{a(n)}$, or equivalently when $x_{a(n)}$ appears in the constraint of the nodal problem of $Q_n(x_{a(n)})$, then the value function Q_n may not be Lipschitz continuous, as is shown by the following examples.

Example 1 Consider the convex nonlinear two-stage problem

$$v^* := \min_{x,z,w} \left\{ x + z \; : \; (z-1)^2 + w^2 \leq 1, \; w = x, \; x \in [0, 1] \right\}.$$

The objective function and all constraints are Lipschitz continuous. The optimal objective value $v^* = 0$, and the unique optimal solution is $(x^*, z^*, w^*) = (0, 0, 0)$. At the optimal solution, the inequality constraint is active. Note that the problem can be equivalently written as $v^* = \min_{0 \leq x \leq 1} x + Q(x)$, where $Q(x)$ is defined on $[0, 1]$ as $Q(x) := \min \left\{ z : \exists w \in \mathbb{R}, \text{ s.t. } (z-1)^2 + w^2 \leq 1, \; w = x \right\} = 1 - \sqrt{1 - x^2}$, which is not locally Lipschitz continuous at the boundary point $x = 1$. Therefore, $Q(x)$ is not Lipschitz continuous on $[0, 1]$.

Example 2 Consider the mixed-integer linear two-stage problem

$$v^* := \min \left\{ 1 - 2x + z \; : \; z \geq x, \; x \in [0, 1], \; z \in \{0, 1\} \right\}.$$

The optimal objective value is $v^* = 0$, and the unique optimal solution is $(x^*, z^*) = (1, 1)$. Note that the problem can be equivalently written as $v^* = \min\{1 - 2x + Q(x) \; : \; 0 \leq x \leq 1\}$, where the function $Q(x)$ is defined on $[0, 1]$ as $Q(x) := \min\{z \in \{0, 1\} : z \geq x\}$, which equals 0 if $x = 0$, and 1 for all $0 < x \leq 1$, i.e. $Q(x)$ is discontinuous at $x = 0$, therefore, it is not Lipschitz continuous on $[0, 1]$.

These examples show a major issue with the introduction of value functions Q_n, namely Q_n may fail to be Lipschitz continuous even when the original problem only has constraints defined by Lipschitz continuous functions. This could lead to failure of algorithms based on approximation of the value functions. In the next section, we will discuss how to circumvent this issue without compromise of feasibility or optimality for a wide range of problems.

2.3 Regularization and penalty reformulation

The main idea of avoiding failure of cutting plane algorithms in multistage dynamic programming is to use some Lipschitz continuous envelope functions to replace the original value functions, which we refer to as *regularized value functions*.

To begin with, we say a function $\psi : \mathbb{R}^d \to \mathbb{R}_+$ is a *penalty function*, if $\psi(x) = 0$ if and only if $x = 0$, and the diameter of its level set $\text{lev}_a(\psi) := \{x \in \mathbb{R}^d : \psi(x) \le \alpha\}$ approaches 0 when $a \to 0$. In this paper, we focus on penalty functions that are locally Lipschitz continuous, the reason for which will be clear from Proposition 2.

For each node n, we introduce a new variable z_n as a local variable of node n and impose the duplicating constraint $x_{a(n)} = z_n$. This is a standard approach for obtaining dual variables through relaxation (e.g. [38]). The objective function can then be written as $f_n(z_n, y_n, x_n)$. Let ψ_n be a penalty function for node $n \in \mathcal{N}$. The new coupling constraint is relaxed and penalized in the objective function by $\sigma_n \psi_n(x_{a(n)} - z_n)$ for some $\sigma_n > 0$. Then the DP recursion with penalization becomes

$$Q_n^{\mathrm{R}}(x_{a(n)}) := \min_{\substack{(x_n, y_n) \in \mathcal{F}_n, \\ z_n \in \mathcal{X}_{a(n)}}} \left\{ f_n(z_n, y_n, x_n) + \sigma_n \psi_n(x_{a(n)} - z_n) + \sum_{m \in \mathcal{C}(n)} p_{nm} Q_m^{\mathrm{R}}(x_n) \right\}, \quad (4)$$

for all $n \in \mathcal{N}$, and Q_n^{R} is referred to as the regularized value function. By convention, $\mathcal{X}_{a(r)} = \{x_{a(r)}\} = \{0\}$ and therefore, penalization $\psi_r(x_{a(r)} - z_r) \equiv 0$ for any $z_r \in \mathcal{X}_{a(r)}$. Since the state spaces are compact, without loss of generality, we can scale the penalty functions ψ_n such that the Lipschitz constant of ψ_n on $\mathcal{X}_{a(n)} - \mathcal{X}_{a(n)}$ is 1. The following proposition shows that Q_n^{R} is a Lipschitz continuous envelope function of Q_n for all nodes n.

Proposition 2 *Suppose ψ_n is a 1-Lipschitz continuous penalty function on the compact set $\mathcal{X}_{a(n)} - \mathcal{X}_{a(n)}$ for all $n \in \mathcal{N}$. Then $Q_n^{\mathrm{R}}(x) \le Q_n(x)$ for all $x \in \mathcal{X}_{a(n)}$ and $Q_n^{\mathrm{R}}(x)$ is σ_n-Lipschitz continuous on $\mathcal{X}_{a(n)}$. Moreover, if the original problem (2) and the penalty functions ψ_n are convex, then $Q_n^{\mathrm{R}}(x)$ is also convex.*

The key idea is that by adding a Lipschitz function ψ_n into the nodal problem, we can make $Q_n^{\mathrm{R}}(x)$ Lipschitz continuous even when $Q_n(x)$ is not. The proof is given in Sect. A.1.2. The optimal value of the regularized root nodal problem

$$v^{\mathrm{reg}} := \min_{(x_r, y_r) \in \mathcal{F}_r} \left\{ f_r(x_{a(r)}, y_r, x_r) + \sum_{m \in \mathcal{C}(r)} p_{rm} Q_m^{\mathrm{R}}(x_r) \right\} \quad (5)$$

is thus an underestimation of v^{prim}, i.e. $v^{\mathrm{reg}} \le v^{\mathrm{prim}}$. For notational convenience, we also define the regularized expected cost-to-go function for each node n as:

$$Q_n^{\mathrm{R}}(x_n) := \sum_{m \in \mathcal{C}(n)} p_{nm} Q_m^{\mathrm{R}}(x_n). \tag{6}$$

Definition 1 For any $\varepsilon > 0$, a feasible root node solution $(x_r, y_r) \in \mathcal{F}_r$ is said to be ε-*optimal* to the regularized problem (4) if it satisfies $f_r(x_{a(r)}, y_r, x_r) + Q_r^{\mathrm{R}}(x_r) \le v^{\mathrm{reg}} + \varepsilon$.

Next we discuss conditions under which $v^{\mathrm{reg}} = v^{\mathrm{prim}}$ and any optimal solution $(x_n, y_n)_{n \in \mathcal{N}}$ to the regularized problem (4) is feasible and hence optimal to the original problem (2). Note that by expanding Q_m^{R} in the regularized problem (4) for all nodes, we obtain the extensive formulation for the regularized problem:

$$v^{\mathrm{reg}} = \min_{\substack{(x_n, y_n) \in \mathcal{F}_n, n \in \mathcal{N} \\ z_n \in \mathcal{X}_{a(n)}}} \sum_{n \in \mathcal{N}} p_n \left(f_n(z_n, y_n, x_n) + \sigma_n \psi_n(x_{a(n)} - z_n) \right). \tag{7}$$

We refer to problem (7) as the *penalty reformulation* and make the following assumption on its exactness for the rest of the paper.

Assumption 2 We assume that the penalty reformulation (7) is exact for the given penalty parameters $\sigma_n > 0, n \in \mathcal{N}$, i.e., any optimal solution of (7) satisfies $z_n = x_{a(n)}$ for all $n \in \mathcal{N}$.

Assumption 2 guarantees the solution of the regularized extensive formulation (7) is feasible for the original problem (1), then by the fact that $v^{\mathrm{reg}} \le v^{\mathrm{prim}}$, is also optimal to the original problem, we have $v^{\mathrm{reg}} = v^{\mathrm{prim}}$. Thus regularized value functions serve as a surrogate of the original value function, without compromise of feasibility of its optimal solutions. A consequence of Assumption 2 is that the original and regularized value functions coincide at all optimal solutions, the proof of which is given in Sect. A.1.3.

Lemma 1 *Under Assumption 2, any optimal solution $(x_n, y_n)_{n \in \mathcal{N}}$ to problem (1) satisfies $Q_n^{\mathrm{R}}(x_{a(n)}) = Q_n(x_{a(n)})$ for all $n \ne r$.*

We illustrate the regularization on the examples through Fig. 1a and b. In Fig. 1a, the value function $Q(x)$ derived in Example 1 is not Lipschitz continuous at $x = 1$. With $\psi(x) = \|x\|$ and $\sigma = 4/3$, we obtain the regularized value function, which coincides with the original one on $[0, 0.8]$ and is Lipschitz continuous on the entire interval $[0, 1]$. In Fig. 1b, the value function $Q(x)$ derived in Example 1 is not continuous at $x = 0$. With $\psi(x) = \|x\|$, $\sigma = 5$, we obtain the regularized value function, which coincides with the primal one on $\{0\} \cup [0.2, 1]$ and is Lipschitz continuous on the entire interval $[0, 1]$. In both examples, it can be easily verified that the penalty reformulation is exact and thus preserves optimal solution.

We comment that Assumption 2 can hold for appropriately chosen penalty factors in various mixed-integer nonlinear optimization problems, including
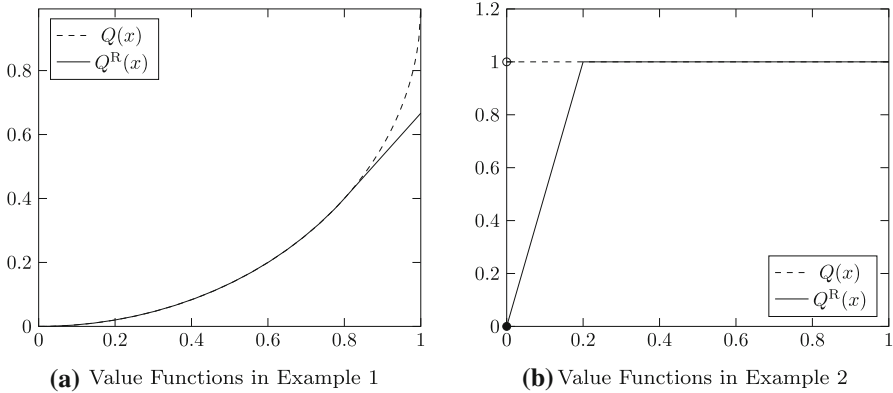
(a) Value Functions in Example 1   **(b)** Value Functions in Example 2

Fig. 1 Value functions in Examples 1 and 2

- convex problems with interior feasible solutions,
- problems with finite state spaces,
- problems defined by mixed-integer linear functions, and
- problems defined by continuously differentiable functions,

if certain constraint qualification is satisfied and proper penalty functions are chosen. We refer the readers to Sect. B in the Appendix for detailed discussions. We emphasize that Assumption 2 should be interpreted as a restriction on the MS-MINLP problem class studied in this paper. Namely all problem instances in our discussion must satisfy Assumption 2 with a given set of penalty functions ψ_n and penalty parameters σ_n, while they can have other varying problem data such as the numbers of stages T and characteristics of the state spaces $\mathcal{X}_n$. In general, it is possible that a uniform choice of σ_n needs to grow with T to satisfy the assumption.

2.4 Generalized conjugacy cuts and value function approximation

In this part, we first introduce generalized conjugacy cuts for nonconvex functions and then apply it to the under-approximation of value functions of MS-MINLP.

2.4.1 Generalized conjugacy cuts

Let $Q : \mathcal{X} \rightarrow \mathbb{R}_+ \cup \{+\infty\}$ be a proper, l.s.c. function defined on a compact set $\mathcal{X} \subseteq \mathbb{R}^d$. Let $\mathcal{U}$ be a nonempty set for dual variables. Given a continuous function $\Phi : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}$, the Φ-conjugate of Q (see e.g., Chapter 11-L in [32]) is defined as

$$Q^{\Phi}(u) = \max_{x \in \mathcal{X}} \{\Phi(x, u) - Q(x)\}. \tag{8}$$

The following generalized Fenchel-Young inequality holds by definition for any $x \in \mathcal{X}$ and $u \in \mathcal{U}$,

$$Q(x) + Q^{\Phi}(u) \geq \Phi(x, u).$$

For any $\hat{u} \in \mathcal{U}$ and an associated maximizer $\hat{x}$ in (8), we define

$$C^{\Phi}(x \mid \hat{u}, \hat{v}) := \hat{v} + \Phi(x, \hat{u}) \tag{9}$$

where $\hat{v} := -Q^{\Phi}(\hat{u})$. Then, the following inequality, derived from the generalized Fenchel-Young inequality, is valid for any $x \in \mathcal{X}$,

$$Q(x) \geq C^{\Phi}(x \mid \hat{u}, \hat{v}), \tag{10}$$

which we call a *generalized conjugacy cut* for the target function Q.

2.4.2 Value function approximation

The generalized conjugacy cuts can be used in the setting of an augmented Lagrangian dual [1] with bounded dual variables. For a nodal problem $n \in \mathcal{N}$, $n \neq r$ and a point $\bar{x} \in \mathcal{X}_{a(n)}$, define $\Phi_n^{\bar{x}}(x, u) := -\langle \lambda, \bar{x} - x \rangle - \rho \psi_n(\bar{x} - x)$, where $u := (\lambda, \rho) \in \mathbb{R}^{d_n + 1}$ are parameters. Consider a compact set of parameters $\mathcal{U}_n = \{(\lambda, \rho) : \|\lambda\|_* \leq l_{n,\lambda}, 0 \leq \rho \leq l_{n,\rho}\}$ with nonnegative bounds $l_{n,\lambda}$ and $l_{n,\rho}$, where $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$. Consider the following dual problem

$$\hat{v}_n := \max_{(\lambda, \rho) \in \mathcal{U}_n} \left\{ \min_{z \in \mathcal{X}_{a(n)}} [Q_n(z) + \langle \lambda, \bar{x} - z \rangle + \rho \psi_n(\bar{x} - z)] \right\}. \tag{11}$$

Denote $\hat{z}_n$ and $(\hat{\lambda}_n, \hat{\rho}_n)$ as an optimal primal-dual solution of (11). The dual problem (11) can be viewed as choosing $(\hat{\lambda}_n, \hat{\rho}_n)$ as the value of $\hat{u}$ in (9), which makes the constant term $-Q^{\Phi}(\hat{u})$ as large as possible, thus makes the generalized conjugacy cut (10) as tight as possible. With this choice of the parameters, a generalized conjugacy cut for Q_n at $\bar{x}$ is given by

$$\begin{aligned} Q_n(x) &\geq C_n^{\Phi_n^{\bar{x}}}(x \mid \hat{\lambda}_n, \hat{\rho}_n, \hat{v}_n) \\ &= -\langle \hat{\lambda}_n, \bar{x} - x \rangle - \hat{\rho}_n \psi_n(\bar{x} - x) + \hat{v}_n, \quad \forall x \in \mathcal{X}_{a(n)}. \end{aligned} \tag{12}$$

Proposition 3 *Given the above definition of (11)–(12), if $(\bar{x}_n, \bar{y}_n)_{n \in \mathcal{N}}$ is an optimal solution to problem (1) and the bound $l_{n,\rho}$ satisfies $l_{n,\rho} \geq \sigma_n$ for all nodes n, then for every node n, the generalized conjugacy cut (12) is tight at $\bar{x}_n$, i.e. $Q_n(\bar{x}_n) = C_n^{\Phi_n^{\bar{x}_n}}(\bar{x}_n \mid \hat{\lambda}_n, \hat{\rho}_n, \hat{v}_n)$.*

The proof is given in Sect. A.1.4. The proposition guarantees that, under Assumption 2, the generalized conjugacy cuts are able to approximate the value functions exactly at any state associated to an optimal solution.

In the special case where problem (2) is convex and $\psi_n(x) = \|x\|$ for all $n \in \mathcal{N}$, the exactness of the generalized conjugacy cut holds even if we set $l_{n,\rho} = 0$, i.e. the conjugacy cut is linear. To be precise, we begin with the following lemma.

Lemma 2 *Let $\mathcal{X} \subset \mathbb{R}^d$ be a convex, compact set. Given a convex, proper, l.s.c. function $Q : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$, for any $x \in \mathcal{X}$, the inf-convolution satisfies*

$$Q\square(\sigma\,\|\cdot\|)(x):=\min_{z\in\mathcal{X}}\{Q(z)+\sigma\,\|x-z\|\}=\max_{\|\lambda\|_{*}\leq\sigma}\min_{z\in\mathcal{X}}\{Q(z)+\langle\lambda,x-z\rangle\}.\tag{13}$$

The proof are given in Sect. A.1.5. Next we show the tightness in the convex case similar to Proposition 3. The proof is given in Sect. A.1.6.

Proposition 4 *Suppose* (2) *is convex and* $\psi_n(x)=\|x\|$ *for all nodes* n. *Given the above definition of* (11)–(12), *if* $(\bar{x}_n,\bar{y}_n)_{n\in\mathcal{N}}$ *is an optimal solution to problem* (1) *and the bounds satisfy* $l_{n,\lambda}\geq\sigma_n$, $l_{n,\rho}=0$ *for all nodes* n, *then for every node* n, *the generalized conjugacy cut* (12) *is exact at* $\bar{x}_n$, *i.e.* $Q_n(\bar{x}_n)=C_n^{\Phi_n^{\bar{x}_n}}(\bar{x}_n\,|\,\hat{\lambda}_n,\hat{\rho}_n,\hat{v}_n)$.

In this case, the generalized conjugacy reduces to the usual conjugacy for convex functions and the generalized conjugacy cut is indeed linear. This enables approximation of the value function that preserves convexity.

Remark 1 This proposition can be generalized to special nonconvex problems with Q extensible to a Lipschitz continuous convex function defined on the convex hull conv $\mathcal{X}$. This is true if $\mathcal{X}$ is the finite set of extreme points of a polytope, e.g., $\{0,1\}^d$. The above discussion provides an alternative explanation of the exactness of the Lagrangian cuts in SDDiP [38] assuming relatively complete recourse.

3 Nested decomposition and dual dynamic programming algorithms

In this section, we introduce a nested decomposition algorithm for general scenario trees, and two dual dynamic programming algorithms for stagewise independent scenario trees. Since the size of the scenario tree could be large, we focus our attention to finding an ε-optimal root node solution x_r^* (see definition (5),) rather than an optimal solution $\{x_n^*\}_{n\in\mathcal{T}}$ for the entire tree.

3.1 Subproblem oracles

Before we propose the new algorithms, we first define subproblem oracles, which we will use to describe the algorithms and conduct complexity analysis. A subproblem oracle is an oracle that takes subproblem information together with the current algorithm information to produce a solution to the subproblem. With subproblem oracles, we can describe the algorithms consistently regardless of convexity.

We assume three subproblem oracles in this paper, corresponding to the forward steps and backward steps of non-root nodes, and the root node step in the algorithms. For non-root nodes, we assume the following two subproblem oracles.

Definition 2 (*Forward Step Subproblem Oracle for Non-Root Nodes*) Consider the following subproblem for a non-root node n,

$$\min_{\substack{(x,y)\in\mathcal{F}_n,\\z\in\mathcal{X}_{a(n)}}}\left\{f_n(z,y,x)+\sigma_n\psi_n(x_{a(n)}-z)+\Theta_n(x)\right\}.\tag{F}$$

where the parent node's state variable $x_{a(n)} \in \mathcal{X}_{a(n)}$ is a given parameter and Θ_n : $\mathcal{X}_n \to \bar{\mathbb{R}}$ is a l.s.c. function, representing an under-approximation of the expected cost-to-go function $Q_n(x)$ defined in (3). The forward step subproblem oracle finds an optimal solution of (F) given $x_{a(n)}$ and Θ_n, that is, we denote this oracle as a mapping $\mathcal{O}_n^{\mathrm{F}}$ that takes $(x_{a(n)}, \Theta_n)$ as input and outputs an optimal solution (x_n, y_n, z_n) of (F) for $n \neq r$.

Recall that the values σ_n for all $n \in \mathcal{N}$ in (F) are the chosen penalty parameters that satisfy Assumption 2. In view of Propositions 3 and 4, we set $l_{n,\lambda} \geq \sigma_n$ and $l_{n,\rho} = 0$ for the convex case with $\psi_n = \|\cdot\|$; or $l_{n,\rho} \geq \sigma_n$ otherwise for the dual variable set $\mathcal{U}_n := \{(\lambda, \rho) : \|\lambda\|_* \leq l_{n,\lambda}, 0 \leq \rho \leq l_{n,\rho}\}$ in the next definition.

Definition 3 (*Backward Step Subproblem Oracles for Non-Root Nodes*) Consider the following subproblem for a non-root node n,

$$\max_{\substack{(\lambda,\rho)\in\mathcal{U}_n \\ }} \min_{\substack{(x,y)\in\mathcal{F}_n, \\ z\in\mathcal{X}_{a(n)}}} \left\{ f_n(z, y, x) + \langle \lambda, x_{a(n)} - z \rangle + \rho \psi_n(x_{a(n)} - z) + \Theta_n(x) \right\}, \quad \text{(B)}$$

where the parent node's state variable $x_{a(n)} \in \mathcal{X}_{a(n)}$ is a given parameter and Θ_n : $\mathcal{X}_n \to \bar{\mathbb{R}}$ is a l.s.c. function, representing an under-approximation of the expected cost-to-go function. The backward step subproblem oracle finds an optimal solution of (B) for the given $x_{a(n)}$ and Θ_n. Similarly, we denote this oracle as a mapping $\mathcal{O}_n^{\mathrm{B}}$ that takes $(x_{a(n)}, \Theta_n)$ as input and outputs an optimal solution $(x_n, y_n, z_n; \lambda_n, \rho_n)$ of (B) for $n \neq r$.

For the root node, we assume the following subproblem oracle.

Definition 4 (*Subproblem Oracle for the Root Node*) Consider the following subproblem for the root node $r \in \mathcal{N}$,

$$\min_{(x,y)\in\mathcal{F}_r} \left\{ f_r(x_{a(r)}, y, x) + \Theta_r(x) \right\}, \quad \text{(R)}$$

where $\Theta_r : \mathcal{X}_r \to \bar{\mathbb{R}}$ is a l.s.c. function, representing an under-approximation of the expected cost-to-go function. The subproblem oracle for the root node is denoted as $\mathcal{O}_r$ that takes Θ_r as input and outputs an optimal solution (x_r, y_r) of (R) for the given function Θ_r.

These subproblem oracles $\mathcal{O}_n^{\mathrm{F}}$, $\mathcal{O}_n^{\mathrm{B}}$, including the parameters σ_n, $l_{n,\lambda}$, and $l_{n,\rho}$ for all $n \neq r$, and $\mathcal{O}_r$ will be given as inputs to the algorithms. They may return any optimal solution to the corresponding nodal subproblem. For numerical implementation, they are usually handled by subroutines or external solvers.

3.2 Under- and over-approximations of cost-to-go functions

We first show how to iteratively construct under-approximation of expected cost-to-go functions using the generalized conjugacy cuts developed in Sect. 2.4. The under-approximation serves as a surrogate of the true cost-to-go function in the algorithm.

Let $i \in \mathbb{N}$ be the iteration index of an algorithm. Assume $(x_n^i, y_n^i)_{n \in \mathcal{N}}$ are feasible solutions to the regularized nodal problem (4) in the i-th iteration. Then the under-approximation of the expected cost-to-go function is defined recursively from leaf nodes to the root node, and inductively for $i \in \mathbb{N}$ as

$$\underline{Q}_n^i(x) := \max \left\{ \underline{Q}_n^{i-1}(x), \sum_{m \in \mathcal{C}(n)} p_{nm} C_m^i(x \mid \hat{\lambda}_m^i, \hat{\rho}_m^i, \underline{v}_m^i) \right\}, \quad \forall x \in \mathcal{X}_n, \quad (14)$$

where $\underline{Q}_n^0 \equiv 0$ on $\mathcal{X}_n$. In the definition (14), C_m^i is the generalized conjugacy cut for Q_m at i-th iteration and $\Phi_m^{x_n^i}(x, \lambda, \rho) = -\langle \lambda, x_n^i - x \rangle - \rho \psi_n(x_n^i - x)$ (cf. (11)–(12)), that is,

$$C_m^i(x \mid \hat{\lambda}_m^i, \hat{\rho}_m^i, \underline{v}_m^i) := -\langle \hat{\lambda}_m^i, x_n^i - x \rangle - \hat{\rho}_m^i \psi_m(x_n^i - x) + \underline{v}_m^i, \quad (15)$$

where $(\hat{x}_m^i, \hat{y}_m^i \hat{z}_m^i; \hat{\lambda}_m^i, \hat{\rho}_m^i) = \mathcal{O}_m^B(x_n^i, \underline{Q}_m^i)$, and $\underline{v}_m^i$ satisfies

$$\underline{v}_m^i = f_m(\hat{z}_m^i, \hat{y}_m^i, \hat{x}_m^i) + \langle \hat{\lambda}_m^i, x_n^i - \hat{z}_m^i \rangle + \hat{\rho}_m^i \psi_m(x_n^i - \hat{z}_m^i) + \underline{Q}_m^i(\hat{x}_m^i). \quad (16)$$

The next proposition shows that $\underline{Q}_n^i$ is indeed an under-approximation of Q_n, the proof of which is given in Sect. A.2.1.

Proposition 5 *For any $n \in \mathcal{N}$, and $i \in \mathbb{N}$, $\underline{Q}_n^i(x)$ is $(\sum_{m \in \mathcal{C}(n)} p_{nm}(l_{m,\lambda} + l_{m,\rho}))$-Lipschitz continuous and*

$$Q_n(x) \geq \underline{Q}_n^i(x), \quad \forall x \in \mathcal{X}_n.$$

Now, we propose the following over-approximation of the regularized expected cost-to-go functions, which is used in the sampling and termination of the proposed nested decomposition and dual dynamic programming algorithms. For $i \in \mathbb{N}$, at root node r, let $(x_r^i, y_r^i) = \mathcal{O}_r(\underline{Q}_r^{i-1})$, and, at each non-root node n, let $(x_n^i, y_n^i, z_n^i) = \mathcal{O}_n^F(x_{a(n)}^i, \underline{Q}_n^{i-1})$. Then the over-approximation of the regularized expected cost-to-go function is defined recursively, from leaf nodes to the child nodes of the root node, and inductively for $i \in \mathbb{N}$ by

$$\overline{Q}_n^i(x) := \begin{cases} \text{conv} \left\{ \overline{Q}_n^{i-1}(x), \sum_{m \in \mathcal{C}(n)} p_{nm} \left(\bar{v}_m^i + \sigma_m \|x - x_n^i\| \right) \right\}, & \text{if (4) is convex} \\[4mm] \min \left\{ \overline{Q}_n^{i-1}(x), \sum_{m \in \mathcal{C}(n)} p_{nm} \left(\bar{v}_m^i + \sigma_m \|x - x_n^i\| \right) \right\}, & \text{otherwise} \end{cases} \quad (17)$$

where $\overline{Q}_n^0 \equiv +\infty$ for any non-leaf node $n \in \mathcal{N}$, $\overline{Q}_n^i \equiv 0$ for any iteration $i \in \mathbb{N}$ and any leaf node n, and $\bar{v}_m^i$ satisfies

$$\bar{v}_m^i = f_m(z_m^i, y_m^i, x_m^i) + \sigma_m \psi_m(x_n^i - z_m^i) + \overline{Q}_m^i(x_m^i). \tag{18}$$

Here, the operation conv$\{f, g\}$ forms the convex hull of the union of the epigraphs of any continuous functions f and g defined on the space $\mathbb{R}^d$. More precisely using convex conjugacy, we define

$$\text{conv}\{f, g\}(x) := (\min\{f(x), g(x)\})^{**}$$
$$= \sup_{\lambda \in \mathbb{R}^d} \inf_{z \in \mathbb{R}^d} \{\min\{f(z), g(z)\} + \langle \lambda, x - z \rangle\}. \tag{19}$$

The key idea behind the upper bound function (17) is to exploit the Lipschitz continuity of the regularized value function $Q_m^R(x)$. In particular, it would follow from induction that $\bar{v}_m^i$ is an upper bound on $Q_m^R(x_n^i)$, and then, by the σ_m-Lipschitz continuity of $Q_m^R(x)$, we have $\bar{v}_m^i + \sigma_m\|x - x_n^i\| \geq Q_m^R(x_n^i) + \sigma_m\|x - x_n^i\| \geq Q_m^R(x)$ for all $x \in X_n$. The next proposition summarizes this property, with the proof given in Sect. A.2.2.

Proposition 6 *For any non-root node $n \in \mathcal{N}$ and $i \geq 1$, $\overline{Q}_n^i(x)$ is $(\sum_{m \in \mathcal{C}(n)} p_{nm}\sigma_m)$-Lipschitz continuous. Moreover, we have $\bar{v}_m^i \geq Q_m^R(x_n^i)$ for any node $m \in \mathcal{C}(n)$ and thus*

$$\overline{Q}_n^i(x) \geq Q_n^R(x), \quad \forall x \in \mathcal{X}_n.$$

3.3 A nested decomposition algorithm for general trees

We first propose a nested decomposition algorithm in Algorithm 1 for a general scenario tree. In each iteration i, Algorithm 1 carries out a forward step, a backward step, and a root node update step. In the forward step, the algorithm proceeds from $t = 1$ to T by solving all the nodal subproblems with the current under-approximation of their cost-to-go functions in stage t. After all the state variables x_n^i are obtained for nodes $n \in \mathcal{N}$, the backward step goes from $t = T$ back to 1. At each node n in stage t, it first updates the under-approximation of the expected cost-to-go function. Next it solves the dual problem to obtain an optimal primal-dual solution pair $(\hat{x}_n^i, \hat{y}_n^i, \hat{z}_n^i; \hat{\lambda}_n^i, \hat{\rho}_n^i)$, which is used to construct a generalized conjugacy cut using (15), together with values $\underline{v}_n^i$ and $\bar{v}_n^i$ calculated with (16) and (18). Finally the algorithm updates the root node solution using the updated under-approximation of the cost-to-go function, and determines the new lower and upper bounds. The incumbent solution (x_r^*, y_r^*) may also be updated as the algorithm output at termination, although it is not used in the later iterations.

Algorithm 1 solves the regularized problem (4) for an ε-optimal root node solution. To justify the ε-optimality of the output of the algorithm, we have the following proposition, the proof of which is given in Sect. A.2.3.

Proposition 7 *Given any $\varepsilon > 0$, if UPPERBOUND $-$ LOWERBOUND $\leq \varepsilon$, then the returned solution (x_r^*, y_r^*) is an ε-optimal root node solution to the regularized problem (4). In particular, if $\overline{Q}_r^i(x_r^{i+1}) - \underline{Q}_r^i(x_r^{i+1}) \leq \varepsilon$ for some iteration index i, then*

Algorithm 1 A nested decomposition algorithm for a general tree

---

Require: scenario tree $\mathcal{T} = (\mathcal{N}, \mathcal{E})$ with subproblem oracles $\mathcal{O}_r, \mathcal{O}_n^{\mathrm{F}}, \mathcal{O}_n^{\mathrm{B}}, n \neq r$, and $\varepsilon > 0$
Ensure: an ε-optimal root node solution (x_r^*, y_r^*) to the regularized problem (4)

1: Initialize: $i \leftarrow 1; \underline{\mathcal{Q}}_n^0 \leftarrow 0, \overline{\mathcal{Q}}_n^0 \leftarrow +\infty \, \forall n : \mathcal{C}(n) \neq \varnothing$ and $\overline{\mathcal{Q}}_n^0 \leftarrow 0 \, \forall n : \mathcal{C}(n) = \varnothing$
2: Evaluate $(x_r^1, y_r^1) = \mathcal{O}_r(0)$
3: Set LOWERBOUND $\leftarrow f_r(x_{a(r)}, y_r^1, x_r^1)$, UPPERBOUND $\leftarrow +\infty$
4: **while** UPPERBOUND $-$ LOWERBOUND $> \varepsilon$ **do**
5:     **for** $t = 1, \ldots, T - 1$ **do** $\triangleright$ i-th forward step
6:         **for** $n \in \mathcal{N}(t)$ **do**
7:             Evaluate $(x_n^i, y_n^i, z_n^i) = \mathcal{O}_n^{\mathrm{F}}(x_{a(n)}^i, \underline{\mathcal{Q}}_n^{i-1})$
8:         **end for**
9:     **end for**
10:     **for** $t = T, \ldots, 1$ **do** $\triangleright$ i-th backward step
11:         **for** $n \in \mathcal{N}(t)$ **do**
12:             Update $\underline{\mathcal{Q}}_n^i$ and $\overline{\mathcal{Q}}_n^i$ using (14) and (17)
13:             Evaluate $(\hat{x}_n^i, \hat{y}_n^i, \hat{z}_n^i; \hat{\lambda}_n^i, \hat{\rho}_n^i) = \mathcal{O}_n^{\mathrm{B}}(x_{a(n)}^i, \mathcal{Q}_n^i)$
14:             Calculate $C_n^i, \underline{v}_n^i$, and $\bar{v}_n^i$ using (15), (16), and (18)
15:         **end for**
16:     **end for**
17:     Update $\underline{\mathcal{Q}}_r^i$ and $\overline{\mathcal{Q}}_r^i$ using (14) and (17) $\triangleright$ root node update
18:     Evaluate $(x_r^{i+1}, y_r^{i+1}) = \mathcal{O}_r(\underline{\mathcal{Q}}_r^i)$
19:     Update LOWERBOUND $\leftarrow f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \underline{\mathcal{Q}}_r^i(x_r^{i+1})$
20:     **if** UPPERBOUND $> f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \overline{\mathcal{Q}}_r^i(x_r^{i+1})$ **then**
21:         Update UPPERBOUND $\leftarrow f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \overline{\mathcal{Q}}_r^i(x_r^{i+1})$
22:         Set $(x_r^*, y_r^*) = (x_r^{i+1}, y_r^{i+1})$
23:     **end if**
24:     $i \leftarrow i + 1$
25: **end while**

UPPERBOUND $-$ LOWERBOUND $\leq \varepsilon$ *and Algorithm* 1 *terminates after the* i-*th iteration.*

3.4 A deterministic sampling dual dynamic programming algorithm

Starting from this subsection, we turn our attention to stagewise independent stochastic problems, which is defined in the following assumption.

Assumption 3 For any $t = 1, \ldots, T - 1$ and any $n, n' \in \mathcal{N}(t)$, the state space, the transition probabilities, as well as the data associated with the child nodes $\mathcal{C}(n)$ and $\mathcal{C}(n')$ are identical. In particular, this implies $\mathcal{Q}_n(x) = \mathcal{Q}_{n'}(x) =: \mathcal{Q}_t(x)$ for all $x \in \mathcal{X}_n = \mathcal{X}_{n'} =: \mathcal{X}_t \subseteq \mathbb{R}^{d_t}$.

We denote $n \sim n'$ for $n, n' \in \mathcal{N}(t)$ for some $t = 1, \ldots, T - 1$, if the nodes n, n' are defined by identical data. We then use $\tilde{\mathcal{N}}(t) := \mathcal{N}(t)/\sim$ to denote the set of nodes with size $N_t := |\tilde{\mathcal{N}}(t)|$ that are defined by distinct data in stage t for all $t = 1, \ldots, T - 1$, i.e. $\tilde{\mathcal{N}} := \cup_{t=0}^{T} \tilde{\mathcal{N}}(t)$ forms a recombining scenario tree [39]. For each node $m \in \tilde{\mathcal{N}}(t)$, we denote $p_{t-1,m} := p_{nm}$ for any $n \in \tilde{\mathcal{N}}(t-1)$ since $p_{n,m} = p_{n',m}$ for any $n, n' \in \tilde{\mathcal{N}}(t-1)$. Due to stagewise independence, it suffices

to keep track of the state of each stage in the algorithm, instead of the state of each node. To be consistent, we also denote the root node solution as (x_0^i, y_0^i) for $i \in \mathbb{N}$. We present the algorithm in Algorithm 2.

Similar to Algorithm 1, each iteration in Algorithm 2 consists of a forward step, a backward step, and a root node update step. In particular, at a node $n \in \tilde{\mathcal{N}}(t)$ with $t < T$, the forward step proceeds to a child node $m \in \tilde{\mathcal{N}}(t+1)$, where the *approximation gap* $\gamma_m^i := \overline{\mathcal{Q}}_t^{i-1}(x_m) - \underline{\mathcal{Q}}_t^{i-1}(x_m)$ is among the largest of all the approximation gaps of states $x_{m'}^i$ of nodes $m' \in \tilde{\mathcal{N}}(t+1)$. Then the state variable of node m is considered the state variable of stage $t(m)$ in the iteration i. Due to stagewise independence, the backward step at each stage t only need to generate cuts for the nodes in the recombining tree $\tilde{\mathcal{N}}$. The optimality of the returned solution (x_0^*, y_0^*) is guaranteed by Proposition 7.

Algorithm 2 Deterministic sampling dual dynamic programming algorithm

---

Require: recombining scenario tree $\tilde{\mathcal{N}}$ with subproblem oracles $\mathcal{O}_r, \mathcal{O}_n^{\text{F}}, \mathcal{O}_n^{\text{B}}, n \neq r$, and $\varepsilon > 0$
Ensure: an ε-optimal root node solution (x_0^*, y_0^*) to the regularized problem (4)

1: Initialize: $i \leftarrow 1$; $\underline{\mathcal{Q}}_t^0 \leftarrow 0$, $\forall t$, $\overline{\mathcal{Q}}_t^0 \leftarrow +\infty \ \forall t \leq T-1$ and $\overline{\mathcal{Q}}_T^0 \leftarrow 0$
2: Evaluate $(x_0^1, y_0^1) = \mathcal{O}_r(0)$
3: Set LOWERBOUND $\leftarrow f_r(x_{a(r)}, y_0^1, x_0^1)$, UPPERBOUND $\leftarrow +\infty$
4: **while** UPPERBOUND $-$ LOWERBOUND $> \varepsilon$ **do**
5:     **for** $t = 1, \ldots, T-1$ **do**   ▷ i-th forward step
6:         **for** $n \in \tilde{\mathcal{N}}(t)$ **do**
7:             Evaluate $(x_n^i, y_n^i, z_n^i) = \mathcal{O}_n^{\text{F}}(x_{t-1}^i, \underline{\mathcal{Q}}_t^{i-1})$
8:             Calculate the gap $\gamma_n^i = \overline{\mathcal{Q}}_t^{i-1}(x_n^i) - \underline{\mathcal{Q}}_t^{i-1}(x_n^i)$
9:         **end for**
10:         Select any $n^*(t) \in \{n \in \mathcal{N}(t) : \gamma_n^i \geq \gamma_{n'}^i, \ \forall n' \in \tilde{\mathcal{N}}(t)\}$, and let $x_t^i \leftarrow x_{n^*(t)}^i$
11:     **end for**
12:     **for** $t = T, \ldots, 1$ **do**   ▷ i-th backward step
13:         Update $\underline{\mathcal{Q}}_t^i$ and $\overline{\mathcal{Q}}_t^i$ using (14) and (17)
14:         **for** $n \in \tilde{\mathcal{N}}(t)$ **do**
15:             Evaluate $(\hat{x}_n^i, \hat{y}_n^i, \hat{z}_n^i; \hat{\lambda}_n^i, \hat{\rho}_n^i) = \mathcal{O}_n^{\text{B}}(x_{t-1}^i, \underline{\mathcal{Q}}_t^i)$
16:             Calculate $C_n^i, \underline{v}_n^i, \bar{v}_n^i$ using (15), (16), and (18)
17:         **end for**
18:     **end for**
19:     Update $\underline{\mathcal{Q}}_0^i$ and $\overline{\mathcal{Q}}_0^i$ using (14) and (17)   ▷ root node update
20:     Evaluate $(x_0^{i+1}, y_0^{i+1}) = \mathcal{O}_r(\underline{\mathcal{Q}}_0^i)$
21:     Update LOWERBOUND $\leftarrow f_r(x_{a(r)}, y_0^{i+1}, x_0^{i+1}) + \underline{\mathcal{Q}}_0^i(x_0^{i+1})$
22:     **if** UPPERBOUND $> f_r(x_{a(r)}, y_0^{i+1}, x_0^{i+1}) + \overline{\mathcal{Q}}_0^i(x_0^{i+1})$ **then**
23:         Update UPPERBOUND $\leftarrow f_r(x_{a(r)}, y_0^{i+1}, x_0^{i+1}) + \overline{\mathcal{Q}}_0^i(x_0^{i+1})$
24:         Set $(x_0^*, y_0^*) = (x_0^{i+1}, y_0^{i+1})$
25:     **end if**
26:     $i \leftarrow i+1$
27: **end while**

3.5 A stochastic sampling dual dynamic programming algorithm

Now we present a stochastic dual dynamic programming algorithm, which uses stochastic sampling rather than deterministic sampling. So, instead of traversing the scenario tree and finding a path with the largest approximation gap, the stochastic sampling algorithm generates M scenario paths before an iteration begins for some $M \geq 1$. To be precise, we introduce the following notations. Let $\mathcal{P} = \prod_{t=1}^{T} \tilde{\mathcal{N}}(t)$ denote all possible scenario paths from stage 1 to stage T. A scenario path is denoted as a T-element sequence $P = (n_1, \ldots, n_T) \in \mathcal{P}$, where $n_t \in \tilde{\mathcal{N}}(t)$ for each $t = 1, \ldots, T$. In the i-th iteration, we sample M independent scenario paths $\mathscr{P}^i = \{P^{i,1}, \ldots, P^{i,M}\}$, and we use $P_t^{i,j}$ to denote the t-th node in the scenario path $P^{i,j}$, i.e., the node in the t-th stage of the j-th scenario path in the i-th iteration, for $1 \leq j \leq M$ and $1 \leq t \leq T$. Since in each iteration, the solutions and the approximations depend on the scenario path $P^{i,j}$, we use two superscripts i and j for solutions and cuts, where a single superscript i is used in the deterministic sampling algorithm. In addition, for every node $n \in \tilde{\mathcal{N}}(t)$ for some stage t, the under-approximation of the expected cost-to-go function is updated over all scenario path index $j = 1, \ldots, M$, with M cuts in total, i.e.,

$$\underline{Q}_t^i(x) := \max \left\{ \underline{Q}_t^{i-1}(x), \sum_{m \in \tilde{\mathcal{N}}(t+1)} p_{tm} C_m^{i,j}(x \mid \hat{\lambda}_m^{i,j}, \hat{\rho}_m^{i,j}, \underline{v}_m^{i,j}), \ 1 \leq j \leq M \right\}, \quad (20)$$

where $C_m^{i,j}$ is the generalized conjugacy cut generated with $(\hat{x}_m^{i,j}, \hat{y}_m^{i,j}, \hat{z}_m^{i,j}; \hat{\lambda}_m^{i,j}, \hat{\rho}_m^{i,j})$ $= \mathcal{O}_m^{B}(x_n^{i,j}, \underline{Q}_{t+1}^i)$ using formula (15). With these notations, the algorithm is displayed in Algorithm 3.

Unlike the preceding two algorithms, Algorithm 3 does not need to construct the over-approximation of the regularized value functions for selecting the child node to proceed with. Instead, it determines the scenario paths before the forward step starts. In the forward step, each nodal problem in the sampled scenario path is solved. Then in the backward step, the dual problems are solved at the nodes that are defined by distinct data, dependent on the parent node's state variable obtained in the forward step. The termination criterion is flexible. In the existing literature [33, 38], statistical upper bounds based on the sampled scenario paths are often used together with the lower bound for terminating the algorithm. In particular for the convex problems, if we set $\sigma_n = +\infty$, which implies $l_{n,\lambda} = +\infty$ in the backward step subproblem oracles $\mathcal{O}_n^B$ for all $n \in \mathcal{N}$, then Algorithm 3 reduces to the usual SDDP algorithm in the literature [15, 33].

4 Upper bounds on iteration complexity of proposed algorithms

In this section, we derive upper bounds on the iteration complexity of the three proposed algorithms, i.e. the bound on the iteration index when the algorithm terminates. These upper bounds on the iteration complexity imply convergence of the algorithm to an ε-optimal root node solution for any $\varepsilon > 0$.

Algorithm 3 Stochastic sampling dual dynamic programming algorithm

---

Require: recombining scenario tree $\tilde{\mathcal{N}}$ with subproblem oracles $\mathscr{O}_r, \mathscr{O}_n^{\mathrm{F}}, \mathscr{O}_n^{\mathrm{B}}, n \neq r$
1: Initialize: $i \leftarrow 1; \underline{\mathcal{Q}}_t^0 \leftarrow 0, \forall t$
2: Evaluate $(x_0^1, y_0^1) = \mathscr{O}_r(0)$
3: **while** some stopping criterion is not satisfied **do**
4:     Sample M scenario paths $\mathscr{P}^i = \{P^{i,1}, \dots, P^{i,M}\}$
5:     **for** $j = 1, \dots, M$ **do**   ▷ i-th forward step
6:         **for** $t = 1, \dots, T - 1$ **do**
7:             Evaluate $(x_t^{i,j}, y_t^{i,j}, z_t^{i,j}) = \mathscr{O}_n^{\mathrm{F}}(x_{t-1}^{i,j}, \underline{\mathcal{Q}}_t^{i-1})$
8:         **end for**
9:     **end for**
10:    **for** $t = T, \dots, 1$ **do**   ▷ i-th backward step
11:       Update $\underline{\mathcal{Q}}_t^i$ using (20)
12:       **for** $j = 1, \dots, M$ **do**
13:          **for** $n \in \mathcal{N}(t)/ \sim$ **do**
14:             Evaluate $(\hat{x}_n^{i,j}, \hat{y}_n^{i,j}, \hat{z}_n^{i,j}; \hat{\lambda}_n^{i,j}, \hat{\rho}_n^{i,j}) = \mathscr{O}_n^{\mathrm{B}}(x_{t-1}^{i,j}, \underline{\mathcal{Q}}_t^i)$
15:             Calculate $C_n^{i,j}$ and $\underline{v}_n^{i,j}$ using (15) and (16)
16:          **end for**
17:       **end for**
18:    **end for**
19:    Update $\underline{\mathcal{Q}}_0^i$ using (20)   ▷ root node update
20:    Evaluate $\mathscr{O}_r(\underline{\mathcal{Q}}_0^i) = (x_0^{i+1}, y_0^{i+1})$
21:    $i \leftarrow i + 1$
22: **end while**

4.1 Upper bound analysis on iteration complexity of algorithm 1

In this section, we discuss the iteration complexity of Algorithm 1. We begin with the definition of a set of parameters used in the convergence analysis. Let ε denote the desired root-node optimality gap ε in Algorithm 1. Let $\delta = (\delta_n)_{n \in \mathcal{N}, \mathcal{C}(n) \neq \varnothing}$ be a set of positive numbers such that $\varepsilon = \sum_{n \in \mathcal{N}, \mathcal{C}(n) \neq \varnothing} p_n \delta_n$. Since $\varepsilon > 0$, such δ_n's clearly exist. Then, we define recursively for each non-leaf node n

$$\gamma_n(\delta) := \delta_n + \sum_{m \in \mathcal{C}(n)} p_{nm} \gamma_m(\delta), \tag{21}$$

and $\gamma_n(\delta) = 0$ for leaf nodes n. For $i \in \mathbb{N}$, recall the approximation gap $\gamma_n^i = \overline{\mathcal{Q}}_n^{i-1}(x_n^i) - \underline{\mathcal{Q}}_n^{i-1}(x_n^i)$ for $n \in \mathcal{N}$. For leaf nodes, $\gamma_n^i \equiv 0$ by definition for all $i \in \mathbb{N}$. In addition, we define the sets of indices $\mathcal{I}_n(\delta)$ for each $n \in \mathcal{N}$ as

$$\mathcal{I}_n(\delta) := \left\{ i \in \mathbb{N} : \gamma_n^i > \gamma_n(\delta) \text{ and } \gamma_m^i \leq \gamma_m(\delta), \forall m \in \mathcal{C}(n) \right\}. \tag{22}$$

Intuitively, the index set $\mathcal{I}_n(\delta)$ consists of the iteration indices when all the child nodes of n have good approximations of the expected cost-to-go function at the forward step solution, while the node n itself does not. The next lemma shows that the backward step for node n in the iteration $i \in \mathcal{I}_n(\delta)$ will reduce the expected cost-to-go function approximation gap at node n to be no more than $\gamma_n(\delta)$.

Lemma 3 *If an iteration index* $i \in \mathcal{I}_n(\delta)$, *i.e.,* $\overline{\mathcal{Q}}_n^{i-1}(x_n^i) - \underline{\mathcal{Q}}_n^{i-1}(x_n^i) > \gamma_n(\delta)$ *and* $\overline{\mathcal{Q}}_m^{i-1}(x_m^i) - \underline{\mathcal{Q}}_m^{i-1}(x_m^i) \leq \gamma_m(\delta)$ *for all* $m \in \mathcal{C}(n)$, *then*

$$\overline{\mathcal{Q}}_n^i(x) - \underline{\mathcal{Q}}_n^i(x) \leq \gamma_n(\delta), \quad \forall x \in \mathcal{X}_n, \|x - x_n^i\| \leq \frac{\delta_n}{2L_n}, \tag{23}$$

where $L_n := \sum_{m \in \mathcal{C}(n)} p_{nm}(l_{m,\lambda} + l_{m,\rho})$ *is determined by the input parameters.*

The proof is given in Sect. A.3.1. Lemma 3 shows that an iteration being in the index set would imply an improvement of the approximation in a neighborhood of the current state. In other words, each $i \in \mathcal{I}_n$ would carve out a ball of radius $\delta_n/(2L_n)$ in the state space $\mathcal{X}_n$ such that no point in the ball can be the forward step solution of some iteration i in $\mathcal{I}_n$. This implies that we could bound the cardinality $|\mathcal{I}_n|$ of $\mathcal{I}_n$ by the size and shape of the corresponding state space $\mathcal{X}_n$.

Lemma 4 *Let* $\mathscr{B} = \{\mathcal{B}_{n,k} \subset \mathbb{R}^{d_n}\}_{1 \leq k \leq K_n, n \in \mathcal{N}}$ *be a collection of balls, each with diameter* $D_{n,k} \geq 0$, *such that* $\mathcal{X}_n \subseteq \bigcup_{k=1}^{K_n} \mathcal{B}_{n,k}$. *Then,*

$$|\mathcal{I}_n(\delta)| \leq \sum_{k=1}^{K_n} \left(1 + \frac{2L_n D_{n,k}}{\delta_n}\right)^{d_n}.$$

The proof is by a volume argument of the covering balls with details given in Sect. A.3.2. We have an upper bound on the iteration complexity of Algorithm 1.

Theorem 1 *Given* $\varepsilon > 0$, *choose values* $\delta = (\delta_n)_{n \in \mathcal{N}, \mathcal{C}(n) \neq \varnothing}$ *such that* $\delta_n > 0$ *and* $\sum_{n \in \mathcal{N}, \mathcal{C}(n) \neq \varnothing} p_n \delta_n = \varepsilon$. *Let* $\mathscr{B} = \{\mathcal{B}_{n,k}\}_{1 \leq k \leq K_n, n \in \mathcal{N}}$ *be a collection of balls, each with diameter* $D_{n,k} \geq 0$, *such that* $\mathcal{X}_n \subseteq \bigcup_{k=1}^{K_n} \mathcal{B}_{n,k}$ *for* $n \in \mathcal{N}$. *If Algorithm* 1 *terminates with an* ε-*optimal root node solution* (x_r^*, y_r^*) *at the end of* i-*th iteration, then*

$$i \leq \sum_{\substack{n \in \mathcal{N}, \\ \mathcal{C}(n) \neq \varnothing}} \sum_{k=1}^{K_n} \left(1 + \frac{2L_n D_{n,k}}{\delta_n}\right)^{d_n}.$$

Proof After the i-th iteration, at least one of the following two situations must happen:

i. At the root node, it holds that $\overline{\mathcal{Q}}_r^i(x_r^{i+1}) - \underline{\mathcal{Q}}_r^i(x_r^{i+1}) \leq \gamma_r(\delta)$, where γ_r is defined in (21).

ii. There exists a node $n \in \mathcal{N}$ such that $\overline{\mathcal{Q}}_n^i(x_n^{i+1}) - \underline{\mathcal{Q}}_n^i(x_n^{i+1}) > \gamma_n(\delta)$, but all of its child nodes satisfy $\overline{\mathcal{Q}}_m^i(x_m^{i+1}) - \underline{\mathcal{Q}}_m^i(x_m^{i+1}) \leq \gamma_m(\delta), \forall m \in \mathcal{C}(n)$. In other words, $i + 1 \in \mathcal{I}_n(\delta)$.

Note that $\gamma_r(\delta) = \delta_r + \sum_{m \in \mathcal{C}(r)} p_{rm} \gamma_m(\delta) = \cdots = \sum_{n \in \mathcal{N}, \mathcal{C}(n) \neq \varnothing} p_n \delta_n$. If case i happens, then by Proposition 7, (x_r^{i+1}, y_r^{i+1}) is an ε-optimal root node solution. Note that case ii can only happen at most $\sum_{n \in \mathcal{N}} |\mathcal{I}_n(\delta)|$ times by Lemma 4. Therefore, we have that

$$i \leq \sum_{\substack{n \in \mathcal{N} \\ \mathcal{C}(n) \neq \varnothing}} \sum_{k=1}^{K_n} \left(1 + \frac{2 L_n D_{n,k}}{\delta_n} \right)^{d_n},$$

when the algorithm terminates. $\qquad\square$

Theorem 1 implies the ε-convergence of the algorithm for any $\varepsilon > 0$. We remark that the form of the upper bound depends on the values δ and the covering balls $\mathcal{B}_{n,k}$, and therefore the right-hand-side can be tightened to the infimum over all possible choices. While it may be difficult to find the best bound in general, in the next section we take some specific choices of δ and $\mathcal{B}$ and simplify the complexity upper bound, based on the stagewise independence assumption.

4.2 Upper bound analysis on iteration complexity of algorithm 2

Before giving the iteration complexity bound for Algorithm 2, we slightly adapt the notations in the previous section to the stagewise independent scenario tree. We take the values $\delta = (\delta_n)_{n \in \tilde{\mathcal{N}}, \mathcal{C}(n) \neq \varnothing}$ such that $\delta_n = \delta_{n'}$ for all $n, n' \in \tilde{\mathcal{N}}(t)$ for some $t = 1, \ldots, T$. Thus we denote $\delta_t = \delta_n$ for any $n \in \tilde{\mathcal{N}}(t)$, and $\delta_0 = \delta_r$. The vector of $\gamma_t(\delta)$ is defined recursively for non-leaf nodes as

$$\gamma_t(\delta) := \gamma_{t+1}(\delta) + \delta_t, \quad \text{if } t \leq T - 1, \tag{24}$$

and $\gamma_T(\delta) = 0$. Let $\gamma_t^i := \overline{\mathcal{Q}}_t^{i-1}(x_t^i) - \underline{\mathcal{Q}}_t^{i-1}(x_t^i)$ and recall that $\gamma_0^i := \gamma_r^i$ for each index i. The sets of indices $\mathcal{I}_t(\delta)$ are defined for $t = 0, \ldots, T - 1$ as

$$\mathcal{I}_t(\delta) := \left\{ i \in \mathbb{N} : \gamma_t^i > \gamma_t(\delta) \text{ and } \gamma_{t+1}^i \leq \gamma_{t+1}(\delta) \right\}. \tag{25}$$

Note that $\gamma_t^i = \max_{n \in \tilde{\mathcal{N}}(t)} \gamma_n^i$ (line 10 in Algorithm 2). By Lemma 3, an iteration $i \in \mathcal{I}_t(\delta)$ implies $\overline{\mathcal{Q}}_t^i(x) - \underline{\mathcal{Q}}_t^i(x) \leq \gamma_t(\delta)$ for all $x \in \mathcal{X}_n$ with $\|x - x_t^i\| \leq \delta_t/(2L_t)$, where $L_t = L_n$ for any $n \in \tilde{\mathcal{N}}(t)$. Moreover, since $\mathcal{X}_n = \mathcal{X}_t$ for $n \in \tilde{\mathcal{N}}(t)$, for any covering balls $\mathcal{B}_{t,k} \subset \mathbb{R}^{d_t}$ with diameters $D_{t,k} \geq 0$, such that $\mathcal{X}_t \subseteq \cup_{k=1}^{K_t} \mathcal{B}_{t,k}$, by the same argument of Lemma 4, we know that

$$|\mathcal{I}_t(\delta)| \leq \sum_{k=1}^{K_t} \left(1 + \frac{2 L_t D_{t,k}}{\delta_t} \right)^{d_t}. \tag{26}$$

We summarize the upper bound on the iteration complexity of Algorithm 2 in the next theorem, and omit the proof since it is almost a word-for-word repetition with the notation adapted as above.

Theorem 2 *Given any $\varepsilon > 0$, choose values $\delta = (\delta_t)_{t=0}^{T-1}$ such that $\delta_t > 0$ and $\sum_{t=0}^{T-1} \delta_t = \varepsilon$. Let $\mathcal{B} = \{\mathcal{B}_{t,k} \subset \mathbb{R}^{d_t}\}_{1 \leq k \leq K_t, 0 \leq t \leq T-1}$ be a collection of balls, each*

with diameter $D_{t,k} \geq 0$, such that $\mathcal{X}_t \subseteq \bigcup_{k=1}^{K_t} \mathcal{B}_{t,k}$ for $0 \leq t \leq T - 1$. If Algorithm 2 terminates with an ε-optimal root node solution (x_0^*, y_0^*) in i iterations, then

$$i \leq \sum_{t=0}^{T-1} \sum_{k=1}^{K_t} \left(1 + \frac{2L_t D_{t,k}}{\delta_t} \right)^{d_t}.$$

We next discuss some special choices of the values δ and the covering ball collections $\mathscr{B}$. First, since $\mathcal{X}_t$ are compact, suppose $\mathcal{B}_t$ is the smallest ball containing $\mathcal{X}_t$. Then we have diam $\mathcal{X}_t \leq D_t \leq 2$diam $\mathcal{X}_t$ where $D_t = \text{diam } \mathcal{B}_t$. Moreover, suppose $L_t \leq L$ for some $L > 0$ and $d_t \leq d$ for some $d > 0$. Then by taking $\delta_t = \varepsilon/T$ for all $0 \leq t \leq T - 1$, we have the following bound.

Corollary 1 *If Algorithm 2 terminates with an ε-optimal root node solution (x_0^*, y_0^*), then the iteration index is bounded by*

$$i \leq T \left(1 + \frac{2LDT}{\varepsilon} \right)^d,$$

where L, d, D are the upper bounds for L_t, d_t, and D_t, $0 \leq t \leq T - 1$, respectively.

Proof Take $\delta_t = \varepsilon/T$ for all $0 \leq t \leq T - 1$ and apply Theorem 2. □

Note that the iteration complexity bound in Corollary 1 grows asymptotically $\mathcal{O}(T^{d+1})$ as $T \to \infty$. Naturally such bound is not satisfactory since it is nonlinear in T with possibly very high degree d. However, by changing the optimality criterion, we next derive an iteration complexity bound that grows linearly in T, while all other parameters, L, D, ε, d, are independent of T.

Corollary 2 *If Algorithm 2 terminates with a $(T\varepsilon)$-optimal root node solution (x_0^*, y_0^*), then the iteration index is bounded by*

$$i \leq T \left(1 + \frac{2LD}{\varepsilon} \right)^d,$$

where L, d, D are the upper bounds for L_t, d_t, and D_t, $0 \leq t \leq T - 1$, respectively.

Proof Take $\delta_t = \varepsilon$ for all $0 \leq t \leq T - 1$ and apply Theorem 2. □

The termination criterion in Corollary 2 corresponds to the usual relative optimality gap, if the total objective is known to grow at least linearly with T, as is the case for many practical problems. Last, we consider a special case where $\mathcal{X}_t$ are finite for all $0 \leq t \leq T - 1$.

Corollary 3 *Suppose the cardinality $|\mathcal{X}_t| \leq K < \infty$ for all $0 \leq t \leq T - 1$, for some positive integer K. In this case, if Algorithm 2 terminates with an ε-optimal root node solution (x_0^*, y_0^*), then the iteration index is bounded by*

$$i \leq TK.$$

Proof Note that when $\mathcal{X}_t$ is finite, it can be covered by degenerate balls $B_0(x)$, $x \in \mathcal{X}_t$. Thus $D_{t,k} = 0$ for $k = 1, \ldots, K_t$ and $K_t \leq K$ by assumption. Apply Theorem 2, we get $i \leq \sum_{t=0}^{T-1} \sum_{k=1}^{K_t} 1 \leq TK$.   □

The bound in Corollary 3 grows linearly in T and does not depend on the value of ε. In other words, we are able to obtain exact solutions to the regularized problem (4) assuming the subproblem oracles.

Remark 2 All the iteration complexity bounds in Theorem 2, Corollaries 1, 2 and 3 are independent of the size of the scenario tree in each stage N_t, $1 \leq t \leq T$. This can be explained by the fact that Algorithm 2 evaluates $1 + N_T + 2 \sum_{t=1}^{T-1} N_t$ times of the subproblem oracles in each iteration.

4.3 Upper bound analysis on iteration complexity of algorithm 3

In the following we study the iteration complexity of Algorithm 3. For clarity, we model the subproblem oracles $\mathcal{O}_n^F$ and $\mathcal{O}_n^B$ as random functions, that are Σ_i^{oracle}-measurable in each iteration $i \in \mathbb{N}$, for any node $n \neq r$, where $\{\Sigma_i^{\text{oracle}}\}_{i=0}^{\infty}$ is a filtration of σ-algebras in the probability space. Intuitively, this model says that the information given by Σ_i^{oracle} could be used to predict the outcome of the subproblem oracles. We now make the following assumption on the sampling step.

Assumption 4 In each iteration i, the M scenario paths are sampled uniformly with replacement, independent from each other and the outcomes of the subproblem oracles. That is, the conditional probability of the j-th sample $P^{i,j}$ taking any scenario $n_t \in \tilde{\mathcal{N}}(t)$ in stage t is almost surely

$$\text{Prob}\left(P_t^{i,j} = n_t \mid \Sigma_{\infty}^{\text{oracle}}, \sigma\{P_{t'}^{i',j'}\}_{(i',j',t') \neq (i,j,t)}\right) = \frac{1}{N_t}, \qquad (27)$$

where $\Sigma_{\infty}^{\text{oracle}} := \cup_{i=1}^{\infty} \Sigma_i^{\text{oracle}}$, and $\sigma\{P_{t'}^{i',j'}\}_{(i',j',t') \neq (i,j,t)}$ is the σ-algebra generated by scenario samples other than the j-th sample in stage t of iteration i.

In the sampling step in the i-th iteration, let $\gamma_t^{i,j} := \mathcal{Q}_t^R(x_t^{i,j}) - \underline{\mathcal{Q}}_t^{i-1}(x_t^{i,j})$ for any $t \leq T - 1$, which is well defined by Assumption 3, and let $\tilde{\gamma}_t^{i,j} := \max\{\mathcal{Q}_t^R(x_n) - \underline{\mathcal{Q}}_t^{i-1}(x_n) : (x_n, y_n, z_n) = \mathcal{O}_n^F(x_{t-1}^{i,j}, \underline{\mathcal{Q}}_n^{i-1}), n \in \tilde{\mathcal{N}}(t)\}$ for each scenario path index $1 \leq j \leq M$. Note that by definition, we have $\gamma_t^{i,j} \leq \tilde{\gamma}_t^{i,j}$ for any $t = 1, \ldots, T - 1$, everywhere in the probability space. We define the sets of indices $\mathcal{I}_t(\delta)$ for each $t = 0, \ldots, T - 1$, similar to those in the deterministic sampling case, as

$$\mathcal{I}_t(\delta) := \bigcup_{j=1}^{M} \left\{i \in \mathbb{N} : \gamma_t^{i,j} > \gamma_t(\delta) \text{ and } \tilde{\gamma}_{t+1}^{i,j} \leq \gamma_{t+1}(\delta)\right\}. \qquad (28)$$

With the same argument, we know that the upper bound (26) on the sizes of $\mathcal{I}_t(\delta)$ holds everywhere for each $t = 0, \ldots, T - 1$. However, since the nodes in the forward

steps are sampled randomly, we do not necessarily have $i \in \cup_{t=0}^{T-1} \mathcal{I}_t(\delta)$ for each iteration index $i \in \mathbb{N}$ before Algorithm 3 first finds an ε-optimal root node solution. Instead, we define an event $A_i(\delta) := \{i \in \cup_{t=0}^{T-1} \mathcal{I}_t(\delta)\} \bigcup \cup_{j=1}^{M} \{\gamma_0^{i-1,j} \leq \gamma_0(\delta) = \varepsilon\}$ for each iteration i, that means either some approximation is improved in iteration i or the algorithm has found an ε-optimal root node solution in iteration $i - 1$. The next lemma estimates the conditional probability of $A_i(\delta)$ given any oracles outcomes and samplings up to iteration i. For simplicity, we define two σ-algebras $\Sigma_i^{\text{sample}} :=$ $\sigma\{P^{i',j'}\}_{i' \leq i, j'=1,\dots,M}$ and $\Sigma_i := \sigma(\Sigma_i^{\text{oracle}}, \Sigma_i^{\text{sample}})$ for each i.

Lemma 5 *Fix any $\varepsilon = \sum_{t=0}^{T-1} \delta_t$. Then the conditional probability inequality*

$$\text{Prob}(A_i(\delta) \mid \Sigma_{i-1}) \geq \nu := 1 - (1 - 1/N)^M,$$

holds almost surely, where $N := \prod_{t=1}^{T-1} N_t$ if $T \geq 2$ and $N := 1$ otherwise.

The proof is given in Sect. A.3.3. Now we are ready to present the probabilistic complexity bound of Algorithm 3, the proof of which is given in Sect. A.3.4.

Theorem 3 *Let $I = I(\delta, \mathscr{B})$ denote the iteration complexity bound in Theorem 2, determined by the vector δ and the collection of state space covering balls $\mathscr{B}$, and ν denote the probability bound proposed in Lemma 5. Moreover, let ι be the random variable of the smallest index such that the root node solution $(x_0^{\iota+1}, y_0^{\iota+1})$ is ε-optimal in Algorithm 3. Then for any real number $\kappa > 1$, the probability*

$$\text{Prob}\left(\iota \geq 1 + \frac{\kappa I}{\nu}\right) \leq \exp\left(\frac{-I\nu(\kappa - 1)^2}{16\kappa}\right).$$

Remark 3 Theorem 3 shows that for a fixed problem (such that $I = I(\delta, \mathscr{B})$ and $N = N_1 \cdots N_{T-1}$ are fixed), given any probability threshold $q \in (0, 1)$, the number of iterations needed for Algorithm 3 to find an ε-optimal root node solution with probability greater than $1 - q$ is $\mathcal{O}(-\ln q/\nu^2)$, which does not depend on I. In particular, if we set $M = 1$, then the number of iterations needed is $\mathcal{O}(-N^2 \ln q)$, which is exponential in the number of stage T if $N_t \geq 2$ for all $t = 1, \dots, T-1$. It remains unknown to us whether there exists a complexity bound for Algorithm 3 that is polynomial in T in general.

5 Lower bounds on iteration complexity of proposed algorithms

In this section, we discuss the sharpness of the iteration complexity bound of Algorithm 2 given in Sect. 4. In particular, we are interested in the question whether it is possible that the iteration needed for Algorithm 2 to find an ε-optimal root node solution grows linearly in T when the state spaces are infinite sets. We will see that in general it is not possible, with or without the assumption of convexity. The following lemma simplifies the discussion in this section.

Lemma 6 *Suppose $f_n(z, y, x)$ is l_n-Lipschitz continuous in z for each $n \in \mathcal{N}$. If we choose $\psi_n(x) = \|x\|$ and $\sigma_n \geq l_n$, then $Q_n^R(x) = Q_n(x)$ on $\mathcal{X}_{a(n)}$ for all non-root nodes $n \in \mathcal{N}$.*

The proof exploits the Lipschitz continuity of f_n and the fact $Q_n^R(x)$ is an under-approximation of $Q_n(x)$ in an inductive argument. The details are given in Sect. A.4.1. In other words, for problems that already have Lipschitz continuous value functions, the regularization does not change the function value at any point. Thus the examples in the rest of this section serve the discussion not only for Algorithm 2, but for more general algorithms including SDDP and SDDiP.

5.1 General lipschitz continuous problems

We discuss the general Lipschitz continuous case, i.e., the nodal objective functions $f_n(z, y, x)$ are l_n-Lipschitz continuous in z but not necessarily convex. In this case we choose to approximate the value function using $\psi_n(x) = \|x\|$ and assume that $l_{n,\rho} \geq l_n$. We can set $l_{n,\lambda} = 0$ for all $n \in \mathcal{N}$, without loss of exactness of the approximation by the Proof of Proposition 3. We begin with the following lemma on the complexity of such approximation.

Lemma 7 *Consider a norm ball $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\| \leq D/2\}$ and a finite set of points $\mathcal{W} = \{w_k\}_{k=1}^K \subset \mathcal{X}$. Suppose that there is $\beta > 0$ and an L-Lipschitz continuous function $f : \mathcal{X} \to \mathbb{R}_+$ such that $\beta < f(w_k) < 2\beta$ for $k = 1, \ldots, K$. Define*

- $\underline{Q}(x) := \max_{k=1,\ldots,K} \{0, f(w_k) - L\|x - w_k\|\}$ *and*
- $\overline{Q}(x) := \min_{k=1,\ldots,K} \{f(w_k) + L\|x - w_k\|\}$.

If $K < \left(\frac{DL}{4\beta}\right)^d$, then $\min_{x \in \mathcal{X}} \underline{Q}(x) = 0$ and $\min_{x \in \mathcal{X}} \overline{Q}(x) > \beta$.

The proof is given in Sect. A.4.2. The lemma shows that if the number of points in $\mathcal{W}$ is too small, i.e. $K < (DL/2\beta)^d$, then the difference between the upper and lower bounds could be big, i.e. $\overline{Q}(\bar{x}) - \underline{Q}(\bar{x}) > \beta$ for some $\bar{x}$. In other words, in order to have a small gap between the upper and lower bounds, we need sufficient number of sample points. This lemma is directly used to provide a lower bound on the complexity of Algorithm 2.

Now we construct a Lipschitz continuous multistage problem defined on a chain, i.e., a scenario tree, where each stage has a single node, $N(t) = 1$ for $t = 1, \ldots, T$. The problem is given by the value functions in each stage as,

$$\begin{cases} Q_r = \min_{x_0 \in \mathcal{X}_r} Q_1(x_0), \\ Q_t(x_{t-1}) = \min_{x_t \in \mathcal{X}_t} \{f_t(x_{t-1}) + Q_{t+1}(x_t)\}, \ 1 \leq t \leq T-1, \quad (29) \\ Q_T(x_{T-1}) = f_T(x_{T-1}). \end{cases}$$

Here for all $t = 1, \ldots, T$, $f_t : \mathcal{X}_t \to \mathbb{R}_+$ is an L-Lipschitz continuous function that satisfies $\beta < f_t(x) < 2\beta$ for all $x \in \mathcal{X}_t$ with $\beta := \varepsilon/T$, the number of stages $T \geq 1$,

and $\varepsilon > 0$ is a fixed constant. The state space $\mathcal{X}_t := \mathcal{B}^d(D/2) \subset \mathbb{R}^d$ is a ball with radius $D/2 > 0$. We remark that ε will be the optimality gap in Theorem 4. So for a fixed optimality gap ε, we construct an instance of multistage problem (29) that will prove to be difficult for Algorithm 2 to solve. Also (29) is constructed such that there is no constraint coupling the state variables x_t in different stages.

By Lemma 6, if we choose $\psi_n(x) = \|x\|$ for all $n \in \mathcal{N}$ and $l_{n,\rho} = L$ for the problem (29), then we have $Q_t^R(x) = Q_t(x)$ for all $t = 1, \ldots, T$. The next theorem shows a lower bound on the iteration complexity of problem (29) with this choice of penalty functions.

Theorem 4 *For any optimality gap $\varepsilon > 0$, there exists a problem of the form* (29) *with subproblem oracles $\mathcal{O}_n^F, \mathcal{O}_n^B$, $n \in \mathcal{N}$, and $\mathcal{O}_r$, such that if Algorithm 2 gives* UPPERBOUND $-$ LOWERBOUND $\leq \varepsilon$ *in the i-th iteration, then*

$$i \geq \left(\frac{DLT}{4\varepsilon} \right)^d.$$

The proof is given in Sect. A.4.3. The theorem shows that in general Algorithm 2 needs at least $\mathcal{O}(T^d)$ iterations before termination. We comment that this is due to the fact that the approximation using generalized conjugacy is tight only locally. Without convexity, one may need to visit many states to cover the state space to achieve tight approximations of the value functions before the algorithm is guaranteed to find an ε-optimal solution.

5.2 Convex lipschitz continuous problems

In the above example for general Lipschitz continuous problem, we see that the complexity of Algorithm 2 grows at a rate of $\mathcal{O}(T^d)$. It remains to answer whether convexity could help us avoid this possibly undesirable growth rate in terms of d. We show that even by using linear cuts, rather than generalized conjugacy cuts, for convex value functions, the complexity lower bound of the proposed algorithms could not be substantially improved. We begin our discussion with a definition.

Definition 5 Given a d-sphere $\mathcal{S}^d(R) = \{x \in \mathbb{R}^{d+1} : \|x\|_2 = R\}$ with radius $R > 0$, a spherical cap with depth $\beta > 0$ centered at a point $x \in \mathcal{S}^d(R)$ is the set

$$\mathcal{S}_\beta^d(R, x) := \{y \in \mathcal{S}^d(R) : \langle y - x, x \rangle \geq -\beta R\}.$$

The next lemma shows that we can put many spherical caps on a sphere, the center of each is not contained in any other spherical cap, the proof of which is given in Sect. A.4.4.

Lemma 8 *Given a d-sphere $\mathcal{S}^d(R)$, $d \geq 2$ and depth $\beta < (1 - \frac{\sqrt{2}}{2})R$, there exists a finite set of points $\mathcal{W}$ with*

$$|\mathcal{W}| \geq \frac{(d^2 - 1)\sqrt{\pi}}{d} \frac{\Gamma(d/2 + 1)}{\Gamma(d/2 + 3/2)} \left(\frac{R}{2\beta} \right)^{(d-1)/2},$$

such that, for any $w \in \mathcal{W}$, $\mathcal{S}_\beta^d(R, w) \cap \mathcal{W} = \{w\}$.

Hereafter, we denote a set of points that satisfies Lemma 8 as $\mathcal{W}_\beta^d(R) \subset \mathcal{S}^d(R)$. Next we construct an L-Lipschitz convex function for any $L > 0$, $\varepsilon > 0$ that satisfies certain properties on $\mathcal{W}_{\varepsilon/L}^d(R)$. The proof is given in Sect. A.4.5.

Lemma 9 *Given positive constants* $\varepsilon > 0, L > 0$ *and a set* $\mathcal{W}_{\varepsilon/L}^d(R)$. *Let* $K := |\mathcal{W}_{\varepsilon/L}^d(R)|$. *For any values* $v_k \in (\varepsilon/2, \varepsilon)$, $k = 1, \ldots, K$, *define a function* $F : \mathcal{B}^{d+1}(R) \to \mathbb{R}$ *as* $F(x) = \max_{k=1,\ldots,K} \{0, v_k + \frac{L}{R} \langle w_k, x - w_k \rangle\}$. *Then F satisfies the following properties:*

1. *F is an L-Lipschitz convex function;*
2. $F(w_k) = v_k$ *for all* $w_k \in \mathcal{W}_{\varepsilon/L}^d(R)$;
3. *F is differentiable at all* w_k, *with* $v_k + \langle \nabla F(w_k), w_l - w_k \rangle < 0$ *for all* $l \neq k$;
4. *For any* $w_l \in \mathcal{W}_{\varepsilon/L}^d(R)$, $\underline{Q}_l(x) := \max_{k \neq l} \{0, v_k + \langle \nabla F(w_k), x - w_k \rangle\}$ *and* $\overline{Q}_l(x) := \text{conv}_{k \neq l} \{v_k + L \|x - w_k\|\}$ *satisfy*

$$\overline{Q}_l(w_l) - \underline{Q}_l(w_l) > \frac{3\varepsilon}{2}.$$

Now we present the multistage convex dual dynamic programming example based on the following parameters: $T \geq 2$ (number of stages), $L > 0$ (Lipschitz constant), $d \geq 3$ (state space dimension), $D = 2R > 0$ (state space diameter), and $\varepsilon > 0$ (optimality gap). Choose any $L_1, \ldots, L_T$ such that $L/2 \leq L_T < L_{T-1} < \cdots < L_1 \leq L$, and then construct finite sets $\mathcal{W}_t := \mathcal{W}_{\varepsilon/((T-1)L_{t+1})}^{d-1}(R) = \{w_{t,k}\}_{k=1}^{K_t}$, $K_t = |\mathcal{W}_t|$ as defined in Lemma 8 for $t = 1, \ldots, T - 1$. Moreover, define convex L_{t+1}-Lipschitz continuous functions F_t for some values $v_{t,k} \in (\varepsilon/(2T-2), \varepsilon/(T-1))$, $k = 1, \ldots, K_t$, and the finite sets $\mathcal{W}_t$. By Assumption 3, we define the stagewise independent scenario tree as follows. There are K_t distinct nodes in each stage $t = 1, \ldots, T - 1$, which can be denoted by an index pair $n = (t, k)$ for $k = 1, \ldots, K_t$, and all nodes are defined by the same data in the last stage T. Then we define our problem by specifying the nodal cost functions $f_r \equiv 0$, $f_{1,k}(x_0, y_1, x_1) := L_1 \|x_1 - w_{1,k}\|$ for $k = 1, \ldots, K_1$, $f_{t,k}(x_{t-1}, y_t, x_t) := F_{t-1}(x_{t-1}) + L_t \|x_t - w_{t,k}\|$ for $k = 1, \ldots, K_t$ and $t = 2, \ldots, T - 1$, and $f_{T,1}(x_{T-1}, y_T, x_T) := F_{T-1}(x_{T-1})$, and state spaces $\mathcal{X}_t = \mathcal{X} = \mathcal{B}^{d+1}(R)$. Alternatively, the value functions can be written as

$$\begin{cases} Q_{1,k} = \min_{x_1 \in \mathcal{X}} \left\{ L_1 \|x_1 - w_{1,k}\| + \mathcal{Q}_1(x_1) \right\}, \ \forall k \leq K_1, \\\\ Q_{t,k}(x_{t-1}) = \min_{x_t \in \mathcal{X}} \left\{ F_{t-1}(x_{t-1}) + L_t \|x_t - w_{t,k}\| + \mathcal{Q}_t(x_t) \right\}, k \leq K_t, \\\\ Q_{T,1}(x_{T-1}) = F_{T-1}(x_{T-1}), \end{cases}$$

$$(30)$$

where the second equation is defined for all $2 \leq t \leq T-1$, and the expected cost-to-go functions as

$$\mathcal{Q}_t(x_t) := \frac{1}{K_t} \sum_{k=1}^{K_t} Q_{t+1,k}(x_t), \quad t = 0, \ldots, T-1.$$

By Lemma 8,

$$\begin{aligned}
K_t &\geq \frac{((d-1)^2 - 1)\sqrt{\pi}}{d-1} \frac{\Gamma((d-1)/2+1)}{\Gamma((d-1)/2+3/2)} \left(\frac{RL_t(T-1)}{2\varepsilon} \right)^{(d-2)/2}, \\
&\geq \frac{d(d-2)\sqrt{\pi}}{d-1} \frac{\Gamma((d/2+1/2)}{\Gamma(d/2+1)} \left(\frac{DL(T-1)}{8\varepsilon} \right)^{(d-2)/2}.
\end{aligned}$$

Since for each value function $Q_{t,k}$ is L_t-Lipschitz continuous, we choose $\sigma_n = L_t$ with $\psi_n(x) = \|x\|$ for any $n = (t,k) \in \tilde{\mathcal{N}}(t)$ and $t = 1, \ldots, T$ such that by Lemma 6 we have $Q_{t,k}(x) = Q_{t,k}^R(x)$ for all $x \in \mathcal{X}$. Moreover, due to convexity, we set $l_{n,\rho} = 0$ for all $n \in \mathcal{N}$ and $l_{n,\lambda} = L_t$ for each $n \in \tilde{\mathcal{N}}(t)$ and $t = 1, \ldots, T$, i.e., the cuts are linear. Following the argument of Proposition 4, we know that such linear cuts are capable of tight approximations. With such a choice of regularization we have the following theorem on the complexity of Algorithm 2.

Theorem 5 *For any optimality gap $\varepsilon > 0$, there exists a multistage stochastic convex problem of the form* (30) *such that, if Algorithm 2 gives* UPPERBOUND $-$ LOWERBOUND $< \varepsilon$ *at i-th iteration, then*

$$i > \frac{1}{3} \frac{d(d-2)\sqrt{\pi}}{d-1} \frac{\Gamma(d/2+1/2)}{\Gamma(d/2+1)} \left(\frac{DL(T-1)}{8\varepsilon} \right)^{(d-2)/2}.$$

The proof is given in Sect. A.4.6. The theorem implies that, even if problem (2) is convex and has Lipschitz continuous value functions, the minimum iteration for Algorithm 2 to get a guaranteed ε-optimal root node solution grows as a polynomial of the ratio T/ε, with the degree being $d/2 - 1$.

We remark that Theorems 4 and 5 correspond to two different challenges of the SDDP type algorithms. The first challenge is that the backward step subproblem oracle may not give cuts that provide the desired approximation, which could happen when the value functions are nonconvex or nonsmooth. Theorem 4 results from the worst case that the backward step subproblem oracle leads to approximations of the value function in the smallest neighborhood.

The second challenge is that different nodes, or more generally, different scenario paths give different states in each stage, so sampling and solving the nodal problem on one scenario path provides little information to the nodal problem on another scenario path. In example (30), the linear cut obtained in each iteration does not provide any information on the subsequent iteration states (unless the same node is sampled again). From this perspective, we believe that unless some special structure of the problem is exploited, any algorithm that relies on local approximation of value functions will

face the "curse of dimensionality," i.e., the exponential growth rate of the iteration complexity in the state space dimensions.

6 Conclusions

In this paper, we propose three algorithms in a unified framework of dual dynamic programming for solving multistage stochastic mixed-integer nonlinear programs. The first algorithm is a generalization of the classic nested Benders decomposition algorithm, which deals with general scenario trees without the stagewise independence property. The second and third algorithms generalize SDDP with sampling procedures on a stagewise independent scenario tree, where the second algorithm uses a deterministic sampling approach, and the third one uses a randomized sampling approach. The proposed algorithms are built on regularization of value functions, which enables them to handle problems with value functions that are non-Lipschitzian or discontinuous. We show that the regularized problem preserves the feasibility and optimality of the original multistage program, when the corresponding penalty reformulation satisfies exact penalization. The key ingredient of the proposed algorithms is a new class of cuts based on generalized conjugacy for approximating nonconvex cost-to-go functions of the regularized problems.

We obtain upper and lower bounds on the iteration complexity of the proposed algorithms on MS-MINLP problem classes that allow exact Lipschitz regularization with predetermined penalty functions and parameters. The complexity analysis is new and deepens our understanding of the behavior of SDDP. For example, it is the first time to prove that the iteration complexity of SDDP depends polynomially on the number of stages, not exponentially, for both convex and nonconvex multistage stochastic programs, and this complexity dependence can be reduced to linear if the optimality gap is allowed to scale linearly with the number of stages, or if all the state spaces are finite sets. These findings resolve a conjecture of the late Prof. Shabbir Ahmed, who inspired us to work on this problem.

A Proofs

In this section, we present the proofs to the theorems, propositions, and lemmas that are not displayed in the main text.

A.1 Proofs for statements in Sect. 2

A.1.1 Proof for Proposition 1

Proof We show that Q_n is l.s.c. by showing the lower level sets $\text{lev}_a(Q_n) = \{z \in \mathcal{X}_{a(n)} : Q_n(z) \leq a\}$ are closed for all $a \in \mathbb{R}$. At any leaf node n, the expected cost-to-go function $Q_n(x_n)$ is zero, thus z is in $\text{lev}_a(Q_n)$ if and only if z is in the projection of the following set $\{(z, y, x) : (x, y) \in \mathcal{F}_n, f_n(z, y, x) \leq a\}$. Since f_n is defined on a compact set $\{(z, y, x) : z \in \mathcal{X}_{a(n)}, (x, y) \in \mathcal{F}_n\}$ and l.s.c. by Assumption 1, we know that the set $\{(z, y, x) : f_n(z, y, x) \leq a\}$ is compact. Moreover, since the projection $(z, y, x) \mapsto z$ is continuous, the image $\text{lev}_a(Q_n)$ is still compact, hence closed.

At any non-leaf node n, suppose Q_m is l.s.c. for all its child nodes $m \in \mathcal{C}(n)$. Then, Q_n is l.s.c. since Q_n is defined in (3) and $p_{nm} > 0$ for all m. A point $z \in \text{lev}_a(Q_n)$ if and only if z is in the projection of the set $\{(z, y, x) : (y, x) \in \mathcal{F}_n, f_n(z, y, x) + Q_n(x) \leq a\}$. Similarly, this shows $\text{lev}_a Q_n$ is closed since f_n, Q_n are l.s.c. and the projection $(z, y, x) \mapsto z$ is continuous. We thus conclude Q_n is l.s.c. for every node n in the scenario tree.

To show claims 1 and 2 in the proposition, take any two points $z_1, z_2 \in \mathcal{X}_{a(n)}$. Suppose $(x_1, y_1), (x_2, y_2) \in \mathcal{F}_n$ are the corresponding minimizers in the definition (2). Therefore, $Q_n(z_1) = f_n(z_1, y_1, x_1) + Q_n(x_1)$ and $Q_n(z_2) = f_n(z_2, y_2, x_2) + Q_n(x_2)$. If f_n is Lipschitz continuous in the first variable, then we have

$$
\begin{aligned}
Q_n(z_1) - Q_n(z_2) &= f_n(z_1, y_1, x_1) + Q_n(x_1) - f_n(z_2, y_2, x_2) - Q_n(x_2) \\
&\leq f_n(z_1, y_2, x_2) + Q_n(x_2) - f_n(z_2, y_2, x_2) - Q_n(x_2) \\
&\leq f_n(z_1, y_2, x_2) - f_n(z_2, y_2, x_2) \leq l_n \|z_1 - z_2\|.
\end{aligned}
$$

Likewise, by exchanging z_1 and z_2, we know that $Q_n(z_2) - Q_n(z_1) \leq l_n \|z_1 - z_2\|$. This proves that Q_n is Lipschitz continuous with the constant l_n.

To show that Q_n is convex, take any $t \in [0, 1]$. Since $\mathcal{X}_{a(n)}$ is convex, Q_n is defined at $tz_1 + (1 - t)z_2$. Thus,

$$
\begin{aligned}
Q_n(tz_1 + (1 - t)z_2) &\leq f_n(tz_1 + (1 - t)z_2, ty_1 + (1 - t)y_2, tx_1 + (1 - t)x_2) \\
&\quad + Q_n(tx_1 + (1 - t)x_2) \\
&\leq tf_n(z_1, y_1, x_1) + (1 - t)f_n(z_2, y_2, x_2) + tQ_n(x_1) \\
&\quad + (1 - t)Q_n(x_2) \\
&= tQ_n(z_1) + (1 - t)Q_n(z_2).
\end{aligned}
$$

The first inequality follows from the definition (2), while the second inequality follows from the convexity of f_n and Q_n. This shows Q_n is convex. □

A.1.2 Proof for Proposition 2

Proof First we show that the partial inf-convolution

$$
f_n \,\square\, (\sigma_n \psi_n)(x_{a(n)}, y_n, x_n) := \min_{z \in \mathcal{X}_{a(n)}} f_n(z_n, y_n, x_n) + \sigma_n \psi_n(x_{a(n)} - z_n)
$$

is σ_n-Lipschitz continuous in the first variable $x_{a(n)}$. Note that the minimum is well-defined since $\mathcal{X}_{a(n)}$ is compact and the functions $f_n, \sigma_n \psi_n$ are l.s.c.. Besides, since $z = x_{a(n)}$ is a feasible solution in the minimization, we know that $f_n \square (\sigma_n \psi_n)(x_{a(n)}, y_n, x_n) \leq f_n(x_{a(n)}, y_n, x_n)$ for all $x_{a(n)} \in \mathcal{X}_{a(n)}$ and $(x_n, y_n) \in \mathcal{F}_n$. Pick any $x_1, x_2 \in \mathcal{X}_{a(n)}$, $(x, y) \in \mathcal{F}_n$, and let $z_1, z_2 \in \mathcal{X}_{a(n)}$ be the corresponding minimizers in the definition of $f_n \square (\sigma_n \psi_n)(x_1, y, x)$ and $f_n \square (\sigma_n \psi_n)(x_2, y, x)$, respectively. By definition,

$$
\begin{aligned}
f_n \square (\sigma_n \psi_n)(x_1, y, x) &- f_n \square (\sigma_n \psi_n)(x_2, y, x) \\
&= f_n(z_1, y, x) + \psi(x_1 - z_1) - f_n(z_2, y, x) - \psi(x_2 - z_2) \\
&\leq f_n(z_2, y, x) + \psi(x_1 - z_2) - f_n(z_2, y, x) - \psi(x_2 - z_2) \leq \sigma_n \|x_1 - x_2\|.
\end{aligned}
$$

Similarly, we can get $f_n \square (\sigma_n \psi_n)(x_2) - f_n \square (\sigma_n \psi_n)(x_1) \leq \sigma_n \|x_1 - x_2\|$ by exchanging x_1, x_2 and z_1, z_2 in the above inequality. Therefore, $f_n \square (\sigma_n \psi_n)$ is σ_n-Lipschitz continuous in the first variable $x_{a(n)}$.

The regularized problem (4) can be viewed as replacing the nodal objective function f_n with the inf-convolution $f_n \square (\sigma_n \psi_n)$. Then by Proposition 1, $Q_n^R(x)$ is σ_n-Lipschitz continuous on $\mathcal{X}_{a(n)}$. Moreover, if the original problem (2) is convex and ψ_n are convex penalty functions, then $f_n \square (\sigma_n \psi_n)$ is also convex. Proposition 1 ensures $Q_n^R(x)$ is also convex on $\mathcal{X}_{a(n)}$. □

A.1.3 Proof for Lemma 1

Proof By definition, we have $Q_n^R(x_n) \leq Q_n(x_n)$ for all $n \in \mathcal{N}$, $n \neq r$. We show the other direction by contradiction. Suppose there exists a node $n \in \mathcal{N}$ such that $Q_n^R(x_{a(n)}) < Q_n(x_{a(n)})$. By definition, there exist $z_m' \in \mathcal{X}_{a(m)}$ and $(x_m', y_m') \in \mathcal{F}_m$ for all nodes in the subtree $m \in \mathcal{T}(n)$, such that

$$
Q_n^R(x_{a(n)}) = \frac{1}{p_n} \sum_{m \in \mathcal{T}(n)} p_m \left[f_m(z_m', y_m', x_m') + \sigma_m \psi_m(x_{a(m)}' - z_m') \right].
$$

We can extend $(x_m', y_m', z_m')_{m \in \mathcal{T}(n)}$ to a feasible solution $(z_m', y_m', x_m')_{m \in \mathcal{N}}$ of the regularized problem by setting $z_m' = x_{a(m)}, y_m' = y_m$, and $x_m' = x_m$ for all $m \notin \mathcal{T}(n)$. Thus

$$
\begin{aligned}
v^{\text{reg}} &\leq \sum_{m \in \mathcal{T}(n)} p_m f_m(z_m', y_m', x_m') + \sum_{m \notin \mathcal{T}(n)} p_m f_m(z_m', y_m', x_m') \\
&= p_n Q_n^R(x_{a(n)}) + \sum_{m \notin \mathcal{T}(n)} p_m f_m(x_{a(m)}, y_m, x_m) \\
&< p_n Q_n(x_{a(n)}) + \sum_{m \notin \mathcal{T}(n)} p_m f_m(x_{a(m)}, y_m, x_m) \\
&= \sum_{m \in \mathcal{T}(n)} p_m f_m(x_{a(m)}, y_m, x_m) + \sum_{m \notin \mathcal{T}(n)} p_m f_m(x_{a(m)}, y_m, x_m) = v^{\text{prim}}.
\end{aligned}
$$

This leads to a contradiction with the assumption that $v^{\text{reg}} = v^{\text{prim}}$. Therefore, we conclude that $Q_n^R(x_{a(n)}) = Q_n(x_{a(n)})$ for all $n \in \mathcal{N}, n \neq r$. □

A.1.4 Proof for Proposition 3

Proof If $l_{n,\rho} \geq \sigma_n$, then $(\lambda, \rho) = (0, \sigma_n)$ is contained in $\mathcal{U}_n$, and therefore, is a dual feasible solution for (11). Thus, we have

$$Q_n(\bar{x}_n) \geq C_n^{\Phi_n^{\bar{x}_n}}(\bar{x}_n \mid \hat{\lambda}_n, \hat{\rho}_n, \hat{v}_n) = \hat{v}_n \geq \min_{z \in \mathcal{X}_{a(n)}} \{Q_n(z) + \sigma_n \psi_n(\bar{x}_n - z)\} = Q_n^R(\bar{x}_n)$$
$$= Q_n(\bar{x}_n),$$

where the first inequality is the validity of the generalized conjugacy cut (10) and the second and the last equality are due to Lemma 1 for $(\bar{x}_n, \bar{y}_n)_{n \in \mathcal{N}}$ being an optimal solution to problem (1). This completes the proof. □

A.1.5 Proof for Lemma 2

Proof The minimums in (13) are well-defined because of the compactness of $\mathcal{X}$ and lower semicontinuity of Q. Take any $x \in \mathcal{X}$. Since both the primal set $\mathcal{X}$ and the dual set $\{\lambda \in \mathbb{R}^d : \|\lambda\|_* \leq \sigma\}$ are bounded, by strong duality (cf. Theorem 3.1.30 in [27]), we have

$$\max_{\|\lambda\|_* \leq \sigma} \min_{z \in \mathcal{X}} \{Q(z) + \langle \lambda, x - z \rangle\} = \min_{z \in \mathcal{X}} \max_{\|\lambda\|_* \leq \sigma} \{Q(z) + \langle \lambda, x - z \rangle\} = \min_{z \in \mathcal{X}} \{Q(z)$$
$$+ \sigma \|x - z\|\},$$

which completes the proof. □

A.1.6 Proof for Proposition 4

Proof By definition, $Q_n^R(x) \leq Q_n \square (\sigma_n \psi_n)(x)$. Since $\psi_n(x) = \|x\|$ is convex, by Proposition 2, $Q_n^R(x)$ is convex. Then by Lemma 2, we have

$$Q_n^R(x) = \max_{\|\lambda\|_* \leq \sigma_n} \min_{z \in \mathcal{X}_{a(n)}} \{Q_n(z) + \langle \lambda, x - z \rangle\}.$$

Therefore,

$$C_n^{\Phi_n^{\bar{x}_n}}(\bar{x}_n \mid \hat{\lambda}_n, \hat{\rho}_n, \hat{v}_n) = \hat{v}_n = \max_{\|\lambda\|_* \leq l_{n,\lambda}} \min_{z \in \mathcal{X}_{a(n)}} \{Q_n(z) + \langle \lambda, \bar{x}_n - z \rangle\}$$
$$\geq \max_{\|\lambda\|_* \leq \sigma_n} \min_{z \in \mathcal{X}_{a(n)}} \{Q_n(z) + \langle \lambda, \bar{x}_n - z \rangle\} \geq Q_n^R(\bar{x}_n).$$

By Lemma 1, $Q_n^R(\bar{x}_n) = Q_n(\bar{x}_n)$ if $(\bar{x}_n, \bar{y}_n)_{n \in \mathcal{N}}$ is an optimal solution to problem (1). Therefore, we conclude that $C_n^{\Phi_n^{\bar{x}_n}}(\bar{x}_n \mid \hat{\lambda}_n, \hat{\rho}_n, \hat{v}_n) = Q_n(\bar{x}_n)$ due to the validness of $C_n^{\Phi_n^{\bar{x}_n}}$ by (10). □

A.2 Proofs for statements in Sect. 3

A.2.1 Proof for Proposition 5

Proof Let $L_n := \sum_{m \in \mathcal{C}(n)} p_{nm}(l_{m,\lambda} + l_{m,\rho})$ for simplicity. We prove the proposition recursively for nodes $n \in \mathcal{N}$, and inductively for iteration indices $i \in \mathbb{N}$. For leaf nodes and the first iteration, it holds obviously because $\underline{\mathcal{Q}}_n^i(x) = 0$ for any leaf node $n \in \mathcal{N}$ with $\mathcal{C}(n) = \varnothing$, and $\underline{\mathcal{Q}}_n^0(x) = 0$ from the definition (14). Now suppose for some $n \in \mathcal{N}$, and $i \in \mathbb{N}$, it holds for all $m \in \mathcal{C}(n)$ that $\underline{\mathcal{Q}}_m^i(x) \leq \mathcal{Q}_m(x)$, $\underline{\mathcal{Q}}_n^{i-1}(x) \leq \mathcal{Q}_n(x)$, and that $\underline{\mathcal{Q}}_n^{i-1}(x)$ is L_n-Lipschitz continuous. Then it follows from (12), (16), and (B) that $C_m^i(x \mid \hat{\lambda}_m^i, \hat{\rho}_m^i, \underline{v}_m^i) \leq \mathcal{Q}_m(x)$ for all $m \in \mathcal{C}(n)$. By (15), C_m^i is $(l_{m,\lambda} + l_{m,\rho})$-Lipschitz continuous so $\sum_{m \in \mathcal{C}(n)} p_{nm} C_m^i$ is L_n-Lipschitz continuous. Thus the pointwise maximum of $\underline{\mathcal{Q}}_n^{i-1}(x)$ and $\sum_{m \in \mathcal{C}(n)} p_{nm} C_m^i(x \mid \hat{\lambda}_n^i, \hat{\rho}_n^i, \underline{v}_n^i)$ (cf. (14)) is still dominated by $\mathcal{Q}_n(x)$ and L_n-Lipschitz continuous. □

A.2.2 Proof for Proposition 6

Proof Let $L_n = \sum_{m \in \mathcal{C}(n)} p_{nm} \sigma_m$ for simplicity in this proof. We prove the statement by induction on the number of iterations i. When $i = 1$, $\overline{\mathcal{Q}}_n^1(x) = \min\{+\infty, \sum_{m \in \mathcal{C}(n)} p_{nm}(\bar{v}_m^i + \sigma_m \|x - x_n^i\|)\} = \sum_{m \in \mathcal{C}(n)} p_{nm}(\bar{v}_m^i + \sigma_m \|x - x_n^i\|)$ which is clearly L_n-Lipschitz continuous. For any leaf node n, $\overline{\mathcal{Q}}_n^1 \equiv 0 = \mathcal{Q}_n^R$ by definition. Going recursively from leaf nodes to the root node, suppose $\overline{\mathcal{Q}}_m^1 \geq \mathcal{Q}_m^R$ for all $m \in \mathcal{C}(n)$ for some node n, then we have

$$
\begin{aligned}
\bar{v}_m^1 &= f_m(z_m^1, y_m^1, x_m^1) + \sigma_m \psi_m(x_n^1 - z_m^1) + \overline{\mathcal{Q}}_m^1(x_m^1) \\
&\geq \min\{f_m(z, y, x) + \sigma_m \psi_m(x_n^1 - z) + \overline{\mathcal{Q}}_m^1(x) : (x, y) \in \mathcal{F}_m, z \in \mathcal{X}_n\} \\
&\geq \min\{f_m(z, y, x) + \sigma_m \psi_m(x_n^1 - z) + \mathcal{Q}_m^R(x) : (x, y) \in \mathcal{F}_m, z \in \mathcal{X}_n\} \\
&= \mathcal{Q}_m^R(x_n^1).
\end{aligned}
\tag{31}
$$

Thus $\overline{\mathcal{Q}}_n^1(x) = \sum_{m \in \mathcal{C}(n)} p_{nm}(\bar{v}_m^1 + \sigma_m \|x - x_n^1\|) \geq \mathcal{Q}_n^R(x)$ for all $x \in \mathcal{X}_n$ by the σ_m-Lipschitz continuity of the regularized value functions $\mathcal{Q}_m^R(x)$ for all $m \in \mathcal{C}(n)$ shown in Proposition 2.

Now assume that the statement holds for all iterations up to $i-1$. For any leaf node n, $\overline{\mathcal{Q}}_n^i \equiv 0 = \mathcal{Q}_n^R$ still holds by definition. For any non-leaf node n, suppose $\overline{\mathcal{Q}}_m^i \geq \mathcal{Q}_m^R$ for all $m \in \mathcal{C}(n)$. Then by the same argument (31), we know that $\bar{v}_m^i \geq \mathcal{Q}_m^R(x_n^i)$. By induction hypothesis, $\overline{\mathcal{Q}}_n^{i-1}(x) \geq \mathcal{Q}_n^R(x)$ for all $x \in \mathcal{X}_n$. So for the cases without convexity, $\overline{\mathcal{Q}}_n^i(x) = \min\{\overline{\mathcal{Q}}_n^{i-1}(x), \sum_{m \in \mathcal{C}(n)} p_{nm}(\bar{v}_m^i + \sigma_m \|x - x_n^i\|)\}$ is L_n-Lipschitz continuous and satisfies $\overline{\mathcal{Q}}_n^i(x) \geq \mathcal{Q}_n^R(x)$ since $\sum_{m \in \mathcal{C}(n)} p_{nm}(\bar{v}_m^i + \sigma_m \|x - x_n^i\|) \geq \mathcal{Q}_n^R(x)$ for all $x \in \mathcal{X}_n$ following Proposition 2.

It remains to show that in the convex case $\overline{Q}_n^i(x)$ is still L_n-Lipschitz continuous and satisfies $\overline{Q}_n^i(x) \geq Q_n^R(x)$ for any $x \in \mathcal{X}_n$. Note that $Q_n^R(x)$ can be naturally extended to the entire space $\mathbb{R}^{d_n} \supset \mathcal{X}_n$ since $Q_n^R \square (L_n \|\cdot\|)(x) = Q_n^R(x)$ for any $x \in \mathcal{X}_n$ by the L_n-Lipschitz continuity of Q_n^R. The above argument of the base case $i = 1$ for the nonconvex case can be directly applied to the convex case over $x \in \mathbb{R}^{d_n}$. Now assume that $\overline{Q}_n^{i-1}(x)$ is L_n-Lipschitz continuous on $\mathbb{R}^{d_n}$ and $\overline{Q}_n^{i-1}(x) \geq Q_n^R(x)$ for $x \in \mathbb{R}^{d_n}$ up to $i - 1$. Since $Q_n'(x) := \min\{\overline{Q}_n^{i-1}(x), \sum_{m \in \mathcal{C}(n)} p_{nm}(\bar{v}_m^i + \sigma_m \|x - x_n^i\|)\}$ is L_n-Lipschitz continuous in $x \in \mathbb{R}^{d_n}$, we claim that the supremum in the definition (19) can be attained within the dual norm ball $\mathcal{B}_*(L_n) := \{\lambda \in \mathbb{R}^{d_n} : \|\lambda\|_* \leq L_n\}$. In fact, for any $\lambda \notin \mathcal{B}_*(L_n)$, the infimum

$$\inf_{z \in \mathbb{R}^{d_n}} \{Q_n'(z) + \langle \lambda, z - x \rangle\} \leq \inf_{z \in \mathbb{R}^{d_n}} \{Q_n'(x) + L_n \|z - x\| + \langle \lambda, z - x \rangle\} = -\infty.$$

As a result, $\overline{Q}_n^i(x)$ is a supremum of L_n-Lipschitz linear functions (of the forms $l(x) := \overline{Q}_n^i(\hat{z}) + \langle \hat{\lambda}, \hat{z} - x \rangle$ where $\hat{\lambda} \in \mathcal{B}_*(L_n)$ and $\hat{z} \in \mathcal{X}_n$) and thus is also an L_n-Lipschitz continuous function. Therefore, $Q_n'(x) \geq Q_n^R(x)$ for all $x \in \mathbb{R}^{d_n}$. By (19), $\overline{Q}_n^i(x) = (Q_n')^{**}(x) \geq (Q_n^R)^{**}(x) = Q_n^R(x)$ for all $x \in \mathbb{R}^{d_n}$. This completes the proof. □

A.2.3 Proof for Proposition 7

Proof From the definition of v^{reg} and Proposition 6,

$$v^{\text{reg}} \leq f_r(x_{a(r)}, y_r^*, x_r^*) + Q_r^R(x_r^*) \leq f_r(x_{a(r)}, y_r^*, x_r^*) + \overline{Q}_r^i(x_r^*) \leq \text{UPPERBOUND}.$$

Since UPPERBOUND $-$ LOWERBOUND $\leq \varepsilon$, we have

$$f_r(x_{a(r)}, y_r^*, x_r^*) + \overline{Q}_r^i(x_r^*) \leq f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \underline{Q}_r^i(x_r^{i+1}) + \varepsilon.$$

Then, using the optimality of (x_r^{i+1}, y_r^{i+1}) given by $\mathcal{O}_r(\underline{Q}_r^i)$ and the fact that $\underline{Q}_r^i(x) \leq Q_r(x)$, we see that

$$f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \underline{Q}_r^i(x_r^{i+1}) \leq \min_{(x,y) \in \mathcal{F}_r} \{f_r(x_{a(r)}, y, x) + Q_r(x)\} = v^{\text{prim}}.$$

Under Assumption 2, $v^{\text{reg}} = v^{\text{prim}}$. Therefore, combining all the above inequalities, we have shown that

$$v^{\text{reg}} \leq f_r(x_{a(r)}, y_r^*, x_r^*) + \overline{Q}_r^i(x_r^*) \leq v^{\text{reg}} + \varepsilon,$$

which means (x_r^*, y_r^*) is an ε-optimal root node solution to the regularized problem (4). Now suppose $\overline{Q}_r^i(x_r^{i+1}) - \underline{Q}_r^i(x_r^{i+1}) \leq \varepsilon$ for some iteration index i. Note that UPPERBOUND $\leq f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \overline{Q}_r^i(x_r^{i+1})$, we have

$$\text{UPPERBOUND} - \text{LOWERBOUND} \le f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \overline{Q}_r^i(x_r^{i+1})$$
$$- (f_r(x_{a(r)}, y_r^{i+1}, x_r^{i+1}) + \underline{Q}_r^i(x_r^{i+1}))$$
$$= \overline{Q}_r^i(x_r^{i+1}) - \underline{Q}_r^i(x_r^{i+1}) \le \varepsilon.$$

Therefore the algorithm terminates after the i-th iteration. $\qquad\square$

A.3 Proofs for statements in Sect. 4

A.3.1 Proof for Lemma 3

Proof By definition (14), $\underline{Q}_m^i(x) \ge \underline{Q}_m^{i-1}(x)$ on $\mathcal{X}_m$ for all $m \in \mathcal{C}(n)$. If the problem is convex with $\psi_n = \|\cdot\|$, then by Lemma 2, we have

$$\underline{v}_m^i = \max_{\|\lambda\|_* \le l_{m,\lambda}} \min_{\substack{(x,y)\in\mathcal{F}_m, \\ z\in\mathcal{X}_n}} \left\{ f_m(z, y, x) + \langle \lambda, x_n^i - z \rangle + \underline{Q}_m^i(x) \right\}$$

$$= \min_{\substack{(x,y)\in\mathcal{F}_m, \\ z\in\mathcal{X}_n}} \left\{ f_m(z, y, x) + l_{n,\lambda}\|x_n^i - z\| + \underline{Q}_m^i(x) \right\}$$

$$\ge \min_{\substack{(x,y)\in\mathcal{F}_m, \\ z\in\mathcal{X}_n}} \left\{ f_m(z, y, x) + \sigma_m\|x_n^i - z\| + \underline{Q}_m^i(x) \right\}$$

as $l_{m,\lambda} \ge \sigma_m$ in Definition 3. Otherwise, by definition (16) and the fact that $(\lambda, \rho) = (0, \sigma_n)$ is a dual feasible solution for the problem (B), we have

$$\underline{v}_m^i = \max_{(\lambda,\rho)\in\mathcal{U}_m} \min_{\substack{(x,y)\in\mathcal{F}_m, \\ z\in\mathcal{X}_n}} \left\{ f_m(z, y, x) + \langle \lambda, x_n^i - z \rangle + \rho\psi_n(x_n^i - z) + \underline{Q}_m^i(x) \right\}$$

$$\ge \min_{\substack{(x,y)\in\mathcal{F}_m, \\ z\in\mathcal{X}_n}} \left\{ f_m(z, y, x) + \sigma_n\psi_n(x_n^i - z) + \underline{Q}_m^i(x) \right\}.$$

Thus in both cases, we have

$$\underline{v}_m^i \ge \min_{\substack{(x,y)\in\mathcal{F}_m, \\ z\in\mathcal{X}_n}} \left\{ f_m(z, y, x) + \sigma_n\psi_n(x_n^i - z) + \underline{Q}_m^{i-1}(x) \right\}$$

$$= f_m(z_m^i, y_m^i, x_m^i) + \sigma_n\psi_n(x_n^i - z_m^i) + \underline{Q}_m^{i-1}(x_m^i)$$

for all $m \in \mathcal{C}(n)$. The last equality is due to the forward step subproblem oracle $\mathcal{O}_m^{\text{F}}(x_n^i, \underline{Q}_m^{i-1})$ in the algorithm. Meanwhile, note that $\overline{Q}_m^i(x) \le \overline{Q}_m^{i-1}(x)$ for $x \in \mathcal{X}_m$. By definition (18), we have

$$\bar{v}_m^i = f_m(z_m^i, y_m^i, x_m^i) + \sigma_n\psi_n(x_n^i - z_m^i) + \overline{Q}_m^i(x_m^i)$$

$$\le f_m(z_m^i, y_m^i, x_m^i) + \sigma_n\psi_n(x_n^i - z_m^i) + \overline{Q}_m^{i-1}(x_m^i)$$

for all $m \in \mathcal{C}(n)$. Note that by definition (17), $\overline{\mathcal{Q}}_n^i(x_n^i) \leq \sum_{m \in \mathcal{C}(n)} p_{nm} \bar{v}_m^i$ and by definitions (14) and (15), $\underline{\mathcal{Q}}_n^i(x_n^i) \geq \sum_{m \in \mathcal{C}(n)} p_{nm} C_m^i(x_n^i \mid \hat{\lambda}_m^i, \hat{\rho}_m^i, \underline{v}_m^i) = \sum_{m \in \mathcal{C}(n)} p_{nm} \underline{v}_m^i$. Therefore,

$$
\begin{aligned}
\overline{\mathcal{Q}}_n^i(x_n^i) - \underline{\mathcal{Q}}_n^i(x_n^i) &\leq \sum_{m \in \mathcal{C}(n)} p_{nm}(\bar{v}_m^i - \underline{v}_m^i) \\
&\leq \sum_{m \in \mathcal{C}(n)} p_{nm}[\overline{\mathcal{Q}}_m^{i-1}(x_m^i) - \underline{\mathcal{Q}}_m^{i-1}(x_m^i)] \leq \sum_{m \in \mathcal{C}(n)} p_{nm} \gamma_m(\delta).
\end{aligned}
$$

Note that $\overline{\mathcal{Q}}_n^i(x)$ is $\left(\sum_{m \in \mathcal{C}(n)} p_{nm} \sigma_m \right)$-Lipschitz continuous by Proposition 6, and $\underline{\mathcal{Q}}_m^i(x)$ is $\left[\sum_{m \in \mathcal{C}(n)} p_{nm}(l_{n,\lambda} + l_{n,\rho}) \right]$-Lipschitz continuous on $\mathcal{X}_n$ by Proposition 5. Since we have $l_{m,\lambda} + l_{m,\rho} \geq \sigma_m$ regardless of convexity of the problem, it holds that $\overline{\mathcal{Q}}_n^i(x)$ and $\underline{\mathcal{Q}}_n^i(x)$ are both L_n-Lipschitz continuous. Therefore, for any $x \in \mathcal{X}_n$, $\|x - x_n^i\| \leq \delta_n/(2L_n)$, we have

$$
\begin{aligned}
\overline{\mathcal{Q}}_n^i(x) - \underline{\mathcal{Q}}_n^i(x) &\leq \overline{\mathcal{Q}}_n^i(x_n^i) - \underline{\mathcal{Q}}_n^i(x_n^i) + 2L_n \|x - x_n^i\| \quad \leq \sum_{m \in \mathcal{C}(n)} p_{nm} \gamma_m(\delta) + \delta_n \\
&= \gamma_n(\delta).
\end{aligned}
$$

This completes the proof. $\qquad\square$

A.3.2 Proof for Lemma 4

Proof We claim that for any $i, j \in \mathcal{I}_n, i \neq j$, then $\|x_n^i - x_n^j\| > \delta_n/(2L_n)$. Assume for contradiction that $\|x_n^i - x_n^j\| \leq \delta_n/(2L_n)$ for some $i < j$ and $i, j \in \mathcal{I}_n(\delta)$. By the definition of $\mathcal{I}_n(\delta)$, $\gamma_m^i \leq \gamma_m(\delta)$ for all $m \in \mathcal{C}(n)$. By Lemma 3, $\overline{\mathcal{Q}}_n^i(x) - \underline{\mathcal{Q}}_n^i(x) \leq \gamma_n(\delta)$ for all $x \in \mathcal{X}_n$, $\|x - x_n^i\| \leq \delta_n/(2L_n)$. Since $j > i$ and $\|x_n^i - x_n^j\| \leq \delta_n/(2L_n)$, this implies $\gamma_n^j = \overline{\mathcal{Q}}_n^j(x_n^j) - \underline{\mathcal{Q}}_n^j(x_n^j) \leq \gamma_n(\delta)$, which is a contradiction with $j \in \mathcal{I}_n(\delta)$. Hence we prove the claim.

Let $\mathcal{B}(R), \mathcal{B}(R, x) \subseteq \mathbb{R}^d$ denote the closed balls with radius $R \geq 0$, centered at 0 and x, respectively. It follows from the claim that the closed balls $\mathcal{B}(\delta_n/(4L_n), x_n^i)$ are non-overlapping for all $i \in \mathcal{I}_n(\delta)$, each with the volume $\text{Vol}\mathcal{B}(\delta_n/(4L_n))$. Thus the sum of the volumes of these balls is $|\mathcal{I}_n(\delta)|\text{Vol}\mathcal{B}(\delta_n/(4L_n))$. Note that for each index $i \in \mathcal{I}_n(\delta)$, $x_n^i \in \mathcal{X}_n$ and hence $x_n^i \in \mathcal{B}_{n,k}$ for some k. The closed ball $\mathcal{B}(\delta_n/(4L_n), x_n^i) \subseteq \mathcal{B}_{n,k} + \mathcal{B}(\delta_n/(4L_n))$, and therefore

$$
\bigcup_{i \in \mathcal{I}_n(\delta)} \mathcal{B}(\delta_n/(4L_n), x_n^i) \subseteq \bigcup_{k=1}^{K_n} (\mathcal{B}_{n,k} + \mathcal{B}(\delta_n/(4L_n))).
$$

It follows that

$$
\mathrm{Vol}\left[\bigcup_{i\in\mathcal{I}_n(\delta)}\mathcal{B}(\delta_n/(4L_n),x_n^i)\right]=|\mathcal{I}_n(\delta)|\cdot\mathrm{Vol}\mathcal{B}(\delta_n/(4L_n))\leq
$$

$$
\mathrm{Vol}\left[\bigcup_{k=1}^{K_n}(\mathcal{B}_{n,k}+\mathcal{B}(\delta_n/(4L_n)))\right]\leq\sum_{k=1}^{K_n}\mathrm{Vol}\left(\mathcal{B}_{n,k}+\mathcal{B}(\delta_n/(4L_n))\right).
$$

Therefore,

$$
|\mathcal{I}_n(\delta)|\leq\sum_{k=1}^{K_n}\frac{\mathrm{Vol}\left(\mathcal{B}_{n,k}+\mathcal{B}(\delta_n/(4L_n))\right)}{\mathrm{Vol}\mathcal{B}(\delta_n/(4L_n))}=\sum_{k=1}^{K_n}\left(1+\frac{2L_nD_{n,k}}{\delta_n}\right)^{d_n}.
$$

This completes the proof.   □

A.3.3 Proof for Lemma 5

Proof For each iteration $i\in\mathbb{N}$, the event $\cup_{j=1}^M\{\gamma_0^{i-1,j}\leq\gamma_0(\delta)=\varepsilon\}$ is Σ_{i-1}-measurable, so it suffices to prove this inequality for its complement in $A_i(\delta)$. Note that

$$
\mathrm{Prob}\{\gamma_t^{i,j}=\tilde{\gamma}_t^{i,j}\mid\Sigma_{i-1}\}\geq\mathrm{Prob}\{P_t^{i,j}=n(\tilde{\gamma}_t^{i,j})\mid\Sigma_{i-1}\},
$$

where $n(\tilde{\gamma}_t^{i,j})$ is the smallest node index $n\in\tilde{\mathcal{N}}(t)$ such that $\mathcal{Q}_t^R(x_n)-\underline{\mathcal{Q}}_t^{i-1}(x_n)=\tilde{\gamma}_t^{i,j}$ for $(x_n,y_n,z_n)=\mathcal{O}_n^F(x_{t-1}^{i,j},\underline{\mathcal{Q}}_n^{i-1})$, which is determined given Σ_{i-1}. Using the same argument as in the Proof of Theorem 1, Lemma 3 shows that the event $\cap_{t=1}^{T-1}\{\gamma_t^{i,j}=\tilde{\gamma}_t^{i,j}\}$ implies the event $\{i\in\cup_{t=0}^{T-1}\mathcal{I}_t(\delta)\}$ and hence the event $A_i(\delta)$ for each $j=1,\ldots,M$. Therefore, since Σ_{i-1} is contained in $\sigma(\Sigma_\infty^{\mathrm{oracle}},\sigma\{P^{i',j'}\}_{(i',j')\neq(i,j)})$, by the independent, uniform sampling (Assumption 4), we have

$$
\mathrm{Prob}(A_i(\delta)\mid\Sigma_{i-1})
$$

$$
\geq\mathrm{Prob}\left(\bigcup_{j=1}^M\bigcap_{t=1}^{T-1}\{\gamma_t^{i,j}=\tilde{\gamma}_t^{i,j}\}\,\middle|\,\Sigma_{i-1}\right)
$$

$$
\geq\mathrm{Prob}\left(\bigcup_{j=1}^M\bigcap_{t=1}^{T-1}\{\gamma_t^{i,j}=n(\tilde{\gamma}_t^{i,j})\}\,\middle|\,\Sigma_{i-1}\right)
$$

$$
=1-\left(1-\mathrm{Prob}\left(\bigcap_{t=1}^{T-1}\{\gamma_t^{i,j}=n(\tilde{\gamma}_t^{i,j})\}\,\middle|\,\Sigma_{i-1}\right)\right)^M=1-(1-1/N)^M.
$$

Here, the last step follows from $\text{Prob}(\bigcap_{t=1}^{T-1}\{\gamma_t^{i,j} = n(\tilde{\gamma}_t^{i,j})\} \mid \Sigma_{i-1}) = \prod_{t=1}^{T-1}\text{Prob}(\{\gamma_t^{i,j} = n(\tilde{\gamma}_t^{i,j})\} \mid \Sigma_{i-1}) = \prod_{t=1}^{T-1}(1/N_t) = N.$ □

A.3.4 Proof for Theorem 3

Proof Let $a_i := \mathbb{1}_{A_i}$ denote the indicator of the event A_i for $i \in \mathbb{N}$, and $S_i := \sum_{i'=1}^{i} a_i$. Note that the event $\{\iota \geq i\}$ implies the event $\{S_i \leq I\}$, so we want to bound probability of the latter for sufficiently large indices i.

By Lemma 5, we see that the adapted sequence $\{S_i - i\nu\}_{i=1}^{\infty}$ is a submartingale with respect to the filtration $\{\Sigma_i\}_{i=1}^{\infty}$, because

$$\mathbb{E}(S_i - i\nu \mid \Sigma_{i-1}) = S_{i-1} - (i-1)\nu + (\mathbb{E}(a_i \mid \Sigma_{i-1}) - \nu) \geq S_{i-1} - (i-1)\nu.$$

Moreover, it has a bounded difference as $S_i + i\nu - (S_{i-1} + (i-1)\nu) = a_i + \nu \leq 2$ almost surely. Now apply the one-sided Azuma-Hoeffding inequality and we get for any $k > 0$ that

$$\text{Prob}(S_i \leq i\nu - k) \leq \exp\left(-\frac{k^2}{8i}\right).$$

For any $\kappa > 1$, take the smallest iteration index i such that $i\nu \geq \kappa I$, and set $k := (\kappa - 1)I$. Since $I \geq \frac{i\nu}{2\kappa}$, the probability bound can be then written as

$$\text{Prob}(\iota \geq i) \leq \text{Prob}(S_i \leq I) \leq \exp\left(-\frac{(\kappa-1)^2 I^2}{8i}\right) \leq \exp\left(-\frac{(\kappa-1)^2 I\nu}{16\kappa}\right)$$

Substitute the left-hand-side with $\text{Prob}(\iota \geq 1 + \frac{\kappa I}{\nu})$ using the definition of i and we have obtained the desired inequality. □

A.4 Proofs for statements in Sect. 5

A.4.1 Proof for Lemma 6

Proof We prove the lemma recursively starting from the leaf nodes. For leaf nodes $n \in \mathcal{N}, \mathcal{C}(n) = \varnothing, Q_n^R(x) = \min_{z \in \mathcal{X}_{a(n)}} Q_n(z) + \sigma_n \psi_n(x - z) \geq \min_{z \in \mathcal{X}_{a(n)}} Q_n(z) + l_n \|x - z\|$. Since Q_n is l_n-Lipschitz continuous, $Q_n(z) \geq Q_n(x) - l_n \|x - z\|$. Therefore, $Q_n^R(x) \geq Q_n(x)$ and by Proposition 2, we know $Q_n^R(x) = Q_n(x)$ for all $x \in \mathcal{X}_{a(n)}$.
Now suppose for a node $n \in \mathcal{N}$, we know that all of its child nodes satisfy $Q_m^R(x) = Q_m(x), \forall x \in \mathcal{X}_n$, for all $m \in \mathcal{C}(n)$. Then by definition,

$$Q_n^R(x_{a(n)}) = \min_{(x,y)\in\mathcal{F}_n, z\in\mathcal{X}_{a(n)}} f_n(z, y, x) + \sigma_n \psi_n(x_{a(n)} - z) + Q_n^R(x).$$

By assumption, we know that $Q_n^R(x) = Q_n(x)$ for all $x \in \mathcal{X}_n$. Therefore, $Q_n^R(x_{a(n)}) = \min_{z \in \mathcal{X}_{a(n)}} Q_n(z) + \sigma_n \psi_n(x_{a(n)} - z) \geq \min_{z \in \mathcal{X}_{a(n)}} Q_n(z) + l_n \|x_{a(n)} - z\|$. Then

again by l_n-Lipschitz continuity of f_n, we conclude that $Q_n^R(x) = Q_n(x)$ for all $x \in \mathcal{X}_{a(n)}$. $\qquad\square$

A.4.2 Proof for Lemma 7

Proof We claim that if $K < \left(\frac{DL}{4\beta}\right)^d$, then there exists a point $\hat{x} \in \mathcal{X}$ such that $\|\hat{x} - w_k\| \geq \frac{2\beta}{L}$ for all $k = 1, \ldots, K$. We prove the claim by contradiction. Suppose such a point does not exist, or equivalently, for any point $x \in \mathcal{X}$, there exists $w_k \in \mathcal{W}$ such that $\|x - w_k\| < \frac{2\beta}{L}$. This implies that the balls $\mathcal{B}(2\beta/L, w_k)$ cover the set $\mathcal{X}$, which leads to

$$\mathrm{Vol}\mathcal{X} \leq \mathrm{Vol}\left(\bigcup_{k=1}^K \mathcal{B}(2\beta/L, w_k)\right) \leq \sum_{k=1}^K \mathrm{Vol}\mathcal{B}(2\beta/L, w_k) = K \cdot \mathrm{Vol}\mathcal{B}(2\beta/L).$$

Therefore, it must hold that $K \geq \mathrm{Vol}\mathcal{X}/\mathrm{Vol}\mathcal{B}(2\beta/L) = \left(\frac{DL}{4\beta}\right)^d$, hence a contradiction.

The existence of $\hat{x}$ guarantees that $f(w_k) - L\|\hat{x} - w_k\| \leq f(w_k) - 2\beta < 0$ for each $k = 1, 2, \ldots, K$. Therefore, $0 \leq \min_{x \in \mathcal{X}} \underline{Q}(x) \leq \underline{Q}(\hat{x}) = \max_{1 \leq k \leq K}\{0, f(w_k) - L\|\hat{x} - w_k\|\} = 0$. From compactness of $\mathcal{X}$ and the continuity of $\overline{Q}(x)$, we have the inequality $\min_{x \in \mathcal{X}} \overline{Q}(x) \geq \min_{1 \leq k \leq K} \overline{Q}(w_k) = \min_{1 \leq k \leq K} f(w_k) > \beta$, which completes the proof.

$\qquad\square$

A.4.3 Proof for Theorem 4

Proof Let us define the forward subproblem oracle $\mathcal{O}_n^F$ in iteration i and stage t as mapping $(x_{t-1}^i, \underline{Q}_{t+1}^{i-1})$ to an optimal solution (x_t^i, z_t^i) of the forward subproblem

$$\min_{x_t, z_t \in \mathcal{X}_t} \left\{ f_t(z_t) + L\|x_{t-1}^i - z_t\| + \underline{Q}_{t+1}^{i-1}(x_t) \right\},$$

and the backward subproblem oracle $\mathcal{O}_n^B$ in iteration i and stage t as mapping $(x_{t-1}^i, \underline{Q}_{t+1}^i)$ to an optimal solution $(\hat{x}_t^i, \hat{z}_t^i; \hat{\lambda}_t^i = 0, \hat{\rho}_t^i = L)$ of the backward subproblem

$$\max_{\substack{\lambda = 0 \\ 0 \leq \rho \leq L}} \min_{x_t, z_t \in \mathcal{X}_t} \left\{ f_t(z_t) + \rho\|x_{t-1}^i - z_t\| + \underline{Q}_{t+1}^i(x_t) \right\}. \tag{32}$$

Note that in the backward subproblem (32), we choose that $l_{n,\lambda} = 0$ and $l_{n,\rho} = L$. It is observed that the objective function in (32) is nondecreasing in ρ. Therefore, $\hat{\rho}_t^i = L$ is always an optimal solution for the outer maximization in (32). The root-node oracle $\mathcal{O}_r$ in iteration i simply solves $\min_{x_0 \in \mathcal{X}} \underline{Q}_1^i(x_0)$ and outputs x_0^{i+1}.

In the backward step (Algorithm 2, step 14) and c.f. the definition (15), the new generalized conjugacy cut in iteration $k \leq i$ is generated by

$$C_t^k(x \mid 0, L, \underline{v}_t^k) = \underline{v}_t^k - L\|x - x_{t-1}^k\| = \underline{v}_t^k - L\|x - x_{t-1}^k\|,$$

for node $t \geq 1$, where $\underline{v}_t^k$ is computed and upper bounded as

$$\underline{v}_t^k = f_t(\hat{z}_t^k) + L\|x_{t-1}^k - \hat{z}_t^k\| + \min_{x_t \in \mathcal{X}_t} \underline{Q}_{t+1}^k(x_t),$$
$$\leq f_t(x_{t-1}^k) + \min_{x_t \in \mathcal{X}_t} \underline{Q}_{t+1}^k(x_t),$$
$$\leq f_t(x_{t-1}^k) + \min_{x_t \in \mathcal{X}_t} \underline{Q}_{t+1}^i(x_t),$$

where the first inequality directly follows from (32), as $z = x_{t-1}^i$ is a feasible solution of the inner minimization problem, and the second inequality is due to the monotonicity $\underline{Q}_{t+1}^k(x) \leq \underline{Q}_{t+1}^i(x)$ for $k \leq i$. Therefore,

$$\underline{Q}_t^i(x) = \max_{k=1,\ldots,i} \left\{ 0, \ C_t^k(x \mid 0, L, \underline{v}_t^k) \right\}, \quad (\text{by } (14)),$$
$$= \max_{k=1,\ldots,i} \left\{ 0, \ \underline{v}_t^k - L\|x - x_{t-1}^k\| \right\},$$
$$\leq \max_{k=1,\ldots,i} \left\{ 0, \ f_t(x_{t-1}^k) - L\|x - x_{t-1}^k\| \right\} + \min_{x_t \in \mathcal{X}_t} \underline{Q}_{t+1}^i(x_t). \quad (33)$$

Similarly, by (18), the upper approximation of the value function is computed and lower bounded as

$$\bar{v}_t^k = f_t(z_t^k) + L\|z_t^k - x_{t-1}^k\| + \overline{Q}_{t+1}^k(x_t^k),$$
$$\geq f_t(x_{t-1}^k) + \overline{Q}_{t+1}^k(x_t^k),$$
$$\geq f_t(x_{t-1}^k) + \min_{x_t \in \mathcal{X}_t} \overline{Q}_{t+1}^k(x_t),$$
$$\geq f_t(x_{t-1}^k) + \min_{x_t \in \mathcal{X}_t} \overline{Q}_{t+1}^i(x_t),$$

where $\hat{\mathcal{X}}_t^k := \arg\min_{x_t \in \mathcal{X}_t} \underline{Q}_{t+1}^{k-1}(x_t)$. Therefore, the over-approximation satisfies

$$\overline{Q}_t^i(x) = \min_{k=1,\ldots,i} \left\{ \bar{v}_t^k + L\|x - x_{t-1}^k\| \right\}, \quad (\text{by } (17)),$$
$$\geq \min_{k=1,\ldots,i} \left\{ f_t(x_{t-1}^k) + L\left\|x - x_{t-1}^k\right\| \right\} + \min_{x_t \in \mathcal{X}_t} \overline{Q}_{t+1}^i(x_t),$$
$$> \frac{\varepsilon}{T} + \min_{x_t \in \mathcal{X}_t} \overline{Q}_{t+1}^i(x_t), \quad (34)$$

where (34) follows from the construction that $f_t(x) > \varepsilon/T$ for all $x \in \mathcal{X}_t$.

Now using (33) and (34), we can prove the statement of the theorem. Suppose the iteration index $i < \left(\frac{DLT}{4\varepsilon}\right)^d$. Denote $w_k := x_{t-1}^k$ for $k = 1, \ldots, i$. Since $\varepsilon/T < f_t(w_k) < 2\varepsilon/T$ by construction, applying Lemma 7, we get $\min_{x_t \in \mathcal{X}_t} \left\{ \max_{k=1,\ldots,i} \left\{ 0, f_t(x_{t-1}^k) - L\|x_t - x_{t-1}^k\| \right\} \right\} = 0$. By (33), $\min_{x_{t-1} \in \mathcal{X}_{t-1}} \underline{Q}_t^i$ $(x_{t-1}) \leq \min_{x_t \in \mathcal{X}_t} \underline{Q}_{t+1}^i(x_t)$ for $t = 1, \ldots, T$. Note at stage T, $\underline{Q}_{T+1}^i \equiv 0$. Therefore, $\min_{x_{t-1} \in \mathcal{X}_{t-1}} \underline{Q}_t^i(x_{t-1}) \leq 0$ for all $t = 1, \ldots, T$. But since $\underline{Q}_t^i(x) \geq 0$ for all $x \in \mathcal{X}_{t-1}$, we have $\min_{x \in \mathcal{X}_{t-1}} \underline{Q}_t^i(x) = 0$ for all $1 \leq t \leq T$. Hence we see that LOWERBOUND $= \min_{x_0 \in \mathcal{X}_0} \underline{Q}_1^i(x_0) = 0$ in iteration i.

Since $\mathcal{X}_t$ is a norm ball, it is compact. So by (34), we have

$$\min_{x_{t-1} \in \mathcal{X}_{t-1}} \overline{Q}_t^i(x_{t-1}) > \varepsilon/T + \min_{x_t \in \mathcal{X}_t} \overline{Q}_{t+1}^i(x_t), \quad \forall 1 \leq t \leq T.$$

This recursion implies that $\min_{x_0 \in \mathcal{X}_0} \overline{Q}_1^i(x_0) > T(\varepsilon/T) = \varepsilon$. According to Algorithm 2, Steps 18-19, we have that in iteration i, UPPERBOUND $= \min_{k=1,\ldots,i} \{\overline{Q}_1^k(x_0^{k+1})\} \geq \min_{k=1,\ldots,i} \{\overline{Q}_1^i(x_0^{k+1})\} \geq \min_{x_0 \in \mathcal{X}_0} \overline{Q}_1^i(x_0) > \varepsilon$. Combining the above analysis, we have UPPERBOUND $-$ LOWERBOUND $> \varepsilon$ in iteration i. Therefore, we conclude that if UPPERBOUND$-$LOWERBOUND $\leq \varepsilon$ at the i-th iteration, then we have $i \geq \left(\frac{DLT}{4\varepsilon}\right)^d$. □

A.4.4 Proof for Lemma 8

Proof Let v_d denote the d-volume for a d-dimensional unit ball. Recall that the d-volume of $\mathcal{S}^d(R)$ is given by $\text{Vol}_d(\mathcal{S}^d(R)) = (d+1)v_{d+1}R^d = \dfrac{(d+1)\pi^{(d+1)/2}}{\Gamma(\frac{d+1}{2}+1)}R^d$. We next estimate the d-volume for the spherical cap $\mathcal{S}_\beta^d(R, x)$. Let $\alpha \in (0, \pi/2)$ denote the central angle for the spherical cap, i.e., $\cos\alpha = 1 - \beta/R$. Since $\beta < (1 - \frac{\sqrt{2}}{2})R$, we know that $\alpha < \pi/4$. Then for any $x \in \mathcal{S}^d(R)$, the d-volume of the spherical cap can be calculated through

$$\text{Vol}_d(\mathcal{S}_\beta^d(R, x)) = \int_0^\alpha \text{Vol}_{d-1}(\mathcal{S}^{d-1}(R\sin\theta))Rd\theta = dv_d R^d \int_0^\alpha (\sin\theta)^{d-1}d\theta.$$

Note that when $\theta \in (0, \alpha)$, $\sin\theta > 0$ and $\cos\theta/\sin\theta > 1$. Therefore, since $d \geq 2$,

$$\text{Vol}_d(\mathcal{S}_\beta^d(R, x)) \leq dv_d R^d \int_0^\alpha (\sin\theta)^{d-1}\frac{\cos\theta}{\sin\theta}d\theta = dv_d R^d \cdot \frac{(\sin\alpha)^{d-1}}{d-1}.$$

By substituting $\sin\alpha = \sqrt{1 - (1 - \beta/R)^2}$, we have that

$$\frac{\text{Vol}_d(\mathcal{S}_\beta^d(R, x))}{\text{Vol}_d(\mathcal{S}^d(R))} \leq \frac{d}{d^2 - 1}\frac{v_d}{v_{d+1}}(\sin\alpha)^{d-1},$$

$$= \frac{d}{d^2 - 1} \frac{v_d}{v_{d+1}} \left(1 - \left(1 - \frac{\beta}{R} \right)^2 \right)^{(d-1)/2}$$

$$\leq \frac{d}{d^2 - 1} \frac{v_d}{v_{d+1}} \left(\frac{2\beta}{R} \right)^{(d-1)/2}.$$

Now suppose $\mathcal{W} = \{w_i\}_{k=1}^K$ is a maximal set satisfying the assumption, that is, for any $w \in \mathcal{S}^d(R)$, $w \notin \mathcal{W}$, there exists $w_k \in \mathcal{W}$ such that $w \in \mathcal{S}^d_\beta(R, w_k)$. Then, $\bigcup_{k=1}^K \mathcal{S}^d_\beta(R, w_k) \supseteq \mathcal{S}^d(R)$, therefore

$$\text{Vol}_d(\mathcal{S}^d(R)) \leq \sum_{k=1}^K \text{Vol}_d(\mathcal{S}^d_\beta(R, w_k)) = |\mathcal{W}| \text{Vol}_d(\mathcal{S}^d_\beta(R, w_1)).$$

Therefore we have

$$|\mathcal{W}| \geq \frac{\text{Vol}_d(\mathcal{S}^d(R))}{\text{Vol}_d(\mathcal{S}^d_\beta(R, w_1))} \geq \left[\frac{d}{d^2 - 1} \frac{v_d}{v_{d+1}} \left(\frac{2\beta}{R} \right)^{(d-1)/2} \right]^{-1}$$

$$= \frac{(d^2 - 1)\sqrt{\pi}}{d} \frac{\Gamma(d/2 + 1)}{\Gamma(d/2 + 3/2)} \left(\frac{R}{2\beta} \right)^{(d-1)/2}.$$

This completes the proof.   □

A.4.5 Proof for Lemma 9

Proof 1. By construction, F is a convex piecewise linear function. Since each linear piece has a Lipschitz constant $(L/R)\|w_k\| = L$, as $\|w_k\| = R$. Thus, F, as the maximum of these linear functions, is also an L-Lipschitz function.
2. Since $w_k \notin \mathcal{S}^d_{\varepsilon/L}(w_l)$ for $l \neq k$, $\langle w_l, w_k - w_l \rangle < -\varepsilon R/L$. Hence, $v_l + L/R \langle w_l, w_k - w_l \rangle < v_l - \varepsilon < 0$. Therefore, $F(w_k) = \max\{0, v_k\} = v_k$.
3. Notice that the above maximum for $F(w_k)$ is achieved at a unique linear piece, which implies that F is differentiable at w_k for all k. The gradient $\nabla F(w_k) = (L/R)w_k$. This gives the inequality in property 3 from the Proof of property 2.
4. The inequality of property 3 also implies that $\underline{Q}_l(w_l) = 0$. Now we show $\overline{Q}_l(w_l) > 3\varepsilon/2$. Since $w_l \notin \mathcal{S}^d_{\varepsilon/L}(w_k)$ for any $k \neq l$, then $\|w_l - w_k\| > \varepsilon/L$ by the Pythagorean theorem. Also $v_k > \varepsilon/2$. So $q_k := v_k + L\|w_l - w_k\| > 3\varepsilon/2$. Since $\overline{Q}_l(w_l)$ is the convex combination of q_k's, we have $\overline{Q}_l(w_l) > 3\varepsilon/2$. This completes the proof.   □

A.4.6 Proof for Theorem 5

Proof First we claim that in any iteration i, for any nodal problem k in stage t, the optimal solution in the forward step (Algorithm 2 line 7) must be $x_t^i = w_{t,k}$. To see this, recall that we set $l_{n,\lambda} = L_t$ and $l_{n,\rho} = 0$ for all $n \in \tilde{\mathcal{N}}(t)$, so by Proposition 5, the

under-approximation of the cost-to-go function $\underline{Q}_t^i(x)$ is L_{t+1}-Lipschitz continuous for all iteration $i \in \mathbb{N}$. So consider the forward step subproblem for node $n = (t, k)$ with $t \geq 2$ in iteration i

$$
\begin{aligned}
\min_{x_t \in \mathcal{X}} & \left\{ F_t(x_{t-1}^i) + L_t \left\| x_t - w_{t,k} \right\| + \underline{Q}_t^i(x_t) \right\} \\
&= F_t(x_{t-1}^i) + \min_{x_t \in \mathcal{X}} \left\{ L_t \left\| x_t - w_{t,k} \right\| + \underline{Q}_t^i(x_t) \right\}.
\end{aligned} \tag{35}
$$

Note that by the L_{t+1}-Lipschitz continuity of $\underline{Q}_t^i$,

$$
L_t \left\| x_t - w_{t,k} \right\| + \underline{Q}_t^i(x_t) \geq \underline{Q}_t^i(w_{t,k}) + (L_t - L_{t+1}) \left\| x_t - w_{t,k} \right\|,
$$

which, alongside with the fact that $L_{t+1} < L_t$, implies that $x_t = w_{t,k}$ is the unique optimal solution to the forward step problem (35). The above argument also works for any node in the stage $t = 1$ by simply removing the constant term $F_t(x_{t-1}^i)$ in the nodal problem (35).

Now we define over- and under-approximations of the value functions for the purpose of this proof. For node $n = (t, k)$, let

$$
\underline{Q}_{t,k}^i(x) := \max_{1 \leq j \leq i} \{0, C_{t,k}^j(x \mid \hat{\lambda}_{t,k}^j, 0, \underline{v}_{t,k}^j)\},
$$

and

$$
\overline{Q}_{t,k}^i(x) := \mathrm{conv}_{1 \leq j \leq i} \{\bar{v}_{t,k}^j + L_t \| x_{t-1}^j - x \| \},
$$

where for each j, by formula (16),

$$
\begin{aligned}
\underline{v}_{t,k}^j &:= \max_{\|\lambda\| \leq L_t} \min_{z, x \in \mathcal{X}} \left\{ F_{t-1}(z) + \langle \lambda, x_{t-1}^j - z \rangle + L_t \left\| x - w_{t,k} \right\| + \underline{Q}_t^j(x) \right\}, \\
&= \max_{\|\lambda\| \leq L_t} \min_{z \in \mathcal{X}} \left\{ F_{t-1}(z) + \langle \lambda, x_{t-1}^j - z \rangle \right\} + \min_{x \in \mathcal{X}} \left\{ L_t \left\| x - w_{t,k} \right\| + \underline{Q}_t^j(x) \right\} \\
&= F_{t-1}(x_{t-1}^j) + \underline{Q}_t^j(w_{t,k})
\end{aligned}
$$

with $-\hat{\lambda}_{t,k}^j \in \partial F_{t-1}(x_{t-1}^j)$ being an optimal solution to the outer maximization problem, and by formula (18)

$$
\begin{aligned}
\bar{v}_{t,k}^j &:= F_{t-1}(z_{t,k}^j) + L_t \| x_{t-1}^j - z_{t,k}^j \| + \overline{Q}_t^j(x_{t,k}^j) \\
&= F_{t-1}(x_{t-1}^j) + \overline{Q}_t^j(x_{t,k}^j).
\end{aligned}
$$

The last equalities of $\underline{v}_{t,k}^j$ and $\bar{v}_{t,k}^j$ are due to the L_t-Lipschitz continuity of F_{t-1}. So we have by the monotonicity of the under- and over-approximations that for each $j \leq i$,

$$
\underline{Q}_{t,k}^i(x) = \max_{1 \leq j \leq i} \{0, F_{t-1}(x_{t-1}^j) + \langle \hat{\lambda}_{t,k}^j, x_{t-1}^j - x \rangle + \underline{Q}_t^j(w_{t,k})\}
$$

$$\leq \max_{1 \leq j \leq i} \{0, F_{t-1}(x_{t-1}^j) + \langle \hat{\lambda}_{t,k}^j, x_{t-1}^j - x \rangle\} + \underline{Q}_t^i(w_{t,k}),$$

and

$$\overline{Q}_{t,k}^i(x) = \text{conv}_{1 \leq j \leq i} \left\{ F_{t-1}(x_{t-1}^j) + L_t \|x_{t-1}^j - x\| + \overline{Q}_t^j(w_{t,k}) \right\}$$

$$\geq \text{conv}_{1 \leq j \leq i} \left\{ F_{t-1}(x_{t-1}^j) + L_t \|x_{t-1}^j - x\| \right\} + \overline{Q}_t^i(w_{t,k}).$$

Therefore,

$$\overline{Q}_{t,k}^i(x) - \underline{Q}_{t,k}^i(x) \geq \overline{Q}_t^i(w_{t,k}) - \underline{Q}_t^i(w_{t,k})$$

$$+ \text{conv}_{1 \leq j \leq i} \{F_{t-1}(x_{t-1}^j) + L_t \|x_{t-1}^j - x\|\} - \max_{1 \leq j \leq i} \{0, F_{t-1}(x_{t-1}^j)$$

$$+ \langle \hat{\lambda}_{t,k}^j, x_{t-1}^j - x \rangle\}. \tag{36}$$

Thus we have $\overline{Q}_{t,k}^i(w_{t-1,k'}) - \underline{Q}_{t,k}^i(w_{t-1,k'}) \geq \overline{Q}_t^i(w_{t,k}) - \underline{Q}_t^i(w_{t,k})$ for any $k' = 1, \ldots, K_{t-1}$, since the last two terms on the right hand side of (36) are over- and under-approximations of the function F_{t-1}, respectively. Moreover, note that $x_{t-1}^j = w_{t-1,k'}$ for some $k' = 1, \ldots, K_{t-1}$ as it is the unique solution in the forward step. By Lemma 9, whenever the node $n' = (t - 1, k')$ is never sampled up to iteration i, we further have

$$\overline{Q}_{t,k}^i(w_{t-1,k'}) - \underline{Q}_{t,k}^i(w_{t-1,k'}) > \frac{3\varepsilon}{2(T - 1)} + \overline{Q}_t^i(w_{t,k}) - \underline{Q}_t^i(w_{t,k}).$$

Recall the definitions (14) and (17), for any $x \in \mathcal{X}$,

$$\underline{Q}_{t-1}^i(x) = \max_{1 \leq j \leq i} \left\{ 0, \frac{1}{K_t} \sum_{k=1}^{K_t} C_{t,k}^j(x \mid \hat{\lambda}_{t,k}^j, 0, \underline{v}_{t,k}^j) \right\} \leq \frac{1}{K_t} \sum_{k=1}^{K_t} \underline{Q}_{t,k}^i(x),$$

and

$$\overline{Q}_{t-1}^i(x) = \text{conv}_{1 \leq j \leq i} \left\{ \frac{1}{K_t} \sum_{k=1}^{K_t} (\bar{v}_{t,k}^j + L_t \|x_{t-1}^j - x\|) \right\} \geq \frac{1}{K_t} \sum_{k=1}^{K_t} \overline{Q}_{t,k}^i(x).$$

Consequently, for any $k' = 1, \ldots, K_{t-1}$,

$$\overline{Q}_{t-1}^i(w_{t-1,k'}) - \underline{Q}_{t-1}^i(w_{t-1,k'}) \geq \frac{1}{K_t} \sum_{k=1}^{K_t} [\overline{Q}_t^i(w_{t,k}) - \underline{Q}_t^i(w_{t,k})],$$

and in addition, for any node $n' = (t - 1, k')$ not sampled up to iteration i,

$$\overline{Q}_{t-1}^i(w_{t-1,k'}) - \underline{Q}_{t-1}^i(w_{t-1,k'}) > \frac{3\varepsilon}{2(T - 1)} + \frac{1}{K_t} \sum_{k=1}^{K_t} [\overline{Q}_t^i(w_{t,k}) - \underline{Q}_t^i(w_{t,k})].$$

Therefore, for any iteration index $i \leq \frac{1}{3}|\mathcal{W}_t|$, $t = 1, \ldots, T-1$, then there are $K_t - i \geq \frac{2}{3}|\mathcal{W}_t|$ nodes not sampled in stage t, which implies

$$\frac{1}{K_{t-1}} \sum_{k'=1}^{K_{t-1}} [\overline{\mathcal{Q}}_{t-1}^i(w_{t-1,k'}) - \underline{\mathcal{Q}}_{t-1}^i(w_{t-1,k'})] > \frac{\varepsilon}{T-1} + \frac{1}{K_t} \sum_{k=1}^{K_t} [\overline{\mathcal{Q}}_t^i(w_{t,k}) - \underline{\mathcal{Q}}_t^i(w_{t,k})].$$

Consequently, $\overline{\mathcal{Q}}_r^i - \underline{\mathcal{Q}}_r^i > (T-1) \cdot \dfrac{\varepsilon}{T-1} = \varepsilon$. Therefore, if UPPERBOUND $-$ LOWERBOUND $= \overline{\mathcal{Q}}_r^i - \underline{\mathcal{Q}}_r^i \leq \varepsilon$ in the iteration i, then

$$i > \frac{1}{3} \min_{t=1,\ldots,T-1} |\mathcal{W}_t| \geq \frac{1}{3} \frac{d(d-2)\sqrt{\pi}}{d-1} \frac{\Gamma(d/2+1/2)}{\Gamma(d/2+1)} \left(\frac{DL(T-1)}{8\varepsilon}\right)^{(d-2)/2}.$$

This completes the proof. □

B Problem classes with exact penalization

In this section, we discuss the problem classes that allows exact penalty reformulation, as stated in Assumption 2. A penalty function $\psi : \mathbb{R}^d \to \mathbb{R}_+$ is said to be sharp, if $\psi(x) \geq c \|x\|$ for all $x \in V \subset \mathbb{R}^d$, for some open neighborhood $V \ni 0$ and some positive scalar $c > 0$.

B.1 Convex problems with interior points

For convex problems, the Slater condition implies that the intersection of the domain $\mathrm{dom}(\sum_{n \in \mathcal{N}} f_n)$ and the feasible sets $\Pi_{n \in \mathcal{N}} \mathcal{F}_n$ has a non-empty interior. We have the following proposition.

Proposition 8 *Suppose the problem* (1) *is convex and satisfies the Slater condition For any sharp penalty functions* ψ_n, *there exist* $\sigma_n > 0$ *such that the penalty reformulation is exact.*

Proof Consider a perturbation vector $w = (w_n)_{n \in \mathcal{N}}$ such that $w_n \in \mathcal{X}_{a(n)} - \mathcal{X}_{a(n)}$ for each $n \in \mathcal{N}$, and define the perturbation function

$$\tau(w) := \min_{(z_n, x_n, y_n) \in \mathcal{X}_{a(n)} \times \mathcal{F}_n} \left\{ \sum_{n \in \mathcal{N}} p_n f_n(z_n, y_n, x_n) \,\middle|\, w_n = x_{a(n)} - z_n, \forall n \in \mathcal{N} \right\}.$$

The function τ is convex and $v^{\mathrm{prim}} = \tau(0)$ by definition. By the Slater condition, $0 \in \mathrm{int}(\mathrm{dom}(\tau))$ so there exists a vector $\lambda \in \mathbb{R}^{|\mathcal{N}|}$ such that $\tau(w) \geq \tau(0) + \langle \lambda, w \rangle$ for all perturbation w. Since ψ_n are sharp, there exist $\sigma_n > 0$ such that $\sum_{n \in \mathcal{N}} \sigma_n \psi_n(w_n) + \langle \lambda, w \rangle > 0$ for all $w \neq 0$. Consequently the penalty reformulation is exact since $v^{\mathrm{reg}} = \min_w \tau(w) + \sum_{n \in \mathcal{N}} \sigma_n \psi_n(w_n)$ and all optimal solutions must satisfy $w_n = x_{a(n)} - z_n = 0$ for all $n \in \mathcal{N}$. □

B.2 Problems with finite state spaces

We say a problem (2) has finite state spaces if the state spaces $\mathcal{X}_n$ are finite sets for all nodes n. Such problems appear in multistage integer programming [38], or when the original state spaces can be approximated through finite ones [19, 39]. The following proposition shows the penalty reformulation is exact whenever the state spaces are finite.

Proposition 9 *For any penalty functions ψ_n, $n \in \mathcal{N}$, if the state spaces are finite, then there exists a finite $\sigma_n > 0$ such that the penalty reformulation (7) is exact.*

Proof Let $d_n := \min_{x \neq z \in \mathcal{X}_{a(n)}} |\psi_n(x - z)|$ for each $n \in \mathcal{N}$. Since ψ_n is a penalty function and the state space $\mathcal{X}_n$ is finite, we know $d_n > 0$. Define c as

$$c := \min_{(z_n, y_n, x_n) \in \mathcal{X}_{a(n)} \times \mathcal{F}_n} \sum_{n \in \mathcal{N}} p_n f_n(z_n, y_n, x_n). \tag{37}$$

Since (37) is a relaxation of the original problem (1) by ignoring coupling constraint $z_n = x_{a(n)}$, then $c \leq v^{\text{prim}}$. We choose $\sigma_n = 1 + (v^{\text{prim}} - c)/(p_n d_n)$ for all $n \in \mathcal{N}$. Now let $(x_n, y_n, z_n)_{n \in \mathcal{N}}$ be an optimal solution to the regularized problem (4). Then if there exists $x_{a(m)} \neq z_m$ for some $m \neq r$, then $p_m \sigma_m \psi_m(x_{a(m)} - z_m) > v^{\text{prim}} - c$. Consequently,

$$v^{\text{reg}} \geq c + \sum_{n \in \mathcal{N}} p_n \sigma_n \psi_n(x_{a(n)} - z_n) \geq c + p_m \sigma_m \psi_m(x_{a(m)} - z_m) > c + v^{\text{prim}} - c$$
$$= v^{\text{prim}}.$$

This is a contradiction since $v^{\text{reg}} \leq v^{\text{prim}}$. Thus, any optimal solution to the reformulation (7) must have $x_{a(n)} = z_n$ for all $n \neq r$, which means the penalty reformulation is exact. □

B.3 Problems defined by mixed-integer linear functions

The problem (1) is said to be defined by mixed-integer linear functions, if all the feasible sets $\mathcal{F}_n$ and the epigraphs epi f_n are representable by mixed-integer variables and non-strict linear inequalities with rational coefficients. Recall that by Assumption 1, the primal problem is feasible, $v^{\text{prim}} > -\infty$. We have the following proposition on the exact penalty reformulation.

Proposition 10 ([13], Theorem 5) *If problem (1) is defined by mixed-integer linear functions and the penalty functions ψ_n are sharp for all $n \in \mathcal{N}$, then there exist $\sigma_n > 0$, such that the penalty reformulation is exact.*

B.4 Problems defined by C^1-functions

The problem (1) is said to be defined by C^1-functions if it is defined by functional constraints using indicator functions in each node $n \in \mathcal{N}$:

$$
f_n(x_{a(n)}, y_n, x_n) = \begin{cases} f_{n,0}(x_{a(n)}, y_n, x_n), & \text{if } g_{n,i}(x_{a(n)}, y_n, x_n) \leq 0, i = 1, \ldots, I_n, \\ +\infty & \text{otherwise.} \end{cases}
$$

with all $f_{n,0}, g_{n,i}, i = 1, \ldots, I_n$ being continuously differentiable. The Karush-Kuhn-Tucker condition at a feasible point $(x_n, y_n)_{n \in \mathcal{N}}$ of (1) says that there exist multipliers $\mu_{n,i} \geq 0, i = 1, \ldots, I_n$, such that

$$
\nabla_{x_n, y_n} \left\{ \sum_{n \in \mathcal{N}} (f_{n,0}(x_{a(n)}, y_n, x_n) - \mu_{n,i} g_{n,i}(x_{a(n)}, y_n, x_n)) \right\} = 0,
$$

$$
\mu_{n,i} g_{n,i}(x_{a(n)}, y_n, x_n) = 0, \quad i = 1, \ldots, I_n.
$$

We have the following proposition on the exactness.

Proposition 11 *Suppose the problem (1) is defined by C^1-functions and the Karush-Kuhn-Tucker condition holds for every local minimum solution of (1). If the penalty functions ψ_n are sharp for all $n \in \mathcal{N}$, then there exist $\sigma_n > 0$ such that the penalty reformulation is exact.*

We give the Proof of Proposition 11 below.

B.4.1 Proof for Proposition 11

We begin by stating a general exact penalization result for problems defined by C^1-functions. Consider the following perturbation function

$$
\begin{aligned}
p(u) := \min_{x \in \mathbb{R}^d} \quad & f(x, u) \\
\text{s.t.} \quad & g_i(x, u) \leq 0, \quad i = 1, \ldots, I, \\
& h_j(x, u) = 0, \quad j = 1, \ldots, J,
\end{aligned} \tag{38}
$$

Here u is the perturbation vector and $u = 0$ corresponds to the original primal problem. Let ψ be a penalty function on $\mathbb{R}^d$ and $\sigma > 0$ a penalty factor. A penalization of the original primal problem $p(0)$ is given by

$$
\begin{aligned}
\min_{x \in \mathbb{R}^d} \quad & f(x, u) + \sigma \psi(u) \\
\text{s.t.} \quad & g_i(x, u) \leq 0, \quad i = 1, \ldots, I, \\
& h_j(x, u) = 0, \quad j = 1, \ldots, J.
\end{aligned} \tag{39}
$$

Naturally we could impose some bound on the perturbation as $\|u\| \leq R_u$. We assume that f, g_i, h_j are continuously differentiable in x and u for all i, j. Moreover, the compactness in Assumption 1 implies that the feasible region prescribed by the inequality constraints $g_i(x, u) \leq 0$ are compact in x for any u, i.e., $X = \{x \in \mathbb{R}^d : \exists u, \|u\| \leq R_u, \text{s.t.} g_i(x, u) \leq 0, i = 1, \ldots, J\}$ is compact. For example, some of the inequalities are bounds on the variables, $\|x\|_\infty \leq 1$. We will show that there exists a penalty factor $\sigma > 0$ such that any optimal solution to (39) is feasible to (38). We next characterize the property of the perturbation function $p(u)$.

Lemma 10 *The perturbation function $p(u)$ is lower semicontinuous.*

Proof Let $X(u) \subset X$ denote the feasible set in x dependent on u. The minimum in the definition is well defined for every u due to the compactness of $X(u)$.

We show that $p(u)$ is lower semicontinuous (lsc) by showing $\liminf_{v \to u} p(v) \geq p(u)$ for any u. Assume for contradiction that for any $\varepsilon > 0$, there exists a sequence $\{v_k\}_{k=1}^\infty$ such that $v_k \to u$ and $p(v_k) \leq p(u) - \varepsilon$. Let $x_k \in \arg\min f(x, v_k)$ and thus $p(v_k) = f(x, v_k)$. Since X is compact, there exists a subsequence x_{k_j} and $z \in X$ such that $x_{k_j} \to z$ as $j \to \infty$. Then by continuity of f, $f(z, u) = \lim_{j \to \infty} f(x_{k_j}, v_{k_j}) \leq p(u) - \varepsilon$. This contradicts with the definition of $p(u)$, since $p(u) = \min_{x \in X(u)} f(x, u) \leq f(z, u) \leq p(u) - \varepsilon$. Therefore $p(u)$ is lsc. □

Now we give the theorem of exact penalization for problems defined by C^1-functions.

Proposition 12 *If the Karush-Kuhn-Tucker condition is satisfied at every local minimum solution of (38), then the penalty reformulation (39) is exact for some finite $\sigma > 0$.*

Proof Let $X(u)$ denote the feasible region of x defined by constraints $g_i(x, u) \leq 0$, $i = 1, \ldots, I$ and $h_j(x, u) = 0$, $j = 1, \ldots, J$. Then $X(u)$ is compact for any u by the continuity of the constraint functions. We show that for every optimal solution $x_0 \in X(0)$, there exists a neighborhood $V(x_0) \ni x_0$ in the x space, $U(x_0) \ni u = 0$ in the u space, and constant $L(x_0) > 0$, such that for all $x \in V(x_0)$ and $u \in U(x_0)$, we have $f(x, u) \geq f(x_0, 0) - L(x_0) \cdot \|u\|$. Then we use this fact together with compactness of $X(0)$ to show the existence of exact penalization. In this proof, the little-o is used to simplify notation, i.e., $o(\|a\|)$ denotes a function $b(a)$ such that $\lim_{a \to 0} |b(a)| / \|a\| = 0$.

Pick any optimal solution $x_0 \in X(0)$. By definition, it is also a local minimum solution. By hypothesis, the KKT condition is satisfied at x_0, that is, there exist $\lambda_i \in \mathbb{R}$, $i = 1, \ldots, I$, and $\mu_j \geq 0$, $j = 1, \ldots, J$ such that

$$\nabla_x f(x_0, 0) + \sum_{i=1}^I \lambda_i \nabla_x g_i(x_0, 0) + \sum_{j=1}^J \mu_j \nabla_x h_j(x_0, 0) = 0,$$

$$h_j(x_0, 0) = 0, \quad j = 1, \ldots, J,$$

$$g_i(x_0, 0) \leq 0, \quad \lambda_i \cdot g_i(x_0, 0) = 0, \quad i = 1, \ldots, I. \tag{40}$$

Since h_j's are continuously differentiable and $h_j(x_0, 0) = 0$, we have

$$\langle \nabla_x h_j(x_0, 0), x - x_0 \rangle + \langle \nabla_u h_j(x_0, 0), u \rangle$$
$$+ o(\|x - x_0\| + \|u\|) = 0, \ j = 1, \dots, J. \tag{41}$$

Let $A \subset I$ denote the set of active inequality constraints. Then similarly we have

$$\langle \nabla_x g_i(x_0, 0), x - x_0 \rangle + \langle \nabla_u g_i(x_0, 0), u \rangle + o(\|x - x_0\| + \|u\|) \leq 0, \ i \in A. \tag{42}$$

For any $i \notin A$, by the continuity of g_i, there exist neighborhoods W_i of x_0 and U_i' of $u = 0$ such that for any $(x, u) \in W_i \times U_i'$, $g_i(x, u) < 0$ remains inactive. Now, from (40), (41), (42), and f being continuously differentiable, we have

$$f(x, u) - f(x_0, 0)$$
$$= \langle \nabla_x f(x_0, 0), x - x_0 \rangle + \langle \nabla_u f(x_0, 0), u \rangle + o(\|x - x_0\| + \|u\|)$$
$$= \langle -\sum_{i \in A} \lambda_i \nabla_x g_i(x_0, 0) - \sum_{j=1}^{J} \mu_j \nabla_x h_j(x_0, 0), x - x_0 \rangle$$
$$+ \langle \nabla_u f(x_0, 0), u \rangle + o(\|x - x_0\| + \|u\|)$$
$$\geq \langle \nabla_u f(x_0, 0) + \sum_{i \in A} \lambda_i \nabla_u g_i(x_0, 0) + \sum_{j=1}^{J} \mu_j \nabla_u h_j(x_0, 0), u \rangle$$
$$+ o(\|x - x_0\| + \|u\|)$$
$$> -L(x_0) \cdot \|u\| + o(\|x - x_0\| + \|u\|),$$

where $L(x_0) := \|\nabla_u f(x_0, 0) + \sum_{i \in A} \lambda_i \nabla_u g_i(x_0, 0) + \sum_{j=1}^{J} \mu_j \nabla_u h_j(x_0, 0)\| + 1 > 0$. By the definition of the little-o notation, there exists a neighborhood $V(x_0) \subset \cap_{i \notin A} W_i$, $x_0 \in V(x_0)$ and $U(x_0) \subset \cap_{i \notin A} U_i$, $0 \in U(x_0)$ such that

$$f(x, u) - f(x_0, 0) \geq -L(x_0) \cdot \|u\|, \quad \forall (x, u) \in V(x_0) \times U(x_0).$$

Now, let $X_{\text{opt}}(0)$ denote the set of optimal solutions of x when $u = 0$. Note that $X_{\text{opt}}(0) \subset X(0)$ is closed due to the continuity of f, h_i, g_j, hence compact. The collection of open sets $\{V(x)\}_{x \in X_{\text{opt}}(0)}$ covers $X_{\text{opt}}(0)$. By compactness, there exists a finite subcollection $\{V(x_k)\}_{k=1}^{K}$ such that $X_{\text{opt}}(0) \subset \cup_{k=1}^{K} V(x_k) =: V$. Let $L := \max_{k=1,\dots,K} L(x_k)$ and $U = \cap_{k=1}^{K} U(x_k)$. Let f^* denote the optimal value for $u = 0$. Then we have

$$f(x, u) \geq f^* - L \cdot \|u\|, \quad \forall (x, u) \in V \times U.$$

To show the inequality for $x \notin V$, define $\tilde{p}(u) := \min_{x \in X(u) \setminus V} f(x, u)$. Note that $\tilde{p}(0) > f^*$ by the definition of $X_{\text{opt}}(0)$. Then by Lemma 10, $p(u)$ is lower semicontinuous, and we know that there exists a neighborhood U' of 0 such that $\tilde{p}(u) > f^*$ for all $u \in U'$. Therefore, for all $u \in U \cap U'$, we have $f(x, u) \geq f^* - L \cdot \|u\|$. Finally,

we can show that the penalization is exact. Since ψ is sharp, there exist an open set $\tilde{U} \subset U \cap U'$, and positive constants $c > 0$ such that

$$\psi(u) \geq c \, \|u\| \ \text{ on } \tilde{U}.$$

Let $M = \min_{u \in \bar{B}_{R_u}(0) \backslash \tilde{U}} \tilde{p}(u) > f^*$, $m = \min_{u \in \bar{B}_{R_u}(0) \backslash \tilde{U}} \psi(u) > 0$ because ψ is a penalty function. Let $\sigma = (M - f^*)/m + 1$. We have $f(x, u) \geq f^* - \sigma \cdot \|u\|$, $\quad \forall\, u \in \bar{B}_{R_u}(0) \backslash \{0\}$, $x \in \cup_u X(u)$. As a result, any optimal solution to the penalization (39) would satisfy $u = 0$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Note that our problem (7) can be written into the form (39) by letting $u = (x_{a(n)}, z_n)_{n \in \mathcal{N}}$, and including the duplicate constraints $z_n - x_{a(n)} = 0$ for any $n \neq r \in \mathcal{N}$ in the equality constraints $h_j(x, u) = 0$. And other constraints $g_i(x, u) \leq 0$ correspond to the functional constraints in the problem (1). Since ψ_n are sharp, the aggregate penalty function defined by $\psi(u) = \sum_{n \in \mathcal{N}} p_n \psi_n(x_{a(n)} - z_n)$ is also sharp. Let σ denote the penalty factor in Proposition 12. Proposition 11 follows from this by letting $\sigma_n = \sigma / p_n$ for all $n \in \mathcal{N}$.

References

1. Ahmed, S., Cabral, F.G., da Costa, B.F.P.: Stochastic Lipschitz dynamic programming Math. Program. **191**, 755–793 (2022)
2. Ahmed, S., King, A.J., Parija, G.: A Multi-stage Stochastic Integer Programming Approach for Capacity Expansion under Uncertainty. Journal of Global Optimization **26**, 3–24 (2003)
3. Baringo, L., Conejo, A.J.: Risk-Constrained Multi-Stage Wind Power Investment. IEEE Transactions on Power Systems **28**(1), 401–411 (2013) https://doi.org/10.1109/TPWRS.2012.2205411. http://ieeexplore.ieee.org/document/6247489/
4. Basciftci, B., Ahmed, S., Gebraeel, N.: Adaptive Two-stage Stochastic Programming with an Application to Capacity Expansion Planning. arXiv:1906.03513 [math] (2019)
5. Baucke, R., Downward, A., Zakeri, G.: A deterministic algorithm for solving multistage stochastic programming problems. Optim. Online p. 25 (2017)
6. Benders, J.F.: Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematik **4**, 238–252 (1962)
7. Birge, J.R.: Decomposition and Partitioning Methods for Multistage Stochastic Linear Programs. Oper. Res. **33**(5), 989–1007 (1985). https://doi.org/10.1287/opre.33.5.989
8. Bradley, S.P., Crane, D.B.: A Dynamic Model for Bond Portfolio Management. Manage. Sci. **19**(2), 139–151 (1972). https://doi.org/10.1287/mnsc.19.2.139
9. Chen, Z.L., Li, S., Tirupati, D.: A scenario-based stochastic programming approach for technology and capacity planning. Computers & Operations Research **29**(7), 781–806 (2002) https://doi.org/10.1016/S0305-0548(00)00076-9. http://linkinghub.elsevier.com/retrieve/pii/S0305054800000769
10. Chen, Z.L., Powell, W.B.: Convergent Cutting-Plane and Partial-Sampling Algorithm for Multistage Stochastic Linear Programs with Recourse. J. Optim. Theory Appl. **102**(3), 497–524 (1999). https://doi.org/10.1023/A:1022641805263
11. Dantzig, G.B., Wolfe, P.: Decomposition Principle for Linear Programs. Oper. Res. **8**(1), 101–111 (1960). https://doi.org/10.1287/opre.8.1.101
12. Escudero, L.F., Kamesam, P.V., King, A.J., Wets, R.J.B.: Production planning via scenario modelling. Annals of Operations Research **43**, 309–335 (1993)
13. Feizollahi, M.J., Ahmed, S., Sun, A.: Exact augmented lagrangian duality for mixed integer linear programming. Math. Program. **161**(1–2), 365–387 (2017)
14. Flach, B., Barroso, L., Pereira, M.: Long-term optimal allocation of hydro generation for a price-maker company in a competitive market: latest developments and a stochastic dual dynamic programming

approach. IET Generation, Transmission & Distribution **4**(2), 299 (2010). https://doi.org/10.1049/iet-gtd.2009.0107

15. Girardeau, P., Leclere, V., Philpott, A.B.: On the Convergence of Decomposition Methods for Multistage Stochastic Convex Programs. Math. Oper. Res. **40**(1), 130–145 (2015). https://doi.org/10.1287/moor.2014.0664

16. Glassey, C.R.: Nested Decomposition and Multi-Stage Linear Programs. Manage. Sci. **20**(3), 282–292 (1973). https://doi.org/10.1287/mnsc.20.3.282

17. Guigues, V.: Convergence analysis of sampling-based decomposition methods for risk-averse multistage stochastic convex programs. SIAM J. Optim. **26**(4), 2468–2494 (2016)

18. Guigues, V.: Dual dynamic programing with cut selection: Convergence proof and numerical experiments. Eur. J. Oper. Res. **258**(1), 47–57 (2017)

19. Hjelmeland, M.N., Zou, J., Helseth, A., Ahmed, S.: Nonconvex Medium-Term Hydropower Scheduling by Stochastic Dual Dynamic Integer Programming. IEEE Transactions on Sustainable Energy **10**(1), 481–490 (2019) https://doi.org/10.1109/TSTE.2018.2805164. http://ieeexplore.ieee.org/document/8289405/

20. Ho, J.K., Manne, A.S.: Nested decomposition for dynamic models. Math. Program. **6**(1), 121–140 (1974). https://doi.org/10.1007/BF01580231

21. Kusy, M.I., Ziemba, W.T.: A Bank Asset and Liability Management Model. Oper. Res. **34**(3), 356–376 (1986). https://doi.org/10.1287/opre.34.3.356

22. Lan, G.: Complexity of stochastic dual dynamic programming. Mathematical Programming pp. 1–38 (2020)

23. Lara, C.L., Mallapragada, D.S., Papageorgiou, D.J., Venkatesh, A., Grossmann, I.E.: Deterministic electric power infrastructure planning: Mixed-integer programming model and nested decomposition algorithm. European Journal of Operational Research **271**(3), 1037–1054 (2018) https://doi.org/10.1016/j.ejor.2018.05.039. http://linkinghub.elsevier.com/retrieve/pii/S0377221718304466

24. Linowsky, K., Philpott, A.B.: On the Convergence of Sampling-Based Decomposition Algorithms for Multistage Stochastic Programs. J. Optim. Theory Appl. **125**(2), 349–366 (2005). https://doi.org/10.1007/s10957-004-1842-z

25. Louveaux, F.V.: A Solution Method for Multistage Stochastic Programs with Recourse with Application to an Energy Investment Problem. Oper. Res. **28**(4), 889–902 (1980). https://doi.org/10.1287/opre.28.4.889

26. Mulvey, J.M., Vladimirou, H.: Stochastic Network Programming for Financial Planning Problems. Manage. Sci. **38**(11), 1642–1664 (1992). https://doi.org/10.1287/mnsc.38.11.1642

27. Nesterov, Y.: Lectures on Convex Optimization, vol. 137. Springer, Berlin (2018)

28. Pereira, M.V.F., Pinto, L.M.V.G.: Stochastic Optimization of a Multireservoir Hydroelectric System: A Decomposition Approach. Water Resour. Res. **21**(6), 779–792 (1985). https://doi.org/10.1029/WR021i006p00779

29. Pereira, M.V.F., Pinto, L.M.V.G.: Multi-stage stochastic optimization applied to energy planning. Math. Program. **52**(1–3), 359–375 (1991). https://doi.org/10.1007/BF01582895

30. Philpott, A., Guan, Z.: On the convergence of stochastic dual dynamic programming and related methods. Operations Research Letters **36**(4), 450–455 (2008) https://doi.org/10.1016/j.orl.2008.01.013. http://linkinghub.elsevier.com/retrieve/pii/S0167637708000308

31. Philpott, A., Wahid, F., Bonnans, F.: MIDAS: A Mixed Integer Dynamic Approximation Scheme. Mathematical Programming **181**, 19–50 (2020)

32. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis, vol. 317. Springer Science & Business Media, Berlin (2009)

33. Shapiro, A.: Analysis of stochastic dual dynamic programming method. European J. Oper. Res. **209**(1), 63–72 (2011) https://doi.org/10.1016/j.ejor.2010.08.007. http://linkinghub.elsevier.com/retrieve/pii/S0377221710005448

34. Shapiro, A., Tekaya, W., da Costa, J.P., Soares, M.P.: Risk neutral and risk averse Stochastic Dual Dynamic Programming method. European J. Oper. Res. **224**(2), 375–391 (2013) https://doi.org/10.1016/j.ejor.2012.08.022. http://linkinghub.elsevier.com/retrieve/pii/S0377221712006455

35. Slyke, R.M.V., Wets, R.: L-Shaped Linear Programs with Applications to Optimal Control and Stochastic Programming. SIAM J. Appl. Math. **17**(4), 638–663 (1969). http://www.jstor.org/stable/2099310

36. Takriti, S., Krasenbrink, B., Wu, L.S.Y.: Incorporating Fuel Constraints and Electricity Spot Prices into the Stochastic Unit Commitment Problem. Oper. Res. **48**(2), 268–280 (2000). https://doi.org/10.1287/opre.48.2.268.12379

37. Zou, J., Ahmed, S., Sun, X.A.: Partially Adaptive Stochastic Optimization for Electric Power Generation Expansion Planning. INFORMS J. Comput. **30**(2), 388–401 (2018). https://doi.org/10.1287/ijoc.2017.0782
38. Zou, J., Ahmed, S., Sun, X.A.: Stochastic dual dynamic integer programming. Math. Program. (2018). https://doi.org/10.1007/s10107-018-1249-5
39. Zou, J., Ahmed, S., Sun, X.A.: Multistage Stochastic Unit Commitment Using Stochastic Dual Dynamic Integer Programming. IEEE Trans. Power Syst. **34**(3), 1814–1823 (2019) https://doi.org/10.1109/TPWRS.2018.2880996. http://ieeexplore.ieee.org/document/8532315/