



Multiscale stochastic optimization: modeling aspects and scenario generation

Martin Glanzer¹ · Georg Ch. Pflug^{1,2}

Received: 4 March 2019
© The Author(s) 2019

Abstract

Real-world multistage stochastic optimization problems are often characterized by the fact that the decision maker may take actions only at specific points in time, even if relevant data can be observed much more frequently. In such a case there are not only multiple decision stages present but also several observation periods between consecutive decisions, where profits/costs occur contingent on the stochastic evolution of some uncertainty factors. We refer to such multistage decision problems with encapsulated multiperiod random costs, as multiscale stochastic optimization problems. In this article, we present a tailor-made modeling framework for such problems, which allows for a computational solution. We first establish new results related to the generation of scenario lattices and then incorporate the multiscale feature by leveraging the theory of stochastic bridge processes. All necessary ingredients to our proposed modeling framework are elaborated explicitly for various popular examples, including both diffusion and jump models.

Keywords Stochastic programming · Scenario generation · Bridge process · Stochastic bridge · Diffusion bridge · Lévy bridge · Compound Poisson bridge · Simulation of stochastic bridge · Multiple time scales · Multi-horizon · Multistage stochastic optimization

1 Introduction

Optimization models over a large time frame can be classified into two types:

✉ Martin Glanzer
martin.glanzer@univie.ac.at

¹ Department of Statistics and Operations Research (DSOR), University of Vienna, Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria

² International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria

- *Multiperiod models* The decisions are made at the very beginning whereas the consequences of the decisions depend on the development of a process over time. A typical example is a buy-and-hold portfolio strategy.
- *Multistage models* Decisions can be made at regular moments in time. Typical examples are active portfolio strategies.

Stochastic multiperiod models are simple from their structure. In contrast, multistage stochastic models are objects of intensive research, see the book of Pflug and Pichler [32]. The purpose of this paper is to introduce models, which incorporate the properties of both, multistage and multiperiod models. The latter deal with the development between the decision stages. Examples for such problems involving different time scales include:

- Supply network extension problems, where major decisions (such as whether to defer, to stage, to mothball, or to abandon a certain infrastructure investment opportunity; cf. [28]) can only be made at strategic time points (say, once every few years), but resulting profits/costs are subject to daily fluctuations of market prices.
- Inventory control problems with limited storage capacity and backlogged/lost demand due to out-of-stock events, where procurement of goods is restricted by logistical constraints/time delays.
- Structured portfolio investment problems, where rebalancing is possible only at given time points (say, once every few weeks due to product terms and conditions), but contained barrier features make profits/losses depend on the full trajectory of asset prices.
- Power plant management problems, where operating plans need to be fixed for a certain period ahead (say, once every few days due to physical constraints avoiding instant reaction to market conditions), but actual profits/losses depend on each tick of the energy market.

To the best of our knowledge, the existing literature does not offer a computational modeling framework designed specifically towards the solution of such multistage stochastic optimization problems, where two different time scales related to one underlying stochastic process are present. The novel approach suggested in this article consists of two parts, each dealing with one of the two time scales. The general idea is to first construct a coarse lattice model for the decision scale and then use a consistent simulation procedure to compute expected profits/costs on a fine time granularity between the decisions. The proposed approach is illustrated in Fig. 1.

Looking only at the coarser decision scale, the requirements to the discrete structure are the same as for any standard multistage stochastic optimization problem. In general, there are three different strategies for the generation of discrete scenarios out of a sample of observed data, as illustrated in Fig. 2. Fans are not an appropriate structure for multistage decision problems, as they cannot reflect the evolution of information. Scenario trees are a popular tool in the literature. However, scenario trees are practically intractable for problems involving a large number of decision stages, due to their exponential growth over time. Therefore, one often reverts to scenario lattices in such cases. While the literature on the construction of scenario trees is relatively rich (see, e.g., [16,18,23,32,33]), the lattice construction literature is rather sparse. The state-of-the art approach is based on the minimization of a distance measure between

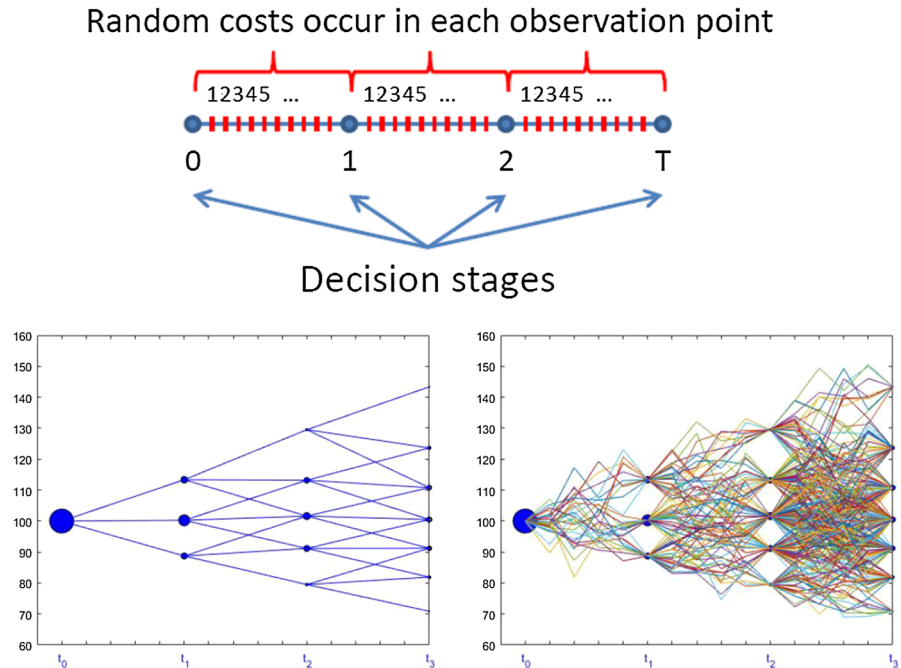


Fig. 1 Multiscale stochastic optimization problems: each decision stage involves multiple observation periods, where actual costs resulting from the proceeding decision are realized. The objective function depends on the whole trajectory, not only on the value of the process at the decision points. Left: a lattice model for the decision stages—node values and transition probabilities are estimated. Probabilities are indicated by the different sizes of the nodes. Right: simulation of the interpolating bridge process between consecutive decision stages to determine realized costs

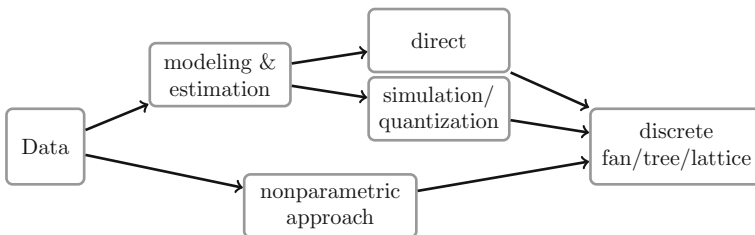


Fig. 2 From data to discrete models for the time evolution

the targeted distribution and its discretization (“optimal quantization”), see [3,25,32]. In this article, we study a lattice generation method along the very upper path of Fig. 2. More precisely, it is a “direct” method for the case when a time-homogeneous Markovian diffusion model is selected in the first step. The approach is purely based on the infinitesimal drift and diffusion coefficient functions of a diffusion model and directly provides a scenario lattice, without requiring a simulation/quantization procedure. While the idea of such a discretization technique appeared already in an early paper by Pflug and Swietanowski [34], it has not been analyzed (or used) yet in the

stochastic optimization literature (cf. the review article of Löhndorf [24]). We make the approach complete in this paper by proving a stability result and error estimate for the optimal value of a generic multistage stochastic optimization problem. In particular, we show that the approximation error regarding the optimal value in the continuous (state space) diffusion model can be controlled when the suggested lattice generation method is applied.

Once the decision time scale has been discretized with a scenario tree/lattice model, a coherent approach for the finer observation time scale requires an interpolation that respects the laws of the underlying stochastic process. This brings us to the theory of stochastic bridges, i.e., processes pinned to a given value at the beginning and the end of a certain time period. We suggest to use a simulation engine to generate a set of paths of the bridge process, and then compute expected profits/costs between decisions based on a Monte-Carlo simulation. This requires a simulatable form of the bridge process. The stochastic processes literature seems to offer mainly abstract theory in this respect. There are some articles on simulation methods (i.e., mainly acceptance-rejection methods) for diffusion bridges and jump-diffusion bridges in the statistical analysis literature since the early 2000's, see [8,9,14,30,36]. However, these methods are inefficient due to a possibly large rejection rate. To make our suggested modeling approach directly applicable, we work out explicitly the bridge process dynamics for some popular diffusion models, including geometric Brownian motion, the Vašíček model, and the Cox–Ingersoll–Ross model. Based on these dynamics, efficient simulation is possible by means of standard discretization schemes for stochastic differential equations. Moreover, we present a simulation scheme for the example of geometric Brownian motion, which operates directly on generated paths from the unconditioned process and thus enables an even more efficient generation of bridge process trajectories. If the cost function is particularly amenable (e.g., linear), a simulation might not even be required, as expected costs can be computed analytically in some models. We also include jump processes in our analysis, as we propose a simulation algorithm for compound Poisson bridges in the case of Normally, Exponentially, or Gamma distributed jump sizes. In particular, we discuss the simulation of the number of jumps of the bridge process and derive the conditional distribution of each jump-size given both the final value of the bridge process as well as the number of jumps in the interval.

The general contribution of this article is to propose a modeling framework and a corresponding scenario generation method, such that an efficient computational solution of multiscale stochastic optimization problems is possible. The details of this contribution are threefold. First, it consists of the general modeling idea, which is based on a consistent but separate scenario generation approach for the two involved time scales. Second, we analyze theoretically a widely unknown direct method for the construction of scenario lattices when the underlying stochastic model is of the diffusion type; this is purely related to the coarser decision time scale. Third, as regards the finer observation time scale, we elaborate the details of a consistent interpolation procedure for a number of popular modeling choices. This includes the presentation of a novel simulation algorithm for compound Poisson bridges.

The outline of the paper is as follows. Section 2 deals with the generation of discrete scenarios as a model for the information flow over the decision stages. In Sect. 3, we present the details related to the suggested interpolation approach for the information

flow through the intermediate observation periods. Section 4 illustrates our modeling framework with a simple multiscale inventory control problem. Moreover, we discuss the applicability and the benefits of the proposed approach. We conclude in Sect. 5.

2 Scenario lattice generation for decision stages

Computational methods for stochastic optimization problems require discrete structures. For multistage problems, *scenario trees* are the standard models for the evolution of uncertainty over time. Scenario trees allow for general path-dependent solutions, as for each node there exists a unique path from the root of the tree. However, scenario trees grow exponentially in the number of stages, a fact that easily overwhelms any computer's memory when it comes to practically-sized problems.¹ Therefore, if the underlying stochastic model is a Markov process, one typically discretizes it in the form of a *scenario lattice*. Lattice models are special cases of graph-based models, where a node does not necessarily have a unique predecessor. Different paths may then connect in a certain node at some later stage. In this way, one can obtain a rich set of paths with relatively few nodes.

The construction of scenario lattices typically works in a two-step procedure. First, one discretizes the marginal distributions for all stages. In a second step, one decides about allowed state transitions and determines conditional transition probabilities between consecutive stages. The state-of-the-art method for such a lattice generation procedure is based on the stagewise minimization of the (Wasserstein) distance between the modeled distribution—which is typically continuous—and its discretization on the lattice. A detailed description of this approach can be found in Löhndorf and Wozabal [25, Sect. 3.2].

We will now study an alternative lattice generation method, which is not based on optimal quantization theory but rather relies on Markov chain approximation results. In particular, this approach allows to construct a scenario lattice directly from the dynamics of a Markovian diffusion process.

2.1 Markov chain approximation for diffusion processes

Birth-and-death Markov chains are discrete stochastic processes defined on the integer grid, where each transition depends only on the current state and allows for three possibilities: to remain in the current state, to move one unit up, or to move one down. Many Markov chains can be approximated by a diffusion process. It works by a transformation of the time scale and a renormalization of the state variable. The idea is, e.g., explained in the book of Karlin and Taylor [20, Ch. 15]. Pflug and Swietanowski [34] have looked at the problem from the converse perspective. They elaborate, without providing error estimates, that any diffusion process possessing a

¹ For the simplest form of a binary tree (which typically will be a rather poor uncertainty model), hourly decisions for a time horizon of one day will correspond to about 17 million nodes, daily decisions for one month will give about 1 billion of nodes, and weekly decisions for one year will result in a magnitude of 10^{15} nodes.

stationary distribution can be approximated by a birth-and-death Markov chain in the following way.

Consider a one-dimensional recurrent Markov process X_t , as defined by

$$\begin{cases} dX_t = \mu(X_t) dt + \sigma(X_t) dW_t \\ X_0 = x_0, \end{cases} \quad (1)$$

where W denotes a standard Brownian motion and the initial value x_0 is a given constant. Throughout the paper, the coefficient functions $\mu(\cdot)$ and $\sigma(\cdot)$ are (as usual) assumed to be square-integrable functions satisfying the following growth conditions:

- $|\mu(x) - \mu(y)| \leq L \cdot |x - y|, \quad |\sigma(x) - \sigma(y)| \leq L \cdot |x - y|,$
- $\mu^2(x) \leq L^2 \cdot (1 + x^2), \quad \sigma^2(x) \leq L^2 \cdot (1 + x^2),$

for some $L > 0$. Notice that the Lipschitz-continuity implies that one may specify a constant L_μ such that $\mu(x) \leq L_\mu + L \cdot |x|$.

Algorithm 2.1 (Markov chain approximation method for diffusion processes) For a diffusion process X as given by (1), define its N -th Markov chain approximation as the process constructed along the following scheme.

1. Choose a strictly monotonic, three times differentiable function $H(x)$ with $H''(0) \leq M < \infty$, for some constant M , as well as functions $g(x)$ and $\tau(x)$ with $|\tau(x)| \leq 1$ for all x , in such a way that the drift and diffusion coefficient functions in (1) are matched:

$$\begin{aligned} \mu(H(x)) &= H'(x)g(H(x)) + \frac{1}{2}H''(x)\tau^2(H(x)) \\ \sigma(H(x)) &= H'(x)\tau(H(x)). \end{aligned}$$

2. Determine the initial state i_0 such that $H(\frac{i_0}{2^N}) = x_0$.
3. Define the transition probabilities

$$\begin{aligned} p_{i,N}^u &:= \left[\frac{1}{2} \left(\tau^2 \left(H \left(\frac{i}{2^N} \right) \right) + \frac{1}{2^N} g \left(H \left(\frac{i}{2^N} \right) \right) \right) \right]_0^1, \\ p_{i,N}^d &:= \left[\frac{1}{2} \left(\tau^2 \left(H \left(\frac{i}{2^N} \right) \right) - \frac{1}{2^N} g \left(H \left(\frac{i}{2^N} \right) \right) \right) \right]_0^1, \\ p_{i,N}^r &:= 1 - p_{i,N}^u - p_{i,N}^d, \end{aligned}$$

where $[x]_0^1 := \min\{\max\{x, 0\}, 1\}$, for jumping up, down, and remaining in its state, respectively.

4. Define the piecewise constant (continuous time) process \tilde{X}^N , where $\tilde{X}_t^N := \tilde{X}_{\lfloor 2^{2N}t \rfloor}^N$ lives in the states $H(\frac{i}{2^N})$; the floor function being denoted by $\lfloor \cdot \rfloor$.

While the idea of Algorithm 2.1 was originally presented in the early paper [34], it has not been analyzed yet in the context of stochastic optimization. We now make

the approach complete by deriving an error estimate for the optimal value of a generic multistage stochastic optimization problem, when the underlying diffusion model is approximated by the method of Algorithm 2.1. We start with some preliminary results required for the proof.

Lemma 2.1 *Let \tilde{X}_t^N be constructed according to Algorithm 2.1 and starting in x_0 at time t_0 . Then, for $t_1 \geq t_0$, the following bound for the second moment holds:*

$$\mathbb{E} \left[\left(\tilde{X}_{t_1}^N \right)^2 \right] \leq x_0^2 \cdot e^{K_1 \cdot (t_1 - t_0)} + o \left(K_2 \cdot (t_1 - t_0) \cdot e^{K_1 \cdot (t_1 - t_0)} \right),$$

where $K_1, K_2 \in \mathbb{R}$ depend only on the Lipschitz(-like) constants controlling the growth of the coefficient functions in (1).

Proof The conditional expected increment of the squared process is given by

$$\begin{aligned} & \mathbb{E} \left[\left(\tilde{X}_{n+1}^N \right)^2 - \left(\tilde{X}_n^N \right)^2 \middle| \left(\tilde{X}_n^N \right)^2 = H^2 \left(\frac{i}{2^N} \right) \right] \\ &= \left[H^2 \left(\frac{i+1}{2^N} \right) - H^2 \left(\frac{i}{2^N} \right) \right] \cdot p_i^{(N)} + \left[H^2 \left(\frac{i}{2^N} \right) - H^2 \left(\frac{i-1}{2^N} \right) \right] \cdot q_i^{(N)} \\ &= \left[\frac{2}{2^N} H \left(\frac{i}{2^N} \right) H' \left(\frac{i}{2^N} \right) + \frac{1}{2^{2N}} [H']^2 \left(\frac{i}{2^N} \right) + \frac{1}{2^{2N}} H \left(\frac{i}{2^N} \right) H'' \left(\frac{i}{2^N} \right) \right] \cdot p_i^N \\ &\quad + \left[-\frac{2}{2^N} H \left(\frac{i}{2^N} \right) H' \left(\frac{i}{2^N} \right) + \frac{1}{2^{2N}} [H']^2 \left(\frac{i}{2^N} \right) + \frac{1}{2^{2N}} H \left(\frac{i}{2^N} \right) H'' \left(\frac{i}{2^N} \right) \right] \cdot q_i^N \\ &\quad + o \left(\frac{1}{2^{2N}} \right) \\ &= \frac{1}{2^{2N}} \left[H \left(\frac{i}{2^N} \right) \left(2H' \left(\frac{i}{2^N} \right) g \left(H \left(\frac{i}{2^N} \right) \right) + H'' \left(\frac{i}{2^N} \right) \tau^2 \left(H \left(\frac{i}{2^N} \right) \right) \right) \right. \\ &\quad \left. + [H']^2 \left(\frac{i}{2^N} \right) \tau^2 \left(H \left(\frac{i}{2^N} \right) \right) \right] + o \left(\frac{1}{2^{2N}} \right) \\ &= \frac{1}{2^{2N}} \left[2H \left(\frac{i}{2^N} \right) \mu \left(H \left(\frac{i}{2^N} \right) \right) + \sigma^2 \left(H \left(\frac{i}{2^N} \right) \right) \right] + o \left(\frac{1}{2^{2N}} \right) \\ &\leq \frac{1}{2^{2N}} \left[2H \left(\frac{i}{2^N} \right) \cdot \left(L_\mu + L \cdot H \left(\frac{i}{2^N} \right) \right) + L^2 \cdot \left(1 + H^2 \left(\frac{i}{2^N} \right) \right) \right] \\ &\quad + o \left(\frac{1}{2^{2N}} \right). \end{aligned}$$

Using the estimate $H(x) \leq 1 + H^2(x)$, we then obtain

$$\mathbb{E} \left[\left(\tilde{X}_{n+1}^N \right)^2 - \left(\tilde{X}_n^N \right)^2 \middle| \left(\tilde{X}_n^N \right)^2 = H^2 \left(\frac{i}{2^N} \right) \right] \leq \frac{1}{2^{2N}} K_1 H^2 \left(\frac{i}{2^N} \right) + o \left(\frac{K_2}{2^{2N}} \right),$$

where $K_1 := 2L_\mu + 2L + L^2$ and $K_2 := L^2 + 2L_\mu$. Then, by the tower property of the expected value, we get

$$\mathbb{E} \left[\left(\tilde{X}_{n+1}^N \right)^2 \right] \leq \mathbb{E} \left[\left(\tilde{X}_n^N \right)^2 \right] \cdot \left(1 + \frac{K_1}{2^{2N}} \right) + o \left(\frac{K_2}{2^{2N}} \right).$$

Applying this iteration scheme, we finally obtain (using the shorthand notation $\tilde{N} := \lfloor 2^{2N}(t_1 - t_0) \rfloor$) the targeted estimate

$$\begin{aligned} \mathbb{E} \left[\left(\tilde{X}_{t_1}^N \right)^2 \right] &\leq \mathbb{E} \left[\left(\tilde{X}_{t_0}^N \right)^2 \right] \cdot \left(1 + \frac{K_1}{2^{2N}} \right)^{\tilde{N}} + o \left(\frac{K_2}{2^{2N}} \right) \sum_{i=0}^{\tilde{N}-1} \left(1 + \frac{K_1}{2^{2N}} \right)^i \\ &\leq x_0^2 \cdot e^{K_1 \cdot (t_1 - t_0)} + o \left(K_2(t_1 - t_0) e^{K \cdot (t_1 - t_0)} \right). \end{aligned}$$

□

Bounds for diffusion processes can be found in the literature. We will use the following result.

Proposition 2.1 *For any even integer $p \geq 2$, the moments of the solution of (1) satisfy the following estimate:*

$$\mathbb{E} [X_t^p] \leq (1 + x_0^p) e^{p(p+1)L^2 t}.$$

Proof See the book of Platen and Heath [35, Lemma 7.8.1].

□

We now establish that weak convergence implies convergence in Wasserstein distance, if the second moments of all involved probability measures are bounded.

Lemma 2.2 *Consider a probability measure P and a sequence of probability measures (P_n) on the compact interval $[-K, K]$, for some $K > 0$. Then, weak convergence implies convergence in Wasserstein distance:*

$$P_n \xrightarrow{w} P \implies \mathfrak{W}(P_n, P) \longrightarrow 0.$$

Proof By Billingsley and Topsøe [7], the weak convergence $P_n \xrightarrow{w} P$ implies the following equivalence: $\sup_{g \in \mathcal{G}} |\int g dP_n - \int g dP| \rightarrow 0$ holds if and only if both

$$\sup_{g \in \mathcal{G}} \sup_{x, y} |g(x) - g(y)| < \infty \tag{2}$$

and

$$\lim_{\delta \rightarrow 0} \sup_{g \in \mathcal{G}} P \left\{ x : \sup_{|x-y| \leq \delta} |g(x) - g(y)| > \varepsilon \right\} \rightarrow 0 \quad \forall \varepsilon > 0 \tag{3}$$

hold. Notice that $\mathcal{G} = \{g \in \text{Lip}(1) : g(0) = 0\}$ on $[-K, K]$ fulfills (2) and (3). As P_n and P have bounded support, it holds $\mathfrak{W}(P_n, P) = \sup_{g \in \text{Lip}(1)} \int g \, dP_n - \int g \, dP$. Thus, it follows $\mathfrak{W}(P_n, P) \rightarrow 0$. \square

Theorem 2.1 *Consider a continuous probability measure P and a sequence of probability measures (P_n) . Suppose that the conditions*

$$\int x^2 \, dP \leq M \quad (4)$$

and

$$\int x^2 \, dP_n \leq M \quad \forall n \quad (5)$$

hold, for some constant $M < \infty$. Then, weak convergence implies convergence in Wasserstein distance:

$$P_n \xrightarrow{w} P \implies \mathfrak{W}(P_n, P) \rightarrow 0.$$

Proof Denote the cdf of P_n by F_n and that of P by F . Notice that, by a version of Čebyšëv's inequality, for $K > 0$ it holds that

$$\int_{-\infty}^{-K} F_n(x) \, dx = \int_{-\infty}^{-K} P_n(-\infty, x] \, dx \leq \int_{-\infty}^{-K} \frac{M}{x^2} \, dx = \frac{M}{K}, \quad (6)$$

and similarly

$$\int_{-\infty}^{-K} F(x) \, dx \leq \frac{M}{K}, \quad \int_K^{\infty} (1 - F_n(x)) \, dx \leq \frac{M}{K}, \quad \int_K^{\infty} (1 - F(x)) \, dx \leq \frac{M}{K}. \quad (7)$$

Now choose K large enough such that $\frac{M}{K} \leq \varepsilon$ holds. Then,

$$\int_{-\infty}^{-K} |F_n(x) - F(x)| \, dx \leq \int_{-\infty}^{-K} F_n(x) \, dx + \int_{-\infty}^{-K} F(x) \, dx \leq 2\varepsilon,$$

and

$$\int_K^{\infty} |F_n(x) - F(x)| \, dx \leq \int_K^{\infty} (1 - F_n(x)) \, dx + \int_K^{\infty} (1 - F(x)) \, dx \leq 2\varepsilon.$$

Define the probability measure P_n^K as P_n conditioned on the interval $[-K, K]$, where we know $P_n([-K, K]) \geq 1 - 2\varepsilon$ by (6) and (7). Define P^K analogously. By Lemma 2.2, it holds

$$\int |F_n^K(x) - F^K(x)| \, dx \rightarrow 0,$$

as $n \rightarrow \infty$. Let $c_n := F_n(K) - F_n(-K)$ and $c := F(K) - F(-K)$. Since F is continuous, $c_n \rightarrow c$. Using

$$F_n^K(x) = \frac{F_n(x) - F_n(-K)}{c_n} \quad \text{and} \quad F^K(x) = \frac{F(x) - F(-K)}{c},$$

we get

$$\begin{aligned} & \int_{-K}^K |F_n^K(x) - F^K(x)| \, dx \\ & \leq \int_{-K}^K |c_n F_n^K(x) - c F^K(x)| \, dx + 2K |F_n(-K) - F(-K)| \\ & \leq c \int_{-K}^K |F_n^K(x) - F^K(x)| \, dx + 2K |c_n - c| + 2K |F_n(-K) - F(-K)|. \end{aligned}$$

Now, for any $\varepsilon > 0$, we can make n such large that

$$2K |c_n - c| \leq \varepsilon, \quad 2K |F_n(-K) - F(-K)| \leq \varepsilon, \quad c \int_{-K}^K |F_n^K(x) - F^K(x)| \, dx \leq \varepsilon.$$

Then, in total, we get $\int_{-K}^K |F_n^K(x) - F^K(x)| \, dx \leq 3\varepsilon$ and finally

$$\mathfrak{W}(P_n, P) = \int |F_n(x) - F(x)| \, dx \leq 7\varepsilon.$$

□

The subsequent result bounds the difference in the value of a diffusion process at a certain future time, if it starts from different values at time zero.

Proposition 2.2 *Define the process $X^z : [0, T] \times \Omega \rightarrow \mathbb{R}$ as the process X defined in (1) but starting in $z \in \{x, y\}$. Assume that for all $t \in [0, T]$ the condition*

$$\begin{aligned} & \int_0^t |\mu(X_s^z)| + |\sigma(X_s^z)|^2 + \frac{|X_s^x - X_s^y| |\mu(X_s^x) - \mu(X_s^y)| + |\sigma(X_s^x) - \sigma(X_s^y)|^2}{|X_s^x - X_s^y|^2} \, ds \\ & < \infty \end{aligned} \tag{8}$$

is satisfied a.e. Then, the following stability of the diffusion process with respect to its starting value holds:

$$\|X_t^x - X_t^y\|_{L^1} \leq |x - y| \cdot \left\| e^{\int_0^t \frac{(X_s^x - X_s^y)(\mu(X_s^x) - \mu(X_s^y)) + \frac{p-1}{2} |\sigma(X_s^x) - \sigma(X_s^y)|^2}{|X_s^x - X_s^y|^2} \, ds} \right\|_{L^q} \tag{9}$$

for any p and q such that $\frac{1}{p} + \frac{1}{q} = 1$.

Proof See Cox et al. [12, Cor. 2.19] □

With the above auxiliary results in hands, we now define a generic multistage stochastic optimization problem. The approximation quality of its optimal value, when the uncertainty process is modeled by a diffusion but approximated on the basis of Algorithm 2.1, is the object that we eventually want to analyze

Definition 2.1 (*GenMSP*) Define a generic multistage stochastic optimization problem (GenMSP) to be of the following form:

$$\left\{ \begin{array}{l} \inf_x \mathbb{E}^{\mathbb{Q}} \left[\sum_{t=0}^T C_t(\xi_t, x_t) \right] \\ \text{s.t. } x_t \in \mathbb{X}_t \quad \forall t = 0, \dots, T \\ x \triangleleft \sigma(\xi). \end{array} \right. \quad (10a)$$

$$(10b)$$

The feasible sets \mathbb{X}_t are assumed to be convex. For the scenario process ξ , we assume $\xi \in L^1(\mathbb{R}, \mathbb{Q})$. The decision process x is required to be adapted to the filtration $\sigma(\xi)$ generated by the scenario process, as is denoted by $x \triangleleft \sigma(\xi)$. Moreover, assume that the cost function $C_t(\cdot, \cdot)$ is convex in the decisions (for any fixed scenario), and Lipschitz continuous (with constant L) w.r.t. the scenario process (for any fixed decision policy). Denote the optimal value of (10), as a function of the underlying probability model \mathbb{Q} , by $v^*(\mathbb{Q})$.

To interpret problem (10), it is the objective to select a nonanticipative (constraint (10b)) decision policy x , which fulfills certain additional constraints (10a), in such a way that cumulative expected costs are minimized. One may think, for instance, in terms of portfolio losses C_t resulting from the stochastic evolution of the financial market ξ_t as well as the selected portfolio composition x_t . Short-selling restrictions would then be an example for “additional constraints” on the decision process.

The concept of the Wasserstein distance² between probability measures will be a key ingredient for our analysis of Algorithm 2.1 in terms of its approximation quality with respect to the optimal solution of GenMSP. In particular, we will rely on the following general stability result for the optimal solution of GenMSP, when the underlying probability model varies.

Proposition 2.3 Consider a GenMSP as defined in Definition 2.1 above. Let the distance between two paths $\xi_{0:t}^{(1)}$ and $\xi_{0:t}^{(2)}$ up to time $t \leq T$ be defined by $\|\xi_{0:t}^{(1)} - \xi_{0:t}^{(2)}\| := \sum_{s=0}^t \|\xi_s^{(1)} - \xi_s^{(2)}\|_1$. Let $\mathbb{Q} \in \{\bar{\mathbb{P}}, \hat{\mathbb{P}}\}$ for two (d -dimensional) Markovian multi-period distributions $\bar{\mathbb{P}}$ and $\hat{\mathbb{P}}$, both defined on some $\Xi \subseteq \mathbb{R}^{d \times T}$. Assume that, for all $t = 0, \dots, T-1$, there exist constants κ_{t+1} and ε_{t+1} such that the Wasserstein distances \mathfrak{W} of the corresponding single-stage conditional transition probability measures $\bar{P}_{t+1}(\cdot | \xi_t)$ and $\hat{P}_{t+1}(\cdot | \xi_t)$ satisfy the conditions

² The definition of the Wasserstein distance can be found in the “Appendix”.

$$\mathfrak{W}\left(\bar{P}_{t+1}\left(\cdot\left|\xi_t^{(1)}\right.\right), \bar{P}_{t+1}\left(\cdot\left|\xi_t^{(2)}\right.\right)\right) \leq \kappa_{t+1} \cdot\left\|\xi_{0:t}^{(1)}-\xi_{0:t}^{(2)}\right\|, \quad(11)$$

$$\mathfrak{W}\left(\bar{P}_{t+1}\left(\cdot\left|\xi_t^{(1)}\right.\right), \hat{P}_{t+1}\left(\cdot\left|\xi_t^{(1)}\right.\right)\right) \leq \varepsilon_{t+1}, \quad(12)$$

uniformly for all paths $\xi_{0:t}^{(1)}, \xi_{0:t}^{(2)}$. Then, the following upper bound for the difference between the optimal values $v^*(\bar{\mathbb{P}})$ and $v^*(\hat{\mathbb{P}})$ holds:

$$\left|v^*(\bar{\mathbb{P}})-v^*(\hat{\mathbb{P}})\right| \leq L \cdot \sum_{t=0}^T \varepsilon_t \prod_{s=t+1}^T\left(1+\kappa_s\right) . \quad(13)$$

Proof Follows immediately from [31, Thm. 6.1] and [32, Lem. 4.27]. \square

We are now ready to formulate the main result of this section.

Theorem 2.2 *Consider a GenMSP according to Definition 2.1. Let the uncertainty process ξ be modeled by a diffusion according to (1). Assume that the coefficient functions satisfy the regularity condition (8). Observe ξ in all decision stages $t=0, \ldots, T$ of GenMSP and denote the resulting discrete-time continuous state-space model by \mathbb{P} . Let ξ be discretized according to the Markov chain approximation method given in Algorithm 2.1 and denote the discrete model resulting from the N -th approximation by $\tilde{\mathbb{P}}^N$. Then, the optimal value $v^*(\tilde{\mathbb{P}}^N)$ of the approximate problem tends to the optimal value $v^*(\mathbb{P})$ of the original problem, as $N \rightarrow \infty$. For fixed N , an error estimate of the form (13) holds.*

Proof We want to show that \mathbb{P} and $\tilde{\mathbb{P}}^N$ satisfy the conditions (11) and (12), with $\varepsilon_t \downarrow 0$ as N increases. Then, the statement follows readily from Proposition 2.3.

The diffusion model satisfies condition (11) by Proposition 2.2. Moreover, since for N large enough it holds

$$\begin{aligned} & \mathbb{E}\left[\tilde{X}_{n+1}^{(N)}-\tilde{X}_n^{(N)}\left|H\left(\frac{i}{2^N}\right)\right.\right] \\ & =\left[H\left(\frac{i+1}{2^N}\right)-H\left(\frac{i}{2^N}\right)\right] \cdot p_i^{(N)}+\left[H\left(\frac{i}{2^N}\right)-H\left(\frac{i-1}{2^N}\right)\right] \cdot q_i^{(N)} \\ & =\frac{1}{2^{2 N}} H^{\prime}\left(\frac{i}{2^N}\right) g\left(H\left(\frac{i}{2^N}\right)\right)+\frac{1}{2^{2 N+1}} H^{\prime \prime}\left(\frac{i}{2^N}\right) \tau^2\left(H\left(\frac{i}{2^N}\right)\right)+o\left(\frac{1}{2^{2 N}}\right) \\ & =\frac{1}{2^{2 N}}\left[\mu\left(H\left(\frac{i}{2^N}\right)\right)\right]+o\left(\frac{1}{2^{2 N}}\right), \end{aligned}$$

as well as

$$\begin{aligned} & \mathbb{E}\left[\left(\tilde{X}_{n+1}^{(N)}-\tilde{X}_n^{(N)}\right)^2\left|H\left(\frac{i}{2^N}\right)\right.\right] \\ & =\left[H\left(\frac{i+1}{2^N}\right)-H\left(\frac{i}{2^N}\right)\right]^2 \cdot p_i^{(N)} \end{aligned}$$

$$\begin{aligned}
& + \left[H\left(\frac{i}{2^N}\right) - H\left(\frac{i-1}{2^N}\right) \right]^2 \cdot q_i^{(N)} \\
& = \frac{1}{2^{2N}} \left[\sigma^2 \left(H\left(\frac{i}{2^N}\right) \right) \right] + o\left(\frac{1}{2^{2N}}\right),
\end{aligned}$$

and

$$\mathbb{E} \left[\left(\tilde{X}_{n+1}^{(N)} - \tilde{X}_n^{(N)} \right)^4 \middle| H\left(\frac{i}{2^N}\right) \right] = o\left(\frac{1}{2^{2N}}\right),$$

it follows the convergence of the finite dimensional distributions

$$\left(\tilde{X}_{\lfloor 2^{2N} t_1 \rfloor}^N, \tilde{X}_{\lfloor 2^{2N} t_2 \rfloor}^N, \dots, \tilde{X}_{\lfloor 2^{2N} T \rfloor}^N \right) \xrightarrow{\mathcal{D}} \left(X_{t_1}, X_{t_2}, \dots, X_T \right),$$

see [20, pg.169]. Since we have constructed the lattice in such a way, that each atom of the distribution of \tilde{X}_t^N is also an atom of the distribution of \tilde{X}_t^M , for all $M \geq N$, it follows also the weak convergence of all conditional probabilities. By Theorem 2.1, the latter implies convergence in Wasserstein distance, as the conditions (4) and (5) hold by Proposition 2.1 and Lemma 2.1, respectively.

Thus, condition (12) is shown to be satisfied. \square

Remark The rescaling of time was necessary in the construction of Algorithm 2.1 in order for Theorem 2.2 to hold. However, notice that the method in essence specifies a ternary transition rule. While blindly using the directly resulting ternary lattice would not rely on any supporting theory, it might still be interesting to test its performance, especially for problems with multiple observation periods but relatively few decisions.

3 Interpolating bridge processes

In Sect. 2, we discussed the generation of discrete scenario trees/lattices out of continuous parametric models, as it is typically required for the computational solution of any multistage stochastic optimization problem. For *multiscale* problems, a discretization of the information flow through all decision stages is not enough, as the stochasticity of the costs between the decision stages is an important factor. In such cases, we suggest to draw on the theory of stochastic bridge processes in order to simulate the behavior of the uncertainty process (with arbitrary granularity of the time increment) between consecutive decisions. In particular, this approach ensures the consistency of the finer multiperiod observation scale and the coarser decision scale by simulating trajectories for the multiperiod costs that connect two decision nodes with each other in a tree/lattice model.

In this section, we make our proposed modeling approach directly applicable by working out the details for several popular examples of stochastic models. In particular, we present a new simulation algorithm for compound Poisson bridges and derive the dynamics for a few diffusion bridge examples in explicit form. From the

latter dynamics, a simulation engine can easily be implemented on the basis of any discretization scheme for stochastic differential equations.³

3.1 Diffusion processes

We start with a generic multi-dimensional model with drift and multiple factors. Afterwards, we derive the bridge process dynamics explicitly for several special cases that are frequently used in the literature. The general theory for diffusion bridges is well-established (see [4,13,37]), but the literature is quite abstract. In particular, we are not aware of any standard textbook that offers explicit examples apart from the basic Brownian bridge. Our relatively simple proof of the subsequent theorem is a generalization and elaboration of the derivations contained in an unpublished manuscript by Lyons [26], that we found online.

Theorem 3.1 *Let X be a d -dimensional n -factor diffusion model, i.e.,*

$$dX_t = \mu(X_t) dt + \Sigma(X_t) dW(t), \quad (14)$$

where $\mu(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}$, $\sigma(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$, and W is an m -dimensional Brownian motion. Then, for $t \in [t_1, t_2]$, the dynamics of X conditioned on both, the starting value x_1 at time t_1 and the final value x_2 at time t_2 , are given by

$$d\hat{X}_t = \left(\mu(\hat{X}_t) + (\Sigma \Sigma^\top(\hat{X}_t)) \nabla_x \log f_{t_2}(x_2 | \hat{X}_t, t) \right) dt + \Sigma(\hat{X}_t) dW(t),$$

where f_{t_2} denotes the transition density of X at time t_2 .

Proof For any $t \in [t_1, t_2]$, denote the density function of the random variable \hat{X}_t by $\hat{f}_t(x | X_{t_1} = x_1, X_{t_2} = x_2)$. Due to Bayes' Theorem and the fact that solutions of SDEs are Markov processes, we may rewrite this function as

$$\hat{f}_t(x | X_{t_1} = x_1, X_{t_2} = x_2) = \frac{f_{t_2}(x_2 | X_t = x) \cdot f_t(x | X_{t_1} = x_1)}{f_{t_2}(x_2 | X_{t_1} = x_1)}. \quad (15)$$

Then, for any Lipschitz-continuous function $h : \mathbb{R}^d \rightarrow \mathbb{R}$, it holds that

$$\begin{aligned} \mathbb{E} \left[h(X_t) \middle| X_{t_1} = x_1, X_{t_2} = x_2 \right] &= \int_{\mathbb{R}^d} h(x) \cdot \hat{f}_t(x | X_{t_1} = x_1, X_{t_2} = x_2) dx \\ &= \frac{1}{f_{t_2}(x_2 | X_{t_1} = x_1)} \mathbb{E} \left[\int_{t_1}^t \left\{ h(X_s) \frac{\partial}{\partial s} f_{t_2}(X_2 | X_s, s) \right. \right. \\ &\quad \left. \left. + \sum_{i=1}^d \left(\frac{\partial}{\partial x^i} [h(X_s) \cdot f_{t_2}(x_2 | X_s, s)] \right) \cdot \mu_i(X_s) \right\} ds \right] \end{aligned}$$

³ See, e.g., the book of Kloeden and Platen [22] for a detailed treatment.

$$\begin{aligned}
& + \frac{1}{2} \sum_{i,j=1}^d \left(\frac{\partial^2}{\partial x^i \partial x^j} [h(X_s) \cdot f_{t_2}(x_2|X_s, s)] \right) \cdot [\Sigma \Sigma^\top(X_t)]_{i,j} \Big\} ds \Big| X_{t_1} = x_1 \Big] \\
& = \frac{1}{f_{t_2}(x_2|X_{t_1} = x_1)} \mathbb{E} \left[\int_{t_1}^t \left\{ \sum_{i=1}^d \mu_i(X_s) \left(f_{t_2}(x_2|X_s, s) \frac{\partial}{\partial x^i} h(X_s) \right) \right. \right. \\
& \quad + \frac{1}{2} \sum_{i,j=1}^d \left(f_{t_2}(x_2|X_s, s) \cdot \frac{\partial^2}{\partial x^i \partial x^j} h(X_s) + 2 \frac{\partial}{\partial x^i} h(X_s) \frac{\partial}{\partial x^j} f_{t_2}(x_2|X_s, s) \right) \\
& \quad \times [\Sigma \Sigma^\top(X_t)]_{i,j} \Big\} ds \Big| X_{t_1} = x_1 \Big],
\end{aligned}$$

by the multi-dimensional Itô lemma and the Kolmogorov backward equation, which ensures

$$\frac{\partial}{\partial s} f_{t_2}(x_2|X_s, s) + \sum_{i=0}^d \mu_i(X_s) \frac{\partial}{\partial x^i} f_{t_2}(x_2|X_s, s) + \frac{1}{2} \sum_{i,j=1}^d \frac{\partial^2}{\partial x^i \partial x^j} f_{t_2}(x_2|X_s, s) = 0.$$

Differentiation with respect to the time parameter gives

$$\begin{aligned}
\frac{\partial}{\partial t} \mathbb{E} [h(X_t) | X_{t_1} = x_1, X_{t_2} = x_2] &= \int_{\mathbb{R}^d} h(x) \frac{\partial}{\partial t} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) dx \\
&= \int_{\mathbb{R}^d} \left\{ \sum_{i=1}^d \mu_i(x) \frac{\partial}{\partial x^i} h(x) + \frac{1}{2} \sum_{i,j=1}^d [\Sigma \Sigma^\top(x)]_{i,j} \frac{\partial^2}{\partial x^i \partial x^j} h(x) \right. \\
&\quad + \sum_{i,j=1}^d [\Sigma \Sigma^\top(x)]_{i,j} \frac{\partial}{\partial x^i} h(x) \frac{\partial}{\partial x^j} \log f_{t_2}(x_2|x, t) \Big\} \\
&\quad \cdot \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) dx.
\end{aligned} \tag{16}$$

The function h is Lipschitz by assumption and thus its gradient is bounded. It can be seen from (15) that $\hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \rightarrow 0$, $\frac{\partial}{\partial x^i} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \rightarrow 0$, as any $x^i \rightarrow \pm\infty$. Therefore, integrating (16) twice by parts gives

$$\begin{aligned}
& \int_{\mathbb{R}^d} h(x) \frac{\partial}{\partial t} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) dx \\
&= \int_{\mathbb{R}^d} \left\{ h(x) \left(- \sum_{i=1}^d \frac{\partial}{\partial x^i} \mu_i(x) \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \right. \right. \\
&\quad \left. \left. - \sum_{i,j=1}^d \frac{\partial}{\partial x^i} [\Sigma \Sigma^\top(x)]_{i,j} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \frac{\partial}{\partial x^j} \log f_{t_2}(x_2|x, t) \right) \right\}
\end{aligned}$$

$$\begin{aligned}
& -\frac{1}{2} \sum_{i,j=1}^d \frac{\partial}{\partial x^i} h(x) \frac{\partial}{\partial x^j} [\Sigma \Sigma^\top(x)]_{i,j} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \Big\} dx \\
& = \int_{\mathbb{R}^d} h(x) \left(-\sum_{i=1}^d \frac{\partial}{\partial x^i} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \left(\mu_i(x) \right. \right. \\
& \quad \left. \left. + \sum_{j=1}^d [\Sigma \Sigma^\top(x)]_{i,j} \frac{\partial}{\partial x^j} \log f_{t_2}(x_2|x, t) \right) \right. \\
& \quad \left. + \frac{1}{2} \sum_{i,j=1}^d \frac{\partial^2}{\partial x^i \partial x^j} \left([\Sigma \Sigma^\top(x)]_{i,j} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \right) \right) dx,
\end{aligned}$$

from which we can deduce

$$\begin{aligned}
\frac{\partial}{\partial t} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) & = -\sum_{i=1}^d \frac{\partial}{\partial x^i} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \cdot v_i(x, t) \\
& \quad + \frac{1}{2} \sum_{i,j=1}^d \frac{\partial^2}{\partial x^i \partial x^j} \hat{f}_t(x|X_{t_1}, t_1, X_{t_2}, t_2) \cdot [\Sigma \Sigma^\top(x)]_{i,j},
\end{aligned} \tag{17}$$

with

$$\begin{aligned}
v & : \mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}^d \\
(x, t) & \mapsto \mu(x) + [\Sigma \Sigma^\top(x)] \nabla_x \log f_{t_2}(x_2|x, t).
\end{aligned}$$

Equation (17) corresponds to the Fokker-Planck equation of the diffusion process

$$dX_t = v(X_t, t) dt + \Sigma(X_t) dW_t.$$

□

We subsequently focus on the one-dimensional case. Let $X_{t_2} = x_2$ be fixed for all examples below.

General state-dependent parameters For a general univariate diffusion process X described by the SDE

$$dX_t = \mu(X_t) dt + \sigma(X_t) dW_t,$$

the dynamics of the associated bridge process are given by

$$d\hat{X}_t = \left(\mu(\hat{X}_t) + \sigma^2(\hat{X}_t) \frac{\partial}{\partial x} \log f_{t_2}(X_2|\hat{X}_t, t) \right) dt + \sigma(\hat{X}_t) dW_t,$$

where f_t denotes the transition density of X at time t .

Vašíček model/Ornstein–Uhlenbeck process The model presented in Vašíček [45] is considered as the first stochastic model for the term structure of interest rates. It is a one-factor model for the short rate, featuring mean reversion. In Vašíček's model, the instantaneous rate r is described by a Gaussian Ornstein–Uhlenbeck process, i.e., as the solution of the SDE

$$dr_t = \kappa(\theta - r_t) dt + \sigma dW_t, \quad (18)$$

where the parameter θ can be interpreted as the long-term mean, κ determines the speed of mean-reversion, and W is a standard Brownian motion; the volatility being specified by the (constant) parameter σ . For $s \leq t$, the transition density of r is that of a Normal distribution, i.e.

$$f_t(x|r_s, s) = \frac{1}{\sqrt{2\pi v(t-s)}} \exp\left(-\frac{(x - \theta + (\theta - r_s)e^{-\kappa(t-s)})^2}{2v(t-s)}\right),$$

where $v(\Delta t) := \frac{\sigma^2(1-e^{-2\kappa\Delta t})}{2\kappa}$. Hence, the derivative of the logarithmized transition density is a closed form expression and the bridge process \hat{r} associated with r , pinned to the value $r_{t_2} = x_2$, is described by the dynamics

$$d\hat{r}_t = \left(\kappa(\theta - \hat{r}_t) + \frac{2\kappa(x_2 - \theta + e^{-\kappa(t_2-t)}(\theta - \hat{r}_t))}{1 - e^{-2\kappa(t_2-t)}} \right) dt + \sigma dW_t. \quad (19)$$

Cox–Ingersoll–Ross (CIR) model/square-root diffusion The second classical interest rate term structure model was introduced in Cox et al. [11]. It is typically referred to as CIR model. The square root diffusion process

$$dr_t = \kappa(\theta - r_t) dt + \sigma\sqrt{r_t} dW_t$$

is used as an improvement of the Vašíček model. The transition density $f(\cdot|\cdot, \cdot)$ of the square-root diffusion process is a cumbersome object but yet an analytic expression. Hence, the bridge process associated with the CIR model is described by tractable dynamics. In particular, for $0 < s < t < T$, we get

$$\begin{aligned} \eta(t, x; x_s, s) &:= \frac{\partial}{\partial x} \log f_t(x|x_s, s) \\ &= \left(\frac{1}{2\kappa(e^{\kappa(t-s)} - 1)^2 x \sigma^2 \sqrt{x x_s e^{\kappa(s+t)}} I_q[v(x)]} \right) \\ &\quad \times \left\{ 2\kappa^2 x x_s (e^{\kappa(2t-s)} - e^{\kappa t}) (I_{q-1}[v(x)] + I_{q+1}[v(x)]) \right. \\ &\quad \left. - (\kappa(e^{\kappa(t-s)} - 1) \sqrt{x x_s e^{\kappa(t+s)}} I_q[v(x)] \right. \\ &\quad \left. \times (2\theta\kappa - \sigma^2 + e^{\kappa(t-s)}(4\kappa x - 2\theta\kappa + \sigma^2)) \right\}, \end{aligned}$$

where

$$v(x) := \frac{4\kappa \sqrt{xx_s} e^{\kappa(t+s)}}{\sigma^2(e^{\kappa t} - e^{\kappa s})}, \quad q := \frac{2\kappa\theta}{\sigma^2} - 1,$$

and $I_\alpha(\cdot)$ denotes the modified Bessel function of the first kind. Then, the bridge process dynamics for the CIR model are given by

$$d\hat{r}_t = \left(\kappa(\theta - \hat{r}_t) + \hat{r}_t \sigma^2 \eta(t_2, x_2; \hat{r}_t, t) \right) dt + \sigma \sqrt{\hat{r}_t} dW_t.$$

Geometric Brownian motion (GBM) For a GBM X , described by

$$dX_t = X_t(\mu dt + \sigma dW_t), \quad (20)$$

the transition density is available as an analytic expression. Thus, the dynamics of the associated bridge process take the explicit form

$$d\hat{X}_t = \hat{X}_t \left(\frac{\log(x_2) - \log(\hat{X}_t) - \left(\mu - \frac{1}{2}\sigma^2\right)(t_2 - t)}{t_2 - t} dt + \sigma dW_t \right).$$

Brownian motion with drift In the simplest case of a Brownian motion with constant drift and volatility, i.e.,

$$dX_t = \mu dt + \sigma dW_t,$$

the associated bridge process is the well-known Brownian bridge. Its dynamics are given by

$$d\hat{X}_t = \frac{x_2 - \hat{X}_t}{t_2 - t} dt + \sigma dW_t.$$

3.1.1 Pathwise construction of the bridge process for GBM

The subsequent result shows how to translate a set of Brownian motion trajectories into a set of geometric Brownian bridge trajectories. Thus, simulation of the GBM bridge is straightforward and requires only the generation of Gaussian random variables.

Proposition 3.1 *Consider a GBM X , as defined in (20). Assume that it starts in $X_{t_1} = x_1$ and it shall be pinned to the value x_2 at time t_2 . Then, for $t_1 \leq t \leq t_2$, the bridge process \hat{X} is given by*

$$\hat{X}_t = x_1 \cdot \exp \left\{ \sigma \left(W_t - W_{t_1} - \frac{t - t_1}{t_2 - t_1} (W_{t_2} - W_{t_1}) \right) + \frac{t - t_1}{t_2 - t_1} \log \left(\frac{x_2}{x_1} \right) \right\}. \quad (21)$$

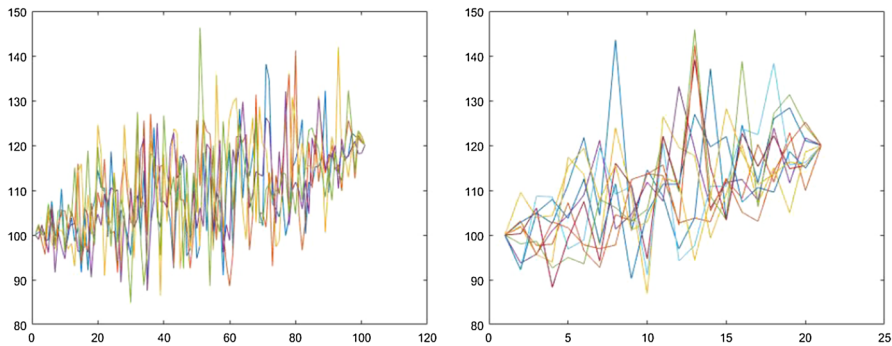


Fig. 3 Sample paths of GBM bridges ($\mu = 0.01$, $\sigma = 0.2$). Left: 100 steps. Right: 20 steps

Proof Obviously, $\hat{X}_{t_1} = x_1$ as well as $\hat{X}_{t_2} = x_2$ do hold. Moreover, let us check that the structure of the original process is indeed preserved by this bridge process. Denote the exponent in (21) by Y_t and consider $x_2 = e^{Y_{t_2}}$ as a lognormally distributed random variable, where $\text{Var}(Y_{t_2}) = \sigma^2(t_2 - t_1)$. Define $\Sigma_{t_1}(t) := \sigma^2(t - t_1)$. Then, for $t_1 \leq s_1 \leq s_2 \leq t_2$, it holds that

$$\begin{aligned} \text{Cov}(Y_{s_1}, Y_{s_2}) &= \Sigma_{t_1}(s_1) - \frac{\Sigma_{t_1}(s_1)}{\Sigma_{t_1}(t_2)} \Sigma_{t_1}(s_2) - \frac{\Sigma_{t_1}(s_2)}{\Sigma_{t_1}(t_2)} \Sigma_{t_1}(s_1) \\ &\quad + \frac{\Sigma_{t_1}(s_1) \cdot \Sigma_{t_1}(s_2)}{(\Sigma_{t_1}(t_2))^2} \Sigma_{t_1}(t_2) + \frac{\Sigma_{t_1}(s_2) \cdot \Sigma_{t_1}(s_1)}{(\Sigma_{t_1}(t_2))^2} \Sigma_{t_1}(t_2) \\ &= \sigma^2(s_1 - t_1), \end{aligned}$$

which confirms that Y is again a Brownian motion and thus \hat{X} is a GBM. \square

Figure 3 shows a collection of sample paths simulated via (21).

3.2 Jump processes

Stochastic processes that do not fluctuate in a continuous manner but rather by sudden jumps, are popular models for a variety of applications. The majority of typical jump models belongs to the class of Lévy processes. Lévy processes are stochastic processes characterized by independent and stationary increments as well as stochastically continuous sample paths.⁴ In addition to their prominence in the physical sciences,⁵ there is a particularly vast literature on Lévy processes as a model for the random evolution of variables present in the financial markets.⁶ As we are dealing with bridge processes here, let us mention the fact that the Markov property of Lévy bridges is inherited from the Markov property of Lévy processes [19, Proposition 2.3.1].

⁴ See, e.g., the books of Applebaum [1], Bertoin [6], or Sato [38] for Lévy processes theory.

⁵ See, e.g., the review article [47] on the subject and other articles contained in [5].

⁶ See, e.g., the books of Cont and Tankov [10], Schoutens [39], Schoutens and Cariboni [40].

3.2.1 Compound Poisson bridges

The most fundamental and prominent jump process is the Poisson process, counting the number of occurrences of some random event. For the modeling of a situation where not only the number of those (quantifiable) events but also their size matters, the compound Poisson process is a natural extension. It is extensively used, e.g., for actuarial applications as insurance companies are naturally not only interested in the number of claims happening to their customers but even more importantly in the claim sizes.⁷

We present a method to simulate sample paths from a compound Poisson bridge process, i.e. a compound Poisson process with given initial and final value (and time). For jump-size distribution families that are closed under convolution or where convolution results in another tractable parametric family, some ingredients to our simulation scheme can be derived analytically and thus efficient simulation is possible. We carry out this exercise for the most popular representatives of jump-size distributions, i.e., the Normal distribution, the Exponential distribution, and the Gamma distribution. For distributions that do not allow for a tractable representation of the required convolution objects, one will have to revert to statistical procedures such as acceptance-rejection methods.

Consider a compound Poisson process X with intensity γ and jump-size distribution given by the density f . To avoid notational conflicts, we reserve the lower index in X_t to describe the process X at time t . In contrast, we use an upper index to enumerate individual jumps (as random variables). The realization of an ' i -th' jump X^i is denoted by x_i . Consider now the process X_t in the interval $[t_1, t_2]$, where we are given the values X_{t_1} and X_{t_2} . Define $c := X_{t_2} - X_{t_1}$. We suggest the simulation of the bridge process to be performed in the following three steps.

I: Simulation of the number of jumps As a first step, simulate from the conditional Poisson process N given the value of the sum $\sum_{i=1}^N X^i = c$. This yields a realization of the number of jumps occurring over the considered time interval $[t_1, t_2]$. The probability function of this object is given by

$$\begin{aligned} \mathbb{P}\left[N = n \mid \sum_{i=1}^N X^i = c\right] &= \frac{f_{\sum_{i=0}^N X^i | N=n}(c) \cdot \mathbb{P}[N = n]}{f_{\sum_{i=1}^N X^i}(c)} \\ &= \frac{f^{*n}(c) \cdot \frac{(\lambda(t_2 - t_1))^n}{n!}}{\sum_{m=0}^{\infty} f^{*m}(c) \cdot \frac{(\lambda(t_2 - t_1))^m}{m!}}. \end{aligned} \quad (22)$$

For simulation purposes we cut the support of this conditional distribution of N to an interval $[0, \bar{N}] \subseteq \mathbb{N}_0$, in such a way that $\mathbb{P}[N > \bar{N} \mid \sum_{i=1}^N X^i = c] < \varepsilon$, for some small value of ε .

⁷ The book of Albrecher and Asmussen [2] includes a comprehensive treatment of the compound Poisson model in risk theory, including not only an exhaustive list of its properties but also a discussion of its wide range of applications. In particular, the problem studied in [2, Chapter V, pg. 146] is of a related flavor to the problem of this section: They characterize a sample path in the compound Poisson risk model given that it leads to ruin.

Consider the Normal distribution as a jump size distribution, i.e. $X^i \stackrel{iid}{\sim} \mathcal{N}(\mu, \sigma^2)$, and let $c > 0$. The convolution of j iid $\mathcal{N}(\mu, \sigma^2)$ distributions is an $\mathcal{N}(j\mu, j\sigma^2)$ distribution. Thus,

$$\begin{aligned} \mathbb{P}\left[N > \bar{N} \mid \sum_{i=1}^N X^i = c\right] &= \sum_{n=\bar{N}+1}^{\infty} \frac{f^{*n}(c) \cdot \left(\frac{(\lambda(t_2-t_1))^n}{n!} e^{-\lambda(t_2-t_1)}\right)}{\sum_{m=0}^{\infty} f^{*m}(c) \cdot \mathbb{P}[N = m]} \\ &\leq \sum_{n=\bar{N}+1}^{\infty} f^{*n}(c) = \sum_{n=\bar{N}}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2n}} \exp\left(-\frac{(c-n\mu)^2}{2n\sigma^2}\right) \\ &\leq \sum_{n=\bar{N}+1}^{\infty} c_1 \cdot \exp(-c_2n) = c_1 \cdot \frac{e^{-c_2(\bar{N}+1)}}{1 - e^{-c_2}}, \end{aligned}$$

where $c_1 := \frac{1}{\sqrt{2\pi\sigma^2}} \exp(\frac{\mu c}{\sigma^2})$, $c_2 := \frac{\mu^2}{2\sigma^2}$. To ensure this upper bound to be smaller than ε , we therefore require $\varepsilon \ll c_1$ and

$$\bar{N} > -\frac{\log(\varepsilon(1 - e^{-c_2})/c_1)}{c_2} - 1.$$

The convolution of j independent Exponential distributions with parameter λ gives an Erlang distribution with parameters λ and j . The convolution of j Gamma(α, θ) distributions gives a Gamma($j\alpha, \theta$) distribution. The Erlang distribution is a special case of the Gamma distribution, i.e. for integer-valued shape parameters. Hence, a criterion for \bar{N} associated with Gamma distributed jumps also gives a criterion associated with Exponentially distributed jumps. From requiring, for the Gamma distribution case,

$$\mathbb{P}\left[N > \bar{N} \mid \sum_{i=1}^N X^i = c\right] \leq \sum_{n=\bar{N}+1}^{\infty} f^{*n}(c) = 1 - \frac{e^{-\theta c}}{c} \sum_{n=1}^{\bar{N}} \frac{(\theta c)^{n\alpha}}{\Gamma(n\alpha)} \leq \varepsilon,$$

we get the condition

$$\sum_{n=1}^{\bar{N}} \frac{(\theta c)^{n\alpha}}{\Gamma(n\alpha)} \geq c(1 - \varepsilon)e^{\theta c},$$

from which one obtains \bar{N} by running an elementary trial and error program.

Having determined the value \bar{N} , one can then easily compute an approximation of $\mathbb{P}[N = n \mid \sum_{i=1}^N X^i = c]$, for all $n = 1, \dots, \bar{N}$, by cutting the sum in the denominator of (22) after the index \bar{N} . The cumulative distribution function is then easily obtained by summation of the single probabilities and inverse transform sampling gives a straightforward simulation scheme by applying its inverse to random draws from the uniform distribution on $[0, 1]$.

II: Simulation of the jumping times Suppose that some value n for the number of jumps in the interval $[t_1, t_2]$ has been simulated by the method outlined above. Then, the precise jumping times are uniformly distributed over this interval. More precisely, the joint distribution of the jumping times (τ_1, \dots, τ_n) equals the law of the order statistics of n independent uniform random variables on $[t_1, t_2]$ (cf., e.g., [10, Prop. 2.9]). Thus, the jumping times can easily be generated by another n calls of a standard (pseudo) random number generator.

III: Simulation of the jump sizes As a last step, let us study the conditional distribution of the summands $X^i, i = 1, \dots, n$, given the value of the sum $\sum_{i=1}^n X^i = c$, where n , the number of jumps, is fixed. The corresponding densities are given by

$$\begin{aligned} f_{X^1|X^1+\dots+X^n}(x|c) &= \frac{f(x)f^{*(n-1)}(c-x)}{f^{*n}(c)} \\ f_{X^2|X^2+\dots+X^n}(x|c-x_1) &= \frac{f(x)f^{*(n-2)}(c-x_1-x)}{f^{*(n-1)}(c-x_1)} \\ &\vdots \\ f_{X^{n-1}|X^{n-1}+X^n}\left(x\left|c-\sum_{j=1}^{n-2}x_j\right.\right) &= \frac{f(x)f(c-\sum_{j=1}^{n-2}x_j-x)}{f^{*2}(c-\sum_{j=1}^{n-2}x_j)} \\ f_{X^n|X^n}\left(x\left|\sum_{j=1}^{n-1}x_j\right.\right) &= \frac{f(x)f^{*0}(c-\sum_{j=1}^{n-1}x_j-x)}{f(c-\sum_{j=1}^{n-1}x_j)} = \delta_{c-\sum_{j=1}^{n-1}x_j}(x), \end{aligned} \quad (23)$$

where we use a version of Bayes rule and the decomposition

$$f_{S_1,S}(x, z) = f_{S_1}(x)f_{S_2}(z-x)$$

for the joint density of the summand S_1 and the sum $S = S_1 + S_2$ of two random variables S_1 and S_2 . The n -fold convolution of the function f with itself is denoted by f^{*n} .

Remark Notice that for the last jump X^n the conditional jump-size distribution is a Dirac distribution with all the mass centered in the remaining gap between the target value and the value of the process after the penultimate jump x^{n-1} . Hence, the proposed simulation procedure ends up in the targeted value with probability one.

For Propositions 3.2–3.4 below, denote the jump size distribution by F . We study the conditional distribution \hat{F}_k of the k -th jump X^k given the value of the sum of the remaining $(n-k+1)$ jumps, i.e. $\sum_{i=k}^n X^i = c - \sum_{j=1}^{k-1} x_j =: C_k$.

Proposition 3.2 (Gaussian jumps) *Let $F \sim \mathcal{N}(\mu, \sigma^2)$. Then, $\hat{F}_k^{\mathcal{N}}$ is a normal distribution with mean $\frac{C_k}{n-k+1}$ and variance $\left(\frac{n-k}{n-k+1}\right)\sigma^2$, for any $1 \leq k \leq n$.*

Proof Using the convolution properties of the Normal distribution, $\hat{F}_k^{\mathcal{N}}$ is characterized by the density function

$$\begin{aligned} f_{X^k|X^k, \dots, X^n}(x|C_k) &= \frac{f(x)f^{*(n-k)}(C_k - x)}{f^{*(n-k+1)}(C_k)} \\ &= \frac{1}{\sqrt{2\pi \left(\frac{n-k}{n-k+1}\right) \sigma^2}} \\ &\quad \times \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} - \frac{((C_k - x) - (n-k)\mu)^2}{2(n-k)\sigma^2} + \frac{(C_k - (n-k+1)\mu)^2}{2(n-k+1)\sigma^2}\right) \\ &= \frac{1}{\sqrt{2\pi \left(\frac{n-k}{n-k+1}\right) \sigma^2}} \exp\left(-\frac{\left(x - \frac{C_k}{n-k+1}\right)^2}{2\left(\frac{n-k}{n-k+1}\right) \sigma^2}\right). \end{aligned}$$

□

Proposition 3.3 (Exponentially distributed jumps) *Let $F \sim \text{Exp}(\lambda)$. Then, \hat{F}_k^{Exp} is a Lomax distribution, for any $1 \leq k \leq n$.⁸ In particular,*

$$\hat{F}_k^{\text{Exp}} \sim \text{Lomax}(k-n, -C_k).$$

Proof Using the convolution properties of the Exponential distribution, \hat{F}_k^{Exp} is characterized by the density function

$$\begin{aligned} f_{X^k|X^k, \dots, X^n}(x|C_k) &= \frac{f(x)f^{*(n-k)}(C_k - x)}{f^{*(n-k+1)}(C_k)} \\ &= \frac{\lambda e^{-\lambda x} \frac{\lambda^{n-k} (C_k - x)^{n-k-1}}{(n-k-1)!} e^{-\lambda(C_k - x)}}{\frac{\lambda^{n-k+1} C_k^{n-k}}{(n-k)!} e^{-\lambda C_k}} \\ &= \frac{(n-k)(C_k - x)^{n-k-1}}{C_k^{n-k}} = \frac{n-k}{C_k} \left(1 - \frac{x}{C_k}\right)^{n-k-1} \\ &= f_{\text{Lom}}(x; -(n-k), -C_k). \end{aligned}$$

□

Remark The Lomax distribution (sometimes also called Pareto type II distribution) is a special case of the generalized Pareto distribution (GP). In particular, it holds

⁸ The density function of the $\text{Lomax}(\alpha, \beta)$ distribution is given by

$$f_{\text{Lom}}(y; \alpha, \beta) = \frac{\alpha}{\beta} \left(1 + \frac{y}{\beta}\right)^{-(\alpha+1)},$$

where α is a shape parameter and $1/\beta$ is a scale parameter.

$\text{Lomax}(\alpha, \beta) \sim \text{GP}(0, 1/\alpha, \beta/\alpha)$. The GP distribution is typically contained in commercial software packages. Built in functions can then be used for straightforward simulation of the Lomax distribution.

Remark Observe in passing, as a quick cross-check of the above results, that in both the Normal and the Exponential distribution case, the derived conditional distributions \hat{F}_n of the last jump X^n have expectation C_n and zero variance.

Proposition 3.4 (Gamma distributed jumps) *Let $F \sim \text{Gamma}(\alpha, \theta)$. Then, \hat{F}_k^Γ is a generalized Beta distribution of first kind, for any $1 \leq k < n$.⁹ In particular,*

$$\hat{F}_k^\Gamma \sim \text{GB1}(1, C_k, \alpha, (n-k)\alpha).$$

Proof Using the convolution properties of the Gamma distribution, \hat{F}_k^Γ is characterized by the density function

$$\begin{aligned} f_{X^k|X^1, \dots, X^n}(x|C_k) &= \frac{f(x)f^{*(n-k)}(C_k - x)}{f^{*(n-k+1)}(C_k)} \\ &= \frac{\frac{\theta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\theta x} \frac{\theta^{(n-k)\alpha}}{\Gamma((n-k)\alpha)} (C_k - x)^{(n-k)\alpha-1} e^{-\theta(C_k-x)}}{\frac{\theta^{(n-k+1)\alpha}}{\Gamma((n-k+1)\alpha)} C_k^{(n-k+1)\alpha-1} e^{-\theta C_k}} \\ &= \frac{\Gamma((n-k+1)\alpha)}{\Gamma(\alpha)\Gamma((n-k)\alpha)} x^{\alpha-1} \frac{(C_k - x)^{(n-k)\alpha-1}}{C_k^{(n-k+1)\alpha-1}} \\ &= \frac{1}{B(\alpha, (n-k)\alpha)} x^{\alpha-1} \frac{\left(1 - \frac{x}{C_k}\right)^{(n-k)\alpha-1}}{C_k^\alpha} \\ &= f_{\text{GB1}}(x; 1, C_k, \alpha, (n-k)\alpha). \end{aligned}$$

□

Remark As the Gamma function $\Gamma(\cdot)$ is only defined for strictly positive arguments, the case $k = n$ is not covered in Proposition 3.4 above. However, we have generally addressed the latter case before.

The simulation scheme for compound Poisson processes, that has been elaborated in this section, is summarized in Algorithm 1 below. It includes the Normal, the Exponential and the Gamma distribution for the jump size. Figure 4 visualizes sample paths generated on the basis of this algorithm.

⁹ The density function of the generalized Beta distribution of first kind is given by

$$f_{\text{GB1}}(y; a, b, p, q) = \frac{|a|y^{ap-1} \left(1 - \left(\frac{y}{b}\right)^a\right)^{q-1}}{b^{ap} B(p, q)},$$

where $B(\cdot)$ denotes Euler's Beta function.

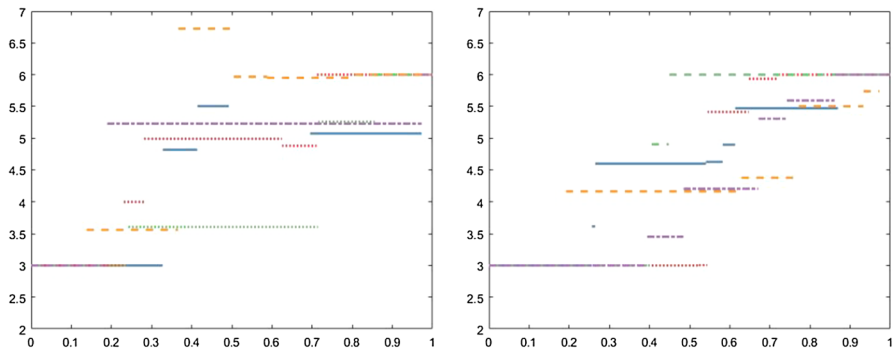


Fig. 4 5 sample paths of compound Poisson bridge processes ($\lambda = 3$). Left: normal jumps with parameters $\mu = 0.5$, $\sigma = 1$. Right: exponential jumps with mean $\gamma = 0.5$

Algorithm 1 A simulation scheme for compound Poisson bridge processes.

Given: X_{t_1}, X_{t_2}

Define: $c := X_{t_2} - X_{t_1}$, $C_k := c - \sum_{j=1}^{k-1} x_j$

I: Simulate a value n for the number of jumps N

1: Determine \tilde{N} such that

$\mathcal{N}(\mu, \sigma^2)$	$\text{Exp}(\gamma)$	$\text{Gam}(\alpha, \theta)$
$\tilde{N} > -\frac{\log\left(\frac{\varepsilon(1-e^{-c_2})}{c_1}\right)}{c_2} - 1$,	$\sum_{n=1}^{\tilde{N}} \frac{(\gamma c)^n}{(n-1)!} > c(1-\varepsilon)e^{\gamma c}$	$\sum_{n=1}^{\tilde{N}} \frac{(\theta c)^{n\alpha}}{\Gamma(n\alpha)} > c(1-\varepsilon)e^{\theta c}$
$c_1 = \frac{\mu c}{\sqrt{2\pi\sigma^2}}, c_2 = \frac{\mu^2}{2\sigma^2}$		

2: For $n = 1, \dots, \tilde{N}$ compute the conditional probabilities

$$\mathbb{P}[N = n | X_{t_2} - X_{t_1} = c] = \frac{f^{*n}(c) \cdot \frac{(\lambda(t_2 - t_1))^n}{n!}}{\sum_{m=1}^{\tilde{N}} f^{*m}(c) \cdot \frac{(\lambda(t_2 - t_1))^m}{m!}},$$

using the convolution properties $\mathcal{N}(\mu, \sigma^2) * \mathcal{N}(\mu, \sigma^2) = \mathcal{N}(2\mu, 2\sigma^2)$, $\text{Exp}(\gamma) * \text{Exp}(\gamma) = \text{Erl}(\gamma, 2)$, $\text{Gam}(\alpha, \theta) * \text{Gam}(\alpha, \theta) = \text{Gam}(2\alpha, \theta)$.

3: Obtain a value n by simulating from the resulting distribution, i.e., by computing its cdf and applying the inverse to a sample from the Uniform distribution on $[0, 1]$

II: Simulate the jumping times

1: Simulate n random variables from the Uniform distribution on $[t_1, t_2]$

III: Simulate the jump sizes

1: For $k = 1, \dots, n$, simulate from the following distributions:

$\mathcal{N}(\mu, \sigma^2)$	$\text{Exp}(\gamma)$	$\text{Gamma}(\alpha, \theta)$
$\mathcal{N}\left(\frac{C_k}{n-k+1}, \left(\frac{n-k}{n-k+1}\right)\sigma^2\right)$	$\text{Lomax}(k-n, -C_k)$	$\text{GB1}(1, C_k, \alpha, (n-k)\alpha)$

Remark While efficient simulation of trajectories of compound Poisson bridges is indeed possible (given a tractable jump-size distribution), the distribution of the bridge process for some time $t \in (t_1, t_2)$ is generally an intractable object. Its cdf consists of the following terms:

$$\begin{aligned}
\hat{F}_{X_t}(x) &= \mathbb{P}[X_t \leq x | X_{t_2} - X_{t_1} = c] \\
&= \sum_{n=0}^{\infty} \sum_{m=0}^n \mathbb{P}\left[\sum_{i=0}^m X^i \leq x \mid X_{t_2} - X_{t_1} = c\right] \\
&\quad \cdot \mathbb{P}[N_t = m | N_T = n] \cdot \mathbb{P}[N_T = n] \\
&= \sum_{n=0}^{\infty} \sum_{m=0}^n \left(\int_0^{x - \sum_{i=0}^{m-1} y_i} \cdots \int_0^{x - y_1} \int_0^x f_1(y_1) \cdots f_{m-1}(y_{m-1}) dy_1 \cdots dy_{m-1} \right) \\
&\quad \cdot \binom{n}{m} \left(\frac{t}{t_2 - t_1} \right)^m \left(1 - \frac{t}{t_2 - t_1} \right)^{n-m} \cdot \frac{(\lambda(t_2 - t_1))^n}{n!} e^{-\lambda(t_2 - t_1)},
\end{aligned}$$

i.e., an infinite sum of the product of a Poisson distribution with parameter $\lambda(t_2 - t_1)$, a Binomial distribution with probability parameter $t/(t_2 - t_1)$, and a complicated multidimensional integral over the conditional densities (using a shorthand notation) given in (23).

Remark (Further Lévy processes) For most Lévy processes, the density function at a given future time is not available in (semi-)closed form. However, in some special cases, bridge processes turn out to be of a surprisingly tractable nature. In the dissertation of Hoyle [19], one can find results for $1/2$ -stable processes, Inverse Gaussian processes and Cauchy processes, which imply that a simulation of associated bridges can be performed in a straightforward way: In the first two cases, by applying a deterministic function to a random draw from the standard Normal distribution; in the third case, the cumulative distribution function is given in terms of standard functions.

4 Illustration by example

In this section, we discuss the proposed modeling approach by a prototypical example. Moreover, we report about an implementation in the context of a real-world industrial application. In order to focus on the essential characteristics of the class of multiscale stochastic optimization problems, we will keep the complexity of the purely illustrative example as simple as possible.

4.1 A simple inventory control problem

Consider a business where some (perishable) goods can be sold for a unit price a . The stock can be replenished each Monday morning for the price b per unit. During the week, the products are sold but the stock cannot be replenished. The demand varies. If the business runs out of stock, then costs c occur depending on the remaining time until the next opportunity to fill the stock. For products left in stock at the end of the week, we assume that only 30% can still be used for the next week, but 70% need to be thrown away.

As a model for the demand, we use the Vašíček model [see (18) in Sect. 3.1]. In particular, for the sake of simplicity we do not consider any seasonal patterns. Let the parameters of the Vašíček model be given by $\theta = 105$, $\kappa = 0.5$, $\sigma = 10$, and the

starting value $x_0 = 100$. Three-stage problems are the smallest instances involving all issues that are typically connected to multistage decision making under uncertainty. Hence, the objective in the subsequent illustrative example is to maximize expected profits over two upcoming weeks.

4.1.1 Modeling the problem

Denote the demand at time $s \in [0, 2]$ by (the continuous random variable) X_s , the stock level at the beginning of week $t \in \{1, 2\}$ by S_t , and the remaining stock level at the end of week t by R_t . One may interpret S_t as the post-decision state with respect to a decision π_{t-1} , which is made before the first random demand of week t is observed. On the other hand, R_t corresponds to the amount left in stock after all the demands of week t have been observed, i.e., R_t is the pre-decision state with respect to π_t . The state transition rule is given by $S_t = 0.3 \cdot R_{t-1} + \pi_{t-1}$, where we assume that the stock is empty in the beginning, that is $R_0 = 0$. We optimize over replenishing policies $\{\pi_0, \pi_1\}$. The profit during week $t \in \{1, 2\}$ is then given by

$$f_t(\pi_{t-1}, X_{t-1:t}) = \begin{cases} a \cdot (S_t - R_t) - b \cdot \pi_{t-1} & \text{if } R_t > 0 \\ a \cdot S_t - b \cdot \pi_{t-1} - c \cdot (1 - \tau_t) & \text{if } R_t = 0, \end{cases}$$

where

$$\tau_{t+1} := \inf \left\{ s \in [0, 1] : \int_t^{t+s} X_u \, du > S_t \right\}$$

represents the first time of week $t + 1$ when the business runs out of stock. Remaining stock items at the end of the planning horizon enter the model with their value, i.e., we add $0.3 \cdot R_2 \cdot a$ to $f_2(\pi_1, X_{1:2})$. The problem can then be summarized as

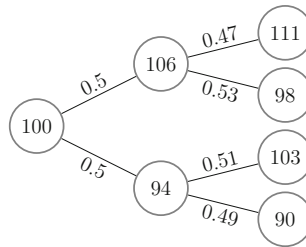
$$\left\| \begin{array}{l} \sup_{\pi} \sum_{t=1}^2 \mathbb{E}[f_t(\pi_{t-1}, X_{t-1:t})] \\ \text{s.t. } \pi \triangleleft \sigma(X), \end{array} \right.$$

where $\sigma(X)$ denotes the filtration generated by the demand process X . We set the problem parameters to $a = 10$, $b = 7$ and $c = 1000$. We observe the demand on an hourly basis, $24/7$.

The key observation here is that profits depend (in a highly nonlinear fashion) on the whole demand trajectory, while a replenishment decision for the stock can only be made once a week. The path-dependency is due to the presence of the stopping times τ_t in the objective. We apply our suggested methodology and generate a collection of paths between each pair of consecutive decision nodes. In such a way, expected profits during the week can be computed by a simple Monte Carlo simulation. The SDE describing the Vašíček bridge process is given in (19).

Table 1 Lattice via Algorithm 2.1—number of nodes at stage t , N -th iteration

$t \backslash N$	1	2	3	4	5
1	$\tilde{X}_4^{(1)}$ 9	$\tilde{X}_{16}^{(2)}$ 33	$\tilde{X}_{64}^{(3)}$ 129	$\tilde{X}_{256}^{(4)}$ 513	$\tilde{X}_{1024}^{(5)}$ 2049
2	$\tilde{X}_8^{(1)}$ 17	$\tilde{X}_{32}^{(2)}$ 65	$\tilde{X}_{128}^{(3)}$ 257	$\tilde{X}_{512}^{(4)}$ 1025	$\tilde{X}_{2048}^{(5)}$ 4097

**Fig. 5** A simple binary tree as a discrete model for the decision stages. The Vašíček model has been discretized by an optimal quantization algorithm

4.1.2 Discretization of decision stages

Following the approach suggested in Sect. 2.1, we choose the functions $H(x) = \sigma x$, $g(x) = \frac{\kappa}{\sigma}(\theta - x)$ and $\tau(x) = 1$, in case of the Vašíček model. Then, the lattice corresponding to the N -th iteration of Algorithm 2.1, discretizing the weekly decision stages $t = 0, 1, 2$, results in the discrete random variables $\tilde{X}_{t \cdot 4^N}^{(N)}$. The corresponding numbers of nodes on the lattice are shown in Table 1. Notice that this lattice construction serves for the discretization of the decision stages only. Hence, the probabilities of different paths between two stages, which end up in the same node, can be summed up and the intermediate nodes and paths do not have to be stored. If one wanted to store the full lattice construction, this would correspond to $(t \cdot 4^N + 1)^2$ nodes, $\frac{1}{2}(3^{t \cdot 4^N + 1} - 1)$ conditional probabilities, and a total of $3^{t \cdot 4^N}$ paths up to time t , for the N -th iteration.

For ease of exposition, we keep the discrete model as small as possible for the current example and focus on illustrating the suggested methodology regarding the multiscale issue. Thus, we use a simple binary tree for the decision stages, which we obtain by a standard optimal quantization algorithm. The tree is visualized in Fig. 5. For industrial applications, trees/lattices with a magnitude of 10^5 – 10^6 nodes are often used. The suggested lattice construction becomes increasingly attractive the more decision stages are involved.

4.1.3 Comparison with other modeling approaches

One might consider alternative discrete structures to model multiscale stochastic optimization problems. However, none of the approaches used for similar purposes in the stochastic programming literature is really comparable to what has been suggested in the present paper. This is due to the following reasons.

- *Large trees/lattices* In principle, one can understand any multiscale problem as a standard multistage problem, where the constraints rule out that decisions are made on the finer observation scale. After all, both scales are associated with the same underlying process. Then, one might simply use a very large tree/lattice model for the uncertainty process, which branches in each observation time. However, this will typically result in computational intractability. Even for the very small illustrative example discussed in this section, where there are only two decision points, hourly observation would already require a structure with 336 branching times. Any tree model would clearly explode even for much smaller instances. A ternary lattice model would involve more than 100 thousand nodes. Compared to our approach, it would require massive resources to construct such a lattice, store it, and compute a solution on it.
- *Reduced trees* If multistage problems grow too large to be modeled on regular scenario trees, it seems popular in the applied stochastic programming literature to use trees that only branch irregularly, i.e., certain branches remain constant up to/after a certain time (cf. [15–17,29]). However, this means to use clairvoyant branches where a computed policy does not reflect the uncertainty faced by the decision maker. In fact, using such degenerate trees violates the fundamentals of multistage stochastic optimization, which is exactly based on the idea of (direct) stochastic lookahead policies. Our approach, on the other hand, does not turn the decision maker clairvoyant up to/after a certain time and is hence perfectly aligned with the fundamental paradigm that a decision policy must reflect the uncertainty faced by the decision maker at any point in time. For our example, the reduction of a tree with 336 branching times to a computational instance would need to be so massive, that basically a fan with very few branchings would remain.
- *Deterministic interpolation function* Given a tree/lattice model for the information flow over the decision stages, one might simply choose a rudimentary interpolation approach, such as a constant or linear interpolation function, to compute the multi-period costs between decisions. However, this is inconsistent as both the decision and the observation scale are actually associated with one and the same uncertainty process. It would mean to completely remove the stochasticity between decisions, whereas our approach takes into account the random fluctuations along the way from one decision node to the other. In the context of our example, a constant interpolation would be completely meaningless, as it would correspond to assuming that all the selling activity occurs in a single instant of time each week. A linear interpolation would simply not be in line with the essence of the problem that one does not know in advance if/when one will run out of stock during the week.
- *Multi-horizon stochastic programming* A solution approach for a class of problems which are of a similar flavour, yet crucially different in nature, is called multi-horizon stochastic programming (see [21,27,41,42,46,48]). Infrastructure planning problems, being the original motivation by Kaut et al. [21], typically involve (rarely happening) *strategic* decisions as well as *operational* tasks (daily business). To overcome the above mentioned memory issue resulting from frequently branching scenario trees, the authors of [21] suggest to start with a tree for the strategic scale only. In a second step, they attach another tree to each node of the strategic-scale tree. The key assumption for the multi-horizon stochastic program-

ming approach to be appropriate is that the strategic scale and all operational scales are independent from each other. In contrast, the approach suggested in the present paper is designed for problems where the two scales are clearly related to the same uncertainty process. Therefore, our approach ensures that different scenarios in between consecutive decisions are eventually bundled in one node (by leveraging the theory of stochastic bridges). Moreover, for our illustrative example each of the “operational” trees would still require 168 branching times, such that again serious simplification would be required to make the approach computationally tractable.

To summarize the major strengths of our model, it

- respects the stochasticity in between decisions,
- ensures the consistency of all involved scales with respect to a single uncertainty process, and
- keeps the problem computationally tractable.

None of the other approaches mentioned above offers those three aspects, which are all essential characteristics of a useful modeling framework for the computational solution of multiscale stochastic optimization problems.

4.1.4 Numerical illustration

We have discussed above the qualitative strengths of our modeling approach. The simple example that we used to exemplify our explanations shall now serve to illustrate numerically two important aspects. First, an appropriate modeling approach is increasingly important the stronger is the path-dependency of the multiperiod costs. In our example, this path-dependency is higher, the larger is the value of the parameter c , which represents the costs that occur during the time span when the agent is out of stock. The second aspect is that the way how the multiperiod costs are modeled, does have a considerable impact on the resulting optimal value, even if the cost-structure does not depend heavily on a particular path. Even if we set $c = 0$, we observe an over-estimated value of almost 3% if we use an ad-hoc linear interpolation instead of our consistent modeling approach. Notice that this impact is related to a problem with only two decision stages.

Table 2 illustrates the above two observations with numbers. The second column shows the optimal values obtained using our modeling approach with bridge processes. The third column shows the values obtained using a simple linear interpolation rule for the intra-week evolution of the demand. How much this changes the optimal value in percent is given in the last column. Our implementation is based on a simple backwards dynamic programming algorithm. Between each pair of consecutive decision nodes, we have used 10k simulations of the bridge process.

4.2 A real-world application

We have implemented the modeling approach suggested in this paper in the context of an industrial project dealing with the valuation of a thermal power plant. While the

Table 2 Numerical illustration of the impact of using our modeling approach versus an inconsistent linear interpolation heuristic

c	Opt. value (bridge process)	Opt. value (linear interp.)	Impact (%)
0	562	577	+ 2.7
10	561	576	+ 2.7
100	556	571	+ 2.7
500	534	552	+ 3.4
1000	507	529	+ 4.34
5000	419	464	+ 10.7

focus of that project lied on the incorporation of model ambiguity into a value function approximation policy, the valuation problem itself was presented to us in the form of a classical multiscale stochastic optimization problem: operating plans for the power plant must be fixed on a weekly basis (for management purposes of all the involved resources), but the intra-week profits resulting from the most recent decision depend on (uncertain) market prices that are observed in 4-h blocks. It is thus required to model a weekly decision scale with 42 observation periods within each week.

A classical tree model over all observation periods would be intractable even for a single week. If we use a ternary lattice model, the first week already involves 1764 nodes. The second week requires 5292 additional nodes and modeling a quarter of a year with a ternary lattice involves about 300 thousand nodes in total. On the other hand, with our approach the lattice model is related to a much coarser time granularity, discretizing only the information flow on the weekly decision scale. Then, a time horizon of a quarter of a year involves only 196 nodes on a (ternary) lattice. Considering the finer observation scale, such a lattice involves 507 different arcs, along which an interpolation is required. An inconsistent interpolation approach would distort the expected costs in each such intra-week segment.

The multiscale modeling approach of the present paper proved to be very useful for this practically-sized problem. In fact, the power plant model—as it was presented to us by our industry partner—turned out to be of such a tractable form that expected intra-week costs could even be calculated by an analytical formula, based on the derived bridge process dynamics for the underlying uncertainty process. If a simulation is required, this obviously slows down the computation process. Still, the approach allows for a scenariowise decomposition with respect to the decision time scale, i.e., of the tree/lattice model. Thus, computational tractability is typically not limited by the multiscale feature of a problem, when our modeling approach is applied.

The studied valuation problem involves an extensive model of the power plant and is based on real data provided to us by the operating energy company. Thus, we refer the reader to our separate paper [44] for all the details.

5 Conclusion

In this article, we have proposed a computational modeling framework for multi-stage stochastic optimization problems with embedded multiperiod problems. We have

named the subject of the study of this problem class *multiscale stochastic optimization*. The suggested approach is based on a separation between the (standard) multistage decision problem, and the problem of determining path-dependent costs between two consecutive decisions. The paper contains a contribution to both parts. One section was dedicated to the construction of scenario lattices as a discrete structure representing a time-homogeneous Markovian diffusion model. In particular, we examined a Markov-chain approximation approach and showed that the approximation error with respect to the optimal value of a generic multistage stochastic optimization problem can be controlled with the suggested methodology. In a second part, we suggested to leverage the theory of stochastic bridges in order to tackle the embedded multiperiod problem, which takes place on a much finer time-scale than the decision scale. We elaborated explicitly several examples of popular diffusion models and proposed a new simulation algorithm for compound Poisson bridges. A simple multiscale inventory control problem finally served to illustrate the proposed methodology and discuss it in the context of a concrete example. Moreover, we reported about an implementation as part of a real-world industrial project, where our approach turned out to be very convenient. The latter may be seen as a proof of concept.

Acknowledgements Open access funding provided by University of Vienna.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix

The Wasserstein distance

Definition 5.1 (*Wasserstein distance*) Let P and \tilde{P} be two probability measures on \mathbb{R}^m . Then, their Wasserstein distance is defined as

$$\mathfrak{W}(P, \tilde{P}) := \inf_{\pi \in \mathbb{X}(P, \tilde{P})} \iint_{\mathbb{R}^m \times \mathbb{R}^m} \|v - w\| \pi(dv, dw),$$

where $\mathbb{X}(P, \tilde{P})$ denotes the set of all probability measures on $\mathbb{R}^m \times \mathbb{R}^m$ with marginals P and \tilde{P} .

If P and \tilde{P} are probability measures on the real line, then their Wasserstein distance can be represented in terms of their cdf's F and \tilde{F} (Vallender [43]):

$$\mathfrak{W}(P, \tilde{P}) = \int |F(x) - \tilde{F}(x)| dx.$$

References

1. Applebaum, D.: Lévy Processes and Stochastic Calculus. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge (2004)
2. Asmussen, S., Albrecher, H.: Ruin Probabilities. Advanced Series on Statistical Science & Applied Probability. World Scientific, Singapore (2010)
3. Bally, V., Pagès, G.: A quantization algorithm for solving multidimensional discrete-time optimal stopping problems. *Bernoulli* **9**(6), 1003–1049 (2003). 12
4. Barczy, M.: Diffusion bridges and affine processes. Habilitation thesis, University of Debrecen, Hungary (2015)
5. Barndorff-Nielsen, O.E., Mikosch, T., Resnick, S.I.: Lévy Processes: Theory and Applications. Birkhäuser, Boston (2001)
6. Bertoin, J.: Lévy Processes. Cambridge Tracts in Mathematics. Cambridge University Press, Cambridge (1998)
7. Billingsley, P., Topsøe, F.: Uniformity in weak convergence. *Z. Wahrscheinlichkeitstheorie Verwandte Geb.* **7**(1), 1–16 (1967)
8. Bladt, M., Finch, S., Sørensen, M.: Simulation of multivariate diffusion bridges. *J. R. Stat. Soc. B* **78**(2), 343–369 (2016)
9. Bladt, M., Sørensen, M.: Simple simulation of diffusion bridges with application to likelihood inference for diffusions. *Bernoulli* **20**(2), 645–675 (2014). 05
10. Cont, R., Tankov, P.: Financial Modelling With Jump Processes. Chapman & Hall, Boca Raton (2004)
11. Cox, J.C., Ingersoll, J.E., Ross, S.: A theory of the term structure of interest rates. *Econometrica* **53**(2), 385–407 (1985)
12. Cox, S., Hutzenthaler, M., Jentzen, A.: Local Lipschitz continuity in the initial value and strong completeness for nonlinear stochastic differential equations. Technical report 2013-35, ETH Zurich (2013). [arXiv:1309.5595v2](https://arxiv.org/abs/1309.5595v2)
13. Fitzsimmons, P., Pitman, J., Yor, M.: Markovian bridges: construction, palm interpretation, and splicing. In: Çinlar, E., Chung, K.L., Sharpe, M.J. (eds.) *Seminar on Stochastic Processes*, 1992, pp. 101–134. Birkhäuser, Boston (1993)
14. Gonçalves, F.B., Roberts, G.O.: Exact simulation problems for jump-diffusions. *Methodol. Comput. Appl. Probab.* **16**(4), 907–930 (2014)
15. Growe-Kuska, N., Heitsch, H., Römisch, W.: Scenario reduction and scenario tree construction for power management problems. In: 2003 IEEE Bologna Power Tech Conference Proceedings, vol. 3 (2003)
16. Heitsch, H., Römisch, W.: Scenario tree modeling for multistage stochastic programs. *Math. Program.* **118**(2), 371–406 (2009)
17. Heitsch, H., Römisch, W.: Scenario tree reduction for multistage stochastic programs. *Comput. Manag. Sci.* **6**(2), 117–133 (2009)
18. Høyland, K., Wallace, S.W.: Generating scenario trees for multi-stage decision problems. *Manag. Sci.* **47**(2), 295–307 (2001). 2
19. Hoyle, A.E.V.: Information-based models for finance and insurance. Ph.D. thesis, Imperial College London (2010)
20. Karlin, S., Taylor, H.M.: A Second Course in Stochastic Processes, vol. 2. Elsevier, Amsterdam Science (1981)
21. Kaut, M., Midthun, K., Werner, A., Tomasgard, A., Hellemo, L., Fodstad, M.: Multi-horizon stochastic programming. *Comput. Manag. Sci.* **11**(1), 179–193 (2014)
22. Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Stochastic Modelling and Applied Probability. Springer, Berlin (2011)
23. Kovacevic, R., Pichler, A.: Tree approximation for discrete time stochastic processes: a process distance approach. *Ann. Oper. Res.* **235**(1), 395–421 (2015)
24. Löhndorf, N.: An empirical analysis of scenario generation methods for stochastic optimization. *Eur. J. Oper. Res.* **255**(1), 121–132 (2016)
25. Löhndorf, N., Wozabal, D.: Gas storage valuation in incomplete markets (2019). http://www.optimization-online.org/DB_FILE/2017/02/5863.pdf. Accessed 18 July 2019
26. Lyons, S.M.J.: Introduction to stochastic differential equations. Technical report, School of Informatics, University of Edinburgh (2013)

27. Maggioni, F., Allevi, E., Tomasgard, A.: Bounds in multi-horizon stochastic programs. *Ann. Oper. Res.* **12**, 1–21 (2018)
28. Maier, S., Pflug, G.Ch., Polak, J.W.: Valuing portfolios of interdependent real options under exogenous and endogenous uncertainties. *Eur. J. Oper. Res.* (2019). <https://doi.org/10.1016/j.ejor.2019.01.055>
29. Moriggia, V., Kopa, M., Vitali, S.: Pension fund management with hedging derivatives, stochastic dominance and nodal contamination. *Omega* **87**, 127–141 (2019)
30. Papaspiliopoulos, O., Roberts, G.: Importance sampling techniques for estimation of diffusion models. In: Kessler, M., Lindner, A., Sørensen, M. (eds.) *Statistical Methods for Stochastic Differential Equations*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability, Chapter 4, pp. 311–335. Taylor & Francis, Boca Raton (2012)
31. Pflug, G.Ch., Pichler, A.: A distance for multistage stochastic optimization models. *SIAM J. Optim.* **22**(1), 1–23 (2012)
32. Pflug, G.Ch., Pichler, A.: *Multistage Stochastic Optimization* Springer Series in Operations Research and Financial Engineering. Springer, Berlin (2014)
33. Pflug, G.Ch.: Scenario tree generation for multiperiod financial optimization by optimal discretization. *Math. Program.* **89**(2), 251–271 (2001)
34. Pflug, G.Ch., Swietanowski, A., Dockner, E.J., Moritsch, H.: The AURORA financial management system: model and parallel implementation design. *Ann. OR* **99**(1–4), 189–206 (2000)
35. Platen, E., Heath, D.: *A Benchmark Approach to Quantitative Finance*. Springer Finance. Springer, Berlin (2006)
36. Pollock, M.: On the exact simulation of (jump) diffusion bridges. In: *Proceedings of the 2015 Winter Simulation Conference, WSC '15*, pp. 348–359. IEEE Press, Piscataway, NJ (2015)
37. Revuz, D., Yor, M.: *Continuous Martingales and Brownian Motion*. Springer, Berlin (2004)
38. Sato, K.-I.: *Lévy Processes and Infinitely Divisible Distributions*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge (1999)
39. Schoutens, W.: *Lévy Processes in Finance: Pricing Financial Derivatives*. Wiley Series in Probability and Statistics. Wiley, New York (2003)
40. Schoutens, W., Cariboni, J.: *Lévy Processes in Credit Risk*. The Wiley Finance Series. Wiley, New York (2010)
41. Seljom, P., Tomasgard, A.: The impact of policy actions and future energy prices on the cost-optimal development of the energy system in Norway and Sweden. *Energy Policy* **106**(C), 85–102 (2017)
42. Skar, C., Doorman, G., Pérez-Valdés, G. A., Tomasgard, A.: A multi-horizon stochastic programming model for the European power system. Censes working paper 2/2016, NTNU Trondheim (2016). ISBN: 978-82-93198-13-0
43. Vallender, S.S.: Calculation of the Wasserstein distance between probability distributions on the line. *Theory Probab. Appl.* **18**(4), 3 (1974)
44. van Ackooij, W., Escobar, D., Glanzer, M., Pflug, G.Ch.: Distributionally robust optimization with multiple time scales: valuation of a thermal power plant (2019). http://www.optimization-online.org/DB_HTML/2019/04/7157.html. Accessed 18 July 2019
45. Vašíček, O.: An equilibrium characterization of the term structure. *J. Finan. Econ.* **5**(2), 177–188 (1977)
46. Werner, A.S., Pichler, A., Midthun, K.T., Hellemo, L., Tomasgard, A.: Risk Measures in Multi-horizon Scenario Trees, pp. 177–201. Springer, Boston (2013)
47. Woyczyński, W.A.: *Lévy Processes in the Physical Sciences*, pp. 241–266. Birkhäuser, Boston (2001)
48. Zhonghua, S., Egging, R., Huppmann, D., Tomasgard, A.: A multi-stage multi-horizon stochastic equilibrium model of multi-fuel energy markets. Censes working paper 2/2016, NTNU Trondheim (2015). ISBN: 978-82-93198-15-4