Stochastic Dynamic Programming Using Optimal Quantizers

Anna Timonina-Farkas *

École Polytechnique Fédérale de Lausanne, Risk, Analytics and Optimization Chair
Schrödinger Fellowship of the Austrian Science Fund (FWF)

Georg Ch. Pflug **

University of Vienna, Institute of Statistics and Operations Research
International Institute for Applied Systems Analysis (Austria), Risk and Resilience Program

Multi-stage stochastic optimization is a well-known quantitative tool for decision-making under uncertainty, which applications include financial and investment planning, inventory control, energy production and trading, electricity generation planning, supply chain management and similar fields. Theoretical solution of multi-stage stochastic programs can be found explicitly only in very exceptional cases due to the complexity of the functional form of the problems. Therefore, the necessity of numerical solution arises. In this article, we introduce a new approximation scheme, which uses optimal quantization of conditional probabilities instead of typical Monte-Carlo simulations and which allows to enhance both accuracy and efficiency of the solution. We enhance accuracy of the estimation by the use of optimal distribution discretization on scenario trees, preserving efficiency of numerical algorithms by the combination with the backtracking dynamic programming. We consider optimality of scenario quantization methods in the sense of minimal Kantorovich-Wasserstein distance at each stage of the scenario tree, which allows to implement both structural and stage-wise information in order to take more accurate decisions for the future, as well as to bound the approximation error. We test efficiency and accuracy of proposed algorithms on the well-known Inventory Control Problem, for which explicit theoretical solution is known, as well as we apply the developed methods to the budget allocation problem for risk-management of flood events in Austria.

Keywords: multi-stage stochastic optimization, scenario trees, optimal quantization, dynamic programming, Kantorovich-Wasserstein distance, inventory control problem, budget allocation problem, natural disasters, floods, risk-management

AMS Subject classification: 90C06, 90C15, 90C39, 90B05, 90B50

1. Introduction

Nowadays, people, companies and technologies in our fast-developing and changing world starting to face more situations and problems, where they need to take decisions under uncertainty in a multi-period environment (e.g. Pflug [18], Pflug and Römisch [19]). Multi-stage stochastic optimization is a well-known mathematical tool for solution of multi-period decision-making problems under uncertainty (e.g. Ermoliev, Marti and Pflug [5], Shapiro, Dentcheva and Ruszczyński [28]). Our goal is to study numerical methods for the solution of these problems by the use of approximation techniques (see Pflug and Pichler [21]).

* anna.farkas@epfl.ch; http://homepage.univie.ac.at/anna.timonina/
** georg.pflug@univie.ac.at; http://stochopt.univie.ac.at

We focus on stochastic processes given by continuous-state probability distributions, estimated data-based and changing over time conditional on new realizations (e.g. Mirkov and Pflug [15], Mirkov [16]). Based on the estimated distributions, we approximate stochastic processes by *scenario trees* (e.g. Heitsch and Römisch [10], Pflug and Pichler [21]), which we directly use to solve multi-stage stochastic optimization problems numerically. Focusing on different types of scenario tree approximation, we search for a compromise of accuracy and efficiency between numerical solution methods for multi-stage stochastic optimization programs.

Mathematically speaking, suppose that a multi-stage expectation-minimization stochastic optimization program is given in the form with loss/profit function $H(x,\xi) = h_0(x_0) + \sum_{t=1}^{T} h_t(x^t, \xi^t)$ (e.g. Pflug and Römisch [19], Pflug [20], Pflug and Pichler [21,22]):

$$\inf_{x \in \mathbb{X}, x \lhd \mathcal{F}} \left\{ \mathbb{E}\left[H(x,\xi) = h_0(x_0) + \sum_{t=1}^{T} h_t(x^t, \xi^t) \right] \right\}, \tag{1}$$

where $\xi = (\xi_1, ..., \xi_T)$ is a continuous-state stochastic process ($\xi_t \in \mathbb{R}^{r_1},\ \forall t = 1, ..., T$) defined on the probability space $(\Omega, \mathcal{F}, P)$ and $\xi^t = (\xi_1, ..., \xi_t)$ is its history up to time t; $\mathcal{F} = (\mathcal{F}_1, ..., \mathcal{F}_T)$ is a filtration on the space $(\Omega, \mathcal{F}, P)$ to which the process ξ is adapted (i.e. ξ_t is measurable with respect to σ-algebra $\mathcal{F}_t,\ \forall t = 1, ..., T$): we denote it as $\xi \lhd \mathcal{F}$, as well as we add the trivial σ-algebra $\mathcal{F}_0 = \{\emptyset, \Omega\}$ as the first element of the filtration $\mathcal{F}$. A sequence of decisions $x = (x_0, ..., x_T)$ ($x_t \in \mathbb{R}^{r_2},\ \forall t = 0, ..., T$) with history $x^t = (x_0, ..., x_t)$ must be also adapted to $\mathcal{F}$: i.e. it must fulfill the non-anticipativity conditions $x \lhd \mathcal{F}$ (e.g. Pflug [20], Pflug and Pichler [21,22]), that means only those decisions are feasible, which are based on the information available at the particular time. $\mathbb{X}$ is the set of constraints on x other than the non-anticipativity constraints.

The approximated problem (2) can be written correspondingly in the form with the loss/profit function $H(\widetilde{x}, \widetilde{\xi}) = h_0(\widetilde{x}_0) + \sum_{t=1}^{T} h_t(\widetilde{x}^t, \widetilde{\xi}^t)$:

$$\inf_{\widetilde{x} \in \mathbb{X}, \widetilde{x} \lhd \widetilde{\mathcal{F}}} \left\{ \mathbb{E}\left[H(\widetilde{x}, \widetilde{\xi}) = h_0(\widetilde{x}_0) + \sum_{t=1}^{T} h_t(\widetilde{x}^t, \widetilde{\xi}^t) \right] \right\}, \tag{2}$$

where the stochastic process ξ is replaced by a scenario process $\widetilde{\xi} = (\widetilde{\xi}_1, ..., \widetilde{\xi}_T)$, such that $\widetilde{\xi}_t \in \mathbb{R}^{r_1},\ \forall t = 1, ..., T$ with $\widetilde{\xi}_t$ being discrete (i.e. $\widetilde{\xi}_t$ takes finite number of values $N_t,\ \forall t = 1, ..., T$). Scenario process $\widetilde{\xi} = (\widetilde{\xi}_1, ..., \widetilde{\xi}_T)$ is defined on a probability space $(\widetilde{\Omega}, \widetilde{\mathcal{F}}, \widetilde{P})$ (e.g. Pflug and Römisch [19], Pflug [20], Pflug and Pichler [21,22]).

The distance between problems (1) and (2) determines the approximation error. Previously, the distance between the initial problem (1) and its approximation (2) was defined only if both processes ξ and $\widetilde{\xi}$ and both filtrations $\mathcal{F}$ and $\widetilde{\mathcal{F}}$ were defined on the same probability space $(\Omega, \mathcal{F}, P)$, meaning that the approximation error was measured as a filtration distance. The introduction of the concept of the *nested distribution* (see Pflug [20], Pflug and Pichler [21,22]), containing in one mathematical object the scenario values as well as the structural information under which decisions have to be made, allowed to bring the problem to the purely distributional setup. The *nested distance* between these distributions was first introduced by Pflug and Pichler [21,22] and turned out to be a multi-stage generalization of the well-known *Kantorovich-Wasserstein distance* defined for single-stage problems (see Kantorovich [12], Pflug and Pichler [21,22], Villani [33]). Minimizing the nested distance, one can enhance the quality of the approximaion and, hence, the solution accuracy.

Existing methods of the nested distance minimization lack efficiency (see Timonina [30] for more details). Stage-wise minimization of the Kantorovich-Wasserstein distance between measures sitting at each stage of scenario trees (i.e. at each $t = 1, ..., T$) partly improves efficiency, providing the upper bound on the minimal nested distance (see Timonina [29,30] for details). In this article, we make a step towards both accurate and efficient solution methods for multi-stage stochastic optimization programs by the combination of stage-wise methods for distributional quantization with a backtracking solution algorithm on scenario trees, which is based on the dynamic programming principle (e.g. Ermoliev, Marti and Pflug [5], Hanasusanto and Kuhn [9], Timonina [30]) and which is especially suitable for high-dimensional multi-stage stochastic optimization programs.

For our further analysis, it is important, that the objective functions of optimization problems (1) and (2) can be rewritten in a way, which allows to separate current decision x_t or $\widetilde{x}_t$ from all previous decisions at stages $(0, 1, ..., t-1)$. For all stages $t = 1, ..., T$, this can be done by introduction of state variables $s_t = \begin{pmatrix} x^{t-1} \\ \xi^{t-1} \end{pmatrix}$ and $\widetilde{s}_t = \begin{pmatrix} \widetilde{x}^{t-1} \\ \widetilde{\xi}^{t-1} \end{pmatrix}$, which accumulate all available at stage t information on previous decisions and on random component realizations (see Shapiro et. al [28] for model state equations for linear optimization). Therefore, optimization problems (1) and (2) would be written in the following form:

$$\inf_{x \in \mathbb{X}, x \lhd \mathcal{F}} \left\{ \mathbb{E}\left[h_0(s_0, x_0) + \sum_{t=1}^{T} h_t(s_t, x_t, \xi_t) \right] \right\}, \tag{3}$$

$$\inf_{\widetilde{x} \in \mathbb{X}, \widetilde{x} \lhd \widetilde{\mathcal{F}}} \left\{ \mathbb{E}\left[h_0(s_0, \widetilde{x}_0) + \sum_{t=1}^{T} h_t(\widetilde{s}_t, \widetilde{x}_t, \widetilde{\xi}_t) \right] \right\}, \tag{4}$$

where we denote by s_0 the initial, a-priori known, state of the stochastic process (for example, one could assume, that $s_0 = \xi_0 := 0$).

Notice, that state variables s_t and $\widetilde{s}_t$ may grow in time as $s_t = \begin{pmatrix} s_{t-1} \\ x_{t-1} \\ \xi_{t-1} \end{pmatrix}$ and $\widetilde{s}_t = \begin{pmatrix} \widetilde{s}_{t-1} \\ \widetilde{x}_{t-1} \\ \widetilde{\xi}_{t-1} \end{pmatrix}$ $\forall t = 1, ..., T$, describing the accumulation of the information. However, in many practical cases, some information accumulated in time becomes irrelevant, which makes vectors s_t and $\widetilde{s}_t$ not so high-dimensional. In the simplest case, if the stochastic process would have the Markovian structure, the next value of the process would depend on its current value only, being conditionally independent of all the previous values of the stochastic process. Furthermore, some of non-Markovian processes can still be represented as Markov chains by expanding the state space so, that it contains all the relevant information.

The article proceeds as follows: Section 2 describes numerical scenario generation methods focusing on random and optimal quantization of scenario trees. In Section 3 we rewrite the multi-stage stochastic optimization problem (3) in the dynamic programming form and we combine optimal scenario generation with backtracking solution methods of multi-stage stochastic optimization problems, which allows to enhance computational efficiency and to reduce the approximation error. Section 4 is devoted to the Inventory Control Problem, for which the explicit theoretical solution is known and, hence, one can test accuracy and efficiency of proposed numerical algorithms. In Section 5 we apply the algorithms to the problem of risk-management of rare events on the example of flood events in Austria.

2. Scenario tree approximation

Numerical approach for the solution of multi-stage stochastic optimization problems is based on the approximation of stochastic process $\xi = (\xi_1, ..., \xi_T)$ by scenario trees. Each random component ξ_t, $\forall t = 1, ..., T$ is described by a continuous-state distribution function. Denote by $P_t(\xi_t)$ the unconditional probability distribution of the random variable ξ_t and let $P_t(\xi_t|\xi^{t-1})$ be the conditional distribution of the random variable ξ_t given the history ξ^{t-1} up to time $t-1$.

Definition 2.1. A stochastic process $\nu = (\nu_1, ..., \nu_T)$ is called *a tree process* (see Pflug and Pichler [20,22], Römisch [26]), if $\sigma(\nu_1), \sigma(\nu_2), ..., \sigma(\nu_T)$ is a filtration[1].

Notice, that *the history process* $(\xi^1, \xi^2, ..., \xi^T)$ of the stochastic process ξ is a tree process by definition, as soon as $\xi^1 = \xi_1$, $\xi^2 = (\xi_1, \xi_2)$,..., $\xi^T = (\xi_1, \xi_2, ..., \xi_T)$. Moreover, every finitely valued stochastic process $\widetilde{\xi} = (\widetilde{\xi}_1, ..., \widetilde{\xi}_T)$ is representable as a *a finitely valued tree*[2] (Timonina [29]). To solve the approximate problem (1) numerically, one should approximate in the best way possible the stochastic process ξ by a finitely valued tree (Pflug and Pichler [22]).
In order to work with general tree structures, let N_t, $\forall t = 1, ..., T$ be the total number of scenarios at the stage t and let n_t^i $(\forall i = 1, ..., N_{t-1}, \forall t = 2, ..., T)$ be the number of quantizers corresponding to the N_{t-1} conditional distributions sitting at the stage t $(\forall t = 2, ..., T)$. Denote $n_1 = N_1$ and notice, that $N_t = \sum_{i=1}^{N_{t-1}} n_t^i, \forall t = 2, ..., T$.

Definition 2.2. Consider a finitely valued stochastic process $\widetilde{\xi} = (\widetilde{\xi}_1, ..., \widetilde{\xi}_T)$ that is represented by the tree with the same number of successors b_t for each node at the stage t, $\forall t = 1, ..., T$. The vector $b = (b_1, ..., b_T)$ is *a bushiness vector* of the tree (e.g. Timonina [29]). Values $b_1, b_2, ..., b_T$ are called *bushiness factors*.

Example 2.3. Figure 1 shows two trees with different bushiness factors. The tree on the left-hand side is a *binary tree* and, therefore, its bushiness vector is $b = [2\ 2\ 2]$; the tree on the right-hand side is a *ternery tree* and, hence, its bushiness vector is $b = [3\ 3\ 3]$ (bushiness factors for both trees are constant and equal to 2 and 3 correspondingly).

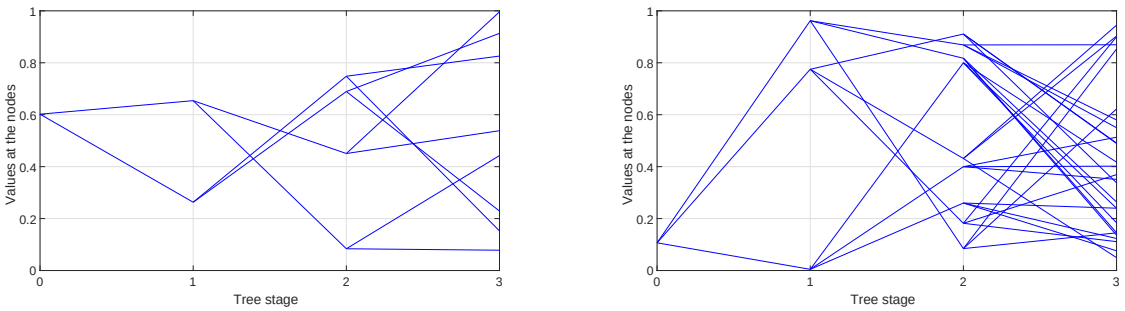


Figure 1. Scenario trees with different bushiness: $b = [2,\ 2,\ 2]$ and $b = [3,\ 3,\ 3]$ correspondingly.

This example demonstrates the univariate case (i.e. $\widetilde{\xi}_t$ is one-dimensional $\forall t$) and, therefore,

[1] Given a measurable space $(\Omega, \mathcal{F})$, a filtration is an increasing sequence of σ-algebras $\{\mathcal{F}_t\}$, $t \geq 0$ with $\mathcal{F}_t \subseteq \mathcal{F}$ such that: $t_1 \leq t_2 \Longrightarrow \mathcal{F}_{t_1} \subseteq \mathcal{F}_{t_2}$. In our case, $\sigma(\nu)$ is the σ-algebra generated by the random variable ν.
[2] The tree, which represents the finitely valued stochastic process $(\widetilde{\xi}_1, ..., \widetilde{\xi}_T)$, is called *a finitely valued tree*.

values sitting on the nodes are shown in the Figure 1. However, in case of multidimensionality of the stochastic process $\widetilde{\xi}$, multidimensional vectors would correspond to each node of the tree and, hence, graphical representation as in Figure 1 would not be possible.
Scenario probabilities are given for each path of the tree and are uniform in this example. $\square$

In order to approximate the stochastic process ξ by a finitely valued tree, one may minimize the distance between the continuous distribution $P_t(\xi_t|\xi^{t-1})$ and a discrete measure sitting, say, on n points, which can be denoted by $\widetilde{P}_t(\widetilde{\xi}_t|\widetilde{\xi}^{t-1}) = \sum_{i=1}^n \widetilde{p}_t^i(\widetilde{\xi}^{t-1})\delta_{z_t^i(\widetilde{\xi}^{t-1})}$, where $z_t^i(\widetilde{\xi}^{t-1})$, $\forall i = 1, ..., n$ are quantizers of the conditional distribution dependent on the history $\widetilde{\xi}^{t-1}$, while $\widetilde{p}_t^i(\widetilde{\xi}^{t-1})$, $\forall i = 1, ..., n$ are the corresponding conditional probabilities. This distance is the well-known *Kantorovich-Wasserstein distance* between measures (Kantorovich [12], Villani [33]):

Definition 2.4. The Kantorovich distance between probability measures P and $\widetilde{P}$ can be defined in the following way:

$$d_{KA}(P, \widetilde{P}) = \inf_{\pi} \Big\{ \int_{\Omega \times \widetilde{\Omega}} d(w, \widetilde{w}) \pi[dw, d\widetilde{w}] \Big\}, \quad \text{subject to } \pi[\cdot \times \widetilde{\Omega}] = P(\cdot) \text{ and } \pi[\Omega \times \cdot] = \widetilde{P}(\cdot), \tag{5}$$

where $d(w, \widetilde{w})$ is the cost function for the transportation of $w \in \Omega$ to $\widetilde{w} \in \widetilde{\Omega}$.

However, this distance neglects the tree structure, taking into account stage-wise available information only, and, therefore, one cannot guarantee that the stage-wise minimization of the Kantorovich-Wasserstein distance would always result in the minimal approximation error between problem (1) and its approximation (2) (see Timonina [30]). To overcome this dilemma, Pflug and Pichler in their work [21] introduced the concept of *nested distributions* $\mathbb{P} \sim (\Omega, \mathcal{F}, P, \xi)$ and $\widetilde{\mathbb{P}} \sim (\widetilde{\Omega}, \widetilde{\mathcal{F}}, \widetilde{P}, \widetilde{\xi})$, which contained information about both processes ξ and $\widetilde{\xi}$ and the stage-wise available information, as well as they defined the *nested distance* (Pflug and Pichler [22]) between problem (1) and its approximation (2) in a purely distributional setup. Further, the nested distance is denoted by $dl(\mathbb{P}, \widetilde{\mathbb{P}})$, where $\mathbb{P}$ refers to the continuous nested distribution of the initial problem (1) and $\widetilde{\mathbb{P}}$ corresponds to the discrete nested distribution, which is the scenario tree approximation of the problem (1).

Definition 2.5. The multi-stage distance (see Pflug and Pichler [21,22]) of order $q \geq 1$ between nested distributions $\mathbb{P}$ and $\widetilde{\mathbb{P}}$ is

$$\begin{aligned} dl_q(\mathbb{P}, \widetilde{\mathbb{P}}) = \inf_{\pi} (& \int d(w, \widetilde{w})^q \pi(dw, d\widetilde{w}))^{\frac{1}{q}}, \\ & \text{subject to } \mathbb{P} \sim (\Omega, \mathcal{F}, P, \xi),\ \widetilde{\mathbb{P}} \sim (\widetilde{\Omega}, \widetilde{\mathcal{F}}, \widetilde{P}, \widetilde{\xi}) \\ & \pi[A \times \widetilde{\Omega} | \mathcal{F}_t \otimes \widetilde{\mathcal{F}_t}](w, \widetilde{w}) = P(A|\mathcal{F}_t)(w),\ (A \in \mathcal{F}_T, 1 \leq t \leq T), \\ & \pi[\Omega \times B | \mathcal{F}_t \otimes \widetilde{\mathcal{F}}_t](w, \widetilde{w}) = \widetilde{P}(B|\widetilde{\mathcal{F}}_t)(\widetilde{w}),\ (B \in \widetilde{\mathcal{F}}_T, 1 \leq t \leq T). \end{aligned} \tag{6}$$

We denote by $dl(\mathbb{P}, \widetilde{\mathbb{P}})$ the nested distance of order $q = 1$, i.e. $dl_1(\mathbb{P}, \widetilde{\mathbb{P}}) = dl(\mathbb{P}, \widetilde{\mathbb{P}})$.

Under the assumption of Lipschitz-continuity of the loss/profit function $H(x, \xi)$ with the Lipschitz constant L_1, the nested distance (6) establishes an upper bound for the approximation error between problems (1) and (2) (Pflug and Pichler [22]), which means $|v(\mathbb{P}) - v(\widetilde{\mathbb{P}})| \leq L_1 dl(\mathbb{P}, \widetilde{\mathbb{P}})$,

where value functions $v(\mathbb{P})$ and $v(\widetilde{\mathbb{P}})$ correspond to optimal solutions of the multi-stage problems (1) and (2). Hence, the nested distribution $\widetilde{\mathbb{P}}$ constructed in such a way, that the nested distance $dl(\mathbb{P}, \widetilde{\mathbb{P}})$ is minimized, leads to a fine approximation of the optimization problem (1). However, due to the complexity of numerical minimization of the nested distance, we use an upper bound introduced in the work of Pflug and Pichler [21]:

$$dl(\mathbb{P}, \widetilde{\mathbb{P}}) \leq \sum_{t=1}^{T} d_{KA}(P_t, \widetilde{P}_t) \prod_{s=t+1}^{T} (L_s + 1), \tag{7}$$

where P_t and $\widetilde{P}_t$ are marginal distributions corresponding to the stage t; L_s, $\forall s = 2, ..., T$ are some constants.

We claim that $\exists \widetilde{P}_t = \sum_{i=1}^{N} \widetilde{p}_t^i \delta_{z_t^i}$ $\forall t = 1, ..., T$ sitting on N discrete points, such that $d_{KA}(P_t, \widetilde{P}_t^N) \leq cN^{-\frac{1}{r_1}}$ if conditions of the Zador-Gersho formula are satisfied (e.g. Graf and Luschgy [8], Pflug and Pichler [21], Timonina [29]). In this case, one derives the bound $|v(\mathbb{P}) - v(\widetilde{\mathbb{P}})| \leq cL_1 N^{-\frac{1}{r_1}} \sum_{t=1}^{T} \prod_{s=t+1}^{T} (L_s + 1)$ from (7), which converges to zero for $N \to \infty$.

Therefore, the concept of tree bushiness allows to obtain the convergence of the nested distance between initial stochastic process and approximate scenario tree to zero, when the bushiness of the scenario tree increases and when scenarios are generated in such a way, that the stage-wise Kantorovich-Wasserstein distance is converging to zero (see Pflug and Pichler [21,22], Timonina [29] for more details).

The speed of the nested distance convergence depends on the method of stage-wise scenario generation. In this article, we focus on the optimal scenario quantization, which calculates probabilities based on the estimated optimal locations, minimizing the Kantorovich-Wasserstein distance at each stage of the scenario tree (e.g. Fort and Pagés [7], Pflug and Römisch [19], Römisch [26], Villani [33], Timonina [29]). This allows to enhance accuracy of the solution of multi-stage stochastic optimization problems with respect to the well-known Monte-Carlo (random) scenario generation.

The overall procedure of the conditional optimal quantization on a tree structure is described below (see Pflug and Römisch [19], Römisch [26], Timonina [29,30] for the details):

Optimal quantization finds n optimal supporting points z_t^i, $i = 1, ..., n$ of conditional distribution $P_t(\xi_t|\xi^{t-1})$, $\forall t = 2, ..., T$ by minimization (over z_t^i, $i = 1, ..., n$) of the distance:

$$\mathcal{D}\left(z_t^1, ..., z_t^n\right) = \int \min_i d\left(u, z_t^i\right) P_t(du|\xi^{t-1}), \tag{8}$$

where $d\left(u, z_t^i\right)$ is the Euclidean distance between points u and z_t^i. At stage $t = 1$, optimal quantization is based on the unconditional distribution $P_1(\xi_1)$. Notice, that there are N_{t-1} conditional distributions at each futher stage of the tree (i.e. $\forall t = 2, ..., T$).

Given the locations of the supporting points z_t^i, their probabilities $\widetilde{p}_t^i$ are calculated by the minimization of the Kantorovich-Wasserstein distance between the measure $P_t(\xi_t|\xi^{t-1})$ and its discrete approximation $\sum_{i=1}^{n} \widetilde{p}_t^i \delta_{z_t^i}$:

$$\min_{\widetilde{p}_t^i, \forall i} d_{KA}\left(P_t(\xi_t|\xi^{t-1}), \sum_{i=1}^{n} \widetilde{p}_t^i \delta_{z_t^i}\right). \tag{9}$$

Figure 2 demonstrates optimal quantizers and their corresponding probabilities for the 3-stage stochastic process $(\xi_1, \xi_2, \xi_{T=3})$, which follows the multivariate Gaussian distribution with mean vector $\mu = (\mu_1, \mu_2, \mu_3)$ and non-singular variance-covariance matrix $C = (c_{s,t})_{t=1,2,3;s=1,2,3}$. Importantly, every conditional distribution of ξ_t given the history ξ^{t-1} is also a normal distribution with known mean and variance (see Lipster and Shiryayev [14] for the details on the form of the conditional distributions).

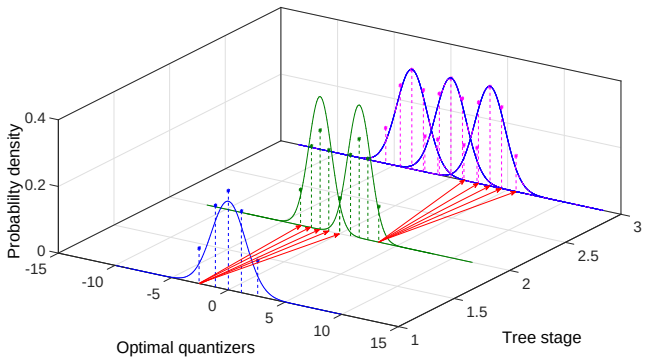


Figure 2. Optimal quantization of the scenario tree with Gaussian random variables $\xi_t,\ t = 1, ..., T$.

Further, we combine the backtracking dynamic programming method with the optimal scenario quantization to compromise between accuracy and efficiency of the numerical solution.

3. Dynamic programming

The idea of the dynamic programming method goes back to pioneering papers of Bellman [1], Bertsekas [2] and Dreyfus [4], who expressed the optimal policy in terms of an optimization problem with iteratively evolving value function (the optimal *cost-to-go* function). These foundational works gave us the theoretical framework for rewriting time separable multi-stage stochastic optimization problems in the dynamic form. More recent works of Bertsekas [3], Keshavarz [13], Hanasusanto and Kuhn [9], Powell [24] are built on the fact that the evaluation of optimal cost-to-go functions, involving multivariate conditional expectations, is a computationally complex procedure and on the necessity to develop numerically efficient algorithms for multi-stage stochastic optimization. We follow this path and propose an accurate and efficient algorithm for dynamic solution of multi-stage problems using optimal quantizers.

In line with formulations (3) and (4), let the *endogenous state* $s_t \in \mathbb{R}^{r_3}$ capture all the decision-dependent information about the past. For simplicity, assume that the dimension r_3 of the endogenous variable s_t does not change in time and that the variable obeys the recursion $s_{t+1} = g_t(s_t, x_t, \xi_{t+1})\ \forall t = 0, ..., T-1$ with the given initial state s_0. Clearly, for endogenous variables with time-varying dimensions this assumption can be replaced with a particular set of constraints.

Now, the optimization problem (3) can be subdivided into multiple single-stage problems, setting aside all future decisions according to the Bellman's principle of optimality.

At the stage $t = T$, one solves the following deterministic optimization problem:

$$V_T(s_T, \xi_T) := \min_{x_T} h_T(s_T, x_T, \xi_T), \quad \text{subject to } x_T \in \mathbb{X}_T,\ x_T \triangleleft \mathcal{F}_T, \tag{10}$$

where all information about the past is encoded into the variable s_T.
At stages $\forall t = T-1, ..., 1$ the following holds in line with formulations (3) and (4):

$$V_t(s_t, \xi_t) := \min_{x_t} \Big\{ h_t(s_t, x_t, \xi_t) + \mathbb{E}\Big[V_{t+1}(s_{t+1}, \xi_{t+1}) \mid \xi_t \Big] \Big\}, \tag{11}$$
$$\text{subject to } x_t \in \mathbb{X}_t,\ x_t \triangleleft \mathcal{F}_t,$$
$$s_{t+1} = g_t(s_t, x_t, \xi_{t+1}).$$

At the stage $t = 0$, the optimal solution of the optimization problem (11) coincides with the optimal solution of the problem (3) and is equal to:

$$V_0(s_0) := \min_{x_0} \Big\{ h_0(s_0, x_0) + \mathbb{E}\Big[V_1(s_1, \xi_1) \Big] \Big\}, \tag{12}$$
$$\text{subject to } x_0 \in \mathbb{X}_0,\ x_0 \triangleleft \mathcal{F}_0,$$
$$s_1 = g_0(s_0, x_0, \xi_1).$$

Notice, that V_0 is deterministic, as there is no random variable realization at the stage $t = 0$.
Problems (10), (11) and (12) can be solved by algorithms proposed in the works of Bertsekas [3], Hanasusanto and Kuhn [9], Keshavarz and Boyd [13], Powell [24]. For example, employing the algorithm proposed in the work of Hanasusanto and Kuhn [9], one uses historical data paths on the endogenous and exogenous variables s_t and ξ_t. Further, in order to evaluate the optimal values at stages $t = T-1, ..., 0$, one uses piecewise linear or quadratical interpolation of $V_{t+1}(s_{t+1}, \xi_{t+1})$ at historical data points and one estimates the conditional expectation $\mathbb{E}[V_{t+1}(s_{t+1}, \xi_{t+1}) \mid \xi_t]$ by the use of the Nadaraya-Watson kernel regression for conditional probabilities [17,34]. This method allows to enhance efficiency of the computation due to the fact that the cost-to-go function is evaluated at historical data points only, which is especially useful for the robust reformulation proposed in the second part of the work of Hanasusanto and Kuhn [9]. However, for the optimization problems of the type (3) and (4) such method may lack accuracy, as it does not consider full information set available at stages $t = 0, ..., T$, taking into account only the part incorporated into conditional probabilities. This may result in underestimation of the optimal value, especially in case of stochastic processes which follow heavy-tailed distribution functions poorly represented by historical data paths.
To avoid this problem, we i) represent the exogenous variable ξ_t by optimal supporting points minimizing the distance function (8) between the conditional distribution $P_t(\xi_t|\xi_{t-1})$ and its discrete approximation. We ii) compute the conditional probabilities at the stage t via the minimization of the Kantorovich-Wasserstein distance (9). The solution of optimization problems (3) and (4) is obtained via dynamic programming (10), (11) and (12), for which we propose iii) an accurate and efficient numerical algorithm based on the optimally quantized scenario trees[3].

[3] The finite scenario tree, which node values at the stage t are the optimal supporting points minimizing the distance function (8) and which corresponding node probabilities are the optimal probabilities satisfying (9), is called the *optimally quantized scenario tree*.

The method proceeds as follows:

- Step 1 - quantize conditional distributions for the exogenous variable $\xi_t,\ \forall t = 1, ..., T$;
- Step 2 - use a grid for the endogenous variable $s_t,\ \forall t = 1, ..., T$;
- Step 3 - solve the dynamic program (10) at the stage T;
- Step 4 - solve the dynamic program (11) at stages $t = 1, ..., T-1$;
- Step 5 - solve the dynamic program (12) at the root of the scenario tree.

Further, we discuss each of these steps in more details:

Step 1 - Scenario tree approximation for the exogenous variable: Fix the scenario tree structure and quantize conditional distributions optimally in the sense of minimal distances (8) and (9). One acquires optimal supporting points sitting at the nodes of the tree and the corresponding conditional probabilities.

Recall, that in order to get optimal quantizers at stages $t = 2, ..., T$ of the tree, we compute N_{t-1} optimal sets of points denoted by $\left\{\{\hat{\xi}_t^1, \hat{\xi}_t^2, ..., \hat{\xi}_t^{n_t^1}\}_{t=2}^T, \{\hat{\xi}_t^{n_t^1+1}, \hat{\xi}_t^{n_t^1+2}, ..., \hat{\xi}_t^{n_t^1+n_t^2}\}_{t=2}^T, ...\right\}$ with the corresponding conditional probabilities $\left\{\{\hat{p}_t^1, ..., \hat{p}_t^{n_t^1}\}_{t=2}^T, \{\hat{p}_t^{n_t^1+1}, ..., \hat{p}_t^{n_t^1+n_t^2}\}_{t=2}^T, ...\right\}$, which minimize the Kantorovich-Wasserstein distance (9).

Step 2 - Grid for the endogenous variable: Use a grid for the endogenous variable $s_t, \forall t = 1, ..., T$. Let us denote points in the grid as $\{\hat{s}_t^k\}_{t=1}^T,\ \forall k = 1, ..., K$. Differently, one can use random trajectories for the endogenous state variable or, as it is in the work of Hanasusanto and Kuhn [9], one can employ the historical data paths for $s_t,\ \forall t = 1, ..., T$.

Step 3 - Dynamic programming at the stage T: Start with the stage $t = T$ and solve the optimization problem (10) approximately using the scenario tree discretization at each node of the stage $t = T$, as well as the grid for the endogenous variable at the stage $t = T$.

Let us denote by $\hat{V}_T(\hat{s}_T^k, \hat{\xi}_T^i)\ \forall k = 1, ..., K,\ i = 1, ..., N_T$ the approximate optimal value of the optimization problem (10), evaluated at the point $\hat{s}_T^k$ of the grid and at the node $\hat{\xi}_T^i$ of the scenario tree. We estimate the value of $\hat{V}_T(\hat{s}_T^k, \hat{\xi}_T^i)$ via the solution of the following optimization problem $\forall k = 1, ..., K,\ \forall i = 1, ..., N_T$:

$$\begin{aligned} \hat{V}_T(\hat{s}_T^k, \hat{\xi}_T^i) = \min_{x_T} & h_T(\hat{s}_T^k, x_T, \hat{\xi}_T^i), \\ & \text{subject to } x_T \in \mathbb{X}_T,\ x_T \triangleleft \mathcal{F}_T, \end{aligned} \tag{13}$$

Step 4 - Dynamic programming at the stage t: Suppose, that we could solve the dynamic optimization problem (10) or (11) at any stage $t+1$ and that we would like to receive the optimal solution of the dynamic problem at the stage t. For this, let us denote by $\hat{V}_t(\hat{s}_t^k, \hat{\xi}_t^i)$ $\forall k = 1, ..., K,\ i = 1, ..., N_t$ the approximate optimal value of the optimization problem (11), evaluated at the point $\hat{s}_t^k$ of the grid and at the node $\hat{\xi}_t^i$ of the scenario tree. We estimate the value of $\hat{V}_t(\hat{s}_t^k, \hat{\xi}_t^i)$ via the solution of the following problem $\forall k = 1, ..., K,\ \forall i = 1, ..., N_t$:

$$\begin{aligned} \hat{V}_t(\hat{s}_t^k, \hat{\xi}_t^i) = \min_{x_t} & \left\{ h_t(\hat{s}_t^k, x_t, \hat{\xi}_t^i) + \sum_{j \in \mathcal{L}_{t+1}^i} [\hat{V}_{t+1}(s_{t+1}^j, \hat{\xi}_{t+1}^j)] \hat{p}_{t+1}^j \right\}, \\ & \text{subject to } x_t \in \mathbb{X}_t,\ x_t \triangleleft \mathcal{F}_t, \\ & s_{t+1}^j = g_t(\hat{s}_t^k, x_t, \hat{\xi}_{t+1}^j),\ \forall j \in \mathcal{L}_{t+1}^i, \end{aligned} \tag{14}$$

where $\mathcal{L}^i_{t+1}$ is the set of node indices of the stage $t+1$ outgoing from the node with index i of the stage t in the scenario tree (these indices are used to define the chosen subtree and, therefore, they help to preserve the information structure, see Figure 3).

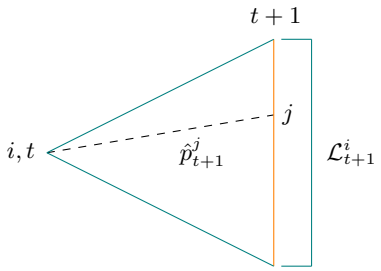


Figure 3. Subtree outgoing from the node i at the stage t of the scenario tree.

Step 5 - Dynamic programming at the root: Analogically to stages $t = T-1, ..., 1$, we evaluate the optimal value $\hat{V}_0(s_0)$ via the solution of the following optimization problem:

$$\hat{V}_0(s_0) := \min_{x_0} \left\{ h_0(s_0, x_0) + \sum_{j=1}^{N_1} [\hat{V}_1(s^j_1, \hat{\xi}^j_1)] \hat{p}^j_1 \right\}, \tag{15}$$
$$\text{subject to } x_0 \in \mathbb{X}_0, \; x_0 \triangleleft \mathcal{F}_0,$$
$$s^j_1 = g_0(s_0, x_0, \hat{\xi}^j_1), \; \forall j = 1, ..., N_1.$$

Notice, that there is only one possible subtree with N_1 nodes outgoing from the root of the tree (i.e. at the stage $t = 0$). This is due to the fact, that the distribution sitting at the stage $t = 1$ is unconditional.

Importantly, in order to solve optimization problems (14) and (15) $\forall t = T-1, ..., 0$, one needs to evaluate the optimal value $\hat{V}_{t+1}(s^j_{t+1}, \hat{\xi}^j_{t+1})$ at the point s^j_{t+1}, which does not necessarily coincide with grid points $\{\hat{s}^k_{t+1}\}$, $\forall k = 1, ..., K$. For this, we approximate the function $\hat{V}_{t+1}(s_{t+1}, \hat{\xi}^j_{t+1})$ continuously in s_{t+1} under assumptions about convexity and monotonicity of functions $h_t(s_t, x_t, \xi_t)$, $g_t(s_t, x_t, \xi_{t+1})$ and $V_{t+1}(s_{t+1}, \xi_{t+1})$ which are discussed in details in Appendix (see Theorems 6.3 and 6.4).

If convexity and monotonicity conditions of Theorems 6.3 or 6.4 hold for functions $h_t(s_t, x_t, \xi_t)$, $g_t(s_t, x_t, \xi_{t+1})$ and $V_{t+1}(s_{t+1}, \xi_{t+1})$ in the dynamic program (14), we can guarantee that the function $V_t(s_t, \xi_t)$ is also convex and monotone. Moreover, these properties stay recursive $\forall t = T, ..., 0$, due to the convexity and monotonicity results of Theorems 6.3 and 6.4.

For dynamic programs (13), (14) and (15), Theorems 6.3 and 6.4 give the possibility to approximate the optimal value function $V_{t+1}(s_{t+1}, \xi_{t+1})$ by a convex and monotone interpolation in s_{t+1} prior to the solution of the corresponding optimization problem and, therefore, to evaluate the optimal value $\hat{V}_{t+1}(s^j_{t+1}, \hat{\xi}^j_{t+1})$ at any point s^j_{t+1}, which does not necessarily coincide with grid points $\{\hat{s}^k_{t+1}\}$, $\forall k = 1, ..., K$.

Further, we use the quadratic approximation of the function $\hat{V}_t(s_t, \hat{\xi}^i_t)$, $\forall i$:

$$\hat{V}_t(s_t, \hat{\xi}^i_t) = {s_t}^T A_i s_t + 2 b_i^T s_t + c_i, \tag{16}$$

where A_i, b_i and c_i are to be estimated by fitting convex and monotone function $\hat{V}_t(s_t, \hat{\xi}^i_t)$ to the points $\hat{V}_t(\hat{s}^k_t, \hat{\xi}^i_t)$, $\forall k = 1, ..., K$.

If conditions of Theorem 6.3 hold, the estimates are obtained via the sum of squares minimization under the constraint, implying monotonicity in the sense $s_1 \succeq s_2 \Rightarrow \hat{V}_t(s_1, \hat{\xi}_t^i) \geq \hat{V}_t(s_2, \hat{\xi}_t^i)$:

$$\frac{\partial \hat{V}_t(s_t, \hat{\xi}_t^i)}{\partial (s_t)_m} \geq 0, \ \forall m = 1, ..., r_3 \Longleftrightarrow A_i s_t + b_i \succeq 0, \tag{17}$$

where $(s_t)_m$ is the m-th coordinate of the vector s_t ($s_t \in \mathbb{R}^{r_3}$).
Differently, if conditions of Theorem 6.4 hold, the opposite constraint should be used, i.e.:

$$\frac{\partial \hat{V}_t(s_t, \hat{\xi}_t^i)}{\partial (s_t)_m} \leq 0, \ \forall m = 1, ..., r_3 \Longleftrightarrow A_i s_t + b_i \preceq 0, \tag{18}$$

which implies monotonicity in the sense $s_1 \succeq s_2 \Rightarrow \hat{V}_t(s_1, \hat{\xi}_t^i) \leq \hat{V}_t(s_2, \hat{\xi}_t^i)$.
Importantly, one does not require monotonicity conditions (17) or (18), if dealing with linear programming (i.e. if functions $h_t(s_t, x_t, \xi_t)$, $g_t(s_t, x_t, \xi_{t+1})$ and $V_{t+1}(s_{t+1}, \xi_{t+1})$ are linear in s_t and x_t). Indeed, linearity conditions are a special case of requirements of Lemma 6.2 and they are recursively preserved in the dynamic programming (see Corollary 6.5).
The quadratic function (16) can be computed efficiently by solving the following semidefinite program $\forall i = 1, ..., N_t$:

$$\begin{aligned} &\min_{A_i, b_i, c_i} \left\{ \sum_{k=1}^{K} \left[(\hat{s}_t^k)^T A_i \hat{s}_t^k + 2 b_i^T \hat{s}_t^k + c_i - \hat{V}_t(\hat{s}_t^k, \hat{\xi}_t^i) \right]^2 \right\}, \\ &\text{subject to} \\ &A_i \in \mathbb{S}^{r_3}, \ b_i \in \mathbb{R}^{r_3}, c_i \in \mathbb{R} \\ &z^T A_i z \geq 0 \ \forall z \in \mathbb{R}^{r_3} \ , A_i \hat{s}_t^k + b_i \succeq 0, \ \forall k = 1, ..., K, \end{aligned} \tag{19}$$

where $\mathbb{S}^{r_3}$ is the set of symmetric matrices and where we use constraint (17) as an example. In case conditions of the Theorem 6.4 are satisfied, the constraint is replaced by the opposite one (18). Furthermore, in case of linearity of the program (i.e. in case conditions of Corollary 6.5 are satisfied), we implement the linear interpolation of value function at the next stage, i.e.:

$$\begin{aligned} &\min_{b_i, c_i} \left\{ \sum_{k=1}^{K} \left[b_i^T \hat{s}_t^k + c_i - \hat{V}_t(\hat{s}_t^k, \hat{\xi}_t^i) \right]^2 \right\}, \\ &\text{subject to } b_i \in \mathbb{R}^{r_3}, c_i \in \mathbb{R} \end{aligned}$$

Algorithm 1 describes the overall dynamic optimization procedure.

Algorithm 1 Dynamic programming with optimal quantizers.

Grid $\{\hat{s}_t^k\}_{t=1}^{T-1}$, $\forall k = 1, ..., K$;
Quantize the scenario tree by finding $\{\hat{\xi}_t^i\}_{t=1}^{T}$ and $\{\hat{p}_t^i\}_{t=1}^{T}$, $\forall i = 1, ..., N_t$ minimizing (8), (9);
for $t = T-1, ..., 0$ **do**
  if $t == T-1$ **then**
    Compute $\hat{V}_{T-1}(\hat{s}_{T-1}^k, \hat{\xi}_{T-1}^i)$, $\forall i, k$ by solving the optimization problem (13);
  else if $0 < t < T-1$ **then**
    Define current node $(\hat{s}_t^k, \hat{\xi}_t^i)$ and evaluate $s_{t+1}^j = g_t(\hat{s}_t^k, x_t, \hat{\xi}_{t+1}^j)$, $\forall j \in \mathcal{L}_{t+1}^i$;
    Interpolate $\hat{V}_{t+1}(s_{t+1}^j, \hat{\xi}_{t+1}^j)$ by quadratic approximation (16) under the monotonicity constraint;
    Solve the optimization problem (14) using the quadratic interpolation (16) at the stage $t+1$;
  else if $t == 0$ **then**
    Solve the optimization problem (15) using the quadratic interpolation (16) at the stage $t = 1$.
  end if
end for

4. Accuracy and efficiency test: an inventory control problem

To compare accuracy and efficiency of numerical algorithms designed for the solution of multi-stage stochastic optimization problems, we employ *the inventory control problem* (see Pflug and Römisch [19], Shapiro et al. [28], Timonina [29]), as the multi-stage stochastic optimization problem, for which the explicit theoretical solution is known.

Considering the univariate case of the inventory control problem, we suppose, that at time t a company needs to decide about order quantity $x_t \in \mathbb{R}_{\{0,+\}}$ for a certain product to satisfy random future demand $\xi_{t+1} \in \mathbb{R}_{\{0,+\}}$, which continuous probability distribution $F_{t+1}(d) = P(\xi_{t+1} \leq d)$ is known explicitly but which realization has not been observed yet. Let T be the number of planning periods. The cost for ordering one piece of the good may change over time and is denoted by $c_{t-1}\ \forall t = 1, ..., T$. Unsold goods may be stored in the inventory with a storage loss $1-l_t\ \forall t = 1, ..., T$. If the demand exceeds the inventory plus the newly arriving order, the demand has to be fulfilled by rapid orders (delivered immediately), for the price of $u_t > c_{t-1}\ \forall t = 1, ..., T$ per piece. The selling price of the good is s_t ($s_t > c_{t-1}\ \forall t = 1, ..., T$).

The optimization problem aims to maximize the following expected cumulative profit:

$$\begin{aligned} &\max_{x_{t-1}\geq 0\ \forall t} \mathbb{E}\left(\sum_{t=1}^{T}[-c_{t-1}x_{t-1} - u_t M_t] + l_T K_T\right), \\ &\text{subject to } x_t \lhd \mathcal{F}_t\ \forall t = 0, ..., T-1, \\ &l_{t-1}K_{t-1} + x_{t-1} - \xi_t = K_t - M_t\ \forall t = 1, ..., T, \end{aligned} \tag{20}$$

where K_t is an uncertain inventory volume right after all sales have been effectuated at time t with K_0 and l_0 set to be zero (i.e. $K_0 := 0$ and $l_0 := 0$), while M_t can be understood as an uncertain shortage at time t.

The optimal solution x^* can be computed explicitly (see Shapiro et al. [28], Pflug and Römisch [19] for more details) and is equal to $x^*_{t-1} = F_t^{-1}\left(\frac{u_t - c_{t-1}}{u_t - l_t}\right) - l_{t-1}K_{t-1},\ \forall t = 1, ..., T$, where $F_t(d) = P_t(\xi_t \leq d)$ is the probability distibution of the random demand ξ_t at any stage t.

The optimization problem (20) can be rewritten in the dynamic programming form.

At the stage $t = T - 1$, one solves the following expectation-maximization problem:

$$\begin{aligned} V_{T-1}(K_{T-1}, M_{T-1}, \xi_{T-1}) := &\max_{x_{T-1}\geq 0}\left\{ - c_{T-1}x_{T-1} + \mathbb{E}\Big[- u_T M_T + l_T K_T \Big| \xi_{T-1}\Big]\right\}, \\ &\text{subject to } l_{T-1}K_{T-1} + x_{T-1} - \xi_T = K_T - M_T, \end{aligned} \tag{21}$$

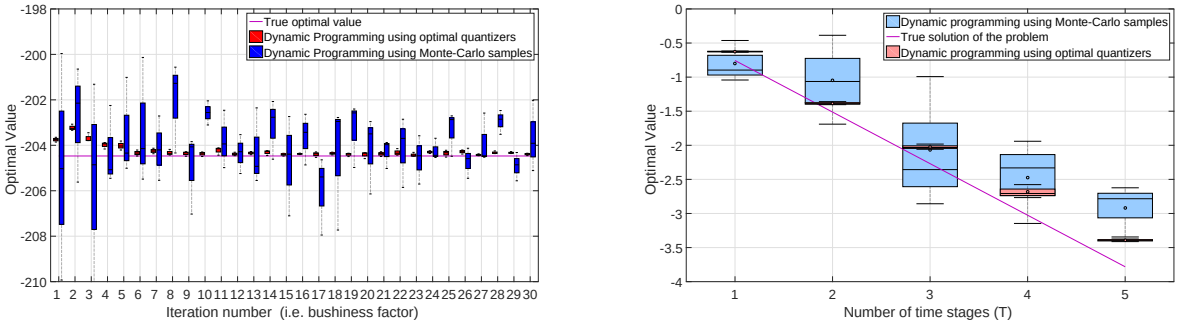
while at any stage $t < T - 1$ the company faces the following decision-making problem:

$$\begin{aligned} V_t(K_t, M_t, \xi_t) := &\max_{x_t\geq 0}\left\{ - c_t x_t + \mathbb{E}\Big[- u_{t+1}M_{t+1} + V_{t+1}(K_{t+1}, M_{t+1}, \xi_{t+1})\Big|\xi_t\Big]\right\}, \\ &\text{subject to } l_t K_t + x_t - \xi_{t+1} = K_{t+1} - M_{t+1},\ \forall t = 0, ..., T-2. \end{aligned} \tag{22}$$

As optimization problems (21) and (22) are linear, the conditions of Corollary 6.5 are satisfied. Therefore, we solve optimization problems (21) and (22) by the use of the Algorithm 1 with linear interpolation of the function $V_t(K_t, M_t, \xi_t)$ in K_t and M_t.

Assuming, that the uncertain multi-period demand $\xi = (\xi_1,\ \xi_2, ..., \xi_T)$ follows the multivariate normal distribution with mean vector $\mu = (\mu_1, \ldots, \mu_T)$ and non-singular variance-covariance matrix $C = (c_{s,t})_{t=1,...,T;s=1,...,T}$, we easily generate future demands at stages $t = 1, ..., T$ using stage-wise optimal or random quantization (see Lipster and Shiryayev [14]).

Figures 4 a. and b. demonstrate the optimal value convergence for the T-stage Inventory Control Problem (20), compared to the true theoretical solution of the problem.



a. Case: 2 stages. b. Case: T stages, $\mu_1 = ... = \mu_T$.

Figure 4. Accuracy comparison of numerical algorithms for solution of multi-stage stochastic optimization problems with unique product.

Figure 4 b. shows, how the optimal value of the Inventory Control Problem (20) changes, when the number of time stages increases. The dependency between the optimal value and the number of time stages is linear in case of Gaussian distribution with $\mu_1 = ... = \mu_T$.

The inventory control problem can be generalized for the case of J goods. In the multi-good multi-stage case, the optimization problem is to maximize the expected cumulative profit:

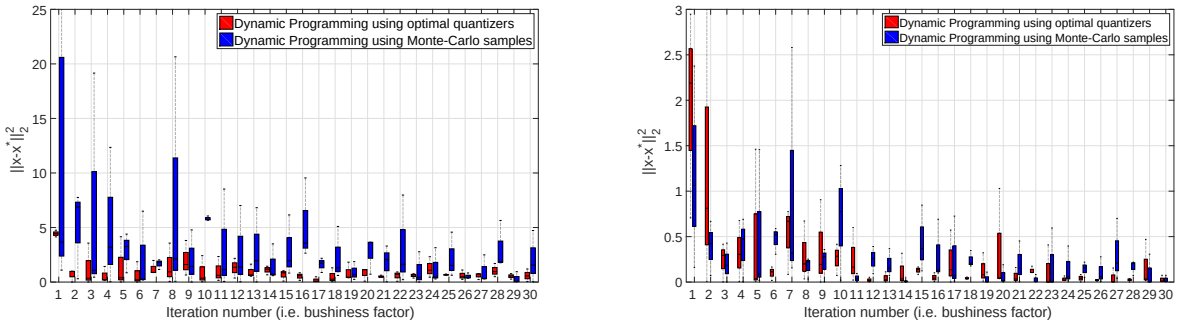
$$\max_{x_{jt}\geq 0,\ \forall j,t} \mathbb{E}\left(\sum_{t=1}^{T}\sum_{j=1}^{J}[-c_{jt-1}x_{jt-1} - u_{jt}M_{jt}] + \sum_{j=1}^{J} l_{jT}K_{jT}\right), \tag{23}$$

$$\text{subject to } x_t \triangleleft \mathcal{F}_t \ \forall t = 0, ..., T-1,$$

$$l_{jt-1}K_{jt-1} + x_{jt-1} - \xi_{jt} = K_{jt} - M_{jt},\ \forall t = 1, ..., T,\ \forall j = 1, ..., J,$$

where index j corresponds to the good $j = 1, ..., J$ and index t corresponds to the time $t = 1, ..., T$, while all notations stay as before.

The optimal solution x^*_{jt} can be computed explicitly (Shapiro et al. [28]) and is equal to $x^*_{jt-1} = F^{-1}_{jt}\left(\frac{u_{jt}-c_{jt-1}}{u_{jt}-l_{jt}}\right) - l_{jt-1}K_{jt-1},\ \forall j, t$, where $F_{jt}(d) = P_{jt}(\xi_{jt} \leq d)$ is the marginal probability distibution of the random demand ξ_{jt} for the product j at the stage t.



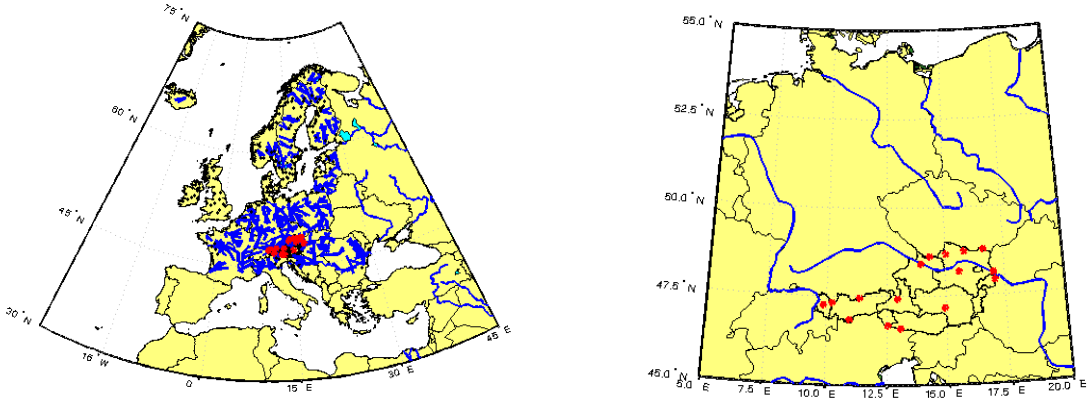
a. Case: 1 product; 2 stages. b. Case: 3 products; 2 stages.

Figure 5. Accuracy comparison of numerical algorithms for solution of multi-stage stochastic optimization problems with different number of products.

Figures 5 a. and b. demonstrate optimal decision convergence for the 2-stage Inventory Control Problem (23) with 1 and 3 products correspondingly, compared to the true theoretical solution of the problem (in the sense of $||x - x^*||_2^2$ convergence). Accuracy of the Algorithm 1 with optimal quantizers is higher in probability than the accuracy obtained via the Monte-Carlo sampling.

5. Risk-management of flood events in Austria

The research, devoted to finding optimal strategies for risk-management of catastrophic events, is motivated by different needs of people on international, national and local policy levels. We consider flood events in Europe as the example of rare but damaging events.
Figure 6 shows European and Austrian river basins subject to flood risk. In the Figure 6 a., one can observe the structure of rivers in Europe, which is used in order to account for regional interdependencies in risk via the *structured coupling approach* (Timonina et al. [31]).



a. Europe. b. Austria.

Figure 6. River basins in Europe and in Austria subject to flood risk.

In order to answer the question about the flood risk in a region, it is necessary to estimate the probability loss distribution, giving information on probabilities of rare events (10-, 50-, 100-year events, etc.) and the amount of losses in case of these events. According to Timonina et al. [31], the risk of flood events can be estimated using structured copulas, avoiding an underestimation of risks. Employing this approach, we estimate national-scale probability loss distributions for Austria for 2030 via Flipped Clayton, Gumbel and Frank copulas (see Timonina et al. [31]):

Year-events	No loss (prob.)	5	10	20	50	100	250	500	1000
Fl. Clayton	P=0.68	0.000	0.030	0.855	2.926	8.017	13.810	16.756	17.761
Gumbel	P=0.67	0.000	0.067	0.847	2.953	7.975	13.107	16.340	17.736
Frank	P=0.63	0.000	0.395	1.147	2.894	7.697	10.213	10.689	10.689

Table 1
Total losses in Austria for 2030 in EUR bln.

Further, we use continuous Fréchet distribution fit to the estimates in the Table 1. This allows to avoid underestimation of losses after low-probability events and is convenient for numerical purposes of scenario generation.

We assume that variables $\xi_t,\ \forall t = 1, ..., T$ are i.i.d. random variables distributed according to the continuous Fréchet distribution. This assumption is valid, when (i) the analysed (e.g. planning) period is not longer than several years and, therefore, the climate change can be neglected, and (ii) when damages imposed by a disaster event do not increase the risk (e.g. past losses do not influence the current loss probability). These conditions are assumed to be valid for Austria according to the Austrian climate change analysis in the Figure 7.

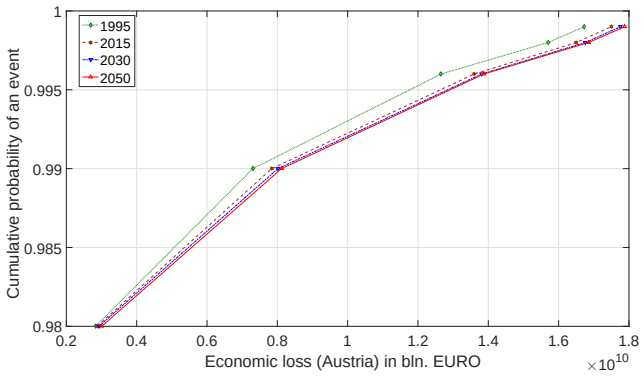
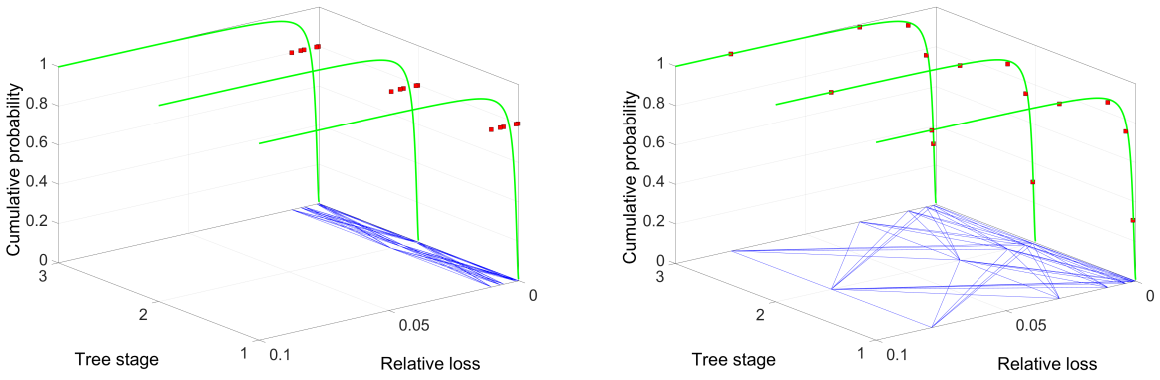


Figure 7. Climate change influences Austrian flood risk.

In the Figure 8, we generate flood losses randomly (Figure 8 a.) and optimally (Figure 8 b.) based on the Fréchet distribution fit. Notice, that the Monte-Carlo method (Figure 8 a.) does not take heavy tails of the distribution into account, especially when historical data are resampled. This problem may lead to over-optimistic decisions for risk-management.



a. Monte-Carlo sampling. b. Optimal quantization.

Figure 8. Quantization of the Fréchet distribution.

Further, we formulate the multi-stage stochastic optimization problem in mathematical terms. For this, consider a government, which may lose a part of its capital S_t at any future times $t = 1, ..., T$ because of random natural hazard events with uncertain relative economic loss ξ_t. As a result of this loss, the country would face a drop in GDP in the end of the year.

Suppose, that under the uncertainty about the amount of loss the decision-maker decides how much of the available budget B_{t-1} to spend on investment x_{t-1} (which influences the capital formation) and on government consumption c_{t-1} in absolute terms $\forall t = 1, ..., T$. Suppose also, that an insurance scheme against natural disasters is available for this country and that the decision about the amount of insurance $z_{t-1},\ \forall t = 1, ..., T$, which is going to be periodically paid from the budget, needs to be made. The insurance premium depends on the expected value of the relative loss $\xi_t,\ \forall t = 1, ..., T$ and is equal to $\pi(\mathbb{E}(\xi_t)) = (1 + l)\mathbb{E}(\xi_t)$, where l is a constant insurance load.

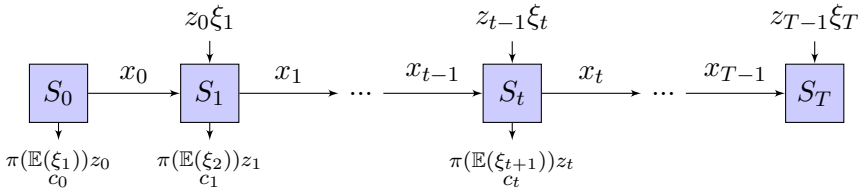
Depending on the goals of the government, different decision-making problems can be stated in terms of multi-stage stochastic optimization programs. We consider a multi-stage model, which describes the decision-making problem in terms of relative capital loss $\xi_t,\ \forall t = 1, ..., T$, while GDP is being modeled in line with the classical Cobb-Douglas production function with constant productivity and maximal weight of capital rather than labor. Available budget is a constant part of GDP in this setting. Hence, $B_t = \alpha S_t$, where S_t is the governmental capital at stages $t = 0, ..., T$ and α is a constant term from the interval $[0,\ 1]$.

Suppose, that the objective of the decision-maker is to maximize the expectation about the weighted government consumption, which aim is to represent overall individual and collective satisfaction of the community at each period $t = 0, ..., T-1$, and the government capital S_T at the final stage, which purpose is to provide enough resources for the future. The multi-stage stochastic optimization program, which describes this decision-making problem is:

$$\begin{aligned}
&\max_{x_t, c_t, z_t} \mathbb{E}\left[(1-\beta)\sum_{t=0}^{T-1}\rho^{-t}u(c_t) + \beta\rho^{-T}u(S_T)\right] \qquad (24)\\
&\text{subject to } x_t,\ z_t,\ c_t \geq 0,\ t = 0, ..., T-1;\ S_0 \text{ is given.}\\
&x_t \triangleleft \mathcal{F}_t,\ c_t \triangleleft \mathcal{F}_t,\ z_t \triangleleft \mathcal{F}_t,\ t = 0, ..., T-1,\\
&S_{t+1} = [(1-\delta)S_t + x_t](1-\xi_{t+1}) + z_t\xi_{t+1},\ t = 0, ..., T-1,\\
&B_t = \alpha S_t = x_t + c_t + \pi(\mathbb{E}(\xi_{t+1}))z_t,\ t = 0, ..., T-1,
\end{aligned}$$

where $u(\cdot)$ is a governmental utility function, which may vary between risk-neutral and risk-averse risk bearing ability for countries with natural disasters (see Hochrainer and Pflug [11] for more details on governmental risk aversion); the discounting factor ρ is gives more (or less) weight for the future capital and consumption; δ is the capital depreciation rate.

The following scheme represents the dynamics of the model in absolute terms:



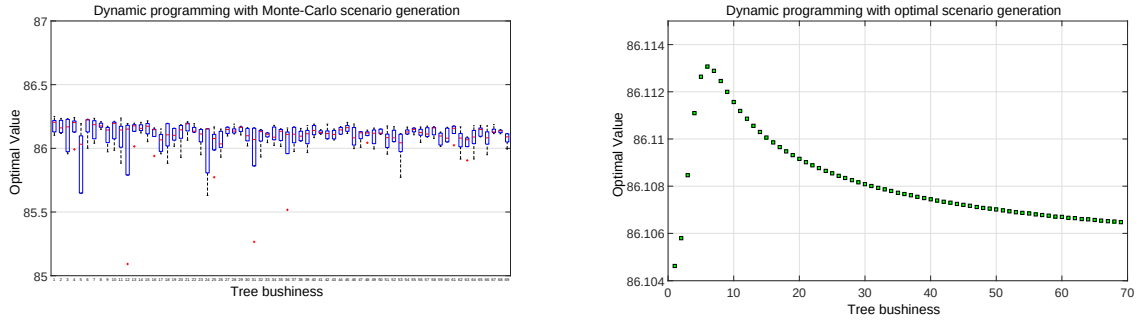
At the initial stage the absolute amount of government consumption is c_0 and the amount of investment in capital formation is x_0. The policy-maker is able to take an insurance, for which he pays $\pi(\mathbb{E}(\xi_1))z_0$ at this stage and gets $z_0\xi_1$ at the next stage. Notice, that in case of no disaster event at the next stage, there is no insurance coverage. Disaster events may happen at stages $t = 1, ..., T$ with probabilities determined by the Fréchet distribution in Figure 8.

In terms of governmental spendings, stages $t = 1, ..., T-1$ are similar to the initial stage, except the fact that the policy-maker also receives the insurance coverage $z_{t-1}\xi_t$, dependent on the insured amount and the magnitude of the disaster event. At the final stage T, no decision is being made and the results are being observed. The insurance coverage $z_{T-1}\xi_T$ is obtained in case of the disaster event.

If formulation (24) satisfies conditions of Theorem 6.3 or 6.4, the optimization problem can be solved via the Algorithm 1. Using simple utility function $u(c) = c$, we solve the optimization problem (24) via the dynamic programming described in the Algorithm 1 with the linear interpolation of the value function, which is possible according to the Corollary 6.5.

As before, we compare Monte-Carlo scenario tree generation with the optimal scenario quantization for the solution of the problem (24) via dynamic programming (Algorithm 1).

In the Figure 9, one can see that the Monte-Carlo scenario generation leads to the convergence of the optimal value in probability, while the stage-wise optimal quantization leads to the convergence in value.



a. Monte-Carlo sampling. b. Optimal quantization.

Figure 9. Optimal value for the problem (24) obtained by the Monte-Carlo and stage-wise optimal scenario generation on scenario trees (in bln. EUR).

The optimal value of the problem (24) is equal to ca. 86.106 bln. EUR and is obtained using parameters in the Table 2.

	S_0 in 2013 Euro bln.	α	β	ρ	δ	γ	insurance load
Values	322.56	0.5	0.9	0.8	0.1	10	0.1

Table 2
Parameters used for the solution (Austrian case study).

In the Figure 10, one can see the dependency of the optimal decision for the problem (24) on the insurance load l (recall, $\pi(\mathbb{E}(\xi_t)) = (1 + l)\mathbb{E}(\xi_t)$). Clearly, the budget allocation decision changes, if the price of the insurance increases. The higher is the price for the insurance, the less amount of budget should be allocated into it.

If $l > 0.01$ and other parameters of the model are as in the Table 2, one can definitely claim, that there is no necessity in taking an insurance, as it is too expensive under the risk estimate in the Table 1, which itself avoids risk underestimation. However, if $l < 0.01$, the optimal strategy would be to allocate some part of the budget into this insurance as it is shown in the Figure 10.

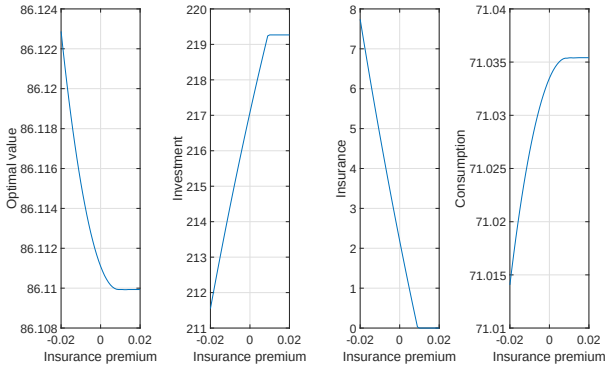


Figure 10. Optimal decision (in EURO bln.) of the problem (24) dependent on the insurance load V.

References

[1] Bellman, Richard E. (1956). "Dynamic Programming," *Princeton University Press*. Princeton, New Jersey.

[2] Bertsekas, Dimitri P. (1976). "Dynamic Programming and Stochastic Control," *Academic Press*. New York.

[3] Bertsekas, Dimitri P. (2007). "Dynamic Programming and Optimal Control," *Athena Scientific* 2(3).

[4] Dreyfus, Stuart E. (1965). "Dynamic Programming and the Calculus of Variations," *Academic Press*. New York.

[5] Ermoliev, Yuri, Kurt Marti, and Georg Ch. Pflug eds. (2004). "Dynamic Stochastic Optimization. Lecture Notes in Economics and Mathematical Systems," *Springer Verlag*. ISBN 3-540-40506-2.

[6] Fishman, S. George. (1995). "Monte Carlo: Concepts, Algorithms, and Applications," *Springer*, New York.

[7] Fort, Jean-Claude, and Gilles Pagés. (2002). "Asymptotics of Optimal Quantizers for Some Scalar Distributions," *Journal of Computational and Applied Mathematics* 146(2), Amsterdam, The Netherlands, Elsevier Science Publishers B. V., pp. 253–275.

[8] Graf, Siegfried, and Harald Luschgy. (2000). "Foundations of Quantization for Probability Distributions," *Lecture Notes in Mathematics* 1730, Springer, Berlin.

[9] Hanasusanto, Grani, and Daniel Kuhn. (2013). "Robust Data-Driven Dynamic Programming," *NIPS Proceedings* 26.

[10] Heitsch, Holger, and Werner Römisch. (2009). "Scenario Tree Modeling for Multi-stage Stochastic Programs," *Mathematical Programming* 118, pp. 371–406.

[11] Hochrainer, Stefan, and Georg Ch. Pflug. (2009). "Natural Disaster Risk Bearing Ability of Governments: Consequences of Kinked Utility," *Journal of Natural Disaster Science* 31(1), pp. 11–21.

[12] Kantorovich, Leonid. (1942). "On the Translocation of Masses," *C.R. (Doklady) Acad. Sci. URSS (N.S.)* 37, pp. 199–201.

[13] Keshavarz, Arezou, and Stephen P. Boyd. (2012). "Quadratic Approximate Dynamic Programming for Input-affine Systems," *International Journal of Robust and Nonlinear Control*.

[14] Lipster, Robert, and Albert N. Shiryayev. (1978). "Statistics of Random Processes," *Springer-Verlag* 2, New York. ISBN 0-387-90236-8.

[15] Mirkov, Radoslava, and Georg Ch. Pflug. (2007). "Tree Approximations of Stochastic Dynamic Programs," *SIAM Journal on Optimization* 18(3), pp. 1082–1105.

[16] Mirkov, Radoslava. (2008). "Tree Approximations of Dynamic Stochastic Programs: Theory and Applications," *VDM Verlag*, pp. 1–176. ISBN 978-363-906-131-4.

[17] Nadaraya, A. Élizbar. (1964). "On Estimating Regression," *Theory of Probability and its Applications* 9(1), pp. 141–142.

[18] Pflug, Ch. Georg. (2001). "Scenario Tree Generation for Multiperiod Financial Optimization by Optimal Discretization," *Mathematical Programming, Series B* 89(2), pp. 251–257.

[19] Pflug, Ch. Georg, and Werner Römisch. (2007). "Modeling, Measuring and Managing Risk," *World Scientific Publishing*, pp. 1–301. ISBN 978-981-270-740-6.

[20] Pflug, Ch. Georg. (2010). "Version-independence and Nested Distributions in Multi-stage Stochastic Optimization," *SIAM Journal on Optimization* 20(3), pp. 1406–1420.

[21] Pflug, Ch. Georg, and Alois Pichler. (2011). "Approximations for Probability Distributions and Stochastic Optimization Problems," *Springer Handbook on Stochastic Optimization Methods in Finance and Energy (G. Consigli, M. Dempster, M. Bertocchi eds.), Int. Series in OR and Management Science* 163(15), pp. 343-387.

[22] Pflug, Ch. Georg, and Alois Pichler. (2012). "A Distance for Multi-stage Stochastic Optimization Models," *SIAM Journal on optimization* 22, pp. 1-23.

[23] Pflug, Ch. Georg, Anna Timonina-Farkas, and Stefan Hochrainer-Stigler. (2017). "Incorporating Model Uncertainty into Optimal Insurance Contract Design," *Insurance: Mathematics and Economics.* http://dx.doi.org/10.1016/j.insmatheco.2016.11.008

[24] Powell, Warren. (2007). "Approximate Dynamic Programming: Solving the Curses of Dimensionality," *Wiley-Blackwell.*

[25] Rojas, Rodrigo, Luc Feyen, Alessandra Bianchi, and Alessandro Dosio. (2012). "Assessment of Future Flood Hazard in Europe Using a Large Ensemble of Bias-corrected Regional Climate Simulations," *Journal of Geophysical Research-Atmospheres* 117(17).

[26] Römisch, Werner. (2010). "Scenario Generation," *Wiley Encyclopedia of Operations Research and Management Science (J.J. Cochran ed.), Wiley.*

[27] Rosen, J. Ben, and Roummel F. Marcia. (2004). "Convex Quadratic Approximation," *Computational Optimization and Applications* 28(173). DOI:10.1023/B:COAP.0000026883.13660.84

[28] Shapiro, Alexander, Darinka Dentcheva, and Andrzej Ruszczyński. (2009). "Lectures on Stochastic Programming: Modeling and Theory," *MPS-SIAM Series on Optimization* 9. ISBN 978-0-898716-87-0.

[29] Timonina, Anna. (2013). "Multi-stage Stochastic Optimization: the Distance Between Stochastic Scenario Processes," *Springer-Verlag Berlin Heidelberg.* DOI 10.1007/s10287-013-0185-3.

[30] Timonina, Anna. (2014). "Approximation of Continuous-state Scenario Processes in Multi-stage Stochastic Optimization and its Applications," *Wien, Univ., Diss.*

[31] Timonina, Anna, Stefan Hochrainer-Stigler, Georg Ch. Pflug, Brenden Jongman, and Rodrigo Rojas (2015). "Structured Coupling of Probability Loss Distributions: Assessing Joint Flood Risk in Multiple River Basins," *Risk Analysis* 35, pp. 2102–2119. DOI:10.1111/risa.12382.

[32] van der Knijff, Johannes, Jalal Younis, and Arie De Roo, A. (2010). "LISFLOOD: a GIS-based Distributed Model for River-basin Scale Water Balance and Flood Simulation," *International Journal of Geographical Information Science* 24 (2).

[33] Villani, Cédric. (2003). "Topics in Optimal Transportation," *Graduate Studies in Mathematics, American Mathematical Society* 58, Providence, RI. ISBN 0-8218-3312-X.

[34] Watson, S. Geoffrey. (1964). "Smooth Regression Analysis," *Sankhya: The Indian Journal of Statistics* Series A, 26(4), pp. 359–372.

6. Appendix

Lemma 6.1. If function $h(s,x)$ is jointly convex in (s,x), then $\min\limits_{x} h(s,x)$ is convex in s.

Lemma 6.2. The following holds true:

1. If $g:\mathbb{R}^n \to \mathbb{R}^n$ is componentwise convex and $V:\mathbb{R}^n \to \mathbb{R}^1$ is convex and monotonically increasing as $y_1 \succeq y_2 \Rightarrow V(y_1) \geq V(y_2)$, then $V \circ g$ is convex;

2. If $g:\mathbb{R}^n \to \mathbb{R}^n$ is componentwise concave and $V:\mathbb{R}^n \to \mathbb{R}^1$ is convex and monotonically decreasing as $y_1 \succeq y_2 \Rightarrow V(y_1) \leq V(y_2)$, then $V \circ g$ is convex;

3. If $g:\mathbb{R}^n \to \mathbb{R}^n$ is linear and $V:\mathbb{R}^n \to \mathbb{R}^1$ is convex and monotone (i.e. either increasing as $y_1 \succeq y_2 \Rightarrow V(y_1) \geq V(y_2)$ or decreasing as $y_1 \succeq y_2 \Rightarrow V(y_1) \leq V(y_2)$), then $V \circ g$ is convex.

Theorem 6.3. Let ξ_0 and ξ_1 be two dependent random variables defined on some probability space $(\Omega, \mathcal{F}, P)$ and let the following hold:

1. Function $h(s, x, \xi_0) : \mathbb{R}^n \to \mathbb{R}^1$ is jointly convex in (s, x) and is monotonically increasing as $s_1 \succeq s_2 \Rightarrow h(s_1, x, \xi_0) \geq h(s_2, x, \xi_0),\ \forall x, \xi_0$;
2. Function $g(s, x, \xi_1) : \mathbb{R}^n \to \mathbb{R}^n$ is componentwise convex in (s, x) and is componentwise increasing in s;
3. Function $V_1(y, \xi_1) : \mathbb{R}^n \to \mathbb{R}^1$ is convex in y and is monotonically increasing as $y_1 \succeq y_2 \Rightarrow V_1(y_1, \xi_1) \geq V_1(y_2, \xi_1),\ \forall \xi_1$.

Then the function $V_0(s, \xi_0) := \min\limits_x \Big\{ h(s, x, \xi_0) + \mathbb{E}\big[V_1(g(s, x, \xi_1), \xi_1) \mid \xi_0\big] \Big\}$ is convex in s and is monotone in the sense of $s_1 \succeq s_2 \Rightarrow V_0(s_1, \xi_0) \geq V_0(s_2, \xi_0),\ \forall \xi_0$.

Theorem 6.4. Let ξ_0 and ξ_1 be two dependent random variables defined on some probability space $(\Omega, \mathcal{F}, P)$ and let the following hold:

1. Function $h(s, x, \xi_0) : \mathbb{R}^n \to \mathbb{R}^1$ is jointly convex in (s, x) and is monotonically decreasing as $s_1 \succeq s_2 \Rightarrow h(s_1, x, \xi_0) \leq h(s_2, x, \xi_0),\ \forall x, \xi_0$;
2. Function $g(s, x, \xi_1) : \mathbb{R}^n \to \mathbb{R}^n$ is componentwise concave in (s, x) and is componentwise increasing in s;
3. Function $V_1(y, \xi_1) : \mathbb{R}^n \to \mathbb{R}^1$ is convex in y and is monotonically decreasing as $y_1 \succeq y_2 \Rightarrow V_1(y_1, \xi_1) \leq V_1(y_2, \xi_1),\ \forall \xi_1$.

Then the function $V_0(s, \xi_0) := \min\limits_x \Big\{ h(s, x, \xi_0) + \mathbb{E}\big[V_1(g(s, x, \xi_1), \xi_1) \mid \xi_0\big] \Big\}$ is convex in s and is monotone in the sense of $s_1 \succeq s_2 \Rightarrow V_0(s_1, \xi_0) \leq V_0(s_2, \xi_0),\ \forall \xi_0$.

Corollary 6.5. Let ξ_0 and ξ_1 be two dependent random variables defined on some probability space $(\Omega, \mathcal{F}, P)$ and let the following hold:

1. Functions $h(s, x, \xi_0) : \mathbb{R}^n \to \mathbb{R}^1$ and $g(s, x, \xi_1) : \mathbb{R}^n \to \mathbb{R}^n$ are linear in (s, x);
2. Function $V_1(y, \xi_1) : \mathbb{R}^n \to \mathbb{R}^1$ is linear y.

Then the function $\Big(h(s, x, \xi_0) + \mathbb{E}\big[V_1(g(s, x, \xi_1), \xi_1) \mid \xi_0\big] \Big)$ is linear in (s, x).

Proof. We do not provide proofs of Lemmas 6.1 and 6.2, Theorems 6.3 and 6.4 and Corollary 6.5 due to their simplicity. □