Comput Optim Appl (2015) 62:641–668
DOI 10.1007/s10589-015-9758-0

Dynamic generation of scenario trees

Georg Ch. Pflug[1,2] · **Alois Pichler**[3]

Received: 5 August 2014 / Published online: 9 May 2015

Abstract This paper presents new algorithms for the dynamic generation of scenario trees for multistage stochastic optimization. The different methods described are based on random vectors, which are drawn from conditional distributions given the past and on sample trajectories. The structure of the tree is not determined beforehand, but dynamically adapted to meet a distance criterion, which measures the quality of the approximation. The criterion is built on transportation theory, which is extended to stochastic processes.

Keywords Decision trees · Stochastic optimization · Optimal transportation

Mathematics Subject Classification 90C15 · 60B05 · 62P05

1 Introduction

Scenario trees are the basic data structure for multistage stochastic optimization problems. They are discretizations of stochastic processes and therefore an approximation to real phenomena. In this paper we describe general algorithms, which approximate the underlying stochastic process with an arbitrary, prescribed precision.

The traditional way *from data to tree models* is as follows:

(i) Historical time series data are collected,

✉ Alois Pichler
aloisp@ntnu.no

1 Department of Statistics and Operations Research, University of Vienna, Vienna, Austria

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

3 Norwegian University of Science and Technology, NTNU, Trondheim, Norway

(ii) a parametric model is specified for the probability law which governs the data process,
(iii) the parameters are estimated on the basis of the observations (and possibly some additional information),
(iv) future scenarios are generated according to the identified probability laws, and finally
(v) these scenarios are concentrated in a tree, typically by stepwise reduction.

In the last step a concept of closeness of scenarios and similarity between the simulated paths and the tree has to be used. Some authors use as a criterion for similarity the coincidence of moments (cf. Wallace et al. [13,14]), others use distance concepts such as the squared norm and a filtration distance (cf. Heitsch and Römisch and others [5,7–11]).

It has been shown in Pflug and Pichler [26] that an appropriate distance concept for stochastic processes and trees is given by the nested distance (see Definition 20 below). The relevant theorem for multistage stochastic optimization (cited as Theorem 21 below) is extended and simplified for the particular case of pushforward measures (Theorem 16). Based on transportation theory this paper presents in addition theorems which are particularly designed to extract scenario trees by employing techniques, which are adopted from stochastic approximation.

Shapiro [30] mentions that it is standard practice for scenario tree construction to sample at every stage conditionally on the scenarios generated at the previous stage. The trees obtained represent a reference model of the initial stochastic process. The paper [1] by Bally et al. considers Markovian processes as the Brownian diffusion processes and applies optimal quantization to construct a tree reference model for option pricing, while Kuhn [16,17] considers more general autoregressive processes. Our algorithms follow this idea as well, but in addition we construct scenario trees from observed paths, which may not have a common past.

A related issue is the choice of the tree topology, or the branching structure of the scenario tree: how bushy and how big should the approximating tree be in order not to exceed a given, maximal discretization error? The cited papers do not address this problem. The scenario reduction methods of Heitsch and Römisch are inspired by their work on squared norm and filtration distances, but do not give explicit error bounds.

We propose here a new way *from data to tree models* as follows:

(i) as above, historical time series data are collected,
(ii) a simulator has to be provided, which allows sampling trajectories from all conditional distributions of the estimated data process,
(iii) a threshold for the maximum discretization error has to be specified, then
(iv) our algorithms generate a tree with automatically chosen topology and maximal chosen discretization error.

The algorithms apply to stochastic processes with higher dimensional state space too, for which an appropriate distance has to be chosen. This is of relevance and importance in many economic applications.

The presented methods can be applied to approximate stochastic optimization problems with continuous-state scenarios by simpler ones defined on discrete-state scenario

trees. Pennanen et al. [12,21,22] investigate the convergence of related multistage stochastic programs.

Outline of the paper. The next section (Sect. 2) recalls the notion of transportation distances on probability spaces. Section 3 provides the mathematical basis for algorithms to approximate probability measures. These algorithms are based on stochastic approximation. Section 4 generalizes the results to stochastic processes and gives the related theorems for stochastic programming based on transportation distance. Section 5 introduces the nested distance and generalizes the results from transportation distances to the nested distance. Further, this section explains how scenario trees can be extracted from a set of trajectories of the underlying process. A series of examples demonstrates that it is possible to extract useful scenario trees even from a sample, which is smaller than the nodes of the scenario tree. In Sect. 6 we discuss the relevance of the algorithms and conclude.

2 Approximation of probability measures

It has been elaborated in a sequence of publications that the nested distance is an appropriate concept to provide a distance for stochastic processes, the basic theorem is provided below (Theorem 16). The nested distance is built on the transportation distance (sometimes also Wasserstein, or Katorovich distance), which is a distance for probability measures.

Numerical integration is easily executed for discrete measures. By the Kantorovich–Rubinstein duality theorem (cf. Villani [32, Theorem 1.14]), the Kantorovich distance provides the smallest possible error bound to compare expectations with respect to different measures. The Kantorovich distance, and the more general variant, the Wasserstein distance, are thus the adequate tool to provide adapted, discrete probability measures for numerical integration or expectation. In addition, the distances themselves can be handled numerically.

Having fast computations in mind we are interested in discrete approximating measures, which have as few as possible supporting points. The approach we outline here allows different weights for the supporting scenarios, as this optimal quantization further improves the approximating precision. Respective measures with equally weighted supporting points are, e.g., described in Pagés [20], but the same author provides also optimal quantizers in a series of related papers.

Definition 1 *(Transportation distance for probability measures)* Assume that P ($\tilde{P}$, resp.) are probability measures on probability spaces Ξ ($\tilde{\Xi}$, resp.), such that for $\xi \in \Xi$ and $\tilde{\xi} \in \tilde{\Xi}$ a *distance* $\mathsf{d}(\xi, \tilde{\xi})$ is defined. To the metric d one may associate the pertaining *Wasserstein distance of order* $r \geq 1$ of probability measures by

$$\mathsf{d}_r(P, \tilde{P}) := \inf \left\{ \left(\iint_{\Xi \times \tilde{\Xi}} \mathsf{d}(\xi, \tilde{\xi})^r \, \pi(d\xi, d\tilde{\xi}) \right)^{1/r} \,\middle|\, \begin{array}{l} \pi \text{ is a probability measure on } \Xi \times \tilde{\Xi} \\ \text{with marginal distributions } P \text{ and } \tilde{P} \end{array} \right\}. \tag{1}$$

The distance d_1 is also called *Kantorovich distance*.

The methods we develop in what follows consider a special probability measure $\tilde{P}$, which is associated with the initial measure P by a transport map.

Remark 2 The Wasserstein distance introduced here is based on the continuous function (a distance) d. For the theory of more general cost functions we refer, e.g., to Schachermayer et al. [2].

2.1 Transport maps

A particular situation arises if the second probability measure $\tilde{P}$ is a pushforward measure (or image measure) of P for some *transport map* $T : \Xi \to \tilde{\Xi}$ linking the spaces, $\tilde{P} = P^T := P \circ T^{-1}$. Then an upper bound for the Wasserstein distance is given by

$$\mathsf{d}_r\left(P, P^T\right)^r \leq \int_{\Xi} \mathsf{d}\left(\xi, T(\xi)\right)^r P(d\xi), \tag{2}$$

because the bivariate measure

$$\pi_T(A \times B) := P\left(A \cap T^{-1}(B)\right) \tag{3}$$

associated with T has the marginals required in (1).

The situation $\tilde{P} = P \circ T^{-1}$ naturally arises in approximations, where the outcome ξ is approximated by $T(\xi)$. Notice that if $T(\xi)$ is a close approximation of ξ, then $\mathsf{d}\left(\xi, T(\xi)\right)$ is small and the integral in (2) is small as well, which makes P^T an approximation of interest for P.

The upper bound (2) is useful in many respects. First, the measure π_T is computationally much easier to handle than a solution of (1), because the integral in (2) is just over Ξ, and not over the product $\Xi \times \tilde{\Xi}$ as in (1). Further, for $r = 2$, Brenier's polar factorization theorem [3,4] implies that the optimal transport plan π solving (1) has the general form (3) for some measure preserving map T, such that involving a transport map is not restrictive. Finally, the transport map allows a generalization to stochastic processes which we address in Sect. 4.2.

2.2 Single-period Wasserstein distance minimization

Assume that P and $\tilde{P}$ are probabilities on $\Xi = \mathbb{R}^m$, which is endowed with the distance

$$\mathsf{d}(\xi, \tilde{\xi}).$$

To the distance d one may associate the pertaining Wasserstein-distance according to (1) in Definition 1. Our goal is to approximate P by the "best" discrete multivariate distribution $\tilde{P}$ sitting on s points $z^{(1)}, \ldots, z^{(s)}$ in the sense that the transportation distance $\mathsf{d}_r(P, \tilde{P})$ is minimized.

Given a collection of points $Z = \left(z^{(1)}, \ldots, z^{(s)}\right)$, which can be collected in a $m \times s$ matrix, introduce the *Voronoi partition* $\mathcal{V}_Z = \left\{V_Z^{(i)} : i = 1, \ldots, s\right\}$ of $\mathbb{R}^m$, where

$$V_Z^{(i)} = \left\{ \xi \in \mathbb{R}^m \,\middle|\, \begin{array}{l} \mathsf{d}(\xi, z^{(i)}) = \min_j \mathsf{d}(\xi, z^{(j)}) \text{ and} \\ \mathsf{d}(\xi, z^{(k)}) > \min_j \mathsf{d}(\xi, z^{(j)}) \text{ for } k < i \end{array} \right\}$$

such that[1]

$$\biguplus_{i \in \{1,\ldots s\}} V_Z^{(i)} = \mathbb{R}^m.$$

For a given probability P on $\mathbb{R}^m$ we use the notation P_Z for the discrete distribution sitting on the points of the set Z with masses $P\left(V_Z^{(i)}\right)$, i.e.,

$$P_Z = \sum_{i=1}^{s} P\left(V_Z^{(i)}\right) \cdot \delta_{z^{(i)}}.$$

Remark 3 Notice that the measure P_Z is induced by the plan T, $P_Z = P^T$, where $T : \Xi \to Z \subset \Xi$ is the transport map

$$T(\xi) := z^{(i)}, \text{ if } \xi \in V_Z^{(i)}.$$

For a fixed P let

$$D(Z) := \int_\Xi \min_{i=1,\ldots s} \mathsf{d}\big(\xi, z^{(i)}\big)^r \, P(d\xi) = \sum_{i=1}^{s} \int_{V_Z^{(i)}} \mathsf{d}\big(\xi, z^{(i)}\big)^r \, P(d\xi). \tag{4}$$

Then

$$\begin{aligned} D(Z)^{1/r} &= \min\left\{\mathsf{d}_r(P, \bar{P}) : \ \bar{P}(Z) = 1\right\} \\ &= \min\left\{\mathsf{d}_r(P, \bar{P}) : \ \bar{P} \text{ sits on the points of the set } Z\right\} = \mathsf{d}_r(P, P_Z), \end{aligned} \tag{5}$$

such that $D(Z)$ measures the quality of the approximation of P, which can be achieved by probability measures with supporting points Z (cf. [6, Lemma 3.4]).

2.2.1 Facility location

The approximation problem is thus reduced to finding the best point set Z (the facility location problem) in (4). This problem is an unconstrained, non-convex, high dimensional (as optimal locations consist of s points in $\mathbb{R}^m$) optimization problem. In addition, the objective is not everywhere differentiable. In general, it is not possible to identify the (a) global solution of the facility location problem, but good locations are often sufficient for applications in stochastic optimization.

In the next section we discuss three algorithms for solving this minimization problem:

[1] The disjoint union $\biguplus_i V_i$ symbolizes that the sets are pairwise disjoint, $V_i \cap V_j = \emptyset$, whenever $i \neq j$.

(i) A deterministic iteration procedure, which is applicable, if the necessary integrations with respect to P can be carried out numerically.
(ii) A *stochastic approximation* procedure, which is based on a sample from P and which converges to a local minimum of D.
(iii) A branch-and-bound procedure, which is also based on a sample from P and which converges to a global minimum of D.

2.2.2 Asymptotics and choice of parameters

Locations, which minimize the objective (5) globally, are called *representative points*. It is a well-known result established by Dudley (cf. Graf and Luschgy [6]) that $D(Z)^{1/r}$ is asymptotically of order $s^{-1/m}$ independently of r. It is desirable to choose $r = 1$ or r small, because $\mathsf{d}_r(P, \bar{P}) \le \mathsf{d}_{r'}(P, \bar{P})$ whenever $r \le r'$ (a consequence of Hölder's inequality) to improve the approximation (5).

On the other hand, choosing the Euclidean distance $\mathsf{d}(\cdot, :) := \|\cdot - : \|$ and $r = 2$ often simplifies computations significantly. In particular it holds for linear random variables f that $\mathbb{E} f = \mathbb{E}_{P_Z} f$, if $P_Z = \sum_{i=1}^{s} P\left(V_Z^{(i)}\right) \cdot \delta_{z^{(i)}}$ and Z is a local minimum of (5) (cf. Graf and Luschgy [6, Remark 4.6]). Hence, (locally) optimal measures are *always* (even for $s = 1$ or $s \le d$) exact for the expectation of linear random variables. In addition, a transportation map is available in the case of $r = 2$, cf. Remark 5 below.

However, the choice of the underlying norm and the parameter $r \ge 1$ is often driven by the particular problem at hand, but $r = 2$ and the Euclidean norm is the preferable choice for linear problems.

3 Algorithms to approximate probability measures

Before introducing the algorithms we mention the differentiability properties of the mapping $Z \mapsto D(Z)$. This is useful as the first order conditions of optimality for (5) require the derivatives to vanish.

Let $\nabla D(Z)$ be the $m \times s$ matrix with column vector $\nabla_{z^{(i)}} D(Z)$ given by the formal derivative

$$\int_{V_Z^{(i)}} r\, \mathsf{d}\big(\xi, z^{(i)}\big)^{r-1} \cdot \nabla_{\xi} \mathsf{d}\big(\xi, z^{(i)}\big)\, P(d\xi), \qquad i = 1, \dots s \tag{6}$$

of (4).

Proposition 4 *If P has a density g with respect to the Lebesgue measure, then $Z \mapsto D(Z)$ is differentiable and the derivative is $\nabla D(Z)$.*

Proof Notice first that by convexity of the distance d the gradient $\nabla_{\xi} \mathsf{d}\big(\xi, z^{(i)}\big)$ in the integral (6) is uniquely determined except on a Lebesgue set of measure zero due to Rademacher's theorem. As P has a Lebesgue density the exception set has measure zero and the integral is well defined. That (6) is indeed the derivative follows by standard means, cf. also Pflug [23, Corollary 3.52, page 184].

3.1 The deterministic iteration

We start with a well known cluster algorithm for partitioning a larger set of points in $\mathbb{R}^m$ into s clusters. Algorithm 1 is a typical example of an algorithm, which clusters a given set of points into subsets of small intermediate distance. The algorithm can be used to generate representative scenarios. We use it only to find good starting configurations for the subsequent optimization algorithm.

Algorithm 1 A typical hierarchical cluster algorithm (complete linkage)

(i) **Sampling**. Suppose that n points $\left\{z^{(1)}, \ldots, z^{(n)}\right\}$ in $\mathbb{R}^m$ endowed with metric d is given. The set $Z = \left\{z^{(i)} : i = 1, \ldots, n\right\}$ is iteratively partitioned into disjoint clusters, such that their number decreases from step to step. At the beginning, each point is a cluster of itself.

(ii) **Iteration.** Suppose that the current partition of the set is $\mathbb{R}^m = \biguplus_j C_j$. Find the pair of clusters (C_j, C_k) for which

$$\sup\left\{\mathsf{d}(z, z') : z \in C_j,\ z' \in C_k\right\}$$

is minimal. Create a new cluster by merging C_j and C_k.

(iii) **Stopping criterion**. If the number of clusters has decreased to the desired number s, then stop. Otherwise goto (ii).

The subsequent single-period algorithm (Algorithm 2, a theory based heuristic) requires integration with respect to P, as well as nonlinear optimizations to be carried out numerically. Since this is a difficult task, especially for higher dimensions, we present an alternative algorithm based on stochastic algorithm later.

Algorithm 2 Discretization of the probability measure P by a discrete probability sitting on s points: a deterministic, but numerically difficult algorithm

(i) **Initialization.** Set $k = 0$ and start with an arbitrary point set $Z(0) = \left\{z^{(i)} : i = 1, \ldots s\right\}$. It is advisable to choose the initial point set according to a cluster algorithm, e.g., to use Algorithm 1 to find clusters and then start with the cluster medians.

(ii) **Voronoi partition.** Find the Voronoi sets $V^{(i)}_{Z(k)}$ for $1 \le i \le s$.

(iii) **Optimization step.** For all i compute the *center of order* r, i.e., let

$$z^{(i)}(k+1) \in \operatorname*{argmin}_y \left\{ \int_{V^{(i)}_{Z(k)}} \mathsf{d}(\xi, y)^r \ P(d\xi) \right\} \tag{7}$$

and form the new set $Z(k+1) = \left\{z^{(i)}(k+1) : i = 1, \ldots s\right\}$.

(iv) **Integration step.** Calculate $D\big(Z(k+1)\big)$. Stop, if $D(Z(k+1)) \ge D(Z(k))$; otherwise set $k := k+1$ and goto (ii).

Remark 5 To compute the argmin in the optimization step (iii) of Algorithm 2 is in general difficult. However, there are two important cases.

(i) Whenever $\mathbb{R}^m$ is endowed with the weighted Euclidean metric $\mathsf{d}\left(\xi,\tilde{\xi}\right)^2 = \sum_{j=1}^m w_j \left|\xi_j - \tilde{\xi}_j\right|^2$ and the order of the Wasserstein distance is $r = 2$, then the argmin is known to be the conditional barycenter, i.e., the conditional expected value

$$z^{(i)}(k+1) = \frac{1}{P\left(V^{(i)}_{Z(k)}\right)} \int_{V^{(i)}_{Z(k)}} \xi \; P(d\xi). \tag{8}$$

This is an explicit formula, which is available in selected situations of practical relevance. Computing (8) instead of (7) may significantly accelerate the algorithm.

(ii) Whenever $\mathbb{R}^m$ is endowed with the weighted ℓ^1-metric $\mathsf{d}\left(\xi,\tilde{\xi}\right) = \sum_{j=1}^m w_j \left|\xi_j - \tilde{\xi}_j\right|$, then $z^{(i)}(k+1)$ in (7) is the componentwise median of the probability P restricted to $V^{(i)}_{Z(k)}$. In general and in contrast to the Euclidean metric, no closed form is available here.

Remark 6 (Initialization) Whenever the probability measure is a measure on $\mathbb{R}^1$ with cumulative distribution function (cdf) G, then the quantiles

$$z^{(i)}(0) := G^{-1}\left(\frac{i-1/2}{s}\right) \qquad (i = 1, 2, \ldots s)$$

can be chosen as initial points for the Wasserstein distance in (i) of Algorithm 2. These points are optimal for the Kolmogorov distance, that is, they minimize $\sup_{z\in\mathbb{R}} \left|P\left((-\infty, z]\right) - \hat{P}_n\left((-\infty, z]\right)\right|$ for the measure P and the empirical measure $\hat{P}_n = \frac{1}{n}\sum_{i=1}^n \delta_{z^{(i)}}$.

For the Wasserstein distance of order $r \geq 1$ even better choices are

$$z^{(i)}(0) = G_r^{-1}\left(\frac{i-1/2}{s}\right) \qquad (i = 1, 2, \ldots s),$$

where G_r is the cdf with density $g_r \sim g^{1/1+r}$ (provided that a density g is available). This is derived in Graf and Luschgy [6, Theorem 7.5]. Their result is even more general and states that the optimal points $z^{(i)}$ asymptotically follow the density

$$g_r = \frac{g^{m/(m+r)}}{\int g^{m/(m+r)}}, \tag{9}$$

whenever the initial probability measure P on $\mathbb{R}^m$ has density g.

The following proposition addresses the convergence of the deterministic iteration algorithm.

Proposition 7 *If $Z(k)$ is the sequence of point sets generated by the deterministic iteration algorithm (Algorithm 2), then*

$$D\big(Z(k+1)\big) \le D\big(Z(k)\big).$$

If $D\big(Z(k^*+1)\big) = D\big(Z(k^*)\big)$ *for some* k^*, *then* $D\big(Z(k)\big) = D\big(Z(k^*)\big)$ *for all* $k \ge k^*$ *and*

$$\nabla_{z^{(i)}} D\big(Z(k^*)\big) = 0$$

for all i.

Proof Notice that

$$\begin{aligned} D(Z(k)) &= \int_{\Xi} \min_j \mathsf{d}(\xi, z^{(j)})^r \; P(d\xi) = \sum_{i=1}^{s} \int_{V_Z^{(i)}} \mathsf{d}\big(\xi, z^{(i)}(k)\big)^r \; P(d\xi) \\ &\ge \sum_{i=1}^{s} \int_{V_Z^{(i)}} \mathsf{d}\big(\xi, z^{(i)}(k+1)\big)^r \; P(d\xi) = \int_{\Xi} \min_j d\big(\xi, z^{(i)}(k+1)\big)^r \; P(d\xi) \\ &= D(Z(k+1)). \end{aligned}$$

If $D(Z(k^*+1)) = D(Z(k^*))$, then necessarily, for all i,

$$z_j^{(i)}(k) \in \operatorname*{argmin}_y \left\{ \int_{V_{Z(k)}^{(i)}} \mathsf{d}\,(\xi, y)^r \; P(d\xi) \right\},$$

which is equivalent to

$$\int_{V_Z^{(i)}} r\, \mathsf{d}\big(\xi, z^{(i)}\big)^{r-1} \cdot \nabla_{\xi} \mathsf{d}\big(\xi, z^{(i)}\big) P(d\xi) = 0 \text{ for all } i$$

by Proposition 4. Hence $\nabla_Z D\big(Z(k^*)\big) = 0$ and evidently, the iteration has reached a fixed point. □

Remark 8 We remark here that the method outlined in Algorithm 2 is related to the k-means method of cluster analysis (see, e.g., McQueen [18]).

3.2 Stochastic approximation

Now we describe how one can avoid the optimization and integration steps of Algorithm 2 by employing stochastic approximation to compute the centers of order r. The stochastic approximation algorithm (Algorithm 3) requires that we can sample an independent, identically distributed (i.i.d.) sequence

$$\xi(1), \ldots, \xi(n)$$

of vectors of arbitrary length n, each distributed according to P.[2]

[2] Generating random vectors can be accomplished by rejection sampling in $\mathbb{R}^m$, e.g., or by a standard procedure as addressed in the Appendix.

Proposition 9 *Suppose that $\mathfrak{F} = (\mathcal{F}_1, \mathcal{F}_2 \ldots)$ is a filtration and (Y_k) is a sequence of random variables, which are uniformly bounded from below and adapted to $\mathfrak{F}$. In addition, let (A_k) and (B_k) be sequences of nonnegative random variables also adapted to $\mathfrak{F}$. If $\sum_k B_k < \infty$ a.s. and the recursion*

$$\mathbb{E}[Y_{k+1}|\,\mathcal{F}_k] \le Y_k - A_k + B_k \tag{10}$$

is satisfied, then Y_k converges and $\sum_k A_k < \infty$ almost surely.

Proof Let $S_k := \sum_{\ell=1}^{k} B_\ell$ and $T_k := \sum_{\ell=1}^{k} A_\ell$. Then (10) implies that

$$\mathbb{E}\left[Y_{k+1} - S_k|\mathcal{F}_k\right] = \mathbb{E}\left[Y_{k+1} - S_{k-1}|\mathcal{F}_k\right] - B_k \le Y_k - S_{k-1} - A_k \le Y_k - S_{k-1}.$$

Hence $Y_{k+1} - S_k$ is a supermartingale, which is bounded from below and which converges a.s. by the supermartingale convergence theorem (cf. Williams [33, Chapter 11]). Since S_k converges by assumption, it follows that Y_k converges almost surely. Notice finally that (10) is equivalent to

$$\mathbb{E}\left[Y_{k+1} - S_k + T_k|\mathcal{F}_k\right] \le Y_k - S_{k-1} + T_{k-1},$$

and by the same reasoning as above it follows that $Y_{k+1} - S_k + T_k$ converges a.s., which implies that $T_k = \sum_{\ell=1}^{k} A_\ell$ converges a.s. □

Proposition 10 *Let $F(\cdot)$ be a real function defined on $\mathbb{R}^m$, which has a Lipschitz-continous derivative $f(\cdot)$. Consider a recursion of the form*

$$X_{k+1} = X_k - a_k f(X_k) + a_k R_{k+1} \tag{11}$$

with some starting point X_0, where $\mathbb{E}[R_{k+1}|R_1, \ldots, R_k] = 0$. If $a_k \ge 0$, $\sum_k a_k = \infty$ and $\sum_k a_k^2 \|R_{k+1}\|^2 < \infty$ a.s., then $F(X_k)$ converges. If further $\sum_k a_k R_{k+1}$ converges a.s., then $f(X_k)$ converges to zero a.s.

Proof Let $Y_k := F(X_k)$ and let K be the Lipschitz constant of f. Using the recursion (11) and the mean value theorem, there is a $\theta \in [0, 1]$ such that

$$\begin{aligned}
F(X_{k+1}) &= F(X_k) + f\big(X_k + \theta(-a_k f(X_k) + a_k R_{k+1})\big)^\top \cdot (-a_k f(X_k) + a_k R_{k+1}) \\
&\le F(X_k) + f(X_k)^\top \cdot (-a_k f(X_k) + a_k R_{k+1}) \\
&\quad + K \cdot \|-a_k f(X_k) + a_k R_{k+1}\|^2 \\
&\le F(X_k) - a_k \|f(X_k)\|^2 + a_k f(X_k)^\top R_{k+1} + 2K a_k^2 \|f(X_k)\|^2 \\
&\quad + 2K a_k^2 \|R_{k+1}\|^2 .
\end{aligned}$$

Taking the conditional expectation with respect to $R_1, \ldots, R_k$ one gets

$$\begin{aligned}\mathbb{E}\left[F(X_{k+1})|R_1,\ldots,R_k\right] &\le F(X_k) - a_k \|f(X_k)\|^2 + 2Ka_k^2 \|f(X_k)\|^2 \\ &\quad + 2Ka_k^2 \|R_{k+1}\|^2 \\ &\le F(X_k) - \frac{a_k}{2}\|f(X_k)\|^2 + 2Ka_k^2 \|R_{k+1}\|^2\end{aligned}$$

for k large enough. Proposition 9, applied for $Y_k = F(X_k)$, $A_k = \frac{a_k}{2}\|f(X_k)\|^2$ and $B_k = 2Ka_k^2\|R_{k+1}\|^2$, implies now that $F(X_k)$ converges and

$$\sum_k a_k \|f(X_k)\|^2 < \infty \; a.s. \tag{12}$$

It remains to be shown that $f(X_k) \to 0$ a.s. Since $\sum_k a_k = \infty$, it follows from (12) that $\liminf_k \|f(X_k)\| = 0$ a.s. We argue now pointwise on the set of probability 1, where $\sum_k a_k \|f(X_k)\|^2 < \infty$, $\liminf_k \|f(X_k)\| = 0$ and $\sum_k a_k R_k$ converges. Suppose that $\limsup_k \|f(X_k)\|^2 > 2\epsilon$. Let $m_\ell < n_\ell < m_{\ell+1}$ be chosen such that

$$\begin{aligned}&\|f(X_k)\|^2 > \epsilon \text{ for } m_\ell < k \le n_\ell \text{ and} \\ &\|f(X_k)\|^2 \le \epsilon \text{ for } n_\ell < k \le m_{\ell+1}.\end{aligned} \tag{13}$$

Let ℓ_0 be such large that

$$\sum_{k=m_{\ell_0}}^{\infty} a_k \|f(X_k)\|^2 \le \frac{\epsilon^2}{2K} \quad \text{and} \quad \left\|\sum_{k=s}^{t} a_k R_{k+1}\right\| < \frac{\epsilon}{2} \quad \text{for all } s,t \ge m_{\ell_0}.$$

Then, for $\ell \ge \ell_0$ and $m_\ell \le k \le n_\ell$, by the recursion (11) and (13), as well as the Lipschitz property of f,

$$\begin{aligned}\left\|f(X_{i+1}) - f(X_{m_\ell})\right\| &\le K\|X_{i+1} - X_{m_\ell}\| = K\left\|\sum_{k=m_\ell}^{i} a_k f(X_k) + a_k R_{k+1}\right\| \\ &\le K\sum_{k=m_\ell}^{i} a_k \|f(X_k)\| + K\left\|\sum_{k=m_\ell}^{i} a_k R_{k+1}\right\| \\ &\le \frac{K}{\epsilon}\sum_{k=m_\ell}^{i} a_k \|f(X_k)\|^2 + \frac{\epsilon}{2} < \epsilon.\end{aligned}$$

Since $\left\|f\left(X_{m_\ell}\right)\right\| \le \epsilon$ it follows that $\limsup_k \|f(X_k)\| \le 2\epsilon$ for every $\epsilon > 0$ and this contradiction establishes the result. □

The following result ensures convergence of an algorithm of stochastic approximation type, which is given in Algorithm 3 to compute useful approximating measures.

Algorithm 3 A *stochastic approximation* algorithm: discretization of a probability measure P by a probability measure sitting on s points

(i) **Initialization.** Sample n random variates from the distribution P. Use a cluster algorithm (e.g., Algorithm 1) to find s clusters. Set $k = 0$ and let $Z(0) = \left(z^{(1)}(0), \ldots, z^{(s)}(0)\right)$ be the cluster medians. Moreover, choose a nonnegative and nonincreasing sequence a_k such that

$$\sum_{k=1} a_k^2 < \infty \text{ and } \sum_{k=1} a_k = \infty.$$

(ii) **Iteration.** Use a new independent sample $\xi(k)$. Find the index i such that

$$\mathsf{d}\big(\xi(k), z^{(i)}(k)\big) = \min_{\ell} \mathsf{d}\big(\xi(k), z^{(\ell)}(k)\big),$$

set

$$z^{(i)}(k+1) := z^{(i)}(k) - a_k \cdot r\, \mathsf{d}\big(\xi(k), z^{(i)}(k)\big)^{r-1} \cdot \nabla_{\xi} \mathsf{d}\big(\xi(k), z^{(i)}(k)\big)$$

and leave all other points unchanged to form the new point set $Z(k+1)$.

(iii) **Stopping criterion.** Stop, if either the predetermined number of iterations are performed or if the relative change of the point set Z is below some threshold ϵ. If not, then set $k = k+1$ and goto (ii).

(iv) **Determination of the probabilities.** After having fixed the final point set Z, generate another sample $(\xi(1), \ldots, \xi(n))$ and find the probabilities

$$p_i = \frac{1}{n} \# \left\{ \ell : \mathsf{d}\left(\xi(\ell), z^{(i)}\right) = \min_k \mathsf{d}\left(\xi(\ell), z^{(k)}\right) \right\}.$$

The final, approximate distribution is $\tilde{P} = \sum_{i=1}^{s} p_i \cdot \delta_{z^{(i)}}$, and the distance is

$$\mathsf{d}_r\left(P, \tilde{P}\right)^r \simeq \frac{1}{n} \sum_{\ell=1}^{n} \min_k \mathsf{d}\left(\xi(\ell), z^{(k)}\right)^r.$$

Theorem 11 *Suppose that the step lengths a_k in Algorithm 3 satisfy*

$$a_k \geq 0, \sum_k a_k = \infty \text{ and } \sum_k a_k^2 < \infty.$$

Suppose further that the assumptions of Proposition 4 are fulfilled. If $Z(k)$ is the sequence of point sets generated by the stochastic approximation algorithm (Algorithm 3), then $D\big(Z(k)\big)$ converges a.s. and

$$\nabla_Z D\big(Z(k)\big) \to 0 \quad a.s.$$

as $k \to \infty$. In particular, if $D(Z)$ has a unique minimizer Z^*, then

$$Z(k) \to Z^* \quad a.s.$$

Proof The matrices $Z(k)$ satisfy the recursion

$$Z(k+1) = Z(k) - a_k \nabla_Z D\big(Z(k)\big) - a_k W(k)$$

with

$$W(k) = \sum_{i=1}^{s} \mathbb{1}_{V^{(i)}_{Z(k)}}\big(\xi(k)\big) \cdot r\, \mathsf{d}\big(\xi(k), z^{(i)}(k)\big)^{r-1} \cdot \nabla_{\tilde{\xi}} \mathsf{d}\big(\xi(k), z^{(i)}(k)\big) - \int_{V^{(i)}_{Z(k)}} r\, \mathsf{d}\big(\xi(k), z^{(i)}(k)\big)^{r-1} \cdot \nabla_{\tilde{\xi}} \mathsf{d}\big(\xi(k), z^{(j)}(k)\big) P(d\xi).$$

Notice that the vectors $W(k)$ are independent and bounded, $\mathbb{E}[W(k)] = 0$ and $\sum_i a_i W(i)$ converges a.s. Proposition 10 applied for $X_k = Z(k)$, $F(\cdot) = D(\cdot)$, $f(\cdot) = \nabla_Z D(\cdot)$ and $R_k = W(k)$ leads to the assertion. □

Remark 12 A good choice for the step sizes a_k in Algorithm 3 is

$$a_k = \frac{C}{(k+30)^{3/4}}.$$

These step sizes satisfy the requirements $\sum_k a_k = \infty$, the sequence a_k is nonincreasing and $\sum_k a_k^2 < \infty$.[3]

Remark 13 A variant of Algorithm 3 avoids determining the probabilities in the separate step (iv) but counts the probabilities p_i on the fly.

3.3 Global approximation

It was mentioned and it is evident that Algorithm 2 converges to a local minimum, which is possibly not a global minimum. There are also algorithms which find the globally optimal discretization. However, these algorithms are so complex that only very small problems, say to find two or three optimal points in $\mathbb{R}^2$ or $\mathbb{R}^3$, can be handled effectively. In addition, the probability measure P must have bounded support.

For the sake of completeness we mention such an algorithm which is able to provide a globally best approximating probability measure located on not more than s supporting points. Algorithm 4 produces successive refinements, which converge to a globally optimal approximation of the initial measure P.

4 Trees, and their distance to stochastic processes

In this section we give bounds for the objective value of stochastic optimization problems. By generalizing an important result from multistage stochastic optimization we

[3] The exponent $s = 3/4$ is a compromise between $s \le 1$ to obtain $\sum_{k=1} \frac{1}{k^s} = \infty$ and $s > \frac{1}{2}$ for $\sum_{k=1} \frac{1}{k^{2s}} < \infty$.

Algorithm 4 Optimal discretizetion of probability P by a probability $\tilde{P}$ sitting on s points: a global optimization algorithm.

- Suppose that the optimal configuration of s points in a bounded set (for simplicity the unit cube $[0,1]^m$ in $\mathbb{R}^m$) is to be found. The optimal configuration is an element of $[0,1]^{m\times s}$. At stage ℓ the unit cube is dissected into smaller cubes, say $[0,1]^m = \bigcup C_j$. By considering all selections $C_{j_1} \times C_{j_2} \times \ldots \times C_{j_s}$ a dissection of the search space is defined. The "local" problem finds a stochastic lower and a stochastic upper bound for

$$\min_{z^{(i)}\in C_{j_i}} \int_\Xi \min_i \mathsf{d}\left(u, z^{(i)}\right)\, P(du).$$

- **Bounding step.** Configurations which have a lower bound larger than the upper bound of another configuration are excluded and not investigated further.
- **Branching step.** The best configuration will be refined by dissecting the pertaining cubes into smaller cubes.
- **Stopping.** If the gap between the upper bound and the lower bound is small enough, then stop.

provide bounds first when the law of the underlying process is approximated by a process with a pushforward measure.

The goal is to construct a valuated probability tree, which represents the process $(\xi_t)_{t=0}^T$ in the best possible way. Trees are represented by a tuple consisting of the treestructure (i.e., the predecessor relations), the values of the process sitting on the nodes and the (conditional) probablities sitting on the arcs of the tree. To be more precise, let $\mathbb{T} = (n, \text{pred}, z, Q)$ represent a tree with

- n nodes;
- a function pred mapping $\{1, 2, \ldots, n\}$ to $\{0, 1, 2, \ldots, n\}$. $\text{pred}(k) = \ell$ means that node ℓ is a direct predecessor of node k. The root is node 1 and its direct predecessor is formally encoded as 0;
- a valuation $z_i \in \mathbb{R}^m$ of each node $i \in \{1, 2, \ldots, n\}$;
- the conditional probability $Q(i)$ of reaching node i from its direct predecessor; for the root we have $Q(1) = 1$.

It is always assumed that these parameters are consistent, i.e., that they form a tree of height T, meaning that all leaves of the tree are at the same level T. The distance of each node to the root is called the *stage* of the node. The root is at stage 0 and the leaves of the tree are at stage T.

Let $\tilde{\Omega}$ be the set of all leaf nodes, which can be seen as a probability space carrying the unconditional probabilities $P(n)$ to reach the leaf node $n \in \tilde{\Omega}$ from the root. Obviously the unconditional probability $\tilde{P}(i)$ of any node i is the product of the conditional probabilities of all its predecessors (direct and indirect).

Let $\text{pred}_t(n)$ denote the predecessor of node n at stage t. These mappings induce a filtration $\tilde{\mathfrak{F}} = (\tilde{\mathcal{F}}_0, \ldots, \tilde{\mathcal{F}}_T)$, where $\tilde{\mathcal{F}}_t$ is the sigma-algebra induced by pred_t. $\tilde{\mathcal{F}}_0$ is the trivial sigma-algebra and $\tilde{\mathcal{F}}_T$ is the power set of $\tilde{\Omega}$. The process $(\tilde{\xi}_t)$ takes the values z_i for all nodes i at stage t with probability $\tilde{P}(i)$.

On the other hand, also the basic stochastic process (ξ_t) is defined on a filtered probability space $\left(\Omega, \mathfrak{F} = (\mathcal{F}_0, \ldots, \mathcal{F}_T), P\right)$, where $\mathcal{F}_0$ is the trivial sigma-algebra. Via the two stochastic processes, the basic process (ξ_t) defined on Ω and its discretization

$(\tilde{\xi}_t)$ defined on $\tilde{\Omega}$, a *distance* between $u \in \Omega$ and $v \in \tilde{\Omega}$ is defined by

$$\mathsf{d}(u, v) = \sum_{t=1}^{T} \mathsf{d}_t \left(\xi_t(u), \tilde{\xi}_t(v) \right), \tag{14}$$

where d_t are distances on $\mathbb{R}^{m_t}$ (the state $\mathbb{R}^{m_t}$ may depend on the stage t, but to keep the notation simple we consider only $\mathbb{R}^m$ processes in what follows, i.e., $m_t = m$ for all t). To measure the quality of the approximation of the process (ξ_t) by the tree $\mathbb{T}$ one may use the nested distance (see Definition 20 below) or its simpler variant, a stagewise transportation bound.

4.1 Approximation of stochastic processes

Different algorithms have been demonstrated in the previous sections to construct a probability measure $\tilde{P} = \sum_{i=1}^{s} p_i \delta_{z_i}$ approximating P. The approximating measures $\tilde{P}$ presented are all induced by the transport map

$$\begin{aligned} T : \Xi &\to Z \\ \xi &\mapsto z_i, \text{ if } \xi \in V_Z^{(i)}. \end{aligned}$$

It holds moreover that $V_Z^{(i)} = \{T = z_i\}$ (and in particular $P\left(V_Z^{(i)}\right) = P(T = z_i)$), which shows that the facility location problems can be formulated by involving just transport maps (cf. Remark 3).

In what follows we generalize the concept of transport maps to stochastic processes. We generalize a central theorem in stochastic optimization, which provides a bound in terms for the pushforward measure for transport maps. We demonstrate that an adequately measurable, finitely valued transport map represents a tree. Further, we employ stochastic approximation techniques again to find a useful tree representing a process. The methods allow computing bounds for the corresponding stochastic optimization problem.

4.2 The main theorem of stochastic optimization for pushforward measures

Consider a stochastic process $\xi = (\xi_t)_{t=0}^{T}$, which is discrete in time. Each component $\xi_t : \Omega \to \Xi_t$ has the state space Ξ_t (which may be different for varying t's). Further let $\Xi := \Xi_0 \times \ldots \Xi_T$ and observe that Ξ_t is naturally embedded in Ξ.

Definition 14 We say that a process $x = (x_t)_{t=0}^{T}$ (with $x_t : \Xi \to \mathbb{X}_t$) is *nonanticipative* with respect to the stochastic process $\xi = (\xi_t)_{t=0}^{T}$, if x_t is measurable with respect to the sigma algebra $\sigma\left(\xi_0, \ldots \xi_t\right)$. We write

$$\begin{aligned} x \lhd \sigma(\xi), \quad &\text{if } x_t \text{ is measurable with respect to the} \\ &\text{sigma algebra } \sigma\left(\xi_0, \ldots \xi_t\right) \text{ for every } t = 0, \ldots T. \end{aligned}$$

It follows from the Doob–Dynkin Lemma (cf. Shiryaev [31, Theorem II.4.3]) that a process x is nonanticipative, if there is a measurable function (denoted x_t again), such that $x_t = x_t(\xi_0, \dots \xi_t)$, i.e., $x_t(\omega) = x_t\big(\xi_0(\omega), \dots \xi_t(\omega)\big)$ for all t.

Definition 15 A transport map $T : \Xi \to \tilde{\Xi}$ is *nonanticipative* if

$$T \circ \xi \lhd \sigma(\xi),$$

that is, each component $T(\xi)_t \in \tilde{\Xi}_t$ depends on the vector $(\xi_0, \dots \xi_t)$, but not on $(\xi_{t+1}, \dots \xi_T)$, i.e, $T(\xi_0, \dots \xi_T)_t = T(\xi_0, \dots \xi_t, \xi'_{t+1}, \dots \xi'_T)_t$ for all $(\xi'_{t+1}, \dots \xi'_T)$ and $t = 0, \dots T$.

We consider first the stochastic optimization problem

$$\min \{\mathbb{E}_P [Q(x, \xi)] : \ x \in \mathbb{X}, \ x \lhd \sigma(\xi)\}, \tag{15}$$

where the decision x is measurable with respect to the process ξ, $x \lhd \sigma(\xi)$.

The following theorem generalizes an important observation (cf. [26, Theorem 11]) to image measures. This outlines the central role of a nonanticipative transport map in stochastic optimization.

Theorem 16 (Stagewise transportation bound) *Let $\mathbb{X}$ be convex and the $\mathbb{R}$-valued function $\tilde{Q} : \mathbb{X} \times \tilde{\Xi} \to \mathbb{R}$ be uniformly convex in x, that is,*

$$\tilde{Q}\big((1-\lambda)x_0 + \lambda x_1, \ \tilde{\xi}\big) \le (1-\lambda)\tilde{Q}\left(x_0, \tilde{\xi}\right) + \lambda \tilde{Q}\left(x_1, \tilde{\xi}\right) \qquad (\tilde{\xi} \in \tilde{\Xi}).$$

Moreover, let $Q : \mathbb{X} \times \Xi \to \mathbb{R}$ be linked with $\tilde{Q}$ by

$$\left|Q\big(x, \xi\big) - \tilde{Q}\left(x, \tilde{\xi}\right)\right| \le c\big(\xi, \tilde{\xi}\big) \quad \text{for all } \xi \in \Xi \text{ and } \tilde{\xi} \in \tilde{\Xi}, \tag{16}$$

where $c : \Xi \times \tilde{\Xi} \to \mathbb{R}$ is a function (called cost function).

Then for every nonanticipative transport map

$$T : \Xi \to \tilde{\Xi}$$

it holds that

$$\left|\inf_{x \lhd \sigma(\xi)} \mathbb{E}_P \, Q\big(x(\xi), \ \xi\big) - \inf_{\tilde{x} \lhd \sigma(\tilde{\xi})} \mathbb{E}_{P^T} \tilde{Q}\left(\tilde{x}\left(\tilde{\xi}\right), \tilde{\xi}\right)\right| \le \mathbb{E}_P \, c\big(\xi, T(\xi)\big). \tag{17}$$

Remark 17 Equation (17) relates the problem (15), the central problem of stochastic optimization, with another stochastic optimization problem on the image measure P^T. The problem on P^T may have a different objective ($\tilde{Q}$ instead of Q), but it is easier to solve, as it is reduced to the simpler probability space with pushforward measure P^T instead of P.

The right hand side of (17), $\mathbb{E}_P \, c\big(\xi, T(\xi)\big)$, is notably not a distance, but an expectation of the cost function c in combination with the transport map T.

Remark 18 In a typical application of Theorem 16 one has that $\tilde{\Xi} \subset \Xi$ and $\tilde{Q}(\cdot) = Q(\cdot)$. Further, $c\big(\xi, \tilde{\xi}\big) = L \cdot \mathsf{d}\big(\xi, \tilde{\xi}\big)$, where d is a distance on $\Xi \times \Xi$ and L, by means of (16), is a Lipschitz constant for the objective function Q.

Proof of Theorem 16 First, let $\tilde{x}$ be any feasible policy with $\tilde{x} \lhd \sigma(\tilde{\xi})$, that is, $\tilde{x}_t = \tilde{x}_t\left(\tilde{\xi}_0, \ldots \tilde{\xi}_t\right)$ for all t. It follows from the measurability of the transport map T that the derived policy $x := \tilde{x} \circ T$ is nonanticipative, i.e., $x \lhd \sigma(\xi)$. By relation (16) it holds for the policy x that

$$\mathbb{E}Q\left(x(\xi), \xi\right) = \mathbb{E}Q\big(\tilde{x}(T(\xi)), \xi\big) \leq \mathbb{E}\tilde{Q}\big(\tilde{x}(T(\xi)), T(\xi)\big) + \mathbb{E}c\big(\xi, T(\xi)\big),$$

and by change of variables thus

$$\mathbb{E}Q\left(x(\xi), \xi\right) \leq \mathbb{E}_{P^T}\tilde{Q}\left(\tilde{x}(\tilde{\xi}), \tilde{\xi}\right) + \mathbb{E}c\left(\xi, T(\xi)\right).$$

One may pass to the infimum with respect to $\tilde{x}$ and it follows, as $x = \tilde{x} \circ T \lhd \sigma(\xi)$, that

$$\inf_{x \lhd \sigma(\xi)} \mathbb{E}Q\left(x(\xi), \xi\right) \leq \inf_{\tilde{x} \lhd \sigma(\tilde{\xi})} \mathbb{E}_{P^T}\tilde{Q}\left(\tilde{x}(\tilde{\xi}), \tilde{\xi}\right) + \mathbb{E}c\left(\xi, T(\xi)\right). \tag{18}$$

For the converse inequality suppose that a policy $x \lhd \sigma(\xi)$ is given. Define

$$\tilde{x} := \mathbb{E}\left(x|T\right), \text{ i.e., } \tilde{x}_t(\tilde{\xi}) := \mathbb{E}\left(x_t \middle| T_t(\cdot) = \tilde{\xi}\right)$$

(Figure 1 visualizes the domain and codomain of this random variable) and note that $\tilde{x} \lhd \sigma(T(\xi))$ by construction and as T is nonanticipative.

As the function $\tilde{Q}$ is convex it follows from Jensen's inequality, conditioned on $\left\{T(\cdot) = \tilde{\xi}\right\}$, that

$$\begin{aligned}
\tilde{Q}\left(\tilde{x}\left(\tilde{\xi}\right), \tilde{\xi}\right) &= \tilde{Q}\left(\mathbb{E}\left(x|T\right)\left(\tilde{\xi}\right), \tilde{\xi}\right) \\
&= \tilde{Q}\left(\mathbb{E}\left(x(\xi) \middle| T(\xi) = \tilde{\xi}\right), \tilde{\xi}\right) \\
&\leq \mathbb{E}\left(\tilde{Q}\left(x(\xi), \tilde{\xi}\right) \middle| T(\xi) = \tilde{\xi}\right).
\end{aligned}$$

By assumption (16) linking Q and $\tilde{Q}$ it holds further that

$$\begin{aligned}
\tilde{Q}\left(\tilde{x}\left(\tilde{\xi}\right), \tilde{\xi}\right) &\leq \mathbb{E}\left(\tilde{Q}\left(x(\xi), \tilde{\xi}\right) \middle| T(\xi) = \tilde{\xi}\right) \\
&= \mathbb{E}\left(\tilde{Q}\big(x(\xi), T(\xi)\big) \middle| T(\xi) = \tilde{\xi}\right) \\
&\leq \mathbb{E}\left(Q\big(x(\xi), T(\xi)\big) + c\big(\xi, T(\xi)\big) \middle| T(\xi) = \tilde{\xi}\right) \\
&= \mathbb{E}\left(Q\big(x(\xi), T(\xi)\big) \middle| T(\xi) = \tilde{\xi}\right) + \mathbb{E}\left(c\big(\xi, T(\xi)\big) \middle| T(\xi) = \tilde{\xi}\right),
\end{aligned}$$

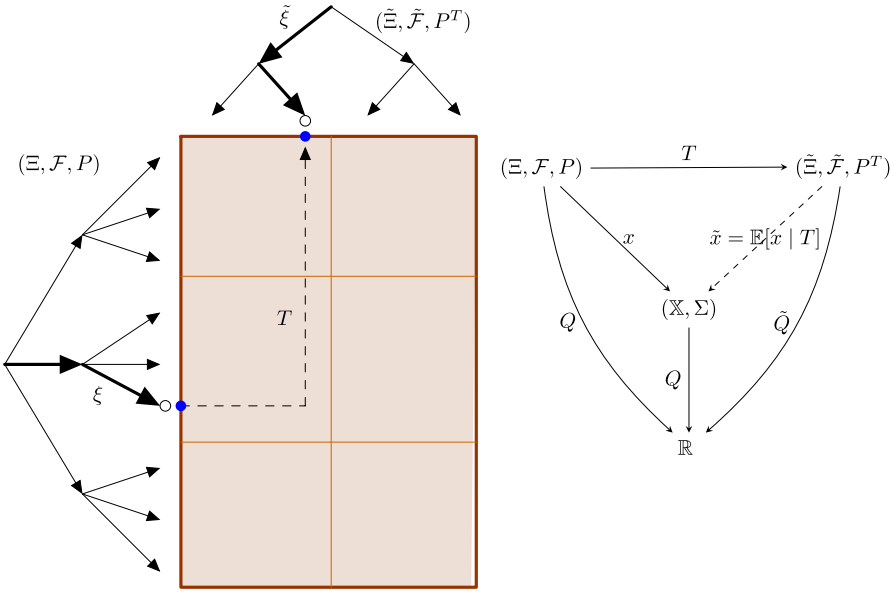

Fig. 1 *Left* the measurable transport map T, mapping trajectories ξ to $\tilde{\xi} = T(\xi)$. *Right* the diagram displays the domain and codomain of the functions involved. The diagram is commutative on average

and by taking expectations with respect to the measure P^T it follows that

$$\mathbb{E}_{P^T} \tilde{Q}\left(\tilde{x}\left(\tilde{\xi}\right), \tilde{\xi}\right) \le \mathbb{E}\, Q\big(x(\xi), T(\xi)\big) + \mathbb{E}\, c\big(\xi, T(\xi)\big).$$

Recall that $x \lhd \sigma(\xi)$ was arbitrary, by taking the infimum it follows thus that

$$\inf_{\tilde{x} \lhd \sigma(T((\xi)))} \mathbb{E}_{P^T} \tilde{Q}\left(\tilde{x}\left(\tilde{\xi}\right), \tilde{\xi}\right) \le \inf_{x \lhd \sigma(\xi)} \mathbb{E}\, Q\big(x(\xi),\ \xi\big) + \mathbb{E}\, c\big(\xi, T(\xi)\big).$$

Together with (18) this is the assertion. □

4.3 Approximation by means of a pushforward measure

In this section we construct a tree by establishing a transport map $T : \Xi \to \tilde{\Xi}$ with the properties of Definition 15. The algorithm is based on stochastic approximation and extends Algorithm 3, as well as an algorithm contained in Pflug [24]. We do not require more than a sample of trajectories (i.e., scenarios). The scenarios may result from observations or from simulation.

Algorithm 5 is the tree equivalent of Algorithm 3. It uses a sample of trajectories to produce a tree approximating the process ξ. The algorithm further provides the estimate $\mathbb{E}\, c(\xi, T(\xi))$, which describes the quality of the approximation of the tree $T \circ \xi$ in comparison with the original process ξ.

Algorithm 5 Generation of a tree with pre-specified structure by stochastic approximation (based on Algorithm 3)

(i) **Initialization.** Set $k = 0$, let $c_E = 0$ set the counters $c(n) = 0$ and let $Z^{(0)}(n) \in \Xi_t$ be chosen for each node n of the tree. Moreover, choose a nonnegative and nonincreasing sequence a_k such that

$$\sum_{k=1} a_k^2 < \infty \text{ and } \sum_{k=1} a_k = \infty.$$

(ii) **Iteration.** Use a new independent trajectory

$$\xi(k) = \big(\xi_0(k), \ldots, \xi_T(k)\big)$$

with law P.
Find a trajectory of successive nodes $n_0, n_1, \ldots n_T$ in the tree with $n_t = \text{pred}_t(n_{t+1})$ such that

$$n_t \in \operatorname*{argmin}_{n' \in \mathcal{N}_t(n_0, \ldots n_{t-1})} \mathsf{d}_t\big(\xi_t(k),\ Z^{(k)}(n')\big),$$

where $\mathcal{N}_t(n_0, \ldots n_{t-1})$ collects all nodes at stage t with predecessors $n_0, \ldots n_{t-1}$. Assign the new values

$$Z^{(k)}(n_t) := Z^{(k-1)}(n_t) - a_k \cdot r\, \mathsf{d}_t\big(\xi_t(k), Z^{(k-1)}(n_t)\big)^{r-1} \cdot \nabla_{\tilde{\xi}} \mathsf{d}_t\big(\xi(k)_t, Z^{(k-1)}(n_t)\big),$$

increase the counters $c(n_t) = c(n_t) + 1$ for the nodes $n_0, n_1, \ldots n_t$ and set $c_E := c_E + \Big(\sum_{t=0}^{T} \mathsf{d}_t\big(\xi(k)_t, Z^{(k-1)}(n_t)\big)\Big)^r$. For the other nodes let the values unchanged, i.e., $Z^{(k)}(n) := Z^{(k-1)}(n)$ whenever $n \notin \{n_0, n_1, \ldots n_T\}$.

(iii) **Stopping criterion.** Stop, if the predetermined number of iterations is performed. If not, then set $k = k + 1$ and goto (ii).

(iv) Set the conditional probabilities $p(n) = c(n)/N$, where N is the total number of random draws. The quantity $\mathbb{E}\mathsf{d}\big(\xi, T(\xi)\big)^r$ is estimated by

$$\mathbb{E}\,\mathsf{d}\left(\xi, T^n(\xi)\right)^r \simeq \frac{1}{N} c_E. \tag{19}$$

Example 19 To demonstrate Algorithm 5 we consider a standard Gaussian random walk in three stages first. The tree with bushiness $(10, 5, 2)$, found after 1000 and 100,000 samples, is displayed in Fig. 2 (left plots). The probability distribution of the leaves is annotated in the plots. The final distribution of the initial process is $N(0, 3)$ (Fig. 2a). The Gaussian walk and the tree are at a distance of $\big(\mathbb{E}\mathsf{d}(\xi, T(\xi))^2\big)^{1/2} \simeq 0.084$, where we have employed the usual Euclidean distance and $r = 2$.

The process which we also consider is the running maximum

$$M_t := \max\left\{\sum_{i=1}^{t'} \xi_i \,:\, t' \le t\right\} \text{ with } \xi_i \sim N(0, 1) \text{ i.i.d.} \tag{20}$$

Note, that the running maximum is *not* a Markovian process. The results of Algorithm 5 are displayed in Fig. 2 (right) for a bushiness of $(3, 3, 3, 2)$. The running maximum process and the tree in Fig. 2c have distance $\big(\mathbb{E}\,\mathsf{d}(\xi, T(\xi))^2\big)^{1/2} \simeq 0.13$.

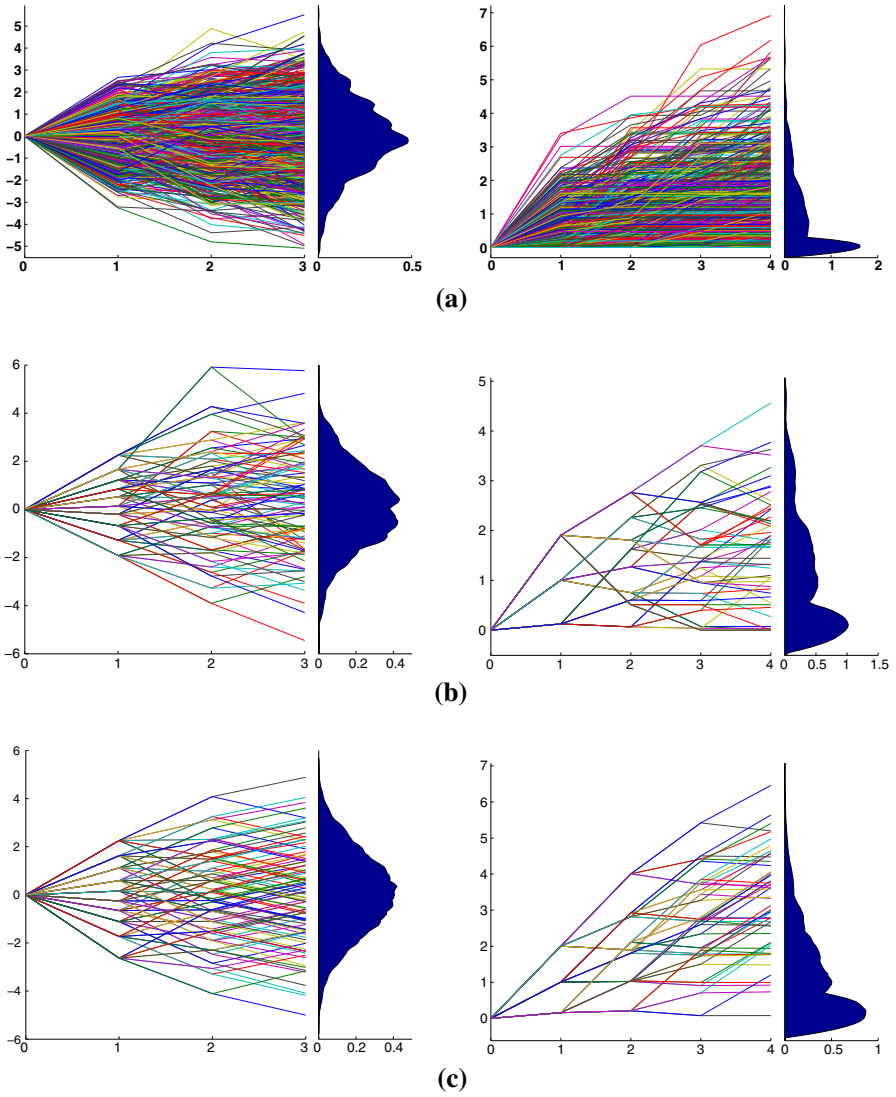

Fig. 2 Trees produced by Algorithm 5 after 1000 (Figure 2b) and 100,000 (Fig. 2c) samples. Annotated is a density plot of the probability distribution at the final stage. **a** 1000 sample paths of the standard Gaussian random walk and the (non-Markovian) running maximum process. **b** Trees with bushiness (10, 5, 2) and (3, 3, 3, 2) approximating the process in Fig. 2a. **c** The transportation bound to the underlying Gaussian process is 0.084, the transportation bound to the non-Markovian running maximum process is 0.13

5 The nested distance

In the previous section we have proposed an algorithm to construct an approximating tree from observed sample paths by stochastic approximation. It is an essential observation that the algorithm proposed establishes a measure $\tilde{P}$ on the second process

which is induced by a nonanticipative transport map T, $\tilde{P} = P^T$. It is a significant advantage of Theorem 16 that the bound

$$\mathbb{E}\, c\big(\xi, T(\xi)\big) \tag{21}$$

in equation (17) is very cheap to compute (Eq. (19) in Algorithm 5, e.g., provides $\mathbb{E}\, c\big(\xi, T(\xi)\big)$ as a byproduct). However, the algorithm is not designed for a prespecified measure $\tilde{P}$ on the second process.

In the general situation the quantity (21) is not symmetric, that is, there does not exist a transportation map $\tilde{T}$, say, such that $\mathbb{E}\, c\big(\xi, T(\xi)\big) = \mathbb{E}\, c\big(\tilde{T}(\tilde{\xi}), \tilde{\xi}\big)$. For this (21) does not extend to a distance of processes, as is the case for the Wasserstein distance for probability measures.

The nested distance was introduced to handle the general situation. In what follows we recall the definition and cite the results, which are essential for tree generation. Then we provide algorithms again to construct approximating trees, which are close in the nested distance.

Definition 20 *(The nested distance, cf.* [25]*)* Assume that two probability models

$$\mathbb{P} = (\Omega, (\mathcal{F}_t), P, \xi) \text{ and } \tilde{\mathbb{P}} = \left(\tilde{\Omega}, \left(\tilde{\mathcal{F}}_t\right), P, \tilde{\xi}\right)$$

are given, such that for $u \in \Omega$ and $v \in \tilde{\Omega}$ a distance $\mathsf{d}(u, v)$ is defined by (14). The nested distance of order $r \geq 1$ is the optimal value of the optimization problem

$$\begin{aligned} &\underset{(\text{in } \pi)}{\text{minimize}} \quad \left(\iint \mathsf{d}\,(u, v)^r\, \pi\,(du, dv)\right)^{1/r} \\ &\text{subject to } \pi\left(A \times \tilde{\Omega} \mid \mathcal{F}_t \otimes \tilde{\mathcal{F}}_t\right) = P\,(A \mid \mathcal{F}_t) \qquad (A \in \mathcal{F}_t,\ t = 0, \ldots T) \text{ and} \\ &\qquad \pi\left(\Omega \times B \mid \mathcal{F}_t \otimes \tilde{\mathcal{F}}_t\right) = \tilde{P}\left(B \mid \tilde{\mathcal{F}}_t\right) \qquad \left(B \in \tilde{\mathcal{F}}_t,\ t = 0, \ldots T\right), \end{aligned} \tag{22}$$

where the infimum in (22) is among all bivariate probability measures $\pi \in \mathcal{P}\left(\Omega \times \tilde{\Omega}\right)$ which are measures for $\mathcal{F}_T \otimes \tilde{\mathcal{F}}_T$. Its optimal value – the nested, or multistage distance – is denoted by

$$\mathsf{dl}_r\left(\mathbb{P}, \tilde{\mathbb{P}}\right).$$

By (14) the distance depends on the image measures induced by $\xi_t : \Omega \to \mathbb{R}^m$ and $\tilde{\xi} : \tilde{\Omega} \to \mathbb{R}^m$ on $\mathbb{R}^m$.

It can be shown by counterexamples that the optimal measure π in (22) cannot be described by a transport map in general.

The following theorem is the counterpart of Theorem 16 for general measures and proved in [26]. However, the nested distance can be applied to reveal the same type of result.

Theorem 21 *Let $\mathbb{P} = (\Omega, (\mathcal{F}_t), P, \xi)$ and $\tilde{\mathbb{P}} = \left(\tilde{\Omega}, \left(\tilde{\mathcal{F}}_t\right), P, \tilde{\xi}\right)$ be two probability models. Assume that $\mathbb{X}$ is convex and the cost function Q is convex in x for any fixed ξ. Moreover let Q be uniformly Hölder continuous in ξ with constant L_β and exponent β, that is*

$$\left|Q(x,\xi) - Q(x,\tilde{\xi})\right| \le L_\beta \left(\sum_{t=1}^{T} d_t(\xi_t, \tilde{\xi}_t)\right)^\beta$$

for all $x \in \mathbb{X}$. Then the optimal value function inherits the Hölder constants with respect to the nested distance,

$$\left|\begin{array}{l}\min\{\mathbb{E}_P[Q(x,\xi)] : x \in \mathbb{X},\ x \lhd \mathfrak{F},\ \mathbb{P} = (\Omega, \mathfrak{F}, P, \xi)\} \\ \quad -\min\left\{\mathbb{E}_{\tilde{p}}\left[Q\left(x,\tilde{\xi}\right)\right] : x \in \mathbb{X},\ x \lhd \tilde{\mathfrak{F}},\ \tilde{\mathbb{P}} = \left(\tilde{\Omega}, \tilde{\mathfrak{F}}, \tilde{P}, \tilde{\xi}\right)\right\}\end{array}\right| \le L_\beta\, \mathsf{dl}_r(\mathbb{P}, \tilde{\mathbb{P}})^\beta \tag{23}$$

for any $r \ge 1$. This bound cannot be improved.

The relation between the nested distance and the single period Wasserstein distance. The nested distance $\mathsf{dl}_r(\mathbb{P}, \tilde{\mathbb{P}})$ can be bounded by the Wasserstein distances of the conditional probabilities as is described in Theorem 23 below. It uses the notion of the K-Lipschitz property.

Definition 22 *(K-Lipschitz property)* Let P be a probability on $\mathbb{R}^{mT}$, dissected into transition probabilities $P_1, \ldots, P_T$ on $\mathbb{R}^m$. We say that P has the *K-Lipschitz property* for $K = (K_1, \ldots, K_{T-1})$, if the transitional probability measures satisfy

$$\mathsf{d}_r\left(P_{t+1}(\cdot|u^t),\ P_{t+1}(\cdot|v^t)\right) \le K_t\, \mathsf{d}\left(u^t, v^t\right) \tag{24}$$

for all $u^t,\ v^t \in \mathbb{R}^{m(t-1)}$ and $t = 1, \ldots, T-1$.

Theorem 23 (Stagewise transportation distance) *Suppose that the probability measure P on $\mathbb{R}^{mT}$ fulfills a $(K_1, \ldots, K_{T-1})$-Lipschitz property and that the conditional distributions of P and $\tilde{P}$ satisfy*

$$\mathsf{d}_r(P_1, \tilde{P}_1) \le \epsilon_1$$

and

$$\mathsf{d}_r\left(P_{t+1}(\cdot|v^t),\ \tilde{P}_{t+1}(\cdot|v^t)\right) \le \epsilon_{t+1} \quad \text{for all } v^t \text{ and } t = 0, \ldots, T-1. \tag{25}$$

Then the nested distance is bounded by

$$\mathsf{dl}_r(\mathbb{P}, \tilde{\mathbb{P}}) \le \sum_{t=1}^{T-1} \epsilon_t \cdot \prod_{s=t}^{T} (1 + K_s). \tag{26}$$

Proof The proof is contained in [19]. □

Remark 24 As variant of the results one may consider distorted distances such as the Fortet–Mourier distance (cf. Rachev [28] or Römisch [29]). Küchler [15] combines a nonanticipativity condition for the approximating process (a pushforward approximation) and a Lipschitz property for the conditional processes (such as (24) in Theorem 23 below) to establish a stability result similar to Theorem 16.

The previous sections address approximations of a probability measure, where the quality of the approximation is measured by the Wasserstein distance. The algorithms described in Sect. 3 can be employed at every node separately to build the tree. To apply the general result (26) it is necessary to condition on the values of the previous nodes (cf. (25)), which have already been fixed in earlier steps of the algorithm. To apply the algorithms described in Sect. 3 in connection with Theorem 23 it is thus necessary that the conditional probability measure is available to compute (25), that is, samples from

$$F_{t+1}\left(\cdot \mid \xi_t, \xi_{t-1}, \ldots, \xi_0\right)$$

can be drawn.

We outline this approach in the next section for a fixed branching structure of the tree, and in the following section with an adaptive branching structure to meet a prescribed approximation quality.

5.1 Fixed branching structure

Algorithm 6 elaborates on generating scenario trees in further detail. The trees are constructed to have b_t successor nodes at each node at level t, the vector $(b_1, \ldots b_T)$ is the *bushiness* of the tree.

Algorithm 6 Tree generation with fixed bushiness

Parameters. Let T be the desired height of the tree and let $(b_1, \ldots, b_T)$ be the given bushiness parameters per stage.

- **Determining the Root.** The value of the process at the root is ξ_0. Its stage is 0. Set the root as the current open node.
- **Successor generation.** Enumerate the tree stagewise from the root to the leaves.
  (i) Let k be the node to be considered next and let $t < T$ be its stage. Let $\xi_0, \xi_1, \ldots \xi_t$ be the already fixed values at node k and all its predecessors. Call the stochastic approximation algorithm (Algorithm 3) to generate b_t points $z^{(1)}, \ldots, z^{(b_t)}$ out of the probability distribution

$$F_{t+1}(\cdot|\xi_t, \xi_{t-1}, \ldots, \xi_0) \tag{27}$$

  and find the corresponding conditional probabilities $p\left(\tilde{z}^{(i)}\right)$.
  (ii) Store the b_t successor nodes, say with node numbers $(n_1, \ldots, n_{b_t})$ of node k and assign to them the values $\xi(n_1) = z^{(1)}, \ldots, \xi(n_{b_t}) = z^{(n_{b_t})}$ as well as their conditional probabilities $q(n_i) = p\left(\tilde{z}^{(i)}\right)$.
- **Stopping Criterion.** If all nodes at stage $T - 1$ have been considered as parent nodes, the generation of the tree is finished. One may then calculate the unconditional probabilities out of the conditional probabilities.

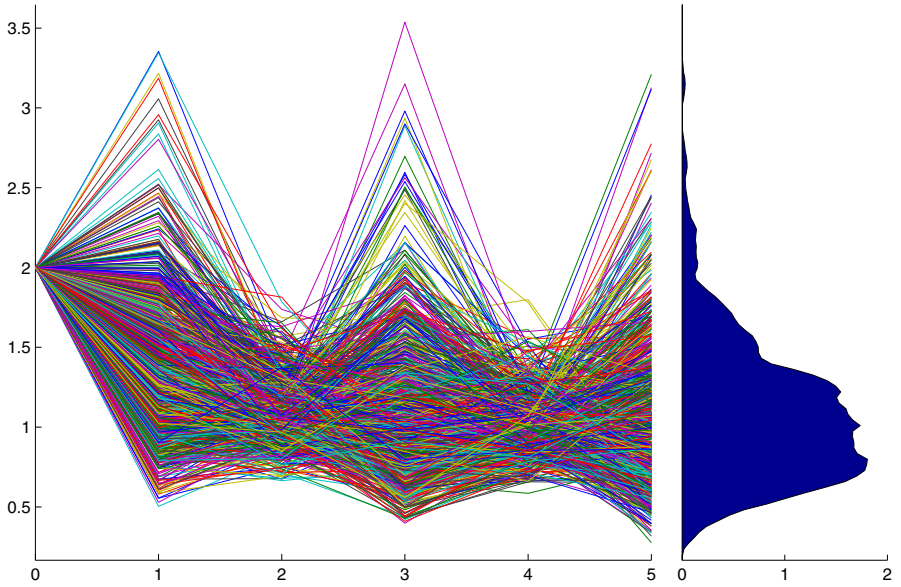

Fig. 3 1000 trajectories of the process (28) in Example 25

The algorithm based on Theorem 23 described in this and the following section have been implemented in order to demonstrate their behavior by using the following example.

Example 25 Consider the Markovian process

$$\xi_0 = 2, \quad \xi_{t+1} = \xi_t^{1-\beta_t} \cdot e^{\eta_t}, \tag{28}$$

where $\beta_0 = 0.62$, $\beta_1 = 0.69$, $\beta_2 = 0.73$, $\beta_3 = 0.75$ and $\beta_4 = 0.77$, and

$$\eta_0 \sim \eta_2 \sim \eta_4 \sim \mathcal{N}\left(0,\ 0.5\beta_t^2\right) \text{ and } \eta_0 \sim \eta_2 \sim \mathcal{N}\left(0,\ 0.2\beta_t^2\right).$$

Notice that the distribution, given the past, is explicitly available by (28), although the conditional variance depends heavily on t.

Figure 3 displays some trajectories of the process (28). A binary tree, approximating the process (28), is constructed by use of Algorithm 6. Figure 4 displays the tree structure, as well as the approximating binary tree.

5.2 Tree, meeting a prescribed approximation precision

Assume that the law of the process to be approximated satisfies the K-Lipschitz property introduced in Definition 22. Theorem 23 then can be used to provide an approximation of the initial process in terms of the nested distance up to a prescribed precision. Algorithm 7 outlines this approach. Again, as in the preceding algorithm,

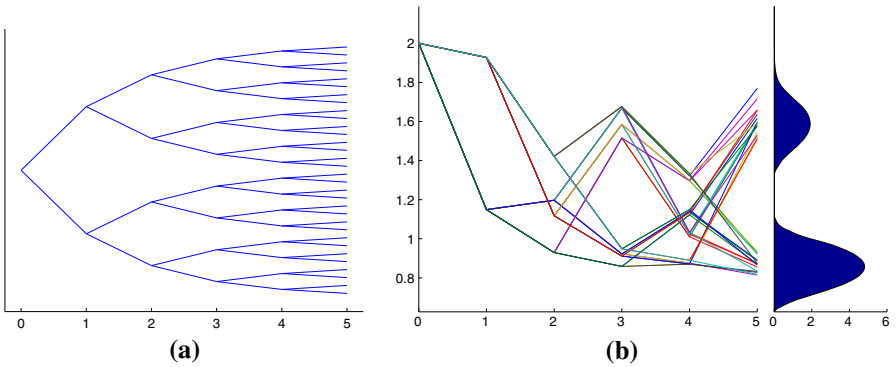

Fig. 4 A binary tree generated by Algorithm 6 approximating process (28), Example 25. **a** The structure of the binary tree. **b** The approximating process

it is necessary to have samples of the distribution $F_{t+1}\left(\cdot|\xi_t, \xi_{t-1}, \ldots, \xi_0\right)$ available, given that the past is revealed up to the present stage t.

The algorithm is demonstrated again for the process (28) in Example 25. Results are displayed in Fig. 5.

Algorithm 7 Dynamic tree generation with flexible bushiness

- **Parameters.** Let T be the desired height of the tree, let $(b_1, \ldots, b_T)$ be the minimal bushiness values and $(d_1, \ldots, d_T)$ the maximal stagewise transportation distances. These two vectors are fixed in advance.
- **Determining the Root.** The value of the process at the root is ξ_0, its stage is 0. Set the root as the current open node.
- **WHILE** there are open nodes **DO**
  - (i) Let k be the next open node and let $t < T$ be its stage. Let $\xi_0, \ldots \xi_{t-1}, \xi_t$ be the already fixed values at node k and at its predecessors. Set the initial number of successors of k to $s = b_{t+1}$.
  - (ii) Call the stochastic approximation algorithm (Algorithm 3) to generate s points $z_1^*, \ldots, z_s^*$ out of the distribution

$$F_{t+1}\left(\cdot|\xi_t, \xi_{t-1}, \ldots, \xi_0\right) \tag{29}$$

    and compute the distance $d = \mathsf{d}\left(F_{t+1}(\cdot|\xi_t, \xi_{t-1}, \ldots, \xi_0),\ \sum_{i=1}^{s} p_i \cdot \delta_{z^{(i)}}\right)$.
  - (iii) If the distance d is larger than d_{t+1}, then increase b by one and return to (ii).
  - (iv) Store the b successor nodes of node k using the values z_k^* as well as their conditional probabilities p_k^* and mark them as open.
- **Stopping Criterion.** If all nodes at stage $T-1$ have been considered as parent nodes, the generation of the tree is finished.

6 Summary

This paper addresses scenario tree generation, which is of interest in many economic and managerial situations, in particular for stochastic multistage optimization.

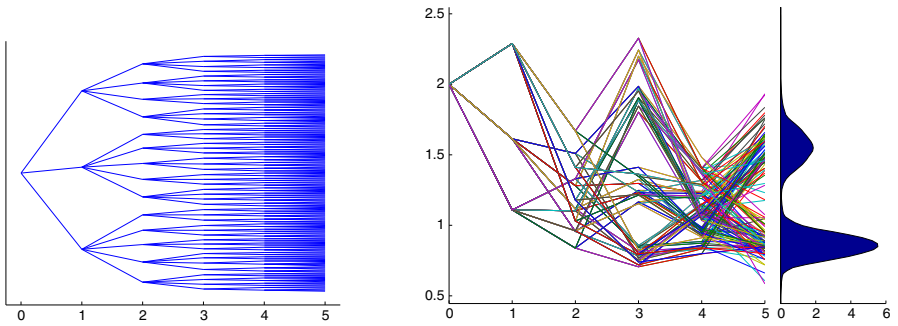

Fig. 5 A tree, dynamically generated by Algorithm 7, approximating the process (28). The maximal stagewise distances were 0.30, 0.15, 0.30, 0.30 and 0.40. The approximating tree process has 390 nodes and 224 leaves. **a** The branching structure of the tree with bushiness (3, 4, 4, 2). **b** The approximating process

It is demonstrated that techniques, which are used to approximate probability measures, can be extended to generate approximating trees, which model stochastic process in finite stages and states.

Various algorithms are shown first to approximate probability measures by discrete measures. These algorithms are combined then at a higher level to provide tree approximations of stochastic processes. The generated trees meet a quality criterion, which is formulated in terms of the nested distance. The nested distance is the essential distance to compare stochastic scenario processes for stochastic optimization.

The algorithms presented require sampling according to probability distributions based on the past evolutions of the process. Several examples and charts demonstrated the quality of the solutions.

Acknowledgments We would like to thank two independent referees. Based on their careful reading and precise feedback we were able to improve the presentation and content significantly. Parts of this paper are addressed in our Springer book *Multistage Stochastic Optimization* [27], which summarizes many more topics in multistage stochastic optimization and which had to be completed before final acceptance of this paper. A. Pichler: The author gratefully acknowledges support of the Research Council of Norway (Grant 207690/E20)

Appendix: Conditional method for multivariate distributions

To generate instances from a multivariate distribution (random vectors) with cdf $F(z_1, \dots z_m)$ one may employ *rejection sampling*, or the *ratio of uniforms* method, or generate the vector component by component by proceeding as follows:

(i) Generate Z_1 from $F_1(z)$, where $F_1(z_1) = F(z_1, \infty, \dots \infty)$ is the first marginal. This can be accomplished by solving

$$U_1 = F_1(Z_1)$$

(the probability integral transform, where U_1 is a uniformly distributed random variable) or by rejection sampling;

(ii) Given the random vector up to dimension $i-1$ one may generate Z_i conditionally on $(Z_1, \ldots Z_{i-1})$ by solving

$$U_i = F_i\left(z_i \mid Z_1, \ldots Z_{i-1}\right) = \frac{F\left(Z_1 \ldots, Z_{i-1}, x_i, \infty, \ldots \infty\right)}{F\left(Z_1, \ldots Z_{i-1}, \infty, \ldots \infty\right)},$$

where U_i is uniformly distributed, and independent from $U_1, \ldots U_{i-1}$.

References

1. Bally, V., Pagès, G., Printems, J.: A quantization tree method for pricing and hedging multidimensional american options. Math. Financ. **15**(1), 119–168 (2005)
2. Beiglböck, M., Goldstern, M., Maresch, G., Schachermayer, W.: Optimal and better transport plans. J. Funct. Anal. **256**(6), 1907–1927 (2009)
3. Brenier, Y.: Décomposition polaire et réarrangement monotone des champs de vecteurs. Comptes Rendus de l'Académie des Sci. Paris Sér. I Math **305**(19), 805–808 (1987)
4. Brenier, Y.: Polar factorization and monotone rearrangement of vector-valued functions. Commun. Pure Appl. Math. **44**(4), 375–417 (1991)
5. Dupačová, J., Gröwe-Kuska, N., Römisch, W.: Scenario reduction in stochastic programming. Math. Program. Ser. A **95**(3), 493–511 (2003)
6. Graf, S., Luschgy, H.: Foundations of Quantization for Probability Distributions. Lecture Notes in Mathematics, vol. 1730. Springer-Verlag, Berlin Heidelberg (2000)
7. Heitsch, H., Römisch, W.: Scenario reduction algorithms in stochastic programming. Comput. Optim. Appl. Stoch. Program. **24**(2–3), 187–206 (2003)
8. Heitsch, H., Römisch, W.: A note on scenario reduction for two-stage stochastic programs. Oper. Res. Lett. **6**, 731–738 (2007)
9. Heitsch, H., Römisch, W.: Scenario tree modeling for multistage stochastic programs. Math. Program. Ser. A **118**, 371–406 (2009)
10. Heitsch, H., Römisch, W.: Scenario tree reduction for multistage stochastic programs. Comput. Manag. Sci. **2**, 117–133 (2009)
11. Heitsch, H., Römisch, W.: Stochastic Optimization Methods in Financ eand Energy. Scenario Tree Generation for Multistage Stochastic Programs, chapter 14, pp. 313–341. Springer Verlag, New York (2011)
12. Hilli, P., Pennanen, T.: Numerical study of discretizations of multistage stochastic programs. Kybernetika **44**(2), 185–204 (2008)
13. Høyland, K., Wallace, S.: Generating scenario trees for multistage decision problems. Manag. Sci. **47**(2), 295–307 (2001)
14. Kaut, M., Wallace, S.W.: Shape-based scenario generation using copulas. Comput. Manag. Sci. **8**(1–2), 181–199 (2011)
15. Küchler, Ch.: On stability of multistage stochastic programs. SIAM J. Optim. **19**(2), 952–968 (2008)
16. Kuhn, D.: Generalized Bounds for Convex Multistage Stochastic Programs. Lecture Notes in Economics and Mathematical Systems. Berlin, DE: Springer (2005)
17. Kuhn, D.: Aggregation and discretization in multistage stochastic programming. Math. Program. **113**(1), 61–94 (2008)
18. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability. Statistics, vol. 1, pp. 281–29. University of California Press, Berkeley (1967)
19. Mirkov, R., Pflug, G.Ch.: Tree approximations of dynamic stochastic programs. SIAM J. Optim. **18**(3), 1082–1105 (2007)
20. Pagès, G.: A space quantization method for numerical integration. J. Comput. Appl. Math. **89**(1), 1–38 (1998)
21. Pennanen, T.: Epi-convergent discretizations of multistage stochastic programs via integration quadratures. Math. Program. **116**(1–2), 461–479 (2009)
22. Pennanen, T., Koivu, M.: Epi-convergent discretizations of stochastic programs via integration quadratures. Numerische Mathematik **100**, 141–163 (2005)

23. Pflug, G.Ch.: Optimization of Stochastic Models. The Kluwer International Series in Engineering and Computer Science, vol. 373. Kluwer Academic Publishers, New York (1996)
24. Pflug, G.Ch.: Scenario tree generation for multiperiod financial optimization by optimal discretization. Math. Program. **89**, 251–271 (2001)
25. Pflug, G.Ch.: Version-independence and nested distribution in multistage stochastic optimization. SIAM J. Optim. **20**, 1406–1420 (2009)
26. Pflug, G.Ch., Pichler, A.: A distance for multistage stochastic optimization models. SIAM J. Optim. **22**(1), 1–23 (2012)
27. Pflug, G.Ch., Pichler, A.: Multistage Stochastic Optimization. Springer, Springer Series in Operations Research and Financial Engineering, New York (2014)
28. Rachev, S.T.: Probability metrics and the stability of stochastic models. Wiley, West Sussex (1991)
29. Römisch, W.: Stability of stochastic programming problems. In: Ruszczyński, A., Shapiro, A. (eds.) Stochastic Programming, Handbooks in Operations Research and Management Science, volume 10, chapter 8. Elsevier, Amsterdam (2003)
30. Shapiro, A.: Inference of statistical bounds for multistage stochastic programming problems. Math. Methods Oper. Res. **58**(1), 57–68 (2003)
31. Shiryaev, A.N.: Probability. Springer, New York (1996)
32. Villani, C.: Topics in Optimal Transportation. Graduate Studies in Mathematics, vol. 58. American Mathematical Society, Providence, RI (2003)
33. Williams, D.: Probability with Martingales. Cambridge University Press, Cambridge (1991)