CIRRELT

Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation

On the Scenario-Tree Optimal-Value Error for Stochastic Programming Problems

Julien Keutchayan
David Munger
Michel Gendreau

January 2017

CIRRELT-2017-05

Bureaux de Montréal :
Université de Montréal
Pavillon André-Aisenstadt
C.P. 6128, succursale Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
Pavillon Palasis-Prince
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

On the Scenario-Tree Optimal-Value Error for Stochastic Programming Problems

Julien Keutchayan[1,*], David Munger[1,2,†], Michel Gendreau[1]

[1] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) and Department of Mathematics and Industrial Engineering, Polytechnique Montréal, P.O. Box 6079, Station Centre-Ville, Montréal, Canada H3C 3A7

[2] Kronos, AD OPT Division, 3535 chemin Queen-Mary, Montréal, Canada, H3V 1H8

Abstract. Stochastic programming problems (SPPs) generally lead to large-scale intractable programs if the number of possible outcomes for the random parameters is large and/or if the problem has many stages. A way to address those problems at lower computational cost is provided by the scenario-tree generation methods, which construct deterministic approximate problems from a selected finite subset of outcomes (called scenarios). When considering a general SPP, the number of scenarios required to keep the optimal-value error within a given range generally grows exponentially with the number of random parameters and stages, which may lead to approximate problems that are themselves intractable. To overcome this fast growth of complexity, there is a need to look for scenario-tree generation methods that are tailored to specific classes of SPPs. In this paper, we provide a theoretical basis to develop such methods by studying the optimal-value error in the context of a general SPP. Specifically, we derive two main results: an error decomposition and an error upper bound. The error decomposition shows the importance of taking into account the recourse functions when designing scenario trees. The error upper bound, which can be written as a sum of worst-case integration errors in some function sets, provides the cornerstone to a new approach that will consist in identifying classes of SPPs based on properties of their recourse functions and in designing scenario trees suitable for them.

Keywords. Multistage stochastic programming, scenario-tree generation, error analysis.

Acknowledgements. The authors would like to thank for their support Hydro-Québec and the Natural Sciences and Engineering Research Council of Canada (NSERC) through the NSERC/Hydro-Québec Industrial Research Chair on the Stochastic Optimization of Electricity Generation, Polytechnique Montréal, and the NSERC Discovery Grants Program.

† This research was carried out while David Munger was research professional at CIRRELT, Université de Montréal.

---

* Corresponding author: Julien.Keutchayan@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2017

1 Introduction

In many stochastic programming problems (SPPs), the uncertain parameters are modeled by random variables that take a large number of values, possibly infinitely many if their probability distributions are continuous. SPPs may also have a large number of stages, as in the long-term horizon planning; see Hobbs [13] for a discussion on long-term planning in energy. Such problems are found in various fields of application, including logistic, finance and energy; we refer to the books of Wallace and Ziemba [37], Bertocchi et al. [3] and Kovacevic et al. [18] for an overview on applications in stochastic programming. For a presentation on the theory, we refer to the books of Prékopa [29], Ruszczyński and Shapiro [32] and Birge and Louveaux [5].

Such SPPs are in general highly computationally challenging to solve exactly, as shown for example in the works of Dyer and Stougie [7] and Hanasusanto et al. [9]. A way to address them at smaller computational cost consists in constructing a deterministic approximate problem by discretizing the random variables to obtain a finite subset of realizations. These realizations are called *scenarios* and this solution approach is known as the *scenario-tree generation*. The scenario-tree generation is subject to a trade-off. One the one hand, the scenario tree must contain a number of scenarios small enough so that the deterministic approximate problem can be solved efficiently by optimization tools. One the other hand, this number must be large enough to provide accurate estimates of the optimal value of the original SPP. This trade-off is fairly easy to satisfy if the problem has a small number of stages and random variables. However, as the number of stages and random variables increase, it becomes more and more difficult to obtain accurate estimates of the optimal value in a reasonable computational time. As a result, an important challenge in stochastic programming is the design of efficient scenario-tree generation methods for SPPs with many stages and/or random variables.

Many methods have been proposed to generate scenarios and scenario trees, we refer in particular to the following works: Shapiro and Homem-de-Mello [35] and Shapiro [33] on Monte Carlo sampling; Pennanen and Koivu [24], Koivu [17] and Leövey and Römisch [19] on quasi-Monte Carlo methods and integration quadrature rules; Høyland and Wallace [16] and Høyland et al. [15] on moment matching methods; Pflug [25] and Pflug and Pichler [28] on optimal quantization methods; Dupačová et al. [6], Heitsch and Römisch [10] and Growe-Kuska et al. [8] on scenario reduction methods. Each scenario-tree generation method is based on particular theoretical or practical justifications. For instance, Monte Carlo and quasi-Monte Carlo for two-stage SPPs are justified by several statistical results on the rate of convergence and bound on the optimal-value error; see Shapiro and Homem-de-Mello [36], Homem-de-Mello [14], Mak et al. [20] and Bastin et al. [1]. For multistage SPPs, asymptotic consistency of the discretization has been studied first by Olsen [21], and more recently by Pennanen [22] who has developed conditions under which the optimal value and the optimal solutions of the deterministic approximate problem converge toward those of the SPP. The optimal-value error has also been studied using probability metrics, which measure the distance between the true probability distribution of the random variables and its scenario-tree approximation sitting on finitely many scenarios; see Pflug and Pichler [26] for a review on probability metrics. Bounds on the optimal-value error by means of probability metrics have been obtained for instance in the works of Heitsch and Römisch [11] and Pflug and Pichler [27]. The derivation relies on several stability results, we refer to the works of Römisch [31] and Heitsch et al. [12] for a detailed analysis on stability.

As of today, the use of scenario-tree generation methods for SPPs with many stages or random variables is limited by the fast growth of the scenario-tree size. This is observed numerically and was proved in the case of Monte Carlo sampling for multistage SPPs by Shapiro [34]. We think that this limitation arises because scenario trees are often not suitable for the SPP they intend to solve. Indeed, the current methods address any SPP the same way by focusing mostly on the discretization quality of the random variables, with little or no regard to the objective function and the constraints that characterize the problem. However, it is reasonable to doubt that a particular way to generate scenario trees can suit most SPPs. For this reason, we think that it is necessary to identify classes of problems (i.e., problems satisfying similar properties) and to develop scenario-tree generation methods tailored to each class. Additionally, the methods must include a way to look for suitable tree structures. This structure is often considered by default with regular branching coefficients, which explains why scenario-tree sizes become too large when the problem has many stages.

Our motivation for doing this work is to find how to build scenario trees better suited for SPPs, and therefore to broaden the applicability of scenario-tree generation methods to problems with many stages and random parameters. With that goal in mind, we study the optimal-value error that results from approximately solving a SPP with a scenario-tree. Specifically, we derive two main results: an error decomposition and an error upper bound, both written as a sum of errors made at each node of the scenario tree. The error decomposition shows how a particular discretization with a scenario tree affects the optimal-value error. The error bound provides a figure of merit to guide the generation of scenario trees that keep the optimal-value error as small as possible. These results are obtained under general assumptions on the SPP and the scenario tree. In particular, we do not rely on a specific way to generate the scenarios and we do not impose a specific type of structure for the tree.

The remainder of this paper is organized as follows. Section 2 contains the preliminaries of the two main results. In particular, Section 2.1 introduces the notation for the SPP, along with five conditions that the problem must satisfy to ensure that the two main results hold. Section 2.2 introduces a more concise notation for the quantities described in Section 2.1, which will simplify the mathematical developments. Section 2.3 introduces the notation for the scenario tree and the deterministic approximate problem. Section 3 and 4 contain the error decomposition and the error bound, respectively. Finally, Section 5 concludes the paper.

2 Preliminaries

We consider a stochastic programming problem with decisions made at integer time stages t ranging from 0 to $T \in \mathbb{N}^*$, where $\mathbb{N}^*$ stands for the positive integers. *Multistage* problems correspond to the case $T \geq 2$, while *two-stage* problems correspond to $T = 1$. For the sake of conciseness, all results in this paper are formulated as if $T \geq 2$, but the reader can easily deduce the corresponding results for two-stage problems.

2.1 Stochastic programming problem formulation

Stochastic programming problems deal with random parameters that are represented by a discrete-time stochastic process of the form $\boldsymbol{\xi} := (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_T)$, defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$. The random vector $\boldsymbol{\xi}_t$ contains the random parameters revealed during period $(t-1, t)$; it has a probability distribution with support $\Xi_t \subseteq \mathbb{R}^{d_t}$, where $d_t \in \mathbb{N}^*$ can be arbitrary large. Throughout this paper, random vectors are distinguished from their realizations by writing the former in bold font. We denote by Ξ and $\Xi_{..t}$ the supports of $\boldsymbol{\xi}$ and $\boldsymbol{\xi}_{..t} := (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_t)$, respectively, and by $\Xi_t(\xi_{..t-1})$ the conditional support of $\boldsymbol{\xi}_t$ given the event $\{\omega \in \Omega \,|\, \boldsymbol{\xi}_{..t-1}(\omega) = \xi_{..t-1}\} \in \mathcal{A}$. We assume no specific probability distribution for the stochastic process $\boldsymbol{\xi}$.

The decision vector y_t at stage $t \in \{0, \dots, T\}$ is said to satisfy the constraints of the problem if it belongs to the set $Y_t \subseteq \mathbb{R}^{s_t}$ of feasible decision vectors, where $s_t \in \mathbb{N}^*$. For the sake of clarity, and without loss of generality, we assume throughout this paper that $s_t = s$ and $d_t = d$ for all t. When $t \geq 1$, we also denote the set of feasible decision vectors by $Y_t(y_{..t-1}; \xi_{..t})$ to emphasize that it may depend on the sequence of decisions $y_{..t-1} := (y_0, \dots, y_{t-1}) \in \mathbb{R}^{st}$ and on the realization $\xi_{..t} \in \Xi_{..t}$ up to stage t. We consider sets of feasible decision vectors that can be represented as the solutions of finitely many equality and inequality constraints, i.e., in the form given by Condition 1.

Condition 1. (i) The decision vector y_0 belongs to Y_0 if and only if y_0 satisfies $g_{0,i}(y_0) = 0$ for $i \in I_0$ and $g_{0,j}(y_0) \leq 0$ for $j \in J_0$, where $g_{0,i}$ is a continuous function for all $i \in I_0 \cup J_0$ and I_0, J_0 are some finite index sets. (ii) For each $t \in \{1, \dots, T\}$ the following holds: the decision vector y_t belongs to $Y_t(y_{..t-1}; \xi_{..t})$ if and only if y_t satisfies $g_{t,i}(y_{..t}; \xi_{..t}) = 0$ for $i \in I_t$ and $g_{t,j}(y_{..t}; \xi_{..t}) \leq 0$ for $j \in J_t$, where $g_{t,i}$ is a Carathéodory integrand for all $i \in I_t \cup J_t$ and I_t, J_t are some finite index sets.

The functions $g_{t,i}$ define the constraints on the decisions; we refer the reader to the book of Rockafellar and Wets [30, Chapter 14], in particular Example 14.29, for the definition of *Carathéodory integrands*.

We denote by $Z_t(\xi_{..t})$ the set of all feasible decision sequences up to stage $t \in \{1, \dots, T\}$ and for a realization $\xi_{..t}$:

$$Z_t(\xi_{..t}) = \{y_{..t} \in \mathbb{R}^{s(t+1)} \,|\, y_{..t-1} \in Z_{t-1}(\xi_{..t-1}),\ y_t \in Y_t(y_{..t-1}; \xi_{..t})\}, \tag{1}$$

and $Z_0 = Y_0$ at stage 0. Note that the set $Z_t(\xi_{..t})$ is the graph of the multivalued function $Z_{t-1}(\xi_{..t-1}) \ni y_{..t-1} \mapsto Y_t(y_{..t-1}; \xi_{..t})$. Therefore, Condition 1 implies that the sets Z_0 and $Z_t(\xi_{..t})$ are closed for every $\xi_{..t} \in \Xi_{..t}$, and that the multivalued function $\xi_{..t} \mapsto Z_t(\xi_{..t})$ is measurable; see Rockafellar and Wets [30, Theorem 14.36].

We also require a boundedness condition on Z_0 and $Z_t(\xi_{..t})$; together with Condition 1, it ensures that Z_0 and $Z_t(\xi_{..t})$ are compact sets for every $\xi_{..t}$.

Condition 2. The set Z_0 is bounded in $\mathbb{R}^s$. For every $t \in \{1, \dots, T\}$ and every $\xi_{..t} \in \Xi_{..t}$, the set $Z_t(\xi_{..t})$ is bounded in $\mathbb{R}^{s(t+1)}$.

We restrict our attention to stochastic programming problems that have a non-empty set of feasible decision vectors and a relative complete recourse at every stage, as shown in Condition 3.

Condition 3. The set Y_0 is non-empty. The set $Y_t(y_{..t-1}; \xi_{..t})$ is non-empty for every $t \in \{1, \dots, T\}$, every $\xi_{..t} \in \Xi_{..t}$ and every $y_{..t-1} \in Z_{t-1}(\xi_{..t-1})$.

We introduce a *total revenue* function $q : \mathbb{R}^{s(T+1)} \times \Xi \to \mathbb{R}$, whose value $q(y; \xi)$ represents the revenues obtained from stage 0 to T for the sequence of decisions $y = (y_0, \dots, y_T)$ and the realization ξ. The stochastic dynamic programming equations (see Bellman [2] and Bertsekas [4]) for this stochastic programming problem are

$$\widehat{Q}_t(y_{..t-1}; \xi_{..t}) := \sup_{y_t \in Y_t(y_{..t-1}; \xi_{..t})} \widetilde{Q}_t(y_{..t}; \xi_{..t}), \quad \forall t \in \{0, \dots, T\}, \tag{2}$$

$$\widetilde{Q}_t(y_{..t}; \xi_{..t}) := \mathbb{E}[\widehat{Q}_{t+1}(y_{..t}; \boldsymbol{\xi}_{..t+1}) \,|\, \boldsymbol{\xi}_{..t} = \xi_{..t}], \quad \forall t \in \{0, \dots, T-1\}, \tag{3}$$

where for $t = T$ the equation (2) is initialized by setting $\widetilde{Q}_t(y_{..t}; \xi_{..t}) := q(y; \xi)$, and for $t = 0$ we remove the arguments $y_{..t-1}$ and $\xi_{..t}$, i.e., $\widetilde{Q}_t(y_{..t}; \xi_{..t}) = \widetilde{Q}_0(y_0)$ and $\widehat{Q}_t(y_{..t-1}; \xi_{..t}) = \widehat{Q}_0$. We refer to the left-hand sides of (2) and (3) as the stage-t *recourse* function and the stage-t *expected recourse* function, respectively.

To ensure that the (expected) recourse functions are well-defined, we add the following two conditions on the total revenue function.

Condition 4. The function $q(\cdot;\xi)$ is upper semi-continuous for every $\xi \in \Xi$ and $q(\cdot;\cdot)$ is measurable jointly in (y, ξ) (i.e., with respect to the Borel sigma-algebra $\mathcal{B}(\mathbb{R}^{s(T+1)} \times \mathbb{R}^{dT})$).

Condition 4 implies that the total revenue function is a *normal integrand*; see [30, Definition 14.27, Corollary 14.34].

Condition 5. There exists a measurable function $h : \Xi \to \mathbb{R}$ such that $|q(y;\xi)| \leq h(\xi)$ for all $\xi \in \Xi$ and all $y \in Z_T(\xi)$, and h satisfies $\mathbb{E}[|h(\boldsymbol{\xi})|] < +\infty$ and $\mathbb{E}[|h(\boldsymbol{\xi})| \,|\, \boldsymbol{\xi}_{..t} = \xi_{..t}] < +\infty$ for all $t \in \{1, \dots, T-1\}$ and all $\xi_{..t} \in \Xi_{..t}$.

Note that Condition 5 requires that the integrability of $h(\boldsymbol{\xi})$ conditional to $\boldsymbol{\xi}_{..t} = \xi_{..t}$ holds for any $\xi_{..t} \in \Xi_{..t}$ and not almost everywhere on $\Xi_{..t}$. The reason for this is that we want the stage-t (expected) recourse functions to be defined everywhere on $\Xi_{..t}$. This will guarantee that the *node errors* and the *subtree errors*, introduced in Section 3, are well-defined even if the scenarios are chosen in a non-random fashion.

We shall show now that the five conditions above guarantee the existence of optimal decision vectors at every stage of the problem and the finiteness of the (expected) recourse functions. We do so by proving recursively, from stage T to 0, the existence of optimal solutions for the optimisation problem at the right-hand side of (2). In the following, we use the notation $\delta_C(\cdot)$ for the function defined as $\delta_C(x) = 0$ if $x \in C$ and $\delta_C(x) = -\infty$ otherwise. Through this notation, we can express the fact that the supremum of a real-valued function f over a set $C \subseteq \mathbb{R}^n$ is written equivalently as the supremum of the extended-real-valued function $f + \delta_C$ over $\mathbb{R}^n$; see [30, Chapter 1] for detailed developments on extended real analysis.

The total revenue function $(y, \xi) \mapsto \widetilde{Q}_T(y;\xi) = q(y;\xi)$ is a finite-valued normal integrand by Condition 4. It follows from this, and from the condition of measurability and compactness of $Z_T(\xi)$, that $(y, \xi) \mapsto \widetilde{Q}_T(y;\xi) + \delta_{Z_T(\xi)}(y)$ is a normal integrand and, for each $\xi \in \Xi$, $y \mapsto \widetilde{Q}_T(y;\xi) + \delta_{Z_T(\xi)}(y)$ is level-bounded in y_T locally uniformly in $y_{..T-1}$; see [30, Definition 1.16, Example 14.32]. Thus, the stage-T recourse function, which takes the form

$$\widehat{Q}_T(y_{..T-1};\xi) = \sup_{y_T \in \mathbb{R}^s} \{\widetilde{Q}_T(y_{..T-1}, y_T;\xi) + \delta_{Z_T(\xi)}(y_{..T-1}, y_T)\}, \tag{4}$$

is also a normal integrand; see [30, Proposition 14.47]. Moreover, let $\xi \in \Xi$ and let us consider the following two cases: (i) If $y_{..T-1} \in Z_{T-1}(\xi_{..T-1})$, then $Z_T(\xi) \neq \emptyset$, and hence the supremum in (4) is attained, $\widehat{Q}_T(y_{..T-1};\xi)$ is finite and an optimal solution $y_T^* =: x_T^*(y_{..T-1};\xi)$ exists, where we introduce the notation $x_T^*(y_{..T-1};\xi)$ to emphasize that it depends on $y_{..T-1}$ and ξ; (ii) If $y_{..T-1} \notin Z_{T-1}(\xi_{..T-1})$, then the supremum in (4) is $-\infty$, and this value is consistent with the fact that any such $y_{..T-1}$ is not a sequence of feasible decisions. Therefore, for every $\xi \in \Xi$ we have

$$\widehat{Q}_T(y_{..T-1};\xi) \begin{cases} \in \mathbb{R} & \text{if } y_{..T-1} \in Z_{T-1}(\xi_{..T-1}); \\ = -\infty & \text{otherwise.} \end{cases} \tag{5}$$

We shall prove now that the stage-$(T-1)$ expected recourse function

$$\widetilde{Q}_{T-1}(y_{..T-1};\xi_{..T-1}) = \mathbb{E}[\widehat{Q}_T(y_{..T-1};\boldsymbol{\xi}) \,|\, \boldsymbol{\xi}_{..T-1} = \xi_{..T-1}], \tag{6}$$

is a normal integrand, which will allow the above arguments to be repeated at stage $T-1$, and recursively to stage 0. Let $\xi_{..T-1} \in \Xi_{..T-1}$ and let us consider the following two cases: (i) If $y_{..T-1} \in Z_{T-1}(\xi_{..T-1})$, then it follows from Condition 5 and an application of Lebesgue's dominated convergence theorem that $\widetilde{Q}_{T-1}(\,\cdot\,;\xi_{..T-1})$ is finite-valued and upper semi-continuous at $y_{..T-1}$; (ii) If $y_{..T-1} \notin Z_{T-1}(\xi_{..T-1})$, then it follows from (5) that $\widehat{Q}_T(y_{..T-1};\xi_{..T-1}, \xi_T) = -\infty$ for all $\xi_T \in$

$\Xi_T(\xi_{..T-1})$, and hence $\widetilde{Q}_{T-1}(y_{..T-1};\xi_{..T-1}) = -\infty$. We deduce from (i)-(ii) that $\widetilde{Q}_{T-1}(\,\cdot\,;\xi_{..T-1})$ is upper semi-continuous on $\mathbb{R}^{sT}$ for every $\xi_{..T-1} \in \Xi_{..T-1}$. Since $\widetilde{Q}_{T-1}$ is also measurable jointly in $(y_{..T-1};\xi_{..T-1})$, we conclude that $\widetilde{Q}_{T-1}$ is a normal integrand; see [30, Corollary 14.34].

Finally, by carrying out the above arguments recursively to stage 0, we conclude that, for every $t \in \{0,\dots,T\}$, every $\xi_{..t} \in \Xi_{..t}$ and every $y_{..t} \in Z_t(\xi_{..t})$, an optimal decision $x_t^*(y_{..t-1};\xi_{..t})$ exists and the values $\widehat{Q}_t(y_{..t-1};\xi_{..t})$ and $\widetilde{Q}_t(y_{..t};\xi_{..t})$ are finite (with the appropriate change of arguments for $t = 0$).

2.2 Decision policy formulation

For the sake of conciseness, it is more convenient for future developments to introduce a single notation for the stage-t (expected) recourse functions defined in (2)-(3). Previously, we introduced $x_t^*(y_{..t-1};\xi_{..t})$ to denote the stage-t optimal decision vector when the realization is $\xi_{..t}$ and the decisions prior to t are $y_{..t-1}$. We can generalize this notation to represent any feasible decision at stage t as a function of $\xi_{..t}$ and $y_{..t-1}$. The development below formalizes this approach and shows the link with the previous formulation.

We model a *decision policy* $x := (x_0,\dots,x_T)$ as a collection of a decision vector $x_0 \in \mathbb{R}^s$ and several *decision functions* x_1, ..., x_T such that the value $x_t(y_{..t-1};\xi_{..t})$ specifies the stage-t decision. The fact that x_t does not depend on future information is known as the *non-anticipativity* condition. Feasibility conditions are now written in terms of function sets: the set of all stage-t feasible decision functions, denoted by $\mathcal{X}_t$, is defined as

$$\mathcal{X}_1 = \{x_1 : Z_0 \times \Xi_1 \to \mathbb{R}^s \,|\, \forall \xi_1 \in \Xi_1,\, \forall y_0 \in Z_0,\, x_1(y_0;\xi_1) \in Y_1(y_0;\xi_1)\}, \tag{7}$$

and for each $t \in \{2,\dots,T\}$,

$$\begin{aligned}\mathcal{X}_t = \Big\{x_t : \bigcup_{\xi_{..t}\in\Xi_{..t}} (Z_{t-1}(\xi_{..t-1}) \times \{\xi_{..t}\}) \to \mathbb{R}^s \,\Big|\, \\ \forall \xi_{..t} \in \Xi_{..t},\, \forall y_{..t-1} \in Z_{t-1}(\xi_{..t-1}),\, x_t(y_{..t-1};\xi_{..t}) \in Y_t(y_{..t-1};\xi_{..t})\Big\}.\end{aligned} \tag{8}$$

At stage 0, we have $\mathcal{X}_0 = Z_0$ ($\mathcal{X}_0$ is a vector set). The set of all feasible decision policies, denoted by $\mathcal{X}$, is given by $\mathcal{X} = \Pi_{t=0}^T \mathcal{X}_t$, where $\Pi_{t=0}^T$ denotes the Cartesian product. The link between decision policies and decision vectors is as follows: for a decision policy $x \in \mathcal{X}$ and a realization $\xi \in \Xi$, the associated decision sequence $y = (y_0,\dots,y_T)$ is given by

$$y_t = \begin{cases} x_0 & \text{if } t = 0; \\ x_t(y_{..t-1};\xi_{..t}) & \text{if } t \in \{1,\dots,T\}. \end{cases} \tag{9}$$

We introduce the (generalized) stage-t recourse function $Q_t : \mathcal{X} \times \Xi_{..t} \to \mathbb{R}$, for every $t \in \{1,\dots,T\}$, whose value $Q_t(x;\xi_{..t})$ represents the conditional expectation of the total revenues obtained by implementing the policy $x \in \mathcal{X}$ given the realization $\xi_{..t} \in \Xi_{..t}$. At stage 0, we define $Q_0 : \mathcal{X} \to \mathbb{R}$. The recourse function at stage $t \in \{0,\dots,T-1\}$ is obtained from the recourse function at stage $t+1$ by the relation

$$Q_t(x;\xi_{..t}) = \mathbb{E}\big[Q_{t+1}(x;\boldsymbol{\xi}_{..t+1}) \,\big|\, \boldsymbol{\xi}_{..t} = \xi_{..t}\big], \quad \forall x \in \mathcal{X},\, \forall \xi_{..t} \in \Xi_{..t}, \tag{10}$$

where for $t = 0$ we remove the argument $\xi_{..t}$, i.e., $Q_0(x) = \mathbb{E}\big[Q_1(x;\boldsymbol{\xi}_1)\big]$, and the relation is initialized at stage T by setting $Q_T(x;\xi) = q(y;\xi)$ with y given by (9).

In this setting, an optimal decision policy for the stochastic programming problem is a decision policy $x^* = (x_0^*,\dots,x_T^*) \in \mathcal{X}$ satisfying the following inequalities:

- at stage T:

$$Q_T(x_{..T-1}, x_T^*; \xi) \geq Q_T(x_{..T-1}, x_T; \xi), \quad \forall \xi \in \Xi, \forall x \in \mathcal{X}; \tag{11}$$

- at stage $t \in \{1, \ldots, T-1\}$:

$$Q_t(x_{..t-1}, x_t^*, x_{t+1..}^*; \xi_{..t}) \geq Q_t(x_{..t-1}, x_t, x_{t+1..}^*; \xi_{..t}), \quad \forall \xi_{..t} \in \Xi_{..t}, \forall x_{..t} \in \Pi_{i=0}^t \mathcal{X}_i; \tag{12}$$

- at stage 0:

$$Q_0(x_0^*, x_{1..}^*) \geq Q_0(x_0, x_{1..}^*), \quad \forall x_0 \in \mathcal{X}_0, \tag{13}$$

where we use the shorthand $x_{..t} := (x_0, \ldots, x_t)$ and $x_{t..}^* := (x_t^*, \ldots, x_T^*)$. The value $Q_0(x^*)$ is the optimal value of the stochastic programming problem.

Intuitively, the inequalities (11)-(13) mean the following: when using $x_{t+1..}^*$ to make decisions at stages $t+1$ to T, the stage-t decision function x_t^* is optimal for the function $x_t \mapsto Q_t(x_{..t-1}, x_t, x_{t+1..}^*; \xi_{..t})$ regardless of the arbitrary policy $x_{..t-1} \in \Pi_{i=0}^{t-1} \mathcal{X}_i$ used to make decisions at stages 0 to $t-1$.

It follows from the five conditions introduced in Section 2.1 that both sides of the inequalities (11)-(13) are well-defined and finite-valued for any feasible policy and any random realization.

2.3 Scenario-tree formulation

The optimal value $Q_0(x^*)$ and the optimal policy x^* of the stochastic programming problem are not readily available in general. The scenario-tree approach to estimate Q_t consists in approximating the right-hand side of (10) as a weighted average of the values of Q_{t+1} for a selection of realizations of $\boldsymbol{\xi}_{t+1}$. In turn, Q_{t+1} is approximated in terms of Q_{t+2}, and this recursive discretization scheme is carried out to stage T, where the values of Q_T are computed directly from the total revenue function q. A tree structure naturally arises from this scheme, in which sibling nodes at stage $t+1$ represent the discrete values of Q_{t+1} whose weighted average approximates the value of Q_t, represented by their common parent node at stage t. The remainder of this section describes the scenario-tree formulation, the corresponding decision functions and the deterministic approximate problem.

The scenario tree is a rooted tree structure $\mathcal{T} = (\mathcal{N}, \mathcal{E})$, with $\mathcal{N}$ the (finite) node set, $\mathcal{E}$ the edge set and n_0 the root node. The structure is such that T edges separate the root from any of the tree leaves. We introduce the notation $C(n)$, $a(n)$ and $t(n)$ to denote, respectively, the set of children nodes of n (the node(s) linked to n at the next stage), the ancestor node of n (the node linked to n at the previous stage) and the stage of n. We also denote $\mathcal{N}^* := \mathcal{N} \setminus \{n_0\}$ and $\mathcal{N}_t := \{n \in \mathcal{N} \mid t(n) = t\}$.

Every node $n \in \mathcal{N}^*$ of the scenario tree carries a positive weight $w^n > 0$ and a discretization point ζ^n of $\boldsymbol{\xi}_{t(n)}$. The latter satisfies

$$\zeta^n \in \begin{cases} \Xi_1 & \text{if } n \in \mathcal{N}_1; \\ \Xi_{t(n)}(\zeta^{..a(n)}) & \text{if } n \in \mathcal{N}^* \setminus \mathcal{N}_1, \end{cases} \tag{14}$$

where $\zeta^{..a(n)}$ denotes the collection of all discretization points on the path from n_0 to $a(n)$, i.e., $\zeta^{..a(n)} := (\zeta^{a^{t(n)-1}(n)}, \ldots, \zeta^{a^2(n)}, \zeta^{a(n)})$ with $a^k(n)$ the k-th ancestor node of n (the node linked to n at the k-th previous stage). The value w^n represents the weight of node n with respect to its sibling nodes. We also define the *absolute weight* W^n of node $n \in \mathcal{N}$, which represents its weight with respect to the whole scenario tree:

$$W^n = \begin{cases} 1 & \text{if } n = n_0; \\ w^n w^{a(n)} \ldots w^{a^{t(n)-1}(n)} & \text{if } n \in \mathcal{N}^*. \end{cases} \tag{15}$$

The feasible decision functions for the deterministic approximate problem are defined in a way similar to (7)-(8): the set of all stage-t feasible decision functions for the approximate problem, denoted by $\widehat{\mathcal{X}}_t$, is given by

$$\widehat{\mathcal{X}}_1 = \{x_1 : Z_0 \times \{\zeta^n \,|\, n \in \mathcal{N}_1\} \to \mathbb{R}^s \,\big|\, \forall n \in \mathcal{N}_1,\, \forall y_0 \in Z_0,\, x_1(y_0; \zeta^n) \in Y_1(y_0; \zeta^n)\}, \tag{16}$$

and for each $t \in \{2, \ldots, T\}$,

$$\widehat{\mathcal{X}}_t = \{x_t : \bigcup_{n \in \mathcal{N}_t} (Z_{t-1}(\zeta^{..a(n)}) \times \{\zeta^{..n}\}) \to \mathbb{R}^s \,\big|\, \forall n \in \mathcal{N}_t,\, \forall y_{..t-1} \in Z_{t-1}(\zeta^{..a(n)}),\, x_t(y_{..t-1}; \zeta^{..n}) \in Y_t(y_{..t-1}; \zeta^{..n})\}. \tag{17}$$

At stage 0, we have $\widehat{\mathcal{X}}_0 = Z_0$ (hence $\widehat{\mathcal{X}}_0 = \mathcal{X}_0$). The set of all feasible decision policies, denoted by $\widehat{\mathcal{X}}$, is given by $\widehat{\mathcal{X}} = \Pi_{t=0}^T \widehat{\mathcal{X}}_t$.

We emphasize that in a general setting there is no inclusion relation between the sets $\mathcal{X}_t$ and $\widehat{\mathcal{X}}_t$, because $\mathcal{X}_t$ contains functions defined on $\Xi_{..t}$, whereas the functions in $\widehat{\mathcal{X}}_t$ are defined on $\{\zeta^{..n} \,|\, n \in \mathcal{N}_t\}$. The set $\{\zeta^{..n} \,|\, n \in \mathcal{N}_t\}$ is a strict subset of $\Xi_{..t}$ whenever the scenario tree does not include all realizations of the stochastic process. It is also important to note that a decision policy $x \in \mathcal{X}$ carries more information than a decision policy $x' \in \widehat{\mathcal{X}}$. Indeed, we can use x to make decisions in the deterministic approximate problem, but we cannot use x' to make decisions in the stochastic programming problem. This is true in general, however, a subtlety arises when the stage-t realization $\xi_{..t}$ coincides with a discretization sequence $\zeta^{..n}$ for some $n \in \mathcal{N}_t$. In that case, any decision policy $x'_{..t} \in \Pi_{i=0}^t \widehat{\mathcal{X}}_i$ can be used to make decisions at stages 0 to t in the stochastic programming problem. This leads us to extend the domain of definition of the stage-t recourse function $Q_t(\,\cdot\,; \xi_{..t})$ to include this particular case; the new definition is

$$Q_t(\,\cdot\,; \xi_{..t}) : \begin{cases} \Pi_{i=0}^t (\mathcal{X}_i \cup \widehat{\mathcal{X}}_i) \times \Pi_{i=t+1}^T \mathcal{X}_i \to \mathbb{R} & \text{if } \xi_{..t} = \zeta^{..n} \text{ for some } n \in \mathcal{N}_t; \\ \mathcal{X} \to \mathbb{R} & \text{otherwise.} \end{cases} \tag{18}$$

At stage 0, we still have $Q_0 : \mathcal{X} \to \mathbb{R}$.

The scenario tree provides estimators for the recourse functions (10). The node-n estimator of the stage-$t(n)$ recourse function $Q_{t(n)}(x; \zeta^{..n})$ is denoted by $\widehat{Q}^n(x)$ and is computed recursively from the estimators at node $m \in C(n)$ by

$$\widehat{Q}^n(x) = \sum_{m \in C(n)} w^m \, \widehat{Q}^m(x), \quad \forall n \in \mathcal{N} \setminus \mathcal{N}_T,\, \forall x \in \Pi_{t=0}^T (\mathcal{X}_t \cup \widehat{\mathcal{X}}_t). \tag{19}$$

At node $n \in \mathcal{N}_T$ the relation is initialized by setting $\widehat{Q}^n(x) = q(y; \zeta^{..n})$, where y is obtained from the policy x and the scenario $\zeta^{..n}$ by (9).

We emphasize that our formulation of the scenario tree estimators is general, since we do not rely on a specific way to obtain the tree structure, the discretization points and the weights. Moreover, we do not require that the weights w^m in (19) sum to one. This generality allows us to cover numerous ways to generate scenario trees. For instance, the fact that the weights need not sum to one can account for the use of *importance sampling* or *quadrature rules* methods; see, e.g., Shapiro [33] and Pennanen [23]. A well-known particular case of the scenario-tree estimator is the so-called *sample average approximation*, which is obtained from (19) by setting $w^n = 1/|C(a(n))|$ and by getting ζ^n through a Monte Carlo sampling method. As a matter of fact, all scenario-tree generation methods cited in Section 1 can be described using the above formulation.

The existence of an optimal decision policy for the deterministic approximate problem follows from the same arguments as in Section 2.1. This optimal decision policy, denoted by $\widehat{x} = (\widehat{x}_0, \ldots, \widehat{x}_T) \in \widehat{\mathcal{X}}$, satisfies the discrete counterpart of (11)-(13):

- at stage T:

$$\widehat{Q}^n(x_{..T-1}, \widehat{x}_T) \geq \widehat{Q}^n(x_{..T-1}, x_T), \quad \forall n \in \mathcal{N}_T,\, \forall x \in \Pi_{t=0}^{T}(\mathcal{X}_t \cup \widehat{\mathcal{X}}_t); \tag{20}$$

- at stage $t \in \{1, \dots, T-1\}$:

$$\widehat{Q}^n(x_{..t-1}, \widehat{x}_t, \widehat{x}_{t+1..}) \geq \widehat{Q}^n(x_{..t-1}, x_t, \widehat{x}_{t+1..}), \quad \forall n \in \mathcal{N}_t,\, \forall x_{..t} \in \Pi_{i=0}^{t}(\mathcal{X}_i \cup \widehat{\mathcal{X}}_i); \tag{21}$$

- at stage 0:

$$\widehat{Q}^{n_0}(\widehat{x}_0, \widehat{x}_{1..}) \geq \widehat{Q}^{n_0}(x_0, \widehat{x}_{1..}), \quad \forall x_0 \in \widehat{\mathcal{X}}_0. \tag{22}$$

The value $\widehat{Q}^{n_0}(\widehat{x})$ is the optimal value of the deterministic approximate problem; it is the estimator of $Q_0(x^*)$ and the value $\widehat{Q}^{n_0}(\widehat{x}) - Q_0(x^*)$ is what we refer to as the optimal-value error. It follows from the five conditions in Section 2.1 that both sides of (20)-(22) are well-defined and finite-valued for any feasible decision policy and any node in the scenario tree.

We end this section by a remark on two cases of equality between the recourse functions and its estimators.

Remark 2.1. Since the stage-T recourse function and its estimator at any node $n \in \mathcal{N}_T$ are both computed directly from the total revenue function, we have that

$$\widehat{Q}^n(x) = Q_T(x; \zeta^{..n}), \quad \forall n \in \mathcal{N}_T,\, \forall x \in \Pi_{t=0}^{T}(\mathcal{X}_t \cup \widehat{\mathcal{X}}_t). \tag{23}$$

Another case of equality is obtained by noticing that the inequality (11) applied at $\xi = \zeta^{..n}$, for any $n \in \mathcal{N}_T$, provides the same optimality condition than (20). Consequently, the decision functions $\widehat{x}_T(\cdot; \zeta^{..n})$ and $x_T^*(\cdot; \zeta^{..n})$ coincide, and hence

$$\widehat{Q}^n(x_{..T-1}, \widehat{x}_T) = Q_T(x_{..T-1}, x_T^*; \zeta^{..n}), \quad \forall n \in \mathcal{N}_T,\, \forall x_{..T-1} \in \Pi_{t=0}^{T-1}(\mathcal{X}_t \cup \widehat{\mathcal{X}}_t). \tag{24}$$

3 Decomposition of the scenario-tree optimal-value error

The main result of this section is Theorem 3.5, which provides a node-by-node decomposition of the scenario-tree optimal-value error $|\widehat{Q}^{n_0}(\widehat{x}) - Q_0(x^*)|$.

We start by introducing the concepts of *node errors* and of *subtree errors*. We see from the stochastic dynamic programming equations (2)-(3) that the optimal-value error results from several errors made by approximating the right-hand sides of (2) and (3), alternatively. We call *node optimization error* and *node discretization error* the errors made by approximating (2) and (3), respectively, at a particular node in the scenario-tree. Their definitions are given explicitly below, by means of the decision policy formulation of Section 2.2.

Definition 3.1 (Node optimization error)**.** *For each stage $t \in \{1, \dots, T-1\}$, we define the* optimization error $\mathrm{E}^n_{\mathrm{opt}}(x_{..t-1})$ *at node $n \in \mathcal{N}_t$ and for a decision policy $x_{..t-1} \in \Pi_{i=0}^{t-1}(\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$ as*

$$\mathrm{E}^n_{\mathrm{opt}}(x_{..t-1}) = Q_t(x_{..t-1}, \widehat{x}_t, x^*_{t+1..}; \zeta^{..n}) - Q_t(x_{..t-1}, x^*_t, x^*_{t+1..}; \zeta^{..n}). \tag{25}$$

At the root node, the optimization error *is*

$$\mathrm{E}^{n_0}_{\mathrm{opt}} = Q_0(\widehat{x}_0, x^*_{1..}) - Q_0(x^*_0, x^*_{1..}). \tag{26}$$

The optimization error is always nonpositive (see the optimality conditions (11)-(13)).

The node-n optimization error measures the error on the stage-$t(n)$ recourse function (2), for $\xi_{..t(n)} = \zeta^{..n}$, made by using at this stage a decision function that is optimal for the deterministic approximate problem instead of the optimal decision function of the stochastic programming problem. We do not define the node optimization error at stage T, because its value would be zero for any $n \in \mathcal{N}_T$ (see Remark 2.1).

Definition 3.2 (Node discretization error)**.** *For each stage $t \in \{0, \dots, T-1\}$, we define the* discretization error $\mathrm{E}^n_{\mathrm{disc}}(x_{..t})$ *at node $n \in \mathcal{N}_t$ and for a decision policy $x_{..t} \in \Pi_{i=0}^t (\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$ as*

$$\mathrm{E}^n_{\mathrm{disc}}(x_{..t}) = \sum_{m \in C(n)} w^m \, Q_{t+1}(x_{..t}, x^*_{t+1..}; \zeta^{..m}) - Q_t(x_{..t}, x^*_{t+1..}; \zeta^{..n}), \tag{27}$$

if $t \in \{1, \dots, T-1\}$, and

$$\mathrm{E}^{n_0}_{\mathrm{disc}}(x_0) = \sum_{m \in C(n_0)} w^m \, Q_1(x_0, x^*_{1..}; \zeta^m) - Q_0(x_0, x^*_{1..}), \tag{28}$$

at the root node.

The node-n discretization error measures the error on the stage-$t(n)$ expected recourse function (3), for $\xi_{..t(n)} = \zeta^{..n}$, made by substituting the expectation with a finite sum over the children nodes of n.

We define now the concept of *subtree errors*. We call a *subtree rooted at node* $n \in \mathcal{N}$ the scenario tree $\mathcal{T}(n) = (\mathcal{N}(n), \mathcal{E}(n))$ obtained by setting n as the root node and by considering only the nodes that are the descendants of n, i.e., with $\mathcal{N}(n) = \{n' \in \mathcal{N} \,|\, \exists k \geq 0 \text{ such that } a^k(n') = n\}$ and $\mathcal{E}(n) = \{(n', n'') \in \mathcal{E} \,|\, (n', n'') \in \mathcal{N}(n) \times \mathcal{N}(n)\}$. Obviously, the subtree rooted at n_0 is the whole scenario tree $(\mathcal{N}, \mathcal{E})$. We distinguish between two subtree errors, which we refer to as optimal and suboptimal. The *optimal subtree error* at node n measures the error between the stage-$t(n)$ recourse function (2), for $\xi_{..t(n)} = \zeta^{..n}$, and its node-$n$ scenario-tree estimator. The *suboptimal subtree error* at node n measures the error between the stage-$t(n)$ expected recourse function (3), for $\xi_{..t(n)} = \zeta^{..n}$, and its node-$n$ scenario-tree estimator.

Definition 3.3 (Subtree errors)**.** *(a) For each stage $t \in \{1, \dots, T\}$, we define the* optimal subtree error $\Delta Q^n(x_{..t-1})$ *at node $n \in \mathcal{N}_t$ and for a decision policy $x_{..t-1} \in \Pi_{i=0}^{t-1} (\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$ as*

$$\Delta Q^n(x_{..t-1}) = \widehat{Q}^n(x_{..t-1}, \widehat{x}_{t..}) - Q_t(x_{..t-1}, x^*_{t..}; \zeta^{..n}). \tag{29}$$

At the root node, the optimal subtree error *is*

$$\Delta Q^{n_0} = \widehat{Q}^{n_0}(\widehat{x}) - Q_0(x^*). \tag{30}$$

(b) For each stage $t \in \{0, \dots, T\}$, we define the suboptimal subtree error $\Delta Q^n_{\mathrm{sub}}(x_{..t})$ *at node $n \in \mathcal{N}_t$ and for a decision policy $x_{..t} \in \Pi_{i=0}^t (\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$ as*

$$\Delta Q^n_{\mathrm{sub}}(x_{..t}) = \begin{cases} \widehat{Q}^n(x) - Q_T(x; \zeta^{..n}) & \text{if } t = T; \\ \widehat{Q}^n(x_{..t}, \widehat{x}_{t+1..}) - Q_t(x_{..t}, x^*_{t+1..}; \zeta^{..n}) & \text{if } t \in \{1, \dots, T-1\}; \\ \widehat{Q}^{n_0}(x_0, \widehat{x}_{1..}) - Q_0(x_0, x^*_{1..}) & \text{if } t = 0. \end{cases} \tag{31}$$

For every node $n \in \mathcal{N}_T$, the subtree errors $\Delta Q^n_{\mathrm{sub}}$ and ΔQ^n are identically zero (see Remark 2.1). In the general setting of the scenario-tree formulation of Section 2.3, we do not know whether the subtree errors have positive or negative values. The node-n_0 optimal subtree error (30) is the scenario-tree optimal-value error that we want to decompose and bound.

The optimal and suboptimal subtree errors at node n gather implicitly all the node errors (optimization and discretization) made in the subtree rooted at n. To find an explicit relation between the subtree errors and the node errors, we need to be able to derive a closed-form representation of a quantity at node n from a recursive representation of this quantity over the nodes in the subtree rooted at n. This is the purpose of the following lemma.

Lemma 3.4. *Let a real value γ^n be assigned to every node $n \in \mathcal{N} \setminus \mathcal{N}_T$ of the scenario tree.*

(a) If a sequence $\{\alpha^n\}$ satisfies the recurrence relation

$$\alpha^n = \begin{cases} \gamma^n + \sum_{m \in C(n)} w^m \alpha^m & \text{if } n \in \mathcal{N} \setminus \mathcal{N}_T; \quad (32) \\ 0 & \text{if } n \in \mathcal{N}_T, \quad (33) \end{cases}$$

then α^n has a closed-form representation at each node $n \in \mathcal{N} \setminus \mathcal{N}_T$ given by

$$\alpha^n = \frac{1}{W^n} \sum_{m \in \mathcal{N}(n) \setminus \mathcal{N}_T} W^m \gamma^m, \tag{34}$$

where $\mathcal{N}(n)$ is the node set of the subtree rooted at n.

(b) If a sequence $\{\beta^n\}$ satisfies the recurrence relation

$$\beta^n \leq \begin{cases} \gamma^n + \sum_{m \in C(n)} w^m \beta^m & \text{if } n \in \mathcal{N} \setminus \mathcal{N}_T; \quad (35) \\ 0 & \text{if } n \in \mathcal{N}_T, \quad (36) \end{cases}$$

then β^n has an upper bound at each node $n \in \mathcal{N} \setminus \mathcal{N}_T$ given by

$$\beta^n \leq \frac{1}{W^n} \sum_{m \in \mathcal{N}(n) \setminus \mathcal{N}_T} W^m \gamma^m. \tag{37}$$

Parts (a) and (b) will be used in deriving the error decomposition theorem and the error bound theorem, respectively.

Proof. (a) Let $\{u^n\}$ and $\{v^n\}$ denote two sequences satisfying the recurrence relation (32)-(33) and the closed-form (34), respectively. We will show by induction that $u^n = v^n$ holds for every node $n \in \mathcal{N} \setminus \mathcal{N}_T$.

Basis. Take an arbitrary $n \in \mathcal{N}_{T-1}$. We have that $\mathcal{N}(n) \setminus \mathcal{N}_T = \{n\}$, hence it follows from (34) that

$$v^n = \frac{1}{W^n} W^n \gamma^n = \gamma^n = u^n. \tag{38}$$

Inductive step. Suppose that $u^m = v^m$ holds for every $m \in \mathcal{N}_t$ for a given stage $t \in \{1, \dots, T-1\}$. Take an arbitrary $n \in \mathcal{N}_{t-1}$. From (34), and using the following decomposition of $\mathcal{N}(n) \setminus \mathcal{N}_T$:

$$\mathcal{N}(n) \setminus \mathcal{N}_T = \{n\} \cup \Big(\bigcup_{m \in C(n)} \mathcal{N}(m) \setminus \mathcal{N}_T \Big), \tag{39}$$

it follows that

$$v^n = \gamma^n + \frac{1}{W^n} \sum_{m\in C(n)} \left[\sum_{l\in\mathcal{N}(m)\setminus\mathcal{N}_T} W^l\,\gamma^l \right] \tag{40}$$

$$= \gamma^n + \frac{1}{W^n} \sum_{m\in C(n)} \left[W^m\,\frac{1}{W^m} \sum_{l\in\mathcal{N}(m)\setminus\mathcal{N}_T} W^l\,\gamma^l \right] \tag{41}$$

$$= \gamma^n + \frac{1}{W^n} \sum_{m\in C(n)} W^m\,v^m \tag{42}$$

$$= \gamma^n + \sum_{m\in C(n)} w^m\,u^m \tag{43}$$

$$= u^n, \tag{44}$$

where the equality (43) holds by the induction hypothesis and by the relation $W^m = W^n w^m$. This proves the inductive step and therefore the final result.

(b) Let $\{\alpha^n\}$ and $\{\beta^n\}$ denote two sequences satisfying the recurrence relation (32)-(33) and (35)-(36), respectively. We will show by induction that $\beta^n \leq \alpha^n$ holds for every node $n \in \mathcal{N}\setminus\mathcal{N}_T$.

Basis. For every node $n\in\mathcal{N}_{T-1}$, it follows from (35)-(36) that $\beta^n \leq \gamma^n = \alpha^n$.

Inductive step. Suppose that $\beta^m \leq \alpha^m$ holds for every node $m\in\mathcal{N}_t$ for a given stage $t\in\{1,\ldots,T-1\}$. Take an arbitrary $n\in\mathcal{N}_{t-1}$. From (35), and using the induction hypothesis, we have that

$$\beta^n \leq \gamma^n + \sum_{m\in C(n)} w^m\,\alpha^m = \alpha^n, \tag{45}$$

which proves the inductive step. The inequality (37) follows immediately using part (a) of this lemma. □

Theorem 3.5. *The scenario-tree optimal-value error can be decomposed into a weighted sum of node discretization errors and node optimization errors as follows:*

$$\Delta Q^{n_0} = \sum_{n\in\mathcal{N}\setminus\mathcal{N}_T} W^n\,[\mathrm{E}^n_{\mathrm{opt}}(\widehat{x}_{..t(n)-1}) + \mathrm{E}^n_{\mathrm{disc}}(\widehat{x}_{..t(n)})], \tag{46}$$

where for $n=n_0$ the term $\mathrm{E}^n_{\mathrm{opt}}(\widehat{x}_{..t(n)-1})$ corresponds to $\mathrm{E}^{n_0}_{\mathrm{opt}}$.

Proof. We start by deriving a recurrence relation for the optimal subtree error at node $n\in\mathcal{N}\setminus\mathcal{N}_T$, by considering successively the cases $t=0$ and $t\in\{1,\ldots,T-1\}$.

At the root node, using successively (19), (26) and (28), we can write ΔQ^{n_0} as follows:

$$\Delta Q^{n_0} = \widehat{Q}^{n_0}(\widehat{x}) - Q_0(x^*) \tag{47}$$

$$= \sum_{m\in C(n_0)} w^m\,\widehat{Q}^m(\widehat{x}) - Q_0(x^*) \tag{48}$$

$$= \sum_{m\in C(n_0)} w^m \left[\widehat{Q}^m(\widehat{x}) - Q_1(\widehat{x}_0, x^*_{1..};\zeta^m)\right] \tag{49}$$

$$+ \sum_{m\in C(n_0)} w^m Q_1(\widehat{x}_0, x^*_{1..};\zeta^m) - Q_0(\widehat{x}_0, x^*_{1..}) \tag{50}$$

$$+ Q_0(\widehat{x}_0, x^*_{1..}) - Q_0(x^*) \tag{51}$$

$$= \sum_{m\in C(n_0)} w^m\,\Delta Q^m(\widehat{x}_0) + \mathrm{E}^{n_0}_{\mathrm{disc}}(\widehat{x}_0) + \mathrm{E}^{n_0}_{\mathrm{opt}}. \tag{52}$$

For every $t \in \{1, \dots, T-1\}$ and every $n \in \mathcal{N}_t$, using successively (19), (27) and (25), we can write $\Delta Q^n(\widehat{x}_{..t-1})$ as follows:

$$\begin{aligned}
\Delta Q^n(\widehat{x}_{..t-1}) &= \widehat{Q}^n(\widehat{x}) - Q_t(\widehat{x}_{..t-1}, x^*_{t..}; \zeta^{..n}) && (53)\\
&= \sum_{m \in C(n)} w^m\, \widehat{Q}^m(\widehat{x}) - Q_t(\widehat{x}_{..t-1}, x^*_{t..}; \zeta^{..n}) && (54)\\
&= \sum_{m \in C(n)} w^m \Big[\widehat{Q}^m(\widehat{x}) - Q_{t+1}(\widehat{x}_{..t}, x^*_{t+1..}; \zeta^{..m})\Big] && (55)\\
&\quad + \sum_{m \in C(n)} w^m\, Q_{t+1}(\widehat{x}_{..t}, x^*_{t+1..}; \zeta^{..m}) - Q_t(\widehat{x}_{..t}, x^*_{t+1..}; \zeta^{..n}) \\
&\quad + Q_t(\widehat{x}_{..t}, x^*_{t+1..}; \zeta^{..n}) - Q_t(\widehat{x}_{..t-1}, x^*_{t..}; \zeta^{..n}) && (56)\\
&= \sum_{m \in C(n)} w^m\, \Delta Q^m(\widehat{x}_{..t}) + \mathrm{E}^n_{\mathrm{disc}}(\widehat{x}_{..t}) + \mathrm{E}^n_{\mathrm{opt}}(\widehat{x}_{..t-1}). && (57)
\end{aligned}$$

Finally, by defining

$$\gamma^n = \begin{cases} \mathrm{E}^{n_0}_{\mathrm{disc}}(\widehat{x}_0) + \mathrm{E}^{n_0}_{\mathrm{opt}} & \text{if } n = n_0; \quad (58)\\ \mathrm{E}^n_{\mathrm{disc}}(\widehat{x}_{..t(n)}) + \mathrm{E}^n_{\mathrm{opt}}(\widehat{x}_{..t(n)-1}) & \text{if } n \in \mathcal{N}^* \setminus \mathcal{N}_T, \quad (59) \end{cases}$$

we see that the sequence $\{\Delta Q^n\}$ satisfies the recurrence relation (32)-(33) (recall that $\Delta Q^n = 0$ for every $n \in \mathcal{N}_T$; see Remark 2.1). Thus, the decomposition (46) follows directly from (34) applied at the root node. $\square$

4 Upper bound on the scenario-tree optimal-value error

The error decomposition of Theorem 3.5, although useful to enlight the contributions of two types of errors in the optimal-value error, cannot be directly used to guide the generation of scenario trees. The reason is that it features node optimization errors, which are difficult to quantity since they depend on the scenario tree solely via the optimal policy $\widehat{x}$. Node discretization errors, conversely, depend directly on the characteristics of a scenario tree, i.e., the structure, the discretization points and the weights. Moreover, discretization errors enjoy a large litterature in numerical integration, such as in quasi-Monte Carlo theory, where they are more often refered to as integration errors (in this paper, we use the term "discretization error" when the integrand is the recourse functions (Definition 3.2) and "integration error" when the integrand is any integrable function (Definition 4.4)).

The main result of this section is Theorem 4.3, which provides an upper bound on the optimal-value that features only node discretization errors. Its derivation does not rely on the decomposition of Theorem 3.5, it is based on the following two lemmas.

Lemma 4.1. *For each stage $t \in \{1, \dots, T\}$, node $n \in \mathcal{N}_t$ and decision policy $x_{..t-1} \in \Pi_{i=0}^{t-1}(\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$, the following holds:*

$$|\Delta Q^n(x_{..t-1})| \le \max_{u \in \{\widehat{x}_t, x^*_t\}} |\Delta Q^n_{\mathrm{sub}}(x_{..t-1}, u)|, \tag{60}$$

and at the root node:

$$|\Delta Q^{n_0}| \le \max_{u \in \{\widehat{x}_0, x^*_0\}} |\Delta Q^{n_0}_{\mathrm{sub}}(u)|. \tag{61}$$

Proof. If $\Delta Q^{n_0} \geq 0$, then it follows from (13) that

$$\begin{aligned} |\Delta Q^{n_0}| &= \widehat{Q}^{n_0}(\widehat{x}) - Q_0(x^*) && (62)\\ &\leq \widehat{Q}^{n_0}(\widehat{x}) - Q_0(\widehat{x}_0, x^*_{1..}) && (63)\\ &= \Delta Q^{n_0}_{\text{sub}}(\widehat{x}_0). && (64) \end{aligned}$$

Conversely, if $\Delta Q^{n_0} < 0$, then it follows from (22) that

$$\begin{aligned} |\Delta Q^{n_0}| &= -\widehat{Q}^{n_0}(\widehat{x}) + Q_0(x^*) && (65)\\ &\leq -\widehat{Q}^{n_0}(x^*_0, \widehat{x}_{1..}) + Q_0(x^*) && (66)\\ &= -\Delta Q^{n_0}_{\text{sub}}(x^*_0). && (67) \end{aligned}$$

This proves the result at the root node.

Similarly, we show now that the result holds for any $t \in \{1, \dots, T-1\}$, $n \in \mathcal{N}_t$ and $x_{..t-1} \in \Pi_{i=0}^{t}(\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$. If $\Delta Q^n(x_{..t-1}) \geq 0$, then it follows from (12) that

$$\begin{aligned} |\Delta Q^n(x_{..t-1})| &= \widehat{Q}^n(x_{..t-1}, \widehat{x}_{t..}) - Q_t(x_{..t-1}, x^*_{t..}; \zeta^{..n}) && (68)\\ &\leq \widehat{Q}^n(x_{..t-1}, \widehat{x}_{t..}) - Q_t(x_{..t-1}, \widehat{x}_t, x^*_{t+1..}; \zeta^{..n}) && (69)\\ &= \Delta Q^n_{\text{sub}}(x_{..t-1}, \widehat{x}_t). && (70) \end{aligned}$$

If $\Delta Q^n(x_{..t-1}) < 0$, then it follows from (21) that

$$\begin{aligned} |\Delta \widehat{Q}^n(x_{..t-1})| &= -Q^n(x_{..t-1}, \widehat{x}_{t..}) + Q_t(x_{..t-1}, x^*_{t..}; \zeta^{..n}) && (71)\\ &\leq -\widehat{Q}^n(x_{..t-1}, x^*_t, \widehat{x}_{t+1..}) + Q_t(x_{..t-1}, x^*_{t..}; \zeta^{..n}) && (72)\\ &= -\Delta Q^n_{\text{sub}}(x_{..t-1}, x^*_t). && (73) \end{aligned}$$

This proves the inequality (60) for any $t \in \{1, \dots, T-1\}$. The inequality for $t = T$ holds trivially since $\Delta Q^n(x_{..T-1}) = \Delta Q^n_{\text{sub}}(x) = 0$ for all $n \in \mathcal{N}_T$ and all $x \in \Pi_{i=0}^{T}(\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$ (see Remark 2.1). $\square$

Lemma 4.2. *For each stage $t \in \{1, \dots, T-1\}$, node $n \in \mathcal{N}_t$ and decision policy $x_{..t-1} \in \Pi_{i=0}^{t-1}(\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$, the following holds:*

$$|\Delta Q^n(x_{..t-1})| \leq \sum_{m \in C(n)} w^m \max_{u \in \{\widehat{x}_t, x^*_t\}} |\Delta Q^m(x_{..t-1}, u)| + \max_{u \in \{\widehat{x}_t, x^*_t\}} |\mathrm{E}^n_{\text{disc}}(x_{..t-1}, u)|, \tag{74}$$

and at the root node:

$$|\Delta Q^{n_0}| \leq \sum_{m \in C(n_0)} w^m \max_{u \in \{\widehat{x}_0, x^*_0\}} |\Delta Q^m(u)| + \max_{u \in \{\widehat{x}_0, x^*_0\}} |\mathrm{E}^{n_0}_{\text{disc}}(u)|. \tag{75}$$

Proof. We first prove the result at the root node. Take an arbitrary $x_0 \in \mathcal{X}_0$ (recall that $\mathcal{X}_0 = \widehat{\mathcal{X}}_0$). Using successively (19), (29) and (28), we can write $\Delta Q^{n_0}_{\text{sub}}(x_0)$ as

$$\begin{aligned} \Delta Q^{n_0}_{\text{sub}}(x_0) &= \widehat{Q}^{n_0}(x_0, \widehat{x}_{1..}) - Q_0(x_0, x^*_{1..}) && (76)\\ &= \sum_{m \in C(n_0)} w^m \, \widehat{Q}^m(x_0, \widehat{x}_{1..}) - Q_0(x_0, x^*_{1..}) && (77)\\ &= \sum_{m \in C(n_0)} w^m \big[\Delta Q^m(x_0) + Q_1(x_0, x^*_{1..}; \zeta^m)\big] - Q_0(x_0, x^*_{1..}) && (78)\\ &= \sum_{m \in C(n_0)} w^m \Delta Q^m(x_0) + \mathrm{E}^{n_0}_{\text{disc}}(x_0). && (79) \end{aligned}$$

Combining the above equality with the inequality (61), in the particular case for which $x_0 \in \{\widehat{x}_0, x_0^*\}$, and applying the triangle inequality yields the result at the root node.

Similarly, we show now that the result holds for any $t \in \{1, \dots, T-1\}$, $n \in \mathcal{N}_t$ and $x_{..t} \in \Pi_{i=0}^{t}(\mathcal{X}_i \cup \widehat{\mathcal{X}}_i)$. Using successively (19), (29) and (27), we can write $\Delta Q_{\mathrm{sub}}^n(x_{..t})$ as

$$\Delta Q_{\mathrm{sub}}^n(x_{..t}) = \widehat{Q}^n(x_{..t}, \widehat{x}_{t+1..}) - Q_t(x_{..t}, x_{t+1..}^*; \zeta^{..n}) \tag{80}$$

$$= \sum_{m \in C(n)} w^m \, \widehat{Q}^m(x_{..t}, \widehat{x}_{t+1..}) - Q_t(x_{..t}, x_{t+1..}^*; \zeta^{..n}) \tag{81}$$

$$= \sum_{m \in C(n)} w^m \left[\Delta Q^m(x_{..t}) + Q_{t+1}(x_{..t}, x_{t+1..}^*; \zeta^{..m})\right] - Q_t(x_{..t}, x_{t+1..}^*; \zeta^{..n}) \tag{82}$$

$$= \sum_{m \in C(n)} w^m \Delta Q^m(x_{..t}) + \mathrm{E}_{\mathrm{disc}}^n(x_{..t}). \tag{83}$$

From the triangle inequality it follows that

$$|\Delta Q_{\mathrm{sub}}^n(x_{..t})| \leq \sum_{m \in C(n)} w^m |\Delta Q^m(x_{..t})| + |\mathrm{E}_{\mathrm{disc}}^n(x_{..t})|. \tag{84}$$

In the particular case for which $x_{..t} = (x_{..t-1}, u)$, with $u \in \{\widehat{x}_t, x_t^*\}$, we combine the above inequality with the inequality (60) to obtain

$$|\Delta Q^n(x_{..t-1})| \leq \max_{u \in \{\widehat{x}_t, x_t^*\}} |\Delta Q_{\mathrm{sub}}^n(x_{..t-1}, u)| \tag{85}$$

$$\leq \sum_{m \in C(n)} w^m \max_{u \in \{\widehat{x}_t, x_t^*\}} |\Delta Q^m(x_{..t-1}, u)| + \max_{u \in \{\widehat{x}_t, x_t^*\}} |\mathrm{E}_{\mathrm{disc}}^n(x_{..t-1}, u)|. \tag{86}$$

This proves the result for any $t \in \{1, \dots, T-1\}$. □

Theorem 4.3. *The scenario-tree optimal-value error is bounded by a weighted sum of node discretization errors as follows:*

$$|\Delta Q^{n_0}| \leq \sum_{n \in \mathcal{N} \setminus \mathcal{N}_T} W^n \max_{u \in \Pi_{i=0}^{t(n)}\{\widehat{x}_i, x_i^*\}} |\mathrm{E}_{\mathrm{disc}}^n(u)|. \tag{87}$$

Proof. Take an arbitrary $t \in \{1, \dots, T-1\}$ and $n \in \mathcal{N}_t$. Using the inequality (74) in the particular case for which $x_{..t-1} \in \Pi_{i=0}^{t-1}\{\widehat{x}_i, x_i^*\}$ yields

$$\max_{v \in \Pi_{i=0}^{t-1}\{\widehat{x}_i, x_i^*\}} |\Delta Q^n(v)| \leq \sum_{m \in C(n)} w^m \max_{v \in \Pi_{i=0}^{t-1}\{\widehat{x}_i, x_i^*\}} \big(\max_{u \in \{\widehat{x}_t, x_t^*\}} |\Delta Q^m(v, u)|\big) \tag{88}$$

$$+ \max_{v \in \Pi_{i=0}^{t-1}\{\widehat{x}_i, x_i^*\}} \big(\max_{u \in \{\widehat{x}_t, x_t^*\}} |\mathrm{E}_{\mathrm{disc}}^n(v, u)|\big)$$

$$= \sum_{m \in C(n)} w^m \max_{(v,u) \in \Pi_{i=0}^{t}\{\widehat{x}_i, x_i^*\}} |\Delta Q^m(v, u)| \tag{89}$$

$$+ \max_{(v,u) \in \Pi_{i=0}^{t}\{\widehat{x}_i, x_i^*\}} |\mathrm{E}_{\mathrm{disc}}^n(v, u)|.$$

At all nodes $n \in \mathcal{N}_T$ the following inequality holds trivially (see Remark 2.1):

$$\max_{v \in \Pi_{i=0}^{T-1}\{\widehat{x}_i, x_i^*\}} |\Delta Q^n(v)| \leq 0. \tag{90}$$

Finally, by defining

$$\beta^n = \begin{cases} \max\limits_{v \in \Pi_{i=0}^{t(n)-1}\{\widehat{x}_i, x_i^*\}} |\Delta Q^n(v)| & \text{if } n \in \mathcal{N}^*; \quad (91) \\ |\Delta Q^{n_0}| & \text{if } n = n_0, \quad (92) \end{cases}$$

and

$$\gamma^n = \begin{cases} \max\limits_{w \in \Pi_{i=0}^{t(n)}\{\widehat{x}_i, x_i^*\}} |\mathrm{E}^n_{\mathrm{disc}}(w)| & \text{if } n \in \mathcal{N}^* \setminus \mathcal{N}_T; \quad (93) \\ \max\limits_{w \in \{\widehat{x}_0, x_0^*\}} |\mathrm{E}^{n_0}_{\mathrm{disc}}(w)| & \text{if } n = n_0, \quad (94) \end{cases}$$

we see that the sequence $\{\beta^n\}$ satisfies the recursive inequalities of Lemma 3.4 (b). Thus, the bound (87) follows directly from (37) applied at the root node. □

Bound in terms of worst-case integration errors

We want now to express the bound in the right-hand side of (87) as a weighted sum of worst-case integration errors in some function sets. To this end, we first introduce the concept of *integration error* at node $n \in \mathcal{N} \setminus \mathcal{N}_T$, which represents the error made by using a scenario-tree approximation to compute the conditional expectation of $f(\boldsymbol{\xi}_{t(n)+1})$ given $\boldsymbol{\xi}_{..t(n)} = \zeta^{..n}$, with f an appropriately integrable function. The node integration error generalizes the node discretization error of Definition 3.2 to the class of all integrable functions.

We denote by $\mathcal{F}_1$ the set of all functions $f : \Xi_1 \to \mathbb{R}$ that are integrable with respect to the distribution of $\boldsymbol{\xi}_1$, and by $\mathcal{F}_t(\xi_{..t-1})$ the set of all functions $f : \Xi_t(\xi_{..t-1}) \to \mathbb{R}$ that are integrable with respect to the conditional distribution of $\boldsymbol{\xi}_t$ given $\boldsymbol{\xi}_{..t-1} = \xi_{..t-1}$.

Definition 4.4 (Node integration error)**.** *For every $t \in \{0, \dots, T-1\}$, we define the* integration error operator $\mathcal{E}^n$ *at node $n \in \mathcal{N}_t$ as*

$$\mathcal{E}^n(f) = \sum_{m \in C(n)} w^m f(\zeta^m) - \mathbb{E}[f(\boldsymbol{\xi}_{t+1}) \,|\, \boldsymbol{\xi}_{..t} = \zeta^{..n}], \quad \forall f \in \mathcal{F}_{t+1}(\zeta^{..n}), \tag{95}$$

if $t \in \{1, \dots, T-1\}$, and

$$\mathcal{E}^{n_0}(f) = \sum_{m \in C(n_0)} w^m f(\zeta^m) - \mathbb{E}[f(\boldsymbol{\xi}_1)], \quad \forall f \in \mathcal{F}_1, \tag{96}$$

at the root node.

The concept of integration error naturally leads to the concept of *worst-case integration error* $\mathrm{E}^n_{\mathrm{wc}}(\mathcal{G})$ at node $n \in \mathcal{N} \setminus \mathcal{N}_T$ for a non-empty function set $\mathcal{G}$:

$$\mathrm{E}^n_{\mathrm{wc}}(\mathcal{G}) := \sup_{f \in \mathcal{G}} |\mathcal{E}^n(f)|. \tag{97}$$

The following function sets of recourse functions are of particular interest to express the bound:

$$\mathcal{Q}^{n_0} = \big\{\xi_1 \mapsto Q_1(x_0, x^*_{1..}; \xi_1) \,\big|\, x_0 \in \{x_0^*, \widehat{x}_0\}\big\}, \tag{98}$$

and for every $t \in \{1, \dots, T-1\}$ and $n \in \mathcal{N}_t$,

$$\mathcal{Q}^n = \big\{\xi_{t+1} \mapsto Q_{t+1}(x_{..t}, x^*_{t+1..}; \zeta^{..n}, \xi_{t+1}) \,\big|\, x_{..t} \in \Pi_{i=0}^t\{\widehat{x}_i, x_i^*\}\big\}. \tag{99}$$

Corollary 4.5 below expresses the bound (87) by means of worst-case integration errors.

Corollary 4.5. *Let $\mathcal{G}^n$, for every $n \in \mathcal{N} \setminus \mathcal{N}_T$, be any function sets satisfying $\mathcal{Q}^{n_0} \subseteq \mathcal{G}^{n_0} \subseteq \mathcal{F}_1$ and $\mathcal{Q}^n \subseteq \mathcal{G}^n \subseteq \mathcal{F}_{t(n)+1}(\zeta^{..n})$. The scenario-tree optimal-value error is bounded by a weighted sum of worst-case integration errors as follows:*

$$|\Delta Q^{n_0}| \leq \sum_{n \in \mathcal{N} \setminus \mathcal{N}_T} W^n \, \mathrm{E}^n_{\mathrm{wc}}(\mathcal{G}^n). \tag{100}$$

Proof. The worst-case integration error (97) and the node discretization error of Definition 3.2 are linked as follows:

$$\mathrm{E}^n_{\mathrm{wc}}(\mathcal{Q}^n) = \max_{u \in \Pi_{i=0}^{t(n)} \{\widehat{x}_i, x_i^*\}} |\mathrm{E}^n_{\mathrm{disc}}(u)|, \quad \forall n \in \mathcal{N} \setminus \mathcal{N}_T. \tag{101}$$

Thus, Theorem 4.3 directly yields the right-hand side of (100) with $\mathcal{Q}^n$ in place of $\mathcal{G}^n$. Moreover, by definition of the worst-case integration error, we have that $\mathrm{E}^n_{\mathrm{wc}}(\mathcal{G}^n) \geq \mathrm{E}^n_{\mathrm{wc}}(\mathcal{Q}^n)$ for every $n \in \mathcal{N} \setminus \mathcal{N}_T$, which completes the proof. $\square$

5 Conclusion

An important challenge in stochastic programming is the generation of efficient scenario trees for solving problems with many stages and/or many random parameters. As of today, solving a problem within a given range of error requires a scenario tree of a size that grows fast with the number of stages and random parameters. We believe that this occurs because current methods focus mostly on approximating the stochastic process, with little or no regard to the revenue function, and they often consider by default regular branching structures only. This paper aims at showing that solution methods could be greatly improved by also taking into account information on the revenue function and the constraints, and by tailoring methods to specific classes of problems. The two theorems on the optimal-value error derived in this paper pave the way to designing such methods.

The first theorem is an exact decomposition of the optimal-value error as a weighted sum of discretization errors and optimization errors made at each node of the scenario tree. This decomposition shows that an inappropriate discretization at a node where the recourse function is ill-behaved (e.g, with large variations) can contribute to most of the total optimal-value error.

The second theorem is an upper bound on the optimal-value error that features only node discretization errors. The optimal-value error can be kept small by choosing a scenario tree with a small value for the bound, hence the bound provides a figure of merit to assess the discretization quality of any scenario tree

In the corollary of the second theorem, we show that the bound can be written as a weighted sum of worst-case integration errors in certain functions sets. These sets contains the recourse functions of the problem, therefore, problems with similar properties for the recourse functions will have similar function sets in the bound. By identifying classes of problems according to these function sets, and by leveraging on Quasi-Monte Carlo theory to compute the worst-case errors, it becomes possible to design a scenario-tree generation method tailored to a specific class of problems. Within a given class, the bound provides the figure of merit to find the most suitable scenario tree, which may satisfy the tight trade-off between a high accuracy of the optimal-value estimates and a low computational times available to solve the problems.

References

[1] Bastin, F., Cirillo, C., and Toint, P. L. (2006). Convergence theory for nonconvex stochastic programming with an application to mixed logit. *Mathematical Programming*, 108(2-3):207–234.

[2] Bellman, R. (1957). *Dynamic Programming.* Princeton University Press.

[3] Bertocchi, M., Consigli, G., and Dempster, M. A. (2011). *Stochastic optimization methods in finance and energy.* Springer.

[4] Bertsekas, D. P. (2000). *Dynamic programming and optimal control*, volume 1. Athena Scientific Belmont, MA, second edition.

[5] Birge, J. R. and Louveaux, F. (2011). *Introduction to stochastic programming.* Springer Science & Business Media, second edition.

[6] Dupačová, J., Gröwe-Kuska, N., and Römisch, W. (2003). Scenario reduction in stochastic programming. *Mathematical programming*, 95(3):493–511.

[7] Dyer, M. and Stougie, L. (2006). Computational complexity of stochastic programming problems. *Mathematical Programming*, 106(3):423–432.

[8] Growe-Kuska, N., Heitsch, H., and Romisch, W. (2003). Scenario reduction and scenario tree construction for power management problems. In *Power Tech Conference Proceedings, 2003 IEEE Bologna*, volume 3, pages 7–pp. IEEE.

[9] Hanasusanto, G. A., Kuhn, D., and Wiesemann, W. (2016). A comment on "computational complexity of stochastic programming problems". *Mathematical Programming*, 159(1):557–569.

[10] Heitsch, H. and Römisch, W. (2003). Scenario reduction algorithms in stochastic programming. *Computational optimization and applications*, 24(2-3):187–206.

[11] Heitsch, H. and Römisch, W. (2009). Scenario tree modeling for multistage stochastic programs. *Mathematical Programming*, 118(2):371–406.

[12] Heitsch, H., Römisch, W., and Strugarek, C. (2006). Stability of multistage stochastic programs. *SIAM Journal on Optimization*, 17(2):511–525.

[13] Hobbs, B. F. (1995). Optimization methods for electric utility resource planning. *European Journal of Operational Research*, 83(1):1–20.

[14] Homem-de Mello, T. (2008). On rates of convergence for stochastic optimization problems under non-independent and identically distributed sampling. *SIAM Journal on Optimization*, 19(2):524–551.

[15] Høyland, K., Kaut, M., and Wallace, S. W. (2003). A heuristic for moment-matching scenario generation. *Computational Optimization and Applications*, 24(2-3):169–185.

[16] Høyland, K. and Wallace, S. W. (2001). Generating scenario trees for multistage decision problems. *Management Science*, 47(2):295–307.

[17] Koivu, M. (2005). Variance reduction in sample approximations of stochastic programs. *Mathematical programming*, 103(3):463–485.

[18] Kovacevic, R. M., Pflug, G. C., and Vespucci, M. T. (2013). *Handbook of risk management in energy production and trading.* Springer.

[19] Leövey, H. and Römisch, W. (2015). Quasi-Monte Carlo methods for linear two-stage stochastic programming problems. *Mathematical Programming*, 151(1):315–345.

[20] Mak, W.-K., Morton, D. P., and Wood, R. K. (1999). Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters*, 24(1):47–56.

[21] Olsen, P. (1976). Discretizations of multistage stochastic programming problems. In Wets, R. J.-B., editor, *Stochastic Systems: Modeling, Identification and Optimization, II*, pages 111–124. Springer.

[22] Pennanen, T. (2005). Epi-convergent discretizations of multistage stochastic programs. *Mathematics of Operations Research*, 30(1):245–256.

[23] Pennanen, T. (2009). Epi-convergent discretizations of multistage stochastic programs via integration quadratures. *Mathematical Programming*, 116(1-2):461–479.

[24] Pennanen, T. and Koivu, M. (2002). Integration quadratures in discretization of stochastic programs. *Stochastic Programming E-Print Series*, (11).

[25] Pflug, G. C. (2001). Scenario tree generation for multiperiod financial optimization by optimal discretization. *Mathematical programming*, 89(2):251–271.

[26] Pflug, G. C. and Pichler, A. (2011). Approximations for probability distributions and stochastic optimization problems. In Bertocchi, M., Consigli, G., and Dempster, M. A., editors, *Stochastic optimization methods in finance and energy*, pages 343–387. Springer.

[27] Pflug, G. C. and Pichler, A. (2012). A distance for multistage stochastic optimization models. *SIAM Journal on Optimization*, 22(1):1–23.

[28] Pflug, G. C. and Pichler, A. (2015). Dynamic generation of scenario trees. *Computational Optimization and Applications*, 62(3):641–668.

[29] Prékopa, A. (1995). *Stochastic programming*. Kluwer Academic Publishers, Dordrecht, Boston.

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

[31] Römisch, W. (2003). Stability of stochastic programming problems. In Ruszczyński, A. and Shapiro, A., editors, *Handbooks in operations research and management science*, volume 10, pages 483–554. Elsevier.

[32] Ruszczyński, A. and Shapiro, A., editors (2003). *Stochastic Programming, volume 10 of Handbooks in Operations Research and Management Science*. Elsevier.

[33] Shapiro, A. (2003). Monte Carlo sampling methods. In Ruszczyński, A. and Shapiro, A., editors, *Handbooks in operations research and management science*, volume 10, pages 353–425. Elsevier.

[34] Shapiro, A. (2006). On complexity of multistage stochastic programs. *Operations Research Letters*, 34(1):1–8.

[35] Shapiro, A. and Homem-de Mello, T. (1998). A simulation-based approach to two-stage stochastic programming with recourse. *Mathematical Programming*, 81(3):301–325.

[36] Shapiro, A. and Homem-de Mello, T. (2000). On the rate of convergence of optimal solutions of Monte Carlo approximations of stochastic programs. *SIAM journal on optimization*, 11(1):70–86.

[37] Wallace, S. W. and Ziemba, W. T. (2005). *Applications of stochastic programming*, volume 5. SIAM.