Scenario generation in stochastic programming

Werner Römisch
Humboldt-University Berlin, Institute of Mathematics,
10099 Berlin, Germany, romisch@math.hu-berlin.de

Abstract

Stability-based methods for scenario generation in stochastic programming are reviewed. In particular, we briefly discuss Monte Carlo sampling, Quasi-Monte Carlo methods, quadrature rules based on sparse grids and optimal quantization. In addition, we provide some convergence results based on recent developments in multivariate integration. The method of optimal scenario reduction and techniques for scenario trees generation are also reviewed.

Many stochastic programming models may be rewritten into the form

$$\min\Big\{\int_\Xi f_0(x,\xi)P(d\xi) : x \in X, \int_\Xi f_k(x,\xi)P(d\xi) \le 0, k = 1,...,K\Big\}, \tag{1}$$

where X is a closed subset of $\mathbb{R}^m$, Ξ a closed subset of $\mathbb{R}^d$, the functions f_k map from $\mathbb{R}^m \times \Xi$ to the extended real numbers $\overline{\mathbb{R}}$ for $k = 0,...,K$, and P is a probability distribution on Ξ. The set X is used to describe all constraints not depending on P, and the set Ξ to contain the support of P. The integrands f_k are assumed to be lower semicontinuous jointly in (x,ξ) implying that all integrals in (1) are well defined (although possibly infinite).

Classical examples are *linear two-stage stochastic programs* and *optimization models with probabilistic constraints*. Linear two-stage models (see Section 1.5.2.1) appear for $K := 0$ and f_0 having the representation

$$f_0(x,\xi) := \langle c(\xi), x\rangle + \inf\{\langle q(\xi), y\rangle : W(\xi)y = h(\xi) - T(\xi)x, y \ge 0\} \tag{2}$$

by means of the infimum of a second stage linear program where some of the coefficients are affine functions of the d-dimensional random vector ξ and the variable x is the first stage decision. Models with probabilistic constraints appear, for example, for $K = 1$, $f_0(x,\xi) = \langle c, x\rangle$ and

$$f_1(x,\xi) = p - \mathbb{1}_{\{\xi\in\Xi: T(\xi)x \ge h(\xi)\}}(\xi),$$

where $\mathbb{1}_B$ denotes the characteristic function of a set B in $\mathbb{R}^d$ and $p \in (0,1)$ is a probability level (see also Section 1.5.6.1).

1 Approximation

Stability results for (1) with respect to approximations Q of the original probability distribution P (see [30] for a survey) state that infimal values $v(P)$ and $v(Q)$ and solution sets $S(P)$ and $S(Q)$ of the stochastic programs (1) with distributions P and Q, respectively, get close if the (uniform) distance of P and Q

$$d_{\mathcal{F}}(P,Q) = \sup_{f\in\mathcal{F}} \Big| \int_{\Xi} f(\xi)P(d\xi) - \int_{\Xi} f(\xi)Q(d\xi) \Big| \tag{3}$$

with $\mathcal{F} = \{f_k(x,\cdot) : x \in X,\ k = 0,\dots,K\}$ gets small, the set X is compact, the objective function $x \mapsto \int_{\Xi} f_0(x,\xi)P(d\xi)$ is Lipschitz continuous on X and a metric regularity condition for the constraint set is satisfied. The latter two conditions are only needed if $K \geq 1$. As we have seen before, typical integrands f in stochastic programs are nondifferentiable or even discontinuous. (For simplicity we assume here that all integrals in (1) are finite for every $x \in X$ and for the probability distribution P and its approximations Q.)

The most important way to approximate P consists in utilizing discrete probability measures Q_n having finite support

$$\mathrm{supp}(Q_n) = \{\xi^1,\dots,\xi^n\} \subset \Xi$$

for some $n \in \mathbb{N}$. The elements of $\mathrm{supp}(Q_n)$ are often called *scenarios*. When replacing P by Q_n in (1) the (multivariate) integrals in (1) reduce to weighted sums and one obtains

$$\min\Big\{ \sum_{i=1}^n q_i f_0(x,\xi^i) : x \in X, \sum_{i=1}^n q_i f_k(x,\xi^i) \leq 0, k = 1,...,K \Big\}, \tag{4}$$

where $q_i = Q_n(\{\xi^i\}) > 0$ is the probability that scenario ξ^i occurs ($i = 1,\dots,n$). Clearly, we have $\sum_{i=1}^n q_i = 1$. The structure of the approximate stochastic programming problem (4) is very close to a standard (linear, nonlinear, integer) optimization model. The only remaining difficulty consists in the need of many evaluations of the functions f_k at the pairs (x,ξ^i), $i = 1,\dots,n$, if the number n of scenarios gets large. But, large n are often unavoidable when recalling numerical integration even in dimension $d = 1$ and all the more for large d as in many applied stochastic programming models, in production, energy, transportation and finance. According to our stability considerations the number n, the scenarios $\xi^i \in \Xi$ and their probabilities q_i for $i = 1,\dots,n$ should be selected such that for given $\varepsilon > 0$ the (absolute) error satisfies

$$e(Q_n) := \sup_{f\in\mathcal{F}} \Big| \int_{\Xi} f(\xi)P(d\xi) - \sum_{i=1}^n q_i f(\xi^i) \Big| \leq \varepsilon \tag{5}$$

with $\mathcal{F}$ defined earlier. Another and more feasible condition consists in utilizing the relative error such that given $\varepsilon \in [0,1)$ we look for a probability measure Q_n such that

$$e(Q_n) \leq \varepsilon e(\bar{Q}_1), \tag{6}$$

where the measure $\bar{Q}_1$ consists of only one distinguished scenario $\bar{\xi}$ with probability one. A more advanced requirement consists in looking for Q_n with the smallest number $n = n_{\min}(\varepsilon, Q_n) \in \mathbb{N}$ of scenarios such that (6) holds. Sometimes even $Q_0 = 0$ is considered in the literature (e.g. [34]) and, thus, the criterion (6) is of the form

$$e(Q_n) \leq \varepsilon \sup_{f \in \mathcal{F}} \Big| \int_{\Xi} f(\xi) P(d\xi) \Big|. \tag{7}$$

The behavior of $e(Q_n)$ with respect to $n \in \mathbb{N}$ and of $n_{\min}(\varepsilon, Q_n)$ with respect to ε is of considerable interest. In both cases the dependence on the dimension d of P is crucial, too.

It is not surprising that the behavior of the two quantities depends heavily on the set $\mathcal{F}$ of integrands as well as on the probability distribution P. Since the set $\mathcal{F}$ in its present form is not very convenient to handle, it might be an alternative to enlarge $\mathcal{F}$ in (5), (6) or (7). But, one has to be careful in this process as is shown next.

If $\mathcal{F}$ is the unit ball in the Banach space $\mathrm{Lip}(\mathbb{R}^d)$ of Lipschitz continuous functions on $\Xi = \mathbb{R}^d$ and if $q_i = \frac{1}{n}$, $i = 1, \ldots, n$, the best possible convergence rate is

$$e(Q_n) = C n^{-\frac{1}{d}} \tag{8}$$

for some constant C depending only on P under the weak assumptions that P is not singular with respect to the Lebesgue measure on $\mathbb{R}^d$ and that the moment $\int_{\mathbb{R}^d} \|\xi\|^{1+\delta} P(d\xi)$ is finite for some $\delta > 0$ (cf. [7, Theorem 6.2]).

The rate (8) suggests that the unit ball in $\mathrm{Lip}(\mathbb{R}^d)$ is too large and one should look for function classes $\mathcal{F}$ satisfying more restrictive conditions. One possibility is offered by the classical Koksma-Hlawka inequality [21, 6]. It states for an integrand f on $\Xi = [0,1]^d$ that is of bounded variation $V(f)$ in the sense of Hardy and Krause that the estimate

$$\Big| \int_{\Xi} f(\xi) d\xi - \sum_{i=1}^{n} q_i f(\xi^i) \Big| \leq V(f) \alpha^*(\lambda^d, Q_n), \tag{9}$$

holds with $q_i = \frac{1}{n}$, $i = 1, \ldots, n$ and with the so-called star-discrepancy α^* defined by

$$\alpha^*(P, Q) = \sup_{\xi \in [0,1]^d} |P([0,\xi)) - Q([0,\xi))|,$$

where $[0,\xi) = \times_{i=1}^{d}[0,\xi_i)$. For problem (1) with $K = 0$ and $\Xi = [0,1]^d$ the Koksma-Hlawka inequality (9) leads to the estimate

$$e(Q_n) \le \sup_{f\in\mathcal{F}} V(f)\,\alpha^*(P,Q_n) = \sup_{x\in X} V(f_0(x,\cdot))\,\alpha^*(P,Q_n)$$

If $P = \lambda^d$ is the uniform distribution on $\Xi = [0,1]^d$ and $q_i = \frac{1}{n}$, $i = 1,\ldots,n$, it is known (cf. [21, Chapter 3.1]) that there exist sequences $(\xi^i)_{i\in\mathbb{N}}$ with

$$\alpha^*(P,Q_n) = O(n^{-1}(\log n)^d). \tag{10}$$

Hence, compared with (8), a much better convergence rate for $e(Q_n)$ can be obtained under stronger assumptions on $\mathcal{F}$. Initiated by the pioneering work [34] the convergence rate is further improved if the set $\mathcal{F}$ is bounded in the weighted tensor product space

$$W_2^{(1,\ldots,1)}([0,1]^d) = \bigotimes_{i=1}^{d} W_2^1([0,1]), \tag{11}$$

where $W_2^1([0,1])$ is the Sobolev space of absolutely continuous real functions whose first derivatives belong to $L_2([0,1])$. The space (11) contains all real functions f on $[0,1]^d$, for which all mixed partial derivatives $\frac{\partial^{|u|}}{\partial\xi_u}f(\xi_u,1)$ exist for almost every $\xi_u \in [0,1]^{|u|}$ and $u \subseteq D = \{1,\ldots,d\}$ and for which the weighted norm

$$\|f\|_{d,\gamma} = \Big(\sum_{u\subseteq D}\prod_{j\in u}\gamma_j^{-1}\int_{[0,1]^{|u|}}\Big|\frac{\partial^{|u|}}{\partial\xi_u}f(\xi_u,1)\Big|^2 d\xi_u\Big)^{\frac{1}{2}}$$

is finite. Here, we denote by $|u|$ the cardinality of $u \subseteq D$, and by $\xi_u \in [0,1]^{|u|}$ the vector containing the components of $\xi \in [0,1]^d$ whose indices are in u. The weights (γ_j) are positive and monotonically decreasing with $\gamma_1 = 1 = \prod_{j\in\emptyset}\gamma_j^{-1}$.

By utilizing the weighted Koksma-Hlawka inequality

$$\Big|\int_{\Xi} f(\xi)d\xi - \frac{1}{n}\sum_{i=1}^{n} f(\xi^i)\Big| \le \mathrm{disc}_\gamma((\xi^i))\|f\|_{d,\gamma}$$

with $\mathrm{disc}(\xi) = \prod_{j=1}^{d}\xi_j - n^{-1}\sum_{\xi^i\in[0,\xi)} i$ and the weighted L_2-star-discrepancy

$$\mathrm{disc}_\gamma((\xi^i)) = \Big(\sum_{\emptyset\neq u\subseteq D}\prod_{j\in u}\gamma_j\int_{[0,1]^{|u|}}\mathrm{disc}^2(\xi_u,1)d\xi_u\Big)^{\frac{1}{2}}$$

instead of $\alpha^*(\lambda^d,Q_n)$ in (9), it became possible in [34, 14] to prove the existence of a sequence $(\xi^i)_{i\in\mathbb{N}}$ such that for any $\delta > 0$ the estimate

$$e(Q_n) \le C(\delta)n^{-1+\delta} \tag{12}$$

holds, where $C(\delta)$ is independent of n and d if the weights (γ_j) satisfy the condition $\sup_d \sum_{j=1}^d \gamma_j < \infty$. While the results in [34, 14] were nonconstructive, it is reported in [19] that certain shifted lattice rules attain the optimal order (12) of convergence.

2 Scenario generation techniques

We briefly discuss here four different scenario generation techniques for stochastic programs without nonanticipativity constraints:

(1) Monte Carlo simulation methods,

(2) Quasi-Monte Carlo methods,

(3) Quadrature rules using sparse grids,

(4) Optimal quantization (or discretization) of probability measures.

2.1 Monte Carlo simulation methods

Monte Carlo methods are based on drawing independent identically distributed (iid) Ξ-valued random samples $\xi^1(\cdot), \ldots, \xi^n(\cdot), \ldots$ (defined on some probability space $(\Omega, \mathcal{A}, \mathbb{P})$) from an underlying probability distribution P (on Ξ) and on using the law of large numbers to obtain

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} f(\xi^i(\omega)) = \int_{\Xi} f(\xi) P(d\xi) \qquad \mathbb{P}\text{-almost surely}$$

for every real continuous and bounded function f on Ξ. Practically, iid samples are approximately obtained by pseudo random number generators as uniform samples in $[0,1]^d$ and later transformed to more general sets and distributions. This technique may be applied, for example, to certain time series models (calibrated to the available statistical data) or to the statistical data directly. For details we refer to Section 1.5.4.1.

2.2 Quasi-Monte Carlo methods

The basic idea of Quasi-Monte Carlo (QMC) methods is to replace random samples in Monte Carlo methods by deterministic points that are *uniformly distributed* in $[0,1]^d$. One possibility for defining the latter property for a sequence $(\xi^i)_{i\in\mathbb{N}}$ is to require that

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} f(\xi^i) = \int_{[0,1]^d} f(\xi) d(\xi) \tag{13}$$

holds for all real continuous functions f on $[0,1]^d$. Basic references for QMC methods are [21, 20]. With $P = \lambda^d$ (the d-dimensional Lebesgue measure) and $Q = Q_n$ denoting the probability measure with support $\mathrm{supp}\,(Q_n) = \{\xi^1, \ldots, \xi^n\}$ and with identical probabilities $\frac{1}{n}$ for all atoms, the estimate (9) highlights the importance of the star-discrepancy of the point set $\{\xi^1, \ldots, \xi^n\}$

$$\mathrm{disc}^*(\xi^1, \ldots, \xi^n) := \alpha^*(\lambda^d, Q_n)$$

in this respect. Hence, one should look for *low discrepancy sequences*, i.e., sequences $(\xi_i)_{i\in\mathbb{N}}$ such that $\mathrm{disc}^*(\xi^1, \ldots, \xi^n)$ is low for all n. Such sequences have the property

$$\frac{1}{2}n^{-1} \leq \mathrm{disc}^*(\xi^1, \ldots, \xi^n) = O(n^{-1}(\log n)^d).$$

In particular, we refer to the sequences discussed in [20, Sect. 5.4] and [26], namely, *Faure*, *Sobol* and *Niederreiter sequences*. The latter are special cases of so-called (t, d)-sequences which in turn are based on (t, m, d)-nets [21, Chapter 4]. The latter (t, m, d)-nets and lattices [21, Chapter 5], [20, Sect. 5.3] represent (finite) *low discrepancy point sets* and are, presently, the most important sources for QMC methods.

If $G : [0, 1]^d \to \Xi$, $\Xi \subseteq \mathbb{R}^d$, is almost everywhere continuous, then (13) means that the sequence (Q_n) converges weakly to λ^d and the continuous mapping theorem [1, Chapter 5] implies that the sequence $(Q_n G^{-1})$ converges weakly to $\lambda^d G^{-1}$. This fact allows to apply QMC methods to many situations in stochastic programming.

2.3 Quadrature rules using sparse grids

Again we consider first the unit cube $[0, 1]^d$ in $\mathbb{R}^d$. Let nested sets of grids in $[0, 1]$ be given

$$\Xi^i = \{\xi^i_1, \ldots, \xi^i_{m_i}\} \subset \Xi^{i+1} \subset [0, 1] \quad (i \in \mathbb{N}),$$

for example, the dyadic grid $\Xi^i = \{\frac{j}{2^i} : j = 0, 1, \ldots, 2^i\}$. Then the point set suggested by Smolyak [35]

$$H(q, d) := \bigcup_{\sum_{j=1}^d i_j = q} \Xi^{i_1} \times \cdots \times \Xi^{i_d} \tag{14}$$

is called a *sparse grid* in $[0, 1]^d$ and points in $H(q, d)$ are also called *hyperbolic cross points*. In case of dyadic grids in $[0, 1]$ $H(q, d)$ consists of all d-dimensional dyadic grids with product of mesh size given by $\frac{1}{2^q}$ (see also [22]).

The corresponding tensor product quadrature rules for $q \geq d$ on $[0, 1]^d$ with respect to the Lebesgue measure λ^d are of the form

$$I(q, d) = \sum_{q-d+1\leq|\mathbf{i}|\leq q} (-1)^{q-|\mathbf{i}|} \begin{pmatrix} d-1 \\ q-|\mathbf{i}| \end{pmatrix} \sum_{j_1=1}^{m_{i_1}} \cdots \sum_{j_d=1}^{m_{i_d}} f(\xi^{i_1}_{j_1}, \ldots, \xi^{i_d}_{j_d}) \prod_{l=1}^{d} a^{i_l}_{j_l}, \tag{15}$$

where $|\mathbf{i}| = \sum_{j=1}^d i_j$ and the coefficients a^i_j ($j = 1, \ldots, m_i$, $i = 1, \ldots, d$) are weights of one-dimensional quadrature rules. Even if the one-dimensional weights are positive, some of the weights in (15) are negative. Hence, an interpretation as scenario-based (discrete) probability measure is no longer possible. We note that the results in [23] (as extension of [34], see Sect. 1) apply to such tensor product quadrature rules.

2.4 Optimal quantization of probability measures

Let D be a metric distance of probability measures on $\mathbb{R}^d$, e.g., the Fortet-Mourier metric ζ_r of order r (to be used in the next section), the minimal L_r-metric ℓ_r used in [7] or some other metric (see [29]) such that the underlying stochastic program behaves stable with respect to D. Furthermore, let P be a given probability distribution on $\mathbb{R}^d$. We are looking for a discrete probability measure Q_n with support $\text{supp}(Q_n) = \{\xi^1, \ldots, \xi^n\}$, $Q_n(\{\xi^i\}) = p_i$, $i = 1, \ldots, n$, such that it is the best approximation to P in the sense

$$D(P, Q_n) = \min\{D(P, Q) : |\text{supp}(Q)| = n, Q(\mathbb{R}^d) = 1\}. \tag{16}$$

In many cases, the distance $D(P, Q)$ may be reformulated as a real function Φ defined on $[0,1]^n \times \mathbb{R}^{dn}$. It attains its global minimum subject to the standard simplex constraint for the first n variables at $(p_1, \ldots, p_n; \xi^1, \ldots, \xi^n)$. Unfortunately, the function Φ is nonconvex and often nondifferentiable, hence, minimizing it is not an easy task.

We refer to [27, 15] and [24] for developing algorithmic procedures for minimizing Φ globally, e.g. stochastic gradient algorithms, stochastic approximation methods and stochastic branch-and-bound techniques.

The methodology of optimal quantization may be extended to multi-stage stochastic programs by incorporating constraints describing the tree structure (see e.g. [15, 27]).

3 Scenario reduction

Let P be a probability measure on $\mathbb{R}^d$ having N scenarios ξ^i with probabilities p_i, $i = 1, \ldots, N$. We consider P as approximation of the original probability distribution of a stochastic program that has to be solved computationally. Due to running time requirements N might be too large and we have to look for an approximation Q_n of P whose support consists of only $n < N$ scenarios out of $\{\xi^1, \ldots, \xi^N\}$. Then two questions arise: (i) Which of the N scenarios should be deleted and (ii) which probabilities should be assigned to the n remaining scenarios ?

In the following we review the stability-based approach for (optimal) scenario reduction developed in the papers [4, 8, 9]. It is based on considering the distance $d_{\mathcal{F}}(P, Q_n)$ (see (3) in Section 1) and in determining Q_n as best approximation of P with respect to $d_{\mathcal{F}}$ among all probability measures whose supports consist of n scenarios out of $\{\xi^1, \ldots, \xi^N\}$. An equivalent formulation of this best approximation problem is as follows:
Let Q_J denote a probability measure on $\mathbb{R}^d$ with $\text{supp}(Q_J) = \{\xi^i : i \in I \setminus J\}$ for some

index set $J \subset I := \{1, \dots, N\}$ and let q_i be the probability of the scenario indexed by i for every $i \in I \setminus J$. Then the minimization problem

$$\min \Big\{ d_{\mathcal{F}}(P, Q_J) : J \subset I, |J| = N - n, q_i \geq 0, i \in I \setminus J, \sum_{i \in I \setminus J} q_i = 1 \Big\} \tag{17}$$

determines some index set J_n and weights $\bar{q}_i \in [0, 1]$ such that the probability measure $Q_n := Q_{J_n}$ with scenarios ξ^i and probabilities $\bar{q}_i$ for $i \in I \setminus J_n$ solves the best approximation problem. The formulation (17) of optimal scenario reduction leads immediately to a decomposition into an *inner* and an *outer* minimization problem, namely,

$$\min_J \Big\{ \inf_q \Big\{ d_{\mathcal{F}}(P, Q_J) : q_i \geq 0, i \in I \setminus J, \sum_{i \in I \setminus J} q_i = 1 \Big\} : J \subset I, |J| = N - n \Big\}. \tag{18}$$

In particular, the approach becomes powerful if solutions and the infimum $D_J(\mathcal{F}, P)$ of the inner problem may be determined explicitly for any index set $\emptyset \neq J \subset I$. In that case, the *optimal redistribution* of the N probabilities p_i, $i \in I$, to the n scenarios indexed by $I \setminus J$ is known and it remains to solve the outer (combinatorial) optimization problem

$$\min\{D_J(\mathcal{F}, P) : J \subset I, |J| = N - n\} \tag{19}$$

at least approximately. Problem (19) is known as n-median problem and as $\mathcal{NP}$-hard.

Unfortunately, for many function classes $\mathcal{F}$ it is impossible to determine $D_J(\mathcal{F}, P)$ as well as the optimal redistribution explicitly. In particular, this is true for models with probabilistic constraints and for two-stage mixed-integer stochastic programs in which case (rectangular, polyhedral) discrepancies appear as distances $d_{\mathcal{F}}$ (cf. [12, 13]). For discrepancies, however, the inner problem may be solved by linear programming [12]. For two-stage linear stochastic programs the integrands $f_0(x, \cdot\,)$ in (2) are often proportional to certain elements of

$$\mathcal{F}_r(\Xi) = \{f : \Xi \to \mathbb{R} : |f(\xi) - f(\tilde{\xi})| \leq c_r(\xi, \tilde{\xi}), \forall \xi, \tilde{\xi} \in \Xi\}$$

for some $r \in \mathbb{N}$ and with the (cost) function c_r defined by

$$c_r(\xi, \tilde{\xi}) := \max\{1, \|\xi\|^{r-1}, \|\tilde{\xi}\|^{r-1}\}\|\xi - \tilde{\xi}\| \quad (\xi, \tilde{\xi} \in \Xi)$$

(see [30, 31]). The corresponding distance $\zeta_r := d_{\mathcal{F}_r}$ is known as Fortet-Mourier metric of order r. In the latter case the inner problem is explicitly solvable and it holds

$$D_J(\mathcal{F}_r, P) = \sum_{j \in J} p_j \min_{i \notin J} \hat{c}_r(\xi^i, \xi^j) \tag{20}$$

$$\bar{q}_i = p_i + \sum_{\substack{j \in J \\ i(j) = i}} p_j \quad \text{and} \quad i(j) \in \arg\min_{i \notin J} \hat{c}_r(\xi^i, \xi^j),\ i \in I \setminus J, \tag{21}$$

i.e., the redistribution rule consists in adding the probability of a deleted scenario indexed by $j \in J$ to the probability of a remaining scenario that is nearest to ξ^j with respect to the distance $\hat{c}_r$ on supp (P). The so-called reduced cost $\hat{c}_r$ has the representation

$$\hat{c}_r(\xi^i, \xi^j) := \min\Big\{\sum_{k=1}^{\ell-1} c_r(\xi^{i_k}, \xi^{i_{k+1}}) : \ell \in \mathbb{N}, i_k \in I, k = 1, \ldots, \ell, i_1 = i, i_\ell = j\Big\}.$$

Two simple heuristic algorithms for solving (19) are proposed in [8, 9]: the *forward (selection)* and the *backward (reduction) heuristic*. To give an idea of the heuristics we give a short description of the forward algorithm. Its basic idea originates from the simple structure of (19) with (20) for the special case $n = 1$. It is of the form

$$\min_{u \in \{1,\ldots,N\}} \sum_{\substack{j=1 \\ j \neq u}}^{N} p_j \hat{c}_r(\xi^u, \xi^j).$$

If the minimum is attained at u^*, the index set $J = \{1, \ldots, N\} \setminus \{u^*\}$ solves (19) for

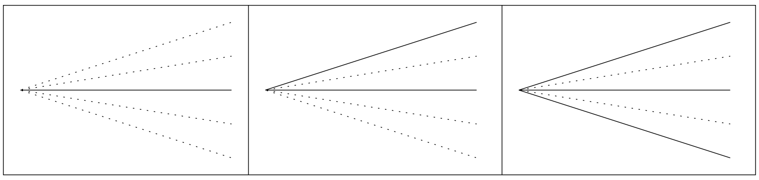

Figure 1: Illustration of selecting the first, second and third scenario out of $N = 5$

$n = 1$. The scenario ξ^{u^*} is taken as the first element of supp(Q). Then the separable structure of D_J is exploited to determine the second element of supp(Q) while the first element is fixed. The process is continued until n elements of supp(Q) are selected.

Forward algorithm for scenario reduction

Step 0: $J^{[0]} := \{1, \ldots, N\}$.

Step k: $u_k \in \arg \min_{u \in J^{[k-1]}} \sum_{j \in J^{[k-1]} \setminus \{u\}} p_j \min_{i \notin J^{[k-1]} \setminus \{u\}} \hat{c}_r(\xi^i, \xi^j)$,

$J^{[k]} := J^{[k-1]} \setminus \{u_k\}$.

Step n+1: Redistribution with $J := J^{[n]}$ via (21).

Similarly, the idea of the backward algorithm is based on the second special case of (19) with (20) for $n = N - 1$.

4 Scenario trees for multistage stochastic programs

The special feature of multistage stochastic programs (see Section 1.5.3.1) consists in imposing information constraints on the decisions. The information flow is modeled by a filtration of σ-fields $\mathcal{A}_t$, $t = 1, \ldots, T$, which is associated to the stochastic input process $\xi = (\xi_t)_{t=1}^T$ defined on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$. Typically, it is required that the σ-field $\mathcal{A}_t$ is generated by the random vector $(\xi_1, \ldots, \xi_t)$. Then the information or *nonanticipativity* constraint means measurability of the decisions x_t with respect to $\mathcal{A}_t$ for every $t = 1, \ldots, T$. Mostly, $t = 1$ refers to the present. Thus, ξ_1 is deterministic and $\mathcal{A}_1 = \{\emptyset, \Omega\}$.

Clearly, any scenario-based approximation of the underlying probability distribution P of ξ has to reflect the growth of the σ-fields. Hence, the scenarios need to be *tree-structured*. In general, there are two ways to generate scenario trees, namely, (i) a tree-structure is prescribed and scenarios are generated via conditional distributions for increasing t starting with a root at $t = 1$, or (ii) in a first step a number of scenarios is generated for the whole horizon $t = 1, \ldots, T$ based on the distribution P and according to some method reported in Sect. 2. Secondly, a tree structure is generated successively by bundling scenarios.

Several specific techniques for generating scenario trees are known from the literature. We refer to the survey [3] and the more recent papers [2], [10, 11], [17, 18], [16], [27, 15, 28], [24], [25], [33]. Most of them are discussed in the introduction of [10].

Finally, we review an application of scenario reduction techniques (see Sect. 3) to generate scenario tree. We consider an original (non-structured) scenario set $\{\xi^1, \ldots, \xi^N\}$ satisfying the root condition $\xi_1^i = \xi_1^*$, $i = 1, \ldots, N$, and the parameter sets $\{1, \ldots, t\}$ for increasing t and start at $t = 2$ with the reduction of the original scenario set restricted to $t = 2$. This leads to (say) k_2 remaining scenarios and certain clusters C_2^k of scenarios for $k = 2, \ldots, k_2$ that are associated to one of the remaining scenarios via the next neighbor property. In general, we obtain (disjoint) partitions or clusters

$$\mathcal{C}_t := \{C_t^1, \ldots, C_t^{k_t}\}, \qquad (k_t \in \mathbb{N})$$

of the index set $I = \{1, \ldots, N\}$ for every $t = 2, \ldots, T$. The following *forward algorithm* leads to a scenario tree process ξ_{tr} whose structure may be controlled by certain tolerances ε_t, $t = 2, \ldots, T$.

Algorithm: *Forward scenario tree generation*

Step 1: Initialization
Set $C_1 = \{I\}$, $k_1 = 1$ and $t := 2$.

Step 2: Cluster computation
Let $\mathcal{C}_{t-1} = \{C_{t-1}^1, \ldots, C_{t-1}^{k_{t-1}}\}$. Perform scenario reduction w.r.t. the Fortet-Mourier metric ζ_r for every scenario subset $\{\xi_t^i\}_{i \in C_{t-1}^k}$ separately for every $k \in \{1, \ldots, k_{t-1}\}$ and only with respect to the tth component. Define index sets J_t^k and I_t^k of deleted and remaining

scenarios and mappings $i_t^k : J_t^k \to I_t^k$ such that

$$i_t^k(j) \in \arg\min_{i \in I_t^k} \hat{c}_r(\xi_t^i, \xi_t^j)\,, \quad (j \in J_t^k),$$

according to the rule (21). Define the partition $\mathcal{C}_t$ and the mapping $\alpha_t : I \to I$ by

$$\mathcal{C}_t := \left\{ \alpha_t^{-1}(i) \,\middle|\, i \in I_t^k,\, k = 1, \ldots, k_{t-1} \right\} \quad \text{and} \quad \alpha_t(j) = \begin{cases} i_t^k(j), & j \in J_t^k\,, \\ j, & \text{else.} \end{cases} \tag{22}$$

$\mathcal{C}_t$ is a refinement of the partition $\mathcal{C}_{t-1}$. If $t < T$ perform the cluster computation for $t = t + 1$ and goto Step 2.

Step 3: Tree generation
According to the partition $\mathcal{C}_T$ and the mapping α_T (see (22)) the scenario tree process ξ_{tr} is defined such that its kth scenario is

$$\xi_{\text{tr}}^k = \left(\xi_1^*, \xi_2^{\alpha_2(i)}, \ldots, \xi_t^{\alpha_t(i)}, \ldots, \xi_T^{\alpha_T(i)}\right) \quad \text{for some} \quad i \in C_T^k$$

with probability $q_k := \sum_{i \in C_T^k} p_i$ for every $k = 1, \ldots, k_T$. The error at step t is given by

$$\text{err}_t := \sum_{k=1}^{k_{t-1}} \sum_{j \in J_t^k} p_j \min_{i \in I_t^k} \hat{c}_r(\xi_t^i, \xi_t^j).$$

Hence, the generation procedure may be controlled by the condition $\text{err}_t \leq \varepsilon_t$ for every $t = 2, \ldots, T$. We refer to [10] for further information and some computational experience.

Acknowledgement: The work of the author is supported by the DFG Research Center MATHEON "Mathematics for key technologies" in Berlin.

References

[1] Billingsley, P.: *Convergence of Probability Measures*, Wiley, New York, 1968.

[2] Casey, M.; Sen, S.: The scenario generation algorithm for multistage stochastic linear programming, *Mathematics of Operations Research* 30 (2005), 615–631.

[3] Dupačová, J.; Consigli, G.; Wallace, S. W.: Scenarios for multistage stochastic programs, *Annals of Operations Research* 100 (2000), 25–53.

[4] Dupačová, J.; Gröwe-Kuska, N.; Römisch, W.: Scenario reduction in stochastic programming: An approach using probability metrics, *Mathematical Programming* 95 (2003), 493–511.

[5] Frauendorfer, K.: Barycentric scenario trees in convex multistage stochastic programming, *Mathematical Programming* Ser. B, 75 (1996), 277–293.

[6] Götz, M.: Discrepancy and the error in integration, *Monatshefte für Mathematik* 136 (2002), 99–121.

[7] Graf, S.; Luschgy, H.: *Foundations of Quantization for Probability Distributions*, Lecture Notes in Mathematics, Vol. 1730, Springer, Berlin, 2000.

[8] Heitsch, H.; Römisch, W.: Scenario reduction algorithms in stochastic programming, *Computational Optimization and Applications* 24 (2003), 187–206.

[9] Heitsch, H.; Römisch, W.: A note on scenario reduction for two-stage stochastic programs, *Operations Research Letters* 35 (2007), 731–738.

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

[11] Heitsch, H.; Römisch, W.: Scenario tree reduction for multistage stochastic programs, *Computational Management Science* 6 (2009), 117–133.

[12] Henrion, R.; Küchler, C.; Römisch, W.: Discrepancy distances and scenario reduction in two-stage stochastic mixed-integer programming, *Journal of Industrial and Management Optimization* 4 (2008), 363–384.

[13] Henrion, R.; Küchler, C.; Römisch, W.: Scenario reduction in stochastic programming with respect to discrepancy distances, *Computational Optimization and Applications* 43 (2009), 67–93.

[14] Hickernell, F. J.; Woźniakowski, H.: Integration and approximation in arbitrary dimensions, *Adv. Comput. Math.* 12 (2000), 25–58.

[15] Hochreiter, R.; Pflug, G. Ch.: Financial scenario generation for stochastic multi-stage decision processes as facility location problem, *Annals of Operations Research* 152 (2007), 257–272.

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

[17] Kuhn, D.: *Generalized bounds for convex multistage stochastic programs*, Lecture Notes in Economics and Mathematical Systems, Vol. 548, Springer, Berlin, 2005.

[18] Kuhn, D.: Aggregation and discretization in multistage stochastic programming, *Mathematical Programming* 113 (2008), 61–94.

[19] Kuo, F. Y.; Sloan, I. H.: Lifting the curse of dimensionality, *Notices of the AMS* 52 (2005), 1320–1328.

[20] Lemieux, C.: Monte Carlo and Quasi-Monte Carlo Sampling, Springer Series in Statistics, Springer, 2009.

[21] Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, Philadelphia, 1992.

[22] Novak, E.; Ritter, K.: High dimensional integration of smooth functions over cubes, *Numerische Mathematik* 75 (1996), 79–97.

[23] Novak, E.; Woźniakowski, H.: Intractability results for integration and discrepancy, *Journal of Complexity* 17 (2001), 388-441.

[24] Pagès, G.; Pham, H.; Printems, J.: Optimal quantization methods and applications to numerical problems in finance, in *Handbook of Computational and Numerical Methods in Finance* (S.T. Rachev Ed.), Birkhäuser, Boston 2004, 253–297.

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

[26] Pennanen, T.; Koivu, M.: Epi-convergent discretizations of stochastic programs via integration quadratures, *Numerische Mathematik* 100 (2005), 141–163.

[27] Pflug, G. Ch.: Scenario tree generation for multiperiod financial optimization by optimal discretization, *Mathematical Programming* 89 (2001), 251–271.

[28] Pflug, G. Ch.; Mirkov, R.: Tree approximations of dynamic stochastic programs, *SIAM Journal on Optimization* 18 (2007), 1082–1105.

[29] Rachev, S. T.; Rüschendorf, L.: *Mass Transportation Problems* Vol. I: Theory, Springer, New York, 1998.

[30] Römisch, W.: Stability of stochastic programming problems, in: *Stochastic Programming* (Ruszczyński, A., Shapiro, A. eds.), Handbooks in Operations Research and Management Science, Volume 10, Elsevier, Amsterdam 2003, 483–554.

[31] Römisch, W.; Wets, R. J-B: Stability of ε-approximate solutions to convex stochastic programs, *SIAM J. Optim.* 18 (2007), 961–979.

[32] Ruszczyński, A.; Shapiro, A. (Eds.): *Stochastic Programming*, Handbooks in Operations Research and Management Science, Volume 10, Elsevier, Amsterdam 2003.

[33] Shapiro, A.: Inference of statistical bounds for multistage stochastic programming problems, *Mathematical Methods of Operations Research* 58 (2003), 57–68.

[34] Sloan, I. H.; Woźniakowski, H.: When are Quasi Monte Carlo algorithms efficient for high-dimensional integration, *Journal of Complexity* 14 (1998), 1–33.

[35] Smolyak, S. A.: Quadrature and interpolation formulas for tensor products of certain classes of functions, *Soviet Math Dokl.* 4 (1963), 240–243.