Chapter 2
The Stationary Deterministic Model and the Basic Solution Procedure

We introduce as a prototype of deterministic dynamic optimization problems a simple allocation problem, give firstly an intuitive and then a formal description of the general problem, and derive the basic solution technique: value iteration and optimality criterion. This allows us to derive structural properties of the solution of the allocation problem.

2.1 A Motivating Example

Example 2.1.1 (Discrete allocation problem) Consider the process of allocating to a single project some parts a_0, a_1, a_2 and a_3 of a resource (such as units of material) of total amount $K := 10$ sequentially at the times $t = 0, 1, 2, 3$. (A simultaneous single allocation to four different projects leads to the same mathematical problem.) The allocation $a_t \in A := \mathbb{N}_{0,10}$ at time t is often called the *consumption.* Obviously the allocations must obey the restrictions $a_0 \leq K$ and $a_t \leq K - \sum_{i=0}^{t-1} a_i$ for $1 \leq t \leq 3$. We assume that a_t yields a reward $u(a_t)$ for some function u on A. The resource still available at time $1 \leq t \leq 4$ is $s_t := 10 - \sum_{i=0}^{t-1} a_i$, and the resource $s_t - a_t$ *not consumed at time t* is often called the investment. (Therefore allocation models often run under the heading ***consumption and investment.***) We denote the sum of allocations in the sequence $y = (a_0, a_1, a_2, a_3)$ by $c(y) := \sum_{t=0}^{3} a_t$. Then $10 - c(y)$ is the terminal resource left over at time 4. We assume that the terminal resource s_4 at the end of the allocation process yields the terminal reward $d \cdot u(s_4)$ for some constant $d \in \mathbb{R}_+$. Thus the case where the terminal resource is worthless can be modeled by the choice $d = 0$.

K. Hinderer et al., *Dynamic Optimization*, Universitext,
DOI 10.1007/978-3-319-48814-1_2

How should the sequence $y = (a_0, a_1, a_2, a_3)$ of allocations be made in order to maximize on the set $A^4(10) := \{y \in A^4 : c(y) \leq 10\}$ the sum of rewards

$$V_{4y}(10) := \sum_{t=0}^{3} u(a_t) + d \cdot u\left(10 - \sum_{t=0}^{3} a_t\right) ?$$

We want to find $V_4(10) := \sup_{y \in A^4(10)} V_{4y}(10)$ and a maximum point $y^* = (a_t^*)_0^3$ of $y \mapsto V_{4y}(10)$. This problem is denoted by $DP_4(10)$, and the problem $DP_n(s)$ for $n \geq 1$ and $s \in \{0, 1, \ldots, 10\}$ is defined analogously. ♦

If in the preceding allocation problem we allowed non-negative *real* actions a_t, some cases where u is concave and differentiable would be solvable by the classical optimization method based on Fermat's criterion in Appendix A.4.16. No counterpart is available in the discrete case of Example 2.1.1, but ***Dynamic Optimization*** will help, as shown in Example 2.4.1 below. Crucial for its applicability is the following property: If we knew an optimal first input a_0^* for $DP_4(10)$, then the vector $(a_t^*)_1^3$ could be found as a maximum point of the problem $DP_3(10 - a_0^*)$.

For arbitrary DPs the basic solution method (explained in detail in Sect. 2.3), runs as follows: If $V_n(s_0)$, $1 \leq n \leq N$, denotes the maximum value of the objective function $y \mapsto V_{ny}(s_0)$, $y \in A^n(s_0)$, one can obtain $V_n(s_0)$ from the function V_{n-1} by maximizing certain functions $a \mapsto W_n(s, a)$, $s \in S$, determined by V_{n-1}, over a certain set $D(s)$. By iterating this step, we see that the problem $DP_N(s_0)$ of maximizing $y \mapsto V_{Ny}(s_0)$ with respect to the N variables $a_0, a_1, \ldots, a_{N-1}$ reduces to a sequence of N maximization problems, each parametrized by s, with respect to a single variable a. We call this approach for the moment the ***DP method***, in contrast to other standard *static* methods. The DP method has several favorable features, as will become apparent later on in many examples:

(a) The sets $D(s)$ and the functions $a \mapsto W_n(s, a)$, $1 \leq n \leq N$, are much simpler than $A^N(s_0)$ and $y \mapsto V_{Ny}(s_0)$, respectively.
(b) The problems of the existence and of uniqueness of a maximum and of maxima on the boundary of $A^N(s_0)$ reduce to the corresponding problems for the function $a \mapsto W_n(s, a)$ of one variable only.
(c) The approach is particularly suited to the many examples where the objective function $y \mapsto V_{Ny}(s_0)$ is recursively defined and where the explicit representation, needed in general for static methods, is cumbersome; see Equation (2.4) below.
(d) The approach provides a general method for studying the important dependence of the maximal n-stage reward $V_n(s_0)$ on the initial state s_0. As an example, if S and all sets $D(s)$ are convex, a direct checking of convexity of the value functions $s \mapsto V_N(s)$ may be impossible, while the Dynamic Optimization method may work; see Chap. 8 below.
(e) In many concrete applications the number N is not known exactly, but only bounds $1 \leq N_1 \leq N \leq N_2 \leq \infty$ are known. Then it is desirable to know V_N for all N within the bounds. In static methods this requires us in general to solve

an N-stage problem for *each* of these N's. On the other hand, as we shall see below, the DP solution for N_2 automatically also contains also the solution for all $N < N_2$.

(f) The DP method provides an exact numerical algorithm for many important problems where S and all sets $D(s)$ are finite.

2.2 The Model

We now give a detailed intuitive background and basic concepts for the general ***problem DP$_N$ (s_0)***; see Fig. 2.1. The object of investigation is some system which starts at ***time*** $t = 0$ in an ***initial state*** s_0, belonging to a set S, called the ***state space.*** The system moves at times $t = 0, 1, \ldots, N-1$ successively to ***states*** s_t, i.e. $s_1, s_2, \ldots, s_N$. This movement is controlled by ***actions*** a_t, i.e. $a_0, a_1, \ldots, a_{N-1}$, respectively, taken by a decision maker at the times $t = 0, 1, \ldots, N-1$, respectively, from a set A, the ***action space.*** When discussing facts which concern states and actions at all times t we often write s and a rather than s_t and a_t, respectively, and we call s the *momentary state* and a the *momentary action.*

In examples, often the state s_t has one of the following meanings: (i) it is a summary of the *history* of the process up to time $t-1$, (ii) it represents information, necessary for the choice of an optimal action a_t, (iii) it depicts the environment in which the process is running at time t.

The number $N \in \mathbb{N}$ is called the ***horizon***, and the time interval $[t, t+1)$ is called the t-th ***period***; at ***stage*** n means *at time* $t := N-n$, $1 \le n \le N$. Each time an action is taken the momentary state of the system is assumed to be known to the decision maker. In general, when the system is in state s, not all actions from the action space A, but only those in a certain non-empty set $D(s) \subset A$ will be admissible. We call $D(s)$ the set of ***admissible actions for state s*** and

$$D := \{(s, a) \in S \times A : a \in D(s)\}$$

the ***constraint set.*** The influence of the decision maker on the transition of the system is described by a mapping $T: D \to S$, the so-called ***transition function***: If at time t the system is in state s_t and if action $a_t \in D(s_t)$ is taken, then the system moves to the new state $s_{t+1} := T(s_t, a_t)$. At time t, i.e. at the beginning of ***period*** t,

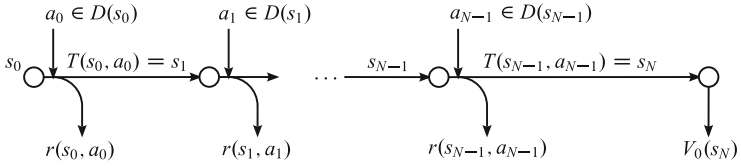

Fig. 2.1 Development of states s_t

a ***one-stage reward*** $r(s_t, a_t) \in \mathbb{R}$, given by the so-called ***one-stage reward function*** r is obtained. In addition, if the movement of the system ends at time N in state s_N, then a ***terminal reward*** $V_0(s_N) \in \mathbb{R}$ is obtained. The same monetary units, obtained at different time points, will have different cash value due to interest. This fact is taken into consideration by a so-called ***discount factor*** $\beta \in \mathbb{R}^+$; this means that the reward $r(s_t, a_t)$ obtained at time t and the terminal reward $V_0(s_N)$ at time N enter the account (2.3) below relative to time $t = 0$ as $\beta^t r(s_t, a_t)$ and $\beta^N V_0(s_N)$, respectively. In most applications early gains are more profitable than later ones, which means that $\beta < 1$.

Summing up we arrive at the following definition.

Definition 2.2.1 A ***(stationary) deterministic (dynamic) program*** (**DP** for short) is a tuple $(S, A, D, T, r, V_0, \beta)$ of the following kind:

- S is the state space.
- A is the action space.
- $D \subset S \times A$ such that all s-sections $D(s) := \{a \in A : (s, a) \in D\} \neq \emptyset$, $s \in S$. D is called the constraint set and $D(s)$ is called the set of admissible actions for state s.
- $T: D \to S$ is the transition function.
- $r: D \to \mathbb{R}$ is the one-stage reward function.
- $V_0: S \to \mathbb{R}$ is the *terminal reward function*.
- $\beta \in \mathbb{R}^+$ is the discount factor.

General assumption Throughout this book we require that both S and A are non-empty sets.

We also call the tuple $(S, A, D, T, r, V_0, \beta)$ the *data* of the DP. The data for the allocation problem from Example 2.1.1 with arbitrary $K \in \mathbb{N}$ are as follows: $S := A := \mathbb{N}_{0,K}$, $D(s) := \mathbb{N}_{0,s}$ for all s, $T(s, a) = s - a$, $r(s, a) := u(a)$ for $(s, a) \in D$; $V_0 = d \cdot u$, and β is arbitrary. One also could model the problem by using for a_t the investment at time t. Then S, A, D, V_0 and β would remain unchanged, while $T(s, a) = a$ and $r(s, a) := u(s - a)$ for $(s, a) \in D$.

Remark 2.2.2 In applications the states and/or actions are often integers or reals, but sometimes they are elements of $\mathbb{Z}^d$ or of $\mathbb{R}^d$ for $d \geq 2$ or they are sets. ◇

Remark 2.2.3 Sometimes the one-stage reward $r(s, a, s')$ also depends on the next state $s' = T(s, a)$. That case is covered by simply replacing s' by $T(s, a)$.

In a few cases, r also depends on β; in particular, if $r(s, a)$ consists of a reward $g(s, a)$ obtained at the end of the momentary period then $r(s, a) = \beta g(s, a)$. A dependence of r on β requires changes only for those few results which deal with the dependence of the solution on β. ◇

Remark 2.2.4 (The discount factor) One must not distinguish in the theory between $\beta < 1$, $\beta = 1$ and $\beta > 1$. On the other hand, for many models the pointwise limit of the value functions V_n for $n \to \infty$, dealt with in Chap. 10, is not defined unless $\beta < 1$. Also in economical applications β is usually smaller than one. If the N

periods have nothing to do with time but mean that a certain activity is executed N times, then only $\beta = 1$ is meaningful. $\diamond$

Any concrete problem $DP_N(s_0)$ must be modeled by an appropriate choice of S, A, D, T, r, V_0 and β as done above for the allocation problem from Example 2.1.1. Particularly important is the choice of the state space S. Sometimes several choices are possible; then it is up to the decision maker's skill to find a formulation that easily admits theoretical analysis and computation. This skill can be acquired only by experience. Dreyfus and Law (1977) speak in the title of their book of the *Art of Dynamic Programming*, i.e. the art of finding an appropriate model. The same authors also suggest (loc. cit., p. 17) a useful mental device, called the *consultant question*, for a skillful choice of the state s_t. Essentially it reads as follows: *The momentary state should consist of the minimal information about the momentary situation you would have to acquire from a firm in case you would be hired to take over the problem and do things optimally from now on.*

We call a ***set continuous [discrete]*** if it is an interval or a product of intervals [an interval in $\mathbb{Z}$ or a product of such intervals]. We call a DP ***continuous [discrete]*** if either S or $D(s)$, $s \in S$, are continuous [if both S and $D(s)$, $s \in S$, are discrete]. The modeling procedure should also include a reflection about the question of whether to use a discrete or a continuous DP. More information on this feature is given before Example 2.4.3 below.

Now we define for a given model DP the ***maximization problem*** $DP_N(s_0)$, determined by an arbitrary horizon N and an arbitrary initial state $s_0 \in S$. Firstly, we say that a ***sequence*** $y := (a_t)_0^{N-1}$ is a ***sequence of admissible actions*** for s_0, if y obeys the restrictions

$$\begin{aligned} a_0 &\in D(s_0), \\ a_1 &\in D(s_1), \text{ where } s_1 := T(s_0, a_0), \\ a_2 &\in D(s_2), \text{ where } s_2 := T(s_1, a_1), \\ &\vdots \\ a_{N-1} &\in D(s_{N-1}), \text{ where } s_{N-1} := T(s_{N-2}, a_{N-2}). \end{aligned} \tag{2.1}$$

(As an example, in the allocation problem from Example 2.1.1 with $N = 4$ and $K = 8$ the action sequence $(4, 3, 1, 0)$ is admissible for s_0 if and only if $s_0 \geq 8$.) In the final state $s_N := T(s_{N-1}, a_{N-1})$ no action is taken. The set of action sequences admissible for s_0 will be denoted by $A^N(s_0)$; it is non-empty because $D(s) \neq \emptyset$ for each s; we have $A^N(s_0) = A^N$ for all s_0 if $D(s) = A$ for all s; $A^N(s_0)$ is finite if all sets $D(s)$ are finite. Even for simple sets $D(s)$ the sets $A^N(s_0)$ can be complicated as seen, for instance, from the allocation problem from Example 2.1.1.

An initial state s_0 and a sequence $(s_t)_1^N$ of states as introduced in (2.1) above describes the evolution of the system under an admissible action sequence $y \in A^N(s_0)$. We call $(s_t)_1^N$ the ***decision process*** generated by (s_0, y). Notice that $s_t = s_t(s_0, (a_i)_0^{t-1})$ is a function of s_0 and of y, since

$$s_1 = T(s_0, a_0),\ s_2 = T(s_1, a_1) = T(T(s_0, a_0), a_1),$$
$$s_3 = T(T(T(s_0, a_0), a_1), a_2),\ \ldots\ .$$

It follows from (2.1) that the sets $A^n(s)$, $n \geq 1$, $s \in S$, have the following sequential structure: $A^1(s) = D(s)$ and

$$A^n(s) = \left\{(a, x) \in D(s) \times A^{n-1} : x \in A^{n-1}(T(s, a))\right\},\ n \geq 2. \tag{2.2}$$

For initial state $s_0 \in S$, admissible action sequence $y = (a_t)_0^{N-1} \in A^N(s_0)$ and for $(s_t)_1^N$ generated by (s_0, y) the ***N-stage reward*** is the real number

$$V_{Ny}(s_0) := \sum_{t=0}^{N-1} \beta^t r(s_t, a_t) + \beta^N V_0(s_N) = r(s_0, a_0) \tag{2.3}$$
$$+ \sum_{t=1}^{N-1} \beta^t r(s_t(s_0, (a_i)_0^{t-1}), a_t) + \beta^N V_0(s_N(s_0, (a_i)_0^{N-1})).$$

Thus $y \mapsto V_{Ny}(s_0)$ is the ***objective function*** of the problem $DP_N(s_0)$. Notice that $V_{Ny}(s_0)$ means the total reward *discounted back* to time $t = 0$; the total reward accumulated at time N is $V_{Ny}(s_0)/\beta^N$. The complicated explicit representation (check it for $N = 3$)

$$V_{Ny}(s_0) = r(s_0, a_0) + \beta r(T(s_0, a_0), a_1) + \beta^2 r(T(T(s_0, a_0), a_1), a_2)$$
$$+ \cdots + \beta^N V_0(T(T(\ldots), a_{N-1})) \tag{2.4}$$

is rarely needed, but for some applications the explicit expression may be useful for checking the correct choice of the data. Also keep in mind that $y \mapsto V_{Ny}(s_0)$ is a function of the form $\sum_{t=0}^{N-1} g_t(s_0, a_0, a_1, \ldots, a_t)$.

Now the N-stage maximization problem $DP_N(s_0)$ for $N \geq 1$ and $s_0 \in S$ reads as follows:

(i) Compute the ***maximal N-stage reward*** *for initial state* s_0

$$V_N(s_0) := \sup\{V_{Ny}(s_0) : y \in A^N(s_0)\}.$$

(ii) Find, if possible, an ***s_0-optimal action sequence***, i.e. a maximum point of the objective function $y \mapsto V_{Ny}(s_0)$.

The function $V_N: S \to (-\infty, \infty]$ is called the ***N-stage value function***. The set of problems $DP_N(s_0)$, $s_0 \in S$, is called the ***problem DP$_N$***. Notice that $V_N(s_0)$ is finite if there exists an s_0-optimal action sequence. The sequence $(V_n)_1^N$ of value functions plays a central role for solving DP_N. From now on we shall mostly write $V_N(s)$ instead of $V_N(s_0)$. Only in rare cases will the value functions have an explicit form. Of course, s_0-optimal action sequences need not exist, and if they exist they need not be unique.

2.3 The Basic Solution Procedure

In the following generalization of Appendix A.4.5 the set $M(b)$ is the b-section of M (cf. Appendix A.3.8).

Lemma 2.3.1 (The joint supremum equals the iterated supremum) *Let B and C be non-empty sets and let v be a function on a set $M \subset B \times C$ for which $M(b) \neq \emptyset$ for all $b \in B$. Then*

$$\sup_{(b,c)\in M} v(b,c) = \sup_{b\in B} \sup_{c\in M(b)} v(b,c).$$

Proof Put $h(b) := \sup\{v(b,c) : c \in M(b)\}$. We have to show that $\sup v = \sup h$. From

$$v(b,c) \le h(b) \le \sup h$$

we get $\sup v \le \sup h$. On the other hand, from $v(b,c) \le \sup v$ we firstly obtain $h(b) \le \sup v$ and then $\sup h \le \sup v$. □

The first step towards the value iteration (2.7) is the next result. For $a \in D(s)$ and $x \in A^{n-1}(T(s,a))$ we denote by (a,x) the n-stage action sequence which first uses a and then x.

Lemma 2.3.2 (The reward iteration, RI for short) *The following holds*

$$\begin{aligned} V_{1a}(s) &= r(s,a) + \beta V_0(T(s,a)), (s,a) \in D, \\ V_{n,(a,x)}(s) &= r(s,a) + \beta V_{n-1,x}(T(s,a)), \\ &\qquad n \ge 2,\ (s,a) \in D,\ x \in A^{n-1}(T(s,a)). \end{aligned} \tag{2.5}$$

Proof The form of V_{1a} follows from (2.3) with $N := 1$ since $a_1 = T(s,a)$. Now assume $n \ge 2$. For $0 \le t \le n-2$ put $s'_t = s_{t+1}$ and $a'_t = a_{t+1}$, and put $s'_{n-1} := s_n$. Then $s'_t = T(s'_{t-1}, a'_{t-1})$ and $x = (a'_t)_0^{n-2}$. It follows easily that x and $(s'_t)_0^{n-1}$ satisfy (2.1) with $N := n-1$ and with $(s_t)_1^N$ and $(a_t)_0^{N-1}$ replaced by $(s'_t)_1^{n-1}$ and $(a'_t)_0^{n-2}$, respectively. This means that $(a'_t)_0^{n-2} \in A^{n-1}(s'_0)$ and that $(s'_t)_1^{n-1}$ is the

decision process generated by (s'_0, x). Now (2.3) yields $V_{n,(a,x)}(s_0) = r(s_0, a_0) + \beta \cdot B$ where

$$B := \sum_{t=1}^{n-1} \beta^{t-1} r(s_t, a_t) + \beta^{n-1} V_0(s_n) = \sum_{t=0}^{n-2} \beta^t r(s'_t, a'_t) + \beta^{n-1} V_0(s'_{n-1})$$
$$= V_{n-1,x}(s'_0) = V_{n-1,x}(T(s_0, a_0)).$$

Inserting B into $V_{n,(a,x)}(s_0) = r(s_0, a_0) + \beta \cdot B$ completes the proof. □

The RI expresses the following fact: The n-stage discounted reward for the initial state s under the action sequence (a, x) equals the sum of the reward in the first period and the discounted $(n-1)$-stage reward for the initial state $T(s, a)$ under the action sequence x. Thus the recursion (2.5) exhibits the sequential structure of the objective functions $y \mapsto V_{ny}(s)$. Moreover, in case of finite S and A the RI is a convenient recursive algorithm for evaluating $V_{Ny}(s)$ on a computer.

We often use the functions $W_n : D \to (-\infty, \infty]$, defined by

$$W_n(s, a) := r(s, a) + \beta V_{n-1}(T(s, a)),\ n \geq 1. \tag{2.6}$$

A mapping f from S into A is called a ***decision rule*** if $f(s) \in D(s)$ for all s. Denote the ***set of all decision rules*** by $\mathbb{F}$. A ***decision rule f_n at stage n*** such that $f_n(s)$ is a maximum point of $a \mapsto W_n(s, a)$ for all s is called a ***maximizer at stage n***. Intuitively, $f_n(s)$ is an *optimal action* at state s when n periods are still ahead. A sequence $(f_n)_N^1 := (f_N, f_{N-1}, \ldots, f_1) \in \mathbb{F}^N$ of decision rules f_n at stage n is called an ***N-stage policy*** and it is called an ***N-stage maximizing policy*** if f_n is a maximizer at stage n for $1 \leq n \leq N$.

Of course, a maximizing policy need not exist, and if it exists, it need not be unique. Sometimes we need the following generalization of the concept of a maximizer at stage n: If w is a function on D, we call a decision rule f a ***maximizer*** *of* w if $f(s)$ is a maximum point of $a \mapsto w(s, a)$ for all s.

Theorem 2.3.3 (Basic Theorem for stationary DPs)

*(a) The value functions V_n satisfy the following recursion, called **value iteration** (**VI** for short):*

$$V_n(s) = \sup\{r(s, a) + \beta V_{n-1}(T(s, a)) : a \in D(s)\},$$
$$= \sup\{W_n(s, a) : a \in D(s)\}, \quad n \geq 1, s \in S. \tag{2.7}$$

(b) Let $N \geq 1$, let s_0 be an arbitrary initial state, let the action sequence $y^* = (a_t^*)_0^{N-1}$ be admissible for s_0 and let $(s_t)_1^N$ be the decision process generated by s_0 and y^*. Then y^* is s_0-optimal if and only if a_t^* is a maximum point of $a \mapsto W_{N-t}(s_t, a)$, $0 \leq t \leq N-1$.

*(c) The **Optimality Criterion** (**OC** for short): Let $N \geq 1$, let s_0 be an arbitrary initial state. If there exists a maximizing policy $(f_n)_N^1$ then:*

(c1) An s_0-optimal action sequence $(a_t^*)_0^{N-1}$ is given by the following ***forward procedure***

$$a_t^* := f_{N-t}(s_t), \quad s_{t+1} := T(s_t, a_t^*), \quad 0 \le t \le N-1. \tag{2.8}$$

If the maximizing sequence $(f_n)_N^1$ is unique, then $(a_t^*)_0^{N-1}$ is the unique s_0-optimal action sequence.

(c2) V_n, $n \ge 1$, is determined by f_n and W_n, since

$$V_n(s) = W_n(s, f_n(s)), \; s \in S.$$

Proof

(a) Fix s. Equation (2.7) follows for $n = 1$ immediately from (2.5). For $n \ge 2$ we use Lemma 2.3.1 for $b := a$, $B := D(s)$, $c := x$, $C := A^{n-1}$, $M := A^n(s)$ and $v(a,x) := V_{n,(a,x)}(s)$. From the recursive property (2.2) of $A^n(s)$ we see that the a-section $M(a)$ of M equals $A^{n-1}(T(s,a))$. Using the RI (2.5) and noting that $\beta > 0$, we obtain

$$\begin{aligned} V_n(s) &= \sup\{V_{n,(a,x)}(s) : (a,x) \in M\} \\ &= \sup_{a \in D(s)} \sup_{x \in A^{n-1}(T(s,a))} [r(s,a) + \beta V_{n-1,x}(T(s,a))] \\ &= \sup_a [r(s,a) + \beta \sup_x V_{n-1,x}(T(s,a))] \\ &= \sup_a [r(s,a) + \beta V_{n-1}(T(s,a))]. \end{aligned}$$

(b) From the VI we infer for $0 \le t \le N-1$, since $s_{t+1} = T(s_t, a_t^*)$, that

$$V_{N-t}(s_t) \ge r(s_t, a_t^*) + \beta V_{N-t-1}(s_{t+1}),$$

with equality if and only if a_t^* is a maximum point of $W_{N-t}(s_t, \cdot)$. Thus

$$\begin{aligned} V_N(s_0) &\ge r(s_0, a_0^*) + \beta V_{N-1}(s_1) \\ &\ge r(s_0, a_0^*) + \beta r(s_1, a_1^*) + \beta^2 V_{N-2}(s_2) \\ &\ge \ldots \ge \sum_0^{N-1} \beta^t r(s_t, a_t^*) + \beta^N V_0(s_N) = V_{Ny^*}(s_0), \end{aligned}$$

and equality holds throughout if and only if for $0 \le t \le N-1$ the action a_t^* is a maximum point of $W_{N-t}(s_t, \cdot)$.

(c) (c1) follows from (b) since $(a_t^*)_0^{N-1}$ satisfies the condition in (b); the assertion about uniqueness is obviously true. (c2) is obvious from (2.6). □

Remark 2.3.4 The VI says that the maximal reward for n periods and initial state s equals the maximum—over the initially admissible actions a—of the sum of the reward earned in the first period and the discounted maximal reward for the last $n-1$ periods and the next state $T(s,a)$ as initial state. This fact is intuitively plausible to such an extent that part of the literature refrains from a proof. ◇

Remark 2.3.5 The computation of $V_N(s_0)$ for some $N \geq 2$ by the VI also yields $V_n(s)$, $1 \leq n \leq N-1$, $s \in S$. Yet the solution of $DP_n(s)$ requires in addition the computation of an s-optimal $y \in A^n(s)$ by means of (2.8). ◇

Remark 2.3.6 The OC yields an s_0-optimal action sequence y^* for *each* s_0. If only an s_0-optimal action sequence for *a single* s_0 is required, it follows from (2.8) that it suffices to compute a maximum point $f_N(s_0)$ of $W_N(s_0,\cdot)$ instead of a whole maximizer f_N at stage N.

The s_0-optimal action sequence obtained by (2.8) is called the ***action sequence*** *generated by* s_0 and the ***maximizing policy*** $(f_n)_N^1$. ◇

Remark 2.3.7 We call Theorem 2.3.3 the *Basic Theorem* as it will play a dominant role throughout Part I and in modified form in the other chapters. Other names used in the literature include ***DP algorithm***, ***method of backward induction*** and above all, ***Bellman's principle of optimality.*** We reserve the latter name for another result, see Supplement 3.6.1. ◇

Remark 2.3.8 While the Basic Theorem 2.3.3 reduces the global N-stage problem to a sequence of N interconnected parametric one-stage optimization problems it does not tell us anything about how to solve the latter problems. For these one depends on methods of non-dynamic optimization. ◇

Remark 2.3.9 The VI holds whether or not there exist s_0-optimal action sequences. Since r is finite, the right-hand side of (2.7) is also defined in the case $V_{n-1}(T(s,a)) = \infty$ by our convention $x + \infty := \infty$ for real x. ◇

The essence of the proof of the VI may be phrased in the simple equation

$$\sup_{(a,x)}[g(a) + \beta h(a,x)] = \sup_a[g(a) + \beta \sup_x h(a,x)].$$

Unfortunately this simple method is not applicable for stochastic DPs.

For $s \in S$ and $n \geq 1$ we call the (possibly empty) set $\boldsymbol{D_n^*(s)}$ of maximum points of $a \mapsto W_n(s,a)$ the ***set of optimal actions*** for stage n and at state s. Thus $(f_n)_N^1$ is maximizing if and only if $f_n(s) \in D_n^*(s)$ for $1 \leq n \leq N$ and all s. The sequence $(D_n)_N^1$ determines for each s_0 *all* solutions of $DP_N(s_0)$ since by Theorem 2.3.3(b) $(a_t^*)_0^{N-1}$ is s_0-optimal if and only if $a_t^* \in D_{N-t}^*(s_t)$ for $0 \leq t \leq N-1$.

Only in rare cases will the value functions and s_0-optimal action sequences have an explicit form. Therefore, the computer-aided numerical solution, possibly after suitable discretization of the state and action space, is important. Assume for simplicity that both S and A are finite. We call the method provided by

the Basic Theorem for solving problem $DP_N(s_0)$ the ***VI algorithm.*** It runs as follows.

One computes, starting with V_0, recursively by means of the ***backward procedure*** (2.7) (i.e. backward in stages) the value functions $V_1, V_2, \ldots, V_{N-1}$ by maximizing the (possibly infinite) functions $a \mapsto W_n(s,a)$ for all $s \in S$ and $1 \le n \le N-1$. After having computed V_n, the function V_{n-1} can be deleted from the memory. In the final step one computes $V_N(s_0)$ by maximizing $a \mapsto W_N(s_0,a)$. In practice one often computes $V_N(s_0)$ for *all* $s_0 \in S$.

The computation of an s_0-optimal action sequence $(a_t^*)_0^{N-1}$ for given initial state s_0 according to Theorem 2.3.3(b) cannot be done simultaneously with the recursive computation of the value functions, as one does not know in advance which sequence of states $s_1, s_2, \ldots$ is generated by s_0 and the optimal action sequence to be constructed. However, while maximizing $W_n(s,\cdot)$, $s \in S$, for $1 \le n \le N-1$ and $W_N(s_0,\cdot)$ one can compute and store a maximum point $f_n(s)$ and $f_N(s_0)$, respectively. Then one obtains an s_0-optimal action sequence by the forward procedure in the OC.

2.4 First Examples

Example 2.4.1 (Solution of Example 2.1.1) We treat this problem for arbitrary $K \in \mathbb{N}$ and N rather than only $K = 10$ and $N = 4$.

(a) After the definition of a DP we have seen that $S = A = \mathbb{N}_{0,K}$, $D(s) = \mathbb{N}_{0,s}$, $T(s,a) = s-a$, $r(s,a) = u(a)$ and $V_0 = d \cdot u$ for some $d \in \mathbb{R}_+$ and an arbitrary function u on $\mathbb{N}_{0,K}$. Note that $A^N(s_0) = \{(a_t)_0^{N-1} \in A^N : \sum_0^{N-1} a_t \le s_0\}$ since $\sum_0^{N-1} a_t \le s_0$ implies $a_t \le s_0 - \sum_{i=0}^{t-1} a_i = s_t$ for $0 \le t \le N-1$. Because of $T(s,a) \le s$, $(s,a) \in D$, the solution of $DP_N(s_0)$ is the same for each DP with $S = A = \mathbb{N}_{0,K}$ whenever $K \ge s_0$; even $S = A = \mathbb{N}_0$ could be used. As a consequence, it suffices to solve $DP_N(K)$ with the choice $S = A = \mathbb{N}_{0,K}$.

From now on assume that u is increasing and that $u(0) = 0 < u(K)$. (The case $u(K) = 0$, i.e. $u \equiv 0$, is trivial.) By Theorem 2.3.3(a) the VI has the form

$$V_n(s) = \max\{u(a) + \beta V_{n-1}(s-a) : a \in \mathbb{N}_{0,s}\}, \quad n \ge 1, s \in \mathbb{N}_{0,K}. \tag{2.9}$$

This implies by induction on $n \ge 0$ that $V_n(0) = 0$. Due to the discreteness of S and A, even for simple utility functions u one can expect only in very rare cases an explicit solution. However, for arbitrary u we can find a numerical solution by means of (2.9). As we show below, for u with *sufficient structure* we also can find structural properties of the solution, i.e. of V_n, of the smallest maximizer f_n at stage n and of those s_0-optimal action sequences $y^* = y^N(s_0) \in A^N(s_0)$ which are generated by s_0 and $(f_n)_N^1$.

Table 2.1 $V_n(s)$, $s \le K := 8$, for $u(a) = \sqrt{a}$, $d = 1.5$, $\beta = 0.8$

	s								
n	0	1	2	3	4	5	6	7	8
1	0.000	1.200	2.200	2.697	3.111	3.493	3.814	4.132	4.415
2	0.000	1.000	1.960	2.760	3.174	3.572	3.903	4.221	4.526
3	0.000	1.000	1.800	2.568	3.208	3.622	3.954	4.272	4.590
4	0.000	1.000	1.800	2.440	3.054	3.566	3.981	4.312	4.630
5	0.000	1.000	1.800	2.440	2.952	3.444	3.858	4.267	4.599
6	0.000	1.000	1.800	2.440	2.952	3.366	3.776	4.169	4.500
7	0.000	1.000	1.800	2.440	2.952	3.366	3.776	4.107	4.435
8	0.000	1.000	1.800	2.440	2.952	3.366	3.776	4.107	4.435
9	0.000	1.000	1.800	2.440	2.952	3.366	3.776	4.107	4.435

Table 2.2 $f_n(s)$, $s \le K := 12$, for $u(a) = \sqrt{a}$, $d = 1.5$, $\beta = 0.8$

	s												
n	0	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	1	1	2	2	2	3	3	4	4	5	5
2	0	1	1	1	2	2	2	3	3	4	4	4	5
3	0	1	1	1	1	2	2	2	3	4	4	4	4
4	0	1	1	1	1	1	2	2	3	4	4	4	4
5	0	1	1	1	1	1	2	2	2	3	4	4	4
6	0	1	1	1	1	2	2	2	2	2	3	4	4
7	0	1	1	1	1	2	2	2	2	3	3	4	4
8	0	1	1	1	1	2	2	2	2	3	4	4	4
9	0	1	1	1	1	2	2	2	2	3	4	4	4

(b) The Tables 2.1 and 2.2 and Fig. 2.2 show the result of computations for $u(a) = \sqrt{a}$, $d = 1.5$ and $\beta = 0.8$. We denote by f_n the smallest maximizer at stage n. One quickly obtains by the forward procedure (2.8) the subsequent K-optimal action sequences y^*; the resulting terminal state s_N can be used as control, since the sum of the actions and of s_N equals K.

$$
\begin{aligned}
&N = 4,\ K = 8, \quad y^* = (3, 2, 1, 1), && s_4 = 1,\ V_4(8) = 4.630\\
&N = 6,\ K = 8, \quad y^* = (2, 2, 1, 1, 1, 0), && s_6 = 1,\ V_6(8) = 4.500\\
&N = 9,\ K = 12,\ y^* = (4, 2, 2, 1, 1, 1, 1, 0, 0), && s_9 = 0,\ V_9(12) = 5.548.
\end{aligned}
$$

From the Tables 2.1 and 2.2 below one will conjecture that $V_n = V_7$ and that $f_n = f_7$ for $n \ge 8$. This rare property can be confirmed by Proposition 4.1.4.

(c) In later sections we systematically study structural properties of the solution of general DPs and apply these to our allocation problem. Here we give several results (c1)–(c7) which were suggested either by intuition or by numerical computations. Some of these results can be proved already here, and some

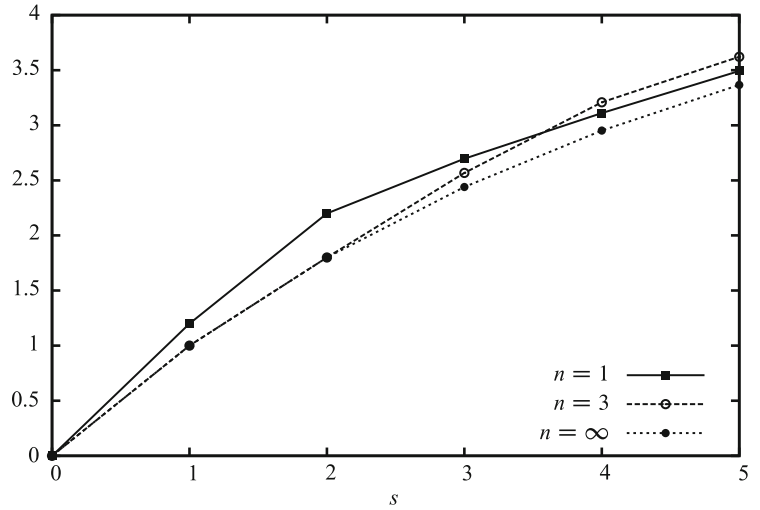

Fig. 2.2 Functions V_n for $n = 1$, $n = 3$ and $n = \infty$ (limit function V) for $K = 8$, $u(a) = \sqrt{a}$, $d = 1.5$ and $\beta = 0.8$

use an ad hoc method rather than the Basic Theorem. Discrete concavity and discrete convexity are defined in Appendix (D.4).

(c1) $V_n(s)$ is increasing in s. This is plausible since we expect from a larger initial resource a larger maximal reward. A proof can be given by Theorem 6.3.5, by Example 6.4.1(c1) or directly as in Problem 2.5.1.

(c2) The number $\overline{s} := \max\{0 \le x < K : u(x) = 0\}$ can be interpreted as follows: If one allocates energy to a technical system, then $\overline{s}+1$ is the minimal allocation which causes the system to work with profit. For $0 \le s \le \overline{s}$ and $n \ge 0$ we have $V_n(s) = 0 = f_{n+1}(s)$. Moreover, each $y \in A^N(s)$ is s-optimal. These statements are simple consequences of Problem 2.5.1(d) and of the VI (2.7).

(c3) $V_n(s)$ is increasing in n for *small* d [decreasing in n for *large* d], e.g. if $d \le 1$ [if $\beta < 1$, $d \ge 1/(1-\beta)$]. (As seen from Table 2.1, $V_n(3)$ is in general neither increasing nor decreasing in n.)

For the *proof* one easily derives from the VI (2.7) by induction on $n \ge 0$ that $V_n(s)$ is increasing [decreasing] in n if $V_1 \ge V_0 = d \cdot u$ [$V_1 \le V_0 = d \cdot u$].

(i) If $d \le 1$ then the VI yields for $s \in \mathbb{N}_{0,K}$:

$$V_1(s) = \max_{0 \le a \le s} [u(a) + \beta d u(s-a)] \ge u(s) \ge du(s) = V_0(s).$$

(ii) Recall that u is increasing on $\mathbb{N}_{0,K}$ (cf. Definition 6.2.1(vii)). Then if $\beta < 1$, $d \ge 1/(1-\beta)$ we have:

$$V_1(s) \le \max_{0 \le a \le s} u(a) + \beta d \max_{0 \le a \le s} u(s-a) = u(s) \cdot (1 + \beta d) \le du(s).$$

(c4) Let u be discretely concave (cf. Appendix (D.1)). Then all value functions V_n are discretely concave, f_n is increasing and its upward jumps have size one; see also Table 2.2. Moreover, if g_n denotes the largest maximizer at stage $n \geq 1$, then a mapping $f: \mathbb{N}_{0,K} \to \mathbb{N}_{0,K}$ is a maximizer at stage n if and only if $f_n \leq f \leq g_n$. All results follow from Theorem 7.1.2 below by using as actions the investments.

(c5) Let u be discretely convex. In Theorem 7.3.3 below we compute the value functions explicitly and show that either $(0, 0, \ldots, 0)$ or $(s_0, 0, \ldots, 0)$ are s_0-optimal. These two action sequences are *extreme* in the sense that they prescribe to consume nothing at all times $0 \leq 1 \leq N-1$ or to consume everything at time $t = 0$, respectively. Moreover, Example 7.3.5 below shows in case $\beta < 1$ and $s > s_0$ that $V_n = u$ for all $n \geq m$ and some $m \in \mathbb{N}$, and that for each s_0 the action sequence $(s_0, 0, \ldots, 0) \in A^N(s_0)$ is s_0-optimal for all $N \geq m$. In particular, if $\beta < 1$, then for some $m \in \mathbb{N}$ we have $V_n = u$ for $n \geq m$, and $(s_0, 0, \ldots, 0) \in A^N(s_0)$ is s_0-optimal for $N \geq m$.

(c6) The value functions V_n, $n \geq 1$, are Lipschitz continuous in d and also in β, both uniformly in s. In fact, for each s and for $d, d' \in \mathbb{R}_+$ we get, using Appendix A.4.4

$$\begin{aligned} |V_n(s,d) - V_n(s,d')| &= |\max_y V_{ny}(s,d) - \max_y V_{ny}(s,d')| \\ &\leq \max_y |V_{ny}(s,d) - V_{ny}(s,d')| = \beta^n |d - d'| \cdot \max u. \end{aligned}$$

Moreover, for each s and for $\beta, \beta' \in (0, 1]$ we get

$$\begin{aligned} |V_n(s,\beta) - V_n(s,\beta')| &\leq \max_y |V_{ny}(s,\beta) - V_{ny}(s,\beta')| \\ &\leq \left(\sum_0^{n-1} |\beta^t - \beta'^t| + d|\beta^n - \beta'^n| \right) \max u \\ &\leq n(n-1+2d) \cdot \max u \cdot |\beta - \beta'|/2. \end{aligned}$$

Here we used that $|\beta^t - \beta'^t| = |\beta - \beta'| \cdot \sum_{i=0}^{t-1} \beta^i \beta'^{t-i} \leq t|\beta - \beta'|$, $t \geq 1$.

(c7) If u is discretely concave and if $\beta = d = 1$ one expects that it is optimal to allocate the resources among the N stages as evenly as possible in the sense that there exists an s_0-optimal action sequence $y^* \in A^N(s_0)$ whose actions differ from each other by not more than one unit. This is true as one can show, using $b := \lfloor s_0/(N+1) \rfloor$, that y^* is s_0-optimal if $(N+1)(b+1) - s_0$ of the components of y^* equal b, and if the remaining ones equal $b+1$. ♦

We conclude our investigation of the allocation problem from Example 2.4.1 by studying the asymptotic behavior of the solution for $n \to \infty$. Such problems are studied in detail for general DPs in Chap. 10; here we can solve it by an ad hoc approach.

Proposition 2.4.2 (Asymptotic properties of Example 2.4.1) *Assume that* $\beta < 1$ *and* $V(0) := 0$ *and define* $V(s)$, $1 \le s \le K$, *by induction on* s *according to*

$$V(s) = \max_{1 \le a \le s} \{u(a) + \beta V(s-a)\}, \quad 1 \le s \le K. \tag{2.10}$$

Then:

(a) $V_n(s)$ *converges for* $n \to \infty$ *to* $V(s)$, $s \in \mathbb{N}_{0,K}$.
(b) V *is increasing and, if* u *is discretely concave, also discretely concave.*
(c) Let $f(s)$ *be a maximum point of* $a \mapsto u(a) + \beta V(s-a)$, $0 \le s \le K$. *For* $n \ge 1$ *and* $1 \le s \le K$ *put* $V_{nf}(s) := V_{ny}(s)$, *where* $y \in A^n(s)$ *is generated by* s *and the policy* $(f)_0^{n-1}$. *Then the decision rule* f *is* ***asymptotically optimal*** *in the sense that for* $1 \le s \le K$

$$|V_n(s) - V_{nf}(s)| \to 0 \text{ for } n \to \infty. \tag{2.11}$$

Proof

(a1) Firstly, assertion (a) holds for $s = 0$ since $V_n(0) = 0 \to 0 = V(0)$ for $n \to \infty$. Next, V is real-valued on the finite set $S = \mathbb{N}_{0,K}$, hence bounded. The same holds for the value functions since $0 \le V_n \le [1/(1-\beta)+d] \cdot \max u$. In fact, the lower bound holds trivially since $u \ge 0$ and $V_n(0) = 0$, and the upper bound follows from (2.10).

(a2) Fix $1 \le s \le K$. Since $\beta < 1$ and $V \ge 0$ we see, using $W(s,a) := u(a) + \beta V(s-a)$ for $0 \le a \le s$, that

$$V(s) = \max_{1 \le a \le s} W(s,a) \le \max_{0 \le a \le s} W(s,a) = \max\{\beta V(s), V(s)\} = V(s),$$

hence $V(s) = \max_{0 \le a \le s} W(s,a)$. Let $\|\cdot\|$ be the maximum-norm on S. Now we get, using Appendix A.1.3(b)

$$\begin{aligned} |V_n(s) - V(s)| &= |\max_a [u(a) + \beta V_{n-1}(s-a)] - \max_a [u(a) + \beta V(s-a)]| \\ &\le \beta \max_a |V_{n-1}(s-a) - V(s-a)| \le \beta \|V_{n-1} - V\|. \end{aligned}$$

Here a runs over $\mathbb{N}_{0,s}$. Now induction on $n \ge 0$ implies $\|V_n - V\| \le \beta^n \|V_0 - V\|$, which proves (a).

(b) This follows from (a) and properties (c1) and (c4) from Example 2.4.1 above since isotonicity and discrete concavity of V_n are easily seen to be preserved when n tends to ∞.

(c) Firstly, from the definition of f we know that $V(s) = u(f(s)) + \beta V(s - f(s))$. The RI (2.5) shows that $V_{nf}(s) = u(f(s)) + \beta V_{n-1,f}(s - f(s))$, $n \ge 1$, where $V_{0f} := V_0$. Now one easily obtains, using induction on $n \ge 0$, that $\|V - V_{nf}\| \le$

$\beta^n \|V - V_0\|$. Finally the assertion follows, using (a2), from

$$\|V_n - V_{nf}\| = \|V_n - V + (V - V_{nf})\| \leq \|V_n - V\| + \|V - V_{nf}\| \leq 2\beta^n \|V - V_0\|. \quad \square$$

Relation (2.11) means that for each s the performance of the ***stationary policy*** consisting of n copies of the decision rule f becomes arbitrarily close to the performance of each s-optimal action sequence in $A^n(s)$ when n tends to ∞.

Now we turn to *continuous* DPs. There are problems such as the freighter problem from Example 4.1.1 below where only a discrete model makes sense. On the other hand, for many problems both a discrete and a continuous version may be formulated; see the allocation problems Example 2.4.1 and Example 2.4.3 below or the linear-quadratic problems in Example 3.1.2 and Example 4.1.7 below.

Here are a few comments on the appropriateness of discrete or continuous versions and on their solutions.

(i) From a rigorous point of view continuous DPs cannot be completely realistic models for applications since they assume infinite divisibility of states and/or of actions. This does not hold in reality; e.g. in the allocation problem arbitrary small investments do not make sense.

(ii) Continuous versions are often considered as *good* approximate descriptions of a discrete model in the sense that the solution of the continuous version is a *good* approximation to the solution of the latter model. In fact this seems to be true in many cases where the actions are measured in small units, e.g. in micro seconds when the resource means time. However, the discrete version often describes the problem equally well or even better than the continuous version.

(iii) We mention some difficulties when using continuous versions according to (ii) as approximations:

  (a) The solution of the continuous version requires, except for a few cases where an explicit solution exists, a discretization of the state and/or action space. Examples of this approach in the literature often include an analysis of the *discretization error*.

  (b) However, in the literature one rarely cares about the *continuation error* made when approximating the discrete version by the continuous version. Moreover, the continuation error and the discretization error should be added.

  (c) In continuous versions one must care about the existence of s_0-optimal action sequences or maximizers, a question often connected with the question of continuity of the value functions.

In view of the preceding comments we emphasize discrete DPs, and keep the treatment of the continuous versions brief. We now treat a continuous counterpart of the discrete Example 2.4.1. The only essential difference in the assumptions is the inclusion of a deterioration/expansion factor $z \in \mathbb{R}^+$.

Example 2.4.3 (Continuous allocation with utility function u)

(a) Consider the following DP: (i) the momentary resource s, consumption a and investment $s-a$ are non-negative reals; $D(s)=[0,s]$; (ii) the resource at time $t+1$ equals $T(s_t,a_t):=z\cdot(s_t-a_t)$ for some $z\in\mathbb{R}^+$. Denote by s_0 the initial resource. We distinguish case 1: $z\le 1$ and case 2: $z>1$. As an example, if the resource consists of a perishable good, $z<1$ may be a factor for the deterioration of the investment s_t-a_t. On the other hand, $z>1$ occurs as interest factor for the investment when the resource is a capital. The maximal resource after N stages equals s_0 in case 1 and $z^N s_0$ in case 2. Therefore in case 1 we use $S=A=[0,K]$ where $K\ge s_0$, and $S=A=\mathbb{R}_+$ in case 2.

Again $r(s,a):=u(a)$ with increasing and non-negative utility function u and $V_0=d\cdot u$ for some $d\in\mathbb{R}_+$. By Theorem 2.3.3 the VI holds and it reads

$$V_n(s)=\sup\{u(a)+\beta V_{n-1}(z\cdot(s-a)):a\in[0,s]\},\ s\in S. \tag{2.12}$$

Again f_n denotes the smallest maximizer at stage n, provided it exists. Sometimes another choice of actions a' (not applicable for the discrete version) is useful: If $s>0$, then $a':=$ the proportion a/s of the momentarily available amount s of resource, which is consumed; if $s=0$, then a' may be chosen arbitrarily in $[0,K]$. The resulting DP$'$ differs from the original DP in the following data: $A'=[0,1]=D'(s)$, $T'(s,a')=s\cdot(1-a')$, $r'(s,a')=u(sa')$. Then DP$'$ has the value functions, starting with $V_0':=du$,

$$s\mapsto V_n'(s)=\sup\left\{u(sa')+\beta V_{n-1}(zs\cdot(1-a')):a'\in[0,1]\right\},\ n\ge 1, \tag{2.13}$$

which intuitively equals V_n. A formal proof uses induction on $n\ge 0$, and for fixed $s>0$ the bijective substitution $a':=a/s$ in $W_n'(s,a')$.

As seen from (2.12) and (2.13), a function h_n from S into $[0,1]$ is a maximizer at stage n in DP$'$ if and only if $s\cdot h_n(s)$ is a maximizer at stage n in DP.

We often use the following *abbreviation*: for $x\in\mathbb{R}$ and $n\in\mathbb{N}_0$ put

$$\sigma_n(x):=\sum_{t=0}^{n-1}x^t=\begin{cases}(1-x^n)/(1-x), & \text{if } x\ne 1,\\ n, & \text{if } x=1;\end{cases} \tag{2.14}$$

in particular, $\sigma_0(x)=0$ and $\sigma_1(x)=\sigma_n(0)=1$ for $n\ge 1$.

(b) Explicit solutions exist rarely, e.g. if $u(a)=\sqrt{a}$ (cf. Example 4.1.6(a)) or if $\beta z=1$ (cf. Problem 4.3.1). However, as we now indicate, for relatively general utility u the subsequent structural properties (b1)–(b7) of the solution are valid.

(b1) $V_n(s)$, $n \geq 1$, is increasing in s, non-negative and finite. Moreover,

$$V_n(s) \leq \sum_{\nu=0}^{n} \beta^\nu u(z^\nu s) + \beta^n d u(z^n s),\ s \in S, \text{ in case 1 and 2},$$
$$V_n \leq [\sigma_n(\beta) + \beta^n d]u \text{ in case 1},$$
$$V_n \leq [1/(1-\beta) + \beta^n d]u \text{ in case 1 and if } \beta < 1.$$

(b2) In case 1, if $0 \leq s \leq \overline{s} := \max\{0 \leq x \leq K : u(x) = 0\}$ and $n \geq 0$, then $V_n(s) = 0$. Moreover, each $y \in A^N(s)$ is s-optimal. These statements may be proved as in Example 2.4.1(c2), observing that $u(\overline{s}) = 0$ by continuity of u.

(b3) $V_n(s)$ is increasing in n [decreasing in n] if $d \leq 1$ [if $\beta < 1$, $d \geq 1/(1-\beta)$]. This may be proved as in Example 2.4.1(c3).

(b4) Let u be concave. Then all value functions V_n are concave, the smallest maximizers f_n, $n \geq 1$, exist and are increasing and $f_n(s') - f_n(s) \leq s' - s$ for $s \leq s'$. This follows from Example 8.2.14 below with $\eta_1 = u_2 \equiv 0$, $\eta_2(x) = zx$, and $u_1 = u$.

(b5) Let u be convex and $\beta < 1$. Then for some $m \in \mathbb{N}$ we have $V_n = u$ for all $n \geq m$, and for each s_0 the action sequence $(s_0, 0, \ldots, 0) \in A^N(s_0)$ is s_0-optimal for all $N \geq m$. This is shown in Example 7.3.5.

(b6) The value functions V_n, $n \geq 1$, are Lipschitz continuous in d and also in β, both uniformly in s. This may be proved as Example 2.4.1(c6).

(b7) As in the discrete version (see Proposition 2.4.2), in case $\beta < 1$ the sequence of value functions converges for $n \to \infty$ uniformly to some function V. However, the proof given in Theorem 10.1.10 differs from the proof of Proposition 2.4.2, and V cannot be defined recursively. ♦

2.5 Problems

Problem 2.5.1 Consider a DP where $S = A = \mathbb{N}_{0,K}$ for some $K \in \mathbb{N}$, $D(s) = \mathbb{N}_{0,s}$, $T(s,a) = s - a$, $r(s,a) = u(a)$ for some function u on A, $V_0(s) = d_0 \cdot u(s)$ for some $d_0 \in \mathbb{R}_+$, $\beta \in [0,1]$. Then for $n \geq 1$ and $s \in S$:

(a) $y = (a_t)_0^{n-1} \in A^n$ belongs to $A^n(s)$ if and only if $\sum_0^{n-1} a_t \leq s$.
(b) if $s \leq s'$ then $A^n(s) \leq A^n(s')$.
(c) if $s \leq s'$ and if u is increasing then $V_{ny}(s) \leq V_{ny}(s')$.
(d) if u is increasing, then $s \mapsto V_n(s)$ is increasing.

Problem 2.5.2 Consider a DP where $S = \mathbb{R}_+$, $A = [0,1]$, $D(s) = [0, \min\{1, s\}]$ and $T(s,a) = s - a$. Then for $n \geq 1$ and $s \in S$:

(a) $y = (a_t)_0^{n-1} \in A^n$ belongs to $A^n(s)$ if and only if $\sum_0^{n-1} a_t \leq \min\{1 + \sum_0^{n-2} a_t, s_0\}$;
(b) the properties (b)–(d) from Problem 2.5.1 remain true.

Problem 2.5.3 (Existence of an optimal action sequence without existence of maximizer) Consider the DP with $S = A = [0, 1]$; $D(s) := [0, s]$; $r(s, a) = 0$ for $s = a = 1$ and $= a/2$ else; $T(s, a) = (1 - a) \cdot s$; $V_0 \equiv 0$ and $\beta = 1$. Show that $(s, (1 - s) \cdot s)$ is the (unique) s-optimal action sequence for $DP_2(s)$, $s \in S$, and there exists no maximizer at stage 1.

2.6 Supplements

Supplement 2.6.1 (The discount factor) In economical applications β is usually smaller than one. In particular, if the length l of each period equals the k-th part of a year, if the annual interest rate equals i percent and if compound interest per period is assumed, then, since discounted rewards correspond to cash values at time zero, we have $1+i = 1/\beta^k = 1/\beta^{1/l}$, hence $\beta = \frac{1}{(1+i)^l} < 1$. Thus the larger l and/or i, the smaller β, and β approaches 1 when l tends to zero. If e.g. $i = 8\,\%$ then $\beta = 0.981$ if l is a quarter of a year and $\beta = 0.9936$ if l is one month. Moreover, if e.g. $l :=$ one hour and if N is not too large, let's say $N = 40$, then β can be practically taken equal to one.

If the N periods have nothing to do with time but mean that a certain activity is executed N times, then only $\beta = 1$ is meaningful.

The case $\beta > 1$ models the situation where the genuine discount factor equals some $\gamma < 1$ and where the one-stage reward increases from period to period (and similarly for the terminal reward) by the factor β/γ.

Supplement 2.6.2 (Changing the definition of an action) By another definition of the action in the continuous allocation problem from Example 2.4.3 one obtains three other formulations as follows, where S, V_0 and β remain unchanged.

(a) If a denotes the amount of the resource *not allocated* momentarily then $A = \mathbb{R}_+$, $D(s) = [0, s]$ for all s, $T(s, a) = z \cdot a$ and $r(s, a) := u(s - a)$ for $(s, a) \in D$.
(b) If a denotes the momentarily allocated *proportion* of the resource then $A = [0, 1] = D(s)$ for all s, $T(s, a) = zs(1 - a)$, $r(s, a) := u(s(1 - a))$ for $(s, a) \in D$.
(c) A further formulation is obtained if a denotes the momentarily not allocated *proportion* of the resource.

For some investigations the above formulations (a)–(c) have slight advantages over the formulation in Example 2.4.3.

http://www.springer.com/978-3-319-48813-4

Dynamic Optimization
Deterministic and Stochastic Models
Hinderer, K.; Rieder, U.; Stieglitz, M.
2016, XXII, 530 p. 22 illus., Softcover
ISBN: 978-3-319-48813-4