

Envelope Theorems for Multi-Stage Linear Stochastic Optimization

Gonalo Tera

Technische Universitat Munchen, Munich, Germany, g.terca@tum.de

David Wozabal

Technische Universitat Munchen, Munich, Germany, david.wozabal@tum.de

We propose a method to compute derivatives of multi-stage linear stochastic optimization problems with respect to parameters that influence the problem’s data. Our results are based on classical envelope theorems, and can be used in problems directly solved via their deterministic equivalents as well as in stochastic dual dynamic programming for which the derivatives of the optimal value are sampled. We derive smoothness properties for optimal values of linear optimization problems, which we use to show that the computed derivatives are valid *almost everywhere* under mild assumptions. We discuss two numerical case studies, demonstrating that our approach is superior, both in terms of accuracy as well as computationally, to naive methods of computing derivatives that are based on difference quotients.

Key words: Markov decision processes, sensitivity analysis, stochastic dual dynamic programming, semi-algebraic sets

1. Introduction

In most planning and decision problems some of the relevant parameters are not known with certainty at the time when decisions are taken. Stochastic optimization deals with such uncertainties in problems that lend themselves to treatment in an optimization framework. More specifically, stochastic optimization is concerned with the solution of optimization problems that involve random quantities as data (Birge and Louveaux 2011, Shapiro et al. 2014).

Most of stochastic programming theory is concerned with obtaining optimal values and optimal decisions in the face of uncertainty. In this paper, we investigate the less researched question of how to calculate derivatives of optimal values with respect to the parameters of the problem. More specifically, we devise a method to determine the sensitivity of the optimal value of a multi-stage stochastic linear optimization problem (MSLP) with respect to a change in a model parameter,

i.e., a part of the model that can not be influenced by the decision maker.

Sensitivities of the optimal value with respect to parameters like costs, prices, characteristics of technical systems, parameters of the underlying stochastic process, or the amount of available resources are of interest for at least two reasons.

Firstly, as in deterministic optimization, derivatives can be interpreted as *shadow prices* of the resource or parameter in question. An example is classical inventory planning, where a planner operates a storage facility with certain properties such as storage cost, storage size, and storage loss. The derivative of the value of the optimal inventory policy with respect to these parameters gives valuable information on the system and cues on how to improve revenues.

Secondly, derivatives provide information on the inherent risks of the optimal policy. Examples include pricing and hedging problems in finance. In this context, the value of an asset or contract can often be viewed as the solution of a stochastic optimization problem, and decision makers are interested in the sensitivities of these values with respect to parameters such as prices, volatilities, and contract provisions. In the literature, these sensitivities are called *Greeks* and are extensively used in financial risk management (Hull 2014). For standard cases in option pricing and a handful of parameters, Greeks can be computed explicitly. However, for the large set of analytically intractable problems, derivatives have to be numerically approximated.

Taking derivatives of optimal values has a long history in the literature. In particular, computing sensitivities with respect to parameters of a deterministic optimization problem is one of the main questions of the field of parametric optimization. In this paper, we build on the extensive literature on linear parametric optimization (see Gal and Greenberg 1997, for an overview), in particular, when studying the question of smoothness of the optimal value of linear optimization problems.

In stochastic optimization, determining sensitivities of optimal values is at the core of many decomposition methods like the L-shaped method (Van Slyke and Wets 1969, Birge and Louveaux 2011). These approaches determine how the recourse function of the next period varies with a change in here-and-now decisions, which usually appear only in the right-hand side of the next stage's problem. Hence, the sensitivities are evaluated with respect to previous stage decision

variables and not the data of the problem, which is different to the focus of this paper. Furthermore, decomposition approaches are usually much more restrictive when it comes to which parameters of the problem are allowed to vary.

We also mention a stream of literature in stochastic optimization that is concerned with the sensitivity of the optimal value with respect to perturbations in the distribution of the involved random variables Dupačová (1987, 1990), Branda and Dupačová (2012). However, the aim of these efforts is rather specific and mainly directed at robustifying solutions with respect to errors made in the estimation of the distributional model for the randomness in the problem.

Our approach differs from the extant literature in the sense that we use classical envelope theorems for linear programs to prove envelope theorems for MSLPs and calculate derivatives with respect to arbitrary parameters of the problem. The proposed approach computes derivatives based on samples from the optimal policy. We prove the validity of our approach by showing that, under weak regularity conditions, the optimal value of a MSLP is almost everywhere differentiable.

Although our results are applicable to general MSLPs, we wrote this paper with Markov decision processes (MDP) in mind. In particular, we intend to calculate derivatives for problems solved using *stochastic dual dynamic programming* (SDDP), which approximately solves stochastic optimization problems with fully continuous state and action space. We demonstrate how sampling from the optimal policy can be used to compute derivatives, without re-solving the problem for several values of the parameter in question, as is done in numerical differentiation.

In a numerical study, we show that our method works well for practical problems. To this end, we set up two case studies: a simple two-stage newsvendor problem which can be solved analytically, and a multi-stage gas storage optimization problem. We use the newsvendor problem to test the convergence of our sampled derivatives to the real ones, and the gas storage optimization to demonstrate that the proposed method scales well for larger problem instances with many decision stages. We use a naïve approach based on difference quotients as a benchmark, and show that our method outperforms this benchmark both in accuracy as well as computationally.

Furthermore, we demonstrate that, for MSLPs with continuous randomness, estimates for the derivative based on samples from the original continuous distribution outperform estimates based on samples from discrete approximations of randomness like scenario trees or scenario lattices, even if the optimal policies were found using the approximate discrete processes.

The paper is structured as follows: In Section 2, we give a short review of classical envelope theorems and discuss the specific situation for linear optimization problems. In particular, we study the question of differentiability and prove that the optimal value of a linear optimization problem is differentiable almost everywhere, jointly in all its data. In Section 3, we apply these results to derive an envelope theorem for MSLP problems and show how derivatives can be sampled in SDDP-type algorithms. Section 4 is devoted to a numerical study demonstrating the practical viability and computational tractability of the proposed approach. Section 5 concludes the paper and discusses avenues for further research.

2. Envelope Theorems

In this section, we introduce envelope theorems for linear optimization problems, which will be the basis for computing sensitivities for MSLPs in the next section. In particular, we investigate when the optimal value of a linear optimization problem is differentiable and show that non-differentiability is in some sense *rare*.

Envelope theorems quantify how much the optimal value of an optimization problem changes for an incremental change in some of the parameters of the problem. Results of this type are important tools in microeconomic theory and have been extensively used to prove central results in comparative statics analysis such as Shepard's lemma, Roy's identity, or Slutsky's equation relating to questions in the theory of the firm as well as consumer choice (see Varian 1992). The first statement of the envelope theorem in its modern form arguably appears in Samuelson (1947). Dynamic envelope theorems in optimal control can, for example, be found in LaFrance and Barney (1991) and the most general results known to the authors appeared in Milgrom and Segal (2002).

The most basic form of the envelope theorem concerns maximizing a sufficiently smooth function $f(x, \theta)$ that depends on a parameter θ as well as on a decision x . Denote by $x^*(\theta)$ the optimal

decision for a given parameter θ and by $V(\theta)$ the corresponding optimal value dependent on θ .

Then clearly,

$$\begin{aligned} V(\theta)' &= \frac{d}{d\theta} \max_x f(x, \theta) = \frac{d}{d\theta} f(x^*(\theta), \theta) \\ &= \frac{\partial}{\partial x} f(x^*(\theta), \theta) \frac{d}{d\theta} x^*(\theta) + \frac{\partial}{\partial \theta} f(x^*(\theta), \theta) = \frac{\partial}{\partial \theta} f(x^*(\theta), \theta) \end{aligned}$$

where the last equality follows from the first-order condition

$$\frac{\partial}{\partial x} f(x^*(\theta), \theta) = 0$$

at the optimal point $x^*(\theta)$.

The above shows that the derivative of the optimal value can be computed knowing only $\frac{\partial}{\partial \theta} f$ and the optimal solution $x^*(\theta)$. In particular, it is not required to know $\frac{d}{d\theta} x^*(\theta)$, i.e., how the optimal solution changes in θ , which is usually much harder to calculate. This property makes the envelope theorem extremely useful in the analysis of optimization problems.

The outlined approach can be extended to constrained optimization problems but generally requires smoothness assumptions on the optimal primal and dual solutions. We use the following classical envelope theorem in its formulation for linear optimization problems for our purposes, which extends the simple derivation above to the case of constrained optimization by considering the Lagrangian of the problem.

THEOREM 1 (Takayama (1985), Theorem 1.F.1.). *Let $\Theta \subseteq \mathbb{R}$ be open and consider differentiable functions $A : \Theta \rightarrow \mathbb{R}^{m \times n}$, $b : \Theta \rightarrow \mathbb{R}^m$, $c : \Theta \rightarrow \mathbb{R}^n$ and a linear programming problem of the form*

$$V(\theta) = \begin{cases} \max_x \langle c(\theta), x \rangle \\ \text{s. t.} & A(\theta)x \leq b(\theta), x \geq 0. \end{cases} \quad (1)$$

If, for a given $\theta \in \Theta$, the optimal primal and dual solutions $x^(\theta)$ and $\lambda^*(\theta)$ are continuously differentiable in θ , then*

$$\frac{d}{d\theta} V(\theta) = \langle \nabla c(\theta), x^*(\theta) \rangle + \left\langle \lambda^*(\theta), \nabla b(\theta) - \frac{d}{d\theta} A(\theta)x^*(\theta) \right\rangle. \quad (2)$$

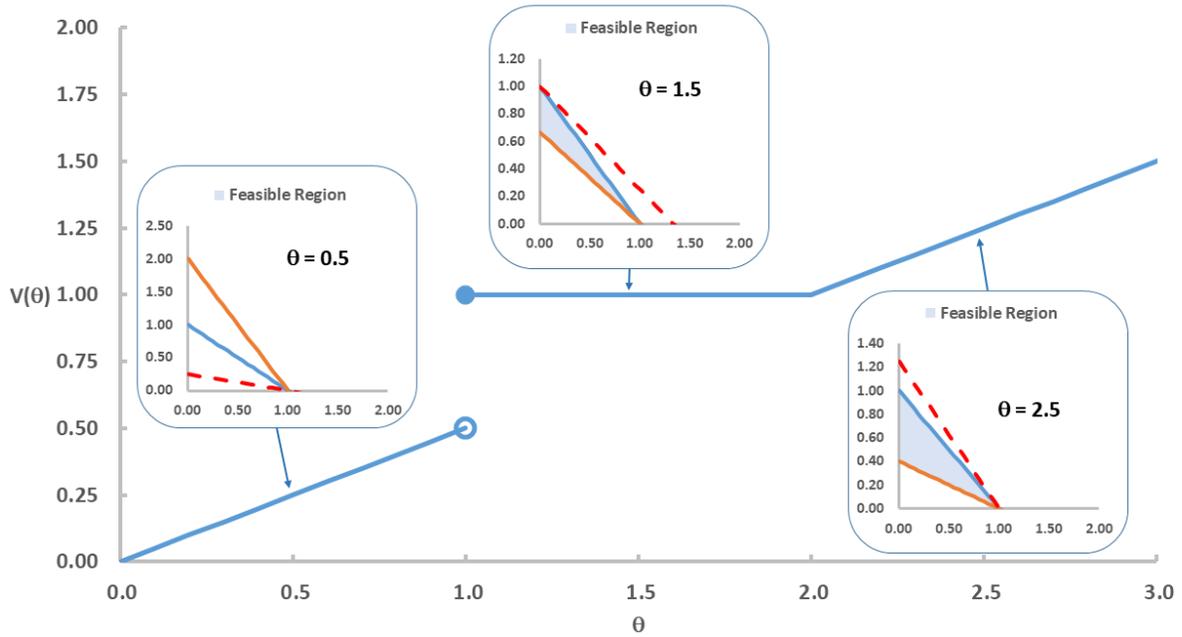


Figure 1 Value function and typical feasible regions for problem (3).

Note that in the above, we use $\frac{d}{d\theta}A(\theta)$ as short-hand for the component-wise derivative with respect to θ .

Looking at the formulation of Theorem 1, it is obvious that, in the case of linear optimization problems, the condition on the smoothness of x^* and λ^* is restrictive and therefore the result is, in general, not applicable for all values of θ . More specifically, it is well known that the solution of linear optimization problems need not be differentiable in the parameters of the problem. The following problem is a modification of an example given in Martin (1975), which illustrates this point.

$$V(\theta) = \begin{cases} \max_{x,y} \frac{1}{2}\theta x + y \\ \text{s.t.} & x + y \leq 1 \\ & x + \theta y \geq 1 \\ & x, y \geq 0. \end{cases} \quad (3)$$

For $\theta \geq 0$, the optimal solution and optimal value are given by

$$\arg \max_{(x,y)} V(\theta) = \begin{cases} \{(0,1)\}, & 1 \leq \theta < 2 \\ \{(1,0)\}, & \theta \notin [1,2] \\ \{(1-\alpha, \alpha) : \alpha \in [0,1]\}, & \theta = 2 \end{cases}, \quad V(\theta) = \begin{cases} 1, & 1 \leq \theta < 2 \\ \frac{\theta}{2}, & \theta \notin [1,2]. \end{cases}$$

Clearly, the optimal value jumps at $\theta = 1$ and kinks at $\theta = 2$. Figure 1 illustrates the optimal value function as well as typical examples of feasible regions in the three relevant sections for θ . An example featuring disjoint feasible regions can be found in Willner (1967).

Hence, for a general linear optimization problem, the optimal value does not even have to be continuous in the parameters of the problem and therefore the envelope theorem cannot be used to calculate $V'(\theta)$ for every θ . However, we observe that there are only finitely many points where $\theta \mapsto V(\theta)$ is not differentiable in the above example. As we will see below, this observation extends to the general case.

In a linear optimization problem, the issue of differentiability of the optimal value is closely related to the notion of uniqueness and non-degeneracy of the solution. Given an optimal simplex tableau, a (feasible) solution is *unique*, if none of the non-basic reduced cost coefficients vanish and it is *non-degenerate*, if and only if none of the basic variables vanish. If both conditions are met, differentiability of the objective follows (see Gal and Greenberg 1997). However, the two conditions are not necessary for the objective to be differentiable. In particular, in problem (3), we observe that for $\theta \notin [1, 2]$ the solution is degenerate, but the value function is differentiable with $V'(\theta) = \frac{1}{2}$.

Another more general sufficient condition for smoothness of V as a function of θ is that the optimal basis remains unchanged in a neighborhood \mathcal{U} of θ , i.e., that the set of binding constraints remains the same for all $\theta \in \mathcal{U}$. If this is the case, then, in \mathcal{U} , all entries of the optimal simplex tableau vary smoothly in θ , and, since the optimal value is a linear function of the entries, so does the optimal value.

Therefore, as a first step, we characterize the points where there are discontinuities or kinks in the value function by characterizing the sets of parameters θ where the optimal basis of a linear optimization problem of a given dimensionality does not change. Similar questions have been studied extensively in the literature on parametric linear programming (see Courtillot 1962, Willner 1967, Barnett 1968, Gal and Greenberg 1997, Dent et al. 1973, Adler and Monteiro 1992, Ward and Wendell 1990). However, results in these papers are usually restricted to either variations in the right-hand side of constraints, parameters in the objective function, or special variations in

the matrix. For our results, a simultaneous variation in all parameters of the problem is required, which, to the best of our knowledge, has not yet been studied.

For further analysis, we find it convenient to re-write problem (1) in standard form by introducing slack variables s_1, \dots, s_k for inequality constraints, rewriting the constraints as

$$Ax + s = b.$$

Defining $y = (x, s)$, we can thus define a new linear optimization problem equivalent to (1) in standard form as

$$V(\theta) = \begin{cases} \max_y \langle d(\theta), y \rangle \\ \text{s.t.} \quad D(\theta)y = b(\theta), y \geq 0. \end{cases} \quad (4)$$

where d and D are correspondingly updated versions of c and A and $\mathcal{D} = (d, D, b)$ represents all the data of the problem. To simplify our arguments, we will frequently view \mathcal{D} as an element of \mathbb{R}^N for some $N \in \mathbb{N}$ by stacking all the vectors and columns of the matrix D on top of each other. Furthermore, we define the short-hand $[n] = \{1, \dots, n\}$.

As mentioned above, we are interested in the case where all the parameters of the linear optimization problem can vary simultaneously, i.e., we study how a variation in $\mathcal{D} \in \mathbb{R}^N$ affects the solution. We first prove a result characterizing the points where the value function of a linear optimization problem is smooth jointly in the parameters of the problem. The proof follows the idea outlined in Freund (1985) for the special case of changing only the matrix D along one line. We deal with the additional complication of arbitrarily changing all elements of \mathcal{D} using results from real algebraic-geometry.

LEMMA 1. *Let $\mathcal{P}(\mathcal{D})$ be a problem of form (4) dependent on data $\mathcal{D} \in \mathbb{R}^N$ with a matrix D of dimensions $m \times n$. Define $\beta \subseteq [n]$ to be a basis with $|\beta| = m$ and the set*

$$R_\beta = \{\mathcal{D} : \beta \text{ is an optimal basis for } \mathcal{P}(\mathcal{D})\}.$$

Then R_β is the union of finitely many connected smooth manifolds of varying dimensions in \mathbb{R}^N .

Proof. Due to the introduction of slack variables for inequality constraints, we know that $m < n$. For any D , define D_β as the square matrix resulting from D with the columns $[n] \setminus \beta$ removed. It holds that

$$R_\beta = \{\mathcal{D} : \det(D_\beta) \neq 0\} \cap \{\mathcal{D} : D_\beta^{-1}b \geq 0\} \cap \{\mathcal{D} : c_\beta D_\beta^{-1}D \geq c\},$$

where the first condition ensures that x with $x_\beta = D_\beta^{-1}b$ and $x_{[n] \setminus \beta} = 0$ is a primal basic solution, the second one restricts to \mathcal{D} such that β is a primal feasible basis, while the third set contains the \mathcal{D} such that β is dual feasible. The last two conditions in combination yield that x is primal optimal and, therefore, that β is an optimal basis for $\mathcal{P}(\mathcal{D})$.

Note that $D_\beta^{-1} = \det(D_\beta)^{-1} \text{adj}(D_\beta)$ with $\text{adj}(A)$ the adjugate of a matrix A . Note further that if $\det(D_\beta) \neq 0$, we can always re-arrange columns of D_β in such a way that $\det(D_\beta) > 0$. In this case, $D_\beta^{-1}b$ yields a basic solution with correspondingly permuted elements. We can therefore write

$$R_\beta = \{\mathcal{D} : \det(D_\beta) > 0\} \cap \{\mathcal{D} : \text{adj}(D_\beta)b \geq 0\} \cap \{\mathcal{D} : c_\beta \text{adj}(D_\beta)D \geq c \det(D_\beta)\}. \quad (5)$$

Let us first investigate the set $\{\mathcal{D} : \text{adj}(D_\beta)b \geq 0\}$. The elements of the adjugate are determinants that can be written as polynomials in the variables of D . Defining $A_{(i,:)}$ as the i -th row of a matrix A , it follows that $P_i(\mathcal{D}) := \text{adj}(D_\beta)_{(i,:)}b$ is a polynomial in the components of b and D , hence, in particular, a polynomial with coefficients from $\mathcal{D} \in \mathbb{R}^N$.

The set $\{\mathcal{D} : P_i(\mathcal{D}) \geq 0, \forall i = 1, \dots, m\} \subseteq \mathbb{R}^N$ is a semi-algebraic set (see Definition 2.1.4 in Bochnak et al. 2013). By the same argument the other two sets in (5) are semi-algebraic and therefore so is R_β . By Proposition 2.9.10 in Bochnak et al. (2013)

$$R_\beta = \bigcup_{i=1}^I \mathcal{S}_i,$$

where $I \in \mathbb{N}$ and \mathcal{S}_i are smooth manifolds in \mathbb{R}^N and there exist diffeomorphisms $h_i : (0, 1)^{n_i} \rightarrow \mathcal{S}_i$, where $0 \leq n_i \leq N$ for all $1 \leq i \leq I$. Since the h_i are continuous and $(0, 1)^{n_i}$ are connected, \mathcal{S}_i are connected sets in \mathbb{R}^N . \square

The following corollary is an easy consequence of Lemma 1 and treats the case that not all of the data varies but only a subset of the entries of A , b , and c .

COROLLARY 1. Consider the set $\{k_1, \dots, k_M\} \subseteq [N]$ with $M \leq N$ and

$$\mathcal{X} = \{\mathcal{D} \in \mathbb{R}^N : \mathcal{D}_k = \mathcal{D}'_k, \forall k \notin \{k_1, \dots, k_M\}\} \simeq \mathbb{R}^M.$$

Then the set of points $\mathcal{D} \in \mathcal{X}$ where there is a basis change in the corresponding linear optimization problem is the union of finitely many connected smooth manifolds of varying dimensions in \mathbb{R}^M .

Finally, we prove a result confirming the intuition that the number of points where the optimal value of a linear optimization problem is not differentiable is *small*.

THEOREM 2. Let $\Theta \subseteq \mathbb{R}$ be an open, connected set and consider invertible bi-continuous functions $A : \Theta \rightarrow \mathbb{R}^{m \times n}$, $b : \Theta \rightarrow \mathbb{R}^m$, $c : \Theta \rightarrow \mathbb{R}^n$ defining the data of the linear problem

$$V(\theta) = \begin{cases} \max_x \langle c(\theta), x \rangle \\ \text{s. t. } A(\theta)x \leq b(\theta). \end{cases} \quad (6)$$

Then the set of points $\theta \in \Theta$ where $V(\theta)$ is not differentiable is finite and therefore has Lebesgue measure 0 in \mathbb{R} . In all other points $\theta \in \Theta$, $V(\theta)$ is smooth in θ .

Proof. Define $\mathcal{D} : \Theta \rightarrow \mathbb{R}^N$ as $\mathcal{D}(\theta) = (A(\theta), b(\theta), c(\theta))$. By the first part of Lemma 1 the sets R_β are the union of finitely many connected sets. Since $\mathcal{D}(\Theta)$ is connected in \mathbb{R}^N , so are the sets $\mathcal{D}(\Theta) \cap \mathcal{S}_i$ and $\mathcal{T}_i = \mathcal{D}^{-1}(\mathcal{S}_i) \subseteq \Theta$. Since there are finitely many $\beta \subseteq [n]$, it follows that Θ can be written as the union of finitely many connected sets. Clearly, the optimal basis for the linear program with data $\mathcal{D}(\theta)$ stays constant for all $\theta \in \mathcal{T}_i$.

Since connected sets in $\Theta \subseteq \mathbb{R}$ are either points or intervals, the set of points Y where the basis of $\mathcal{D}(\theta) \subseteq \Theta$ changes is of finite cardinality and thus has Lebesgue measure zero. Since for every $\theta \in \Theta \setminus Y$ there is a neighborhood where the basis of the problem with data $\mathcal{D}(\theta)$ stays the same, the optimal simplex tableau is a smooth function of $\mathcal{D}(\theta)$ and therefore $\theta \mapsto V(\theta)$ is smooth in a neighborhood of θ . \square

The result above is very much in line with intuition about linear programming. It is nevertheless non-trivial, since it shows that under the given natural restrictions there is no way to find functions, however complicated, such that there is a significant number of points θ where the optimal value is non-smooth, even if all the data of the problem varies simultaneously.

In particular, Theorem 2 implies that if a point $\theta \in \Theta$ is chosen at random, then the probability that the function $V(\theta)$ is not differentiable at θ has probability zero. This means that the envelope theorem can be applied almost everywhere and the set Y , defined in the proof of Theorem 2, can be ignored in practical applications.

3. Envelope Theorems for Multi-Stage Stochastic Programming

We start this section by applying the results from Section 2 to prove envelope theorems for MSLP problems. These results yield expressions for the derivatives which are independent of the method used to solve the problem. In particular, we argue that when solving discrete MSLPs as deterministic equivalents, our results can be directly applied. In Section 3.2, we review SDDP-type decomposition algorithms to solve MSLPs with an emphasis on *Approximate Dual Dynamic Programming* (ADDP), which is based on scenario lattices as discretizations for Markov processes. We show how ADDP can be used to approximate a policy for a MSLP with continuous randomness. Lastly, we demonstrate how the results from Section 3.1 can be used in sampling-based algorithms to sample derivatives of MSLPs.

3.1. Envelope Theorems for General MSLPs

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space with a probability measure \mathbb{P} , a filtration $\mathcal{F} = (\mathcal{F}_1, \dots, \mathcal{F}_T)$, and $\xi = (\xi_1, \dots, \xi_T)$ a random process with $\xi_t : (\Omega, \mathcal{F}_t) \rightarrow \mathbb{R}^{M_t}$. We denote by $\xi^t = (\xi_1, \dots, \xi_t)$ the history of the random process until period t . Let further Θ be an open set in \mathbb{R} .

We consider a general MSLP with T stages where the relevant data in stage t depends on the realization of the randomness and on a parameter $\theta \in \Theta$. The general form of the problem can be written as

$$V_0(\theta) = \begin{cases} \max_{x_1, x_2, \dots, x_T} & \mathbb{E} \left[\sum_{t=1}^T \langle c_t(\theta, \xi_t), x_t(\theta, \xi_t) \rangle \right] \\ \text{s.t.} & W_t(\theta, \xi_t) x_t(\theta, \xi_t) + T_t(\theta, \xi_t) x_{t-1}(\theta, \xi_t) \leq h_t(\theta, \xi_t), \quad t = 1, 2, \dots, T, \end{cases} \quad (7)$$

where the expectation is with respect to \mathbb{P} and $W_t(\theta, \xi_t)$, $T_t(\theta, \xi_t)$, $c_t(\theta, \xi_t)$, and $h_t(\theta, \xi_t)$ are matrices and vectors of fitting dimension. We define the data of the problem in stage t to be $\mathcal{D}_t(\theta, \xi_t) = (W_t(\theta, \xi_t), T_t(\theta, \xi_t), c_t(\theta, \xi_t), h_t(\theta, \xi_t))$. By stacking all the vectors and columns of the matrices in

$\mathcal{D}_t(\theta, \xi_t)$, we can consider $\mathcal{D}_t(\theta, \xi_t)$ as an element in \mathbb{R}^{N_t} for some $N_t \in \mathbb{N}$ and consider $\mathcal{D}_t(\theta, \xi_t)$ as a function $\mathcal{D}_t: \Theta \times \mathbb{R}^{M_t} \rightarrow \mathbb{R}^{N_t}$. Note that the decisions x_1, \dots, x_T depend on ξ and are therefore random variables. To avoid cluttered notation, in the following, we suppress the dependency of W_t , T_t , c_t , h_t , and x_t on ξ_t .

Note that we formulate the problem in such a way that x_t depends on previous decisions x_1, \dots, x_{t-1} just by its dependency on x_{t-1} . This property can always be enforced by augmenting the state space of period t by the variables in x_1, \dots, x_{t-1} . It follows that the problem can be rewritten in its dynamic programming formulation

$$V_t(x_{t-1}, \xi_t, \theta) = \begin{cases} \max_{x_t} \langle c_t(\theta), x_t \rangle + \mathbb{E}[V_{t+1}(x_t, \xi_{t+1}, \theta) | \xi^t] \\ \text{s.t. } W_t(\theta)x_t + T_t(\theta)x_{t-1} \leq h_t(\theta) \end{cases} \quad (8)$$

with terminal condition $V_{T+1} \equiv 0$ and x_0 given. The assumption of $V_{T+1} \equiv 0$ is not essential, and all results still hold, if V_{T+1} is replaced by a piece-wise linear concave function. For the following, it will be convenient to define so-called *post-decision* value functions (see Powell 2011) as

$$\bar{V}_{t+1}(x_t, \xi^t, \theta) = \mathbb{E}(V_{t+1}(x_t, \xi_{t+1}, \theta) | \xi^t), \quad \forall t \in [T].$$

We proceed by applying the results from Section 2 to a problem where the image measure of ξ is made up of finitely many atoms, i.e., where all conditional distributions $\xi_{t+1} | \xi^t$ are finitely supported. When dealing with discrete processes ξ , the expectations of the value functions V_t , which are concave functions of x_t , can be written as a minimum of finitely many affine functions as stated in the lemma below (for a proof see for example Philpott and Guan 2008, Shapiro 2011, Löhndorf et al. 2013).

LEMMA 2. *If ξ is finitely supported, then, for every realization of ξ_t and fixed $\theta \in \Theta$, $x_t \mapsto \bar{V}_t(x_t, \xi_t, \theta)$ is a concave, piece-wise linear function.*

For what follows we assume that for a given θ , the optimal policy $x_t^*(\theta, \xi)$ is known, i.e., that we know the solution of the MSLP.

THEOREM 3. *Let $\mathcal{D}_t : \Theta \times \mathbb{R}^{M_t} \rightarrow \mathbb{R}^{N_t}$ and $\theta \mapsto \mathcal{D}_t(\theta, \alpha)$ be differentiable, invertible, and bi-continuous for every $\alpha \in \mathbb{R}^{M_t}$ and every $t \in [T]$. Consider a problem of the form (7) based on a discrete process $\xi = (\xi_1, \dots, \xi_T)$ with optimal value $V(\theta)$ and optimal policy x_t^* . If for $\theta \in \Theta$ the problem has a solution, the derivative of the optimal value at θ can be written as*

$$V'(\theta) = \mathbb{E} \left[\sum_{t=1}^T \langle \nabla c_t(\theta), x_t^* \rangle + \left\langle \lambda_t^*, \nabla h_t(\theta) - \frac{d}{d\theta} T_t(\theta) x_{t-1}^* - \frac{d}{d\theta} W_t(\theta) x_t^* \right\rangle \right], \quad (9)$$

for (Lebesgue) almost all $\theta \in \Theta$, where $\lambda_t^* = \lambda_t^*(\theta, \xi^t)$ are the optimal dual solutions associated with the constraints in stage t .

Proof. We start our argument in the last stage T where, for given ξ , θ , and x_{T-1} , the optimization problem is deterministic and equal to

$$V_T(x_{T-1}, \xi_T, \theta) = \begin{cases} \max_{x_T} \langle c_T(\theta), x_T \rangle \\ \text{s.t. } W_T(\theta) x_T + T_T(\theta) x_{T-1} \leq h_T(\theta). \end{cases}$$

By Theorem 1 and Theorem 2 for almost all θ

$$\frac{\partial}{\partial \theta} V_T(x_{T-1}, \xi_T(\theta)) = \langle \nabla c_T(\theta), x_T^* \rangle + \left\langle \lambda_T^*, \nabla h_T(\theta) - \frac{d}{d\theta} T_T(\theta) x_{T-1} - \frac{d}{d\theta} W_T(\theta) x_T^* \right\rangle \quad (10)$$

for the optimal primal and dual solutions x^* and λ^* .

By Theorem 6.7 in Elstrodt (2002) and the fact that V_T is smooth in a neighborhood of a point θ where it is differentiable and that the problem is bounded, we have that

$$\frac{\partial}{\partial \theta} \mathbb{E} [V_T(x_{T-1}, \xi_T, \theta) | \xi^{T-1}] = \mathbb{E} \left[\frac{\partial}{\partial \theta} V_T(x_{T-1}, \xi_T, \theta) | \xi^{T-1} \right]. \quad (11)$$

Note that the set of problematic θ , i.e., the points $\theta \in \Theta$ where V_T is not smooth in θ , changes with each realization of ξ_T . However, since the underlying process is finitely supported, by Theorem 2 there are only finitely many θ where any of the functions $\theta \mapsto V_T(x_{T-1}, \xi_T, \theta)$ are not differentiable.

Combining (10) and (11) yields

$$\begin{aligned} \frac{\partial}{\partial \theta} V_{T-1}(x_{T-2}, \xi_{T-1}, \theta) &= \langle \nabla c_{T-1}(\theta), x_{T-1}^* \rangle + \mathbb{E}[\langle \nabla c_T(\theta), x_T^* \rangle | \xi_{T-1}] \\ &\quad + \left\langle \lambda_{T-1}^*, \nabla h_{T-1}(\theta) - \frac{d}{d\theta} T_{T-1}(\theta) x_{T-2} - \frac{d}{d\theta} W_{T-1}(\theta) x_{T-1}^* \right\rangle \end{aligned}$$

$$+ \mathbb{E} \left[\left\langle \lambda_T^*, \nabla h_T(\theta) - \frac{d}{d\theta} T_T(\theta) x_{T-1} - \frac{d}{d\theta} W_T(\theta) x_T^* \right\rangle \middle| \xi^{T-1} \right].$$

Note that Theorem 1 and Theorem 2 apply to the above problem, since the post-decision value function is a piece-wise linear function by Lemma 2, which can be represented in an linear programming formulation.

Continuing in this fashion, we arrive at

$$V'(\theta) = \mathbb{E} \left[\sum_{t=1}^T \langle \nabla c_t(\theta), x_t^* \rangle + \left\langle \lambda_t^*, \nabla h_t(\theta) - \frac{d}{d\theta} T_t(\theta) x_{t-1}^* - \frac{d}{d\theta} W_t(\theta) \right\rangle \right].$$

Note that, since there are finitely many stages and that in each stage there are finitely many points $\theta \in \Theta$ where the corresponding value functions are not differentiable, the union of these points still has finite cardinality and therefore is a Lebesgue null set. \square

As mentioned in Section 2, the fact that the derivative exists (Lebesgue) almost surely guarantees that the probability of picking a θ at random where $V'(\theta)$ does not exist is zero, which implies that this possibility can be ignored in practice. Of course, this argument is predicated on the tacit assumption of a continuous distribution on Θ governing the random choice of θ . If the sampling distribution for some reason would have atoms at the discontinuities of the value function, this would obviously not hold.

Note that the above derivative can be readily computed, if the stochastic optimization problem is solved as one large monolithic linear program, via a deterministic equivalent formulation, for example using scenario trees. In this case all primal and dual solutions for all possible scenarios are known and (9) can be easily evaluated.

Next we investigate the case when ξ has a continuous distribution. Since in this case the post-decision value functions are no longer piece-wise linear, the above proof does not work. We circumvent this problem, by applying Lemma 1 to the space of data vectors \mathcal{D}_t spanned by the random elements in ξ_t and requiring stricter smoothness conditions on the coefficient functions.

THEOREM 4. *Let $\mathcal{D}_t : \Theta \times \mathbb{R}^{M_t} \rightarrow \mathbb{R}^{N_t}$ be a function such that $\alpha \mapsto \mathcal{D}_t(\theta, \alpha)$ is a smooth diffeomorphism for all $\theta \in \Theta$ and all $t \in [T]$. Let further $V(\theta)$ be the optimal value of a problem of the form*

(7) and all the conditional distributions $\xi_t|\xi_{t-1}$ be absolutely continuous with respect to the Lebesgue measure in \mathbb{R}^{M_t} . Then $V'(\theta)$ can be computed by formula (9) as in Theorem 3 for all $\theta \in \Theta$.

Proof. Note that (10) still holds. For a given θ and a given $\alpha \in \mathbb{R}^{M_{T-1}}$, the problem in stage $(T-1)$ can be written as

$$\begin{aligned} & \max_{x_{T-1}} \langle c_{T-1}(\theta, \alpha), x_{T-1} \rangle + \mathbb{E}[V_T(x_{T-1}, \xi_T, \theta) | \xi^{T-1}(\theta, \alpha) = \alpha] \\ & \text{s.t. } W_{T-1}(\theta, \alpha)x_{T-1} + T_{T-1}(\theta, \alpha)x_{T-2} \leq h_{T-1}(\theta, \alpha), \end{aligned}$$

which is equivalent to the linear optimization problem

$$\begin{aligned} & \max_{x_{T-1}} \langle c_{T-1}(\theta, \alpha), x_{T-1} \rangle + \bar{V}_T(x_{T-1}^*, \alpha, \theta) + \langle x_{T-1}^* - x_{T-1}, \nabla \bar{V}'(x_{T-1}^*, \alpha, \theta) \rangle \\ & \text{s.t. } W_{T-1}(\theta, \alpha)x_{T-1} + T_{T-1}(\theta, \alpha)x_{T-2} \leq h_{T-1}(\theta, \alpha), \end{aligned}$$

where x_{T-1}^* is the optimal solution to the problem.

By Corollary 1, the M_t dimensional subspace in \mathbb{R}^{N_t} which is affected by changes in α can be partitioned into $I \in \mathbb{N}$ smooth manifolds \mathcal{S}_i , $i \in I$ such that the basis of the resulting linear optimization problem remains constant. By the properties of the mapping \mathcal{D}_t , the sets

$$\mathcal{T}_i = \mathcal{S}_i \cap \mathcal{D}_t(\theta, \mathbb{R}^{M_{T-1}})$$

are smooth manifolds too, and so are the sets

$$\mathcal{V}_i = \mathcal{D}_t(\mathcal{T}_i, \theta)^{-1} \subseteq \mathbb{R}^{M_{T-1}}.$$

We have established that for a given θ the sets \mathcal{V}_i are realizations of the random parameters of the problem that all lead to the same optimal basis. Therefore a basis change can only occur at the boundaries of sets \mathcal{V}_i , which are themselves smooth manifolds of dimension at most $(M_{T-1} - 1)$. By an application of Sard's theorem (see for example Lemma 13.5 in Maggi 2012) it follows that such sets have Lebesgue measure 0.

Therefore the points α where the value function of the corresponding problem is not differentiable form a set Y with Lebesgue measure 0 and, by the assumption on absolute continuity of the underlying process in $\mathbb{R}^{M_{T-1}}$, has probability 0. It follows that the set Y can be ignored when computing the derivative of \bar{V}_{T-2} . The rest of the proof proceeds by backward induction in the same manner as the proof of Theorem 3. \square

Note that in the continuous case, the smoothing effect caused by the expectation results in differentiability *everywhere* – not just *almost everywhere* as in the discrete case. As argued above for practical application this distinction is inconsequential.

3.2. Approximate Dual Dynamic Programming

In this section, we briefly describe SDDP-type methods with a focus on the particular variant ADDP to be used in Section 4 to compute derivatives by sampling. SDDP solves linear MDPs with relatively complete recourse by iteratively approximating the value functions of the problem (Pereira and Pinto 1991, Philpott and Guan 2008, Shapiro 2011).

A problem of the form (7) is an MDP if the underlying random process ξ is a Markov process. Note that due to the structure of the problem and the Markov property of the process, it is enough to know the state (x_{t-1}, ξ_t) to make a decision in t . This is in contrast to the more general formulation discussed in the last section, where the whole history of the process ξ^t may be required for decision making.

In the following, we call ξ_t the environmental state of the problem, while x_{t-1} is called the resource state. Together ξ_t and x_{t-1} constitute the state of the MDP. Note that the resource state can be influenced by the decisions in earlier stages while the environmental state evolves independently of the decisions. This conceptual separation enables us to approximate the value functions in two steps: First, we search for an optimal set of representative discrete states for the environmental state, which we organize in a *scenario lattice*. Second, we use a version of SDDP that approximates the value function at each node of the lattice by a concave, piece-wise linear function of the resource state. This contrasts conventional SDDP, as it allows the data process to be Markovian, rather than requiring stage-wise independence.

We refer to this two-stage procedure as *Approximate Dual Dynamic Programming* to emphasize that we construct an approximate solution for an MDP which involves an approximation in the Markov process (the scenario lattice) and an approximation in the solution (the approximate value functions).

The standard discrete representations of random processes are *scenario trees*, which allow for a general theory of stochastic optimization encompassing problems without relatively complete recourse which need not be necessarily linear (or even convex) in their decisions and are allowed to depend on the entire history ξ^t of the stochastic process (see, e.g., Pflug and Pichler 2014). A scenario tree represents the random evolution of a stochastic process as a graph containing nodes and arcs (see Figure 2, left). Nodes represent states of the stochastic process and are organized in layers which correspond to points in time that form the stages of the stochastic optimization problem. Arcs exist between nodes in consecutive stages and represent possible state transitions with associated probabilities. The probabilities of all arcs leaving a node sum to 1 and model the conditional probability of state transitions in this node. The tree contains a single root node at stage 1, which represents the state of the process in the deterministic present.

Scenario trees are trees in a graph theoretic sense, because every node has a unique predecessor resulting in a graph without cycles. This requirement implies that at a given node of a tree the entire history of the process can be reconstructed by tracing state transitions backwards to the root node of the tree.

When the underlying stochastic process is Markovian, the conditional distributions of future randomness at stage t , and therefore the decisions, do not depend on the entire history ξ^t but only on the environmental state ξ_t . If we would discretize the Markov process to a scenario tree, many branches of the tree would have identical sub trees which can be merged without loss of information. We therefore use a slight modification of scenario trees which make use of this property by allowing for a node to have multiple predecessors. In line with the terminology often used in mathematical finance, we refer to such a *recombining* scenario tree as a *scenario lattice*. Scenario lattices are natural representations of Markov processes in the same way as scenario trees are natural representations of general stochastic processes.

The obvious advantage of lattices over trees is the possibility of recombination, which makes it possible to represent more scenarios with a given number of nodes as is demonstrated in Figure 2 on the right. This avoids the exponential explosion of nodes that inevitably occurs in scenario

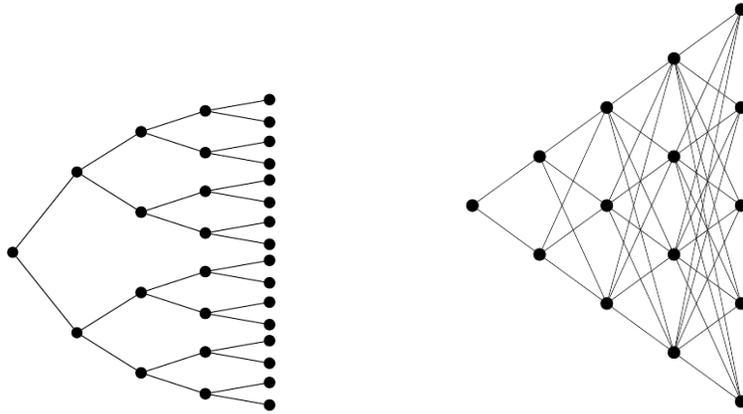


Figure 2 A tree with 31 nodes representing 16 scenarios on the left and a lattice with 15 nodes representing 120 scenarios on the right.

trees with purely stochastic state transitions, i.e., in trees where every node has more than one successor. The ensuing complexity of the associated stochastic optimization problems is a challenge for scenario tree-based stochastic optimization with many decision stages.

The goal of lattice construction is to build approximations such that the optimal policy for the lattice process yields a close to optimal policy for the true process. For the purpose of this paper, we use a stochastic gradient algorithm outlined in Bally and Pagès (2003) to construct scenario lattices from simulations of the environmental state ξ of the problem. We denote the lattice process by $\tilde{\xi} = (\tilde{\xi}_1, \dots, \tilde{\xi}_T)$, where the nodes in stage t are denoted by $\tilde{\xi}_{tk}$, $k \in [N_t]$.

Once we have constructed a scenario lattice as a discrete representation of the underlying stochastic process, we solve the resulting MDP using SDDP, a simulation-based algorithm that iteratively refines approximations \hat{V}_{tk} of the post-decision value function on the nodes of the lattice. SDDP alternates between simulating the approximate optimal policy and updating the approximate post-decision value functions, which, in turn, define the approximate optimal policy of the next iteration. The approximate value function at stage t in node k is represented by a point-wise maximum of supergradients of the true post-decision value functions \bar{V}_{tk} . Convergence follows by the fact that \bar{V}_{tk} are piece-wise linear by Lemma 2 and therefore only a finite number of cuts are required to make the approximation exact.

More specifically, at each iteration, $l \in [L]$, SDDP performs a *forward pass*, during which a sequence of paths on the lattice, $(\hat{\xi}_t^l)_{t=1}^T$, are sampled to generate a sequence of resource states, $(\hat{x}_t^l)_{t=1}^T$. Denoting by $\hat{V}_{t+1,k}^l(x_t, \tilde{\xi}_{tk}, \theta)$ the approximate post-decision value functions after the l -th iteration of the algorithm for node $\tilde{\xi}_{tk}$ in stage t , the sampled resource states can be obtained by following the incumbent approximate optimal policy, i.e.,

$$\hat{x}_t^l \in \arg \max_{x_t} \left\{ \langle \hat{c}_t^l, x_t \rangle + \hat{V}_{t+1,k}^l(x_t, \hat{\xi}_t^l, \theta) : \hat{W}_t^l x_t + \hat{T}_t^l \hat{x}_{t-1}^l \leq \hat{h}_t \right\}, \quad t \in [T], \quad (12)$$

where $\hat{x}_0^l = x_0$, $\hat{V}_{T+1}^0 \equiv 0$ and the data $\hat{\mathcal{D}}_t$ of the problem is a function of $\hat{\xi}_t^l$ and θ .

After each forward pass, SDDP performs a *backward pass* using the dual solution of the problems (12) to add new hyperplanes to the model of the approximate post-decision value functions. The approximate value functions obtained in this way are upper bounds of the problem's true value functions and the approximate optimal policy converges to the optimal policy almost surely in finitely many iterations (see Philpott and Guan 2008, Shapiro 2011, Löhdorf et al. 2013, for further details).

The approximate optimal policy $\hat{x}^* = (\hat{x}_t^*)_{t=1}^T$ is defined as in (12) with $\hat{V}_{t+1,k}^l$ replaced by the final value function approximations \hat{V}_t after convergence of the algorithm. This policy is optimal for the Markov process represented by the scenario lattice. However, we can lift the policy to the original - possibly continuous - process used to generate the lattice. To that end, for an observed trajectory ξ_1, \dots, ξ_T of the original process we compute the decisions as follows

$$\hat{x}_t^* \in \arg \max_{x_t} \left\{ \langle c_t, x_t \rangle + \hat{V}_t(x_t, \tilde{\xi}_{tn}, \theta) : W_t x_t + T_t \hat{x}_{t-1}^* \leq h_t \right\}, \quad (13)$$

whereby $n = \arg \min_m \left\{ \|\tilde{\xi}_{tm} - \xi_t\|_2^2, m \in [N_t] \right\}$. This implies that the problem is solved with the data $\mathcal{D}_t(\theta, \xi_t) = (W_t(\theta, \xi_t), T_t(\theta, \xi_t), c_t(\theta, \xi_t), h_t(\theta, \xi_t))$ determined by the sample ξ_t using the post-decision value function $x_t \mapsto \hat{V}(x_t, \tilde{\xi}_{tn})$ from the lattice node $\tilde{\xi}_{tn}$ that is closest to ξ_t . For use in Algorithm 1 below, we define

$$\mathcal{S}_t(\xi_t, x_{t-1}, \theta) = \{(x_t^*, \lambda_t^*) : x_t^*, \lambda_t^* \text{ are primal and dual optimal solutions of (13)}\}.$$

The above approximation yields a policy on the original process that is piece-wise constant in the environmental state ξ_t and piece-wise linear in the resource state x_{t-1} . Rounding to a lattice node is made possible by the fact that each node contains a value function which can be used to make decisions for all possible resource states in t .

Note that problem (13) and its solution are entirely determined by the simulation of the process at time t , ξ_t , and the initial resource state of the problem. Compare this to the situation where scenario trees are used: in a tree each node carries the information on the whole history ξ^t of the process and thereby also a unique decision x_t . Hence, while it is possible to round to the next node in a scenario tree in an analogous fashion, there is no way to come to a decision other than the one stored on the node without solving the whole problem again for the rest of the planning horizon.

Hence, the policy resulting from the ADDP algorithm is flexible enough to deal with different resource states and thereby to evaluate the resulting policy on processes other than the one represented by the scenario lattice used to solve the problem. This is an advantage in general and also when computing derivatives as we will demonstrate in Section 4.

3.3. Sampling Derivatives in SDDP

As discussed in the previous section, ADDP solves MSLPs by approximating the value functions of the problems. The resulting approximations are used in problems of the type (8) to find approximately optimal decisions. Hence, for a given realization of ξ , the problem is solved time step per time step for that specific realization. The expressions for the derivatives found in Theorem 3, which are based on primal and dual solutions for all scenarios, can therefore not be evaluated directly. However, the approximate value functions effectively define a policy that can be used to generate samples of optimal decisions, which can be used to approximate (9).

We start by proving a result about the properties of sampled estimates for derivatives of MSLPs, which assumes that the optimal policy is known to the decision maker and can be used to sample primal and dual optimal solutions.

THEOREM 5. *Let the conditions for Theorem 3 or Theorem 4 be fulfilled and let $(\hat{\xi}_{tn}^k)_{t \in [T], n \in [N_t], k \in [K]}$ be an i.i.d. sample of size $K \in \mathbb{N}$ from the process ξ and $\theta \in \Theta$. For $k \in [K]$, denote the resulting data as $\mathcal{D}_t^k = (W_t^k(\theta, \hat{\xi}_t^k), T_t^k(\theta, \hat{\xi}_t^k), c_t(\theta, \hat{\xi}_t^k), h_t^k(\theta, \hat{\xi}_t^k))$ and let x_{kt}^* and λ_{kt}^* be the optimal primal and dual solution for the corresponding scenarios.*

1. *The sample average estimate*

$$\hat{V}'_K(\theta) = K^{-1} \sum_{k=1}^K \sum_{t=1}^T \langle \nabla c_t^k, x_{kt}^* \rangle + \lambda_{tk}^* \left(\nabla h_t^k - \frac{d}{d\theta} T_t^k x_{k,t-1}^* - \frac{d}{d\theta} W_t^k(\theta') x_{k,t}^* \right). \quad (14)$$

converges almost surely to the true value $V(\theta)$ for (almost) all $\theta \in \Theta$ as $K \rightarrow \infty$.

2. *If the second moment of $V'_K(\theta)$ exists, then*

$$\frac{\sqrt{K}}{\sigma} (\hat{V}'_K(\theta) - V'(\theta)) \xrightarrow{d} \mathcal{N}(0, 1), \text{ as } K \rightarrow \infty$$

where \xrightarrow{d} denotes convergence in distribution and σ is the standard deviation of V'_1 .

Proof. The proofs of 1 and 2 follow directly from the law of large numbers and the central limit theorem, respectively. \square

The second part of the above theorem allows to construct confidence regions around the sampled derivatives $\hat{V}'_K(\theta)$. The existence of second moments, which is a condition for the application of the central limit theorem, can, for example, be established by the boundedness of optimal solutions in combination with $\nabla_{\theta} \mathcal{D}_t \in \mathcal{L}^1(\Omega, \mathcal{F}_t)$ for all $t \in [T]$. These conditions are usually fairly easy to check in real-world examples as will be demonstrated in Section 4.

In the case where the original process is known to be the discrete Markov process or if there is no other model of the randomness available other than the scenario lattice, the above theorem can directly be used to calculate derivatives of the optimal value. This might for example be the case, if the scenario lattice is directly estimated from data without the intermediary step of a statistical model (see for example Löhndorf et al. 2013).

When approximating processes ξ by lattices $\tilde{\xi}$, we obtain policies that are optimal for $\tilde{\xi}$, instead of the actual process ξ , and the derivatives based on samples from $\tilde{\xi}$ will, according to Theorem 5, yield the derivative of the optimal value of the stochastic optimization problem defined on $\tilde{\xi}$.

Data: $K \in \mathbb{N}$, x_0 , θ , $(\tilde{\xi}_{tn})_{t \in [T], n \in [N_t]}$, $(\hat{V}_{tn})_{t \in [T], n \in [N_t]}$

Result: $\hat{V}'_K(\theta)$

```

1  $\hat{V}'_K(\theta) \leftarrow 0$ 
2 for  $k \leftarrow 1$  to  $K$  do
3    $\hat{\xi}_1^k = \xi_1 = (W_1, T_1, c_1, h_1)$ 
4   for  $t \leftarrow T$  to  $1$  do
5     Find  $(x_{tn}^*, \lambda_{tn}^*) \in \mathcal{S}_t(\xi_t^k, x_{t-1,n}^*, \theta)$ .
6      $\hat{V}'_K(\theta) \leftarrow \hat{V}'_K(\theta) + \frac{1}{K} [\langle \nabla c_t^k, x_{kt}^* \rangle + \lambda_{kt}^* (\nabla h_t^k - \frac{d}{d\theta} T_t^k x_{k,t-1}^* - \frac{d}{d\theta} W_t^k(\theta) x_{kt}^*)]$ 
7      $\hat{\xi}_t^k = (W_t^k, T_t^k, c_t^k, h_t^k) \leftarrow \text{sample}_t(\hat{\xi}_t, \hat{\xi}_{t-1}^k)$ 
8   end
9 end

```

Algorithm 1: Pseudo-code for the computation of derivatives in ADDP.

This differs from the derivative based on the true process ξ for two reasons: firstly, the process is changed from ξ to $\tilde{\xi}$ and secondly, the applied policy is optimal for the lattice process $\tilde{\xi}$ and not for the original process. Hence, the resulting derivatives will only be an approximation of the real ones, which, if the approximation of ξ by $\hat{\xi}$ is good, will be sufficiently accurate.

We apply the procedure described in Algorithm 1 to obtain the samples used to calculate derivatives of the optimal value. Apart from $\theta \in \Theta$ and an initial resource state x_0 , the algorithm takes as inputs the optimal policy found by ADDP in the form of the lattice discretization of the process $(\tilde{\xi}_{tn})_{t \in [T], n \in [N_t]}$, and the set of approximate value functions $(\hat{V}_{tn})_{t \in [T], n \in [N_t]}$ – one per node of the scenario lattice. The algorithm samples trajectories $(\hat{\xi}_t^k)_{k \in [K], t \in [T]}$ of the stochastic process (line 7) and in every period t for every k solves problem (13) by finding the lattice node closest to $\hat{\xi}_t^k$ (line 5) and then using the respective post-decision value function stored at that node to make a decision. Note that, as in (13), the rest of the problem data is determined by $\hat{\xi}_t^k$.

If the original process ξ is not given as a lattice, then there are two possible ways of sampling in line 7 of the algorithm: samples can be drawn from the lattice $\tilde{\xi}$ or they can be sampled from the original process ξ . In the first case there is no need for the rounding in line 5 and the computed derivatives are the derivatives of the problem formulated on the lattice.

As we will demonstrate for an example in the next section, we can improve the estimate for the derivative by applying the optimal policy found with ADDP on the lattice to samples from the original process, i.e., use the second method of sampling and lift the policy as described in (13) to make decisions. In this case, the applied policy is no longer optimal for the process used for sampling. However, we can still use Theorem 5 to calculate the derivative of the value of the lifted policy for the real process. It turns out that sampling from the true process increases the accuracy of the estimates for the derivatives in an example considered in the next section, because it removes part of the bias resulting from the discretization error which is introduced by the lattice approximation.

4. Numerical Examples

In this section, we calculate derivatives for optimal values in two examples. As benchmarks, we use analytical solutions in one of the problems and naïve estimates for the derivatives, which are based on difference quotients.

More specifically, for a problem with an optimal value $V(\theta)$ depending on a parameter θ , we calculate the symmetric difference quotient

$$\hat{V}'_{DQ} = \frac{V(\theta_0 + \epsilon) - V(\theta_0 - \epsilon)}{2\epsilon} \quad (15)$$

as an approximation of the derivative. We use the symmetric difference, since it yields a more stable estimate of the derivative than the one-sided difference quotient. Nevertheless, the expression (15) is susceptible to several distortions. Firstly, for any given $\epsilon > 0$, $\hat{V}'_{DQ}(\theta)$ is only an approximation of $V'(\theta)$, which gets better as $\epsilon \rightarrow 0$. Secondly, for small ϵ , errors in the evaluation of \hat{V}'_{DQ} are amplified due to the numerical instability of the expression, which arises because of the division by the small number ϵ . In fact, if ϵ is close to machine precision ε , the calculated values are essentially meaningless. Thirdly, in our case the values of $V(\theta_0 \pm \epsilon)$ are calculated using a sampling-based algorithm and are therefore random, which introduces another level of inaccuracy.

Dealing with the first two problems involves finding an ϵ which represents a good trade-off between the bias introduced by larger values of ϵ and the instability introduced by smaller levels of

ϵ . Unfortunately, there is no exact method to find an optimal trade-off between these two opposing effects. Defining ε to be the machine precision, we choose

$$\epsilon = \theta \sqrt[4]{\varepsilon}$$

using a rule of thumb described in Jäckel (2002). Furthermore, we mitigate the problem of randomness in the estimate by using the same sample $\hat{\xi}$ for the evaluation of $V(x + \epsilon)$ and $V(x - \epsilon)$. As argued in Shapiro et al. (2014), this reduces the variance of the estimate \hat{V}'_{DQ} , provided that the covariance of $V(x + \epsilon)$ and $V(x - \epsilon)$ is positive, which is obviously true in our case.

All computations presented in this section have been performed with MATLAB and the stochastic optimization problems have been solved using the MATLAB interface of QUASAR (see www.quantego.com), which provides an implementation of ADDP as a JAVA library. Unless otherwise stated, we use the actual process ξ and not the lattice process $\tilde{\xi}$ for sampling the derivatives as described in Section 3.

4.1. The Newsvendor Problem

We first study a two-stage newsvendor problem, which is a classical example in stochastic programming reminiscent of basic procurement and inventory problems. Although the problem only has two stages, it is still useful to us, since it is one of the few stochastic optimization problems with a closed-form solution, allowing us to compare approximated derivatives to the true derivatives of the optimal value.

The problem is set up as follows: The newsvendor decides in the morning how many newspapers she will purchase from the publisher, not knowing the demand during the day. She sells the papers to her clients for a known price, and excess newspapers have to be discarded at the end of the day. The problem is cast as a two-stage stochastic optimization problem: in the first stage the newsvendor decides how many papers to order, in the second stage (knowing the demand) she decides how many papers to sell and how many to discard.

Let x_b be the amount of ordered newspapers, x_s be the amount of papers sold, and D be the random demand during the day. Then the problem can be written as the following two-stage MSLP

$$V = \begin{cases} \max_{x_b, x_s} & -cx_b + \mathbb{E}[px_s] \\ \text{s.t.} & x_s \leq x_b, \quad \text{a.s.} \\ & x_s \leq D, \quad \text{a.s.} \\ & x_b \geq 0, \end{cases}$$

where p is the sales price and c is the price the newsvendor pays to the publisher. The Lagrangian of the problem is

$$\mathcal{L}(x_b, x_s, \lambda_1, \lambda_2, \lambda_3) = -cx_b + \lambda_1 x_b + \mathbb{E}[px_s + \lambda_2(x_b - x_s) + \lambda_3(D - x_s)]. \quad (16)$$

Assuming that $D \sim F$, it follows from elementary calculation (see Birge and Louveaux 2011) that the optimal solution and optimal value of the problem are

$$x_b^* = F^{-1}\left(\frac{p-c}{p}\right), \quad V = -cx_b^* + p \int_{-\infty}^{\infty} \min(x, x_b^*) dF(x),$$

respectively.

For our computations, we will use $p = 1$, $c = 0.2$ and a normal distribution with mean $\mu = 100$ and $\sigma = 20$ to model the random demand D . For later use, we write $D = \mu + \sigma X$, where $X \sim N(0, 1)$. We use scenario lattices with 100 nodes in the second stage that are built from 100,000 samples of F .

We start by calculating derivatives with respect to the sales price p , which is an example where $\theta = p$ appears in the objective function coefficients. For this purpose, we use the linear function $p \mapsto p$, which clearly fulfills the requirements of Theorems 2 and 3. Furthermore, the setup of the problem together with the choice of the normal distribution fulfills the conditions of Theorem 5 regarding the existence of second moments. According to these results, we have

$$V'(p) = \frac{\partial}{\partial p} \mathcal{L}(x_b^*, x_s^*, \lambda_1, \lambda_2, \lambda_3) = \mathbb{E}(x_s^*),$$

where x_s^* are the optimal sales decisions. Following Theorem 5, we obtain the approximation

$$V'(p) = \mathbb{E}(x_s^*) \approx K^{-1} \sum_{k=1}^K x_{ks}^* = \hat{V}'_K(p)$$

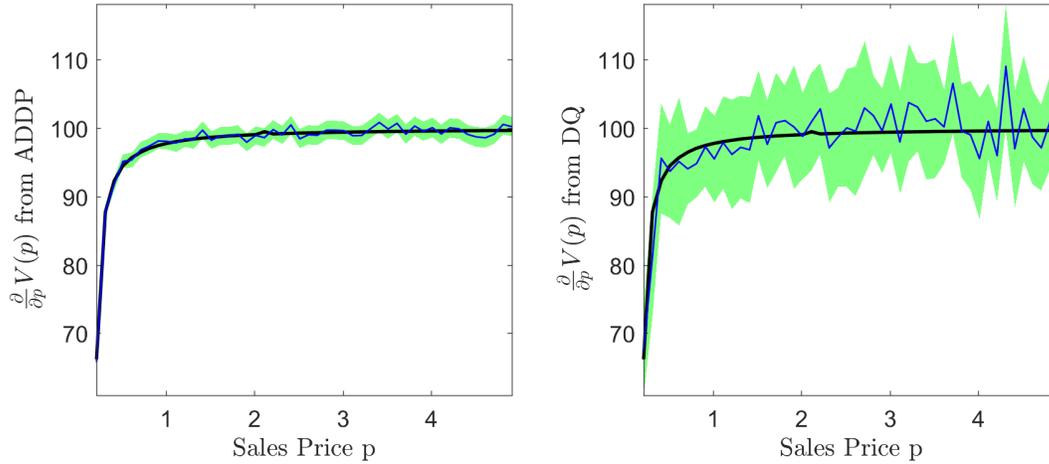


Figure 3 Derivative of the value function V with respect to sales price p as a function of p evaluated at 48 equally spaced points between 0.1 and 5. The bold black line is the true derivative.

by drawing $K = 1,000$ demand samples from $D \sim N(\mu, \sigma)$ and using the optimal policy found by ADDP in the scenario lattice to calculate optimal solutions x_{ks}^* .

Figure 3 shows the derivative of the value function with respect to the sales price. In both plots, the bold black curve corresponds to the true derivative computed from the analytical solution. The plot on the left illustrates the results using the sampling-based method, while the plot on the right displays the numerical derivative computed using the difference quotient method. The green regions are the confidence intervals around the respective solutions, which are calculated using Theorem 5 for the ADDP and normal confidence regions for the difference quotient with standard deviations calculated from 30 independent evaluations of \hat{V}'_{DQ} .

The plot shows that $\hat{V}'_K(p)$ is more precise than $\hat{V}'_{DQ}(p)$ in the sense that deviations from the actual derivative are smaller for the ADDP-based method (0.5011 on average) than for the method based on the difference quotients (2.168 on average). Furthermore, we see that the confidence bounds for $\hat{V}'_K(p)$ are clearly narrower than those of $\hat{V}'_{DQ}(p)$, indicating that the sampling-based estimate is more reliable.

Turning to the issue of computational cost, we note that the computation of \hat{V}'_{DQ} requires two evaluation of the optimal value, while sampling from the optimal policy described in Algorithm 1

is comparatively cheap. Therefore, we expect the runtime of the computation of \hat{V}'_{DQ} to be roughly twice of the runtime required to compute \hat{V}'_K . To obtain a measurement, we recorded the runtimes required for both estimates for the 48 measurements of \hat{V}'_{DQ} and \hat{V}'_K depicted in Figure 3. In line with the above, we find the ratio between the average computation time of \hat{V}'_{DQ} and \hat{V}'_K to be $r = 1.91 \pm 0.21$. This confirms that \hat{V}'_K is computationally superior to \hat{V}'_{DQ} .

Next, we calculate the derivative of the optimal value function with respect to the standard deviation of the random demand, which is a parameter on the right-hand side of the constraints of the optimization problem. To that end, we use the linear function in $\sigma \mapsto \mu + \sigma X$, which fulfills the requirements of Theorems 3 and 4 as well as the conditions on the moments in Theorem 5.

Taking the derivative of the Lagrangian with respect to σ yields

$$V'(\sigma) = \frac{\partial}{\partial \sigma} \mathcal{L}(x_b^*, x_s^*, \lambda_1^*, \lambda_2^*, \lambda_3^*) = \mathbb{E}[\lambda_2^* X],$$

where λ_2^* is the optimal dual solution corresponding to the constraint $x_s \leq D$. Hence, the derivative with respect to σ can be obtained by multiplying λ_2^* with the realizations $X \sim N(0, 1)$, which define D . As before, to calculate derivatives, we approximate

$$V'(\sigma) = \mathbb{E}[\lambda_2^* X] \approx K^{-1} \sum_{k=1}^K \lambda_{2tk}^* X^k = \hat{V}'_K(\sigma).$$

The results of the comparison between \hat{V}'_{DQ} and $\hat{V}'_K(\sigma)$ is shown in Figure 4. We again use $K = 1,000$ samples for the estimation of $\hat{V}'_K(\sigma)$ and computed the confidence interval for $\hat{V}'_{DQ}(\sigma)$ based on 30 independent repetitions of the calculation.

Again, our method clearly outperforms the benchmark in terms of accuracy, with an average errors of 0.0171 versus 0.0463 for \hat{V}'_{DQ} , as well as in the variance of the estimator as can be seen by the green confidence bands.

As mentioned above, we solve MSLPs using scenario lattices as discretizations of the underlying stochastic process but use simulations from the actual process, in this case from the normal distribution, to calculate the derivatives by rounding to the nearest lattice node as described in Section 3.3. In the following, to evaluate the merit of this strategy, we systematically compare the errors

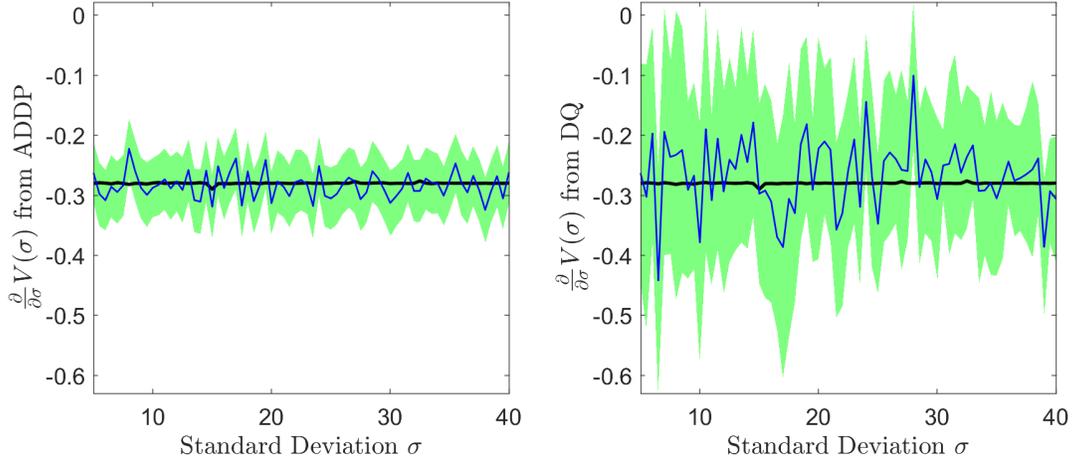


Figure 4 Derivative of the value function V with respect to sales price σ as a function of σ evaluated at 71 equally spaced points between 5 and 40. The bold black line is the true derivative.

(relative to the true derivatives), when computing $\hat{V}'_K(p)$ and $\hat{V}'_K(\sigma)$ based on simulations from the lattice and the true process for a varying number of nodes in the lattice and a varying number of samples used to calculate the derivatives. In particular, we use 50, 100, 200 lattice nodes per stage and 1,000, 10,000, and 100,000 samples to compute derivatives.

Since the calculated derivatives are subject to random variations, we repeat calculations 30 times, each time varying the parameter p from 0.3 to 5 in steps of 0.1 and the parameter σ from 5 to 40 in steps of 1. We repeat this analysis for all combinations of nodes and number of samples.

We define

$$\Delta_{lat}^{p,s} = \sum_{l=1}^L |\hat{V}'_{K,lat}(p_l) - V'(p_l)|, \quad \Delta_{proc}^{p,s} = \sum_{l=1}^L |\hat{V}'_{K,proc}(p_l) - V'(p_l)|$$

where p_l are the prices for which the derivatives $\hat{V}'_{K,lat}$ and $\hat{V}'_{K,proc}$ are estimated based on samples from the lattices and the processes, respectively. We report the differences in average errors, i.e.,

$$\Delta^p = \frac{1}{30} \sum_{s=1}^{30} \Delta_{lat}^{p,s} - \Delta_{proc}^{p,s}$$

and test whether these quantities are significantly different from 0 using unpaired two-sample t-tests. The values for Δ^σ are computed in an analogous fashion.

Furthermore, we report the frequency $F^{lat,proc}$ with which the true derivative of the optimal value is outside the symmetric 98% confidence bounds from Theorem 5 around the estimates $\Delta_{lat,proc}^{p,s}$ and $\Delta_{lat,proc}^{\sigma,s}$. Deviations from the expected 2% violations occur for two reasons: Firstly, we solve an approximate (discrete) problem, which introduces a bias to the estimates. Secondly, when sampling from the lattice the samples do not come from the true distribution but from an approximation. By sampling from the *actual* normal distribution, we can avoid the second source of distortion.

The results of the analysis are reported in Table 1. Inspecting the outcomes for $\Delta^{p,\sigma}$, we see that the numbers are mostly positive, implying that the errors are larger when sampling from the scenario lattice. The only exception are two values for 50 and 100 nodes and 1,000 samples, which are the cases where the variance of the estimates is the highest, due to the relatively small number of samples. Correspondingly, we note that these two values are not significantly different from 0. As the number of samples increases the estimates from the process perform markedly better and differences are significant at least at the 5% level and unambiguously at the 0.1% level for estimates based on 100,000 samples. We conclude that the bias in the estimates can be consistently reduced when sampling from the true underlying stochastic process by lifting the policy from the lattice to the continuous distribution of demands.

Turning our attention to the violations $F^{lat,proc}$ in the two right panels of Table 1, we observe that both F^{lat} and F^{proc} are decreasing in the number of nodes. Having more nodes improves the quality of the approximation by the scenario lattice and thereby reduces the bias between the true and the approximated solution, bringing F closer to its correct value of 2%. Furthermore, we observe that F increases in the number of samples used to calculate the derivatives. The reason for this is that confidence bands get narrower when the number of samples increases, and the discretization bias is therefore more often *detected*. Comparing the values for lattices and the process, we notice that, as expected, confidence intervals for the estimates based on the samples from the process are more accurate than those computed from lattice samples.

We also note that for both lattice- and process-based estimates, the absolute difference between the analytical derivative and the sampled derivative reduces with increasing sample size and with

Samples	Nodes	$\Delta^{p,\sigma}$			$100 \times F^{lat}$			$100 \times F^{proc}$		
		1k	10k	100k	1k	10k	100k	1k	10k	100k
Price	50	-0.1238	0.2953*	0.5875***	2.8	7.3	21.7	2.6	6.1	17.6
	100	-0.2516	0.2904*	0.7348***	2.3	4.7	16.7	2.7	3.9	8.5
	200	0.1843	0.4501***	1.1861***	2.2	3.5	15.8	1.9	2.4	4.4
Std	50	0.0177	0.0725***	0.0821***	5.7	14.9	55.0	5.3	8.5	42.6
	100	0.0216	0.0362***	0.0568***	5.3	8.9	40.7	4.6	6.3	22.6
	200	0.0135	0.0099*	0.0261***	4.9	5.8	19.9	5.0	5.3	9.0

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table 1 Comparison of average absolute errors for derivatives with respect to sales price and standard deviation of demand when sampling from the scenario lattice and the process in the first panel. Percentage violations of the confidence bounds in panel 2 and 3.

an increased number of nodes in the scenario lattice, both of which improve the approximation for obvious reasons. Results of this analysis are available upon request.

Summarizing, we remark that, if possible, it is preferable to sample from the true underlying process instead of from the discretization used to solve the stochastic optimization problem. We view the possibility to do so as a major advantage of SDDP-type solution methods, which yield policies that can be evaluated for arbitrary stochastic processes by rounding.

4.2. Gas Storage

In this section, we study the problem of gas storage pricing and operation on the spot market for natural gas hosted on the *National Balancing Point* (NBP) in the United Kingdom as a multi-stage stochastic optimization problem. The aim of planning is to compute the value of owning a storage plant for one year based on optimal operational and trading strategies, which capitalizes on the yearly seasonality of gas prices as well as on statistical arbitrage from short-term fluctuations in gas prices.

We formulate the problem as the following MSLP with random spot prices for gas P_t in stage t of the problem

$$V = \begin{cases} \max_{x_t^b, x_t^s, s_t} & \sum_{t=1}^T \mathbb{E}[P_t(x_t^s - x_t^b)] \\ \text{s.t.} & l_t = l_{t-1} + \eta x_t^b - \eta^{-1} x_t^s, \quad t = 2, \dots, T \\ & s_t \leq C, \quad t = 1, \dots, T \\ & x_t^b, x_t^s, s_t \geq 0, \quad t = 1, \dots, T. \end{cases}$$

where x_t^s and x_t^b correspond to withdrawn and injected (sold and bought) quantities in stage t , respectively, and l_t denotes the storage level. Furthermore, we assume that the storage has a maximal storage capacity of C and an efficiency factor $\eta \in [0, 1]$. For the purpose of our computations, we use $C = 1$ and $\eta = 1$, if not stated otherwise.

We model the stochastic gas spot prices as a geometric Ornstein-Uhlenbeck process (see Schwartz 1997), with time-dependent mean reversion levels μ_t given by the following stochastic differential equation

$$\begin{aligned} dP_s &= \kappa(\mu_s - \log(P_s))P_s ds + \sigma P_s dW_s \\ \mu_s &= A \sin(2\pi s + \phi) \end{aligned}$$

where time is normalized in such a way that one year corresponds to a time difference of 1. Note that the model captures yearly seasonalities in the gas price by mean reversion to the trigonometric function μ_t where A is the amplitude and ϕ the phase of the function, while the periodicity is fixed to one year.

As is common in pricing, we work the risk-neutral measure by adding a constant market price of risk λ to the deterministic trend (see Schwartz 1997)

$$dP_s = \kappa(\mu_s - \log(P_s) - \lambda)P_s ds + \sigma P_s dW_s^*, \quad (17)$$

where dW_t^* is the increment of the Brownian motion under the equivalent martingale measure.

Solving (17) using Itô's Lemma, we find that P_s is log-normally distributed with

$$P_s = P_0 \exp \left[e^{-\kappa s} + \kappa \int_0^s e^{\kappa(t-s)} \mu_t^* dt + \sigma \int_0^s e^{\kappa(t-s)} dW_t^* \right]$$

where $\mu_t^* = \mu_t - \frac{\sigma^2}{2\kappa} - \lambda$. The arbitrage-free futures prices implied by the model are given by

$$\begin{aligned} \log(F_s) &= \mathbb{E}_0(P_s) = \mathbb{E}_0(\log(P_s)) + 2^{-1} \text{Var}_0(\log(P_s)) \\ &= \log(P_0)e^{-\kappa s} + \kappa \int_0^s e^{\kappa(x-s)} \mu_s^* dx + \frac{\sigma^2}{4\kappa} (1 - e^{-2s\kappa}) \\ &= \log(P_0)e^{-\kappa s} + \frac{\kappa A}{4\pi^2 + \kappa^2} (-2\pi (\cos(2\pi s + \phi) - e^{-\kappa s} \cos(\phi))) \end{aligned}$$

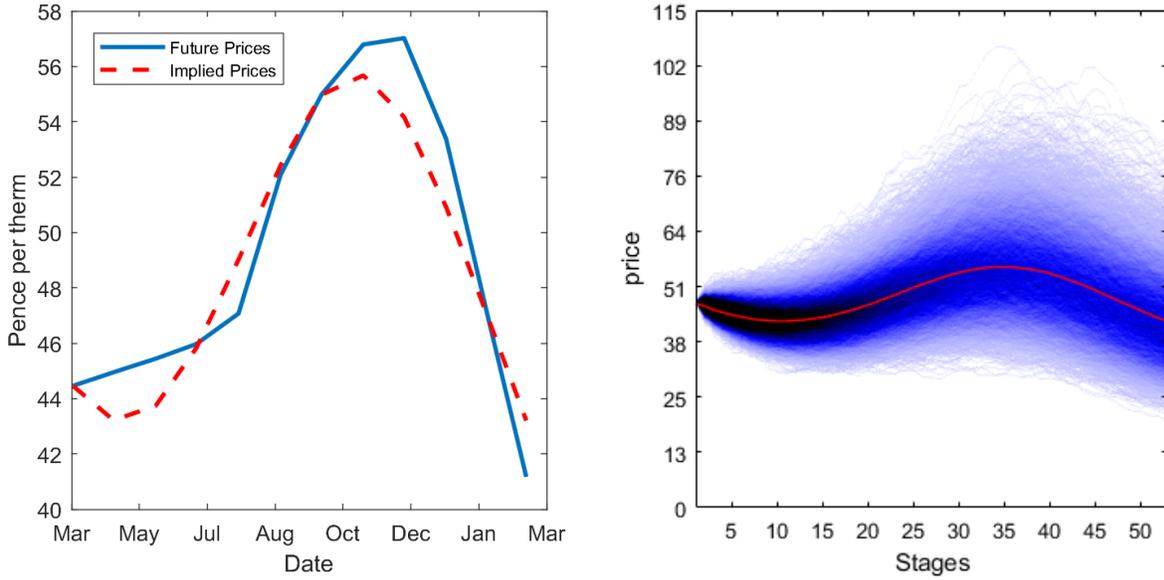


Figure 5 Fit of the price model (17) to the actual futures curve (left) and a plot of 10,000 paths from the calibrated process (right) in weekly resolution for one year with the red line representing the average price.

$$+ \kappa (\sin(2\pi s + \phi) - e^{-\kappa s} \sin(\phi)) - \left(\frac{\sigma^2}{\kappa} + \lambda \right) (1 - e^{-\kappa s}) + \frac{\sigma^2}{4\kappa} (1 - e^{-2\kappa s}), \quad (18)$$

where \mathbb{E}_0 and Var_0 are the expectation and the variance under the risk-neutral measure.

We calibrate the parameters of the model by fitting the implied futures prices (18) to an observed monthly future price curve for delivery at NBP in April 2018 to March 2019 from the 29th of March of 2018.¹ More specifically, we minimize the absolute difference between observed and implied prices with the function *multistart* from the MATLAB global optimization toolbox using the spot price on the 29th of March as P_0 .

The calibration results in parameter values of

$$(\kappa, A, \sigma, \lambda, \phi) = (0.189, 4.993, 0.199, -3.407, 5.203).$$

The fit of the model to the observed futures curve seems satisfactory for our purposes and is depicted in Figure 5 on the left and a plot of 10,000 simulations from the fitted process can be found in the right panel of the same figure.

¹ Data source: <https://www.ercr.energy/>

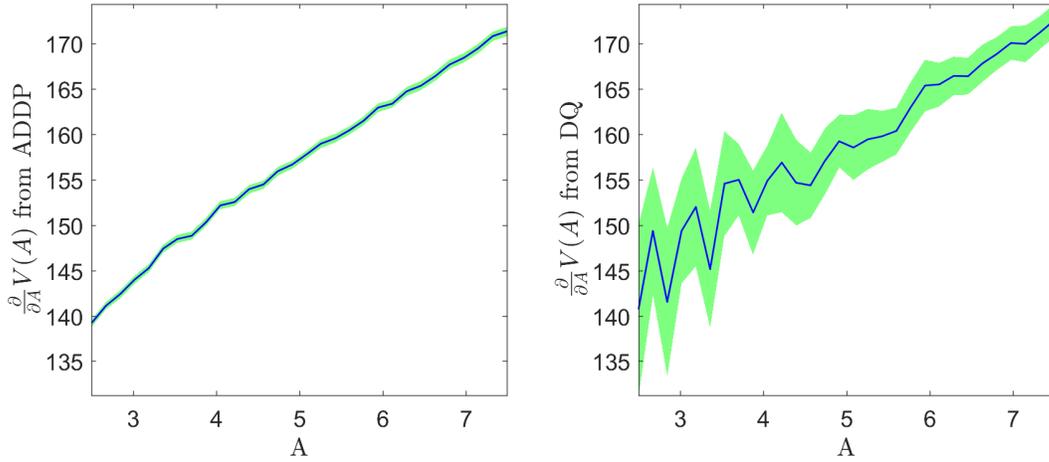


Figure 6 Comparing $\frac{\partial V}{\partial A}$ calculated via sampling (left) and via the difference quotient method (right).

For our computations, we discretize (17) to weekly time steps and assume a planning horizon from April 2018 to March 2019 (52 weeks) with the deterministic state being the 29.03.2018. This results in a rather large stochastic optimization problem with 53 stages.

As a first exercise, we consider the derivative of the optimal value with respect to the amplitude A of the seasonal variation, i.e., we calculate the sensitivity of the value of the gas storage with respect to the magnitude of the summer/winter spread in gas prices. To this end we calculate the derivative of the Lagrangian of the problem with respect to A arriving at

$$V'(A) = \mathbb{E} \left[\sum_{t=1}^T (x_t^{s*} - x_t^{b*}) \frac{\partial}{\partial A} P_t \right] \approx K^{-1} \sum_{k=1}^K \sum_{t=1}^T (x_{kt}^{s*} - x_{kt}^{b*}) \frac{\partial}{\partial A} P_t = \hat{V}'_K(A)$$

with x_{kt}^{b*} and x_{kt}^{s*} the optimal buy and sell decisions in scenario k and

$$\begin{aligned} \frac{\partial}{\partial A} P_t &= \kappa P_t \int_0^t e^{\kappa(s-t)} \sin(2\pi s + \phi) ds \\ &= \frac{\kappa P_t}{4\pi^2 + \kappa^2} \left(-2\pi (\cos(2\pi t + \phi) - e^{-\kappa t} \cos(\phi)) + \kappa (\sin(2\pi t + \phi) - e^{-\kappa t} \sin(\phi)) \right). \end{aligned}$$

Clearly, all the conditions for Theorems 3 to 5 are fulfilled in this example due to the smoothness of the involved functions and the distributional assumptions in (17).

In Figure 6, we compare $\hat{V}'_K(A)$ using $K=10,000$ samples with the estimates $\hat{V}'_{DQ}(A)$. Although, unlike in the newsvendor example, we do not have an analytic solution as reference, it seems

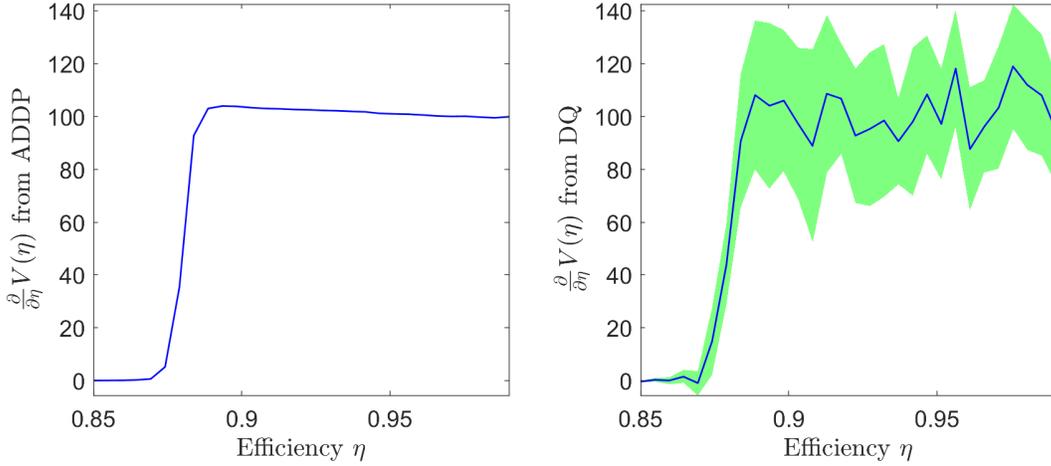


Figure 7 Comparing $\frac{\partial V}{\partial \eta}$ calculated via sampling (left) and via the difference quotient method (right).

obvious that $\hat{V}'_K(A)$ captures the true shape of $V'(A)$ much better than $\hat{V}'_{DQ}(A)$ as the latter violates monotonicity and shows strong local variability, which is unlikely to be a feature of the actual derivative. This is also reflected in the wide confidence bounds for the estimate based on the difference quotient, which are again based on 30 independent evaluations. In contrast, the confidence bounds for the sampled derivatives are rather narrow.

Finally, we calculate the derivative with respect to the flow efficiency factor η . Taking the derivative of the Lagrangian of the problem with respect to η yields

$$V'(\eta) = \mathbb{E} \left[\sum_{t=1}^T \lambda_t^* \left(x_t^{b*} + \frac{1}{\eta^2} x_t^{s*} \right) \right] \approx K^{-1} \sum_{k=1}^K \sum_{t=1}^T \lambda_{kt}^* \left(x_{kt}^{b*} + \frac{1}{\eta^2} x_{kt}^{s*} \right) = \hat{V}'_K(\eta)$$

where λ_t^* are the optimal dual solutions assigned to the storage balance equations. Note that the above derivative is the derivative with respect to a parameter in the matrix, i.e., the left-hand side of the constraints. As above, we use $K=10,000$ to estimate $\hat{V}'_K(\eta)$.

The results of the analysis are depicted in Figure 7 and point to the same conclusion as previous results: $\hat{V}'_{DQ}(\eta)$ is a much more error-prone estimator than $\hat{V}'_K(\eta)$. In particular, the zig-zag patterns in \hat{V}'_{DQ} and the large confidence bounds lead to the conclusion that $\hat{V}'_K(\eta)$ is clearly superior.

5. Conclusion

In this paper, we propose a method to calculate derivatives of value functions of MSLPs at points where these functions are differentiable. We base our results on classical envelope theorems and

use some novel findings on the smoothness of the optimal value of linear optimization problems to establish that the value functions in question are differentiable almost everywhere. We use the latter property of differentiability almost everywhere to establish that the derivatives computed with our method are valid with probability one.

The simple two-stage newsvendor example in Section 4 demonstrates that our method clearly outperforms a naïve computation of the derivative via the difference quotient in terms of average errors, the variance of the estimates, and the required computation time. These findings extend to the more elaborate gas storage optimization example in Section 4.2, which shows that the method works well on large problems with many stages.

Another interesting take-away of our numerical analysis is that the estimates improve when we use the real stochastic process, instead of the scenario lattice, to generate samples in Algorithm 1. In particular, we observe that the former estimates lead to significantly smaller errors on average and to more reliable confidence intervals.

This paper opens some avenues for further research. In particular, it would be interesting to use the results from Section 2 to study second derivatives as well as mixed derivatives of MSLPs based on existing second-order envelope theorems. This could prove useful in the context of financial risk management, which is often based on *Greeks* that represent second derivatives of option values. A prominent example is the gamma, which is the second derivative of the value of a contingent claim with respect to the value of the underlying.

Furthermore, it would be interesting to apply the methods developed in the paper to practical problems. In particular, pricing and hedging problems in finance, where there is no analytical solution for the value of certain contracts, might be a rewarding topic for further study.

References

- I. Adler and R. Monteiro. A geometric view of parametric linear programming. *Algorithmica*, 8:161–176, 12 1992.
- V. Bally and G. Pagès. A quantization algorithm for solving multidimensional discrete-time optimal stopping problems. *Bernoulli*, 9(6):1003–1049, 2003.

- S. Barnett. A simple class of parametric linear programming problems. *Operations Research*, 16(6):1160–1165, 1968.
- J.R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer Series in Operations Research and Financial Engineering. Springer New York, 2011.
- J. Bochnak, M. Coste, and M.F. Roy. *Real Algebraic Geometry*. A Series of Modern Surveys in Mathematics. Springer Berlin Heidelberg, 2013.
- M. Branda and J. Dupačová. Approximation and contamination bounds for probabilistic programs. *Annals of Operations Research*, 193(1):3–19, Mar 2012.
- M. Courtillot. New methods in mathematical programming—on varying all the parameters in a linear-programming problem and sequential solution of a linear-programming problem. *Operations Research*, 10(4):471–475, 1962.
- W. Dent, R. Jagannathan, and M. R. Rao. Parametric linear programming: Some special cases. *Naval Research Logistics Quarterly*, 20(4):725–728, 1973.
- J. Dupačová. The minimax approach to stochastic programming and an illustrative application. *Stochastics*, 20(1):73–88, 1987. ISSN 0090-9491.
- J. Dupačová. Stability and sensitivity-analysis for stochastic programming. *Annals of Operations Research*, 27(1):115–142, 1990.
- J. Elstrodt. *Maß- und Integrationstheorie*. Grundwissen Mathematik. Springer, 2002.
- R. Freund. Postoptimal analysis of a linear program under simultaneous changes in matrix coefficients. In R. W. Cottle, editor, *Mathematical Programming Essays in Honor of George B. Dantzig Part I*, pages 1–13. Springer, Berlin, Heidelberg, 1985.
- T. Gal and H.J. Greenberg. *Advances in sensitivity analysis and parametric programming*. International series in operations research & management science. Kluwer Academic Publishers, Boston, 1997.
- J.C. Hull. *Options, Futures, and Other Derivatives*. Pearson Education, 2014.
- P. Jäckel. *Monte Carlo Methods in Finance*. The Wiley Finance Series. Wiley, 2002.
- J. LaFrance and L.D. Barney. The envelope theorem in dynamic optimization. *Journal of Economic Dynamics and Control*, 15:355–385, 02 1991.

- N. Löhndorf, D. Wozabal, and S. Minner. Optimizing trading decisions for hydro storage systems using approximate dual dynamic programming. *Operation Research*, 61:810–823, 2013.
- F. Maggi. *Sets of Finite Perimeter and Geometric Variational Problems: An Introduction to Geometric Measure Theory*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2012.
- D.H. Martin. On the continuity of the maximum in parametric linear programming. *Journal of Optimization Theory and Applications - J OPTIMIZ THEOR APPL*, 17:205–210, 05 1975.
- P. Milgrom and I. Segal. Envelope theorems for arbitrary choice sets. *Econometrica*, 70(2):583–601, 2002.
- M.V.F. Pereira and L.M.V.G. Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52(2):359–375, 1991.
- G. Ch. Pflug and A. Pichler. *Multistage Stochastic Optimization*. Springer Series in Operations Research and Financial Engineering, 2014.
- A. B. Philpott and Z. Guan. On the convergence of stochastic dual dynamic programming and related methods. *Operations Research Letters*, 36(4):450 – 455, 2008.
- W.B. Powell. *Approximate dynamic programming. Solving the curses of dimensionality*. Wiley, 2011.
- P.A. Samuelson. *Foundations of economic analysis*. Harvard economic studies. Harvard University Press, 1947.
- E.S. Schwartz. The stochastic behavior of commodity prices: Implications for valuation and hedging. *The Journal of Finance*, 52(3):923–973, 1997.
- A. Shapiro. Analysis of stochastic dual dynamic programming method. *European Journal of Operational Research*, 209(1):63 – 72, 2011.
- A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory, Second Edition*. MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 2014.
- A. Takayama. *Mathematical Economics*. Cambridge University Press, 1985.
- R. M. Van Slyke and R. Wets. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663, 1969.
- H.R. Varian. *Microeconomic Analysis*. Norton International edition. Norton, 1992.

J.E. Ward and R.E. Wendell. Approaches to sensitivity analysis in linear programming. *Annals of Operations Research*, 27(1):3–38, Dec 1990.

L.B. Willner. On parametric linear programming. *SIAM Journal on Applied Mathematics*, 15(5):1253–1257, 1967.