Ann Oper Res
DOI 10.1007/s10479-006-0140-6

# Financial scenario generation for stochastic multi-stage decision processes as facility location problems

**Ronald Hochreiter · Georg Ch. Pflug**



**Abstract** The quality of multi-stage stochastic optimization models as they appear in asset liability management, energy planning, transportation, supply chain management, and other applications depends heavily on the quality of the underlying scenario model, describing the uncertain processes influencing the profit/cost function, such as asset prices and liabilities, the energy demand process, demand for transportation, and the like. A common approach to generate scenarios is based on estimating an unknown distribution and matching its moments with moments of a discrete scenario model. This paper demonstrates that the problem of finding valuable scenario approximations can be viewed as the problem of optimally approximating a given distribution with some distance function. We show that for Lipschitz continuous cost/profit functions it is best to employ the Wasserstein distance. The resulting optimization problem can be viewed as a multi-dimensional facility location problem, for which at least good heuristic algorithms exist. For multi-stage problems, a scenario tree is constructed as a nested facility location problem. Numerical convergence results for financial mean-risk portfolio selection conclude the paper.

**Keywords** Stochastic programming · Multi-stage financial scenario generation

## 1 Introduction

A large class of decision problems involves decision stages and uncertainty. Examples are multi-stage portfolio optimization or asset liability management problems, energy production models, as well as models in telecommunication, transportation, supply chain management. For a recent overview see Ruszczynski and Shapiro (2003) and Wallace and Ziemba (2005). A common feature of these models is the fact that a stochastic process describing the uncertain

R. Hochreiter (✉) · G. Ch. Pflug
Department of Statistics and Decision Support Systems, University of Vienna, Universitätsstraße 5/9, 1010 Vienna, Austria
e-mail: ronald.hochreiter@univie.ac.at

G. Ch. Pflug
e-mail: georg.pflug@univie.ac.at

environment (asset prices, insurance claims, energy demand, communication load, ...) is the most important part of the input data. Typically, these stochastic processes are estimated from historical data and calibrated using some prior information. For subsequent decision models, one needs a numerically tractable approximation, which is small enough to allow for reasonable calculation times, but is large enough to capture the important features of the problem.

The goal in modeling relevant stochastic processes by scenario trees is the following: assume that a discrete-time continuous (or highly-dimensional discrete) space stochastic process $(\xi_t)_{t=0,1,2,\ldots,T}$ is given, where $\xi_0 = x_0$ represents today's value and is constant. The distribution of this process may be the result of a parametric or non-parametric estimation based on historical data. The state space may be univariate ($\mathbb{R}^1$) or multivariate ($\mathbb{R}^k$). We look for a simple approximated stochastic process $(\tilde{\xi}_t)$, which takes only finitely many values and which is as close as possible to the original process $(\xi_t)$, and has a predetermined structure as a tree at the same time. Denote the finite state space of $\tilde{\xi}_t$ by $S_t$, i.e. $\mathbb{P}\{\tilde{\xi}_t \in S_t\} = 1$. Let $c(t)$ be the cardinality of $S_t$. We have that $c(0) = 1$. If $x \in S_t$, let $b(x, t)$ be the branching factor of $x$, i.e. $b(x, t) = \#\{y : \mathbb{P}\{\tilde{\xi}_{t+1} = y | \tilde{\xi}_t = x\} > 0\}$.

The process $(\tilde{\xi}_t)_{t=0,\ldots,T}$ can be represented as a tree, where the root is $(x_0, 0)$ and the node $(x, t)$ and $(y, t+1)$ are connected by an arc, if $\mathbb{P}\{\tilde{\xi}_t = x, \tilde{\xi}_{t+1} = y\} > 0$. The collection of all branching factors $b(x, t)$ determines the size of the tree. We may choose the branching factors beforehand and independent of $x$. In this case, the structure of the tree is determined by the vector $B = (b_0, b_1, \ldots, b_{T-1})$, where $b_t$ denotes the number of successors per node in stage $t$. Let $N = (n_1, n_2, \ldots, n_T)$ be the vector of the total number of nodes in each stage ($n_0 = 1$). The approximation problem is an optimization problem of one of the following types and is most often determined by the chosen scenario generation method:

**Given-structure problem.** Which discrete process $(\tilde{\xi}_t)$, $t = 0, \ldots, T$ with given branching structure $(b_0, b_1, \ldots, b_{T-1})$ is closest to a given process $(\xi_t)$, $t = 0, \ldots, T$? The notion of closeness has to be defined in an appropriate manner. The total number of scenarios is $\prod_{t=0}^{T-1} b_t$.

**Stagewise fixed-structure problem.** The total number of nodes per stage $(n_1, n_2, \ldots, n_T)$ is fixed, thus there will be $n_T$ scenarios.

**Free-structure problem.** The process $(\xi_t)$, $t = 0, \ldots, T$ has to be approximated by $(\tilde{\xi}_t)$, $t = 0, \ldots, T$, but its branching structure and number of nodes per stage $(n_1, \ldots, n_{T-1})$ is completely free except for the fact that the total number of nodes in the final stage (= number of scenarios) $n_T$ is predetermined.

This paper is organized as follows: Section 2 presents an overview of scenario generation techniques for stochastic optimization problems. Section 3 reviews different approximation techniques for single-stage, uni-variate, continuous distributions—both from a theoretical and numerical point of view. In Section 4, one specific methodology for calculating multi-stage scenario trees based on the proposed distances is shown in detail. The quality of this multi-stage scenario generator is substantiated by numerical convergence results applied to financial portfolio management. Section 5 concludes the paper.

## 2 Scenario generation for stochastic programs

We refer to Dupačová et al. (2000) and the references therein for an overview of scenario generation methods. In real-life (decision support) applications, the necessary input for a

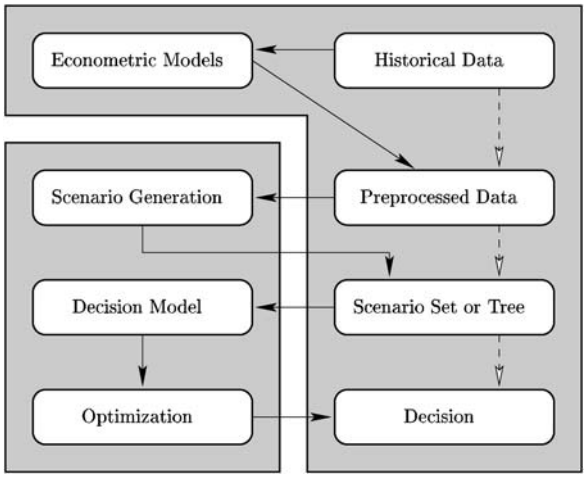



**Fig. 1** Scenario generation workflow: From data to decision

successful calculation of appropriate future scenarios consists of historical data, expert opinion, and econometric models. The general workflow is depicted in Fig. 1. A good method uses a combination of all three sources of information: theoretical considerations (econometric models) reduce the class of acceptable tree models to parametric classes or classes with given structure, historical data is the basis of the estimation process and expert knowledge can be used e.g. to weigh historical data or to change implied trends according to the experts view about future developments.

### 2.1 Approximations of stochastic optimization problems

We assume that the stochastic optimization problem is

$$\min\left\{\int h(x,\omega)\,dG(\omega) : x \in \mathbb{X}\right\}, \tag{1}$$

where $G$ is either a continuous or a discrete random variable with a huge number of mass points on $\mathbb{R}^k$. The scenario approximation problem is to find a simple distribution $\tilde{G}$ with only few mass points such that the problem

$$\min\left\{\int h(x,\omega)\,d\tilde{G}(\omega) : x \in \mathbb{X}\right\}, \tag{2}$$

approximates the original problem (1) well. It is difficult to obtain qualitative answers to the question of how good an approximation is. From the optimization point of view, the goal of the approximation would be such that the difference between the objective functions of the two problems

$$\sup\left\{\left|\int h(x,\omega)\,dG(\omega) - \int h(x,\omega)\,d\tilde{G}(\omega)\right| : x \in \mathbb{X}\right\} \tag{3}$$

is small. To put it differently, the problem is to find an appropriate distance $d$, such that minimizing $d(G,\tilde{G})$ leads to a small value in (3).

From the results of stability of stochastic programs (Rachev and Römisch, 2002) and from certain approximation results (Pflug, 2001) it is known that optimal approximations of stochastic programs in the sense of Eq. (3) can be achieved by minimizing some probability metric (distance) with $\zeta$-structure (as introduced by Zolotarev (1983)), which denotes uniform distances of expectations of functions taken from a class $\mathcal{H}$ of measurable functions. These distances of distributions are typically of the form

$$d_{\mathcal{H}}(G, \tilde{G}) = \sup\left\{\Big| \int h(\omega)\, dG(\omega) - \int h(\omega)\, d\tilde{G}(\omega)| : h \in \mathcal{H}\right\}. \tag{4}$$

We call $\mathcal{H}$ separating, if $d_{\mathcal{H}}(G, \tilde{G}) = 0$ implies that $G = \tilde{G}$, in this case $d$ is called a distance. The family $\mathcal{H}$ defining the distance should be chosen in view of problem (1). If $\mathcal{H}$ contains the functions $\{h(x, \cdot) : x \in \mathbb{X}\}$, then obviously,

$$\Big| \int h(x, \omega)\, dG(\omega) - \int h(x, \omega)\, d\tilde{G}(\omega)\Big| \leq d_{\mathcal{H}}(G, \tilde{G}).$$

Important function classes for the approximation of stochastic programs are (a) Lipschitz continuous functions (Section 2.2), and (b) piecewise constant functions with a given structure of discontinuity sets (Section 2.3).

## 2.2 Lipschitz continuous functions

For $p = 0$ and $p \geq 1$ and $\Omega \subset I\!R^k$ introduce classes $\mathcal{H}_p(\Omega)$ of Lipschitz continuous functions of order $p$:

$$\mathcal{H}_0(\Omega) = \mathcal{H}_1(\Omega) \cap \left\{h \in C_b(\Omega) : \sup_{\omega\in\Omega}|h(\omega)| \leq 1\right\}$$

$$\mathcal{H}_p(\Omega) = \{h : \Omega \to I\!R : |h(\omega) - h(\tilde{\omega})| \leq \max(1, \|\omega\|, \|\tilde{\omega}\|)^{p-1}\|\omega - \tilde{\omega}\| \ \forall \omega, \tilde{\omega} \in \Omega\}$$

The corresponding distances, i.e.

$$d(G, \tilde{G}) = \sup\left\{\Big| \int_\Omega h(\omega) dG(\omega) - \int_\Omega h(\omega) d\tilde{G}(\omega)\Big| : h \in \mathcal{H}_p(\Omega)\right\} \tag{5}$$

are called

- $p = 0$: *Bounded Lipschitz* metric ($d_{BL}$),
- c $p = 1$: *Wasserstein (Kantorovich)* metric ($d_W$),
- $1 < p < \infty$: *Fortet-Mourier* metric of order $p$ ($d_{FM_p}$),
- $p = \infty$: *Kolmogorov (Uniform)* metric ($d_K$).

The Kolmogorov metric coincides with the uniform distance of distribution functions on $I\!R^k$.

### *2.2.1 Wasserstein (Kantorovich) distance*

The goal of the approximation is to find a simple model in such a way that its solution performs well when used as a proxy for the solution of the more complex model. We illustrate this

idea for a two stage stochastic program of the form

$$\min c(x) + \int Q(x,\omega)\, dG(\omega) = \int h(x,\omega)\, dG(\omega),$$

where $c(x)$ are the first stage costs, $Q(x,\cdot)$ are the second stage costs and $h(x,\omega) = c(x) + Q(x,\omega)$. We assume that the family of functions $u \mapsto h(x,\omega)$ is uniformly Lipschitz, i.e. that there is a constant $L$ such that for all $x$: $|h(x,\omega) - h(x,\tilde{\omega})| \le L|\omega - \tilde{\omega}|$. Then we have the inequality

$$\sup_x \left| \int h(x,\omega)\, dG(\omega) - \int h(x,\omega)\, d\tilde{G}(\omega) \right| \le L d_W(G,\tilde{G}), \tag{6}$$

where the Wasserstein distance $d_W$ is defined as in (5). Thus, choosing $d_W(G,\tilde{G})$ as small as possible, one gets that

$$\sup_x \left| \int h(x,\omega)\, dG(\omega) - \int h(x,\omega)\, d\tilde{G}(\omega) \right|$$

is also small. Suppose that $x^*$ is the minimizer of $x \mapsto \int h(x,\omega)\, dG(\omega)$ and $\tilde{x}^*$ is the minimizer of $x \mapsto \int h(x,\omega)\, d\tilde{G}(\omega)$. Then, using the solution $\tilde{x}^*$ as a proxy for the solution $x$ one gets that the error can be bounded by

$$\int h(\tilde{x}^*,\omega)\, dG(\omega) - \int h(x^*,\omega)\, dG(\omega) \le 2 \sup_x \left| \int h(x,\omega)\, dG(\omega) - \int h(x,\omega)\, d\tilde{G}(\omega) \right|.$$

Thus, making $d_W(G,\tilde{G})$ small, then, by (6), the error is controlled. The Wasserstein distance is related to the mass transportation problem (Monge, 1781), see also Rachev (1991), by the following theorem.

**Theorem 1** (Kantorovich-Rubinstein).

$$\begin{aligned} d_W(G;\tilde{G}) = \inf\{ & I\!E(|X - \tilde{X}|;\ \textit{where the joint distribution } (X,\tilde{X}) \\ & \textit{is arbitrary, but the marginal distributions are fixed} \\ & \textit{such that } X \sim G; \tilde{X} \sim \tilde{G}\} \end{aligned} \tag{7}$$

For one-dimensional distributions, $d_W$ is defined as

$$d_W(G,\tilde{G}) = \int_\Omega |G(\omega) - \tilde{G}(\omega)| = \int_\Omega |G^{-1}(\omega) - \tilde{G}^{-1}(\omega)|,$$

where $G^{-1}(\omega) = \sup\{v : G(v) \le \omega\}$ (see Vallander (1973)). Among all one-dimensional $\tilde{G}$, which sit on the mass points $z_1, z_2, \ldots z_m$, the one closest to $G$ in $d_W$-distance has masses

$$p_i = G(\frac{z_i + z_{i+1}}{2}) - G(\frac{z_i + z_{i-1}}{2}),$$

where $z_0 = -\infty$ and $z_{m+1} = +\infty$. For this $\tilde{G}$,

$$d_W(G, \tilde{G}) = \sum_{i=1}^{m} \int_{\frac{z_{i-1}+z_i}{2}}^{\frac{z_i+z_{i+1}}{2}} |\omega - z_i| \, dG(\omega),$$

the infimum in (7) is attained. The optimal joint distribution $(X, \tilde{X})$ describes, how the mass with distribution $G$ should be transported with minimal effort to yield the new mass distribution $\tilde{G}$.

The Wasserstein distance is also defined for the multi-variate case, see Rachev and Rüschendorf (1998).

*2.2.2 Fortet-Mourier and Wasserstein distance*

It is possible to relate the Fortet-Mourier distance to the Wasserstein distance. Let

$$\chi_p(u) = \begin{cases} u & |u| \le 1 \\ |u|^p \operatorname{sgn}(u) & |u| \ge 1 \end{cases}$$

Notice that $\chi_p^{-1}(u) = \chi_{1/p}(u)$. It was shown in Pflug (2001) that the Lipschitz constants of order $p$ satisfy $L_p(f \circ \chi_p) \le p \cdot L_1(f)$ and $L_1(f \circ \chi_{1/p}) \le L_p(f)$, and therefore

$$\frac{1}{p} d_W(G \circ \chi_{1/p}, \tilde{G} \circ \chi_{1/p}) \le d_{FM_p}(G, \tilde{G}) \le d_W(G \circ \chi_{1/p}, \tilde{G} \circ \chi_{1/p}). \tag{8}$$

$G \circ \chi_{1/p}$ is the distribution of $|X|^p \operatorname{sgn}(X)$, if $X$ has distribution $G$. Based on relation (8), we may replace the problem of approximating with respect to the Fortet-Mourier distance by the easier problem of approximation with respect to the Wasserstein distance by the following algorithm:

- Choose a power $p$ depending on the underlying problem.
- Transform $G$ by $\chi_{1/p}$ to get $G^{(1/p)} = G \circ \chi_{1/p}$.
- Approximate $G^{(1/p)}$ by a distribution $\tilde{G}^{(1/p)}$ sitting on $m$ points in such a way that the Wasserstein distance $d_W(G^{(1/p)}, \tilde{G}^{(1/p)})$ is minimal.
- Transform back: $\tilde{G} = \tilde{G}^{(1/p)} \circ \chi_p$

Suppose that $d_W(G^{(1/p)}, \tilde{G}^{(1/p)}) \le \epsilon$. Then, by (8), $d_{FM_p}(G, \tilde{G}) \le \epsilon$ and $|\int \omega^q \, dG(\omega) - \int \omega^q \, d\tilde{G}(\omega)| \le p\epsilon$, for $1 \le q \le p$. In addition, for all Lipschitz functions $h$, $|\int h(\omega) \, dG(\omega) - \int h(\omega) \, d\tilde{G}(\omega)| \le L(h)\,\epsilon$. Thus, the difference of integrals for all Lipschitz functions, as well as the difference of all moments of order at most $p$ can be controlled.

2.3 Piecewise constant functions with a given structure of discontinuity sets

Let $\mathcal{B}$ denote a set of all Borel subsets of $\Omega$. The $\mathcal{B}$-discrepancy is the distance defined by

$$d_D(G, \tilde{G}) = \sup_{B \in \mathcal{B}} |G(B) - \tilde{G}(B)|. \tag{9}$$

For different structures of $\Omega$ and $\mathcal{B}(\Omega)$ there are various types of discrepancy distances. As a special case, if $\Omega = I\!R^k$ and $\mathcal{B}(\Omega) = \{(-\infty, \omega] : \omega \in I\!R^k\}$ then one arrives at the Kolmogorov

distance (see above). Low-discrepancy (quasi-random) sequences have been applied to the approximation of stochastic programs by Pennanen and Koivu (2005) successfully. These methodologies can be used to discretize a discrete-time, continuous-space stochastic process into a (highly-dimensional) discrete-space process for easier numerical processing with the methodologies presented below.

### 2.4 Applied scenario generation

In practical applications, such as financial investment management, theoretical considerations of approximations are sometimes condemned. Simpler approaches are preferred, which are easier to implement as well as simpler to communicate. However, this causes a loss of quality, which often leads to a non-acceptability of multi-stage stochastic programming methods for practical usage.

## 3 Single-stage approximations

If the depth of the tree is 1, we have the single-stage case. In this case, the problem reduces to the problem of approximating a given probability distribution $G$ by a discrete distribution $\tilde{G}$ having a predetermined number of mass points.

### 3.1 Matching moments

In terms of probability metrics, if $\mathcal{H}$ is the class of all power functions on $\mathbb{R}$, then the *moment matching distance* is obtained

$$d_{MM}(G,\tilde{G}) = \sup\left\{ \int \omega^p \, dG(\omega) - \int \omega^p \, d\tilde{G}(\omega) : 1 \le p \le M \right\}$$

Moment matching might not lead to a distance, even if all moments are considered ($M = \infty$). A simple example of infinitely many different distributions, all having the same moments of any order is due to Heyde (1963).

Matching statistical moments, especially matching the first four moments of a probability distribution introduced by Høyland and Wallace (2001) is a commonly used method. However, moment matching may lead to strange results as is illustrated in two examples below.

First, consider the two densities $g_1$ and $g_2$, which are plotted in Fig. 2.

$$\begin{aligned} g_1(x) = 0.39876\,[\,&\exp(-|x+0.2297|^3)\mathbb{I}_{\{x\le -1.2297\}} \\ +\;&\exp(-|x+0.2297|)\mathbb{I}_{\{-1.2297<x\le -0.2297\}} \\ +\;&\exp(-0.4024\cdot(x+0.2297))\mathbb{I}_{\{-0.2297<x\le 2.2552\}} \\ +\;&1.09848\cdot(0.4024x+0.29245)^{-6}\mathbb{I}_{\{2.2552<x\}}]. \end{aligned}$$

$$g_2(x) = 0.5962\mathbb{I}_{\{|x|\le 0.81628\}} + 0.005948\mathbb{I}_{\{0.81628<|x|\le 3.05876\}}.$$

Both densities are unimodal and coincide in the first four moments, which are: $m_1 = 0$, $m_2 = 0.3275$, $m_3 = 0$, $m_4 = 0.72299$. The fifth moments however could not differ more: While $g_1$ has infinite fifth moment, the fifth moment of $g_2$ is zero. Density $g_1$ is asymmetric,

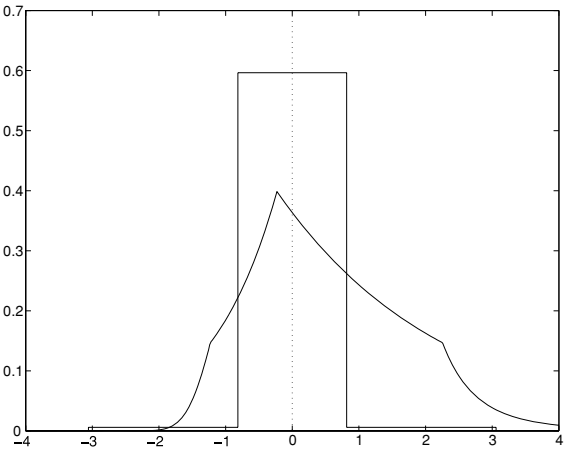



**Fig. 2** Two densities with identical first four moments

has a sharp cusp at the point $-0.2297$ and unbounded support. In contrast, $g_2$ is symmetric around 0, has a flat density there, has finite support and possesses all moments. The quantiles differ drastically: While for $g_1$ the probability of the interval $(-0{,}81628, 0.81628)$ is 43.56%, the same interval has probability 97.33% for $g_2$.

Furthermore, the following four distributions also coincide in all first four moments:

1. A uniform distribution in the interval $[-2.44949, 2.44949]$.
2. The mixture of two normal distributions $N(1.244666, 0.450806)$ and $N(-1.244666, 0.450806)$ with equal weights 0.5.
3. The discrete distribution

| Value | −2.0395 | −0.91557 | 0 | 0.91557 | 2.0395 |
|---|---|---|---|---|---|
| Probability | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 |

4. The discrete distribution

| Value | −3.5 | −1.4 | 0 | 1.4 | 3.5 |
|---|---|---|---|---|---|
| Probability | 0.013 | 0.429 | 0.1162 | 0.429 | 0.013 |

These distributions are shown in Fig. 3, i.e. distribution 1, 2, 4 (left), and 1, 2, 3 (right) respectively. A visual inspection shows, that these distributions do not have much in common. The comparison is even more striking, if we solve the same stochastic program with the four different distributions. Let $X$ be distributed according to $G$ and let $a$ be the solution of the stochastic program

$$\min\{I\!E([X-a]^+ + 1.3[X-a]^-) : a \in I\!R\}. \tag{10}$$

Here $[u]^+ = \max(u, 0)$ and $[u]^- = -\min(u, 0)$. Problem (10) is the well-known news-vendor problem. We solved the same problem for the four mentioned distributions of the random demand variable $X$. The results are shown in Table 1, and are quite different.

**Table 1** Solutions of news-vendor problem

| Distribution | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Solution | −0.3194 | −0.5040 | 0 | −1.4 |

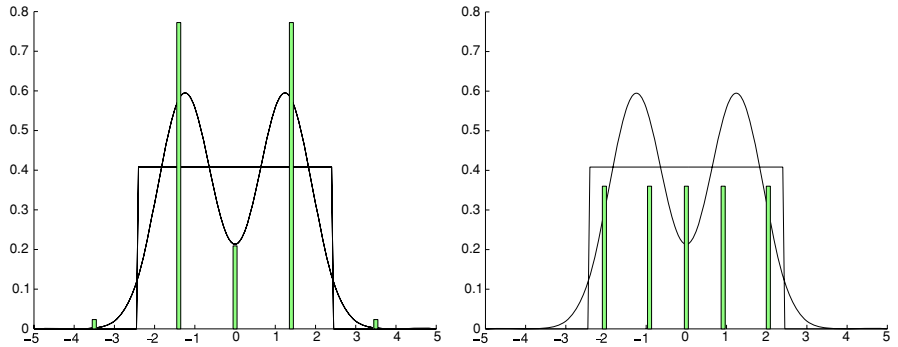


**Fig. 3** Four distributions with identical first four moments

Although moment matching may perform better than crude random sampling and adjusted random sampling for stochastic asset liability management problems as shown e.g. by Kouwenberg (2001), it is obviously awkward to use this methodology in terms of reliability and credibility of the approximations.

### 3.2 Kolmogorov-Smirnov (Uniform) distance

Besides the moment matching method, a common approach is to minimize the Kolmogorov-Smirnov distance between the original distribution $G$ and its approximation $\tilde{G}$, which coincides with the uniform distance of distribution functions on $\mathbb{R}$. The optimal approximation of a continuous distribution by a distribution sitting on $n$ mass points $z_1, \ldots, z_n$ with probabilities $p_1, \ldots, p_n$ with respect to the Kolmogorov-Smirnov distance is given by

$$z_i = G^{-1}\left(\frac{2i-1}{2n}\right), \quad p_i = \frac{1}{n} \tag{11}$$

for univariate distributions. Notice that every mass point $z_i$ has the same probability $p_i = \frac{1}{n}$ and therefore tails are not well represented. This is illustrated in Fig. 4.

### 3.3 Wasserstein distance and the facility location problem

Due to the Kantorovich-Rubinstein theorem (Section 2.2.1) the problem of minimizing the Wasserstein distance between a distribution $G$ and a distribution $\tilde{G}$ with $n$ points can be viewed as a facility location problem: Find the locations $z_1, \ldots, z_n$ of facilities in such a manner that the mean travel distance to the nearest location $\int \min_i |\omega - z_i| \, dG(\omega)$ is minimized.

Suppose we want to approximate the $t$-Student distribution $G$ with two degrees of freedom, i.e. density $(2 + x^2)^{-3/2}$ by a discrete distribution sitting on 5 points. The optimal approximation with the Kolmogorov-Smirnov distance is $\tilde{G}_1$ and shown in Table 2, while minimizing the Wasserstein distance returns the approximated distribution $\tilde{G}_2$, which is shown in Table 3. These results are compared in Fig. 4, where one can see, that the minimization of the Wasserstein distance leads to a much better approximation of the tails. Minimizing the Wasserstein distance minimizes the difference in expectation for all Lipschitz functions. However, it may not lead to the best approximation for higher moments. If a further refined approximation of higher moments is also required, the Fortet-Mourier distance may be used.

**Table 2** $t$-Student(2) Approximation: KS distance $\tilde{G}_1$

| Value | −1.8856 | −0.6172 | 0 | 0.6172 | 1.8856 |
|---|---|---|---|---|---|
| Probability | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 |

**Table 3** $t$-Student(2) Approximation: Wasserstein $\tilde{G}_2$

| Value | −4.58 | −1.56 | 0 | 1.56 | 4.58 |
|---|---|---|---|---|---|
| Probability | 0.0446 | 0.2601 | 0.3906 | 0.2601 | 0.0446 |

**Table 4** Approximation: Wasserstein with transformation

| Value | −4.8216 | −1.6416 | 0 | 1.6416 | 4.8216 |
|---|---|---|---|---|---|
| Probability | 0.0404 | 0.1615 | 0.4038 | 0.1615 | 0.0404 |

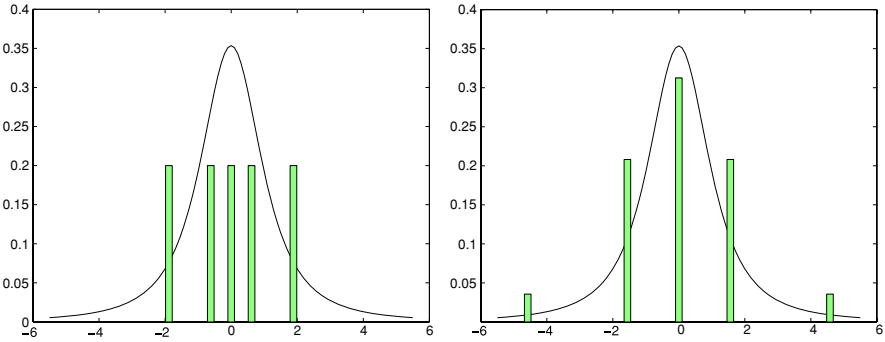



**Fig. 4** Approximation: Kolmogorov-Smirnov (left), Wasserstein (right)

Approximating tails and higher moments is useful for many applications, especially in the area of financial management. Reconsider the example of approximating the $t$-Student distribution with two degrees of freedom. Applying the stretching function $\chi_2$ to the $t$-distribution, one gets the distribution with density

$$\begin{cases} (2+x^2)^{-3/2} & \text{if } |x| \le 1, \\ \frac{1}{2\sqrt{x}}(2+|x|)^{-3/2} & \text{if } |x| > 1. \end{cases}$$

This distribution has heavier tails than the original $t$-distribution. We approximate this distribution w.r.t. the Wasserstein distance by a 5-point distribution and retransform the mass points using the transformation $\chi_{1/2}$ one gets the result shown in Table 4. Compared to the previous one, this approximation has an even better coverage of the tails.

## 4 Multi-stage approximations

### 4.1 Multi-stage approximations

There are basically two ways of constructing multi-stage scenario trees, if the complete reference process is available. One can either build a tree starting from the root node (forward process) or from the leaf nodes (backward process). Nearly all proposed methods use some sort of forward scheme. One exception is the scenario tree reduction method suggested by

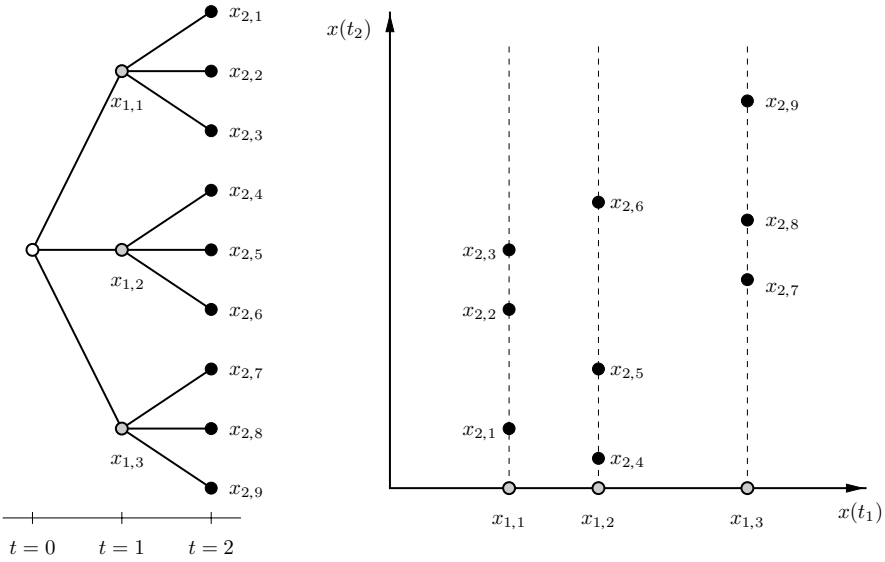



**Fig. 5** Scenario tree (left) and multi-dimensional facility location (right)

Heitsch and Römisch (2003) and Dupačová, Gröwe-Kuska, and Römisch (2003), which is also based on a minimization of probability metrics. Forward-type scenario generators calculate the tree recursively starting at the root node. At the root node, a discretization for the first stage is calculated with the chosen discretization algorithm, while calculations at successor nodes for subsequent stages are based on a set of data conditioned to predecessor approximations. Suppose a chosen value for the first stage is $x$. A possibility is to condition the reference process $\xi_t$ to the set $\{\omega : |\xi_1 - x| \leq \epsilon\}$, where $\epsilon$ is selected in such a way that enough trajectories are contained in this set. The successor nodes of $x$ are then calculated as if $x$ were the root and the conditional process were the entire process. This approach facilitates a solution to the given structure problem.

Without a given reference process, other modifications for subsequent stages can be deployed—almost exclusively in a forward fashion. To build multi-stage given-structure trees with moment matching approximations, Høyland and Wallace (2001) suggest to calculate four target moments (and a worst-case factor) from a continuous density, which is estimated from seven quantiles, which an expert predicted for future outcomes (e.g. future return of different asset classes within financial management), and the correlation—calculated with a set of corresponding historical data—for the root approximation. For approximations of successor nodes, the first and the second moment are modified according to some pre-specified rules.

The proposed method in this paper applies a backward recursive technique based on multi-dimensional facility location problems with additional constraints (multi-stage constraints). One draw-back is, that facility location problems are known to be $\mathcal{NP}$-hard. Approximation algorithms and heuristics have to be applied to calculate numerical solutions. Common approaches are mainly based on meta-heuristics as well as linear programming relaxation techniques.

The relation between a scenario tree and a facility location map is shown in Fig. 5. Each stage of the tree equals one dimension of the corresponding facility location problem. Notice

that this problem is not just a multi-variate extension of the facility location problem, because in order to obtain a tree with given branching structure, the facilities have to be located in such a manner that only $b_0$ different first coordinates of the facilities may exist. If $b_0$ streets have to be selected, then $b_1$ different second coordinates may be chosen conditionally on these streets, and so on. The selected values in the last stage represent the facilities.

For numerical reasons, if the original distribution is continuous (e.g. estimated distributions), some method should be applied to generate a highly-dimensional discrete version of the continuous space without quantitative losses. The methods presented by Pennanen and Koivu (2005) and Koivu (2005) are ideally suited for this operation, because implementations are readily available. The proofs for the quantity argument of these methods can be found in (Pennanen, 2005).

In most cases, scenarios are already discrete, e.g. simulated scenario paths of some fitted econometric time series model. In this discrete case, we have a finite set of (multi-variate) scenario paths, where the vector of values is known for each stage from the root stage to the terminal stage—for each path. It is feasible to use a set of paths, that already exhibit a tree structure, or are combined in another way at intermediary stages between root and terminal nodes. If the input is a tree, we speak about scenario (tree) reduction, if paths are used, then we speak about scenario (tree) generation. Additionally, any already connected structure can always be expanded to a set of paths.

If the process $\xi_t$ takes values in $\mathbb{R}^k$, then the same procedures can be used to estimate multivariate trees, since the Wasserstein and Fortet-Morier distances are also well defined in several dimensions—often as weighted version, where one can assign different weights to each dimension. Thus we may construct multivariate, multi-stage trees.

Consider the following numerical example to visualize the connection between the *scenario tree view* and the *facility location view* applied to real financial data. Daily data of the Standard and Poors 500 Index from January 1, 1995 to January 1, 2005 have been used to fit an ARMA(1,1)/GJR(1,1) time series model, from which 400 paths have been simulated for a possible future development for the next 3 months. The estimation and simulation has been calculated with the MatLab GARCH Toolbox. These paths are shown in the left part (scenario tree view) of Fig. 6. The right part shows the facility location view of these 400 paths. To obtain a useful visualization, only two stages ($T = 2$) are considered, hence we obtain a facility location problem in $\mathbb{R}^2$. The two-dimensional view shows the value of each simulated path on February 1, 2005 and April 1, 2005. Each facility corresponds to one path.

An implementable algorithm, which resembles the multi-dimensional facility location problem as shown by Pflug (2001) in the aforementioned backward fashion, can be summarized as follows: it mainly applies a backward distance minimization (nested clustering) with dimension reduction, and builds the scenario tree from the root node, based on these clusterings. Given a fixed structure of $n_t = [n_1, \ldots, n_T]$ nodes per stage $t$ for all $T$ stages, and some chosen distance $d(\cdot)$:

1. (Initial Cluster) Cluster $n_T$ centers in $\mathbb{R}^T$ with distance $d(\cdot)$, set $i := T - 1$. Store association of data points and clusters.
2. (Intermediate Clustering) Cluster $n_i$ centers in $\mathbb{R}^i$ for stage $i$. Only use data from stage 1 to $i$. Again, store association of data points and clusters.
3. (Stopping criterion) If not at root stage ($i > 1$), reduce $i$ by 1 and go to (2).
4. (Build predecessor list) For each stage $i := 2$ to $T$ use the cluster/data point association list to assign predecessors correctly.

This algorithm depends on some clustering algorithm as its basic building block, which is executed $T$ times. Unfortunately, the problem of partitioning (or clustering) $n$ data points

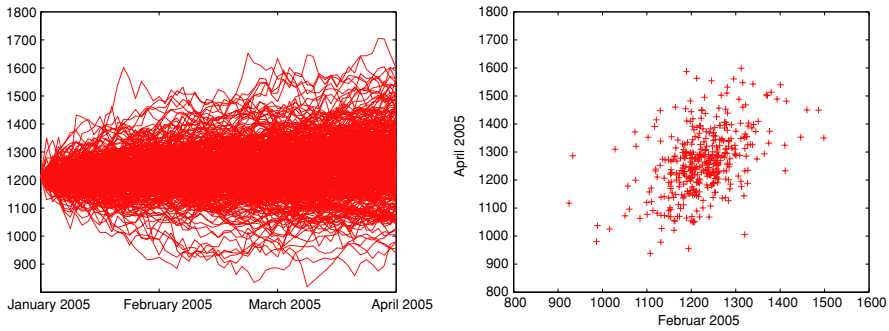



**Fig. 6** Simulated scenario paths (left) in facility location setting (right)

into $k$ disjoint subsets containing $n_j$ data points, by minimizing some distance criterion is generally $\mathcal{NP}$-complete, and polynomial time algorithms are generally not available, as already mentioned above. Besides using the well-known $k$-means algorithm, which was related to scenario generation in (Pflug, 2001) or some other randomized heuristic, a very good choice is using a greedy clustering algorithm. The run-time of a primitive greedy algorithm is $O((n \cdot (n-1)/2) + (n-k) \cdot (n \cdot (n-1)/2)) \approx O(n^3)$, hence polynomial. Basically, the numerical problem besides calculating the distance matrix, i.e. the distance of each point in the initial data matrix to the others, is caused by $(n-k)$ times finding the minimum in the distance matrix. Of course, this algorithm can be further refined, to gain speedups, especially by using parallel computation techniques. A special advantage of the greedy algorithm is, that there is no random action involved, such that experiments can easily be repeated. The real advantage of this heuristic, which on one hand cannot guarantee a global optimum is, that it inherently considers outliers (extreme events) at the very end of the clustering process. This effect makes it most attractive for stochastic programming problems, as one of the main objectives is to hedge against extreme events. Those events will receive a small probability, but will be considered.

The algorithm was implemented in MatLab. The final scenario tree using the 400 simulated paths described above and applying the Wasserstein distance is shown in Fig. 7. This tree has a stagewise fixed structure with $N = (12, 40)$, i.e. 40 scenarios. The probability distribution of these 40 scenarios is depicted in Fig. 8. One clearly sees that extreme scenarios are included, but with a lower probability. These 40 scenarios are drawn as circles in the facility location view. Those $n_2 = 40$ facilities are located on $n_1 = 12$ streets.

### 4.2 Numerical convergence of multi-stage approximations

We consider a simple multi-stage stochastic asset management problem to verify numerical properties of the proposed scenario generator. A discrete-time investment horizon $\mathcal{T}$ with stages $t = 0, \ldots, T$ is considered. Let $B$ be the initial budget invested at the root node. Investments are only allowed until stage $T-1$. The dynamics of holdings in each asset class are described by $x_{i,t} = \xi_i^t x_i^{t-1} + b_{i,t} - s_{i,t}$ for each asset $i \in \mathcal{I}$, where $\xi_{i,t}$ is the (stochastic) return on investment $i$ in period $[t-1, t]$, $b_{i,t}$ and $s_{i,t}$ denote the purchases and sales of investment $i$ at stage $t$, and $x_{i,t}$ is the portfolio value of investment $i$ in period $[t-1, t]$. Budget constraints guarantee that the total expenses do not exceed revenues. Portfolio constraints give bounds for the allowed range of portfolio weights, i.e. let $l_i$ and $u_i$ be relative lower and

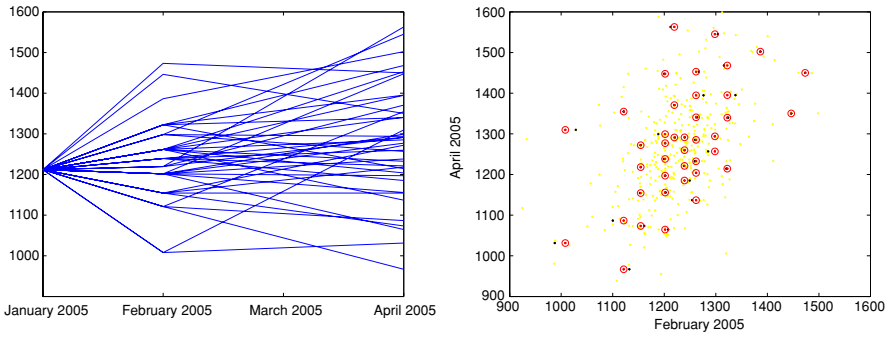
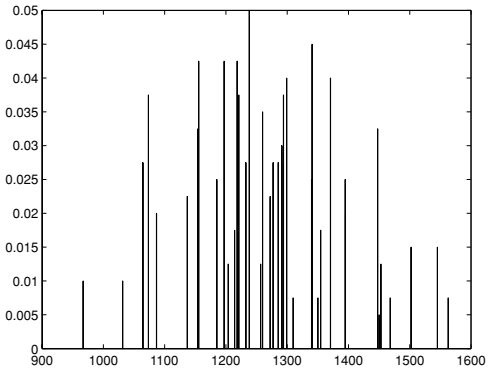



**Fig. 7** Generated scenario tree (left) in facility location setting (right)

**Fig. 8** Probabilities of the final stage of the generated scenario tree




upper investment limit on asset $i$. Furthermore, let $f_t$ be the (deterministic) cash in-flow at stage $t$, and $w_t$ the total wealth at stage $t$.

For the objective function, the terminal mean-risk approach has been chosen, i.e. one maximizes expected total wealth and considers the (terminal) Average Value at Risk (AVaR, also called Conditional Value at Risk (CVaR)) of the total (terminal) wealth as the risk functional. Let $\mathrm{AVaR}_\alpha$ be defined as the solution of the optimization problem

$$\inf\{a + \frac{1}{1-\alpha}I\!E[Y-a]^+ : a \in I\!R\},$$

where $Y$ is a random cost variable. With a finite set of scenarios this optimization problem can be reformulated as a linear program (see Rockafellar and Uryasev (2000)).

The resulting optimization problem in tree formulation is

$$\begin{array}{lll}
\max_x & \sum_{n\in\mathcal{N}^T} p_n w_n + \kappa(\gamma - \sum_{n\in\mathcal{N}^T} \frac{p_n z_n}{1-\alpha}) & \\
\text{subject to} & \sum_{i\in\mathcal{I}} x_{1,i} \le B = w_1 & \\
& x_{n,i} \le \xi_{n,i} x_{P(n),i} + b_{n,i} - s_{n,i} & \forall n \in \mathcal{N}^t, \forall i \in \mathcal{I} \\
& l_i w_n \le x_{n,i} \le u_i w_n & \forall n \in \mathcal{N}^t, \forall i \in \mathcal{I} \\
& \sum_{i\in\mathcal{I}} b_{n,i} \le \sum_{i\in\mathcal{I}} s_{n,i} & \forall n \in \mathcal{N}^t \\
& w_n = \sum_{i\in\mathcal{I}} x_{n,i} + f_{S(n)} & \forall n \in \mathcal{N}^t \\
& w_n = (\sum_{i\in\mathcal{I}} \xi_{n,i} x_{p(n),i}) + f_{S(n)} & \forall n \in \mathcal{N}^T \\
& z_n \ge \gamma - w_n & \forall n \in \mathcal{N}^T
\end{array} \tag{12}$$

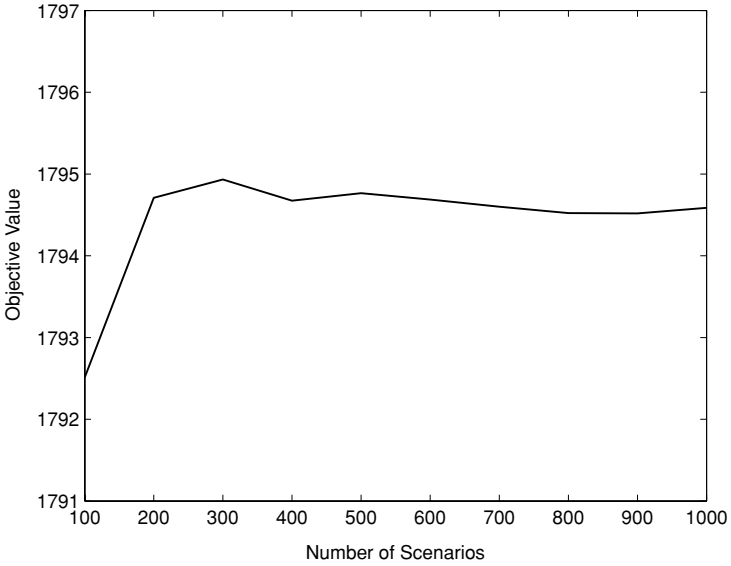



**Fig. 9** Numerical convergence of the objective value

where $\mathcal{N}^T$ is the set of terminal nodes, and $\mathcal{N}^t$ the set of nodes, at which trading occurs. The function $P(n)$ returns the predecessor node of node $n$. $p_n$ is the probability of the whole scenario, that terminates at (terminal) node $n$. The function $S(n)$ returns the stage of node $n$. $\alpha$ is the AVaR quantile level, and $\kappa$ the risk aversion parameter. $z_n$ and $\gamma$ are auxiliary variables for the linear programming reformulation of the AVaR problem. Transaction costs are not considered in this tree model, but can be added easily.

The linear programs were modeled with AMPL Fourer, Gay, and Kernighan (2002) and solved with the MOSEK interior-point method solver. The portfolio optimization framework was implemented in MatLab.

To keep the model simple, two assets have been taken, of which one is stochastic and the other one is deterministic. Fig. 9 shows the convergence of the objective function for a stage-wise fixed tree with 5 stages (excluding the root stage) with structure $[k, 2k, 3k, 4k, 5k]$ for $k = 20$ to 200. 2000 scenario paths have been sampled from the S&P 500 model, that has already been introduced at the beginning of the chapter. No deterministic cash inflow has been used, and the lower limit $l$ was set to 0, and the upper limit $u$ to 1. One can see, that even with a small set of scenarios, the value of the objective function is stable, which is one of the most important properties from a numerical point of view.

## 5 Conclusions

We have demonstrated the relation between optimal scenario generation and multi-dimensional facility location. Common approaches for generating scenario trees are matching of statistical moments or minimizing the Kolmogorov-Smirnov distance (uniform distance of distribution functions), together with some forward tree building method. While moment matching may lead to strange approximations, the uniform distance does not take care of tails and higher moments. The advantage of the proposed method is that it combines a good

approximation of moments (if the appropriate distance is taken) with a good, controllable approximation of the tails. The creation of non-equally weighted scenarios is especially important for multi-stage trees, as this approach ensures that extreme events are considered even if the size of the approximated set is small. Numerical evidence for this effect has been shown with a multi-stage stochastic mean-risk financial programming problem.

Whenever the objective of the approximation is to achieve a controlled matching of certain moments and a controllable coverage of heavy tails, scenario generation based on multi-dimensional facility location should be taken into consideration. Future research on this topic includes unbiased numerical comparisons between different scenario generation methods based on different multi-stage times-series and optimization models, as well as methods to determine the optimal branching factor of scenario trees.

**Acknowledgements** The work described in this paper was partially supported by the Austrian Science Fund as part of AURORA Project under contract SFBF1106.

## References

Dupačová, J., G. Consigli, and S.W. Wallace. (2000). "Generating Scenarios for Multistage Stochastic Programs." *Annals of Operations Research*, 100, 25–53.

Dupačová, J., N. Gröwe-Kuska, and W. Römisch. (2003). "Scenario Reduction in Stochastic Programming: An Approach Using Probability Metrics." *Mathematical Programming, Series A*, 95, 493–511.

Fourer, R., D. Gay, and B. Kernighan. (2002). *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press/Brooks/Cole Publishing Company.

Heitsch, H. and W. Römisch. (2003). "Scenario Reduction Algorithms in Stochastic Programming." *Computational Optimization and Applications*, 24(2–3), 187–206.

Heyde, C. (1963). "On the Property of the Lognormal Distribution." *Journal of the Royal Statistical Society, Series B*, 25, 392–393.

Høyland, K. and S.W. Wallace. (2001). "Generating Scenario Trees for Multistage Decision Problems." *Management Science*, 47, 295–307.

Koivu, M. (2005). "Variance Reduction in Sample Approximations of Stochastic Programs." *Mathematical Programming, Series A*, 103(3), 463–485.

Kouwenberg, R. (2001). "Scenario Generation and Stochastic Programming Models for Asset Liability Management." *European Journal of Operation Research*, 134, 279–292.

Monge, G. (1781). "Mémoire sur la théorie des déblais et remblais." Hist. Acad. R. Sci. Paris.

Pennanen, T. (2005). "Epi-Convergent Discretizations of Multistage Stochastic Programs." *Mathematics of Operations Research*, 30(1), 245–256.

Pennanen, T. and M. Koivu. (2005). "Epi-Convergent Discretizations of Stochastic Programs via Integration Quadratures." *Numerische Mathematik*, 100(1), 141–163.

Pflug, G. (2001). "Scenario Tree Generation for Multiperiod Financial Optimization by Optimal Discretization." *Mathematical Programming, Series B*, 89, 251–257.

Rachev, S. and W. Römisch. (2002). "Quantitative Stability in Stochastic Programming: The Method of Probability Metrics." *Mathematics of Operations Research*, 27, 792–818.

Rachev, S.T. (1991). *Probability Metrics and the Stability of Stochastic Models*. New York: John Wiley and Sons.

Rachev, S.T. and L. Rüschendorf. (1998). *Mass Transportation Problems. Vol. I*, Probability and its Applications. New York: Springer-Verlag.

Rockafellar, R. and S. Uryasev. (2000). "Optimization of Conditional Value-at-Risk." *The Journal of Risk*, 2(3), 21–41.

Ruszczynski, A. and A. Shapiro (Eds.). (2003). *Stochastic Programming*, Vol. 10 of *Handbooks in Operations Research and Management Science*. Elsevier.

Vallander, S. (1973). "Calculation of the Wasserstein Distance Between Probability Distributions on the Line." *Theory of Probability and Its Applications*, 18, 784–786.

Wallace, S.W. and W.T. Ziemba (Eds.). (2005). *Applications of Stochastic Programming*, MPS-SIAM Series on Optimization. SIAM.

Zolotarev, V. (1983). "Probability Metrics." *Theory of Probability and Its Applications*, 28, 278–302.