ZAMM · Z. angew. Math. Mech. **81** (2001) , 1–15

Dupačová, J.; Sladký, K.

Comparison of Multistage Stochastic Programs with Recourse and Stochastic Dynamic Programs with Discrete Time

When solving a dynamic decision problem under uncertainty it is essential to choose or to build a suitable model taking into account the nature of the real-life problem, character of input data, availability of software and computer technology. The purpose of this paper is to discuss similarities and differences of two candidate approaches connected with discrete time decision processes and with uncertainties of probabilistic nature.

Keywords: Multistage stochastic programs with recourse, dynamic programming, Markov decision processes

MSC (1991): 90C15, 90C39, 90C40

1 Introduction and Motivation

Multistage stochastic programs and stochastic dynamic programming problems with discrete time parameter deal essentially with the same types of problems – the dynamic and stochastic decision processes. They were initiated approximately in the same period: In 1955–1965, the basic ideas, problem formulations and solution concepts were elaborated and, at the same time, the first applications were successfully solved. However, they did follow an independent development: They have been included in Mathematical Subject Classifications under different numbers and were considered mostly competitive. Recognition of similarities and complementary features has been rare. See the illustrative chart and comments in Section 2; we refer to recent monographs [7], [25], [31], [32], [39] for full quotations. Since the contemporary rapid increase of computer efficiency as well the recent development of tractable computational approaches enable efficient solutions of complex dynamic stochastic problems, there is a call for comparisons of these approaches and there are instance of their combinations [41].

In this article we shall discuss similarities and differences of multistage stochastic programs (MSP) with recourse and stochastic dynamic programs (SDP) with discrete time parameter and with a fixed finite horizon. The main distinction is in the decision concept, in different structures used in their formulation and, consequently, also in different solution methods. On the contrary to the multistage stochastic programs, most of the motivation for the research on dynamic programming models come from a class of operations research and engineering applications where it is the decision rule that is primarily of interest and the horizon is very long, e.g. inventory control [2]; hence the insistence on finding a rule that depends on the observed state and not on the information we may infer about the underlying stochastic phenomena. An appropriate definition of state is then the central point of dynamic programming formulations (see e.g. [4], [27] or [32] for basic concepts in dynamic programming) whereas in the context of multistage stochastic programs states usually do not appear.

Sections 2 and 3 present briefly the multistage stochastic programs and stochastic dynamic programs with discrete time, respectively. The emphasis lies in careful listening assumptions needed to formulate and solve these problems. Illustrative examples are given in Section 4 and the conclusions formulated in Section 5.

2 Multistage Stochastic Programs

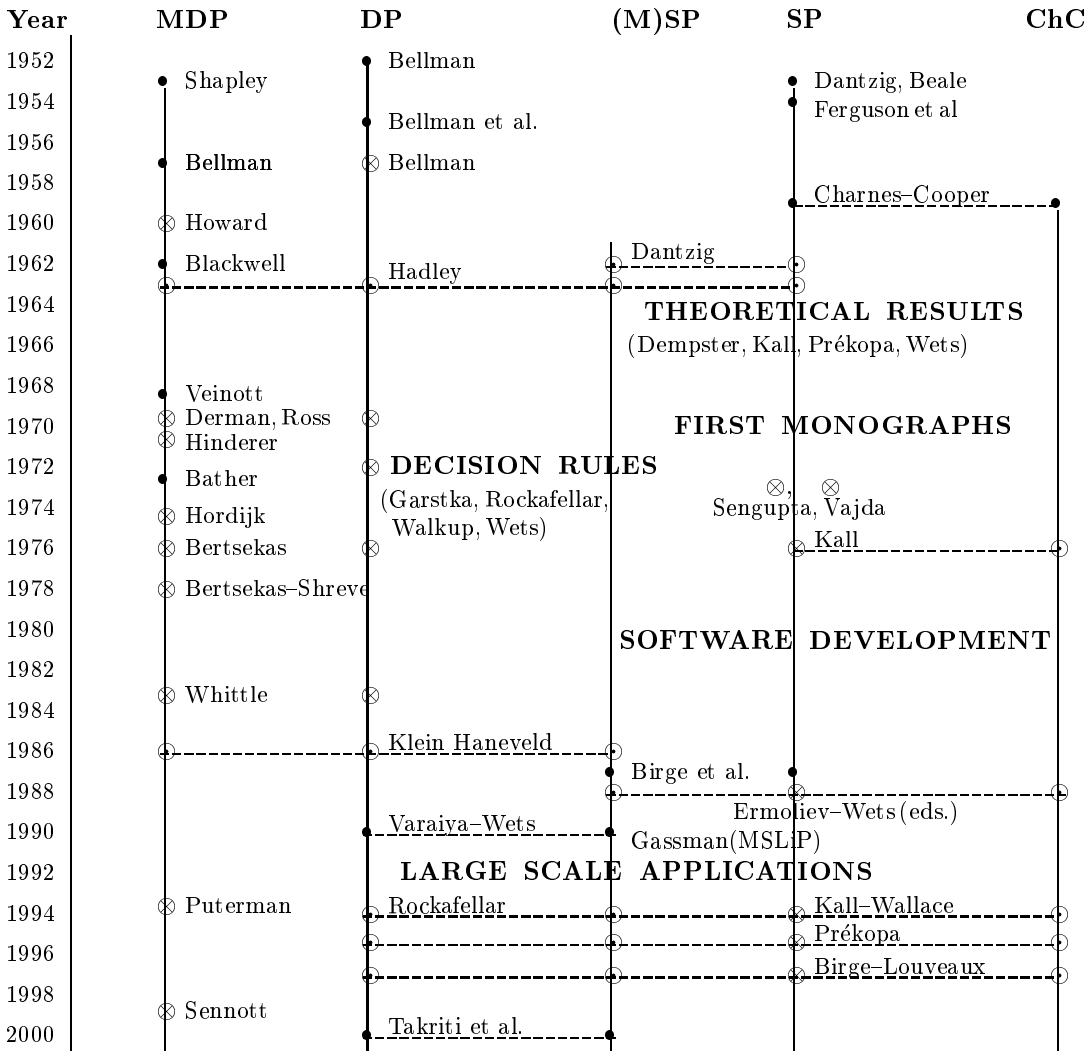
2.1 Basic formulations

In the general *T-stage stochastic program* we think of a stochastic data process $\omega = (\omega_1, \ldots, \omega_{T-1})$ or $\omega = (\omega_1, \ldots, \omega_T)$ and of a decision process $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)$. The $\boldsymbol{x}_t$'s are real n_t-vectors, while the random elements ω_t may be of quite general nature; mostly, they are real random vectors as well. The realizations of ω are called also trajectories or scenarios. We denote by P the probability distribution of ω and by Ω its support and we assume:

Basic assumption. The probability distribution P of ω is known and independent of the decision $\boldsymbol{x}$.

We refer to [42], [45] for a possible modeling of partial information, to [12] for a review on applications of stochastic programming under incomplete information and to [42] for a discussion of problems with probability distributions dependent on decisions.

HISTORY AND CONNECTIONS



• ... seminal paper ⊙ ... chapter in ⊗ selected monograph

MDP ... Markov Decision Processes **DP** ... Dynamic Programming

(M)SP ... (Multistage) Stochastic Programming **ChC** ... Chance-Constraints

The sequence of decisions and observations is

$$\boldsymbol{x}_1, \omega_1, \boldsymbol{x}_2, \omega_2, \ldots, \boldsymbol{x}_{T-1}, \omega_{T-1}, \boldsymbol{x}_T. \tag{1}$$

The decision process is *nonanticipative* in the sense that decisions taken at any stage of the process do not depend on future *realizations* of the data process or on future decisions whereas the past information as well as the the probabilistic specification $(\Omega, \mathcal{F}, P)$ of the process ω *are* exploited. This requirement can be mathematically expressed as follows: Denote $\mathcal{F}_{t-1} \subseteq \mathcal{F}$ the σ-field generated by the observations of the part $\omega^{t-1} := (\omega_1, \ldots, \omega_{t-1})$ of the stochastic data process that precedes stage t. The dependence of the t-th stage decision $\boldsymbol{x}_t$ only on these past observations means that $\boldsymbol{x}_t$ is $\mathcal{F}_{t-1}$-adapted or, in other words, that $\boldsymbol{x}_t$ is measurable with respect to $\mathcal{F}_{t-1}$. In each stage t, the decision is limited by explicit constraints that may depend on the previous decisions $\boldsymbol{x}^{t-1} := (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1})$ and on past observations of ω^{t-1}. Thus the decision at stage t is $\boldsymbol{x}_t = \boldsymbol{x}_t(\boldsymbol{x}^{t-1}, \omega^{t-1})$, or more precisely, $\boldsymbol{x}_t = \boldsymbol{x}_t(\boldsymbol{x}^{t-1}, \omega^{t-1}, P)$.

The outcome attributed to the sequence (1) is quantified by a function $f_0(\boldsymbol{x}, \omega)$. The aim is to minimize the expected value $Ef_0(\boldsymbol{x}, \omega)$ under both deterministic constraints $\boldsymbol{x}_t \in \mathcal{X}_t \forall t$ ($\mathcal{X}_t$ given sets in R^{n_t}), $f_{1i}(\boldsymbol{x}_1) \leq 0$, $i =$

$1, \dots, m_1$, and constraints

$$f_{ti}(\boldsymbol{x}^t, \omega^{t-1}) \le 0,\ i = 1, \dots, m_t,\ t = 2, \dots, T$$

that may depend on previous decisions and observations; here, $f_{ti}\ \forall t, i$ are real functions.

In the sequel we shall suppose that all functions are measurable with respect to ω and all expectations exist (this is certainly fulfilled if Ω is a finite set). Relations containing random elements are assumed to hold with probability 1. To simplify this exposition we shall assume in addition that all infima are attained; hence we shall write min instead of inf. This assumption implies that the sets defined by the t-stage constraints, $t = 1, \dots, T$,

$$\boldsymbol{x}_t \in \mathcal{X}_t : f_{ti}(\boldsymbol{x}^{t-1}, \boldsymbol{x}_t, \omega^{t-1}) \le 0,\ i = 1, \dots, m_t \tag{2}$$

are nonempty for all histories $\boldsymbol{x}^{t-1}, \omega^{t-1}$. The first-stage constraints do not depend on the random element.

The corresponding T-stage stochastic program reads:

$$\text{minimize} \quad Ef_0(\boldsymbol{x}_1, \boldsymbol{x}_2(\boldsymbol{x}_1, \omega_1), \dots, \boldsymbol{x}_T(\boldsymbol{x}^{T-1}, \omega^{T-1}), \omega) \tag{3}$$

$$\text{subject to} \quad \boldsymbol{x}_t \in \mathcal{X}_t,\ t = 1, \dots, T \ \text{ and}$$

$$f_{1i}(\boldsymbol{x}_1) \le 0, i = 1, \dots, m_1, \quad f_{ti}(\boldsymbol{x}^t, \omega^{t-1}) \le 0,\ i = 1, \dots, m_t,\ t = 2, \dots, T \tag{4}$$

Realizations of ω_T, i.e., those behind the horizon, do not affect the decision process, but they may contribute to the overall observed costs. Thus the decision process *may* be affected by the *probability distribution* of ω_T.

Various schemes were considered to reduce the T-stage stochastic program (3)–(4) to a sequence of similar t-stage programs, $t < T$. If ω_T is not considered, the objective functions are then defined recursively as

$$\psi_T(\boldsymbol{x}^T, \omega^{T-1}) \equiv f_0(\boldsymbol{x}, \omega),\ \psi_t(\boldsymbol{x}^t, \omega^{t-1}) = E_{\omega_t|\omega^{t-1}} \min_{\boldsymbol{x}_{t+1}} \psi_{t+1}(\boldsymbol{x}^{t+1}, \omega^t),\ t = 2, \dots, T-1 \tag{5}$$

and $\psi_1(\boldsymbol{x}_1) = E_{\omega_1} \min_{\boldsymbol{x}_2} \psi_2(\boldsymbol{x}_1, \boldsymbol{x}_2, \omega_1)$. The minimization is carried over the respective t-stage constraints (2) and the symbol $E_{\omega|\omega'}$ denotes the expectation with respect to ω conditioned by ω'.

To relate an optimal solution of the T-stage problem to those minimizing the t-stage objective functions ψ_t, certain boundedness assumptions concerning sets defined by the t-stage constraint and convexity of f_0 as a function of $\boldsymbol{x}$, have to be fulfilled; see [35]. Then not only the canonical projections of the optimal solution $\hat{\boldsymbol{x}}^T$ of the T-stage problem are optimal solutions of the t-stage problems, $t < T$, but also the optimal solutions of the t-stage problems can be extended to an optimal solution of the T-stage problem. For instance, if $\hat{\boldsymbol{x}}_1 \in \arg\min \psi_1(\boldsymbol{x}_1)$ over the first-stage constraints $\boldsymbol{x}_1 \in \mathcal{X}_1$ and $f_{1i}(\boldsymbol{x}_1) \le 0$, $i = 1, \dots, m_1$, then the next component of the optimal solution, $\hat{\boldsymbol{x}}_2(\hat{\boldsymbol{x}}_1, \omega_1)$ is obtained by solving $\min_{\boldsymbol{x}_2} \psi_2(\hat{\boldsymbol{x}}_1, \boldsymbol{x}_2, \omega_1)$ over the second-stage constraints, etc. By introducing a fictious decision $\boldsymbol{x}_{T+1}$ which does not influence the value of the objective function f_0 these results may be extended also to problems which include ω_T.

Under additional assumptions, e.g., for

$$f_0(\boldsymbol{x}, \omega) = f_{10}(\boldsymbol{x}_1) + \sum_{t=2}^{T} f_{t0}(\boldsymbol{x}^{t-1}, \boldsymbol{x}_t, \omega^{t-1}) \tag{6}$$

the scheme (5) can be written as a sequence of *nested two-stage stochastic programs* of the following type:

$$\text{minimize } Ef_0(\boldsymbol{x}, \omega) := f_{10}(\boldsymbol{x}_1) + E_{\omega_1}\phi_1(\boldsymbol{x}_1, \omega_1) \tag{7}$$

$$\text{subject to} \quad \boldsymbol{x}_1 \in \mathcal{X}_1 \ \text{ and } \ f_{1i}(\boldsymbol{x}_1) \le 0,\ i = 1, \dots, m_1,$$

where for $t = 2, \dots, T$, for given $\boldsymbol{x}_1, \dots, \boldsymbol{x}_{t-1}$ and observed realizations of $\omega_1, \dots, \omega_{t-1}$, $\phi_{t-1}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_{t-1}, \omega_1, \dots, \omega_{t-1})$ denotes the optimal value of the stochastic program

$$\text{minimize } f_{t0}(\boldsymbol{x}_t, \omega^{t-1}) + E_{\omega_t|\omega^{t-1}}\left\{\phi_t(\boldsymbol{x}^{t-1}, \boldsymbol{x}_t, \omega^{t-1}, \omega_t)\right\} \qquad \text{subject to (2).} \tag{8}$$

Here, $\phi_T \equiv 0$ or it is an explicitly given function of $\boldsymbol{x}_1, \dots, \boldsymbol{x}_T, \omega_1, \dots, \omega_T$ if the contribution of ω_T is considered. The two terms in the definition of functions ϕ_{t-1} may be interpreted as the costs attributed to the decision $\boldsymbol{x}_t$ at stage t augmented for the expected minimal future costs.

It is important to realize that the *stages do not necessarily refer to time periods*, they correspond to steps in the decision process. The main emphasis is on the first-stage decisions which consist of all decisions that have to be selected before a further information is revealed whereas the second-stage decisions are allowed to adapt to this information, etc. In some applications the importance of the best first-stage decisions is evident: The examples

are the decision about the capacity of a new water reservoir, about an initial contract or allocation of funds or the initial charge decision for the metal melting process.

The formulation (7)–(8) resembles the backward recursion common in *stochastic dynamic programming problems*, see Section 3. In spite of this formal similarity the form (7)–(8) does not enter the numerical procedures for solving such stochastic programs. As the model formulation in stochastic programming and the algorithmic solution are separated there exist a large variety of stochastic programming models of various mathematical properties and with problem specific numerical approaches.

The dynamic decision process is approximated by optimal solutions obtained by repeated solution of similar stochastic programs which are *rolled forward* in time, i.e., by solving the problem repeatedly starting always with the new state of the system attained by application of the obtained optimal first-stage decision and using updated and/or shifted data trajectories. To enable a future continuation of such forward decision process behind the horizon one has to treat the end-effect in an appropriate way: to add constraints for the last stage, e.g., on the water level in the reservoirs, to include the expected value of the achieved state of the system into the objective function, to penalize the value of the expected outstanding debt, or to extend the problem for an additional "steady" stage and related constraints, cf. [19].

Characterization of *decision rules* (or recourse decisions) for stochastic programs was studied for instance in [16], [44]. The well-known properties of solutions of deterministic linear programs imply that in the two-stage stochastic linear programs with random right-hand sides $\boldsymbol{b}_2(\omega_1)$ the decision rule $\boldsymbol{x}_2(\boldsymbol{x}_1, \omega_1)$ is continuous piecewise linear in $\boldsymbol{b}_2(\omega_1)$. This result can be extended also to random transition matrices $\boldsymbol{B}_2$ and bounds $\boldsymbol{u}_2$, $\boldsymbol{l}_2$, it holds true also when the only random coefficients are $\boldsymbol{c}_2(\omega_1)$ in the second-stage objective function. However, optimal piecewise linear decision rules need not exist even for three-stage stochastic linear programs with random, interstage independent right-hand sides; for an example see [44]. An exception are problems with discrete probability distributions. Under special assumptions which guarantee existence of a fixed optimal basis even an optimal linear decision rule can be obtained, cf. [18]. These results are further discussed and extended to nonlinear recourse problems in [16]. Existence of continuous decision rules can be proved for convex multistage stochastic programs under additional assumptions about the properties of the problem and for a special class of "laminary" probability distributions, cf. [34]; discrete probability distributions with an arbitrary dependence structure and probability distributions which fulfil assumption of interstage independence belong into this class.

A special case of (7)–(8) is the following multistage *stochastic linear program with recourse* where all functions f (with arbitrary indices) in the above scheme are linear in the decision variables:

$$\text{minimize} \quad \boldsymbol{c}_1^\top \boldsymbol{x}_1 + E_{\omega_1}\phi_1(\boldsymbol{x}_1, \omega_1) \tag{9}$$

subject to

$$\boldsymbol{A}_1\boldsymbol{x}_1 = \boldsymbol{b}_1, \quad \boldsymbol{l}_1 \le \boldsymbol{x}_1 \le \boldsymbol{u}_1,$$

where the functions $\phi_{t-1}, t = 2, \dots, T,$ are defined recursively as

$$\phi_{t-1}(\boldsymbol{x}^{t-1}, \omega^{t-1}) = \min_{\boldsymbol{x}_t} \left[\boldsymbol{c}_t(\omega^{t-1})^\top \boldsymbol{x}_t + E_{\omega_t|\omega^{t-1}}\, \phi_t(\boldsymbol{x}^{t-1}, \boldsymbol{x}_t, \omega^{t-1}, \omega_t)\right] \tag{10}$$

subject to

$$\sum_{\tau=1}^{t-1} \boldsymbol{B}_{t\tau}(\omega^{t-1})\, \boldsymbol{x}_\tau + \boldsymbol{A}_t(\omega^{t-1})\, \boldsymbol{x}_t = \boldsymbol{b}_t(\omega^{t-1}), \quad \boldsymbol{l}_t(\omega^{t-1}) \le \boldsymbol{x}_t \le \boldsymbol{u}_t(\omega^{t-1})$$

and $\phi_T \equiv 0$ or a given function of $\boldsymbol{x}$ and ω.

Here, the $\boldsymbol{A}_t$'s are (m_t, n_t) matrices and the remaining vectors and matrices are of consistent dimensions. For the first stage, the values of all elements in $\boldsymbol{b}_1, \boldsymbol{c}_1, \boldsymbol{A}_1, \boldsymbol{l}_1, \boldsymbol{u}_1$ are known. Again, the main decision variable is $\boldsymbol{x}_1$ that corresponds to the first stage.

According to our assumption, an optimal solution of (10) exists for all t and all considered histories $\boldsymbol{x}^{t-1}$, ω^{t-1} – the case of the *relatively complete recourse*.

For purposes of applications one approximates the true probability distribution P of ω by a discrete probability distribution concentrated on a finite number of atoms, called scenarios.

2.2 Scenario-based stochastic linear programs

Without loss of generality we shall work now with problems (9)–(10) in which $\boldsymbol{B}_{t\tau} \equiv 0$ for $\tau < t-1$ and we shall assume that the distribution of ω is concentrated on a finite number of scenarios. Accordingly, the supports Ω_t of marginal probability distributions of the components $\omega_t \, \forall t$ and the supports of conditional probability distributions

of ω_t conditioned by past realizations of $\omega_1, \dots \omega_{t-1}$ are finite sets. The associated conditional probabilities are called the *arc* probabilities. A special common arrangement of the data process is the *scenario tree* in which each value of ω^t corresponds to one node k_{t+1} at the stage $t+1$ and each node has a unique ancestor, $a(k_{t+1})$ – the value of the corresponding ω^{t-1} assigned to one of nodes at the stage t. The root of the tree, node indexed by 1, is ancestor of all nodes $k_2 = 2, \dots, K_2$ at the second stage, etc. In some cases, it is expedient to use sets $\mathcal{D}(k_t)$ of *descendants* of k_t which consist of those nodes k_{t+1} which can descend from k_t with a nonzero probability. To each k_t one assigns the t-th stage decision vector $\boldsymbol{x}_{k_t}$. This allows to rewrite the *T-stage scenario-based stochastic linear program with additive recourse* in the following *arborescent* form:

Minimize

$$\boldsymbol{c}_1^\top \boldsymbol{x}_1 + \sum_{k_2=2}^{K_2} p_{k_2} \boldsymbol{c}_{k_2}^\top \boldsymbol{x}_{k_2} + \sum_{k_3=K_2+1}^{K_3} p_{k_3} \boldsymbol{c}_{k_3}^\top \boldsymbol{x}_{k_3} + \ldots + \sum_{k_T=K_{T-1}+1}^{K_T} p_{k_T} \boldsymbol{c}_{k_T}^\top \boldsymbol{x}_{k_T} \tag{11}$$

subject to

$$\begin{array}{lllll}
\boldsymbol{A}_1 \boldsymbol{x}_1 & & & = \boldsymbol{b}_1 & \\
\boldsymbol{B}_{k_2} \boldsymbol{x}_1 + \boldsymbol{A}_{k_2} \boldsymbol{x}_{k_2} & & & = \boldsymbol{b}_{k_2}, & k_2 = 2, \dots, K_2 \\
\boldsymbol{B}_{k_3} \boldsymbol{x}_{a(k_3)} + \boldsymbol{A}_{k_3} \boldsymbol{x}_{k_3} & & & = \boldsymbol{b}_{k_3}, & k_3 = K_2+1, \dots, K_3 \\
\ddots \quad \ddots & & & \vdots & \\
\boldsymbol{B}_{k_T} \boldsymbol{x}_{a(k_T)} + \boldsymbol{A}_{k_T} \boldsymbol{x}_{k_T} & & & = \boldsymbol{b}_{k_T}, & k_T = K_{T-1}+1, \dots, K_T
\end{array}$$

$$\boldsymbol{l}_{k_t} \le \boldsymbol{x}_{k_t} \le \boldsymbol{u}_{k_t}, \qquad k_t = K_{t-1}+1, \dots, K_t, \quad t = 1, \dots, T. \tag{12}$$

The problem is based on the used scenarios, i.e., on $K_T - K_{T-1}$ sequences of possible realizations of coefficients $(\boldsymbol{c}_{k_t}, \boldsymbol{A}_{k_t}, \boldsymbol{B}_{k_t}, \boldsymbol{b}_{k_t}, \boldsymbol{l}_{k_t}, \boldsymbol{u}_{k_t})$ in the objective function (11), in recourse matrices, transition matrices and right-hand sides in the constraints for all stages, and on the *path* probabilities $p_{k_t} > 0 \,\forall\, k_t$, $\sum_{k_t=K_{t-1}+1}^{K_t} p_{k_t} = 1$, $t = 2, \dots, T$, of partial sequences of these coefficients, hence, probabilities of realizations of $\omega^{t-1} \,\forall t$.

Probabilities p^σ of scenarios ω^σ of ω, i.e., the path probabilities assigned to the terminal nodes, are obtained by multiplication of the (conditional) *arc* or *transition probabilities* related with the corresponding sequences of realizations. Nonanticipativity constraints are included here in an implicit form whereas decomposition of (11)–(12) along scenarios requires that the nonanticipativity constraints are spelled out in an explicit way.

The size of the linear program (11)–(12) can be very large and usefulness of special numerical techniques is obvious. Its special structure suggests exploitation of decomposition techniques, both with respect to stages (cf. [7], [17] for adaptation of the Benders decomposition algorithm or [31] for basis decomposition methods) and along scenarios. In the last case, the nonanticipativity constraints are spelled out explicitly and are subsequently relaxed within an augmented Lagrangian type method. This idea can be applied also to nonlinear multistage stochastic programs, see e.g. [36] for the progressive hedging algorithm or [29] for the diagonal quadratic approximation method. A further possibility is to use interior-point-based methods along with their parallel implementation, for instance [10], [38]. The stochastic extension of the MPS input format [6] allows unified description, storing of multistage stochastic problems and data. Nevertheless, exploiting the properties and the structure of the solved problems is the key to effective algorithms. An example are multistage stochastic programs with network structure, cf. [30] who report numerical results from an application from the insurance industry with 9 stages, with up to 16392 scenarios, where the arborescent form will have a half million constraints and 1.3 million variables.

As a consequence of these achievements, it is possible to solve numerically large scale linearly constrained convex stochastic programs based on scenarios, e.g., (11)–(12) with the linear objective function (11) replaced by

$$f_1(\boldsymbol{x}_1) + \sum_{k_2=2}^{K_2} p_{k_2} f_{k_2}(\boldsymbol{x}_{k_2}) + \sum_{k_3=K_2+1}^{K_3} p_{k_3} f_{k_3}(\boldsymbol{x}_{k_3}) + \ldots + \sum_{k_T=K_{T-1}+1}^{K_T} p_{k_T} f_{k_T}(\boldsymbol{x}_{k_T})$$

where f_k, $k = 1, \dots, K_T$ are convex functions (compare with (6)), and with (12) replaced by constraints displaying a full or partial path dependence.

Besides the formulation of goals and constraints and identification of the driving random process ω, building a scenario-based multistage stochastic program requires specification of the horizon, stages and generation of the input in the form of scenario tree. We refer to [15] for methods of generating scenario trees and to [14] for discussion related with the choice of the horizon and stages. For purposes of the present paper, we mostly assume that the horizon is finite and both the horizon and stages have been already fixed. We note only that the rapid increase of the size of the solved stochastic programs along with an increasing number of stages calls for a compromise as to

modeling of dynamic features (e.g., the number of stages) and the representation of the underlying random process (e.g., the number of scenarios).

In real-life applications, it seems to be the modeling part of the problem and a meaningful generation of scenarios which have become the most demanding task. Whereas the interstage independence, i.e., mutual independence of $\omega_1, \ldots, \omega_T$ is neither essential for formulation of scenario-based stochastic program (11)–(12) nor for its solution by linear programming techniques, it plays an important role whenever bounds and stopping rules are to be constructed in course of iterative or sample based procedures. This means that such algorithms known for two-stage stochastic programs with fixed complete recourse, e.g. [21], [28], can be generalized to algorithms for multistage stochastic programs with scenario independent recourse matrices $\boldsymbol{A}_t$ and mutually independent $\omega_1, \ldots, \omega_T$, see e.g. [24]. Possibility of their generalization to problems with dependent components of ω is limited to problems with a special stochastic specification (e.g., multinormal right-hand sides).

3 Multistage Stochastic Programs and Dynamic Programming

3.1 Basic connections

To show the connections between multistage stochastic programs and stochastic dynamic programs, let us consider again the sequence (1) assuming that the next stage of the considered process is entirely determined by the state, decision and random data occurring in the current stage. In particular, we make the following assumptions concerning the dynamics of the system:

Assumption 1. For every stage $t = 1, \ldots, T$

$$\boldsymbol{x}_t = (\boldsymbol{s}_t, d_t) \quad \text{with} \quad \boldsymbol{s}_{t+1} = F_t(\boldsymbol{s}_t, d_t, \omega_t) \quad \text{and} \quad \boldsymbol{x}_1 = (\boldsymbol{s}_1, d_1) \ \text{ given}, \tag{13}$$

where $\boldsymbol{s}_t \in S_t,\ d_t \in D_t,\ \omega_t \in \Omega_t$ and $F_t(\cdot,\cdot,\cdot)$ is a mapping from $S_t \times D_t \times \Omega_t$ onto S_{t+1}.

The variable $\boldsymbol{s}_t$ (resp. d_t) is called the *state* (resp. *decision*) at stage t. Similarly, S_t (resp. D_t) is the *state space* (resp. *decision space*) at stage t. Let $d^t := (d_1, d_2, \ldots, d_t)$ be a sequence of decisions (called also *policy*) controlling the considered process. We restrict ourselves on Markovian (memoryless) policies, i.e. we assume that $d_t = G_t(\boldsymbol{s}_t)$, $\forall t$.

A further simplification can be obtained by assuming mutual independence of ω_t's, called in multistage stochastic programs the *interstage independence assumption*.

Assumption 2. The random variables $\omega_1, \ldots, \omega_t, \ldots, \omega_T$ are mutually independent.

For the sake of computational tractability we restrict the class of objective functions as follows:

Assumption 3. The objective function (3) is separable with respect to the stage index t. In particular, we assume that $f_0(\boldsymbol{x}, \omega)$ is additive, i.e.

$$f_0(\boldsymbol{x}_1, \boldsymbol{x}_2(\boldsymbol{x}_1, \omega_1), \ldots, \boldsymbol{x}_T(\boldsymbol{x}^{T-1}, \omega^{T-1}), \omega) = f_{10}(\boldsymbol{s}_1, d_1) + \sum_{t=1}^{T} \bar{f}_t(\boldsymbol{s}_t, d_t, \omega_t) \tag{14}$$

The "Markovian" property for generating the sequences of states by (13) of Assumption 1 along with mutual independence of ω_t (interstage independence by Assumption 2) yields the following important implication:

$$\text{Prob}\,[\boldsymbol{s}_{t+1} = \boldsymbol{s} | \boldsymbol{s}_1, \ldots, \boldsymbol{s}_t; d_1, \ldots, d_t] = \text{Prob}\,[\boldsymbol{s}_{t+1} = \boldsymbol{s} | \boldsymbol{s}_t,\ d_t] := p_t(\boldsymbol{s}_t, \boldsymbol{s}; d_t) \tag{15}$$

and the form of objective function (3) follows from Assumption 3:

$$E_\omega f_0(\boldsymbol{x}_1, \boldsymbol{x}_2(\boldsymbol{x}_1, \omega_1), \ldots, \boldsymbol{x}_T(\boldsymbol{x}^{T-1}, \omega^{T-1}), \omega) = c_1(\boldsymbol{s}_1; d_1) + \sum_{t=2}^{T} E_\omega c_t(\boldsymbol{s}_t; d_t) \tag{16}$$

where $c_1(\boldsymbol{s}_1; d_1) := f_{10}(\boldsymbol{s}_1, d_1) + E_{\omega_1} \bar{f}_1(\boldsymbol{s}_1, d_1, \omega_1)$ and $c_t(\boldsymbol{s}_t; d_t) := E_{\omega_t} \bar{f}_t(\boldsymbol{s}_t, d_t, \omega_t)$ for $t = 2, \ldots, T$.

The following technical assumption leads to an essential simplification for notations, analysis and solution of stochastic dynamic programming problems.

Assumption 4. For $t = 1, \dots, T$, the sets $S_t \equiv S$ and the sets $D_t \equiv D$ are finite. Moreover, the full information is available about the state $\boldsymbol{s}_t$ at any stage t and all expectations exist.

The collection of transition probabilities $p_t(\boldsymbol{s}, \boldsymbol{s}'; d)$ along with the initial (possibly degenerated) distribution and the decision rule $d^T = (d_1(\boldsymbol{s}_1), \dots, d_T(\boldsymbol{s}_T))$ fully describes the "probabilistic" development of the considered process without any reference to the transition function F in (13). Since the objective function is assumed to be separable, under Assumptions $1, 2, 3, 4$ the considered multistage stochastic control problem can be rewritten in the form referred often as a classical *Discrete Time Markov Decision Process* or *Discrete Time Stochastic Dynamic Program*:

Given the probabilities $p_t(\boldsymbol{s}, \boldsymbol{s}'; d)$ (where $\boldsymbol{s}, \boldsymbol{s}' \in S$ and $d \in D$) find a sequence of decisions $d^T = (d_1(\boldsymbol{s}_1), \dots, d_T(\boldsymbol{s}_T))$ (where $d_t \in D$) that minimizes $E_\omega \sum_{t=1}^T c_t(\boldsymbol{s}_t; d_t)$.

Under the above assumptions the real valued functions $V_t(\boldsymbol{s}) := \min_{d^T} E_\omega \left[\sum_{\tau=t}^T c_\tau(\boldsymbol{s}_\tau; d_\tau) | \boldsymbol{s}_t = \boldsymbol{s} \right]$ can be calculated for $t = 1, \dots, T-1$ by the following "backward recursion"

$$V_t(\boldsymbol{s}) = \min_{d \in D} \{ c_t(\boldsymbol{s}; d) + \sum_{\boldsymbol{s}' \in S} p_t(\boldsymbol{s}, \boldsymbol{s}'; d) \cdot V_{t+1}(\boldsymbol{s}') \} \tag{17}$$

with $V_T(\boldsymbol{s}_T) = 0$. (See e.g. [3], [4], [22]). Hence, for a given initial state $\boldsymbol{s}_1$ of the system, $V_1(\boldsymbol{s}_1)$ is the optimal value of the objective function $E_\omega \sum_{t=1}^T c_t(\boldsymbol{s}_t; d_t)$.

The analysis can be easily extended for compact decision space D. In case that the state space S is countably infinite, the analysis used for finite state models can be extended under some additional assumptions. Unfortunately, for the general state space an exact analysis of the above discrete time stochastic dynamic programs is complicated; for details see e.g. [5] or [22] for an exact measure theoretic analysis of models with general state space.

Solution of the optimization problems (17) can be hard or easy depending on the application. However, implementation of the "backward recursion" requires storing the values of the functions $V_t(\cdot)$ and of the optimal decisions for all states of the system; therefore for many applications the number of states is prohibitively large. This "curse of dimensionality" in dynamic programming is partly due to the dimension of the state space and makes storing values and decisions for all states impossible. To limit the number of states, one may aggregate the states, decompose the dynamic program into smaller related dynamic programs, etc. See [27] for references. Another possibility is to solve an approximate problem using properties of Markov Decision Chains which results in a numerically tractable method; see the next subsection.

3.2 Discrete Time Stochastic Dynamic Programs

Let us consider at stages $t = 1, 2, \dots$ a nonhomogeneous Markov chain with finite state space $\mathcal{S} = \{1, \dots, N\}$. If the process is found to be in state $i \in \mathcal{S}$ then a decision $k = 1, 2, \dots, K$ must be selected. Selecting decision k in state i state j is reached in the next stage with (stage independent) probability $p(i, j; k)$ and an immediate cost $c(i, j; k)$ (not depending on the current stage) is incurred; $c(i; k) = \sum_{j=1}^N p(i, j; k)\, c(i, j; k)$ is the expected cost incurred in state $i \in \mathcal{S}$ if decision k is selected. Such decision process is called a *discrete time Markov decision process*. The data $p(i, j; k)$, $c(i; k)$ are assumed to be known to the decision maker.

Policy Π controlling the process is a rule prescribing the decision to be taken after each transition in any state of the chain. We restrict on Markov (memoryless) policies, i.e. decision rules taking into account only the number of transitions t and the current state $s_t \in \mathcal{S}$ of the chain. A policy which takes at all times the same decision rule is called stationary. We write $\Pi \equiv (\pi^1, \pi^2, \dots)$ where π^t, the decision rule at stage t, is an N-vector whose i-th component is the decision at the t-th transition when the chain is in state i. Stationary policy Π is identified by $\Pi \sim (\pi)$.

Let $P(\pi^t)$ be the $N \times N$ matrix whose ij-th element equals $p(i, j; \pi_i^t)$ and let $P^T(\Pi) = P(\pi^1) \dots P(\pi^T)$ (for convenience we set $P^0(\Pi) = I$, the identity matrix). Similarly, $c(\pi^t)$ denotes the $N \times 1$ vector whose i-th element equals $c(i; \pi_i^t)$. If $\Pi \sim (\pi)$ (i.e. if Π is stationary) then $P^T(\Pi) = (P(\pi))^T$.

In what follows we denote by $v^T(\Pi)$ the vector of total expected costs (i.e. the sum of expected costs) incurred in the T next transitions; its i-th element $v_i^T(\Pi)$, $i = 1, \dots, N$, equals the total expected costs incurred provided the chain starts in state $i \in \mathcal{S}$ and policy Π is followed. Obviously

$$v^T(\Pi) = \sum_{t=1}^T P^{t-1}(\Pi)\, c(\pi^t) \quad \text{with} \quad \lim_{T \to \infty} \frac{1}{T} v^T(\Pi) = g(\Pi) \tag{18}$$

provided the limit exists (the i-th element of $g(\Pi)$, denoted $g_i(\Pi)$, is the long run average expected costs if the Markov chain starts in state i). Observe that for a stationary policy $\Pi \sim (\pi)$ we get $v^T(\Pi) = \sum_{t=1}^{T} (P(\pi))^{t-1} c(\pi)$.

In contrast to multistage stochastic programs, there are no principle difficulties in the study of stochastic dynamic programs with long or even infinite time horizon. To study such models we recall some facts from Markov chain theory (cf. e.g. [26]). It is well known that the following matrices exist:

$P^* = \lim_{n\to\infty} \frac{1}{n} \sum_{k=0}^{n-1} P^k$ (the Cesaro steady state limit), $\quad Z = (I - P + P^*)^{-1}$ (the fundamental matrix of P).

If P has one recurrent class then the rows of P^* are identical and equal to the stationary probability distribution (row) vector $p^* = [p_1^*, \ldots, p_N^*]$ (i.e. the row of P^*) of the considered discrete-time Markov chain. Moreover, if P is also aperiodic (i.e. if one is the only eigenvalue of P with the modulus equal to one) then $\lim_{n\to\infty} P^n = P^*$.

In what follows we make the following simplifying assumption.

Assumption 5. $P(\pi)$ has a single class of recurrent states for any π.

In virtue of the above facts for T tending to infinity we have for a stationary policy $\Pi \sim (\pi)$

$$g(\Pi) = g(\pi) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} (P(\pi))^{t-1} c(\pi) = P^*(\pi) c(\pi). \tag{19}$$

where the rows of $P^*(\pi)$ are identical, and so are also the elements of $g(\pi)$.

Policy $\hat{\Pi}$ is called average optimal if

$$\limsup_{T\to\infty} \frac{1}{T} v^T(\hat{\Pi}) \le \limsup_{T\to\infty} \frac{1}{T} v^T(\Pi) \qquad \text{for every policy } \Pi. \tag{20}$$

The following facts are well-known to the researchers in dynamic programming:

(i) For every π there exists an $N \times 1$ vector $w(\pi)$ (unique up to additive constant) such that

$$g(\pi) + w(\pi) = c(\pi) + P(\pi)\, w(\pi). \tag{21}$$

Moreover, under condition $P^*(\pi)\, w(\pi) = 0$, the constant vector $g(\pi)$ (given by (19)) along with $w(\pi) = Z(\pi)\,(I - P^*(\pi))\, c(\pi)$ are the unique vectors fulfilling (21).

(ii) There exists $\hat{\pi}$, unique vector $\hat{g} = g(\hat{\pi})$ and vector $\hat{w} = w(\hat{\pi})$ (unique up to an additive constant) such that for every decision vector π

$$\hat{g} + \hat{w} = c(\hat{\pi}) + P(\hat{\pi})\, \hat{w} \le c(\pi) + P(\pi)\, \hat{w}. \tag{22}$$

Moreover, under condition $P^*(\hat{\pi})\, \hat{w} = 0$, $\hat{w}$ is the unique solution fulfilling (22).

(iii) For an arbitrary policy $\Pi = (\pi^t)$ it holds

$$v^T(\Pi) = T\hat{g} + \hat{w} - P^T(\Pi)\hat{w} + \sum_{t=1}^{T} P^{t-1}(\Pi)\, \varphi(\pi^t) \quad \text{where} \quad \varphi(\pi) = c(\pi) - \hat{g} + P(\pi)\hat{w} - \hat{w}. \tag{23}$$

Hence stationary policy $\hat{\Pi} \sim (\hat{\pi})$ fulfilling condition (22) must be average optimal policy, and a (nonstationary) policy $\Pi = (\pi^t)$ is average optimal if and only if $\lim_{T\to\infty} T^{-1} \sum_{t=1}^{T} P^{t-1}(\Pi)\, \varphi(\pi^t) = 0$.

Up to now we have mostly assumed that the time horizon is *rolling forward*. The only exception is the "backward dynamic programming recursion" (17) mentioned in Section 3. In what follows we show that results analogous to (23) can be also obtained in case of a *rolling backward* horizon. To this end, fix the number of stages T and rewrite (18) for $t = 1, \ldots, T$ as

$$V_{T-t}^{(T)}(\Pi) = \sum_{m=t}^{T} P^{m-1}(\Pi)\, c(\pi^m) = c(\pi^t) + P(\pi^t) V_{T-t-1}^{(T)}(\Pi) \quad \text{where} \quad v_0^{(T)}(\Pi) = 0. \tag{24}$$

Observe that the i-th element of $V_{T-t}^{(T)}(\Pi)$, denoted $V_{T-t}^{(T)}(\Pi)(i)$, is the total expected cost obtained at stages $t, t+1, \ldots, T$ on condition that the process was in state i at stage t and policy Π is followed.

Denoting $n = T - t$, from (24) we can conclude that

$$V_n^{(T)}(\Pi) = c(\pi^{T-n}) + P(\pi^{T-n})V_{n-1}^{(T)}(\Pi) \leq \min_{\pi} \left[c(\pi) + P(\pi)V_{n-1}^{(T)}(\Pi)\right] \tag{25}$$

(observe that since in each state the decision can be selected independently, the above (componentwise) vectorial minimum exists; π is the decision vector whose i-th element is the decision selected in state i.)

In virtue of these facts we can write the "backward recursion" of dynamic programming for finding minimum values of $v^t(\Pi)$ over all policies $\Pi = (\pi^t)$, denoted by $\hat{V}_t$, as follows

$$\hat{V}_{n+1} = \min_{\pi} \left[c(\pi) + P(\pi)\hat{V}_n\right] = c(\hat{\pi}_n) + P(\hat{\pi}_n)\hat{V}_n \tag{26}$$

where $\hat{V}_0$ (not necessarily equal to zero) is given.

Obviously, in virtue of (23) $n^{-1}\hat{V}_n$ must converge to the constant vector $\hat{g}$ of average minimal costs, i.e.

$$\lim_{n\to\infty} n^{-1}\hat{V}_n = \hat{g} \quad \text{and} \quad |\hat{V}_n - n\hat{g}| \text{ is bounded.}$$

Unfortunately, policies calculated from (26) can be nonstationary and quite irregular even for large n, but it can be shown (cf. [1]) that under the assumption that all $P(\pi)$ are aperiodic

$$\lim_{n\to\infty} (\hat{V}_n - n\hat{g}) = \hat{w} + const. \tag{27}$$

where *const.* ($N \times 1$ vector with equal elements) depends on "initial condition" $\hat{V}_0$.

¿From (27) follows immediately that $\lim_{n\to\infty}(\hat{V}_{n+1} - \hat{V}_n) = \hat{g}$ and it can be shown that ($\hat{V}_{n+1}(i)$ is the i-th element of $\hat{V}_{n+1}$)

$$\max_i \left[\hat{V}_{n+1}(i) - \hat{V}_n(i)\right] = \overline{b}_n, \qquad \min_i \left[\hat{V}_{n+1}(i) - \hat{V}_n(i)\right] = \underline{b}_n, \tag{28}$$

are the upper and lower bounds on $\hat{g}$ that converge monotoneously to the values of $\hat{g}$ and the convergence is geometric. From this we can conclude that for any naturals $m > n$

$$(m-n)\underline{b}^n\, e \leq \hat{V}_m - \hat{V}_n \leq (m-n)\overline{b}^n\, e \tag{29}$$

(e is reserved for a unit column vector of appropriate dimension).

This simple fact (called the "turnpike planning theorem", cf. [40]) can be extremely useful for calculating the values of the maximum total expected return for a large horizon T.

4 Illustrative Examples

In this section, we shall illustrate by examples some of the items discussed Sections 2 and 3.

4.1 The flower-girl problem

The flower-girl problem introduced in [9] and studied also in [12], [14] is a simple multistage stochastic program. The flower-girl sells roses at price c and has to buy them at cost p before she starts selling. Flowers left over at the end of the day can be stored and sold the next day, when she starts selling the old roses. The roses cannot be carried over more than one additional day at the end of which they are thrown away. The demand is random, ω_t denotes the demand on the t-th day. Whereas x_1 has to be bought without any knowledge of the realization of the random demand, the flower-girl can adapt the subsequent orders x_t, $t > 1$ to the demand observed during the previous days. Her goal is to maximize the total expected profit.

The horizon is related to the number of days for which the flower-girl continues selling roses without any break (and also to the fact that our formulation treats only one-period carry over). Assume first, that the flower-girl sells roses only during the weekend, orders the amount x_1 on Friday evening, observes the demand ω_1 on Saturday, stores the unsold roses (without any additional cost) and, possibly, buys $x_2(\omega_1)$ new roses. The demand ω_2 on Sunday determines the amount of unsold roses to be thrown away. Denote $s_2(\omega_1)$ the stock left for the subsequent day and $z(\omega_1, \omega_2)$ the amount of unsold roses at the end of the second day.

All decision variables are nonnegative and subject to constraints

$$x_1 - s_2(\omega_1) \leq \omega_1, \qquad x_2(\omega_1) + s_2(\omega_1) - z(\omega_1, \omega_2) \leq \omega_2.$$

If the demand ω_1, ω_2 is known in advance, the objective function is $(c-p)(x_1 + x_2(\omega_1)) - cz(\omega_1, \omega_2)$ and one of optimal solutions is to buy $x_1 = \omega_1$ and $x_2 = \omega_2$ roses which gives the maximal profit of $(c-p)(\omega_1 + \omega_2)$.

Consider now a scenario-based version of this 3-stage problem. The scenario tree consists of K branches corresponding to the considered realizations $\omega_{1k}, k = 1, \ldots, K$, of the demand ω_1 on the first day, their probabilities are p_k, $k = 1, \ldots, K$. Possible realizations of demand $\omega_{2k\sigma}$ for the second day may be conditional on ω_{1k}. We denote $\mathcal{D}(\omega_{1k})$ the set of descendants of ω_{1k}, and $p(k, \sigma)$ their (conditional) probabilities. The problem is

$$\text{maximize } (c-p)x_1 + \sum_{k=1}^{K} p_k[(c-p)x_2(\omega_{1k}) - c \sum_{\sigma \in \mathcal{D}(\omega_{1k})} p(k,\sigma) z(\omega_k, \omega_{2k\sigma})]$$

subject to

$$x_1 - s_2(\omega_{1k}) \le \omega_{1k}, \ k = 1, \ldots, K$$

$$x_2(\omega_{1k}) + s_2(\omega_{1k}) - z(\omega_1, \omega_{2k\sigma}) \le \omega_{2k\sigma}, \ \sigma \in \mathcal{D}(\omega_{1k}), \ k = 1, \ldots, K$$

and nonnegativity constraints. The total number of scenarios $(\omega_{1k}, \omega_{2k\sigma})$ equals the number of all descendants of ω_{1k}, $k = 1, \ldots, K$.

The generalization to T-stage problem is obvious; we index by t all decision variables related with the stage t, i.e., the amount of roses ordered (x), stored (s) and thrown away (z) at the end of the $(t-1)$st day. We obtain:

$$\text{maximize } (c-p)x_1 + E\{(c-p) \sum_{t=2}^{T-1} x_t(\omega^{t-1})) - c \sum_{t=2}^{T} z_t(\omega^{t-1})\}$$

subject to

$$x_1 + s_1 - s_2(\omega_1) - z_2(\omega_1) \le \omega_1$$

$$x_t(\omega^{t-1}) + s_t(\omega^{t-1}) - s_{t+1}(\omega^t) - z_{t+1}(\omega^t) \le \omega_t, \ t = 2, \ldots, T-1$$

$$s_t(\omega^{t-1}) - z_{t+1}(\omega^t) \le \omega_t, \ t = 1, \ldots, T-1$$

with $s_T(\omega) \equiv 0$ and nonnegativity of all variables. In case that the initial supply $s_1 = 0$, one gets $z_2(\omega_1) \equiv 0$. The main decision variables (controls) are x_t for $t = 1, \ldots, T-1$, the state of the system is described by s_t for $t = 1, ..., T-1$ and by z_T (which is at the same time the only decision variable at the last stage). The number of stages equals one plus the number of days for which the flower-girl sells roses without any break. The scenario based formulation of the T-stage problem can be written in the arborescent form or in scenario split form with explicit nonanticipativity constraints.

Imagine now that the flower-girl wants to earn as much as possible during the two months of her high school vacations; such 63 stages problem may be solvable thanks to its simple form. Still some other possibilities should be examined. Her problem may be rolled forward in time with a substantially shorter horizon, say, with $T = 8$ which covers a whole week. This means that the flower-girl decides as if she plans to maximize her profit over each one-week period and solves the problem every day with a known (possibly non-zero) initial supply of roses and with a new scenario tree spanning over the next $T-1$ days. Another possibility is aggregation of stages. With a long horizon and random parameters only on the right-hand sides of constraints, one may apply the idea of [19] designed for problems with an infinite horizon: One chooses a tractable horizon T and adds one stage which takes into account the remaining stages $t \ge T$.

For the simplest three stage case with independent ω_1, ω_2 it is possible to derive the decision rules explicitly (cf. [12]). We rewrite the problem in the nested form (7)–(8):

$$\max_{x_1 \ge 0}[(c-p)x_1 + E_{\omega_1}\phi_1(x_1, \omega_1)] \tag{30}$$

where

$$\phi_1(x_1, \omega_1) = \max_{s_2 \ge 0}[q_1(s_2) : s_2 \ge x_1 - \omega_1] \tag{31}$$

with

$$q_1(s_2) = \max_{x_2 \ge 0}[(c-p)x_2 + E_{\omega_2}\phi_2(x_2, s_2, \omega_2)]. \tag{32}$$

Finally,

$$\phi_2(x_2, s_2, \omega_2) = \max_{z \ge 0}[-cz : z \ge x_2 + s_2 - \omega_2] = -c(x_2 + s_2 - \omega_2)^+. \tag{33}$$

Hence, the optimal decision rule for the 3rd stage as obtained by solving (33) is $z^* = (x_2 + s_2 - \omega_2)^+$. To get the optimal recourse x_2, notice that except of the amount of stock s_2, the stochastic program (32) has the form of the newsboy problem. The optimal recourse x_2^* is then

$$x_2^*(x_1, \omega_1, s_2) = [u_{1-\alpha}(P_2) - s_2]^+ \tag{34}$$

where $u_{1-\alpha}(P_2)$ is the $(1-\alpha)$ quantile of the marginal distribution P_2 of ω_2 for $\alpha = p/c$. Monotonicity of q_1 (with $x_2 = [u_{1-\alpha}(P_2) - s_2]^+$) implies that the optimal value of s_2 in (31) is $s_2^*(x_1, \omega_1) = (x_1 - \omega_1)^+$. Substituting to (32) and (31) results in

$$\phi_1(x_1, \omega_1) = (c-p)[u_{1-\alpha}(P_2) - (x_1 - \omega_1)^+]^+ - cE_2[\max(u_{1-\alpha}(P_2); (x_1 - \omega_1)^+) - \omega_2]^+. \tag{35}$$

We conclude that in case of interstage independence and for an arbitrary nonnegative first-stage decision x_1, the second-stage decisions $s_2^*(x_1, \omega_1), x_2^*(x_1, \omega_1)$ are piecewise linear in ω_1 and the 3rd stage decision $z^* = (x_2^* + s_2^* - \omega_2)^+$ is piecewise linear in ω_1, ω_2. As to the optimal first-stage decision x_1, it is necessary to solve problem (30) with (35) substituted for $\phi_1(x_1, \omega_1)$. For discrete probability distributions the piecewise linearity of the recourse decisions implies that the optimal first-stage decision will occur in one of the breakpoints of the resulting polyhedral objective function.

Relaxation of distributional assumptions to ω_2 dependent on ω_1 means that the quantile in (34) will depend on ω_1 so that, in general, the piecewise linearity of the second-stage decisions with respect to ω_1 gets lost. An exception is the joint normal distribution where the quantiles $u_{1-\alpha}(P_2|\omega_1)$ of the conditional distributions of ω_2 conditioned by the value ω_1 of the demand on the first day are linear in ω_1.

Because of the genesis of the flower-girl problem with the state of the system identified by the number of roses s_t, $t = 1, \ldots, T-1$ available for selling on the t-day before a new order is placed and by the number of roses z_T to be thrown away at the end, its scenario-based formulation satisfies the requirement of a finite state space. (Notice, that the flower-girl problem should be more realistically formulated as an *integer* stochastic program.)

Under simplifying assumption that the random demands are independent (see Assumption 2) and identically distributed, the flower-girl problem can be easily formulated as a stochastic dynamic program and can be also solved by the backward recursion of dynamic programming. Such approach allows for characterization of the optimal policy for all states of the system and accommodates easily problems with a long horizon.

Let ω_t, $t = 1, 2, \ldots$ (random demands on the t-th day) be independent such that for all t

$$\text{Prob}[\omega_t = k-1] = p_k \quad \text{for} \quad k = 1, \ldots, K, \ \sum_{k=1}^{K} p_k = 1.$$

Since at most $K-1$ roses can be sold every day, we can restrict on policies storing at most $K-1$ roses on each day. Of course, the flower-girl starts selling one day old roses if possible.

Recalling that x_t (the control variable) denotes the number of fresh roses ordered for the t-th day ($t = 1, 2, \ldots$) and that s_t is reserved for the state variable (i.e. the stock of one day old roses left from the $(t-1)$-th day and $s_1 \equiv 0$) we immediately conclude that for any $t \geq 1$

$$s_{t+1} \leq x_t, \quad s_t + x_t \leq K - 1. \tag{36}$$

Hence for any nonnegative integers i, k such that $i + k \leq K - 1$ the transition probabilities are

$$\begin{aligned}
p(i,k;k) = \text{Prob}[s_{t+1} = k | s_t = i, x_t = k] &= \sum_{m=1}^{i+1} p_m \\
p(i,j;k) = \text{Prob}[s_{t+1} = j | s_t = i, x_t = k] &= p_{i+k+1-j} \qquad \text{for } j = 1, \ldots, k-1 \\
p(i,0;k) = \text{Prob}[s_{t+1} = 0 | s_t = i, x_t = k] &= \sum_{m=i+k+1}^{K} p_m \quad \text{for } k \geq 1
\end{aligned} \tag{37}$$

and are equal to zero otherwise.

If the order is $x_t = k$ on the t-th day when the stock left from the day $(t-1)$ equals $s_t = i$ roses, this order price is pk and the expected amount of the money obtained by selling the roses is equal to $c \sum_{j=1}^{i+k+1} (j-1)\, p_j$, i.e. the expected profit in notation used for the additive objective function (cf. (14)) with $d_t = k$, $\boldsymbol{s}_t = i$ equals

$$E_\omega \bar{f}_t(\boldsymbol{s}_t, d_t, \omega_t) \equiv c(i;k) = c \sum_{j=1}^{i+k+1} (j-1)\, p_j - pk \quad \forall t. \tag{38}$$

Now we are ready to solve the flower-girl problem as a standard stochastic dynamic programming problem. The "backward dynamic programming recursion" (17) for $i = 0, 1, \ldots, K-1$ reads:

$$V_t(i) = \max_{k=1,\ldots,K} \left[c(i;k) + \sum_{j=0}^{K-1} p(j,i;k) V_{t+1}(j) \right] \quad \text{where} \quad V_T(i) = 0 \tag{39}$$

and can be easily solved for the considered finite state space $\mathcal{S} = \{0, 1, \ldots, K-1\}$ and for a large horizon T. Obviously, the integrality of decision variables is exploited. Also the turnpike planning theorem (cf. (35)) can be helpful for problem with a very long time horizon.

However, it is hard to apply the backward recursion in more complicated problems as to the dimensionality of the state vector and/or in presence of numerous state and control constraints. In the context of the flower-girl problem, think about a whole set of traded flowers with various carry-over constraints, inclusion of a limited store space or of a substitution effect in the (random) demand.

4.2 Maintenance problem

Consider an equipment consisting of n substantial components. The attribute of each component to the equipment operating is expressed by a vector $\boldsymbol{a}_j$ $(j = 1, \ldots, n)$ of dimension $m < n$ considered as the j-th column of the matrix $\boldsymbol{A}$ (an $m \times n$ matrix). The equipment can operate according to several modes specified by m-vectors $\boldsymbol{b}(k)$ depending on value of the mode parameter k $(k = 1, \ldots, K)$. A feasible state of the equipment operating according to the k-th mode is given by the column vector (of dimension n) denoted $\boldsymbol{s} = [s^{(1)}, \ldots, s^{(n)}]^\top$ such that $\sum_{j=1}^n \boldsymbol{a}_j s^{(j)} = \boldsymbol{b}(k)$, $s^{(j)}_{\min} \le s^{(j)} \le s^{(j)}_{\max}$ with given $s^{(j)}_{\min}$, $s^{(j)}_{\max}$ $(j = 1, \ldots, n)$, written equivalently as

$$\boldsymbol{A}\boldsymbol{s} = \boldsymbol{b}(k), \qquad \boldsymbol{s}_{\min} \le \boldsymbol{s} \le \boldsymbol{s}_{\max}. \tag{40}$$

We are looking for a feasible state for which the incurred costs given by $\boldsymbol{c}^\top \boldsymbol{s}$ are minimized.

In virtue of the above conditions *we may consider only a finite number of feasible state vectors* $\boldsymbol{s}$*, the basic solutions of the linear programs* (40).

The above problem is a static one. If we assume some development over time, i.e. that the equipment is periodically checked and maintained at discrete time points $t = 1, \ldots, T$, we add time indices to the state variables; in particular, we consider $\boldsymbol{s}_t$ for $t = 1, \ldots, T$, instead of $\boldsymbol{s}$. Moreover, we assume that based on the state $\boldsymbol{s}_t$ and the mode of the equipment k_t at time $t = 1, \ldots, T$ a decision d_t is taken at time t.

In addition we assume that the development of the equipment over time is uncertain, but depends on the current state and the current mode of the equipment and on the selected decisions. In particular, we assume that based on the current state $\boldsymbol{s}_t$, current mode k_t of the equipment and the decision d_t taken at time t two things will happen:

(i) the equipment will operate at the next time point according to the operating mode $\ell \in \{1, \ldots, K\}$ with given probability $p(k_t, \ell; (\boldsymbol{s}_t, d_t))$,

(ii) transition costs $c(k_t, \ell)$ will be incurred.

From the above formulation it is clear that such a problem can be formulated in the terms of a classical finite state Markovian decision problem. In particular, supposing that the equipment starts operating in mode $k_1 = 1$, the trajectory of the system is as follows:

$$(k_1, \boldsymbol{s}_{k_1}; d_{k_1}; k_2, \boldsymbol{s}_{k_2}; d_{k_2}; k_3; \ldots k_t; \boldsymbol{s}_{k_t}; d_{k_t}; \ldots k_T, \boldsymbol{s}_{k_T}) \tag{41}$$

with $k_1 = 1$. Observe that for the selected operating mode $k_1 = 1$, state vector at time point $t = 1$ fulfills condition $\boldsymbol{A}\boldsymbol{s}_1 = \boldsymbol{b}(1)$, and similarly if the equipment is operating at time $t = 2, \ldots, T$ in mode k_t the state vector $\boldsymbol{s}_{k_t}$ fulfills condition $\boldsymbol{A}\boldsymbol{s}_{k_t} = \boldsymbol{b}(k_t)$.

Since the sequence of operating modes forms a Markov chain governed by transition probabilities $p(k, \ell; (\boldsymbol{s}, d))$, following the above decision policy the expected costs incurred are given by:

$$\begin{aligned} &\boldsymbol{c}^\top \boldsymbol{s}_1 + \sum_{k_2=1}^{K} p(1, k_2; (\boldsymbol{s}_1, d_1))[c(1, k_2) + \boldsymbol{c}^\top \boldsymbol{s}_{k_2} + \sum_{k_3=1}^{K} [p(k_2, k_3; (\boldsymbol{s}_{k_2}, d_{k_2})) \\ &[c(k_2, k_3) + \boldsymbol{c}^\top \boldsymbol{s}_{k_3} + \ldots + \sum_{k_{T-1}=1}^{K} p(k_{T-1}, k_T; \boldsymbol{s}_{k_{T-1}}, d_{k_{T-1}}))[c(k_{T-1}, k_T) + \boldsymbol{c}^\top \boldsymbol{s}_{k_T}] \ldots]] \end{aligned} \tag{42}$$

For the numerical solution of the initial dynamic programming problem formulation "the principle of optimality in dynamic programming" can be used and also problems with long time horizon and state dependent probabilities can be effectively treated.

Now let us formulate the dynamic maintenance problem as a scenario-based stochastic linear program with additive recourse given by (11)–(12) under the above conditions (i), (ii) and (40). We can readily see the similarities and the differences:

Comparing (11) with (42) we can observe that p_{k_t} in (11) are replaced by the products of transition probabilities

$$p(k_1, k_2; (\boldsymbol{s}_1, d_1)) \dots p(k_{t-1}, k_t; (\boldsymbol{s}_{t-1}, d_{t-1})).$$

Moreover, the terms

$$p(k_1, k_2; (\boldsymbol{s}_1, d_1)) \dots p(k_{t-1}, k_t; (\boldsymbol{s}_{t-1}, d_{t-1})) \cdot c(k_{t-1}, k_t)$$

reflect both the transition and its costs, and all transition matrices $\boldsymbol{B}_{k_t}$ occurring in (12) are equal to zero.

The random coefficients appear thus only on the right-hand sides and are driven by the Markov process whose trajectories correspond to the sequences of indices k_t of the considered modes. Assume that the transition probabilities $p(k_t, k_{t+1})$ may be different at different stages t but they do not depend on $\boldsymbol{s}$. Then the absence of matrices $\boldsymbol{B}$ in the constraints means that we are left with a *multiperiod two-stage stochastic linear program*. Its scenarios are all possible sequences $\sigma = \{1, k_2, \dots, k_T\}$ of states of the Markov chain occurring with probabilities

$$p^\sigma = p(1, k_2; d_1) p(k_2, k_3; d_{k_2}) \dots p(k_{T-1}, k_T; d_{k_{T-1}}).$$

Let us denote
$\boldsymbol{s}(\sigma)$ the scenario σ dependent vector composed of $\boldsymbol{s}_{k_2}, \dots, \boldsymbol{s}_{k_T}$,
$\boldsymbol{h}(\sigma)$ the scenario σ dependent vector composed of the right-hand sides $\boldsymbol{b}(k_2), \dots, \boldsymbol{b}(k_T)$,
$\widehat{\boldsymbol{A}}$ the quasidiagonal matrix with $T-1$ identical diagonal blocks $\boldsymbol{A}$,
$\hat{\boldsymbol{q}}$ the vector consisting of $T-1$ identical subvectors $\boldsymbol{c}$, and
$\gamma(\sigma) = c(1, k_2) + c(k_2, k_3) + \dots c(k_{T-1}, k_T)$.

The resulting stochastic program reads:

$$\max \left[\boldsymbol{c}^\top \boldsymbol{s}_1 + \sum_s p^\sigma \left(\gamma(\sigma) + \hat{\boldsymbol{q}}^\top \boldsymbol{s}(\sigma) \right) \right]$$

subject to

$$\boldsymbol{A}\boldsymbol{s}_1 = \boldsymbol{b}(1), \quad \widehat{\boldsymbol{A}}\boldsymbol{s}(\sigma) = \boldsymbol{h}(\sigma) \quad \forall \sigma \qquad \text{and the box constrains (40) on all } \boldsymbol{s} \text{ vectors.}$$

Notice that the objective function can be rewritten in the form (42).

There are various options how to solve such problem – a large scale linear program whose size depends on the number of scenarios to be included. There are techniques how to reduce the number of scenarios taking into account precision requirements. On the other hand, the stochastic programming formulation does not include possibility of probabilities dependent on decisions.

5 Conclusions

The two discussed approaches used for modeling and solution of discrete time dynamic decision problems under uncertainty are not competitive, they are merely complementary having different favorable and unfavorable features. Let us summarize:

For dynamic programming, definition of state of the system is essential. The structure of the problem is tied with the solution method – the backward recursion connected with the principle of optimality. A usual assumption is a finite number of possible states of the system and the goal is to get an optimal decision rule. This goal together with the backward recursion puts considerable requirements on the memory. On the other hand, problems with very distant or infinite horizon and problems with state and decision dependent transition probabilities can be treated efficiently provided that the structure of the problem fits well the solution method (based on certain Markov properties, separability of the objective function, etc.), that the dimension of the state vector is not too big and that the number of constraints is limited.

Multistage stochastic programs do not use the notion of state and their formulation is not connected with any prescribed solution technique. Therefore, there exists a variety of stochastic programs along with various

solution procedures. The emphasis is on the first-stage decision, mostly a continuous vector. It is possible to avoid special requirements on the Markov structure of the problem. Numerous constraints can be included, nevertheless, integrality of decision variables is a drawback, not an advantage from the computational point of view. The resulting problems are large mathematical programs, their dimensionality increases rapidly with increasing number of stages. The probability distribution of random parameters, which is assumed independent on the decisions, is approximated before or in course of numerical procedures.

Acknowledgement. This work was partly supported by the research project "Mathematical Methods in Stochastics" – MSM113200008 and by Grants 201/99/0264 and 402/99/1136 of the Grant Agency of the Czech Republic.

References

1 Bather, J.: Optimal decision procedures for finite Markov chains I, II, III. Adv. in Appl. Probab. **5** (1973), 328 – 339, 521 – 540, 541 – 553.
2 Bellman, R.; Glicksberg, I.; Gross, O.: On the optimal inventory equation. Management Sci. **2** (1955), 83 – 104.
3 Bellman, R. A.: Dynamic programming. Princeton Univ. Press, Princeton, NJ 1957.
4 Bertsekas, D. P.: Dynamic programming and stochastic control Academic Press, New York 1976.
5 Bertsekas, D. P.; Shreve, S. E.: Stochastic optimal control – the discrete time case. Academic Press, New York 1978.
6 Birge, J. R. et al.: A standard format for multiperiod stochastic linear programs. COAL Newsletter **17** (1987), 1 – 19.
7 Birge, J. R.; Louveaux, F.: Introduction to stochastic programming. Springer, New York 1997.
8 Blackwell, D.: Discrete dynamic programming. Ann. Math. Statist. **33** (1962), 719 – 726.
9 Casimir, R. J.: The newsboy and the flower-girl. Omega **18** (1990), 395–398.
10 Censor, Y.; Zenios, S. A.: Parallel optimization: theory, algorithms, and applications. Oxford Univ. Press, Oxford 1997.
11 Derman, C.: Finite state Markovian decision processes. Academic, New York 1970.
12 Dupačová, J.: Applications of stochastic programming under incomplete information. J. Comput. Appl. Math. **56** (1994), 113 – 125.
13 Dupačová, J.: Multistage stochastic programs: The-state-of-the-art and selected bibliography. Kybernetika **31** (1995), 151 – 174.
14 Dupačová, J.: Applications of stochastic programming: Achievements and questions. European J. Oper. Res. (2001), to appear.
15 Dupačová, J.; Consigli, G.; Wallace, S. W.: Scenarios for multistage stochastic programs. Ann. Oper. Res. **100** (2001), 25 – 53.
16 Garstka, S. J.; Wets, R. J.-B.: On decision rules in stochastic programming. Math. Programming **7** (1974), 117 – 143.
17 Gassmann, H. I.: MSLiP: A computer code for the multistage stochastic linear programming problem. Math. Programming **47** (1990), 407 – 423.
18 Grinold, R. C.: A class of constrained linear control problems with stochastic coefficients. In: Dempster, M. A. A. (ed.): Stochastic programs, Academic Press, New York 1980, pp. 97 – 108.
19 Grinold, R. C.: Infinite horizon stochastic programs. SIAM J. Control Optim. **24** (1986), 1246 – 1260.
20 Hadley, G.: Nonlinear and dynamic programming. Addison–Wesley, Reading 1964.
21 Higle, J.; Sen, S.: Stochastic decomposition. A statistical method for large scale stochastic linear programming. Kluwer, Dordrecht 1996.
22 Hinderer, K.: Foundations of non-stationary dynamic programming with discrete time parameter. Lecture Notes in Operations Research and Mathematical Systems 33. Springer, Berlin 1970.
23 Hordijk, A.: Dynamic programming and Markov potential theory. Mathematical Centre, Amsterdam 1974.
24 Infanger, G.: Planning under uncertainty: Solving large-scale stochastic linear programs. Boyd and Fraser, Danvers 1994.
25 Kall, P.; Wallace, S. W.: Stochastic programming. Wiley, Chichester 1994.
26 Kemeny, J. G.; Snell, J. L.: Finite Markov chains. Van Nostrand, Princeton, N.J. 1960.
27 Kleywegt, A. J.; Shapiro, A.: Stochastic optimization. In: Salvendy, G. (ed.): Handbook of industrial engineering, 3rd edition. Wiley, New York (to appear).
28 Mayer, J.: Stochastic linear programming algorithms: a comparison based on model management systems. Gordon and Breach Sci. Publ. 1998.
29 Mulvey, J. M.; Ruszczyński, A.: A new scenario decomposition method for large scale stochastic optimization. Oper. Res. **43** (1995), 477 – 490.
30 Nielsen, S. S.; Zenios, S. A.: Solving stochastic network programs on massively parallel computers. Math. Programming **73** (1996), 227 – 250.
31 Prékopa, A.: Stochastic programming. Kluwer, Dordrecht and Académiai Kiadó, Budapest 1995.

32 Puterman, M. L.: Markov decision processes – discrete stochastic dynamic programming. Wiley, New York 1994.
33 Rockafellar, R. T.: Cost-to-go in multistage stochastic programming. Lecture presented at the Minisymposium on Stochastic Programming, Berlin, January 22–23, 1994.
34 Rockafellar, R. T.; Wets, R. J.-B.: Continuous versus measurable recourse in N-stage stochastic programming. J. Math. Anal. Appl. **48** (1974), 836 – 859.
35 Rockafellar, R. T.; Wets, R. J.-B.: Nonanticipativity and $\mathcal{L}^1$-martingales in stochastic optimization problem. Math. Programming Study **6** (1976), 170 – 187.
36 Rockafellar, R. T.; Wets, R. J.-B.: Scenarios and policy aggregation in optimization under uncertainty. Math. Oper. Res. **16** (1991), 119 – 147.
37 Ross, S. M.: Applied probability models with optimization applications. Holden Day, San Fracisco 1970.
38 Ruszczyński, A.: Parallel decomposition of multistage stochastic programming problems. Math. Programming **58** (1993), 201 – 228.
39 Sennott, L. I.: Stochastic dynamic programming and the control of queueing system. Wiley, New York 1999.
40 Shapiro, J. F.: Turnpike planning horizons for a Markovian decision model. Management Sci. **14** (1968), 292 – 306.
41 Takriti, S.; Krasenbrink, B.; Wu, L. S.-Y.: Incorporating fuel constraints and electricity spot prices into the stochastic unit commitment problem. Oper. Res. **48** (2000), 268 – 280.
42 Varaiya, P.; Wets, R. J.-B.: Stochastic dynamic optimization approaches and computation. In: Iri, M.; Tanabe, K. (eds.): Mathematical programming, KTKSci Publ., Tokyo 1990, pp. 309–332. See also WP-88-87, IIASA Laxenburg.
43 Veinott. A. F. Jr.: On discrete dynamic programming with sensitive discount optimality criteria. Ann. Math. Statist. **40** (1969), 1635 – 1660.
44 Walkup, W. W.; Wets, R. J.-B.: A note on decision rules for stochastic programs. J. Comput. System Sci. **2** (1968), 305 – 311.
45 Wets, R. J.-B.: Stochastic programming. In: Nemhauser, G. L. et al. (eds.): Handbooks in OR & MS **1**, Chapter VII, Elsevier 1989, pp. 573 – 629.
46 Whittle, P.: Optimization over time – dynamic programming and stochastic control, Vol. II. Wiley, New York 1983.

Received June 30, 2001,

Addresses: Prof. Dr. Jitka Dupačová, Department of Probability and Mathematical Statistics, Charles University, Sokolovská 83, CZ-186 75 Prague 8, Czech Republic, email: dupacova@karlin.mff.cuni.cz; Dr. Karel Sladký, Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 4, CZ-182 08 Prague 8, Czech Republic, email: sladky@utia.cas.cz