Robust Dual Dynamic Programming

Angelos Georghiou
Desautels Faculty of Management, McGill University, Canada, angelos.georghiou@mcgill.ca

Angelos Tsoukalas
Olayan School of Business, American University of Beirut, Lebanon, angelos.tsoukalas@aub.edu.lb

Wolfram Wiesemann
Imperial College Business School, Imperial College London, United Kingdom, ww@imperial.ac.uk

Multi-stage robust optimization problems, where the decision maker can dynamically react to consecutively observed realizations of the uncertain problem parameters, pose formidable theoretical and computational challenges. As a result, the existing solution approaches for this problem class typically determine suboptimal solutions under restrictive assumptions. In this paper, we propose a *robust dual dynamic programming* (RDDP) scheme for multi-stage robust optimization problems. The RDDP scheme takes advantage of the decomposable nature of these problems by bounding the costs arising in the future stages through lower and upper cost to-go functions. For problems with uncertain technology matrices and/or constraint right-hand sides, our RDDP scheme determines an optimal solution in finite time. If also the objective function and/or the recourse matrices are uncertain, our method converges asymptotically (but deterministically) to an optimal solution. Our RDDP scheme does not require a relatively complete recourse, and it offers deterministic upper and lower bounds throughout the execution of the algorithm. We demonstrate the promising performance of our algorithm in a stylized inventory management problem.

Key words: robust optimization, multi-stage problems, dual dynamic programming, error bounds

1. Introduction

In this paper, we study *multi-stage robust optimization problems* of the form

$$
\begin{array}{lll}
\text{minimize} & \displaystyle\max_{\boldsymbol{\xi}\in\Xi} \sum_{t=1}^{T} \boldsymbol{q}_t(\boldsymbol{\xi}_t)^\top \boldsymbol{x}_t(\boldsymbol{\xi}^t) & \\
\text{subject to} & \boldsymbol{W}_1(\boldsymbol{\xi}_1)\,\boldsymbol{x}_1(\boldsymbol{\xi}^1) \geq \boldsymbol{h}_1(\boldsymbol{\xi}_1) & \forall \boldsymbol{\xi}\in\Xi \\
& \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1}(\boldsymbol{\xi}^{t-1}) + \boldsymbol{W}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_t(\boldsymbol{\xi}^t) \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) & \forall \boldsymbol{\xi}\in\Xi,\ \forall t = 2,\ldots,T \\
& \boldsymbol{x}_t(\boldsymbol{\xi}^t) \in \mathbb{R}^{n_t},\ \boldsymbol{\xi}\in\Xi \text{ and } t = 1,\ldots,T, &
\end{array}
\tag{1}
$$

where the parameters $\boldsymbol{\xi}_t$ are revealed at the beginning of stage t, $t = 1,\ldots,T$, and the decision $\boldsymbol{x}_t$ is taken afterwards under the full knowledge of $\boldsymbol{\xi}^t = (\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_t)$. A solution to problem (1) is immunized against all parameter realizations $\boldsymbol{\xi} = (\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_T)$ in the uncertainty set Ξ, which we assume to be stage-wise rectangular. The cost vectors $\boldsymbol{q}_t$, the technology matrices $\boldsymbol{T}_t$, the recourse matrices $\boldsymbol{W}_t$ and the right-hand side vectors $\boldsymbol{h}_t$ may depend affinely on $\boldsymbol{\xi}_t$. We assume that $\boldsymbol{\xi}_1$ is deterministic, and hence $\boldsymbol{x}_1$ is a here-and-now decision. Note that we do *not* assume that problem (1) has a relatively complete recourse, that is, there may be partial solutions $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_t$ satisfying the constraints up to time period t that cannot be extended to complete solutions $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_T$ satisfying all constraints up to time period T.

Remark 1. Problem (1) can accommodate constraints that link decisions of more than two time periods by augmenting $\boldsymbol{x}_t$ appropriately. In fact, we can model generic constraints of the form

$$\sum_{s=1}^{t-1} \boldsymbol{T}'_{t,s}(\boldsymbol{\xi}_t)\, \boldsymbol{x}_s(\boldsymbol{\xi}^s) + \boldsymbol{W}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_t(\boldsymbol{\xi}^t) \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \qquad \forall \boldsymbol{\xi} \in \Xi,\ \forall t = 2, \ldots, T$$

by replacing the decision vectors $\boldsymbol{x}_t \in \mathbb{R}^{n_t}$ with their augmented counterparts $\boldsymbol{x}'_t \in \mathbb{R}^{n^t}$, $n^t = n_1 + \ldots + n_t$, whose components satisfy $x'_{t,i}(\boldsymbol{\xi}^t) = x'_{t-1,i}(\boldsymbol{\xi}^{t-1})$, $i = 1, \ldots, n^{t-1}$, as well as setting $\boldsymbol{T} = (\boldsymbol{T}'_{t,1}, \ldots, \boldsymbol{T}'_{t,t-1})$. The resulting formulation can be readily brought into the form of problem (1).

The multi-stage robust optimization problem (1) is convex, but it involves infinitely many decision variables and constraints. In fact, problem (1) is NP-hard even for $T = 2$ time stages (Guslitser 2002), and polynomial-time solvable subclasses of problem (1) are both rare and restrictive (Anderson and Moore 1990, Bertsimas et al. 2010, Gounaris et al. 2013). As a result, any solution scheme for problem (1) has to trade off the competing goals of optimality and computational tractability.

Tractable conservative approximations to problem (1) can be obtained by restricting the decisions $\boldsymbol{x}_t$ to functions with pre-selected structure, which are called decision rules. Popular classes of decision rules include affine (Ben-Tal et al. 2004, Kuhn et al. 2011), segregated affine (Chen et al. 2008, Chen and Zhang 2009, Goh and Sim 2010), piecewise affine (Georghiou et al. 2015) and polynomial as well as trigonometric functions (Bertsimas et al. 2011) of the parameters $\boldsymbol{\xi}^t$. Decision rule approximations typically scale polynomially with the size of the problem. However, they require the solution of monolithic and often dense optimization problems, and they may thus still be difficult to solve in practice. Moreover, obtaining good estimates of the incurred suboptimality can be challenging (Kuhn et al. 2011, Hadjiyiannis et al. 2011). Decision rules have recently also been used in conjuction with the Fourier-Motzkin elimination, see Zhen et al. (2016). For a survey of the decision rule literature, we refer to Delage and Iancu (2015).

Despite the aforementioned pessimistic complexity result, instances of problem (1) with $T = 2$ stages have been solved successfully via Benders' decomposition. These methods split problem (1) into a convex master problem involving the first-stage decisions and a non-convex subproblem that provides cuts for the worst-case second-stage costs. The subproblems can be solved to global optimality using mixed-integer linear programming (Jiang et al. 2010, Thiele et al. 2010, Zhao et al. 2013), dynamic programming or vertex enumeration (Jiang et al. 2010, Thiele et al. 2010). Alternatively, the subproblems can be solved to local optimality via an outer approximation algorithm (Bertsimas et al. 2013). In a similar spirit, Zeng and Zhao (2013) and Ayoub and Poss (2016) use semi-infinite programming techniques to solve the two-stage variant of problem (1) to optimality. Here, the convex master problem is a relaxation of problem (1) that involves a finite subset of the realizations $\boldsymbol{\xi} \in \Xi$, and the non-convex subproblem identifies realizations that should be added to the master problem. Again, the subproblems can be solved to global optimality or local optimality. All of the aforementioned Benders' decomposition and semi-infinite programming approaches provide upper and lower bounds in each iteration. Moreover, if the subproblems are solved to global optimality, then the approaches converge to the optimal solution in finite time.

While instances of problem (1) with $T > 2$ stages could in principle be solved in finite time by an adaptation of the nested Benders' decomposition for stochastic programs (Louveaux 1980, Birge 1985), the curse of dimensionality incurred by the construction and processing of exponentially large scenario trees would limit such a scheme to problems with a small number of time stages. Instead, multi-stage instances of problem (1) have to date been solved by adaptations of semi-infinite programming techniques. Bertsimas and Georghiou (2015) solve a variant of problem (1) which is restricted to piecewise affine decisions rules $\boldsymbol{x}_t$ with a pre-specified number of pieces. In their approach, the non-convex master problem determines the best decision within the selected function class over a finite subset of the realizations $\boldsymbol{\xi} \in \Xi$, and the non-convex subproblem identifies the realizations to be added to the master problem. Bertsimas and Dunning (2016) and Postek and den Hertog (2016) propose to solve problem (1) by iteratively partitioning the uncertainty set Ξ into smaller subsets. Here, the convex master problem determines constant or affine decision rules $\boldsymbol{x}_t$ for each set of the partition, and the subproblem identifies a refined partition for the master problem. While the approaches of Bertsimas and Georghiou (2015), Bertsimas and Dunning (2016) and Postek and den Hertog (2016) can incorporate integer decisions in every stage, no finite time convergence has been established for either of them.

In this paper, we propose a *robust dual dynamic programming* (RDDP) scheme for problem (1). Similar to the popular stochastic dual dynamic programming (SDDP) method for stochastic programs (Pereira and Pinto 1991, Shapiro 2011), our algorithm decouples problem (1) into T two-stage subproblems which approximate the costs arising in future stages through bounds on the cost to-go functions. Contrary to the SDDP scheme, however, our algorithm maintains *both* lower *and* upper bounds on these cost to-go functions. Our algorithm iteratively solves the T subproblems in sequences of forward and backward passes. Each forward pass determines an approximate worst-case realization of the parameters $\boldsymbol{\xi}$ and the associated recourse decisions $\boldsymbol{x}_1(\boldsymbol{\xi}^1), \ldots, \boldsymbol{x}_T(\boldsymbol{\xi}^T)$. While both SDDP and our RDDP scheme select the recourse decisions using the lower cost to-go bounds, our RDDP scheme selects the approximate worst-case parameter realizations using the upper cost to-go bounds, as opposed to randomly sampling them as in SDDP. This turns out to be crucial for the correctness of our algorithm, and it enables us to equip our scheme with deterministic convergence guarantees, in contrast to the probabilistic convergence guarantees offered by SDDP. The subsequent backward pass refines the upper and lower cost to-go approximations along the selected decisions. For the case where only the technology matrices and/or the constraint right-hand sides in (1) are uncertain, the subproblems of our RDDP scheme maximize a convex function over a convex set and can therefore be solved as mixed-integer programs or via vertex enumeration. For the case where the objective function and/or the recourse matrices in (1) are uncertain, the subproblems constitute bilinear programs.

The contributions of this paper may be summarized as follows.

1. We develop an RDDP scheme for problem (1). In contrast to the existing solution approaches for problem (1), our method accommodates for uncertainty in all problem components (objective function, technology and recourse matrices as well as right-hand sides), it does not require a relatively complete recourse, it offers deterministic upper and lower bounds on the optimal value of (1) in each iteration, and it comes with deterministic convergence guarantees.
2. For instances of problem (1) with uncertain technology matrices and/or constraint right-hand sides as well as polyhedral, stage-wise rectangular uncertainty sets Ξ_t, our scheme finds an

optimal solution in finitely many iterations. To our best knowledge, none of the previously proposed solution approaches for problem (1)—apart from the nested Benders' decomposition, which is limited to small problems—offers such a guarantee for $T > 2$ stages.

3. We show that our RDDP scheme extends to instances of problem (1) with uncertain objective functions and recourse matrices. We argue that in this case, no method relying on piecewise affine approximations of the cost to-go functions can converge in finite time. Nevertheless, our RDDP scheme converges asymptotically to an optimal solution in a deterministic fashion.
4. We present numerical results which indicate that the attractive theoretical properties of our RDDP scheme are accommodated by a good practical performance in a stylized inventory management case study.

The remainder of the paper is structured as follows. We formulate and discuss the optimization problem of interest in Section 2. Section 3 develops and analyzes our RDDP scheme. Section 4 compares our algorithm with robust versions of the nested Benders' decomposition and SDDP. We present numerical results in Section 5, and we offer concluding remarks in Section 6. For ease of exposition, the main paper focuses on instances of problem (1) with a relatively complete recourse and deterministic objective functions and recourse matrices. The extension of our algorithm to problems without a relatively complete recourse and uncertain objective functions and recourse matrices, as well as some auxiliary proofs, are relegated to the e-companion.

Notation. The *conjugate* (or *Legendre-Fenchel transformation*) of an extended real-valued function $f : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\}$ is defined by $f^\star(\boldsymbol{y}) = \sup\{\boldsymbol{y}^\top \boldsymbol{x} - f(\boldsymbol{x}) \, : \, \boldsymbol{x} \in \mathbb{R}^n\}$. In particular, the conjugate of the *indicator function* $\delta_{\mathcal{X}} : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\}$ defined through $\delta_{\mathcal{X}}(\boldsymbol{x}) = 0$ if $\boldsymbol{x} \in \mathcal{X}$ and $\delta_{\mathcal{X}}(\boldsymbol{x}) = \infty$ otherwise, is the *support function* $\sigma_{\mathcal{X}} : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\}$ defined through $\sigma_{\mathcal{X}}(\boldsymbol{y}) = \sup\{\boldsymbol{y}^\top \boldsymbol{x} \, : \, \boldsymbol{x} \in \mathcal{X}\}$. For an extended real-valued function $f : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\}$, the *lower convex envelope* $\operatorname{env} f$ is the pointwise largest convex function that is weakly below f. We denote the vector of all ones by $\mathbf{e}$, and its dimension will be clear from the context.

2. Problem Formulation

We first list and discuss the assumptions that we make throughout the main paper. Afterwards, we present an equivalent nested formulation of the multi-stage robust optimization problem (1) that forms the basis of our RDDP scheme.

2.1. Assumptions

The main paper studies problem (1) under the following additional assumptions:

(A1) Boundedness. The feasible region of problem (1) is bounded, that is, any feasible solution satisfies $\|\boldsymbol{x}_t(\boldsymbol{\xi}^t)\| \leq M$ for all $\boldsymbol{\xi} \in \Xi$ and $t = 1, \ldots, T$, where M is a sufficiently large number.

(A2) Relatively Complete Recourse. Problem (1) has a relatively complete recourse, that is, any partial solution $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t$ satisfying the constraints up to time period t can be extended to a complete solution $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ that satisfies all constraints for every realization $\boldsymbol{\xi} \in \Xi$.

(A3) Uncertainty Set. The uncertainty set is bounded, polyhedral and stage-wise rectangular, that is, $\Xi = \bigtimes_{t=1}^{T} \Xi_t$ for bounded sets $\Xi_t = \{\boldsymbol{\xi}_t \in \mathbb{R}^{k_t} : \boldsymbol{F}_t \boldsymbol{\xi}_t \leq \boldsymbol{g}_t\}$, $\boldsymbol{F}_t \in \mathbb{R}^{l_t \times k_t}$ and $\boldsymbol{g}_t \in \mathbb{R}^{l_t}$.

(A4) Uncertain Parameters. The objective coefficients $\boldsymbol{q}_t \in \mathbb{R}^{n_t}$ and the recourse matrices $\boldsymbol{W}_t \in \mathbb{R}^{m_t \times n_t}$ are deterministic, while $\boldsymbol{T}_t : \Xi_t \mapsto \mathbb{R}^{m_t \times n_{t-1}}$ and $\boldsymbol{h}_t : \Xi_t \mapsto \mathbb{R}^{m_t}$ are affine in $\boldsymbol{\xi}_t$.

The assumptions (A1) and (A2) merely serve to simplify the exposition. It is straightforward to extend our RDDP scheme to unbounded feasible regions, but the extension introduces cumbersome case distinctions to cater for rays in the feasible region. The extension to multi-stage robust optimization problems without a relatively complete recourse is relegated to the e-companion.

Assuming a bounded polyhedral uncertainty set in (A3) allows us to solve the non-convex subproblems of our RDDP scheme as mixed-integer linear programs or via vertex enumeration, and it is essential to guarantee the finite time convergence of our RDDP scheme. The e-companion shows, however, that our RDDP scheme still converges asymptotically for generic non-convex uncertainty sets. Stage-wise rectangularity ressembles the stage-wise independence assumption in SDDP schemes, and it essentially precludes the existence of an 'intertemporal budget' for unfavourable outcomes. Stage-wise rectangularity is required since our RDDP scheme operates on a nested formulation of problem (1) where the future costs incurred in stages $t+1, \ldots, T$ must only depend on $\boldsymbol{x}_t$ and not on the realized values of $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_t$. Nevertheless, the assumption of stage-wise rectangularity can be relaxed for the constraint right-hand sides $\boldsymbol{h}_t$.

Remark 2 (Autoregressive Right-Hand Sides). Assume that the constraint right-hand sides in problem (1) follow p-th order vector autoregressive processes of the form $\boldsymbol{h}_t(\boldsymbol{\xi}^t) = \boldsymbol{h}_t^0 + \sum_{\tau=t-p}^{t} \boldsymbol{H}_{t\tau} \boldsymbol{\xi}_\tau$. By augmenting the decisions $\boldsymbol{x}_t$ to $\boldsymbol{x}'_t = (\boldsymbol{x}_t, \boldsymbol{y}_{t,t-p}, \ldots, \boldsymbol{y}_{t,t-1})$ with $\boldsymbol{y}_{t,\tau}(\boldsymbol{\xi}^t) = \boldsymbol{\xi}_\tau$,

$\tau = t-p,\ldots,t-1$, we can recover an instance of problem (1) that satisfies the assumption (A3) if we replace the constraint right-hand sides $\boldsymbol{h}_t$ with $\boldsymbol{h}_t^0 + \sum_{\tau=t-p}^{t-1} \boldsymbol{H}_{t\tau}\boldsymbol{y}_{t,\tau}(\boldsymbol{\xi}^t) + \boldsymbol{H}_{tt}\boldsymbol{\xi}_t$.

The assumption (A4) precludes uncertain objective functions and recourse matrices. While the e-companion shows that our RDDP scheme remains valid for generic multi-stage robust optimization problems with uncertain objective functions and recourse matrices, finite convergence can no longer be guaranteed, and the subproblems of our RDDP scheme become bilinear programs that can no longer be solved as mixed-integer linear programs or via vertex enumeration. We remark, however, that a restricted form of objective uncertainty can be modeled despite the assumption (A4).

Remark 3 (Lag-1 Uncertain Objective Functions). Consider a variant of the multi-stage robust optimization problem (1) with a lag-1 objective of the form

$$\text{minimize} \qquad \max_{\boldsymbol{\xi}\in\Xi} \sum_{t=1}^{T-1} \boldsymbol{q}_{t+1}(\boldsymbol{\xi}_{t+1})^\top \boldsymbol{x}_t(\boldsymbol{\xi}^t).$$

We can recover an instance of problem (1) that satisfies the assumption (A4) if we augment the decisions $\boldsymbol{x}_t$ to $\boldsymbol{x}_t' = (\boldsymbol{x}_t, y_t)$, $t=2,\ldots,T$, replace the objective function with $\sum_{t=2}^{T} y_t(\boldsymbol{\xi}^t)$ and add the additional constraints $y_t(\boldsymbol{\xi}^t) \geq \boldsymbol{q}_t(\boldsymbol{\xi}_t)^\top \boldsymbol{x}_{t-1}(\boldsymbol{\xi}^{t-1})$ for all $t=2,\ldots,T$.

2.2. Nested Problem Formulation

Under the assumptions (A3) and (A4), problem (1) has the equivalent nested formulation

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{q}_1^\top \boldsymbol{x}_1 + \mathcal{Q}_2(\boldsymbol{x}_1) \\ \text{subject to} \quad & \boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1 \\ & \boldsymbol{x}_1 \in \mathbb{R}^{n_1}, \end{aligned} \tag{1'}$$

where we omit the dependence of $\boldsymbol{W}_1$ and $\boldsymbol{h}_1$ on the deterministic parameters $\boldsymbol{\xi}_1$, and where the *stage-t worst-case cost to-go function* $\mathcal{Q}_t$ is defined as $\mathcal{Q}_t(\boldsymbol{x}_{t-1}) = \max\{Q_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t) : \boldsymbol{\xi}_t \in \Xi_t\}$ with

$$Q_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t) \;=\; \left[\begin{array}{ll} \text{minimize} & \boldsymbol{q}_t^\top \boldsymbol{x}_t + \mathcal{Q}_{t+1}(\boldsymbol{x}_t) \\ \text{subject to} & \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1} + \boldsymbol{W}_t\boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \\ & \boldsymbol{x}_t \in \mathbb{R}^{n_t} \end{array}\right], \qquad t = 2,\ldots,T,$$

and $\mathcal{Q}_{T+1}(\boldsymbol{x}_T) = 0$ for all $\boldsymbol{x}_T \in \mathbb{R}^{n_T}$. In the following, we refer to Q_t as the *stage-t nominal cost to-go problem.* In general, this problem can evaluate to $+\infty$ or $-\infty$ if it is infeasible or unbounded, respectively. The assumptions (A1)–(A3) preclude both of these cases, however, and we thus conclude that Q_t always attains finite values on the real line.

Standard arguments show that the stage-t worst-case cost to-go function $\mathcal{Q}_t$ is convex and piecewise affine for $t = 2, \ldots, T$ (Birge and Louveaux 1997, Theorem 40). However, the stage-t nominal cost to-go problem $Q_t(\boldsymbol{x}_{t-1}; \cdot)$ is convex in the realization $\boldsymbol{\xi}_t$ of the uncertain parameters, which renders the maximization over $\boldsymbol{\xi}_t$ in $\mathcal{Q}_t$ difficult. We now formalize this intuition.

Theorem 1. *Determining the stage-t worst-case costs $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ is strongly NP-hard if the technology matrix $\boldsymbol{T}_t$ or the right-hand side vector $\boldsymbol{h}_t$ is uncertain, even if $\mathcal{Q}_{t+1}(\boldsymbol{x}_t) = 0$ for all $\boldsymbol{x}_t \in \mathbb{R}^{n_t}$.*

Proof. Deferred to Section EC.1 in the e-companion. □

Remark 4 (State and Control Variables). In many applications, the worst-case costs $\mathcal{Q}_{t+1}$ only depend on a few components of the decision vector $\boldsymbol{x}_t$ which are termed *state variables* (Bertsekas 2007). In problem (1), the state variables in stage t correspond to the non-zero columns of the technology matrix $\boldsymbol{T}_{t+1}$ in stage $t+1$. The remaining components of $\boldsymbol{x}_t$ represent *control variables* which are required to determine the optimal course of action within stage t, but which have no (direct) bearing on the future costs quantified by $\mathcal{Q}_{t+1}$. In an inventory control problem, for example, the state variables may record the inventory levels of each product, whereas additional control variables could be needed to determine the order quantities as well as how much of each product is sold at what price. Similar to SDDP, the runtime of our RDDP scheme can be reduced by only carrying over the state variables to the next time stage. We will revisit this point in the numerical examples in Section 5. In the interest of a concise notation, however, we notationally suppress the distinction between state and control variables in the remainder of the paper.

3. Robust Dual Dynamic Programming

If we knew the stage-$(t+1)$ worst-case cost to-go function $\mathcal{Q}_{t+1}$, we could evaluate $\mathcal{Q}_t$ at any specific point $\boldsymbol{x}_{t-1}$ by solving a max-min problem. Note, however, that $\mathcal{Q}_{t+1}$ is itself unknown for $t < T$. In the following, we iteratively construct lower and upper bounds $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ on each worst-case cost to-go function $\mathcal{Q}_t$, $t = 2, \ldots, T$, that satisfy $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \leq \mathcal{Q}_t(\boldsymbol{x}_{t-1}) \leq \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1})$ for all $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$. Given an upper bound $\overline{\mathcal{Q}}_{t+1}$ and a point $\boldsymbol{x}_{t-1}$, we can bound $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ from above by solving

$$\begin{aligned} \overline{Q}_t(\boldsymbol{x}_{t-1}) \;=\; \text{maximize} \quad & \left[\begin{array}{ll} \text{minimize} & \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \\ \text{subject to} & \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \\ & \boldsymbol{x}_t \in \mathbb{R}^{n_t} \end{array}\right] \\ \text{subject to} \quad & \boldsymbol{\xi}_t \in \Xi_t, \end{aligned} \tag{2}$$

which amounts to a max-min problem. The optimal value of this problem coincides with $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ if $\overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) = \mathcal{Q}_{t+1}(\boldsymbol{x}_t)$ for all $\boldsymbol{x}_t \in \mathbb{R}^{n_t}$. Since $\overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \geq \mathcal{Q}_{t+1}(\boldsymbol{x}_t)$ by assumption, $\overline{Q}_t(\boldsymbol{x}_{t-1})$ bounds $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ from above. We will show in Section 3.2 that problem (2) can be solved as a mixed-integer linear program or via vertex enumeration if the upper bound $\overline{\mathcal{Q}}_{t+1}$ is convex and piecewise affine.

Similarly, we can bound $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ from below by fixing any realization $\boldsymbol{\xi}_t \in \Xi_t$ and solving

$$\underline{Q}_t(\boldsymbol{x}_{t-1}; \boldsymbol{\xi}_t) \;=\; \left[\begin{array}{ll} \text{minimize} & \boldsymbol{q}_t^\top \boldsymbol{x}_t + \underline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \\ \text{subject to} & \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \\ & \boldsymbol{x}_t \in \mathbb{R}^{n_t} \end{array}\right].$$

This problem bounds $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ from below since it only considers a single realization $\boldsymbol{\xi}_t \in \Xi_t$ and employs the lower bound $\underline{\mathcal{Q}}_{t+1}$ on the stage-$(t+1)$ worst-case cost to-go function $\mathcal{Q}_{t+1}$. Note that contrary to the upper bound problem $\overline{Q}_t(\boldsymbol{x}_{t-1})$, which maximizes over all possible stage-t realizations $\boldsymbol{\xi}_t \in \Xi_t$, the lower bound problem $\underline{Q}_t(\boldsymbol{x}_{t-1}; \boldsymbol{\xi}_t)$ is solved for a particular realization $\boldsymbol{\xi}_t$ and hence depends on both $\boldsymbol{x}_{t-1}$ and $\boldsymbol{\xi}_t$. For a convex and piecewise affine lower bound $\underline{\mathcal{Q}}_{t+1}$, $\underline{Q}_t(\boldsymbol{x}_{t-1}; \boldsymbol{\xi}_t)$ can be readily cast as a linear program. In the next subsection, we will bound the stage-t worst-stage cost to-go function $\mathcal{Q}_t$ from below through supporting hyperplanes of $\underline{Q}_t(\cdot; \boldsymbol{\xi}_t)$. To this end, we note that the problem dual to $\underline{Q}_t(\boldsymbol{x}_{t-1}; \boldsymbol{\xi}_t)$ is

$$\begin{array}{ll} \text{maximize} & [\boldsymbol{h}_t(\boldsymbol{\xi}_t) - \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1}]^\top \boldsymbol{\pi}_t - \underline{\mathcal{Q}}^\star_{t+1}(\boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t) \\ \text{subject to} & \boldsymbol{\pi}_t \in \mathbb{R}^{m_t}_+, \end{array}$$

which is a linear program. Here, $\underline{\mathcal{Q}}^\star_{t+1}$ is the conjugate of $\underline{\mathcal{Q}}_{t+1}$, and it is piecewise affine and convex whenever $\underline{\mathcal{Q}}_{t+1}$ is. Due to assumption (A2), strong linear programming duality holds between

$\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ and its dual whenever $\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t) > -\infty$. For a fixed parameter realization $\boldsymbol{\xi}_t \in \Xi_t$, the optimal solution $\boldsymbol{\pi}_t$ of the dual problem provides the supporting hyperplane

$$\boldsymbol{x}_{t-1} \mapsto [\boldsymbol{h}_t(\boldsymbol{\xi}_t) - \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\pi}_t - \underline{\mathcal{Q}}^\star_{t+1}(\boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t)$$

that touches $\underline{Q}_t(\cdot;\boldsymbol{\xi}_t)$ at $\boldsymbol{x}_{t-1}$. Note that this hyperplane does not support the stage-t worst-case cost to-go function $\mathcal{Q}_t$ in general since the lower bound $\underline{\mathcal{Q}}_{t+1}$ can strictly underestimate $\mathcal{Q}_{t+1}$ and $\boldsymbol{\xi}_t$ may not be the worst parameter realization for $\mathcal{Q}_t$.

In the remainder of this section, we describe our RDDP scheme (Section 3.1), we show how the subproblems (2) can be solved as mixed-integer linear programs or via vertex enumeration (Section 3.2), and we study the convergence of our method (Section 3.3).

3.1. Robust Dual Dynamic Programming Scheme

Our RDDP scheme iteratively refines lower and upper bounds on the worst-case cost to-go functions $\mathcal{Q}_t$. To this end, the algorithm proceeds in *forward passes*, which generate sequences of approximate worst-case parameter realizations $\overline{\boldsymbol{\xi}}_t^{\text{fw}}$ and their associated recourse decisions $\underline{\boldsymbol{x}}_t$, $t = 2,\ldots,T$, as well as *backward passes*, which refine the lower and upper bounds $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ along these decisions $\underline{\boldsymbol{x}}_t$. Both bounds are essential for our method, which distinguishes our algorithm from the nested Benders' decomposition and SDDP schemes (see Section 4). Intuitively speaking, we can regard the nested problem formulation (1') as a T-stage zero-sum game between a decision maker aiming to minimize the overall costs and an adversary attempting to maximize these costs. We determine the optimal value of this game by iteratively refining progressive approximations of the payoffs for both the adversary (via the upper bounds $\overline{\mathcal{Q}}_t$) and the decision maker (via the lower bounds $\underline{\mathcal{Q}}_t$). Once we have determined a strategy pair $(\boldsymbol{\xi}, \boldsymbol{x})$ for the adversary and the decision maker such that the upper and lower bounds in each stage t coincide at $\boldsymbol{x}_t$, both strategies are optimal and we have found a worst-case parameter realization $\boldsymbol{\xi}$ and a worst-case optimal decision $\boldsymbol{x}$.

Our RDDP scheme can be summarized as follows.

1. **Initialization:** Set $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) = -\infty \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$, $t = 2,\ldots,T$ (lower bound),

   $\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) = +\infty \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$, $t = 2,\ldots,T$ (upper bound),

   $\underline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = \overline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = 0 \ \forall \boldsymbol{x}_T \in \mathbb{R}^{n_T}$ (zero terminal costs).

2. **Stage-1 Problem:** Let $\underline{\boldsymbol{x}}_1$ be an optimal solution to the lower bound stage-1 problem

$$\begin{array}{ll} \text{minimize} & \boldsymbol{q}_1^\top \boldsymbol{x}_1 + \underline{\mathcal{Q}}_2(\boldsymbol{x}_1) \\ \text{subject to} & \boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1 \\ & \boldsymbol{x}_1 \in \mathbb{R}^{n_1}. \end{array}$$

If $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$, terminate: $\underline{\boldsymbol{x}}_1$ is optimal with costs $\boldsymbol{q}_1^\top \underline{\boldsymbol{x}}_1 + \underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$. Else, go to Step 3.

3. **Forward Pass:** For $t = 2, \ldots, T-1$, let $\overline{\boldsymbol{\xi}}_t^{\text{fw}}$ be an optimal solution to the upper bound problem $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ and $\underline{\boldsymbol{x}}_t$ be an optimal solution to the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\text{fw}})$.

4. **Backward Pass:** For $t = T, \ldots, 2$, let $\overline{\boldsymbol{\xi}}_t^{\text{bw}}$ be an optimal solution to the upper bound problem $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$. If $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}) < \overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$, then update the upper bound

$$\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \leftarrow \text{env}\left(\min\left\{\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}),\ \overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}) + \delta_{\{\underline{\boldsymbol{x}}_{t-1}\}}(\boldsymbol{x}_{t-1})\right\}\right) \qquad \forall \boldsymbol{x}_{t-1} \in \mathbb{R}_{n_{t-1}},$$

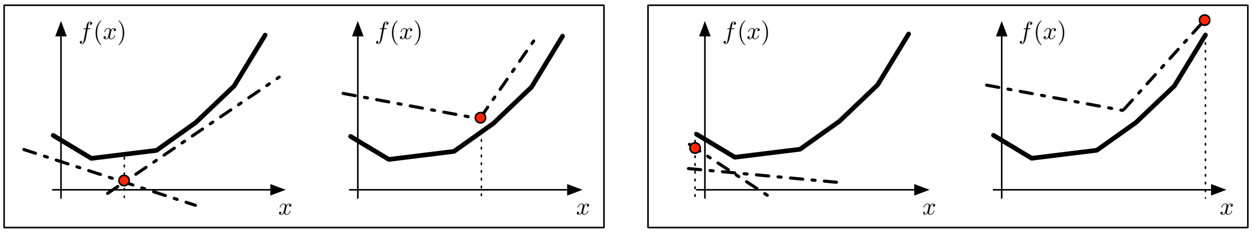
where 'env' denotes the lower convex envelope and $\delta_{\{\underline{\boldsymbol{x}}_{t-1}\}}$ is the indicator function for the singleton set $\{\underline{\boldsymbol{x}}_{t-1}\}$, that is, $\delta_{\{\underline{\boldsymbol{x}}_{t-1}\}}(\boldsymbol{x}_{t-1}) = 0$ if $\boldsymbol{x}_{t-1} = \underline{\boldsymbol{x}}_{t-1}$ and $\delta_{\{\underline{\boldsymbol{x}}_{t-1}\}}(\boldsymbol{x}_{t-1}) = +\infty$ otherwise.

Let $\boldsymbol{\pi}_t$ be the shadow price of an optimal solution to the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\text{bw}})$. If $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\text{bw}}) > \underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$, then update the lower bound

$$\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \leftarrow \max\left\{\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}),\ [\boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\text{bw}}) - \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\text{bw}})\, \boldsymbol{x}_{t-1}]^\top \boldsymbol{\pi}_t - \underline{\mathcal{Q}}_{t+1}^\star(\boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t)\right\}\ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}. \tag{3}$$

After Step 4 has been completed for all stages $t = T, \ldots, 2$, go back to Step 2.

After initializing the lower and upper bounds $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ on the worst-case cost to-go functions $\mathcal{Q}_t$ in Step 1, Step 2 solves a relaxation of problem (1') where the stage-2 worst-case cost to-go function $\mathcal{Q}_2$ is replaced with its lower bound $\underline{\mathcal{Q}}_2$. Due to the assumption (A2), the relaxed stage-1 problem is feasible as long as there is a decision $\boldsymbol{x}_1$ satisfying $\boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1$, which we assume to be the case in order to avoid trivially infeasible instances of (1). Moreover, since the feasible region of the relaxed stage-1 problem is compact by assumption (A1), the stage-1 problem always attains its optimal value. The optimal value of the stage-1 problem can be $-\infty$ if the worst-case cost approximation $\underline{\mathcal{Q}}_2$ has not yet been updated. In this case, any solution attaining this objective value—which is guaranteed to exist due to the assumption (A1)—is considered to be optimal. The



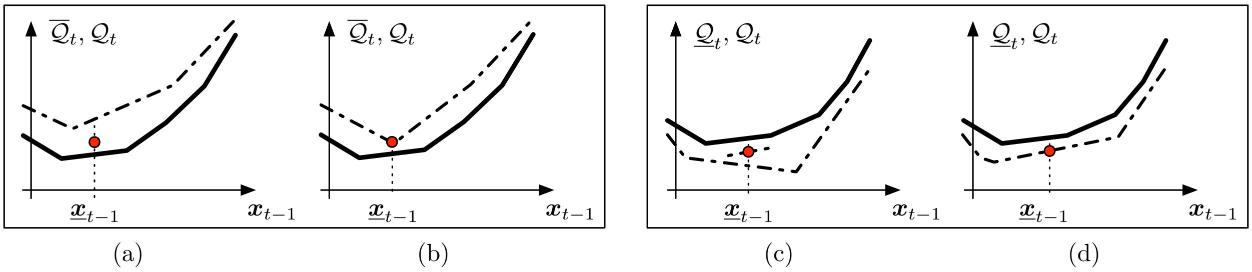
(a) (b) (c) (d)

Figure 1 **Left:** The minimizer of the bold function $f(x)$ can be found by iteratively refining lower bounds at their minimum and terminating once the lower bound touches $f(x)$ at its minimum (a). If the search iteratively refined upper bounds at their minimum instead (b), then the algorithm could prematurely terminate in a suboptimal point (circle). **Right:** The reverse situation arises if we search for the maximizer of $f(x)$.

algorithm terminates if the first-stage decision $\underline{\boldsymbol{x}}_1$ satisfies $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$. Indeed, in this case $\underline{\boldsymbol{x}}_1$ optimizes a relaxation of problem (1’) and simultaneously attains the same objective value in (1’) itself, which implies that $\underline{\boldsymbol{x}}_1$ must be optimal in (1’).

The algorithm conducts a forward pass in Step 3, which generates a sequence of approximate worst-case parameter realizations $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}}$ and associated recourse decisions $\underline{\boldsymbol{x}}_t$, $t = 2, \ldots, T-1$. Note that the parameter realizations $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}}$ are selected from $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$, which employ the *upper bound* cost to-go functions $\overline{\mathcal{Q}}_{t+1}$, whereas the decisions $\underline{\boldsymbol{x}}_t$ are chosen based on $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{fw}})$, which utilize the *lower bound* cost to-go functions $\underline{\mathcal{Q}}_{t+1}$. This ensures that the refinements of the upper and lower bounds do not terminate prematurely in local optima. The intuition behind the use of different bounds for the maximization over $\boldsymbol{\xi}_t$ and the minimization of $\boldsymbol{x}_t$ is explained in Figure 1.

The algorithm proceeds with a backward pass in Step 4, which refines the upper and lower bounds $\overline{\mathcal{Q}}_t$ and $\underline{\mathcal{Q}}_t$ along the decision trajectory $\underline{\boldsymbol{x}}_t$ computed in Step 3. The upper bounds $\overline{\mathcal{Q}}_t$ are updated using the upper bound problems $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$, whereas the lower bounds $\underline{\mathcal{Q}}_t$ are updated using the lower bound problems $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})$. In both cases, the convexity of the worst-case cost to-go functions $\mathcal{Q}_t$ allows us to update the bounds also in the neighborhood of the candidate solution $\underline{\boldsymbol{x}}_{t-1}$. This is achieved through the use of convex envelopes (for $\overline{\mathcal{Q}}_t$) and supporting hyperplanes (for $\underline{\mathcal{Q}}_t$), see Figure 2. Note that $\overline{\mathcal{Q}}_T(\underline{\boldsymbol{x}}_{T-1}) = \mathcal{Q}_T(\underline{\boldsymbol{x}}_{T-1}) = \underline{\mathcal{Q}}_T(\underline{\boldsymbol{x}}_{T-1})$ after Step 4 since $\overline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = \mathcal{Q}_{T+1}(\boldsymbol{x}_T) = \underline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = 0$ for all $\boldsymbol{x}_T \in \mathbb{R}^{n_T}$ and therefore $\overline{Q}_T(\underline{\boldsymbol{x}}_{T-1}) = \underline{Q}_T(\underline{\boldsymbol{x}}_{T-1}; \overline{\boldsymbol{\xi}}_T^{\mathrm{bw}}) = \mathcal{Q}_T(\underline{\boldsymbol{x}}_{T-1})$. The improved upper and lower bounds are then propagated to the earlier time stages. The refined



(a) (b) (c) (d)

Figure 2 **Left:** Updating the upper bound $\overline{\mathcal{Q}}_t$ (dashed line) on the worst-case costs $\mathcal{Q}_t$ (bold line) with the solution $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ (circle) in (a) yields the refined upper bound in (b). **Right:** Updating the lower bound $\underline{\mathcal{Q}}_t$ (dashed line) on the worst-case costs $\mathcal{Q}_t$ (bold line) with the supporting hyperplane corresponding to the optimal solution of $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})$ (circle) in (c) yields the refined lower bound in (d).

upper bounds $\overline{\mathcal{Q}}_{t+1}$ imply that the approximate worst-case parameter realizations $\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}$ might have changed, which is why we do not reuse the realizations $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}}$ from the forward pass in Step 4.

For the special case where $T = 2$, our algorithm reduces to a variant of Benders' decomposition that maintains both lower and upper bounds on the worst-case cost function $\mathcal{Q}_2$. In that case, the computation of the upper bounds would be redundant, however, since $\mathcal{Q}_3(\boldsymbol{x}_2) = 0$ for all $\boldsymbol{x}_2 \in \mathbb{R}^{n_2}$, and hence the refined lower bound $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$ coincides with the exact worst-case costs $\mathcal{Q}_2(\underline{\boldsymbol{x}}_1)$ after every iteration of the algorithm. Maintaining both lower and upper bounds on the cost functions $\mathcal{Q}_t$ becomes crucial for $T > 2$ stages, on the other hand, since the lower bound values $\underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$ tend to strictly underestimate the exact worst-case costs $\mathcal{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ for $t < T$.

We will show in Proposition 2 below that the upper and lower bounds $\overline{\mathcal{Q}}_t$ and $\underline{\mathcal{Q}}_t$ satisfy $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \leq \mathcal{Q}_t(\boldsymbol{x}_{t-1}) \leq \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1})$ for all $t = 2, \ldots, T$ and $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$. Our algorithm thus provides deterministic upper and lower bounds $\boldsymbol{q}_1^\top \underline{\boldsymbol{x}}_1 + \overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$ and $\boldsymbol{q}_1^\top \underline{\boldsymbol{x}}_1 + \underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$ on the optimal value of the multi-stage robust optimization problem (1) in every iteration. In fact, the previous bounding argument allows us to tighten the upper bound on (1) to $\min\{\boldsymbol{q}_1^\top \boldsymbol{x}_1 + \overline{\mathcal{Q}}_2(\boldsymbol{x}_1) : \boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1,\ \boldsymbol{x}_1 \in \mathbb{R}^{n_1}\}$. We now show that at any time during the execution of the algorithm, the upper bounds $\overline{\mathcal{Q}}_t$, $t = 2, \ldots, T$, also allow us to construct a feasible decision policy $\boldsymbol{x}_t(\boldsymbol{\xi}^t)$.

Proposition 1 **(Feasible Solutions)**. *Any first-stage decision $\boldsymbol{x}_1$ satisfying $\overline{\mathcal{Q}}_2(\boldsymbol{x}_1) < +\infty$ can be completed to a feasible decision policy $\boldsymbol{x}_t(\boldsymbol{\xi}^t)$, $t = 1, \ldots, T$, defined through $\boldsymbol{x}_1(\boldsymbol{\xi}^1) = \boldsymbol{x}_1$ and*

$$\boldsymbol{x}_t(\boldsymbol{\xi}^t) \in \mathop{\arg\min}_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) : \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1}(\boldsymbol{\xi}^{t-1}) + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \right\}, \ \boldsymbol{\xi} \in \Xi \ \textit{and}\ t = 2, \ldots, T,$$

and this decision policy satisfies $\sum_{t=1}^{T} \boldsymbol{q}_t^\top \boldsymbol{x}_t(\boldsymbol{\xi}^t) \leq \boldsymbol{q}_1 \boldsymbol{x}_1 + \overline{\mathcal{Q}}_2(\boldsymbol{x}_1)$ *for all* $\boldsymbol{\xi} \in \Xi$.

Proof. Fix any $t = 2, \ldots, T$ and $\hat{\boldsymbol{\xi}} \in \Xi$. We then have that

$$\begin{aligned} \boldsymbol{q}_t^\top \boldsymbol{x}_t(\hat{\boldsymbol{\xi}}^t) + \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t(\hat{\boldsymbol{\xi}}^t)) \;&=\; \left[\begin{array}{ll} \text{minimize} & \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \\ \text{subject to} & \boldsymbol{T}_t(\hat{\boldsymbol{\xi}}_t)\, \boldsymbol{x}_{t-1}(\hat{\boldsymbol{\xi}}^{t-1}) + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\hat{\boldsymbol{\xi}}_t) \\ & \boldsymbol{x}_t \in \mathbb{R}^{n_t} \end{array} \right] \\ &\leq\; \overline{Q}_t(\boldsymbol{x}_{t-1}(\hat{\boldsymbol{\xi}}^{t-1})) \;\leq\; \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}(\hat{\boldsymbol{\xi}}^{t-1})), \end{aligned} \tag{4}$$

where the identity is due to the definition of the policy $\boldsymbol{x}_t$, and the first inequality holds since the minimization problem is a variant of the upper bound problem $\overline{Q}_t(\boldsymbol{x}_{t-1}(\hat{\boldsymbol{\xi}}^{t-1}))$ in which Ξ_t is replaced with the singleton subset $\{\hat{\boldsymbol{\xi}}_t\}$. In view of the second inequality, we show in the following that $\overline{Q}_t(\boldsymbol{x}_{t-1}) \leq \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1})$ for all $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$. The statement of the proposition then follows from an iterative application of equation (4) and the fact that $\overline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = 0$ for all $\boldsymbol{x}_T \in \mathbb{R}^{n_T}$.

To show that $\overline{Q}_t(\boldsymbol{x}_{t-1}) \leq \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1})$ for all $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$, fix any $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$ with $\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) < +\infty$. The epigraph of $\overline{\mathcal{Q}}_t$ can be expressed as the Minkowski sum of the set $\{\boldsymbol{0}\} \times \mathbb{R}_+$ and the convex hull of the points $(\underline{\boldsymbol{x}}_{t-1}^i, \overline{Q}_t^{\ell_i}(\underline{\boldsymbol{x}}_{t-1}^i))$, $i = 1, 2, \ldots$, where $\overline{Q}_t^{\ell_i}(\underline{\boldsymbol{x}}_{t-1}^i)$ denotes the upper bound problem solved for the candidate solution $\underline{\boldsymbol{x}}_{t-1}^i$ in iteration ℓ_i. By construction of the upper bound updates in Step 4, $\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) < +\infty$ implies that $\boldsymbol{x}_{t-1}$ can be expressed as a convex combination of previously visited candidate solutions $\underline{\boldsymbol{x}}_{t-1}^i$, that is, we have $(\boldsymbol{x}_{t-1}, \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1})) = \sum_i \lambda_i \cdot (\underline{\boldsymbol{x}}_{t-1}^i, \overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}^i))$ for some $\boldsymbol{\lambda} \geq \boldsymbol{0}$ with $\mathbf{e}^\top \boldsymbol{\lambda} = 1$. We then have that

$$\overline{Q}_t(\boldsymbol{x}_{t-1}) \;=\; \overline{Q}_t\left(\sum_i \lambda_i \cdot \underline{\boldsymbol{x}}_{t-1}^i\right) \;\leq\; \sum_i \lambda_i \cdot \overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}^i) \;\leq\; \sum_i \lambda_i \cdot \overline{Q}_t^{\ell_i}(\underline{\boldsymbol{x}}_{t-1}^i) \;=\; \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}),$$

where the first inequality follows from the fact that the mapping $\boldsymbol{x}_{t-1} \mapsto \overline{Q}_t(\boldsymbol{x}_{t-1})$ is convex, see Propositions 2.9 and 2.22 in Rockafellar and Wets (2009), and the second inequality holds since for every time stage t, the cost to-go approximations $\overline{\mathcal{Q}}_t$ form a pointwise monotonically non-increasing function sequence throughout the execution of the algorithm. □

Proposition 1 enables us to complete a first-stage decision $\boldsymbol{x}_1$ to a feasible decision for each stage $t = 1, \ldots, T$ in response to the observed parameter realizations $\boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_T$ by solving one linear program per time stage. Lower costs may be obtained by a shrinking horizon variant of our RDDP scheme that updates the worst-case cost to-go approximations $(\underline{\mathcal{Q}}_t, \overline{\mathcal{Q}}_t) \leftarrow (\underline{\mathcal{Q}}_{t+1}, \overline{\mathcal{Q}}_{t+1})$ and the time horizon $T \leftarrow T-1$ and restarts the algorithm at Step 2 after each observed realization $\boldsymbol{\xi}_t$. This is the case since the solutions to multi-stage robust optimization problems are typically not Pareto optimal, see Iancu and Trichakis (2014) and de Ruiter et al. (2016).

The algorithm outline described above can be modified in multiple ways. We conclude this section with a variant of our RDDP scheme that we will employ in our numerical results.

Remark 5 (Selection of Next Stage). Our RDDP scheme iterates between full forward passes in Step 3 and full backward passes in Step 4. As we will see in Section 3.3, however, the correctness and convergence of our algorithm extends to variants that heuristically switch between forward and backward steps, as long as we move to consecutive time stages. For example, we can select the next stage based on a comparison of the *forward gap* $\text{gap}_{\text{fwd}} = \overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}) - \underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$, which quantifies the cumulative gap of the approximations $(\underline{\mathcal{Q}}_t, \overline{\mathcal{Q}}_t), \ldots, (\underline{\mathcal{Q}}_T, \overline{\mathcal{Q}}_T)$, with the *backward gap* $\text{gap}_{\text{bwd}} = (\overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) - \underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)) - \text{gap}_{\text{fwd}}$, which measures the cumulative gap of the approximations $(\underline{\mathcal{Q}}_2, \overline{\mathcal{Q}}_2), \ldots, (\underline{\mathcal{Q}}_{t-1}, \overline{\mathcal{Q}}_{t-1})$. Heuristic stage selection strategies have been successfully applied to the nested Benders' decomposition for stochastic programs, see Louveaux (1980) and Birge (1985).

3.2. Solution of the Subproblems $\overline{Q}_t(\boldsymbol{x}_{t-1})$

Theorem 1 implies that the max-min problem $\overline{Q}_t(\boldsymbol{x}_{t-1})$ in equation (2) is strongly NP-hard if the technology matrix $\boldsymbol{T}_t$ or the right-hand side vector $\boldsymbol{h}_t$ is uncertain. We now show how this problem can be solved as a mixed-integer linear program or via vertex enumeration.

While it seems natural to transform the max-min problem (2) into a single-stage problem by dualizing the embedded minimization problem in (2), this reformulation would result in a bilinear program whose objective function maximizes the inner product of $\boldsymbol{h}_t(\boldsymbol{\xi}_t) - \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1}$ with the newly introduced dual variables. Instead, we follow the approach of Dempe (2002) and Zeng and Zhao (2013) and replace the embedded minimization problem in (2) with its Karush-Kuhn-Tucker optimality conditions. To this end, we first reformulate problem (2) as

$$\begin{array}{ll} \text{maximize} & \theta \\ \text{subject to} & (\boldsymbol{x}_t, \theta) \in \arg\min \left\{ \theta : \begin{bmatrix} \theta \geq \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \\ \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \\ \boldsymbol{x}_t \in \overline{\mathcal{Q}}_t^{\mathrm{F}}, \;\; \theta \in \mathbb{R} \end{bmatrix} \right\} \\ & \boldsymbol{\xi}_t \in \Xi_t, \end{array} \tag{2'}$$

where the polyhedron $\overline{\mathcal{Q}}_t^{\mathrm{F}} = \{\boldsymbol{x}_t \in \mathbb{R}^{n_t} : \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) < +\infty\}$ describes the stage-t decisions with finite upper cost to-go bounds. The upper bound $\overline{\mathcal{Q}}_t$ in the embedded objective function as well as the set $\overline{\mathcal{Q}}_t^{\mathrm{F}}$ can be represented through a convex combination model (also known as the 'lambda method') as in generalized linear programming, see Dantzig (1963, §24) and Magnanti et al. (1976).

The problems (2) and (2') both have finite optimal values if and only if

$$\forall \boldsymbol{\xi}_t \in \Xi_t \;\; \exists \boldsymbol{x}_t \in \overline{\mathcal{Q}}_t^{\mathrm{F}} \;\; : \;\; \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t). \tag{5}$$

One readily verifies that if the condition (5) is satisfied, then (2) and (2') share the same optimal value as well as the same set of maximizers $\boldsymbol{\xi}_t$. On the other hand, the optimal values of (2) and (2') differ whenever the condition (5) is violated. Indeed, in that case there is a $\boldsymbol{\xi}_t \in \Xi_t$ such that no feasible solution $\boldsymbol{x}_t \in \mathbb{R}^{n_t}$ to the inner problem in (2) satisfies $\overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) < +\infty$, and the optimal value of (2) is $+\infty$. Since the inner minimization problem in (2') is infeasible for this $\boldsymbol{\xi}_t$, however, the associated set of minimizers $(\boldsymbol{x}_t, \theta)$ in (2') is empty, and $\boldsymbol{\xi}_t$ is infeasible in the problem (2').

Note that the relatively complete recourse assumption (A2) guarantees that the condition (5) holds once the upper cost to-go approximation $\overline{\mathcal{Q}}_{t+1}$ is sufficiently refined. In that case, we can replace the embedded minimization problem in (2') with its Karush-Kuhn-Tucker optimality conditions. The resulting reformulation contains bilinear constraints that correspond to the complementary slackness conditions. These constraints can be reformulated as linear constraints if we introduce auxiliary binary variables and adopt a Big-M reformulation, see Zeng and Zhao (2013, §3.2). Alternatively, we can replace the complementary slackness conditions with SOS-1 constraints, which enforce for each primal constraint that either the constraint's slack or the associated dual variable (or both) are zero. SOS-1 constraints avoid the necessity to choose a Big-M constant and are natively supported by many state-of-the-art mixed-integer linear programming solvers.

In the early stages of our RDDP scheme, the condition (5) may be violated when the upper cost to-go approximation $\overline{\mathcal{Q}}_{t+1}$ has not yet been sufficiently refined. To detect this case, we first solve a variant of problem (2') in which the inner problem minimizes the distance of $\boldsymbol{z}_t$ to the set $\overline{\mathcal{Q}}_t^{\mathrm{F}}$. If the optimal value of this auxiliary problem is zero, then we proceed to solve problem (2') as described above. Otherwise, the optimal solution $\boldsymbol{\xi}_t$ to the auxiliary problem is used as the optimal solution to the upper bound problem $\overline{Q}_t(\boldsymbol{x}_{t-1})$. Note that $\overline{Q}_t(\boldsymbol{x}_{t-1}) = +\infty$ in this case.

Standard arguments show that as long as the objective coefficients $\boldsymbol{q}_t$ and the recourse matrices $\boldsymbol{W}_t$ are deterministic, then the optimal value of the embedded minimization problem in (2) is convex in the uncertain problem parameters $\boldsymbol{\xi}_t$, and thus the problem (2) is maximized at an extreme point $\boldsymbol{\xi}_t \in \operatorname{ext} \Xi_t$ of the uncertainty set Ξ_t. We can then solve the problem by solving $|\operatorname{ext} \Xi_t|$ linear programs, each of which corresponds to an instance of the embedded minimization problem in (2) for a fixed parameter realization $\boldsymbol{\xi}_t \in \operatorname{ext} \Xi_t$. Note that the infeasible instances of these linear programs correspond to extreme points $\boldsymbol{\xi}_t \in \operatorname{ext} \Xi_t$ for which the upper bound problem $\overline{Q}_t(\boldsymbol{x}_{t-1})$ evaluates to $+\infty$. This 'brute force' vertex enumeration approach can be computationally attractive if the number of uncertain problem parameters per stage is small or if sufficient resources are available to solve the linear programs in parallel. Moreover, the approach can benefit from speedup techniques such as warm-starting and bunching (Birge and Louveaux 1997, §5.4).

3.3. Finite Convergence Guarantee

To study the convergence of our RDDP scheme, we make the following two assumptions:

(C1) The solution method for the upper bound problems $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ returns an optimal extreme point of the uncertainty set Ξ_t.

(C2) The solution method for the lower bound problems returns an optimal basic feasible solution of the linear program corresponding to the epigraph reformulation of $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \boldsymbol{\xi}_t)$.

The first assumption is satisfied by construction if we employ a vertex enumeration scheme to solve $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$. If we solve $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ as a mixed-integer linear program, the assumption could be violated if $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ has multiple optimal solutions. In this case, an optimal extreme point of Ξ_t

can be recovered by adding a regularization term $\boldsymbol{\rho}_t^\top \boldsymbol{\xi}_t$ to the objective function of $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ such that $\boldsymbol{\rho}_t^\top \not\propto \boldsymbol{F}_{ti}^\top$, $i=1,\ldots,l_t$, that is, $\boldsymbol{\rho}_t$ is not parallel to any of the halfspaces defining Ξ_t, and $\|\boldsymbol{\rho}_t\|$ is small. Assumption (C2) is mild since $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1};\boldsymbol{\xi}_t)$ constitutes a linear program, and every major linear programming software package offers the option to return an optimal basic feasible solution. In the early iterations of our RDDP scheme, the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1};\boldsymbol{\xi}_t)$ may evaluate to $-\infty$. In this case, we assume that the solution method returns any extreme point $\boldsymbol{x}_t$ of the set $\{\boldsymbol{x}_t \in \mathbb{R}^{n_t} : \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\underline{\boldsymbol{x}}_{t-1} + \boldsymbol{W}_t\boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t)\}$ for which $\boldsymbol{q}_t^\top\boldsymbol{x}_t + \underline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) = -\infty$.

The presented RDDP scheme starts with the trivial bounds $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) = -\infty$ and $\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) = +\infty$. These bounds are refined in the backward passes in Step 4. We first show that these updates preserve the bounding properties of $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$.

Proposition 2 **(Bounding Property)**. *Throughout the algorithm, the bounds $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ satisfy*

$$\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \;\leq\; \mathcal{Q}_t(\boldsymbol{x}_{t-1}) \;\leq\; \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \qquad \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}},\ \forall t = 2,\ldots,T,$$

that is, $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ bound the worst-case cost to-go function $\mathcal{Q}_t$ from below and above, respectively.

Proof. The statement trivially holds after Step 1 of the algorithm. We now show that the bounding properties of $\overline{\mathcal{Q}}_t$ and $\underline{\mathcal{Q}}_t$ are preserved in Step 4 of the algorithm.

The first part of Step 4 calculates an upper bound on the worst-case costs $\mathcal{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ since $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}) \geq \mathcal{Q}_t(\underline{\boldsymbol{x}}_{t-1})$. Indeed, $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ involves $\overline{\mathcal{Q}}_{t+1}$, which by assumption bounds the worst-case cost to-go function $\mathcal{Q}_{t+1}$ from above. The convexity of $\mathcal{Q}_t$ then implies that the refined bound

$$\operatorname{env}\left(\min\left\{\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}),\ \overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}) + \delta_{\{\underline{\boldsymbol{x}}_{t-1}\}}(\boldsymbol{x}_{t-1})\right\}\right)$$

continues to bound the worst-case cost to-go function $\mathcal{Q}_t$ from above.

The second part of Step 4 calculates a lower bound on the worst-case costs $\mathcal{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ since

$$\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1};\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) \;\leq\; \max_{\boldsymbol{\xi}_t\in\Xi_t} \underline{Q}_t(\underline{\boldsymbol{x}}_{t-1};\boldsymbol{\xi}_t) \;\leq\; \mathcal{Q}_t(\boldsymbol{x}_{t-1}),$$

where the second inequality holds since $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1};\boldsymbol{\xi}_t)$ relies on $\underline{\mathcal{Q}}_{t+1}$. Moreover, we have

$$\begin{aligned}
&[\boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) - \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})\,\boldsymbol{x}_{t-1}]^\top\boldsymbol{\pi}_t - \underline{\mathcal{Q}}_{t+1}^\star(\boldsymbol{W}_t^\top\boldsymbol{\pi}_t - \boldsymbol{q}_t)\\
\leq\; &\max_{\boldsymbol{\pi}\in\mathbb{R}_+^{m_t}} \left\{[\boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) - \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})\,\boldsymbol{x}_{t-1}]^\top\boldsymbol{\pi} \;- \underline{\mathcal{Q}}_{t+1}^\star(\boldsymbol{W}_t^\top\boldsymbol{\pi} \;- \boldsymbol{q}_t)\right\} \qquad \forall \boldsymbol{x}_{t-1}\in\mathbb{R}^{n_{t-1}},
\end{aligned}$$

and weak linear programming duality implies that the right-hand side of this inequality bounds $\underline{Q}_t(\boldsymbol{x}_{t-1};\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})$—and, *a fortiori*, the worst-case costs $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$—from below. □

Proposition 2 implies that the stage-1 problem is a relaxation of problem (1'), and that any solution $\underline{\boldsymbol{x}}_1$ satisfying $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$ must also satisfy $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \mathcal{Q}_2(\underline{\boldsymbol{x}}_1)$. Thus, any such solution $\underline{\boldsymbol{x}}_1$ optimizes a relaxation of problem (1') and simultaneously attains the same objective value in (1'), that is, it must be optimal in (1'). We have thus arrived at the following result.

Corollary 1 **(Correctness)**. *If the RDDP scheme terminates, then it returns an optimal solution to the multi-stage robust optimization problem* (1)*.*

We now prove that our RDDP scheme converges in finite time. To this end, we first show that there are finitely many different lower and upper bounds $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ that the algorithm can generate during its execution. We then show that each backward pass refines at least one of these bounds.

Lemma 1 **(Lower Bounds)**. *For a fixed instance of problem* (1)*, there are finitely many different lower bound cost functions $\underline{\mathcal{Q}}_t$, $t = 2,\ldots,T$, that the algorithm can generate.*

Proof. We prove the statement via backward induction on the time stage t.

The stage-T lower bound problem $\underline{Q}_T(\underline{\boldsymbol{x}}_{T-1};\overline{\boldsymbol{\xi}}_T^{\mathrm{bw}})$ contains the worst-case cost to-go function $\underline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = 0$, $\boldsymbol{x}_T \in \mathbb{R}^{n_T}$, which implies that the conjugate cost to-go function satisfies $\underline{\mathcal{Q}}_{T+1}^{\star} = \delta_{\{\mathbf{0}\}}$. Hence, the dual problem associated with $\underline{Q}_T(\underline{\boldsymbol{x}}_{T-1};\overline{\boldsymbol{\xi}}_T^{\mathrm{bw}})$ simplifies to

$$\begin{array}{ll} \text{maximize} & [\boldsymbol{h}_T(\overline{\boldsymbol{\xi}}_T^{\mathrm{bw}}) - \boldsymbol{T}_T(\overline{\boldsymbol{\xi}}_T^{\mathrm{bw}})\,\underline{\boldsymbol{x}}_{T-1}]^\top \boldsymbol{\pi}_T \\ \text{subject to} & \boldsymbol{\pi}_T \in \mathbb{R}_+^{m_T},\ \boldsymbol{W}_T^\top \boldsymbol{\pi}_T = \boldsymbol{q}_T. \end{array}$$

This is a linear program with finitely many basic feasible solutions, none of which depends on the candidate solution $\underline{\boldsymbol{x}}_{T-1}$ or the parameter realization $\overline{\boldsymbol{\xi}}_T^{\mathrm{bw}}$. Since the assumption (C2) ensures that the supporting hyperplane from Step 4 of the algorithm corresponds to a basic feasible solution of this dual problem, we thus conclude that for every $\overline{\boldsymbol{\xi}}_T^{\mathrm{bw}}$ there are finitely many different supporting hyperplanes that can be added to the lower bound $\underline{\mathcal{Q}}_T$ throughout the execution of the algorithm. The statement then follows for stage T since there are finitely many extreme points $\overline{\boldsymbol{\xi}}_T^{\mathrm{bw}} \in \Xi_t$ that can emerge as optimal solutions to the upper bound problem according to the assumption (C1).

The dual problem associated with the stage-t lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1};\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})$ is

$$\begin{aligned} \text{maximize} \quad & [\boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) - \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})\, \underline{\boldsymbol{x}}_{t-1}]^\top \boldsymbol{\pi}_t - \varphi \\ \text{subject to} \quad & \varphi \geq \underline{\mathcal{Q}}_{t+1}^\star(\boldsymbol{\mu}),\ \boldsymbol{\mu} = \boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t \\ & \boldsymbol{\pi}_t \in \mathbb{R}_+^{m_t},\ \boldsymbol{\mu} \in \mathbb{R}^{n_t},\ \varphi \in \mathbb{R}. \end{aligned} \tag{6}$$

It involves the conjugate $\underline{\mathcal{Q}}_{t+1}^\star$ of the worst-case cost to-go approximation $\underline{\mathcal{Q}}_{t+1}$. By construction, $\underline{\mathcal{Q}}_{t+1}$ is a piecewise affine convex function. Moreover, the induction hypothesis implies that $\underline{\mathcal{Q}}_{t+1}$ has finitely many different pieces, and it takes on finitely many different shapes throughout the algorithm. Hence, its conjugate $\underline{\mathcal{Q}}_{t+1}^\star$ is also a piecewise affine convex function with finitely many different pieces that takes on finitely many different shapes. Problem (6) can then be cast as a linear program with finitely many basic feasible solutions, none of which depends on $\underline{\boldsymbol{x}}_{t-1}$ or $\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}$. A similar argument as in the previous paragraph then shows that there are finitely many different supporting hyperplanes that can be added to the lower bound $\underline{Q}_t$ throughout the algorithm. □

Lemma 2 **(Upper Bounds)**. *For a fixed instance of problem* (1)*, there are finitely many different upper bound cost functions* $\overline{\mathcal{Q}}_t$*,* $t = 2,\ldots,T$*, that the algorithm can generate.*

Proof. We first show that the RDDP scheme only visits finitely many different candidate solutions $\underline{\boldsymbol{x}}_t$, $t = 1,\ldots,T$, throughout its execution. We then prove that at each $\underline{\boldsymbol{x}}_t$, $t = 1,\ldots,T-1$, the upper bound cost to-go function $\overline{\mathcal{Q}}_{t+1}$ can only be refined finitely many times.

We show the first statement by forward induction on t. The algorithm only visits finitely many different candidate solutions $\underline{\boldsymbol{x}}_1$ since it generates a finite number of lower bound cost functions $\underline{\mathcal{Q}}_2$ (see Lemma 1) and the employed solution method returns an optimal basic feasible solution by assumption (C2). Consider now the stage $t > 1$. The induction hypothesis implies that there are finitely many candidate solutions $\underline{\boldsymbol{x}}_{t-1}$ that the algorithm visits, and the assumption (C1) ensures that every maximizer $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}}$ of the upper bound problem $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ is an extreme point of Ξ_t. Hence, there are finitely many realizations $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}} \in \Xi_t$ that can emerge as optimizers of the forward pass in stage t. Since the algorithm also generates finitely many different progressive approximations $\underline{\mathcal{Q}}_{t+1}$, only finitely many candidate solutions $\underline{\boldsymbol{x}}_t$ can emerge as optimizers of $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1};\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}})$.

We show the second statement by backward induction on t. The upper bound cost to-go function $\overline{\mathcal{Q}}_T(\underline{\boldsymbol{x}}_{T-1})$ can only be refined once for every $\underline{\boldsymbol{x}}_{T-1}$, after which $\overline{\mathcal{Q}}_T(\underline{\boldsymbol{x}}_{T-1}) = \mathcal{Q}_T(\underline{\boldsymbol{x}}_{T-1})$ since

$\overline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = \mathcal{Q}_{T+1}(\boldsymbol{x}_T) = 0$ for all $\boldsymbol{x}_T \in \mathbb{R}^{n_T}$. Consider now the stage $t < T$. The first part of the proof and the induction hypothesis imply that a finite number of upper bound cost to-go functions $\overline{\mathcal{Q}}_{t+1}$ are generated throughout the algorithm. Thus, we solve finitely many different upper bound problems $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ for each $\underline{\boldsymbol{x}}_{t-1}$, and $\overline{\mathcal{Q}}_t$ can only be refined finitely many times at $\underline{\boldsymbol{x}}_{t-1}$. □

We are now in the position to prove the finite convergence of our RDDP scheme.

Theorem 2 **(Finite Termination)**. *The RDDP scheme terminates in finite time.*

Proof. We show that in each iteration of the algorithm, at least one of the bounds $\overline{\mathcal{Q}}_t$ or $\underline{\mathcal{Q}}_t$, $t = 2, \ldots, T$, must be refined in Step 4. The result then follows from Lemmas 1 and 2.

Assume to the contrary that none of the bounds $\overline{\mathcal{Q}}_t$, $\underline{\mathcal{Q}}_t$ was refined in Step 4 of the algorithm. We show below via backward induction on t that in this case $\underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}) = \mathcal{Q}_t(\underline{\boldsymbol{x}}_{t-1}) = \overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$ for all $t = 2, \ldots, T$. This would imply, however, that $\underline{\boldsymbol{x}}_1$ satisfies the termination criterion $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$ at the beginning of that iteration. Thus, Step 4 is never reached, which yields a contradiction.

We begin the backward induction with stage T. We have

$$\overline{\mathcal{Q}}_T(\underline{\boldsymbol{x}}_{T-1}) \;=\; \overline{Q}_T(\underline{\boldsymbol{x}}_{T-1}) \;=\; \mathcal{Q}_T(\underline{\boldsymbol{x}}_{T-1}) \;=\; \underline{Q}_T(\underline{\boldsymbol{x}}_{T-1}; \overline{\boldsymbol{\xi}}_T^{\mathrm{bw}}) \;=\; \underline{\mathcal{Q}}_T(\underline{\boldsymbol{x}}_{T-1}),$$

where the first and the last identity hold since the neither of bounds is refined, and the intermediate two equalities follow from the fact that $\overline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = \mathcal{Q}_{T+1}(\boldsymbol{x}_T) = \underline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = 0$ for all $\boldsymbol{x}_T \in \mathbb{R}^{n_T}$.

In the same way, we obtain for stage $t < T$ that

$$\begin{aligned}
\underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}) &\overset{\text{(a)}}{=} \underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) \\
&= \min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \underline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \,:\, \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})\, \underline{\boldsymbol{x}}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) \right\} \\
&\overset{\text{(b)}}{\leq} \min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \,:\, \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})\, \underline{\boldsymbol{x}}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) \right\} \\
&\overset{\text{(c)}}{=} \min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}(\boldsymbol{x}_t) \,:\, \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}})\, \underline{\boldsymbol{x}}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}}) \right\} \\
&= \overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}) \overset{\text{(d)}}{=} \overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}),
\end{aligned}$$

where (a) and (d) hold since $\underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$ and $\overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$ are not updated, respectively, (b) is due to Proposition 2, and (c) follows from $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}} = \overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}$, which holds since the bound $\overline{\mathcal{Q}}_{t+1}$ has not been updated. Since $\underline{\boldsymbol{x}}_t$ is optimal in the minimization problem in the second row and by induction hypothesis achieves the same objective value in the third row, we can strengthen (b) to an equality. Proposition 2 then implies that $\underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}) = \mathcal{Q}_t(\underline{\boldsymbol{x}}_{t-1}) = \overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$. □

4. Comparison with Nested Benders' Decomposition and SDDP

We now compare our RDDP scheme with adaptations of the classical nested Benders' decomposition and stochastic dual dynamic programming (SDDP) to the multi-stage robust optimization problem (1). Contrary to the nested Benders' decomposition and SDDP, our RDDP scheme maintains both lower and upper bounds on the worst-case cost to-go functions $\mathcal{Q}_t$. As we have discussed in Section 3.1, the upper bounds are crucial to determine approximate worst-case scenarios $\overline{\boldsymbol{\xi}}_t^{\text{fw}}$ and $\overline{\boldsymbol{\xi}}_t^{\text{bw}}$ that ensure the convergence of the algorithm.

The nested Benders' decomposition avoids upper cost to-go bounds by solving the lower bound problems $\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ *for all extreme point parameter trajectories* $(\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_T) \in \operatorname{ext}\Xi_1 \times \ldots \times \operatorname{ext}\Xi_T$ in every iteration. Thus, the nested Benders' decomposition necessarily considers the worst-case trajectory in the forward pass of each iteration. SDDP, on the other hand, *randomly* selects a single extreme point parameter trajectory $(\boldsymbol{\xi}_1,\ldots,\boldsymbol{\xi}_T)$ in the forward pass of each iteration. As long as every parameter trajectory can be selected with a positive probability, SDDP thus considers the worst-case trajectory with a probability that approaches 1 over sufficiently many iterations.

4.1. Nested Benders' Decomposition

Standard arguments show that the optimal value of the stage-t nominal cost to-go problem $Q_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ in the nested formulation (1') of the multi-stage robust optimization problem (1) is convex in $\boldsymbol{\xi}_t$. Since the stage-t worst-case cost to-go function $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ maximizes $Q_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ over $\boldsymbol{\xi}_t \in \Xi_t$, we can restrict our attention in the problems (1) and (1') to parameter realizations $\boldsymbol{\xi} \in \operatorname{ext}\Xi$ that are extreme points of the uncertainty set Ξ. This allows us to solve the multi-stage robust optimization problem via nested Benders' decomposition (Louveaux 1980, Birge 1985).

To describe the nested Benders' approach for solving problem (1), let $\hat{\Xi}_1 = \{\boldsymbol{\xi}^1\}$ for the deterministic first-stage parameters $\boldsymbol{\xi}^1$, as well as $\hat{\Xi}_t = \operatorname{ext}\Xi_t$ and $\hat{\Xi}^t = \hat{\Xi}_1 \times \ldots \times \hat{\Xi}_t$, $t = 2,\ldots,T$. By construction, we have $\hat{\Xi}^T = \operatorname{ext}\Xi$. The algorithm can then be formulated as follows.

1. **Initialization:** Set $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}^{t-1}) = -\infty \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$, $\boldsymbol{\xi}^{t-1} \in \hat{\Xi}^{t-1}$, $t = 2,\ldots,T$,

   $\underline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T;\boldsymbol{\xi}^T) = 0 \ \forall \boldsymbol{x}_T \in \mathbb{R}^{n_T}$, $\boldsymbol{\xi}^T \in \hat{\Xi}^T$ (zero terminal costs).

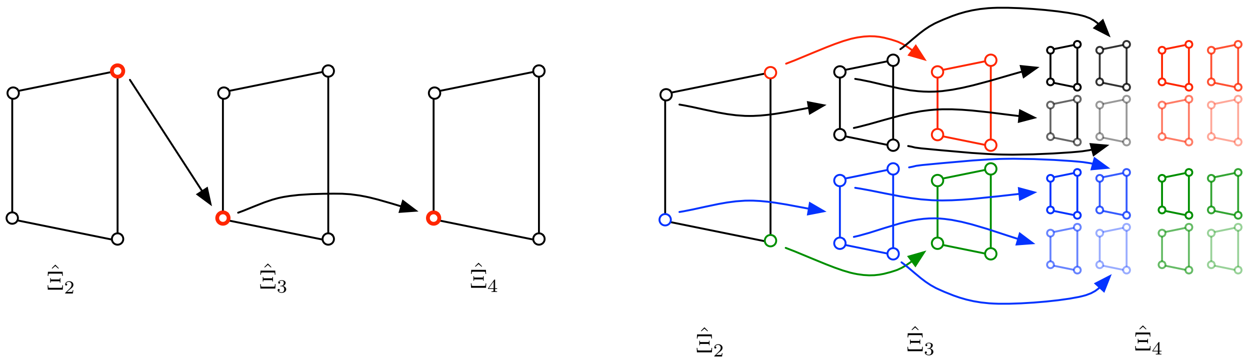


Figure 3 **Left:** For the stage-wise uncertainty sets Ξ_2, Ξ_3 and Ξ_4 (rectangles), our RDDP scheme with the vertex enumeration approach from Section 3.2 considers the $3 \cdot 4 = 12$ extreme points in $\hat{\Xi}_2$, $\hat{\Xi}_3$ and $\hat{\Xi}_4$ (circles). In each time stage t, our algorithm proceeds with the most adverse extreme point $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}}$ (red circles). **Right:** For the same uncertainty sets, the nested Benders' decomposition considers the $4^3 = 64$ extreme point *trajectories* in $\hat{\Xi}_2 \times \hat{\Xi}_3 \times \hat{\Xi}_4$. (For clarity of exposition, only a subset of the arcs is shown.)

2. **Stage-1 Problem:** Let $\underline{\boldsymbol{x}}_1(\boldsymbol{\xi}^1)$ be an optimal solution to the lower bound stage-1 problem

$$\begin{array}{ll} \text{minimize} & \boldsymbol{q}_1^\top \boldsymbol{x}_1 + \underline{\mathcal{Q}}_2(\boldsymbol{x}_1;\boldsymbol{\xi}^1) \\ \text{subject to} & \boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1 \\ & \boldsymbol{x}_1 \in \mathbb{R}^{n_1}. \end{array}$$

3. **Forward Pass:** For $t = 2, \ldots, T-1$ and each $\boldsymbol{\xi}^t \in \hat{\Xi}^t$, let $\underline{\boldsymbol{x}}_t(\boldsymbol{\xi}^t)$ be an optimal solution to the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}(\boldsymbol{\xi}^{t-1});\boldsymbol{\xi}_t)$.

4. **Backward Pass:** For $t = T, \ldots, 2$ and each $\boldsymbol{\xi}^t \in \hat{\Xi}^t$, let $\boldsymbol{\pi}_t$ be the shadow price of an optimal solution to the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}(\boldsymbol{\xi}^{t-1});\boldsymbol{\xi}_t)$. If $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}(\boldsymbol{\xi}^{t-1});\boldsymbol{\xi}_t) > \underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}(\boldsymbol{\xi}^{t-1});\boldsymbol{\xi}^{t-1})$, then update the lower bound

$$\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}^{t-1}) \leftarrow \max\left\{\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}^{t-1}),\ [\boldsymbol{h}_t(\boldsymbol{\xi}_t) - \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\pi}_t - \underline{\mathcal{Q}}^\star_{t+1}(\boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t;\boldsymbol{\xi}^t)\right\}$$

$$\forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}.$$

If Step 4 has been completed for all stages $t = T, \ldots, 2$ and $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1(\boldsymbol{\xi}^1);\boldsymbol{\xi}^1)$ has not been updated, terminate: $\underline{\boldsymbol{x}}_1(\boldsymbol{\xi}^1)$ is optimal with costs $\boldsymbol{q}_1^\top \underline{\boldsymbol{x}}_1(\boldsymbol{\xi}^1) + \underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1(\boldsymbol{\xi}^1);\boldsymbol{\xi}^1)$. Else, go back to Step 2.

Under the conditions (C1) and (C2) from Section 3.3, the correctness and finite convergence of the algorithm follow along the lines of Louveaux (1980) and Birge (1985). The nested Benders' decomposition has to solve lower bound problems $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}(\boldsymbol{\xi}^{t-1}); \boldsymbol{\xi}_t)$ and maintain separate lower cost to-go bounds $\underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1}(\boldsymbol{\xi}^{t-1}); \boldsymbol{\xi}^{t-1})$ for all parameter *trajectories* $\boldsymbol{\xi}^t \in \hat{\Xi}^t = \hat{\Xi}_1 \times \ldots \times \hat{\Xi}_t$, $t = 1, \ldots, T$, in the forward and backward passes of every iteration. By maintaining upper bounds, on the other hand, our RDDP scheme improves upon nested Benders in two ways. Firstly, our RDDP scheme only considers the extreme points in each individual stage-wise uncertainty set $\hat{\Xi}_1$, $\ldots$, $\hat{\Xi}_T$ in every iteration. This enables us to perform forward and backward iterations even in large problems, where the nested Benders decomposition would not complete a single iteration. Secondly, our RDDP scheme is likely to disregard benign trajectories that do not correspond to the worst case. This ensures that under normal circumstances, our RDDP scheme only considers a small fraction of the trajectories over the course of the algorithm, even if the problem is solved to optimality. Figure 3 further illustrates these differences.

Remark 6 (Cut Sharing). For multi-stage robust optimization problems whose constraints only couple decisions of consecutive time periods and whose uncertainty sets are stage-wise rectangular, each cost to-go function $\mathcal{Q}_t$ only depends on the previous state $\boldsymbol{x}_{t-1}$, and a cut produced at the node $\boldsymbol{\xi}^t \in \hat{\Xi}^t$ is valid for all other nodes in period t. In this case, the nested Benders' decomposition can be improved by maintaining a single lower cost to-go bound $\underline{\mathcal{Q}}_t : \mathbb{R}^{n_{t-1}} \mapsto \mathbb{R}$ for each time period. This requires less memory and allows for a faster algorithm execution due to the sharing of cuts between the various subproblems in each stage (Morton 1996, Infanger and Morton 1996). We stress that this improved variant of the nested Benders' decomposition still needs to solve lower bound problems for all parameter trajectories. Indeed, the convergence of the decomposition scheme crucially relies on the cost to-go function approximations eventually becoming tight at all candidate solutions $\underline{\boldsymbol{x}}_t(\boldsymbol{\xi}^t)$ that are optimal responses to some worst-case parameter trajectory $\boldsymbol{\xi} \in \hat{\Xi}_1 \times \ldots \times \hat{\Xi}_T$. In the absence of any information about these worst-case trajectories, the nested Benders' decomposition needs to consider all trajectories $\boldsymbol{\xi} \in \hat{\Xi}_1 \times \ldots \times \hat{\Xi}_T$ in every iteration.

4.2. Stochastic Dual Dynamic Programming

Since we can restrict our attention to extreme point parameter trajectories $\boldsymbol{\xi} \in \operatorname{ext} \Xi$ in problem (1), we can also solve the problem via an adaptation of the well-known stochastic dual dynamic programming (SDDP) approach (Pereira and Pinto 1991, Shapiro 2011, Shapiro et al. 2013). As in the previous subsection, we define $\hat{\Xi}_1 = \{\boldsymbol{\xi}^1\}$, $\hat{\Xi}_t = \operatorname{ext} \Xi_t$ and $\hat{\Xi}^t = \hat{\Xi}_1 \times \ldots \times \hat{\Xi}_t$ for $t = 2, \ldots, T$.

With this notation, the SDDP algorithm for problem (1) can be formulated as follows.

1. **Initialization:** Set $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) = -\infty \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$, $t = 2, \ldots, T$ (lower bound),

   $$\underline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = 0 \ \forall \boldsymbol{x}_T \in \mathbb{R}^{n_T} \text{ (zero terminal costs).}$$

2. **Stage-1 Problem:** Let $\underline{\boldsymbol{x}}_1$ be an optimal solution to the lower bound stage-1 problem

   $$\begin{aligned} \text{minimize} \quad & \boldsymbol{q}_1^\top \boldsymbol{x}_1 + \underline{\mathcal{Q}}_2(\boldsymbol{x}_1) \\ \text{subject to} \quad & \boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1 \\ & \boldsymbol{x}_1 \in \mathbb{R}^{n_1}. \end{aligned}$$

   Stop if some termination criterion is met (see below). Else, go to Step 3.

3. **Forward Pass:** For $t = 2, \ldots, T-1$, select $\overline{\boldsymbol{\xi}}_t^{\text{fw}} \in \hat{\Xi}_t$ randomly and let $\underline{\boldsymbol{x}}_t$ be an optimal solution to the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\text{fw}})$.

4. **Backward Pass:** For $t = T, \ldots, 2$, let $\overline{\boldsymbol{\xi}}_t^{\text{bw}}$ be a maximizer of $\max\{\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \boldsymbol{\xi}_t) : \boldsymbol{\xi}_t \in \Xi_t\}$ and let $\boldsymbol{\pi}_t$ be the shadow price of an optimal solution to the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\text{bw}})$. If $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\text{bw}}) > \underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$, then update the lower bound

   $$\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \leftarrow \max\left\{\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}), \ [\boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\text{bw}}) - \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\text{bw}})\, \boldsymbol{x}_{t-1}]^\top \boldsymbol{\pi}_t - \underline{\mathcal{Q}}_{t+1}^\star(\boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t)\right\} \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}.$$

   After Step 4 has been completed for all stages $t = T, \ldots, 2$, go back to Step 2.

Despite maintaining only a single lower cost to-go bound $\underline{\mathcal{Q}}_t : \mathbb{R}^{n_{t-1}} \mapsto \mathbb{R}$ for each time stage, SDDP avoids the solution of lower bound problems for all parameter trajectories $\boldsymbol{\xi} \in \operatorname{ext} \Xi$ in each iteration by randomly selecting a single parameter trajectory $(\overline{\boldsymbol{\xi}}_1^{\text{fw}}, \ldots, \overline{\boldsymbol{\xi}}_T^{\text{fw}}) \in \operatorname{ext} \Xi$ in every forward pass. Under the conditions (C1) and (C2) from Section 3.3, the SDDP scheme asymptotically converges to an optimal solution of problem (1) with probability 1 as long as every trajectory $\boldsymbol{\xi} \in \operatorname{ext} \Xi$ is selected with positive probability in the forward pass (Shapiro 2011, Proposition 3.1).

The absence of deterministic upper bounds on the optimal value of the multi-stage robust optimization problem (1) requires a different termination criterion. Shapiro (2011) proposes to terminate when the optimal values $\boldsymbol{q}_1^\top \underline{\boldsymbol{x}}_1 + \underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$ of the stage-1 problems stabilize over successive iterations of the algorithm, which amounts to a statistical guarantee that the SDDP scheme will not make significant progress in the next iteration(s). In fact, it appears difficult to design a tractable termination criterion based on the convergence to the optimal value of problem (1).

In conclusion, the nested Benders' decomposition and SDDP primarily differ in their selection of the approximate worst-case parameter trajectories in the forward pass. The nested Benders' decomposition guarantees a finite and deterministic convergence by examining all (exponentially many) parameter trajectories, whereas SDDP achieves an asymptotic and stochastic convergence by randomly selecting a single parameter trajectory in each iteration. Our RDDP scheme, on the other hand, selects approximate worst-case parameter trajectories based on the upper bounds $\overline{\mathcal{Q}}_t$ on the worst-case cost to-go functions $\mathcal{Q}_t$. This allows us to avoid considering all parameter trajectories in each iteration while still maintaining a finite and deterministic convergence. The purpose of the upper bounds in our RDDP scheme is reminiscent of the role of the progressive bounds in branch-and-bound schemes for discrete optimization problems. While we may in principle need to investigate all parameter trajectories $\boldsymbol{\xi} \in \operatorname{ext} \Xi$ during the execution of our RDDP scheme, the upper bounds guide the selection of parameter trajectories towards those that are likely to result in the worst-case costs, and they allow us to exclude benign parameter trajectories from further consideration. Likewise, while a branch-and-bound scheme may in principle need to investigate all feasible solutions, the progressive bounds guide the search towards promising areas of the feasible region, and they allow us to swiftly fathom branches with provably suboptimal solutions.

Remark 7. The absence of upper bounds implies that each iteration of the SDDP scheme can be performed very efficiently. We can exploit this in our RDDP scheme by mixing 'lightweight' SDDP iterations (where the forward passes randomly sample extreme point trajectories) with the more 'heavyweight' RDDP iterations (where the forward passes judiciously choose extreme point trajectories based on the NP-hard upper bound problems). This is particularly advantageous in the early iterations of our RDDP scheme, where the upper bounds have not yet been sufficiently refined and thus carry little information. The finite and deterministic convergence of our RDDP scheme is unaffected by this modification as long as we regularly perform RDDP iterations.

5. Numerical Experiments

We now analyze the computational performance of our RDDP scheme in the context of an inventory management problem. In particular, we will investigate how the runtime of the RDDP scheme is affected by the number of time stages as well as the number of decision variables and uncertain problem parameters in each time stage. We will also compare our RDDP scheme with the solutions obtained from restricting the multi-stage robust optimization problem (1) to affine decision rules (Ben-Tal et al. 2004, Kuhn et al. 2011), as well as the adaptations of the nested Benders' decomposition and SDDP schemes outlined in Section 4.

All optimization problems in this section were solved in single-threaded mode with the Gurobi Optimizer 6.0.3 software package (see `www.gurobi.com`) on a 2.9GHz computer with 8GB RAM. Our C++ implementation of the RDDP scheme uses the vertex enumeration procedure of Section 3.2 to solve the subproblems (2) as well as the gap-based heuristic outlined in Remark 5 to select the next time period in the forward/backward passes.

We consider a single-echelon multi-item inventory management problem with uncertain customer demands and deterministic inventory holding and backlogging costs. The problem setup follows the description in Section 5.1 of Bertsimas and Georghiou (2015). At the beginning of each time period $t = 1, \ldots, T$, the uncertain customer demands $D_{tp}(\boldsymbol{\xi}_t)$ for the products $p = 1, \ldots, P$ are observed. Demands are served, as far as possible, from the inventories $I_{tp}(\boldsymbol{\xi}^t)$. Any excess demand is backlogged at a unit cost of c_{B} per period, and any excess inventory incurs a unit holding cost of c_{H} per period. After observing the demands in period t, the inventory of product p can be restocked through a standard order $x_{tp}(\boldsymbol{\xi}^t)$ arriving at the end of period t (with unit costs c_x) or through an express order $y_{tp}(\boldsymbol{\xi}^t)$ arriving immediately (with unit costs $c_y > c_x$). In the latter case, the deliveries $y_{tp}(\boldsymbol{\xi}^t)$ can be used to serve the demands in period t. We impose product-wise upper bounds B_x and B_y on the standard and express ordering quantities $x_{tp}(\boldsymbol{\xi}^t)$ and $y_{tp}(\boldsymbol{\xi}^t)$, respectively, as well as cumulative bounds B_{C} on the stage-wise express orders $\sum_p y_{tp}(\boldsymbol{\xi}^t)$. We assume that all inventories are empty at the beginning of the planning horizon. The objective is to determine an ordering policy that minimizes the worst-case sum of the ordering, backlogging and inventory holding costs over all anticipated demand realizations.

The problem can be formulated as the following instance of problem (1):

$$\begin{array}{lll} \text{minimize} & \displaystyle\max_{\boldsymbol{\xi}\in\Xi} \sum_{t=1}^{T}\sum_{p=1}^{P} c_x x_{tp}(\boldsymbol{\xi}^t) + c_y y_{tp}(\boldsymbol{\xi}^t) + c_{\mathrm{H}}\left[I_{tp}(\boldsymbol{\xi}^t)\right]_+ + c_{\mathrm{B}}\left[-I_{tp}(\boldsymbol{\xi}^t)\right]_+ & \\ \text{subject to} & \left.\left.\begin{array}{l} I_{tp}(\boldsymbol{\xi}^t) = I_{t-1,p}(\boldsymbol{\xi}^{t-1}) + x_{t-1,p}(\boldsymbol{\xi}^{t-1}) + y_{tp}(\boldsymbol{\xi}^t) - D_{tp}(\boldsymbol{\xi}_t) \\ x_{tp}(\boldsymbol{\xi}^t) \in [0, B_x],\ \ y_{tp}(\boldsymbol{\xi}^t) \in [0, B_y] \end{array}\right\} \forall p = 1,\ldots,P \atop \displaystyle\sum_{p=1}^{P} y_{tp}(\boldsymbol{\xi}^t) \le B_{\mathrm{C}} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \right\} & \begin{array}{l}\forall t = 1,\ldots,T, \\ \forall \boldsymbol{\xi}\in\Xi\end{array} \\ & I_{tp}(\boldsymbol{\xi}^t),\ x_{tp}(\boldsymbol{\xi}^t),\ y_{tp}(\boldsymbol{\xi}^t) \in \mathbb{R},\ \boldsymbol{\xi}\in\Xi,\ t = 1,\ldots,T \text{ and } p = 1,\ldots,P, & \end{array}$$

where we define $I_{0p}(\boldsymbol{\xi}^0) \equiv 0$ and $x_{0p}(\boldsymbol{\xi}^0) \equiv 0$ for $p = 1,\ldots,P$. Note that in this formulation, negative product inventories $I_{tp}(\boldsymbol{\xi}^t)$ correspond to backlogged customer demands. As usual, the maximum operators $[\cdot]_+$ in the objective function can be removed through an epigraph formulation. The problem has a relatively complete recourse since any partial feasible policy (I_{tp}, x_{tp}, y_{tp}), $t = 1,\ldots,\tau$ and $p = 1,\ldots,P$, can be completed to a nonanticipative feasible policy by setting $x_{tp}(\boldsymbol{\xi}^t) = y_{tp}(\boldsymbol{\xi}^t) = 0$ for all $t = \tau+1,\ldots,T$ and $p = 1,\ldots,P$. Moreover, the variable substitution $I_{tp} \leftarrow I_{tp} + x_{tp}$, $t = 1,\ldots,T$ and $p = 1,\ldots,P$, reduces the state variables to P per stage, see Remark 4.

We assume that the product demands in each period are governed by a factor model of the form

$$D_{tp}(\boldsymbol{\xi}_t) = \begin{cases} 2 + \sin\left(\dfrac{\pi(t-1)}{6}\right) + \dfrac{1}{k}\displaystyle\sum_{i=1}^{k} \Phi_{pi}\xi_{ti} & \text{for } p = 1,\ldots,\left\lceil \dfrac{P}{2} \right\rceil, \\ 2 + \cos\left(\dfrac{\pi(t-1)}{6}\right) + \dfrac{1}{k}\displaystyle\sum_{i=1}^{k} \Phi_{pi}\xi_{ti} & \text{for } p = \left\lceil \dfrac{P}{2} \right\rceil + 1,\ldots,P, \end{cases}$$

where $\boldsymbol{\xi}_t \in \Xi_t = [-1,1]^k$ and the factor loading coefficients Φ_{pi} are chosen uniformly at random from $[-1,1]$. By construction, the product demands thus satisfy $D_{tp}(\boldsymbol{\xi}_t) \in [0,4]$ for all $t = 1,\ldots,T$, $p = 1,\ldots,P$ and $\boldsymbol{\xi}\in\Xi$. We set $c_x = 0$ and select $c_y \in [0,2]$ uniformly at random. The absence of standard ordering costs reflects the idea that neither the inventory nor the backlogged demands should grow indefinitely over time, which implies that the cumulative standard ordering quantities should be proportional to the observed demands and thus represent sunk costs. Instead, the decision maker aims to minimize the overall number of express deliveries, which are charged at a premium of c_y per unit. We choose $c_{\mathrm{H}}, c_{\mathrm{B}} \in [0,2]$ uniformly at random, we set the product-wise upper bounds to $B_x = B_y = 10$, and we fix $B_{\mathrm{C}} = 0.3P$. In each of the numerical experiments below, we generate 25 random problem instances according to this procedure.

Instance	Trajectories	Runtime	Memory
5-4-3	256	1.3s	18MB
5-4-4	4,096	44.6s	260MB
5-4-5	65,536	924.23s	20.2GB
5-4-6	1,048,576	—	—

Table 1 Number of parameter trajectories as well as the runtime and memory requirements of the nested Benders' decomposition for instances with 5 products, 4 primitive uncertainties and 3–6 time stages. The last instance class could not be solved as the computer ran out of memory.

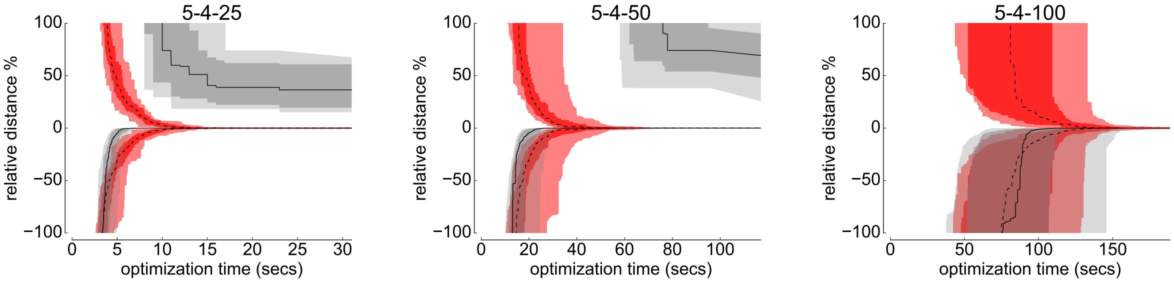


Figure 4 Comparison of our RDDP scheme with the affine decision rules and SDDP for instances with 5 products, 4 primitive uncertainties and 10 (left), 50 (middle) and 100 (right) time stages. The dashed (bold) line represents the median, the dark red (gray) region the range spanned by the first and third quartile, and the light red (gray) region the range spanned by the 10% and 90% quantile of the suboptimalities of the upper and lower bounds of the RDDP scheme (affine decision rules and SDDP).

In our first experiment, we fix the number of products to $P = 5$ and the number of primitive uncertainties to $k = 4$ per stage, and we investigate the scalability of the nested Benders' decomposition, our RDDP scheme, the affine decision rule approximation and SDDP as the planning horizon T varies. The results are presented in Table 1, Figure 4 and the first part of Table EC.1 in the e-companion. Each instance class is identified by the label P–k–T, where P, k and T refer to the numbers of products, primitive uncertainties per stage and time stages, respectively. Figure 4 reports the suboptimality of the upper and lower bounds of our RDDP scheme, the upper bound provided by the affine decision rule approximation as well as the lower bound offered by SDDP as a function of the optimization time. Table EC.1 reports the corresponding optimality gaps, which we define as the sums of the individual suboptimalities of the respective upper and lower bounds.

Table 1 reveals that the nested Benders' decomposition is only viable for small problem instances. In fact, in each iteration the algorithm has to solve lower bound problems and store value function

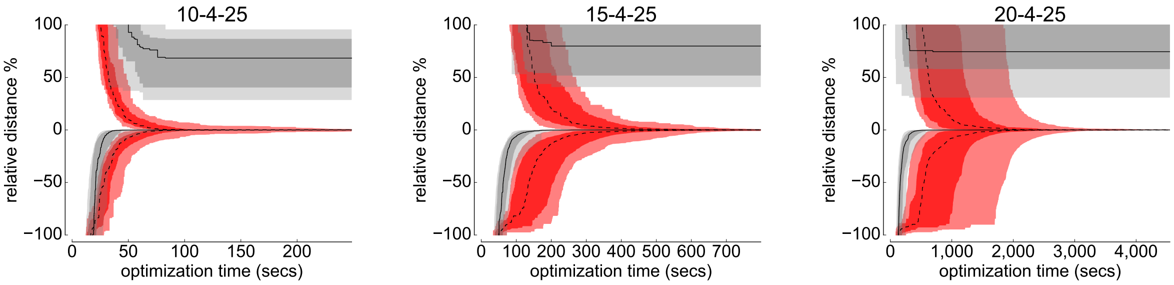


Figure 5 Comparison of our RDDP scheme with the affine decision rules and SDDP for instances with 25 time stages, 4 primitive uncertainties and 10 (left), 15 (middle) and 20 (right) products. The graphs have the same interpretation as in Figure 4.

approximations for all of the $(2^k)^{T-1}$ parameter trajectories. This implies that both the runtimes and the memory requirements grow quickly as the problem size increases. Perhaps more surprisingly, Figure 4 and Table EC.1 show that our RDDP scheme scales more gracefully with the number of time stages than the affine decision rule approximation. Indeed, we were not able to solve the affine decision rule problems for $T \geq 75$ stages since the solver ran out of memory. This is owed to the fact that the size of the optimization problems associated with the affine decision rule approximation is of the order $\mathcal{O}(PkT^2)$ and thus grow quadratically in the time horizon T. In contrast, the vertex enumeration procedure of our RDDP scheme has to solve $\mathcal{O}(2^kT)$ linear programs of size $\mathcal{O}(P)$ in each iteration. While the number of iterations in our RDDP scheme can in principle scale exponentially in the number of time stages T, this theoretical worst-case behavior has not been observed in the experiments. We also observe that affine decision rules incur a significant optimality gap. Indeed, while affine decision rules are optimal if we chose $P = 1$ and $B_y = B_C = 0$, see Bertsimas et al. (2010), they are no longer guaranteed to be optimal for $P > 1$ and/or $B_y, B_C \neq 0$ (Bertsimas et al. 2010, §4.5). Finally, we observe that SDDP converges faster than our RDDP scheme for the smaller instances, while it requires approximately the same time for the larger instances. We stress, however, that SDDP only offers lower bounds on the optimal objective value, whereas our RDDP scheme simultaneously computes both lower and upper bounds within a comparable amount of time. We will revisit this aspect further below.

In our second experiment, we fix the planning horizon to $T = 25$ periods and the number of primitive uncertainties to $k = 4$ per stage, and we investigate the scalability of our RDDP scheme,

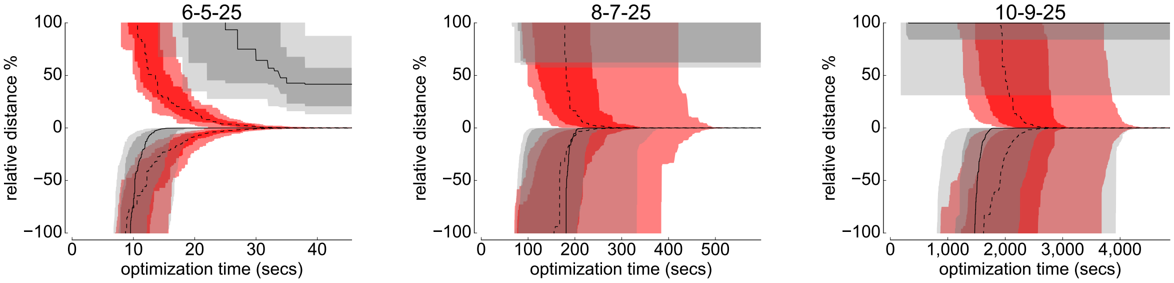


Figure 6 Comparison of our RDDP scheme with the affine decision rules and SDDP for instances with a planning horizon of 25 stages and $(P,k)=(6,5),(8,7),(10,9)$ products and primitive uncertainties (from left to right). The graphs have the same interpretation as in Figure 4.

	Initial inventories $I_{0p}(\boldsymbol{\xi}^0)$									
Order	20%		25%		30%		35%		40%	
frequency Δ	Solved	Gap	Solved	Gap	Solved	Gap	Solved	Gap	Solved	Gap
5	70%	18%	60%	20%	40%	20%	20%	72%	0%	100%
7	20%	13%	50%	17%	80%	5%	10%	26%	0%	100%
10	0%	14%	0%	14%	20%	18%	10%	23%	10%	73%

Table 2 Performance of the SDDP scheme in the modified inventory management problem. The sizes of the initial inventories are relative to the worst-case cumulative product demands. For each instance class specified by $I_{0p}(\boldsymbol{\xi}^0)$ and Δ, the table provides the percentage of instances solved to optimality ('solved') as well as the average optimality gap for the remaining instances ('gap'). All results are averages over 10 randomly generated instances.

the affine decision rule approximation and SDDP as the number of products P varies. The results are shown in Figure 5 and the second part of Table EC.1. The runtimes of the affine decision rule approximation are more competitive than in the previous experiment, which is expected from the previous complexity estimates. Nevertheless, the gap between the upper bound provided by the affine decision rule approximation and the lower bound offered by SDDP remains disappointing.

In our third experiment, we fix the planning horizon to $T=25$ periods, and we simultaneously vary the number k of primitive uncertainties per stage as well as the number of products P. The results are shown in Figure 6 and the third part of Table EC.1. As expected, the resulting problems are considerably more difficult to solve. Nevertheless, our RDDP scheme is able to close to gap to about 1% in all of the considered instances within one hour.

While being unable to prove optimality, the lower bounds provided by the SDDP scheme have been tight for all of the problems studied so far. To see that this is not the case in general, our last experiment considers a variant of our inventory management problem in which $(c_x, B_x) = (1, +\infty)$, $(c_y, B_y, B_{\mathrm{C}}) = (0, 0, 0)$, $c_{\mathrm{H}} = 0$ and $c_{\mathrm{B}} = +\infty$. In other words, we wish to determine the minimal amount of standard deliveries that satisfy all of the demand instantaneously. For such instances to be feasible, we allow for strictly positive initial inventories $I_{0p}(\boldsymbol{\xi}^0)$, $p = 1, \ldots, P$. We furthermore assume that the product demands are governed by a factor model as before, but we augment the uncertainty set by a 'high demand' scenario $\overline{\boldsymbol{\xi}}_t$ in which $D_{tp}(\overline{\boldsymbol{\xi}}_t) = 20$ for all products $p = 1, \ldots, P$. Finally, we assume that orders are only allowed every $\Delta \in \mathbb{N}$ periods. Table 2 shows that the SDDP scheme typically fails to find the optimal solutions in this problem variant within one hour runtime, despite 'converging' within one minute in all of the instances. This is due to *(i)* the fact that the worst-case costs are determined by a single demand trajectory (which is typically missed if sampling randomly) and *(ii)* the presence of initial inventories and order restrictions. Taken together, both properties imply that many iterations (using the 'right' parameter trajectories) are needed to relay information about the optimal decisions across the time stages in the backward passes. Note that the suboptimality cannot be detected by the SDDP scheme since the method does not provide upper bounds. Our RDDP scheme correctly identifies the optimal solution in all of these instances.

6. Conclusions

We presented a 'vanilla version' of the robust dual dynamic programming (RDDP) scheme that can serve as a blueprint for various improvements and extensions. Indeed, we believe that an attractive feature of our scheme is its versatility, which opens up several promising avenues for future research.

From a computational perspective, the solution of the max-min subproblems (2) deserves further attention. The vertex enumeration scheme presented in Section 3.2, for example, can benefit from various speedup techniques such as warm-starting and bunching (Birge and Louveaux 1997, §5.4). More generally, the convergence of our RDDP scheme is not affected if the subproblems are solved suboptimally in early stages of the algorithm's execution. This observation enables us to solve the subproblems as two-stage robust optimization problem in (piecewise) linear decision rules.

From a methodological perspective, it would be interesting to explore the extensions of the RDDP scheme to multi-stage robust optimization problems with discrete recourse decisions (Zou et al. 2016) as well as multi-stage distributionally robust optimization problems (Delage and Ye 2010, Wiesemann et al. 2014). In the latter case, the connections between the decomposability of problem (1) into the nested formulation (1') and the theory of rectangular ambiguity sets (Shapiro 2016) and time consistent dynamic risk measures (Shapiro et al. 2014) are of particular interest.

Acknowledgments

The authors gratefully acknowledge funding from the Imperial Business Analytics Centre and the EPSRC grants EP/M028240/1 and EP/M027856/1, as well as the constructive comments of the associate editor and the two referees that led to significant improvements of the manuscript.

References

Anderson, B. D. O., J. B. Moore. 1990. *Optimal Control: Linear Quadratic Methods*. Dover.

Ayoub, J., M. Poss. 2016. Decomposition for adjustable robust linear optimization subject to uncertainty polytope. *Computational Management Science* **13**(2) 219–239.

Ben-Tal, A., A. Goryashko, E. Guslitzer, A. Nemirovski. 2004. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming A* **99**(2) 351–376.

Bertsekas, D. P. 2007. *Dynamic Programming and Optimal Control*. Athena Scientific.

Bertsimas, D., I. Dunning. 2016. Multistage robust mixed integer optimization with adaptive partitions. *Operations Research* **64**(4) 980–998.

Bertsimas, D., A. Georghiou. 2015. Design of near optimal decision rules in multistage adaptive mixed-integer optimization. *Operations Research* **63**(3) 610–627.

Bertsimas, D., D. A. Iancu, P. A. Parrilo. 2010. Optimality of affine policies in multi-stage robust optimization. *Mathematics of Operations Research* **35**(2) 363–394.

Bertsimas, D., D. A. Iancu, P. A. Parrilo. 2011. A hierarchy of near-optimal policies for multistage adaptive optimization. *IEEE Transactions on Automatic Control* **56**(12) 2809–2824.

Bertsimas, D., E. Litvinov, X. A. Sun, J. Zhao, T. Zheng. 2013. Adaptive robust optimization for the security constrained unit commitment problem. *IEEE Transactions on Power Systems* **28**(1) 52–63.

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

Birge, J. R., F. Louveaux. 1997. *Introduction to Stochastic Programming*. Springer.

Chen, X., M. Sim, P. Sun, J. Zhang. 2008. A linear decision-based approximation approach to stochastic programming. *Operations Research* **56**(2) 344–357.

Chen, X., Y. Zhang. 2009. Uncertain linear programs: Extended affinely adjustable robust counterparts. *Operations Research* **57**(6) 1469–1482.

Dantzig, G. B. 1963. *Linear Programming and Extensions*. Princeton University Press.

de Ruiter, F. J. C. T., R. C. M. Brekelmans, D. den Hertog. 2016. The impact of the existence of multiple adjustable robust solutions. *Mathematical Programming,* **160**(1–2) 531–545.

Delage, E., D. A. Iancu. 2015. Robust multistage decision making. *INFORMS Tutorials in Operations Research* 20–46.

Delage, E., Y. Ye. 2010. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research* **58**(3) 596–612.

Dempe, S. 2002. *Foundations of Bilevel Programming*. Kluwer Academic Publishers.

Garey, M. R., D. S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.

Georghiou, A., W. Wiesemann, D. Kuhn. 2015. Generalized decision rule approximations for stochastic programming via liftings. *Mathematical Programming* **152**(1) 301–338.

Goh, J., M. Sim. 2010. Distributionally robust optimization and its tractable approximations. *Operations Research* **58**(4) 902–917.

Gounaris, C. E., W. Wiesemann, C. A. Floudas. 2013. The robust capacitated vehicle routing problem under demand uncertainty. *Operations Research* **61**(3) 677–603.

Guslitser, E. 2002. Uncertainty-immunized solutions in linear programming. Master's thesis, Technion.

Hadjiyiannis, M. J., P. J. Goulart, D. Kuhn. 2011. A scenario approach for estimating the suboptimality of linear decision rules in two-stage robust optimization. *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*.

Iancu, D. A., N. Trichakis. 2014. Pareto efficiency in robust optimization. *Management Science* **60**(1) 130–147.

Infanger, G., D. P. Morton. 1996. Cut sharing for multistage stochastic linear programs with interstage dependency. *Mathematical Programming* **75**(2) 241–256.

Jiang, R., M. Zhang, G. Li, Y. Guan. 2010. Two-stage robust power grid optimization problem. Available on Optimization Online.

Kuhn, D., W. Wiesemann, A. Georghiou. 2011. Primal and dual linear decision rules in stochastic and robust optimization. *Mathematical Programming* **130**(1) 177–209.

Limongelli, C., R. Pirastu. 1994. Exact solution of linear equation systems over rational numbers by parallel p-adic arithmetic. RISC Report Series 94–25, University of Linz.

Louveaux, F. V. 1980. A solution method for multistage stochastic programs with recourse with application to an energy investment problem. *Operations Research* **28**(4) 889–902.

Magnanti, T. L., J. F. Shapiro, M. H. Wagner. 1976. Generalized linear programming solves the dual. *Management Science* **22**(11) 1195–1203.

Morton, D. P. 1996. An enhanced decomposition algorithm for multistage stochastic hydroelectric scheduling. *Annals of Operations Research* **64**(1) 211–235.

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

Postek, K. S., D. den Hertog. 2016. Multi-stage adjustable robust mixed-integer optimization via iterative splitting of the uncertainty set. *INFORMS Journal on Computing* **28**(3) 553–574.

Rockafellar, R. T., R. J.-B. Wets. 2009. *Variational Analysis*. 3rd ed. Springer.

Shapiro, A. 2011. Analysis of stochastic dual dynamic programming method. *European Journal of Operational Research* **209**(1) 63–72.

Shapiro, A. 2016. Rectangular sets of probability measures. *Operations Research* **64**(2) 528–541.

Shapiro, A., D. Dentcheva, A. Ruszczyski. 2014. *Lectures on Stochatic Programming: Modeling and Theory*. 2nd ed. SIAM.

Shapiro, A., W. Tekaya, M. P. Soares, J. P. da Costa. 2013. Worst-case-expectation approach to optimization under uncertainty. *Operations Research* **61**(6) 1435–1449.

Thiele, A., T. Terry, M. Epelman. 2010. Robust linear optimization with recourse. Tech. rep., Lehigh University and University of Michigan.

Wiesemann, W., D. Kuhn, M. Sim. 2014. Distributionally robust convex optimization. *Operations Research* **62**(6) 1358–1376.

Zeng, B., L. Zhao. 2013. Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters* **41**(5) 457–561.

Zhao, C., J. Wang, J.-P. Watson, Y. Guan. 2013. Multi-stage robust unit commitment considering wind and demand response uncertainties. *IEEE Transactions on Power Systems* **28**(3) 2708–2717.

Zhen, J., D. den Hertog, M. Sim. 2016. Adjustable robust optimization via Fourier-Motzkin elimination. Available on Optimization Online.

Zou, J., S. Ahmed, X. A. Sun. 2016. Nested decomposition of multistage stochastic integer programs with binary state variables. Available on Optimization Online.

Electronic Companion

EC.1. Proof of Theorem 1

We first recall the Integer Programming (IP) Feasibility problem:

Integer Programming Feasibility.

Instance. Given are $\boldsymbol{F} \in \mathbb{Z}^{l \times k}$ and $\boldsymbol{g} \in \mathbb{Z}^l$.

Question. Is there a vector $\boldsymbol{z} \in \{0,1\}^k$ such that $\boldsymbol{F}\boldsymbol{z} \leq \boldsymbol{g}$?

The IP Feasibility problem is well-known to be strongly NP-hard (Garey and Johnson 1979).

Proof of Theorem 1. We claim that the following problem can be cast as a stage-t worst-case cost problem $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ with an uncertain technology matrix $\boldsymbol{T}_t$ or uncertain right-hand sides $\boldsymbol{h}_t$:

$$\begin{array}{ll} \text{maximize} & \min\limits_{\boldsymbol{x} \in \mathbb{R}^k} \left\{ \mathbf{e}^\top \boldsymbol{x} : \boldsymbol{x} \geq \boldsymbol{\xi},\ \boldsymbol{x} \geq \mathbf{e} - \boldsymbol{\xi} \right\} \\ \text{subject to} & \boldsymbol{F}\boldsymbol{\xi} \leq \boldsymbol{g} \\ & \boldsymbol{\xi} \in [0,1]^k \end{array} \tag{EC.1}$$

Indeed, for the stage-t worst-case cost problem $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ with uncertain technology matrix $\boldsymbol{T}_t$, we set $n_t = k_t = k$, $m_t = 2k$, $\Xi_t = \{\boldsymbol{\xi}_t \in [0,1]^{k_t} : \boldsymbol{F}\boldsymbol{\xi}_t \leq \boldsymbol{g}\}$ and

$$\boldsymbol{q}_t = \mathbf{e} \in \mathbb{R}^k, \quad \mathcal{Q}_{t+1}(\boldsymbol{x}_t) = 0\ \forall \boldsymbol{x}_t \in \mathbb{R}^{n_t}, \quad \boldsymbol{T}_t(\boldsymbol{\xi}_t) = \begin{bmatrix} \operatorname{diag}(-\boldsymbol{\xi}_t) & \mathbf{0} \\ \mathbf{0} & \operatorname{diag}(\boldsymbol{\xi}_t) \end{bmatrix} \in \mathbb{R}^{2k \times 2k},$$

$$\boldsymbol{x}_{t-1} = \mathbf{e} \in \mathbb{R}^{2k}, \quad \boldsymbol{W}_t = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{I} \end{bmatrix} \in \mathbb{R}^{2k \times k} \quad \text{and} \quad \boldsymbol{h}_t(\boldsymbol{\xi}_t) = \begin{bmatrix} \mathbf{0} \\ \mathbf{e} \end{bmatrix} \in \mathbb{R}^{2k},$$

while for the stage-t worst-case cost problem $\mathcal{Q}_t(\boldsymbol{x}_{t-1})$ with uncertain right-hand sides $\boldsymbol{h}_t$, we set $n_t = k_t = k$, $m_t = 2k$, $\Xi_t = \{\boldsymbol{\xi}_t \in [0,1]^{k_t} : \boldsymbol{F}\boldsymbol{\xi}_t \leq \boldsymbol{g}\}$ and

$$\boldsymbol{q}_t = \mathbf{e} \in \mathbb{R}^k, \quad \mathcal{Q}_{t+1}(\boldsymbol{x}_t) = 0\ \forall \boldsymbol{x}_t \in \mathbb{R}^{n_t}, \quad \boldsymbol{T}_t(\boldsymbol{\xi}_t) = \mathbf{0} \in \mathbb{R}^{2k \times 2k},$$

$$\boldsymbol{x}_{t-1} = \mathbf{0} \in \mathbb{R}^{2k}, \quad \boldsymbol{W}_t = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{I} \end{bmatrix} \in \mathbb{R}^{2k \times k} \quad \text{and} \quad \boldsymbol{h}_t(\boldsymbol{\xi}_t) = \begin{bmatrix} \boldsymbol{\xi}_t \\ \mathbf{e} - \boldsymbol{\xi}_t \end{bmatrix} \in \mathbb{R}^{2k}.$$

We now show that the answer to an instance $(\boldsymbol{F}, \boldsymbol{g})$ of the IP Feasibility problem is affirmative if and only if the optimal value of the corresponding instance of (EC.1) is k. Assume first that the IP Feasibility problem is solved by $\boldsymbol{z} \in \{0,1\}^k$. By construction, $\boldsymbol{z}$ is feasible in (EC.1) and attains the objective value k. At the same time, the optimal value of any instance of (EC.1) is at most k since $\boldsymbol{x} = \mathbf{e}$ is always feasible in the inner minimization problem of (EC.1). We thus conclude that the optimal value of (EC.1) is indeed k, as desired. Assume now that the optimal value of (EC.1) is k. In that case, there must be a feasible solution $\boldsymbol{\xi} \in [0,1]^k$ with $\boldsymbol{F}\boldsymbol{\xi} \leq \boldsymbol{g}$ for which the inner minimization problem is optimized by $\boldsymbol{x} = \mathbf{e}$. From the constraints of the minimization problem we then conclude that $\boldsymbol{\xi} \in \{0,1\}^k$, that is, $\boldsymbol{\xi}$ solves the IP Feasibility problem. □

EC.2. Multi-Stage Robust Optimization without Relatively Complete Recourse

In this section we relax the assumption (A2) from the main paper, which stipulates that the multi-stage robust optimization problem (1) has a relatively complete recourse. We continue to assume that problem (1) satisfies the other three conditions (A1), (A3) and (A4).

Since problem (1) no longer satisfies (A2), the worst-case cost to-go functions $\mathcal{Q}_t$ become extended real-valued functions that attain the value $\mathcal{Q}_t(\boldsymbol{x}_{t-1}) = +\infty$ whenever the decision $\boldsymbol{x}_{t-1}$ cannot be completed to a feasible solution $\boldsymbol{x}_{t-1}, \ldots, \boldsymbol{x}_T$ for all subsequent realizations $\boldsymbol{\xi}_t, \ldots, \boldsymbol{\xi}_T$ of the uncertain parameters. While it turns out that the upper and lower bounds $\overline{\mathcal{Q}}_t$ and $\underline{\mathcal{Q}}_t$ from the main paper retain their bounding property in this more general setting, the lower bounds by construction never attain the value $+\infty$, and the algorithm may thus fail to identify an optimal solution. To address this issue, we will replace the lower bounds $\underline{\mathcal{Q}}_t$ with $\underline{\mathcal{Q}}_t + \delta_{\hat{\mathcal{X}}_t}$, where $\hat{\mathcal{X}}_t$ represents an outer approximation of the *stage-t feasible region* $\mathcal{X}_t = \{\boldsymbol{x}_t \in \mathbb{R}^{n_t} : \mathcal{Q}_{t+1}(\boldsymbol{x}_t) < +\infty\}$.

One readily verifies that the stage-t feasible region satisfies the recursion $\mathcal{X}_T = \mathbb{R}^{n_T}$ and

$$\mathcal{X}_{t-1} = \left\{ \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} : \begin{bmatrix} \forall \boldsymbol{\xi}_t \in \Xi_t \ \exists \boldsymbol{x}_t \in \mathcal{X}_t : \\ \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \end{bmatrix} \right\}, \ t = T, \ldots, 2.$$

Thus, for a previously implemented stage-$(t-1)$ decision $\boldsymbol{x}_{t-1}$ and a parameter realization $\boldsymbol{\xi}_t$, a stage-t decision $\boldsymbol{x}_t \in \mathbb{R}^{n_t}$ is feasible in stage t *and* can be extended to a feasible solution $\boldsymbol{x}_t, \ldots, \boldsymbol{x}_T$ if and only if $\boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t)$ and $\boldsymbol{x}_t \in \mathcal{X}_t$. Likewise, a first-stage decision $\boldsymbol{x}_1 \in \mathbb{R}^{n_1}$ is feasible and extendable to a complete decision policy if and only if $\boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1$ and $\boldsymbol{x}_1 \in \mathcal{X}_1$.

Similar to the stage-wise worst-case cost to-go functions $\mathcal{Q}_t$, the stage-wise feasible regions $\mathcal{X}_t$ are unknown for $t < T$. We therefore have to confine ourselves to outer approximations $\hat{\mathcal{X}}_t$ of $\mathcal{X}_t$ that are refined throughout the algorithm. To this end, we define the *stage-t feasibility problem* for a given outer approximation $\hat{\mathcal{X}}_t$ of $\mathcal{X}_t$ as

$$\begin{aligned} \overline{\mathcal{Q}}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1}) \ = \ \text{maximize} \quad & \begin{bmatrix} \text{minimize} & \mathbf{e}^\top \boldsymbol{v}_t \\ \text{subject to} & \boldsymbol{T}_t(\boldsymbol{\xi}_t)\, \boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t + \boldsymbol{v}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \\ & \boldsymbol{x}_t \in \hat{\mathcal{X}}_t, \ \boldsymbol{v}_t \in \mathbb{R}^{m_t}_+ \end{bmatrix} \\ \text{subject to} \quad & \boldsymbol{\xi}_t \in \Xi_t. \end{aligned}$$

The stage-t feasibility problem $\overline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1})$ closely resembles the upper bound problem $\overline{Q}_t(\boldsymbol{x}_{t-1})$, except that it is restricted to second-stage decisions $\boldsymbol{x}_t \in \hat{\mathcal{X}}_t$ and that its objective function minimizes the sum of the constraint violations incurred in $\overline{Q}_t(\boldsymbol{x}_{t-1})$. Since we only solve the problem when $\hat{\mathcal{X}}_t \neq \emptyset$, the inner minimization in $\overline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1})$ is guaranteed to be feasible for every realization of the uncertain parameters even if $\overline{Q}_t(\boldsymbol{x}_{t-1})$ is infeasible, which allows us to solve the problem via mixed-integer linear programming techniques or vertex enumeration, see Section 3.2.

Similar to the lower bound problem $\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$, we will utilize the optimal solution to the second-stage problem in $\overline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1})$ for a fixed parameter realization $\boldsymbol{\xi}_t \in \Xi_t$, which we obtain from

$$\underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t) \;=\; \left[\begin{array}{ll} \text{minimize} & \mathbf{e}^\top \boldsymbol{v}_t \\ \text{subject to} & \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1} + \boldsymbol{W}_t\boldsymbol{x}_t + \boldsymbol{v}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t) \\ & \boldsymbol{x}_t \in \hat{\mathcal{X}}_t,\;\; \boldsymbol{v}_t \in \mathbb{R}^{m_t}_+ \end{array}\right].$$

For a polyhedral outer approximation $\hat{\mathcal{X}}_t$, this problem is a linear program with the associated dual

$$\begin{array}{ll} \text{maximize} & [\boldsymbol{h}_t(\boldsymbol{\xi}_t) - \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\mu}_t - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \\ \text{subject to} & \boldsymbol{\mu}_t \in [0,1]^{m_t}, \end{array}$$

where $\sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{y}_t) = \sup\{\boldsymbol{x}_t^\top \boldsymbol{y}_t : \boldsymbol{x}_t \in \hat{\mathcal{X}}_t\}$ is the support function of $\hat{\mathcal{X}}_t$. Strong duality holds between $\underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ and its dual problem since $\underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ is feasible and $\boldsymbol{\mu}_t = \mathbf{0}$ is feasible in the dual.

We continue to use the upper and lower bound problems $\overline{Q}_t$ and $\underline{Q}_t$ from the main paper, but we replace the domains $\boldsymbol{x}_t \in \mathbb{R}^{n_t}$ in the lower bound problems with $\boldsymbol{x}_t \in \hat{\mathcal{X}}_t$. We will only solve the upper bound problems when $\overline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1}) = 0$, which implies that the inner minimization in the upper bound problems will be feasible for every realization of the uncertain parameters, and we can solve them via mixed-integer linear programming or vertex enumeration, see Section 3.2. The incorporation of $\hat{\mathcal{X}}_t$ in the lower bound problem $\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ leads to the new dual:

$$\begin{array}{ll} \text{maximize} & [\boldsymbol{h}_t(\boldsymbol{\xi}_t) - \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\pi}_t - \underline{\mathcal{Q}}_{t+1}^\star(\boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t - \boldsymbol{y}_t) - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{y}_t) \\ \text{subject to} & \boldsymbol{\pi}_t \in \mathbb{R}^{m_t}_+,\;\; \boldsymbol{y}_t \in \mathbb{R}^{n_t} \end{array}$$

In the following, strong duality will hold between $\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ and its dual since we only solve $\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ when $\underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t) = 0$, in which case $\underline{Q}_t(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ is feasible. Note that $\sigma_{\mathbb{R}^{n_t}}(\boldsymbol{y}_t) = 0$ if

$\boldsymbol{y}_t = \boldsymbol{0}$ and $\sigma_{\mathbb{R}^{n_t}}(\boldsymbol{y}_t) = +\infty$ otherwise. Thus, the new dual problem reduces to the dual problem from the main paper if we set $\hat{\mathcal{X}}_t = \mathbb{R}^{n_t}$ and $\boldsymbol{y}_t = \boldsymbol{0}$.

We now present our revised RDDP scheme for instances of (1) violating the assumption (A2):

1. **Initialization:** Set $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) = -\infty \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$, $t = 2, \ldots, T$ (lower bound),

   $\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) = +\infty \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$, $t = 2, \ldots, T$ (upper bound),

   $\underline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = \overline{\mathcal{Q}}_{T+1}(\boldsymbol{x}_T) = 0 \ \forall \boldsymbol{x}_T \in \mathbb{R}^{n_T}$ (zero terminal costs),

   $\hat{\mathcal{X}}_t = \mathbb{R}^{n_t}$, $t = 1, \ldots, T$ (outer approximation of feasible region).

2. **Stage-1 Problem:** Let $\underline{\boldsymbol{x}}_1$ be an optimal solution to the lower bound stage-1 problem

$$\begin{array}{ll} \text{minimize} & \boldsymbol{q}_1^\top \boldsymbol{x}_1 + \underline{\mathcal{Q}}_2(\boldsymbol{x}_1) \\ \text{subject to} & \boldsymbol{W}_1 \boldsymbol{x}_1 \geq \boldsymbol{h}_1 \\ & \boldsymbol{x}_1 \in \hat{\mathcal{X}}_1. \end{array}$$

   If this problem is infeasible, terminate: problem (1) is infeasible. Otherwise, if $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$, terminate: $\underline{\boldsymbol{x}}_1$ is optimal with costs $\boldsymbol{q}_1^\top \underline{\boldsymbol{x}}_1 + \underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$. Else, set $t = 2$ and go to Step 3.

3. **Forward Pass:** Let $\boldsymbol{\xi}_t^{\mathrm{F}}$ be an optimal solution to the stage-t feasibility problem $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1})$.

   (a) If $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}) > 0$, then let $\boldsymbol{\mu}_t$ be the shadow price of an optimal solution to the problem $\underline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}; \boldsymbol{\xi}_t^{\mathrm{F}})$ and update the outer approximation of the stage-$(t-1)$ feasible region

$$\hat{\mathcal{X}}_{t-1} \leftarrow \hat{\mathcal{X}}_{t-1} \cap \left\{ \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} \, : \, [\boldsymbol{h}_t(\boldsymbol{\xi}_t^{\mathrm{F}}) - \boldsymbol{T}_t(\boldsymbol{\xi}_t^{\mathrm{F}})\, \boldsymbol{x}_{t-1}]^\top \boldsymbol{\mu}_t - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \leq 0 \right\}.$$

   If $\hat{\mathcal{X}}_{t-1} = \emptyset$, terminate: problem (1) is infeasible. Otherwise, repeat Step 3 for stage $t-1$ (if $t > 2$) or go back to Step 2 (if $t = 2$).

   (b) If $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}) = 0$, then let $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}}$ be optimal in $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$, let $\underline{\boldsymbol{x}}_t$ be optimal in $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{fw}})$, and repeat Step 3 for stage $t+1$ (if $t < T-1$) or go to Step 4 (if $t = T-1$).

4. **Backward Pass:** For $t = T, \ldots, 2$, let $\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}$ be an optimal solution to the upper bound problem $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$. If $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}) < \overline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$, then update the upper bound

$$\overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \leftarrow \operatorname{env}\left(\min\left\{ \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}),\, \overline{Q}_t(\underline{\boldsymbol{x}}_{t-1}) + \delta_{\{\underline{\boldsymbol{x}}_{t-1}\}}(\boldsymbol{x}_{t-1}) \right\} \right) \qquad \forall \boldsymbol{x}_{t-1} \in \mathbb{R}_{n_{t-1}}.$$

   Let $(\boldsymbol{\pi}_t, \boldsymbol{y}_t)$ be the shadow price of an optimal solution to the lower bound problem $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})$. If $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) > \underline{\mathcal{Q}}_t(\underline{\boldsymbol{x}}_{t-1})$, then update the lower bound

$$\begin{aligned} \underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \leftarrow \max\Big\{ & \underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}),\, [\boldsymbol{h}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}) - \boldsymbol{T}_t(\overline{\boldsymbol{\xi}}_t^{\mathrm{bw}})\, \boldsymbol{x}_{t-1}]^\top \boldsymbol{\pi}_t - \\ & \underline{\mathcal{Q}}_{t+1}^\star(\boldsymbol{W}_t^\top \boldsymbol{\pi}_t - \boldsymbol{q}_t - \boldsymbol{y}_t) - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{y}_t) \Big\} \ \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}. \end{aligned} \qquad \text{(EC.2)}$$

After Step 4 has been completed for all stages $t = T, \ldots, 2$, go back to Step 2.

The algorithm largely resembles the RDDP scheme from the main paper. The key difference lies in the forward pass, which now iteratively constructs outer approximations $\hat{\mathcal{X}}_t$ of the stage-t feasible regions $\mathcal{X}_t$. To this end, we ensure that in each stage $t = 2, \ldots, T$, the selected decision $\underline{\boldsymbol{x}}_{t-1}$ can be extended to a decision $\boldsymbol{x}_t \in \hat{\mathcal{X}}_t$ satisfying $\boldsymbol{T}_t(\boldsymbol{\xi}_t)\underline{\boldsymbol{x}}_{t-1} + \boldsymbol{W}_t\boldsymbol{x}_t \geq \boldsymbol{h}_t(\boldsymbol{\xi}_t)$ under each possible parameter realization $\boldsymbol{\xi}_t \in \Xi_t$. If this is not possible, which is the case if and only if the feasibility problem $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1})$ does not evaluate to zero, then the previously selected decision $\underline{\boldsymbol{x}}_{t-1}$ cannot be completed to a nonanticipative decision policy under the current approximation of the feasible region. In that case, either $\hat{\mathcal{X}}_t = \emptyset$ or the approximation $\hat{\mathcal{X}}_{t-1}$ is updated through a feasibility cut. If $\hat{\mathcal{X}}_t = \emptyset$, then the problem must be infeasible. Otherwise, the introduced feasibility cut ensures that $\underline{\boldsymbol{x}}_{t-1}$ is no longer contained in $\hat{\mathcal{X}}_{t-1}$, and the forward pass is repeated for stage $t - 1$.

We now analyze the convergence of the revised RDDP scheme. The algorithm starts with the trivial outer approximations $\hat{\mathcal{X}}_t = \mathbb{R}^{n_t}$ of the stage-wise feasible regions $\mathcal{X}_t$. These outer approximations are refined in the forward passes in Step 3.

Lemma EC.1. *Each time the algorithm reaches Step 4, we have* $\overline{Q}_t^{F}(\underline{\boldsymbol{x}}_{t-1}) = 0$ *for all* $t = 2, \ldots, T$.

Proof. By construction of the forward pass, we have $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}) = 0$ for every $t = 2, \ldots, T$ *at some stage* in Step 3. We show that for every $t = 2, \ldots, T$, we also have $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}) = 0$ *at the end* of Step 3. Assume to the contrary that $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}) > 0$ for some $t = 2, \ldots, T$ at the end of Step 3. In that case, the outer approximation $\hat{\mathcal{X}}_t$ of the stage-t feasible region $\mathcal{X}_t$ must have changed after $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1})$ has been solved the last time. This is not possible, however, since $\hat{\mathcal{X}}_t$ is changed only when $\overline{Q}_{t+1}^{\mathrm{F}}(\underline{\boldsymbol{x}}_t) > 0$, in which case the problem $\overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1})$ is solved again. □

Lemma EC.1 ensures that the upper bound problems $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ in the forward and backward passes are only solved when their inner minimization problems are feasible for every realization of the uncertain parameters, and we can thus continue to solve these problems via mixed-integer linear programming or vertex enumeration, see Section 3.2. The lemma also ensures that strong duality holds between the lower bound problems $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \boldsymbol{\xi}_t)$ and their duals since the problems $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \boldsymbol{\xi}_t)$ are only solved when they are feasible.

We now show that the bounding property of $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ is preserved in our revised RDDP scheme.

PROPOSITION EC.1 **(Bounding Property)**. *Throughout the algorithm, the bounds $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ as well as the outer approximations $\hat{\mathcal{X}}_t$ of the stage-wise feasible regions $\mathcal{X}_t$ satisfy*

$$\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) + \delta_{\hat{\mathcal{X}}_{t-1}}(\boldsymbol{x}_{t-1}) \le \mathcal{Q}_t(\boldsymbol{x}_{t-1}) \le \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \qquad \forall \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}, \forall t = 2, \ldots, T,$$

that is, $\underline{\mathcal{Q}}_t + \delta_{\hat{\mathcal{X}}_{t-1}}$ and $\overline{\mathcal{Q}}_t$ bound $\mathcal{Q}_t$ from below and above, respectively.

PROOF. To prove that $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) + \delta_{\hat{\mathcal{X}}_{t-1}}(\boldsymbol{x}_{t-1}) \le \mathcal{Q}_t(\boldsymbol{x}_{t-1})$ for all $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$ and $t = 2, \ldots, T$, we show that *(i)* $\delta_{\hat{\mathcal{X}}_{t-1}}(\boldsymbol{x}_{t-1}) = +\infty$ only when $\mathcal{Q}_t(\boldsymbol{x}_{t-1}) = +\infty$ and *(ii)* $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \le \mathcal{Q}_t(\boldsymbol{x}_{t-1})$. To prove the first statement, we show that $\mathcal{X}_t \subseteq \hat{\mathcal{X}}_t$ for all $t = 1, \ldots, T$ by backward induction on the time stage t. Indeed, we have $\mathcal{X}_T = \hat{\mathcal{X}}_T$ by construction. To show that $\mathcal{X}_t \subseteq \hat{\mathcal{X}}_t$ implies $\mathcal{X}_{t-1} \subseteq \hat{\mathcal{X}}_{t-1}$, we show that $\boldsymbol{x}_{t-1} \in \mathcal{X}_{t-1}$ satisfies every feasibility cut $\{\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} : [\boldsymbol{h}_t(\boldsymbol{\xi}_t^{\mathrm{F}}) - \boldsymbol{T}_t(\boldsymbol{\xi}_t^{\mathrm{F}})\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\mu}_t - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \le 0\}$ that is added to $\hat{\mathcal{X}}_{t-1}$ in Step 3 in some iteration. Indeed, we have

$$\begin{aligned}
\mathcal{X}_{t-1} &= \left\{ \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} : \max_{\boldsymbol{\xi}_t \in \Xi_t} \min_{\boldsymbol{x}_t, \boldsymbol{v}_t} \left\{ \mathbf{e}^\top \boldsymbol{v}_t : \begin{bmatrix} \boldsymbol{T}_t(\boldsymbol{\xi}_t)\,\boldsymbol{x}_{t-1} + \boldsymbol{W}_t \boldsymbol{x}_t + \boldsymbol{v}_t \ge \boldsymbol{h}_t(\boldsymbol{\xi}_t) \\ \boldsymbol{x}_t \in \mathcal{X}_t, \;\; \boldsymbol{v}_t \in \mathbb{R}_+^{m_t} \end{bmatrix} \right\} \le 0 \right\} \\
&\subseteq \left\{ \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} : \underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1}; \boldsymbol{\xi}_t^{\mathrm{F}}) \le 0 \right\} \\
&\subseteq \left\{ \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} : \max_{\boldsymbol{\mu}_t \in [0,1]^{m_t}} \left\{ [\boldsymbol{h}_t(\boldsymbol{\xi}_t^{\mathrm{F}}) - \boldsymbol{T}_t(\boldsymbol{\xi}_t^{\mathrm{F}})\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\mu}_t - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \right\} \le 0 \right\} \\
&\subseteq \left\{ \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} : [\boldsymbol{h}_t(\boldsymbol{\xi}_t^{\mathrm{F}}) - \boldsymbol{T}_t(\boldsymbol{\xi}_t^{\mathrm{F}})\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\mu}_t - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \le 0 \right\}.
\end{aligned}$$

Here, the first identity follows from the definition of $\mathcal{X}_{t-1}$ and $\mathcal{X}_t$. The first inclusion holds since $\underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1}; \boldsymbol{\xi}_t^{\mathrm{F}})$ is equivalent to a variant of the max-min problem in the first row where we restrict Ξ_t to $\{\boldsymbol{\xi}_t^{\mathrm{F}}\}$ and replace $\mathcal{X}_t$ with $\hat{\mathcal{X}}_t$, which by the induction hypothesis contains $\mathcal{X}_t$. The second inclusion follows from weak linear programming duality, and the final inclusion holds since we restrict the feasible region $[0,1]^{m_t}$ of the maximization problem in the penultimate row to $\{\boldsymbol{\mu}_t\}$. Thus, $\boldsymbol{x}_{t-1} \in \mathcal{X}_{t-1}$ satisfies every feasibility cut introduced in Step 3, that is, $\mathcal{X}_{t-1} \subseteq \hat{\mathcal{X}}_{t-1}$.

The proof that $\underline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1}) \le \mathcal{Q}_t(\boldsymbol{x}_{t-1})$ and $\mathcal{Q}_t(\boldsymbol{x}_{t-1}) \le \overline{\mathcal{Q}}_t(\boldsymbol{x}_{t-1})$ directly follows from the proof of Proposition 2, which remains valid for our revised RDDP scheme due to Lemma EC.1. □

In analogy to Section 3.3, Proposition EC.1 implies that the stage-1 problem is a relaxation of problem (1'). Thus, infeasibility of the stage-1 problem implies infeasibility of problem (1'), and any solution $\underline{\boldsymbol{x}}_1$ satisfying $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \overline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1)$ must also satisfy $\underline{\mathcal{Q}}_2(\underline{\boldsymbol{x}}_1) = \mathcal{Q}_2(\underline{\boldsymbol{x}}_1)$ and therefore be optimal in problem (1'). We have thus arrived at the following result.

COROLLARY EC.1 **(Correctness)**. *If the RDDP scheme terminates, then it either returns an optimal solution to problem* (1) *or it correctly identifies infeasibility of the problem.*

To show that our revised RDDP scheme converges in finite time, we again assume that the conditions (C1) and (C2) from the main paper hold, and we make two additional assumptions.

(C3) The solution method for the stage-wise feasibility problems $\overline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1})$ returns an optimal extreme point of the uncertainty set Ξ_t.

(C4) The solution method for the stage-wise feasibility problems $\underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$ returns an optimal basic feasible solution of the linear program corresponding to the epigraph reformulation of $\underline{Q}_t^{\mathrm{F}}(\boldsymbol{x}_{t-1};\boldsymbol{\xi}_t)$.

The assumptions closely resemble the conditions (C1) and (C2) from the main paper.

We first show that there are finitely many different outer approximations $\hat{\mathcal{X}}_t$ of each feasible region $\mathcal{X}_t$ that our revised RDDP scheme can generate during its execution.

LEMMA EC.2 **(Outer Approximations)**. *For a fixed instance of problem* (1), *there are finitely many different outer approximations* $\hat{\mathcal{X}}_t$, $t=1,\ldots,T$, *that the algorithm can generate.*

PROOF. As in the proof of Lemma 1, we employ a backward induction on the time stage t. For $t=T$, the dual to the stage-T feasibility problem $\underline{Q}_T^{\mathrm{F}}(\underline{\boldsymbol{x}}_{T-1};\boldsymbol{\xi}_T^{\mathrm{F}})$ is

$$\begin{aligned} &\text{maximize} && [\boldsymbol{h}_T(\boldsymbol{\xi}_T^{\mathrm{F}}) - \boldsymbol{T}_T(\boldsymbol{\xi}_T^{\mathrm{F}})\,\underline{\boldsymbol{x}}_{T-1}]^\top \boldsymbol{\mu}_T - \sigma_{\hat{\mathcal{X}}_T}(\boldsymbol{W}_T^\top \boldsymbol{\mu}_T) \\ &\text{subject to} && \boldsymbol{\mu}_T \in [0,1]^{m_T}. \end{aligned}$$

Since $\hat{\mathcal{X}}_T = \mathcal{X}_T = \mathbb{R}^{n_T}$ throughout the algorithm, we have $\sigma_{\hat{\mathcal{X}}_T}(\boldsymbol{W}_T^\top \boldsymbol{\mu}_T) = 0$ if $\boldsymbol{W}_T^\top \boldsymbol{\mu}_T = \mathbf{0}$ and $\sigma_{\hat{\mathcal{X}}_T}(\boldsymbol{W}_T^\top \boldsymbol{\mu}_T) = +\infty$ otherwise. Thus, the dual problem can be rewritten as

$$\begin{aligned} &\text{maximize} && [\boldsymbol{h}_T(\boldsymbol{\xi}_T^{\mathrm{F}}) - \boldsymbol{T}_T(\boldsymbol{\xi}_T^{\mathrm{F}})\,\underline{\boldsymbol{x}}_{T-1}]^\top \boldsymbol{\mu}_T \\ &\text{subject to} && \boldsymbol{W}_T^\top \boldsymbol{\mu}_T = \mathbf{0} \\ &&& \boldsymbol{\mu}_T \in [0,1]^{m_T}. \end{aligned}$$

This is a linear program with finitely many basic feasible solutions, none of which depends on the candidate solution $\underline{\boldsymbol{x}}_{T-1}$ or the parameter realization $\boldsymbol{\xi}_T^{\mathrm{F}}$. Since the assumption (C4) ensures that the feasibility cut determined in Step 3 of the algorithm corresponds to a basic feasible solution of this dual problem, we thus conclude that for every $\boldsymbol{\xi}_T^{\mathrm{F}}$ there are finitely many different feasibility cuts that can be added to $\hat{\mathcal{X}}_{T-1}$ throughout the execution of the algorithm. The statement then follows for stage T since there are finitely many extreme points $\boldsymbol{\xi}_T^{\mathrm{F}} \in \Xi_T$ that can emerge as optimal solutions to the stage-T feasibility problem $\overline{Q}_T^{\mathrm{F}}(\underline{\boldsymbol{x}}_{T-1})$ according to the condition (C3).

For $t < T$, the dual to the stage-t feasibility problem $\underline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}; \boldsymbol{\xi}_t^{\mathrm{F}})$ can be written as

$$\begin{aligned} &\text{maximize} && [\boldsymbol{h}_t(\boldsymbol{\xi}_t^{\mathrm{F}}) - \boldsymbol{T}_t(\boldsymbol{\xi}_t^{\mathrm{F}})\,\underline{\boldsymbol{x}}_{t-1}]^\top \boldsymbol{\mu}_t - \varphi \\ &\text{subject to} && \varphi \geq \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \\ &&& \boldsymbol{\mu}_t \in [0,1]^{m_t}, \end{aligned} \tag{EC.3}$$

where $\sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t)$ is a support function over the outer approximation $\hat{\mathcal{X}}_t$ of the stage-t feasible set $\mathcal{X}_t$. The set $\hat{\mathcal{X}}_t$ is polyhedral by construction, and the induction hypothesis implies that the set is described by finitely many halfspaces and that it takes on finitely many different shapes throughout the algorithm. Hence, the support function $\sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t)$ is a piecewise affine convex function with finitely many different pieces that takes on finitely many different shapes throughout the algorithm. Problem (EC.3) can then be cast as a linear program with finitely many basic feasible solutions, none of which depends on $\underline{\boldsymbol{x}}_{t-1}$ or $\boldsymbol{\xi}_t^{\mathrm{F}}$. A similar argument as in the previous paragraph then shows that there are finitely many different feasibility cuts that can be added to the outer approximation $\hat{\mathcal{X}}_{t-1}$ throughout the algorithm. □

We are now in the position to prove the finite convergence of our revised RDDP scheme.

Theorem EC.1 **(Finite Termination)**. *The revised RDDP scheme terminates in finite time.*

Proof. We show that the forward pass in each iteration completes in finite time. The statement of the theorem then follows from the fact that the proofs of Lemmas 1 and 2 as well as Theorem 2 remain valid for the revised RDDP scheme due to Lemma EC.1.

Assume that the forward pass of an iteration would cycle indefinitely. In that case, there would be at least one stage $(t-1) \in \{1, \ldots, T-1\}$ such that the outer approximation $\hat{\mathcal{X}}_{t-1}$ of the stage-$(t-1)$ feasible region $\mathcal{X}_{t-1}$ is updated infinitely many times in Step 3. Fix any such update

$$\hat{\mathcal{X}}_{t-1} \leftarrow \hat{\mathcal{X}}_{t-1} \cap \left\{ \boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}} \,:\, [\boldsymbol{h}_t(\boldsymbol{\xi}_t^{\mathrm{F}}) - \boldsymbol{T}_t(\boldsymbol{\xi}_t^{\mathrm{F}})\,\boldsymbol{x}_{t-1}]^\top \boldsymbol{\mu}_t - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \leq 0 \right\},$$

where $\boldsymbol{\mu}_t$ is the shadow price of an optimal solution to the problem $\underline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1};\boldsymbol{\xi}_t^{\mathrm{F}})$. Note that $\underline{\boldsymbol{x}}_{t-1} \in \hat{\mathcal{X}}_{t-1}$ before the update since $\underline{\boldsymbol{x}}_{t-1}$ minimizes $\underline{Q}_{t-1}(\underline{\boldsymbol{x}}_{t-2};\overline{\boldsymbol{\xi}}_{t-1}^{\mathrm{fw}})$. By construction, we have that

$$[\boldsymbol{h}_t(\boldsymbol{\xi}_t^{\mathrm{F}}) - \boldsymbol{T}_t(\boldsymbol{\xi}_t^{\mathrm{F}})\,\underline{\boldsymbol{x}}_{t-1}]^\top \boldsymbol{\mu}_t - \sigma_{\hat{\mathcal{X}}_t}(\boldsymbol{W}_t^\top \boldsymbol{\mu}_t) \;=\; \underline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1};\boldsymbol{\xi}_t^{\mathrm{F}}) \;=\; \overline{Q}_t^{\mathrm{F}}(\underline{\boldsymbol{x}}_{t-1}) \;>\; 0,$$

that is, the updated set $\hat{\mathcal{X}}_{t-1}$ no longer contains $\underline{\boldsymbol{x}}_{t-1}$. We thus conclude that each update would have to result in a different outer approximation $\hat{\mathcal{X}}_{t-1}$, which contradicts Lemma EC.2. □

EC.3. Uncertain Objective Functions and Recourse Matrices

This section relaxes the assumption (A4) from the main paper, which stipulates that the objective coefficients and the recourse matrices of the multi-stage robust optimization problem (1) are deterministic. We also allow for generic (possibly non-convex) compact stage-wise uncertainty sets Ξ_t as long as the rectangularity condition $\Xi = \times_{t=1}^{T} \Xi_t$ remains satisfied. To simplify the proofs, we continue to assume that problem (1) satisfies the other two conditions (A1) and (A2).

For ease of exposition, we consider the following, slightly modified variant of problem (1):

$$\begin{array}{lll} \text{minimize} & \displaystyle\max_{\boldsymbol{\xi} \in \Xi} \sum_{t=1}^{T} \boldsymbol{q}_t^\top \boldsymbol{x}_t(\boldsymbol{\xi}^t) & \\ \text{subject to} & \boldsymbol{f}_1(\boldsymbol{x}_1) \leq \mathbf{0} & \forall \boldsymbol{\xi} \in \Xi \\ & \boldsymbol{f}_t(\boldsymbol{x}_{t-1}(\boldsymbol{\xi}^{t-1}), \boldsymbol{\xi}_t, \boldsymbol{x}_t(\boldsymbol{\xi}^t)) \leq \mathbf{0} & \forall \boldsymbol{\xi} \in \Xi, \, \forall t = 2, \ldots, T \\ & \boldsymbol{x}_t(\boldsymbol{\xi}^t) \in \mathbb{R}^{n_t}, \, \boldsymbol{\xi} \in \Xi \text{ and } t = 1, \ldots, T, & \end{array} \tag{EC.4}$$

where $\boldsymbol{f}_1 : \mathbb{R}^{n_1} \mapsto \mathbb{R}^{m_1}$ and $\boldsymbol{f}_t : \mathbb{R}^{n_{t-1}} \times \mathbb{R}^{k_t} \times \mathbb{R}^{n_t} \mapsto \mathbb{R}^{m_t}$, $t = 2, \ldots, T$, are generic functions that describe the state transitions between two successive time stages.

We change the assumptions (A3) and (A4) from the main paper as follows:

(A3') Uncertainty Set. The uncertainty set Ξ is compact and stage-wise rectangular, that is, $\Xi = \times_{t=1}^{T} \Xi_t$ for compact (but possibly non-convex) sets $\Xi_t \subseteq \mathbb{R}^{k_t}$.

(A4') Uncertain Parameters. The transition functions $\boldsymbol{f}_t(\cdot, \boldsymbol{\xi}_t, \cdot)$, $t = 2, \ldots, T$, are jointly quasi-convex in their first and last arguments for every $t = 2, \ldots, T$ and $\boldsymbol{\xi}_t \in \Xi_t$. Moreover, for all $t = 2, \ldots, T$, $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$ and $\boldsymbol{\xi}_t \in \Xi_t$ there exists $\boldsymbol{x}_t \in \mathbb{R}^{n_t}$ such that $\boldsymbol{f}_t(\boldsymbol{x}_{t-1}, \boldsymbol{\xi}_t, \boldsymbol{x}_t) < \mathbf{0}$.

The assumption (A3') admits, among others, ellipsoidal and semidefinite uncertainty sets, as well as uncertainty sets involving discrete parameters. Likewise, the assumption (A4') allows for a broad range of linear and nonlinear state transitions. In particular, the choice $\boldsymbol{f}_t(\boldsymbol{x}_{t-1}, \boldsymbol{\xi}_t, \boldsymbol{x}_t) = \boldsymbol{h}_t(\boldsymbol{\xi}_t) - \boldsymbol{T}_t(\boldsymbol{\xi}_t)\boldsymbol{x}_{t-1} - \boldsymbol{W}_t(\boldsymbol{\xi}_t)\boldsymbol{x}_t$ recovers instances of problem (1) with uncertain recourse matrices. Using an epigraph reformulation, problem (EC.4) can also accommodate uncertain objective coefficients. The Slater conditions ensure that strong duality holds for the lower bound problems $\underline{Q}_t(\boldsymbol{x}_{t-1}; \boldsymbol{\xi}_t)$, whose associated dual problems provide the cuts for the lower bounds $\underline{\mathcal{Q}}_t$ in the RDDP scheme.

We apply the following two modifications to our RDDP scheme from the main paper:

(i) We replace $\pm\infty$ in the initial bounds $\underline{\mathcal{Q}}_t$, $\overline{\mathcal{Q}}_t$ with a sufficiently large number $\pm M$.

(ii) We update the lower bounds $\underline{\mathcal{Q}}_t$ in both the forward and the backward passes.

The first modification is used as a technical device to ensure uniform convergence of the cost to-go approximations $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$. Note that a finite bound M is guaranteed to exist since the feasible region and the stage-wise uncertainty sets Ξ_t are bounded. The second modification simplifies our convergence argument, but it can also be motivated by the following remark.

Remark EC.1 (Lower Bound Updates in Forward Pass). Our RDDP scheme updates the lower and upper worst-case cost to-go approximations $\underline{\mathcal{Q}}_t$ and $\overline{\mathcal{Q}}_t$ in the backward passes only. This is justified for the upper bounds $\overline{\mathcal{Q}}_t$ since the optimal values of the upper bound problems $\overline{Q}_t(\underline{\boldsymbol{x}}_{t-1})$ in the backward passes are guaranteed to be at most as large as those in the preceding forward passes. However, since $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw}} \neq \overline{\boldsymbol{\xi}}_t^{\mathrm{bw}}$ in general, the supporting hyperplanes corresponding to the optimal solutions of $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{fw}})$ in the forward pass may still provide useful information after the supporting hyperplanes of the backward pass have been introduced. It may thus be beneficial to refine the lower cost to-go approximations $\underline{\mathcal{Q}}_t$ in both the forward and the backward passes.

Standard results from convex analysis, such as Propositions 2.9 and 2.22 from Rockafellar and Wets (2009), show that under the assumptions (A3') and (A4'), the stage-wise worst-case cost to-go functions $\mathcal{Q}_t$, $t = 2, \ldots, T$, remain convex. However, the revised assumption (A4') implies that the functions $\mathcal{Q}_t$ may no longer be piecewise affine, even if the stage-wise uncertainty sets Ξ_t are polyhedral and all transition functions $\boldsymbol{f}_t$ are linear in $\boldsymbol{\xi}_t$ and jointly linear in $\boldsymbol{x}_{t-1}$ and $\boldsymbol{x}_t$. To illustrate this, consider the following instance of problem (EC.4):

$$\begin{array}{lllll}
\text{minimize} & \displaystyle\max_{\boldsymbol{\xi}\in\Xi} \; -2x_1(\xi^1) + x_{21}(\boldsymbol{\xi}^2) + x_{22}(\boldsymbol{\xi}^2) & & & \\
\text{subject to} & x_{21}(\boldsymbol{\xi}^2) = -\dfrac{2}{3}\xi_2 x'_{21}(\boldsymbol{\xi}^2), & x'_{21}(\boldsymbol{\xi}^2) = \xi_2 x''_{21}(\boldsymbol{\xi}^2), & x''_{21}(\boldsymbol{\xi}^2) = \xi_2 & \forall \boldsymbol{\xi} \in \Xi \\
 & x_{22}(\boldsymbol{\xi}^2) = \xi_2 x'_{22}(\boldsymbol{\xi}^2), & x'_{22}(\boldsymbol{\xi}^2) = \xi_2 x_1(\xi^1) & & \forall \boldsymbol{\xi} \in \Xi \\
 & \multicolumn{4}{l}{x_1(\xi^1) \in [0,3], \;\; x_{21}(\boldsymbol{\xi}^2), x'_{21}(\boldsymbol{\xi}^2), x''_{21}(\boldsymbol{\xi}^2), x_{22}(\boldsymbol{\xi}^2), x'_{22}(\boldsymbol{\xi}^2) \in \mathbb{R}, \; \boldsymbol{\xi} \in \Xi\,,}
\end{array} \tag{EC.5}$$

where the uncertainty set is $\Xi = \Xi_1 \times \Xi_2 = \{1\} \times [0,3]$. Problem (EC.5) is an instance of a linear multi-stage robust optimization problem with uncertain recourse matrices. In fact, if the recourse matrices in problem (EC.5) were deterministic, then the functions $\mathcal{Q}_t$ would be piecewise affine, and Theorems 2 and EC.1 would imply that the problem could be solved in finite time by our RDDP scheme. Straightforward variable substitutions reveal that the problem is equivalent to

$$\min_{x_1 \in [0,3]} \max_{\xi \in [0,3]} -\frac{2}{3}\xi^3 + \xi^2 x_1 - 2x_1,$$

and $-\frac{2}{3}\xi^3 + \xi^2 x_1 - 2x_1$ describes the tangent of the nonlinear function $\frac{1}{3}x_1^3 - 2x_1$ at $\xi \in [0,3]$. The cost to-go function of problem (EC.5) thus satisfies $\mathcal{Q}_2(x_1) = \frac{1}{3}x_1^3 - 2x_1$, and it is minimized at $x_1^\star$ satisfying the first-order condition $(x_1^\star)^2 - 2 = 0$, that is, at $x_1^\star = \sqrt{2}$.

Consider now a solution approach for the generic multi-stage robust optimization problem (EC.4) that operates on piecewise affine approximations of the cost to-go function $\mathcal{Q}_2$ in problem (EC.5). Assume that the slopes or the breakpoints of these approximations are derived from primal or dual linear programming formulations of the second-stage problem in (EC.5). These linear programs can be solved exactly both by the simplex algorithm and by interior point methods (by rounding to the nearest extreme point, which can be done in polynomial time for linear programs). All basic feasible solutions of these linear programs have a bit-length that is bounded by a polynomial of the size of the input data (Limongelli and Pirastu 1994). Consequently, all breakpoints of the cost to-go approximations have a polynomial bit-length as well. However, since the solution to problem (EC.5) is an irrational number, we cannot expect it to emerge as a breakpoint of a piecewise affine cost to-go approximation after finitely many iterations. Assuming that the solution approach selects breakpoints of the cost to-go approximations as candidate solutions, the solution approach thus cannot converge to the optimal solution of the problem (EC.5) in finitely many iterations.

We now show that our RDDP scheme asymptotically converges to an optimal solution of the generic multi-stage robust optimization problem (EC.4). Our convergence proof does not require the assumptions (C1) and (C2) from the main paper. For ease of exposition, however, we assume that all feasible stage-t solutions $\boldsymbol{x}_t$ are contained in a bounded set $\mathcal{X}_t$, and that there is a compact subset $\mathcal{X}_t^\infty \subseteq \operatorname{rel\,int} \mathcal{X}_t$ such that every accumulation point $\underline{\boldsymbol{x}}_t^\infty$ of our RDDP scheme satisfies $\underline{\boldsymbol{x}}_t^\infty \in \mathcal{X}_t^\infty$. The existence of the sets $\mathcal{X}_t$ is guaranteed by the assumption (A1). The stipulated existence of the sets $\mathcal{X}_t^\infty$ can be relaxed at the expense of further case distinctions in the proofs below.

In the following, we denote by $\underline{\mathcal{Q}}_t^{\mathrm{fw},\ell}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell})$ the cost to-go approximations after the forward and the backward pass in iteration ℓ of the RDDP scheme, respectively. An inspection of the proof of Proposition 2 reveals that the bounding property of $\underline{\mathcal{Q}}_t^{\mathrm{fw},\ell}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell})$ still holds for all $\boldsymbol{x}_{t-1} \in \mathbb{R}^{n_{t-1}}$. We now show that these approximations, as well as the cost to-go functions $\mathcal{Q}_t$, also satisfy certain regularity conditions.

Lemma EC.3. *The cost to-go approximations $\underline{\mathcal{Q}}_t^{fw,\ell}$ and $(\overline{\mathcal{Q}}_t^{bw,\ell}, \underline{\mathcal{Q}}_t^{bw,\ell})$, their limits $\underline{\mathcal{Q}}_t^{fw,\infty}$ and $(\overline{\mathcal{Q}}_t^{bw,\infty}, \underline{\mathcal{Q}}_t^{bw,\infty})$ as well as the true cost to-go functions $\mathcal{Q}_t$ are uniformly continuous over $\mathcal{X}_t^\infty$. Moreover, the functions $\underline{\mathcal{Q}}_t^{fw,\ell}$ and $(\overline{\mathcal{Q}}_t^{bw,\ell}, \underline{\mathcal{Q}}_t^{bw,\ell})$ converge uniformly over $\mathcal{X}_t^\infty$.*

Proof. The pointwise limits $\underline{\mathcal{Q}}_t^{\mathrm{fw},\infty}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\infty})$ of the cost to-go approximations $\underline{\mathcal{Q}}_t^{\mathrm{fw},\ell}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell})$ over $\mathcal{X}_t$ exist due to the monotone convergence theorem and the fact that the images of $\underline{\mathcal{Q}}_t^{\mathrm{fw},\ell}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell})$ are restricted to the interval $[-M, +M]$.

The cost to-go approximations $\underline{\mathcal{Q}}_t^{\mathrm{fw},\ell}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell})$, their limits $\underline{\mathcal{Q}}_t^{\mathrm{fw},\infty}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\infty})$ and the true cost to-go functions $\mathcal{Q}_t$ are convex over $\mathcal{X}_t$ and hence continuous over $\mathcal{X}_t^\infty$. The Heine-Cantor theorem then implies that these functions are in fact uniformly continuous over $\mathcal{X}_t^\infty$.

The approximations $\underline{\mathcal{Q}}_t^{\mathrm{fw},\ell}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell})$ constitute monotone sequences of functions. Since the functions in these sequences and their limits are continuous over $\mathcal{X}_t^\infty$, we conclude from Dini's theorem that $\underline{\mathcal{Q}}_t^{\mathrm{fw},\ell}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell})$ converge to $\underline{\mathcal{Q}}_t^{\mathrm{fw},\infty}$ and $(\overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}, \underline{\mathcal{Q}}_t^{\mathrm{bw},\infty})$ uniformly. □

Lemma EC.3 allows us to study the behavior of $\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}(\underline{\boldsymbol{x}}_{t-1}^\ell)$ and $\underline{\mathcal{Q}}_t^{\mathrm{bw},\ell}(\underline{\boldsymbol{x}}_{t-1}^\ell)$ as $\ell \to \infty$. It is instrumental in the following two results, which show that the upper and lower bounds $\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}(\underline{\boldsymbol{x}}_{t-1}^\ell)$ and $\underline{\mathcal{Q}}_t^{\mathrm{bw},\ell}(\underline{\boldsymbol{x}}_{t-1}^\ell)$ converge to the true costs in the vicinity of every accumulation point $\underline{\boldsymbol{x}}_t^\infty$.

Lemma EC.4. *Let $\underline{\boldsymbol{x}}_t^\infty$ be any accumulation point of the sequence $\underline{\boldsymbol{x}}_t^\ell$, and let $\overline{\mathcal{Q}}_t^{bw,\infty}$ be the limit functions of the sequences $\overline{\mathcal{Q}}_t^{bw,\ell}$, $t = 1, \ldots, T$ and $\ell = 1, 2, \ldots$ Then $\overline{\mathcal{Q}}_t^{bw,\infty}(\underline{\boldsymbol{x}}_{t-1}^\infty) \leq \boldsymbol{q}_t^\top \underline{\boldsymbol{x}}_t^\infty + \overline{\mathcal{Q}}_{t+1}^{bw,\infty}(\underline{\boldsymbol{x}}_t^\infty)$ for all $t = 2, \ldots, T$.*

Proof. By possibly going over to subsequences, we can assume that $\underline{\boldsymbol{x}}_t^\ell$ itself converges to $\underline{\boldsymbol{x}}_t^\infty$, $\ell = 1, 2, \ldots$ The upper bound then satisfies

$$\begin{aligned}
\overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_{t-1}^\infty) &\overset{\text{(a)}}{=} \lim_{l \to \infty} \overline{\mathcal{Q}}_t^{\mathrm{bw},\ell+1}(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}) \\
&\overset{\text{(b)}}{=} \lim_{l \to \infty} \max_{\boldsymbol{\xi}_t \in \Xi_t} \min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell+1}(\boldsymbol{x}_t) \, : \, \boldsymbol{f}_t(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}, \boldsymbol{\xi}_t, \boldsymbol{x}_t) \leq \boldsymbol{0} \right\} \\
&\overset{\text{(c)}}{\leq} \lim_{l \to \infty} \max_{\boldsymbol{\xi}_t \in \Xi_t} \min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell}(\boldsymbol{x}_t) \, : \, \boldsymbol{f}_t(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}, \boldsymbol{\xi}_t, \boldsymbol{x}_t) \leq \boldsymbol{0} \right\}
\end{aligned}$$

$$
\begin{aligned}
&\overset{\text{(d)}}{=} \lim_{l\to\infty} \min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \overline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell}(\boldsymbol{x}_t) \,:\, \boldsymbol{f}_t(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}, \overline{\boldsymbol{\xi}}_t^{\mathrm{fw},\ell+1}, \boldsymbol{x}_t) \leq \boldsymbol{0} \right\} \\
&\overset{\text{(e)}}{\leq} \lim_{l\to\infty} \boldsymbol{q}_t^\top \underline{\boldsymbol{x}}_t^{\ell+1} + \overline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell}(\underline{\boldsymbol{x}}_t^{\ell+1}) \\
&\overset{\text{(f)}}{=} \boldsymbol{q}_t^\top \underline{\boldsymbol{x}}_t^{\infty} + \overline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_t^{\infty}).
\end{aligned}
$$

By Lemma EC.3, the identity (a) holds since

$$
\begin{aligned}
\lim_{l\to\infty} \left\| \overline{\mathcal{Q}}_t^{\mathrm{bw},\ell+1}(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}) - \overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_{t-1}^{\infty}) \right\| \leq \lim_{l\to\infty} \Bigg[& \left\| \overline{\mathcal{Q}}_t^{\mathrm{bw},\ell+1}(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}) - \overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}) \right\| + \\
& \left\| \overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}) - \overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_{t-1}^{\infty}) \right\| \Bigg],
\end{aligned}
$$

where the first expression on the right-hand side converges to zero since $\overline{\mathcal{Q}}_t^{\mathrm{bw},\ell}$ converges uniformly to $\overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}$, while the second expression converges to zero since $\underline{\boldsymbol{x}}_{t-1}^{\ell}$ converges to $\underline{\boldsymbol{x}}_{t-1}^{\infty}$ and $\overline{\mathcal{Q}}_t^{\mathrm{bw},\infty}$ is continuous. The identity (b) follows from the update of the upper bound in the backward pass of iteration $\ell+1$. The inequality (c) holds since $\overline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell+1} \leq \overline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell}$. The identity (d) follows directly from the definition of $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw},\ell+1}$. The inequality (e) holds since $\underline{\boldsymbol{x}}_t^{\ell+1}$ minimizes the problem

$$
\min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \underline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell}(\boldsymbol{x}_t) \,:\, \boldsymbol{f}_t(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}, \overline{\boldsymbol{\xi}}^{\mathrm{fw}_t,\ell+1}, \boldsymbol{x}_t) \leq \boldsymbol{0} \right\}
$$

in the forward pass of iteration $\ell+1$, and it is therefore feasible in problem (d). The identity (f), finally, holds for the same reasons as the identity (a). □

Lemma EC.5. *Let $\underline{\boldsymbol{x}}_t^{\infty}$ be any accumulation point of the sequence $\underline{\boldsymbol{x}}_t^{\ell}$, and let $\underline{\mathcal{Q}}_t^{bw,\infty}$ be the limit functions of the sequences $\underline{\mathcal{Q}}_t^{bw,\ell}$, $t=1,\ldots,T$ and $\ell=1,2,\ldots$ Then $\underline{\mathcal{Q}}_t^{bw,\infty}(\underline{\boldsymbol{x}}_{t-1}^{\infty}) \geq \boldsymbol{q}_t^\top \underline{\boldsymbol{x}}_t^{\infty} + \underline{\mathcal{Q}}_{t+1}^{bw,\infty}(\underline{\boldsymbol{x}}_t^{\infty})$ for all $t=2,\ldots,T$.*

Proof. By possibly going over to subsequences, we can assume that $\underline{\boldsymbol{x}}_t^{\ell}$ itself converges to $\underline{\boldsymbol{x}}_t^{\infty}$, $\ell=1,2,\ldots$ The lower bound then satisfies

$$
\begin{aligned}
\underline{\mathcal{Q}}_t^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_{t-1}^{\infty}) &\overset{\text{(a)}}{=} \lim_{l\to\infty} \underline{\mathcal{Q}}_t^{\mathrm{bw},\ell+1}(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}) \\
&\overset{\text{(b)}}{\geq} \lim_{l\to\infty} \underline{\mathcal{Q}}_t^{\mathrm{fw},\ell+1}(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}) \\
&\overset{\text{(c)}}{=} \lim_{l\to\infty} \min_{\boldsymbol{x}_t \in \mathbb{R}^{n_t}} \left\{ \boldsymbol{q}_t^\top \boldsymbol{x}_t + \underline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell}(\boldsymbol{x}_t) \,:\, \boldsymbol{f}_t(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}, \overline{\boldsymbol{\xi}}_t^{\mathrm{fw},\ell+1}, \boldsymbol{x}_t) \leq \boldsymbol{0} \right\} \\
&\overset{\text{(d)}}{=} \lim_{l\to\infty} \boldsymbol{q}_t^\top \underline{\boldsymbol{x}}_t^{\ell+1} + \underline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\ell}(\underline{\boldsymbol{x}}_t^{\ell+1}) \\
&\overset{\text{(e)}}{=} \boldsymbol{q}_t^\top \underline{\boldsymbol{x}}_t^{\infty} + \underline{\mathcal{Q}}_{t+1}^{\mathrm{bw},\infty}(\underline{\boldsymbol{x}}_t^{\infty}).
\end{aligned}
$$

Here, the identity (a) is explained by the same reasons as the identity (a) in the proof of Lemma EC.4. The inequality (b) holds since $\underline{\mathcal{Q}}_t^{\mathrm{bw},\ell+1} \geq \underline{\mathcal{Q}}_t^{\mathrm{fw},\ell+1}$. The identity (c) follows from the update in the forward pass of iteration $\ell+1$, the definition of $\overline{\boldsymbol{\xi}}_t^{\mathrm{fw},\ell+1}$, as well as the fact that strong duality holds for the lower bound problems $\underline{Q}_t(\underline{\boldsymbol{x}}_{t-1}^{\ell+1}; \overline{\boldsymbol{\xi}}_t^{\mathrm{fw},\ell+1})$. The identity (d) holds by definition of $\underline{\boldsymbol{x}}_t^{\ell+1}$. The identity (e), finally, holds for the same reasons as (a). □

Equipped with the Lemmas EC.4 and EC.5, we can now prove the main result of this section.

THEOREM EC.2. *Let $\underline{\boldsymbol{x}}_t^{\infty}$ be any accumulation point of the sequence $\underline{\boldsymbol{x}}_t^{\ell}$, and let $\overline{\mathcal{Q}}_t^{bw,\infty}$ and $\underline{\mathcal{Q}}_t^{bw,\infty}$ be the limit functions of the sequences $\overline{\mathcal{Q}}_t^{bw,\ell}$ and $\underline{\mathcal{Q}}_t^{bw,\ell}$, respectively, $t = 1,\ldots,T$ and $\ell = 1,2,\ldots$ Then $\overline{\mathcal{Q}}_t^{bw,\infty}(\underline{\boldsymbol{x}}_{t-1}^{\infty}) = \underline{\mathcal{Q}}_t^{bw,\infty}(\underline{\boldsymbol{x}}_{t-1}^{\infty})$ for all $t = 2,\ldots,T$.*

PROOF. Employing a backward induction on the time stage t, the statement follows from Proposition 2, Lemmas EC.4 and EC.5 as well as the fact that the terminal cost to-go bounds satisfy $\overline{\mathcal{Q}}_{T+1}^{\mathrm{bw},\infty}(\boldsymbol{x}_T) = 0 = \underline{\mathcal{Q}}_{T+1}^{\mathrm{bw},\infty}(\boldsymbol{x}_T)$ for all $\boldsymbol{x}_T \in \mathbb{R}^{n_T}$. □

From Proposition 2 we can thus conclude that any accumulation point of our RDDP scheme is an optimal solution to the generic multi-stage robust optimization problem (EC.4).

EC.4. Detailed Numerical Results

Table EC.1 provides further details of the numerical results from Section 5.

	RDDP				**ADR**
Instance	20%	10%	5%	1%	(%, time)
5-4-25	7.5s	8.7s	9.7s	12.3s	(36.4%, 23.0s)
5-4-50	26s	29s	33s	45s	(69%, 117s)
5-4-75	56s	63s	68s	80s	—
5-4-100	102s	113s	121s	133s	—
10-4-25	48s	57s	65s	81s	(68.3%, 83s)
12-4-25	105s	119s	135s	184s	(70.3%, 162s)
14-4-25	164s	197s	238s	316s	(79.4%, 177s)
16-4-25	228s	307s	381s	523s	(91.6%, 252s)
18-4-25	591s	658s	732s	896s	(66.0%, 434s)
20-4-25	978s	1,174s	1,422s	1,999s	(74.5%, 691s)
6-5-25	20s	22s	26s	33s	(41.6%, 38s)
7-6-25	52s	59s	65s	77s	(95.9%, 96s)
8-7-25	202s	213s	229s	276s	(103%, 106s)
9-8-25	568s	587s	627s	702s	(105%, 179s)
10-9-25	2,201s	2,324s	2,438s	2,714s	(99.8%, 2,162s)

Table EC.1 Optimization times required by the RDDP scheme to reduce the optimality gaps to 20%, 10%, 5% and 1% for different instances classes. The column 'ADR' reports the suboptimalilty of the affine decision rules and their runtimes. All numbers correspond to the median values over the randomly generated instances.