DYNAMIC PROGRAMMING ON RECURSIVE REWARD SYSTEMS

Furukawa, Nagata
Department of Mathematics, Kyushu University

Iwamoto, Seiichi
Department of Mathematics, Kyushu University

https://doi.org/10.5109/13108

出版情報：統計数理研究. 17 (1/2), pp.103-126, 1976-03. Research Association of Statistical Sciences
バージョン：
権利関係：

DYNAMIC PROGRAMMING ON RECURSIVE REWARD SYSTEMS

By

Nagata FURUKAWA* and Seiichi IWAMOTO*

(Received September 30, 1975)

1. Introduction.

Dynamic programming (DP) has been introduced by R. Bellman [2] as an important technique to solve non-linear programming problems in which a sequence of decisions has to be chosen in an optimal manner. Bellman, in his book, proposed "Principle of Optimality" to show that the determination of an optimal policy can be reduced to the solution of an optimality equation, i. e., a functional equation that should be satisfied by an optimal return. Although Principle of Optimality is a proposition which needs mathematical reasoning, his justification for the principle was not in a precise mathematical form. For this reason, the scope of cost structure to which the principle is applicable has been left unexplained.

Afterward, G. L. Nemhauser [9] gave a sufficient condition for the cost structure in order that an optimality equation holds true. His condition is that the cost function should have both a separability property and a monotonicity property. Nemhauser did not make explicit the relation between the effectiveness of Bellman's principle and the justification for an optimality equation — the relation is no more trivial under his condition.

In this paper we shall be concerned with the optimization of finite-stage sequential decision processes. We shall give rigorous proofs for the justification of optimality equations and for the effectiveness of optimality principles with two meanings, without assuming the existence of maximum values of returns. Our condition is that the cost function should have a recursiveness property, a monotonicity property and a Lipschitz condition. Our recursiveness is essentially same as the separability in Nemhauser sense. Our monotonicity has two senses: one is a wide sense, and the other a strict sense. The monotonicity properties in the wide and the strict senses, together with the recursiveness and the Lipschitz condition, induce optimality principles in a weak and a strong senses, respectively. Bellman's Principle of Optimality is well to be identified, in our terms, a principle in the strong sense. Our principle in the weak sense has not been introduced in other literatures as far as the authors know. If we assume the existence of maximum values of returns like Nemhauser did, then the Lipschitz condition can be suppressed from hypotheses in our arguments.

In this paper we treat both deterministic and stochastic cases. Section 2 is

* Department of Mathematics, Kyushu University, Fukuoka.

devoted to the former case, and Section 3 to the later case. In Section 3, we anew introduce a stochastic recursiveness and a stochastic strict monotonicity, and develope arguments parallel to those in deterministic case.

Section 4 classifies DP problems from the viewpoint of the reward system (RS). The family of RSs considered in this paper, in both deterministic and stochastic cases, explicitly covers all types of returns discussed by Nemhauser. Our RSs are, for instance, recursive additive, multiplicative additive, divided additive, logarithmic additive, and exponential additive. Especially in deterministic case, backward power, forward power, maximum and minimum RSs are added to those cited above.

The last section sets forth a variety of examples in which optimal policies and optimal returns are given.

2. Deterministic case.

2.1. Notation and definitions.

In the deterministic case a dynamic programming problem is specified by five elements S, $\{A_n\}$, $\{T_n\}$, $\{g_{mn}\}$ and d. The set of *states* S of some system is a nonempty Borel subset of the J-dimensional Euclidean space R^J. For each n $(1 \leqq n \leqq N)$, A_n is a mapping from S to the space of nonempty Borel subsets of R^K. For $s \in S$, $A_n(s)$ means the set of *actions* available to us at the state s in the n-th stage. Let Γ_n denote the set $\{(s, a) \mid a \in A_n(s), s \in S\}$. For each n $(1 \leqq n \leqq N)$, T_n is a Borel measurable mapping from Γ_n to S. $\{T_n\}$ means the *transition law* of the system: if we choose an action $a \in A_n(s)$ after observing the state s of the system in the n-th stage, then the system moves to a new state s' uniquely determined by $s' = T_n(s, a)$. For each m and n $(1 \leqq m \leqq n \leqq N+1)$, g_{mn} is a real-valued Borel measurable function defined on $\Gamma_m \times \Gamma_{m+1} \times \cdots \times \Gamma_{n-1} \times S \times R^1$. In the case when $m=n$, here, the set $\Gamma_m \times \Gamma_{m+1} \times \cdots \times \Gamma_{n-1} \times S \times R^1$ is interpreted as $S \times R^1$. We assume, for every n, g_{nn} satisfies the relation that $g_{nn}(s; c) = c$ for all $s \in S$ and all $c \in R^1$. The *terminal reward function* d is a real-valued Borel measurable function defined on S. We shall call the set of functions $\{g_{mn}\}$ and d a *reward system* (RS). If starting from an initial state s_1 we observe a sequence of states $s_1, s_2, \cdots, s_N$ and terminate at s_{N+1} by choosing actions $a_1, a_2, \cdots, a_N$ in turn, then we receive a total reward $g_{1,N+1}$ $(s_1, a_1, s_2, a_2, \cdots, a_N, s_{N+1}; d(s_{N+1}))$. Given the number N, our problem is then to maximize the total reward. The maximizing problem is called the *N-stage problem*. We shall call $g_{1,N+1}$ $(s_1, a_1, s_2, a_2, \cdots, s_{N+1}; d(s_{N+1}))$ an *objective function* (OF).

Let H_n denote the set $\Gamma_1 \times \Gamma_2 \times \cdots \times \Gamma_{n-1} \times S$. H_n means the space of partial histories $(s_1, a_1, s_2, a_2, \cdots, s_n)$ consisting of $(2n-1)$ elements. An element of H_n is denoted by h_n. A finite sequence $\pi = \{\pi_1, \pi_2, \cdots, \pi_N\}$ is called a *deterministic policy* for the N-stage problem, if π_n is a Borel measurable mapping from H_n to R^K for each n, and if for each n, $\pi_n(h_n) \in A_n(s_n)$ for all $h_n \in H_n$ where s_n is the last component of h_n. Suppose, throughout Section 2, a policy means a deterministic policy unless otherwise stated. A policy $\pi = \{\pi_1, \pi_2, \cdots, \pi_N\}$ is called *Markov*, if for each n π_n depends only on the last component of h_n. A Markov policy is denoted by $\{f_1, f_2, \cdots, f_N\}$. Let $\Pi^N_{\mathfrak{D}}$ and $\Pi^N_{\mathfrak{M}}$ denote the set of all policies and the set of all Markov policies, respectively, for the N-stage problem. For a policy $\pi=\{\pi_1, \pi_2, \cdots, \pi_N\}$, let ${}^n\pi$ denote the final subsequence $\{\pi_{n+1}, \pi_{n+2}, \cdots, \pi_N\}$. For each n $(0\leqq n\leqq N-1)$ let

$$I^{N-n}({}^n\pi)(h_{n+1})=g_{n+1,N+1}(s_{n+1}, a^0_{n+1}, s^0_{n+2}, \cdots, a^0_N, s^0_{N+1}; d(s^0_{N+1})), \tag{2.1}$$

where

$$a^0_j=\pi_j(h_j) \qquad j=n+1, n+2, \cdots, N,$$

$$s^0_k=T_{k-1}(s^0_{k-1}, a^0_{k-1}) \qquad k=n+2, n+3, \cdots, N+1,$$

$$s^0_{n+1}=s_{n+1}.$$

In the definition (2.1) the upper suffix of I means the number of stages left for us to take actions. For a Markov policy $\pi=\{f_1, f_2, \cdots, f_N\}$, (2.1) is interpreted as

$$\begin{aligned} I^{N-n}(f_{n+1}, f_{n+2}, \cdots, f_N)(s_{n+1}) \\ =g_{n+1,N+1}(s_{n+1}, a^0_{n+1}, s^0_{n+2}, \cdots, a^0_N, s^0_{N+1}; d(s^0_{N+1})), \end{aligned} \tag{2.2}$$

where

$$a^0_j=f_j(s^0_j) \qquad j=n+1, n+2, \cdots, N,$$

$$s^0_k=T_{k-1}(s^0_{k-1}, a^0_{k-1}) \qquad k=n+2, n+3, \cdots, N+1,$$

$$s^0_{n+1}=s_{n+1}.$$

Substituting $n=0$ into (2.1) we have

$$I^N(\pi)(s_1)=g_{1,N+1}(s_1, a^0_1, s^0_2, a^0_2, \cdots, s^0_{N+1}; d(s^0_{N+1})), \tag{2.3}$$

that is what we want to maximize with respect to π in the set $\Pi^N_{\mathfrak{D}}$. A policy π^* for the N-stage problem is said to be *optimal*, if $I^N(\pi^*)(s)\geqq I^N(\pi)(s)$ for all $s\in S$ and all $\pi\in\Pi^N_{\mathfrak{D}}$.

2.2. Recurrence relations and the principle of optimality.

At first we shall set up two general assumptions which are valid for this subsection.

(G1). For each n $(1\leqq n\leqq N)$, Γ_n has a Borel Selector, i.e., there exists a Borel mapping S to R^K such that graph $\phi_n\subset\Gamma_n$.

Let

$$L_n(s)=\sup_{(a_n, s_{n+1}, \cdots, s_{N+1})\in A_n(s)\times\Gamma_{n+1}\times\cdots\times\Gamma_N\times S} |g_{n,N+1}(s, a_n, s_{n+1}, \cdots, s_{N+1}; d(s_{N+1}))|. \tag{2.4}$$

(G2). For each n $(1\leqq n\leqq N+1)$ and each s, $L_n(s)$ is finite.

Many authors have worked on the existence of a Borel Selector. We like to present here a sufficient condition for the existence of a Borel Selector which is a slight modification of the result by Arsenin and Ljapunov [1].

PROPOSITION 2.1. *Let $X=R^J$ and $Y=R^K$. Let Γ be a Borel subset of $X\times Y$ such that for each $x\in X$ the x-section of Γ is σ-compact in Y. Then* $\mathrm{proj}_X\Gamma$ *is Borel and*

Γ has a Borel Selector defined on $\mathrm{proj}_X\Gamma$.

We now prepare two lemmas.

LEMMA 2.1. *For any $\pi\in\Pi^N_{\mathscr{D}}$ and any $s_1\in S$, there exists a Markov policy $\hat{\pi}$ such that*

$$I^N(\pi)(s_1)=I^N(\hat{\pi})(s_1).$$

PROOF. The proof is easy by the use of Borel Selectors $\{\phi_n\}$ assured in the assumption (G1).

LEMMA 2.2. *For any $s_1\in S$ and any $\varepsilon>0$, there exists a Markov policy $\hat{\pi}$ such that*

$$I^N(\hat{\pi})(s_1)\geqq \sup_{\pi\in\Pi^N_{\mathscr{D}}} I^N(\pi)(s_1)-\varepsilon.$$

PROOF. The proof is easy from Lemma 2.1.

For each n, let

$$^{N-n}\Pi^N_{\mathscr{D}}=\{^{N-n}\pi\,|\,\pi\in\Pi^N_{\mathscr{D}}\}.$$

That is, $^{N-n}\Pi^N_{\mathscr{D}}$ means the set of all final subpolicies with n components. And let

$$u^n(h_{N-n+1})=\sup_{^{N-n}\pi\in{}^{N-n}\Pi^N_{\mathscr{D}}} I^n(^{N-n}\pi), \tag{2.5}$$

where $I^n(^{N-n}\pi)$ is given by (2.1). Then we have the following

THEOREM 2.1. *Let*

$$v^n(s_{N-n+1})=\sup_{\{f_1,f_2,\cdots,f_n\}\in\Pi^n_{\mathscr{M}}} I^n(\{f_1,f_2,\cdots,f_n\})(s_{N-n+1}) \tag{2.6}$$

for $n=0, 1, \cdots, N$, where $I^n(\{f_1,f_2,\cdots,f_n\})$ is given by (2.2). *Then it holds that*

$$u^n(h_{N-n+1})=v^n(s_{N-n+1}) \quad \textit{for all} \quad h_{N-n+1}\in H_{N-n+1}, \tag{2.7}$$

where s_{N-n+1} in the right-hand side is the last component of h_{N-n+1} in the left-hand side.

PROOF. It suffices to show the equality (2.7) in the case when $n=N-1$, for we can show it similarly in general case.

Let

$$\bar{u}^{N-1}(s_2)=\sup_{\pi\in\Pi^{N-1}_{\mathscr{D}}} I^{N-1}(\pi)(s_2), \tag{2.8}$$

then it follows directly from the definitions of u^{N-1} and $\bar{u}^{N-1}$ that

$$u^{N-1}(h_2)\geqq\bar{u}^{N-1}(s_2) \quad \text{for all} \quad h_2\in H_2, \tag{2.9}$$

where s_2 is the last component of h_2 in the left-hand side. For $h_n=(s_1, a_1, s_2, \cdots, s_n)$, let ^1h_n denote $(s_2, a_2, \cdots, s_n)$. Let $(s_1, a_1)\in\Gamma_1$ and $\pi=\{\pi_1, \pi_2, \cdots, \pi_N\}\in\Pi^N_{\mathscr{D}}$ fix arbitrarily. By considering $\pi_n(h_n)$ as the function of ^1h_n since (s_1, a_1) is fixed, define π'_n by

$$\pi'_n(^1h_n)=\pi_n(s_1, a_1, {}^1h_n) \quad \text{for all} \quad {}^1h_n,$$
$$\text{for} \quad n=2, 3, \cdots, N.$$

Let $\pi'=\{\pi'_2, \pi'_3, \cdots, \pi'_N\}$, then π' becomes an element of $\Pi^{N-1}_{\mathscr{D}}$. Since $^1\pi$ starting from

(s_1, a_1, s_2) yields the expected reward as much as π' starting from s_2, we have

$$I^{N-1}({}^1\pi)(s_1, a_1, s_2) \leqq \sup_{\tilde{\pi}\in\Pi_{\mathscr{D}}^{N-1}} I^{N-1}(\tilde{\pi})(s_2).$$

Then we have

$$u^{N-1}(s_1, a_1, s_2) \leqq \bar{u}^{N-1}(s_2).$$

This holds for all $s_2 \in S$. Since (s_1, a_1) is arbitrary, we have

$$(2.10) \qquad u^{N-1}(h_2) \leqq \bar{u}^{N-1}(s_2) \qquad \text{for} \quad h_2 \in H_2.$$

It follows from (2.9) and (2.10) that

$$u^{N-1}(h_2) = \bar{u}^{N-1}(s_2) \qquad \text{for} \quad h_2 \in H_2,$$

which implies that $u^{N-1}(h_2)$ is independent of (s_1, a_1).

We now fix s_2 arbitrarily. By applying Lemma 2.2 to the $(N-1)$-stage problem starting from s_2, we have

$$v^{N-1}(s_2) \geqq \bar{u}^{N-1}(s_2) - \varepsilon.$$

Letting $\varepsilon \downarrow 0$, we have

$$v^{N-1}(s_2) \geqq \bar{u}^{N-1}(s_2).$$

Since the converse inequality is trivially true, we get

$$v^{N-1}(s_2) = \bar{u}^{N-1}(s_2) = u^{N-1}(h_2) \qquad \text{for} \quad h_2 \in H_2,$$

which completes the proof.

We now introduce two properties of $\{g_{mn}\}$ which play a crucial role leading us to "the principle of optimality".

DEFINITION 2.1. $\{g_{mn}\}$ is said to be *recursive*, if for each m and n such that $m \leqq n-2$, all $(s_m, a_m, \cdots, s_n) \in \Gamma_m \times \Gamma_{m+1} \times \cdots \times \Gamma_{n-1} \times S$ and all $c \in R^1$, it holds that

$$g_{mn}(s_m, a_m, s_{m+1}, a_{m+1}, \cdots, s_n; c)$$
$$= g_{m,m+1}(s_m, a_m, s_{m+1}; g_{m+1,n}(s_{m+1}, a_{m+1}, \cdots, s_n; c)).$$

DEFINITION 2.2. $\{g_{mn}\}$ is said to be *monotone* (*strictly monotone*), if $c_1 < c_2$ implies $g_{n,n+1}(s, a, s'; c_1) \leqq (<) g_{n,n+1}(s, a, s'; c_2)$ for all n and all $(s, a, s') \in \Gamma_n \times S$.

The following proposition is immediate from Definition 2.2.

PROPOSITION. 2.2. *If* $\{g_{mn}\}$ *is recursive and monotone (strictly monotone), then* $c_1 < c_2$ *implies* $g_{mn}(s_m, a_m, \cdots, s_n; c_1) \leqq (<) g_{mn}(s_m, a_m, \cdots, s_n; c_2)$ *for each* m, n *such that* $m \leqq n-2$ *and all* $(s_m, a_m, \cdots, s_n) \in \Gamma_m \times \Gamma_{m+1} \times \cdots \times \Gamma_{n-1} \times S$.

We shall make use of the following condition which means a Lipschitz condition in a weak sense.

CONDITION (L). For each $n (1 \leqq n \leqq N)$ there exists a positive number K_n such that

$$g_{n,n+1}(s, a, s'; c+\varepsilon) - g_{n,n+1}(s, a, s'; c) \leqq K_n \varepsilon$$

for all $(s, a, s') \in \Gamma_n \times S$, all $c \in R^1$ and for sufficiently small $\varepsilon > 0$.

In Theorem 2.1 we have proved that $u^n(h_{N-n+1})$ depends only on the last component of h_{N-n+1}. Hence we shall write $u^n(s_{N-n+1})$ or, simply, $u^n(s)$ in place of

$u^n(h_{N-n+1})$, hereafter. The spirit of the following theorem can be found in Theorem 1 of Karp and Held [8].

THEOREM 2.2. *Let* $\{g_{mn}\}$ *be recursive and monotone. Let Condition* (L) *be satisfied. Then* $\{u_n\}$ *satisfies the recurrence relations:*

$$\begin{aligned} &u^n(s)=\sup_{a\in A_{N-n+1}(s)} g_{N-n+1,N-n+2}(s,\ a,\ T_{N-n+1}(s,\ a)\ ;\ u^{n-1}(T_{N-n+1}(s,\ a))) \\ (2.11)\qquad &\qquad\qquad\qquad\qquad\qquad \textit{for}\quad s\in S,\ n=1,\ 2,\ \cdots,\ N, \\ &u^0\equiv d. \end{aligned}$$

PROOF. It suffices to show the relation in the case when $n=N$, for we can show it similarly in general case.

From Theorem 2.1 we have

$$\begin{aligned} u^N(s)&=\sup_{\{f_1,f_2,\cdots,f_N\}\in\Pi^N_{\mathcal{M}}} I^N(\{f_1,f_2,\cdots,f_N\})(s) \\ &=\sup_{\{f_1,f_2,\cdots,f_N\}\in\Pi^N_{\mathcal{M}}} g_{1,N+1}(s,\ a_1^0,\ s_2^0,\ a_2^0,\ \cdots,\ s_{N+1}^0\ ;\ d(s_{N+1}^0)), \end{aligned}$$

where

$$\begin{aligned} &a_j^0=f_j(s_j^0) \qquad j=1,\ 2,\ \cdots,\ N, \\ &s_k^0=T_{k-1}(s_{k-1}^0,\ a_{k-1}^0) \qquad k=2,\ 3,\ \cdots,\ N+1, \\ &s_1^0=s. \end{aligned}$$

Hence from the recursiveness and monotonicity we have

$$\begin{aligned} u^N(s)&=\sup_{\{f_1,f_2,\cdots,f_N\}\in\Pi^N_{\mathcal{M}}} g_{1,2}(s,\ a_1^0,\ s_2^0\ ;\ g_{2,N+1}(s_2^0,\ a_2^0,\ \cdots,\ s_{N+1}^0\ ;\ d(s_{N+1}^0))) \\ (2.12)\qquad &\leqq \sup_{f_1\in\Pi^1_{\mathcal{M}}} g_{1,2}(s,\ a_1^0,\ s_2^0\ ;\ u^{N-1}(s_2^0)) \\ &=\sup_{a\in A_1(s)} g_{1,2}(s,\ a,\ T_1(s,\ a)\ ;\ u^{N-1}(T_1(s,\ a))). \end{aligned}$$

Next, take $a\in A_1(s)$ arbitrarily. Let $\varepsilon>0$ be sufficiently small. Applying Lemma 2.2 to the $(N-1)$-stage problem starting from $\hat{s}=T_1(s,\ a)$, we find a Markov policy $\pi^*=\{f_2^*,\ f_3^*,\ \cdots,\ f_N^*\}\in\Pi^{N-1}_{\mathcal{M}}$ such that

$$u^{N-1}(\hat{s})\leqq I^{N-1}(\pi^*)(\hat{s})+\varepsilon,$$

which implies from the monotonicity that

$$g_{1,2}(s,\ a,\ \hat{s}\ ;\ u^{N-1}(\hat{s}))\leqq g_{1,2}(s,\ a,\ \hat{s}\ ;\ I^{N-1}(\pi^*)(\hat{s})+\varepsilon).$$

By making use of Condition (L) and the recursiveness in the right-hand side, we get

$$g_{1,2}(s,\ a,\ \hat{s}\ ;\ u^{N-1}(\hat{s}))\leqq u^N(s)+K_1\varepsilon.$$

Letting $\varepsilon\downarrow 0$, we have

$$(2.13)\qquad \sup_{a\in A_1(s)} g_{1,2}(s,\ a,\ T_1(s,\ a)\ ;\ u^{N-1}(T_1(s,\ a)))\leqq u^N(s).$$

From (2.12) and (2.13) finally we get

$$u^N(s)=\sup_{a\in A_1(s)} g_{1,2}(s, a, T_1(s, a); u^{N-1}(T_1(s, a))).$$

This completes the proof.

THEOREM 2.3 (*Weak Principle of Optimality*). *Let* $\{g_{mn}\}$ *be same as in Theorem 2.2, and let Condition* (L) *be satisfied. Then the optimal policy* $\pi^*=\{\pi_1^*, \pi_2^*, \cdots, \pi_N^*\}$ *for the N-stage problem satisfies the relations:*

$$\begin{aligned} (2.14)\qquad & g_{1,N-n+1}(s_1, a_1^*, s_2^*, a_2^*, \cdots, s_{N-n+1}^*; I^n({}^{N-n}\pi^*)(h_{N-n+1}^*)) \\ &=g_{1,N-n+1}(s_1, a_1^*, s_2^*, a_2^*, \cdots, s_{N-n+1}^*; u^n(s_{N-n+1}^*)), \\ & \qquad \textit{for} \quad s_1\in S \textit{ and } n=1, 2, \cdots, N-1, \end{aligned}$$

where

$$\begin{aligned} & a_j^*=\pi_j^*(h_j^*) \qquad j=1, 2, \cdots, N, \\ & s_k^*=T_{k-1}(s_{k-1}^*, a_{k-1}^*) \qquad k=2, 3, \cdots, N+1, \\ & s_1^*=s_1. \end{aligned}$$

PROOF. Since π^* is optimal, we have

$$\begin{aligned} (2.15)\qquad u^N(s)&=I^N(\pi^*) \\ &=g_{1,2}(s, a_1^*, s_2^*; I^{N-1}({}^1\pi^*)(h_2^*)) \qquad \text{for} \quad s\in S. \end{aligned}$$

By using the monotonicity in the right-hand side of (2.15), we get

$$\begin{aligned} u^N(s)&\leqq g_{1,2}(s, a_1^*, s_2^*; u^{N-1}(s_2^*)) \\ (2.16)\qquad &\leqq \sup_{a\in A_1(s)} g_{1,2}(s, a, T_1(s, a); u^{N-1}(T_1(s, a))) \\ &=u^N(s), \end{aligned}$$

where the last equality is due to Theorem 2.2. Combining (2.15) and (2.16) we have

$$g_{1,2}(s, a_1^*, s_2^*; I^{N-1}({}^1\pi^*)(h_2^*))=g_{1,2}(s, a_1^*, s_2^*; u^{N-1}(s_2^*)).$$

Thus it has been shown that (2.14) is true in the case when $n=N-1$.

For general case we can prove it in the manner similar to the case when $n=N-1$ by the repeated use of the recursiveness.

It should be noted that the assertion of Theorem 2.3 is very similar to Bellman's principle of optimality, but somewhat weaker than Bellman's one. Namely, the relations (2.14) follow from Bellman's principle; on the contrary, the converse may not be true. The following theorem shows that Bellman's principle can be obtained from our Weak Principle by replacing the monotonicity property with the strict monotonicity property.

THEOREM 2.4 (*Principle of Optimality*). *Let* $\{g_{mn}\}$ *be recursive and strictly monotone, and let Condition* (L) *be satisfied. Let* $\pi^*=\{\pi_1^*, \pi_2^*, \cdots, \pi_N^*\}$ *be optimal. Then we have*

$$(2.17)\qquad u^n(s_{N-n+1}^*)=I^n({}^{N-n}\pi^*)(s_{N-n+1}^*) \qquad \textit{for} \quad n=1, 2, \cdots, N \textit{ and } s_1\in S,$$

where

$$a_j^* = \pi_j^*(h_j^*) \qquad j=1, 2, \cdots, N,$$

$$s_k^* = T_{k-1}(s_{k-1}^*, a_{k-1}^*) \qquad k=2, 3, \cdots, N+1,$$

$$s_1^* = s_1.$$

PROOF. The theorem follows directly from Theorem 2.3 by virtue of the assumption that $\{g_{mn}\}$ is strictly monotone.

COROLLARY 2.1. *Let the same assumptions as in Theorem 2.4 be placed. Then there exists a Markov policy which is equivalent to the optimal policy.*

PROOF. Trivial from (2.17).

THEOREM 2.5. *Let the same assumptions as in Theorem 2.2 be placed. Suppose that we can construct a Markov policy $\pi^* = \{f_1^*, f_2^*, \cdots, f_N^*\}$ in order that the following relations be satisfied.*

$$\begin{aligned} &g_{n,n+1}(s, f_n^*(s), T_n(s, f_n^*(s)); u^{N-n}(T_n(s, f_n^*(s)))) \\ &\qquad = \sup_{a \in A_n(s)} g_{n,n+1}(s, a, T_n(s, a); u^{N-n}(T_n(s, a))) \\ &\qquad\qquad\qquad \textit{for} \quad s \in S \textit{ and } n=1, 2, \cdots, N. \end{aligned} \tag{2.18}$$

Then π^* is optimal.

PROOF. From Theorem 2.2 and (2.18) we have

$$\begin{aligned} &g_{n,n+1}(s, f_n^*(s), T_n(s, f_n^*(s)); u^{N-n}(T_n(s, f_n^*(s)))) = u^{N-n+1}(s) \\ &\qquad\qquad\qquad \text{for} \quad s \in S \text{ and } n=1, 2, \cdots, N. \end{aligned} \tag{2.19}$$

For any fixed s_{N-1} we have, from the recursiveness and the relations (2.19) with $n=N$ and $n=N-1$ substituted, that

$$\begin{aligned} &g_{N-1,N+1}(s_{N-1}, f_{N-1}^*(s_{N-1}), s_N^0, f_N^*(s_N^0), T_N(s_N^0, f_N^*(s_N^0)); d(T_N(s_N^0, f_N^*(s_N^0)))) \\ &\qquad = g_{N-1,N}(s_{N-1}, f_{N-1}^*(s_{N-1}), s_N^0; u^1(s_N^0)) \\ &\qquad = u^2(s_{N-1}), \end{aligned} \tag{2.20}$$

where $s_N^0 = T_{N-1}(s_{N-1}, f_{N-1}^*(s_{N-1}))$. Since s_{N-1} is arbitrary, (2.20) holds for all $s_{N-1} \in S$. Inductively we have

$$\begin{aligned} I^N(\pi^*)(s_1) &= g_{1,N+1}(s_1, f_1^*(s_1), s_2^*, f_2^*(s_2), \cdots, s_{N+1}^*; d(s_{N+1}^*)) \\ &= u^N(s_1) \qquad \text{for} \quad s_1 \in S, \end{aligned}$$

where $s_n^* = T_{n-1}(s_{n-1}^*, f_{n-1}^*(s_{n-1}^*))$ for $n=2, 3, \cdots, N$, and $s_1^* = s_1$. Thus π^* is optimal. This completes the proof.

REMARK 1. In the case when $L_n(s)$ is bounded on S for each n, in Theorems 2.2~2.5 Condition (L) can be slightly weakened to the following condition:

CONDITION (L′). For each n $(1 \leqq n \leqq N)$ there exists a positive number K_n such that

$$g_{n,n+1}(s, a, s'; c+\varepsilon) - g_{n,n+1}(s, a, s'; c) \leqq K_n \varepsilon$$

for all $(s, a, s') \in \Gamma_n \times S$, all c such that $|c| \leqq M$ and for sufficiently small $\varepsilon > 0$, where

$M=\sup_{\substack{1\leqq n\leqq N+1\\ s\in S}}|L_n(s)|$.

REMARK 2. In the practical problems of dynamic programming, it very often happens to us that for each n and s, $u^n(s)$ is attained by the maximum rather than by the supremum, that is, for each n and s, there exists an optimal policy, which may depend on s, for the n-stage problem starting from s. In those cases Condition (L) can be suppressed from the hypotheses in each theorem.

3. Stochastic case.

3.1. Nontation and definitions.

A stochastic dynamic programming problem is defined in the manner parallel to the deterministic case. Namely, a dynamic programming problem in the stochastic case is specified by five elements S, $\{A_n\}$, $\{q_n\}$, $\{g_{mn}\}$ and d. Every element is same as in the deterministic case except for $\{q_n\}$. For each n, q_n is a regular conditional probability measure on S given $S\times R^K$. $\{q_n\}$ means the *stochastic transition law* of the system: when the system is in state s at the n-th stage and we take action a, the system moves to a new state s' selected according to the probability law $q_n(\cdot\,|\,s, a)$.

A finite sequence $\pi=\{\pi_1, \pi_2, \cdots, \pi_N\}$ is called a *random policy* for the N-stage problem, if π_n, for each n, is a regular conditional probability measure on R^K given H_n such that

$$\pi_n(A_n(s_n)\,|\,h_n)=1 \qquad \text{for all} \quad h_n\in H_n\,,$$

where s_n is the last component of h_n. Suppose, throughout Section 3, a policy means a random policy unless otherwise stated. Let $\Pi^N_{\mathscr{R}}$ denote the set of all (random) policies for the N-stage problem. A Markov policy and the set $\Pi^N_{\mathscr{M}}$ are defined in the same way as in the deterministic case. For any policy $\pi=\{\pi_1, \pi_2, \cdots, \pi_N\}$, ${}^n\pi$ denotes the final subsequence $\{\pi_{n+1}, \pi_{n+2}, \cdots, \pi_N\}$ as stated in Section 2.1. Let $E^{{}^n\pi}(h_{n+1})$ denote the integral operator with respect to the probability measure induced from ${}^n\pi$ and $\{q_{n+1}, \cdots, q_N\}$ over the space of $(a_{n+1}, s_{n+2}, \cdots, s_{N+1})$ given h_{n+1}. Viewing $E^{{}^n\pi}(h_{n+1})$ as a function of h_{n+1}, we shall denote it by $E^{{}^n\pi}$. For each n $(0\leqq n\leqq N-1)$, let

$$(3.1)\qquad \mathscr{J}^{N-n}({}^n\pi)(h_{n+1})=E^{{}^n\pi}(h_{n+1})[g_{n+1,N+1}(s_{n+1}, a_{n+1}, s_{n+2}, \cdots, s_{N+1}\,;\, d(s_{N+1}))]\,,$$

or as a function of h_{n+1} let

$$(3.2)\qquad \mathscr{J}^{N-n}({}^n\pi)=E^{{}^n\pi}[g_{n+1,N+1}(s_{n+1}, a_{n+1}, s_{n+2}, \cdots, s_{N+1}\,;\, d(s_{N+1}))]\,.$$

Substituting $n=0$ into (3.1) we have

$$(3.3)\qquad \mathscr{J}^{N}(\pi)(s_1)=E^{\pi}(s_1)[g_{1,N+1}(s_1, a_1, s_2, \cdots, s_{N+1}\,;\, d(s_{N+1}))]\,.$$

Our optimizing problem is to maximize $\mathscr{J}^N(\pi)(s_1)$ with respect to π in the set $\Pi^N_{\mathscr{R}}$. A policy $\pi^*\in\Pi^N_{\mathscr{R}}$ is said to be *optimal*, if $\mathscr{J}^N(\pi^*)(s)\geqq\mathscr{J}^N(\pi)(s)$ for all $s\in S$ and all $\pi\in\Pi^N_{\mathscr{R}}$.

3.2. Recurrence relations and the principle of optimality.

In addition to the general assumption (G1) we shall impose the following assumptions throughout this subsection.

(G3). d is bounded on S.

(G4). For each n $(1\leqq n\leqq N)$ and for every bounded function w on S, $g_{n,n+1}(s, a, s'; w(s'))$ is bounded on $\Gamma_n\times S$. Note that (G2) is taken its place by the joint use of (G3) and (G4).

Next we shall give stochastic versions of the recursiveness property and of the strict monotonicity property.

DEFINITION 3.1. $\{g_{mn}\}$ is said to be *stochastically recursive*, if for any $\pi\in\Pi^N_{\mathfrak{R}}$, any m, n such that $m\leqq n-2$ and any bounded Borel measurable function w on H_n, it holds that

$$E^{m\pi}[g_{m,n}(s_m, a_m, s_{m+1}, \cdots, s_n; w(h_n))]$$
$$=g_{m,m+1}(s_m, a_m, s_{m+1}; E^{m\pi}[g_{m+1,n}(s_{m+1}, a_{m+1}, \cdots, s_n; w(h_n))])\quad \text{a.s. } (P^\pi),$$

where P^π denotes the probability measure induced from π.

Let $H^{s_1}_n$ be the s_1-section of H_n for $s_1\in S$. Let $e_\pi(s_1)$ denote the probability measure on $H^{s_1}_n$ induced from π and $\{q_n\}$.

DEFINITION 3.2. $\{g_{mn}\}$ is said to be *stochastically strictly monotone*, if it is monetone in the sense of definition 2.2, and if for any n $(1\leqq n\leqq N)$, any $\pi\in\Pi^N_{\mathfrak{R}}$ and any $s_1\in S$ it holds that if there exist a Borel set $B\subset H^{s_1}_{n-1}$ and Borel functions u, v on H_{n+1} such that $u<v$ on B and $e_\pi(s_1)(B)>0$, then we have

$$\int_B g_{n,n+1}(s_n, a_n, s_{n+1}; u(h_{n+1}))de_\pi(s_1)$$
$$<\int_B g_{n,n+1}(s_n, a_n, s_{n+1}; v(h_{n+1}))de_\pi(s_1).$$

In this subsection we shall frequently use "monotone" and Condition (L) which have been both given in Section 2.2, besides "stochastically recursive" and "stochastically strictly monotone" just given above.

LEMMA 3.1. *Let $\{g_{mn}\}$ be stochastically recursive and monotone in the sense of Definition 2.2. Let Condition (L) be satisfied. Then for any $s_1\in S$, any $\varepsilon>0$ and any $\pi\in\Pi^N_{\mathfrak{R}}$, there exists a Markov policy $\hat{\pi}$ such that $\mathcal{J}^N(\hat{\pi})(s_1)\geqq\mathcal{J}^N(\pi)(s_1)-\varepsilon$.*

PROOF. We can show the lemma in the same way as in authors' previous theorem (Theorem 5.1 in [3] and its correction [4]), and so the details are omitted.

For each n, let

$$^{N-n}\Pi^N_{\mathfrak{R}}=\{^{N-n}\pi\,|\,\pi\in\Pi^N_{\mathfrak{R}}\}.$$

Define the conditional expectation of reward in parallel with (2.5) by the following:

$$U^n(h_{N-n+1})=\sup_{^{N-n}\pi\in{}^{N-n}\Pi^N_{\mathfrak{R}}}\mathcal{J}^N(^{N-n}\pi).\tag{3.4}$$

Then we have

THEOREM 3.1. *Let*

$$V^n(s_{N-n+1})=\sup_{\{f_1,f_2,\cdots,f_n\}\in\Pi^n_{\mathfrak{M}}}\mathcal{J}^n(\{f_1, f_2, \cdots, f_n\})(s_{N-n+1})\tag{3.5}$$

for $n=0, 1, \cdots, N$. Then under the same assumptions as in Lemma 3.1 it holds that

$$U^n(h_{N-n+1}) = V^n(s_{N-n+1}) \qquad \textit{for} \quad h_{N-n+1} \in H_{N-n+1}, \tag{3.6}$$

where s_{N-n+1} *is the last component of* h_{N-n+1}.

PROOF. Follow the same line as in Theorem 2.1 by the use of Lemma 3.1.

By virtue of the above theorem we shall anew express $U^n(h_{N-n+1})$ by $U^n(s_{N-n+1})$ or by $U^n(s)$ for short, hereafter.

The following lemma is a slight modification of Theorem 7.1 of Strauch [10].

LEMMA 3.2. *For each* n, U^n *is universally measurable.*

Gościński and Jakubowski gave, in Theorem of [5], a stochastic version of the recurrence relations given by Karp and Held [8]. The following theorem is a refinement of the result by the former authors, but is not written in terms of stochastic automaton as they did.

THEOREM 3.2. *Let* $\{g_{mn}\}$ *be stochastically recursive and monotone in the sense of Definition* 2.2. *Let Condition* (L) *be satisfied. Then the following recurrence relations hold:*

$$\begin{aligned} &U^n(s) = \sup_{a \in A_{N-n+1}(s)} \int_S g_{N-n+1,N-n+2}(s, a, s'; U^{n-1}(s'))dq_{N-n+1}(s' \mid s, a) \\ &\qquad\qquad\qquad\qquad \textit{for} \quad s \in S \textit{ and } n = 1, 2, \cdots, N. \\ &U^0 \equiv d, \end{aligned} \tag{3.7}$$

or symbolically

$$\begin{aligned} &U^n(s) = \sup_{a \in A_{N-n+1}(s)} E^a[g_{N-n+1,N-N+2}(s, a, s'; U^{n-1}(s'))] \\ &\qquad\qquad\qquad\qquad \textit{for} \quad s \in S \textit{ and } n = 1, 2, \cdots, N, \\ &U^0 \equiv d. \end{aligned} \tag{3.8}$$

REMARK. It should be noted that the integrals in (3.7) and (3.8) are well defined by virtue of Lemma 3.2 and the measurability of g_{mn}.

PROOF. Since $U^n = V^n$ for each n from Theorem 3.1, we have

$$\begin{aligned} U^N(s_1) &= \sup_{\{f_1, f_2, \cdots, f_N\} \in \Pi^N_{\mathfrak{M}}} E^{\{f_1, f_2, \cdots, f_N\}}[g_{1,N+1}(s_1, a_1, s_2, \cdots, s_{N+1}; d(s_{N+1}))] \\ &= \sup_{\{f_1, f_2, \cdots, f_N\} \in \Pi^N_{\mathfrak{M}}} E^{f_1}[g_{1,2}(s_1, a_1, s_2; E^{\{f_2, \cdots, f_N\}}[g_{2,N+1}](s_2))] \\ &\leqq \sup_{f_1} E^{f_1}[g_{1,2}(s_1, a_1, s_2; U^{N-1}(s_2))] \end{aligned}$$

by the use of the stochastic recursiveness and the monotonicity. Thus we have

$$U^N(s_1) \leqq \sup_{a \in A_1(s_1)} E^a[g_{1,2}(s_1, a, s_2; U^{N-1}(s_2))]. \tag{3.9}$$

To show the converse inequality, we need the following lemma.

LEMMA. *For any probability measure* p *on* S *and any* $\varepsilon > 0$, *there exists a Markov policy* $\hat{\pi}$ *such that* $p\{\mathcal{J}^N(\hat{\pi}) \geqq U^N - \varepsilon\} = 1$.

It is noted that the assertion of this lemma is more strengthened than that of Lemma 3.1, but the proof follows the line of it.

Now take $a \in A_1(s_1)$ arbitrarily. By the use of above lemma applied to the probability measure $q_1(\cdot \mid s_1, a)$, we find a Markov policy $\hat{\pi}$ such that

$$\mathcal{G}^{N-1}(\hat{\pi}) \geqq U^{N-1}-\varepsilon \quad \text{a.e.} \quad (q_1(\cdot \mid s_1, a)).$$

From the monotonicity and Condition (L) then we have

$$\begin{aligned}(3.10) \qquad E^a[g_{1,2}(s_1, a, s_2; U^{N-1}(s_2))] &\leqq E^a[g_{1,2}(s_1, a, s_2; \mathcal{G}^{N-1}(\hat{\pi})(s_2)+\varepsilon)] \\ &\leqq E^a[g_{1,2}(s_1, a, s_2; \mathcal{G}^{N-1}(\hat{\pi})(s_2))]+K_1\varepsilon.\end{aligned}$$

Let $\pi^*=\{a, \hat{\pi}\}$, then π^* is a policy in reality and the first term in the last side of (3.10) is equal to the expectation of reward received from π^*. Hence (3.10) leads us to

$$E^a[g_{1,2}(s_1, a, s_2; U^{N-1}(s_2))] \leqq U^N(s_1)+K_1\varepsilon.$$

By letting $\varepsilon \downarrow 0$, we have

$$(3.11) \qquad \sup_{a \in A_1(s_1)} E^a[g_{1,2}(s_1, a, s_2; U^{N-1}(s_2))] \leqq U^N(s_1).$$

It can be seen from (3.9) and (3.11) that the recurrence relation holds in the case when $n=N$.

For general case, follow the line of the above.

THEOREM 3.3 (*Weak Principle of Optimality*). *Let* $\{g_{mn}\}$ *be same as in Theorem* 3.2. *Let Condition* (L) *be satisfied. Then the optimal policy* π^* *satisfies the following relations:*

$$\begin{aligned}(3.12) \qquad & E^{\pi^*}[g_{1,N-n+1}(h_{N-n+1}; E^{N-n_{\pi^*}}[g_{N-n+1,N+1}(s_{N-n+1}, a_{N-n+1}, \cdots, s_{N+1}; d(s_{N+1}))])] \\ &= E^{\pi^*}[g_{1,N-n+1}(h_{N-n+1}; U^n(s_{N-n+1}))] \qquad \textit{for} \quad n=1, 2, \cdots, N.\end{aligned}$$

PROOF. Follow the same line as in Theorem 2.3 by the use of (3.8).

THEOREM 3.4 (*Principle of Optimality*). *Let* $\{g_{mn}\}$ *be stochastically recursive and stochastically strictly monotone. Let Conditon* (L) *be satisfied. Then the optimal policy* π^* *satisfies the following relations:*

$$\begin{aligned}& E^{N-n_{\pi^*}}[g_{N-n+1,N+1}(s_{N-n+1}, a_{N-n+1}, \cdots, s_{N+1}; d(s_{N+1}))] \\ &= U^n(s_{N-n+1}) \quad a.s. \ (P^{\pi^*}), \qquad \textit{for} \quad n=1, 2, \cdots, N.\end{aligned}$$

PROOF. Easy from Theorem 3.3 and from the stochastic strict monotonicity.

The following theorem supplies us an algorithm for finding an optimal policy.

THEOREM 3.5. *Let the assumptions be same as in Theorem* 3.3. *Take a Markov policy* $\pi^*=\{f_1^*, f_2^*, \cdots, f_N^*\}$ *in order that the following relations be satisfied:*

$$\begin{aligned}& E^{f_n^*}[g_{n,n+1}(s, f_n^*(s), s'; U^{N-n}(s'))] = \sup_{a \in A_n(s)} E^a[g_{n,n+1}(s, a, s'; U^{N-n}(s'))] \\ &\qquad\qquad \textit{for} \quad s \in S \ \textit{and} \ n=1, 2, \cdots, N.\end{aligned}$$

Then π^* *is optimal.*

PROOF. Follow the same line as in Theorem 2.5.

4. Reward systems.

We classify the DPs according to the reward system (RS). The corresponding objective function (OF) of each RS is explicitly written down. We give sufficient conditions for the $\{g_{mn}\}$ to satisfy the monotonicity, strict monotonicity, stochastic monotonicity, recursiveness, stochastic recursiveness, and Condition (L) (or (L′)). Throughout this section each DP is assumed to have the fixed elements, S, $\{A_n\}_{1\leqq n\leqq N}$, $\{q_n\}_{1\leqq n\leqq N}$ (or $\{T_n\}_{1\leqq n\leqq N}$) except for the element $\{g_{mn}\}_{1\leqq m<n\leqq N+1}$, and $\mathscr{B}(X)$ (resp. $\mathcal{C}(X)$) denotes the set of all real-valued Borel measurable (resp. continuous) functions on X. The following § 4.1-4.16 correspond to § 6.1-6.16 in [6], respectively, which have treated the game version of our DP.

4.1. Non-stationary recursive DP.

A DP has a *non-stationary recursive* RS $\{\{g_n\}_{1\leqq n\leqq N}, d\}$ if each g_{mn} is expressed as follows:

$$g_{mn}(s_m, a_m, s_{m+1}, \cdots, l(s_n))=g_m(s_m, a_m, s_{m+1}\,;\, g_{m+1}(s_{m+1}, a_{m+1}, s_{m+2}\,;\cdots$$
$$g_{n-1}(s_{n-1}, a_{n-1}, s_n\,;\, l(s_n))\cdots))\,,$$

where $g_n\in\mathscr{B}(S\times A\times S\times R)$. Then the OF is

$$R_N(h)=g_1(s_1, a_1, s_2\,;\, g_2(s_2, a_2, s_3\,;\cdots g_N(s_N, a_N, s_{N+1}\,;\, d(s_{N+1}))\cdots))\,.$$

4.2. Stationary recursive DP.

A DP has a *stationary recursive* RS $\{g, d\}$ if $g_n=g$, $1\leqq n\leqq N$ in the non-stationary recursive RS.

4.3. Stationary recursive DP with stage-wise reward $\{r_n\}$.

A DP has a *stationary recursive* RS *with stage-wise reward* $\{r_n\}$ if in the stationary recursive RS

$$g(s_n, a_n, s_{n+1}\,;\, l(s_{n+1}))=g(r_n(s_n, a_n, s_{n+1})\,;\, l(s_{n+1}))\qquad 1\leqq n\leqq N\,,$$

where $g\in\mathcal{C}(R\times R)$ and $r_n\in\mathscr{B}(S\times A\times S)$. Then the OF is

$$R_N(h)=g(r_1(s_1, a_1, s_2)\,;\, g(r_2(s_2, a_2, s_3)\,;\cdots g(r_N(s_N, a_N, s_{N+1})\,;\, d(s_{N+1}))\cdots))\,.$$

The DPs in § 4.4-4.16 can be viewed as the finite harizontal version of MDPs in [3].

4.4. Non-stationary recursive additive DP.

A DP has a *non-stationary recursive additive* RS $\{\{r_n\}_{1\leqq n\leqq N}, \{\beta_n\}_{1\leqq n\leqq N}, \{t_n\}_{1\leqq n\leqq N+1}, d\}$ if in the non-stationary recursive RS

$$(4.1)\qquad g_n(s, a, s'\,;\, l(s'))=t_n(r_n(s, a, s'))+\beta_n(s, a, s')t_{n+1}(l(s'))\,,$$

where $t_n\in\mathcal{C}(R)$ and $\beta_n\in\mathscr{B}(S\times A\times S)$, $\beta_n\geqq 0$. We call t_n and β_n *n-th translator* of the stage-wise reward and *n-th accmulator* of the RS, respectively. Then the OF is

$$R_N(h)=t_1(r_1(s_1, a_1, s_2))+\beta_1(s_1, a_1, s_2)t_2(r_2(s_2, a_2, s_3))$$
$$+\beta_1(s_1, a_1, s_2)\beta_2(s_2, a_2, s_3)t_3(r_3(s_3, a_3, s_4))+\cdots$$

$$+\beta_1(s_1, a_1, s_2)\beta_2(s_2, a_2, s_3)\cdots\beta_{N-1}(s_{N-1}, a_{N-1}, s_N)t_N(r_N, (s_N, a_N, s_{N+1}))$$

$$+\beta_1(s_1, a_1, s_2)\beta_2(s_2, a_2, s_3)\cdots\beta_N(s_N, a_N, s_{N+1})t_{N+1}(d(s_{N+1}))\,.$$

Of course this RS is stochastically recursive (therefore, recursive). Note that $\beta_n>0$ implies strict monotonicity and stochastically strict monotonicity and that $\beta_n\geqq 0$ implies monotonicity. Condition (L) is satisfied provided that $0\leqq\beta_n\leqq K_n<\infty$, $t_n(c+\varepsilon)-t_n(c)\leqq K'_n\varepsilon$ for $c\in R^1$, sufficiently small $\varepsilon>0$, where K_n, $K'_n>0$ are constant.

4.5. Stationary recursive additive DP.

A DP has a *stationary recursive additive* RS $\{\{r_n\}, \{\beta_n\}, t, k\}$ if $t_n=t$, $1\leqq n\leqq N+1$ in the non-stationary recursive additive RS.

4.6. Additive DP.

A DP has an *additive* RS $\{\{r_n\}, d\}$ if $\beta_n\equiv 1$, $t(r)=r$ in the stationary recursive additive RS. Then the OF is

$$R_N(h)=r_1(s_1, a_1, s_2)+\cdots+r_N(s_N, a_N, s_{N+1})+d(s_{N+1})\,.$$

The additive RS, which is typical and important in DP problems, is stochastically recursive, stochastically strictly monotone, and strictly monotone, and satisfies Condition (L).

4.7. Non-stationary multiplicative additive DP.

A DP has a *non-stationary multiplicative additive* RS $\{\{r_n\}, \{t_n\}, d\}$ if $\beta_n=r_n$ $1\leqq n\leqq N$ in the non-stationary recursive additive RS, where $r_n\in\mathscr{B}(S\times A\times S)$, $r_n\geqq 0$. Then the OF is

$$\begin{aligned}R_N(h)=&t_1(r_1(s_1, a_1, s_2))+r_1(s_1, a_1, s_2)t_2(r_2(s_2, a_2, s_3))\\&+\cdots+r_1(s_1, a_1, s_2)r_2(s_2, a_2, s_3)\\&\cdots r_{N-1}(s_{N-1}, a_{N-1}, s_N)t_N(r_N(s_N, a_N, s_{N+1}))\\&+r_1(s_1, a_1, s_2)r_2(s_2, a_2, s_3)\cdots r_N(s_N, a_N, s_{N+1})t_{N+1}(d(s_{N+1}))\,.\end{aligned}$$

4.8. Stationary multiplicative DP.

A DP has a *stationary multiplicative additive* RS $\{\{r_n\}, t, d\}$ if $t_n=t$, $1\leqq n\leqq N+1$ in the non-stationary multiplicative additive RS.

4.9. Non-stationary divided additive DP.

A DP has a *non-stationary divided additive* RS $\{\{r_n\}, \{t_n\}, d\}$ if $\beta_n=1/r_n$, $1\leqq n\leqq N$ in the non-stationary recursive additive RS, where $r_n\in\mathscr{B}(S\times A\times S)$, $\inf r_n(\cdot)>0$. Then the OF is

$$R_N(h)=t_1(r_1(s_1, a_1, s_2))+\frac{t_2(r_2(s_2, a_2, s_3))}{r_1(s_1, a_1, s_2)}+\cdots$$
$$+\frac{t_N(r_N(s_N, a_N, s_{N+1}))}{r_1(s_1, a_1, s_2)r_2(s_2, a_2, s_3)\cdots r_{N-1}(s_{N-1}, a_{N-1}, s_N)}$$
$$+\frac{t_{N+1}(d(s_{N+1}))}{r_1(s_1, a_1, s_2)r_2(s_2, a_2, s_3)\cdots r_N(s_N, a_N, s_{N+1})}$$

4.10. Stationary divided additive DP.

A DP has a *stationary divided additive* RS $\{\{r_n\}, t, d\}$ if $t_n=t$, $1\leqq n\leqq N+1$ in the non-stationary divided additive RS.

4.11. Non-stationary logarithmic additive DP.

A DP has a *non-stationary logarithmic additive* RS $\{\{r_n\}, \{t_n\}, d\}$ if $\beta_n=\log r_n$, $1\leqq n\leqq N$ in the non-stationary recursive additive RS, where $r_n\in\mathscr{B}(S\times A\times S)$, $r_n\geqq e$. Then the OF is

$$R_N(h)=t_1(r_1(s_1, a_1, s_2))+(\log r_1(s_1, a_1, s_2))t_2(r_2(s_2, a_2, s_3))$$
$$+\cdots+(\log r_1(s_1, a_1, s_2))(\log r_2(s_2, a_2, s_3))$$
$$\cdots(\log r_{N-1}(s_{N-1}, a_{N-1}, s_N))t_N(r_N(s_N, a_N, s_{N+1}))$$
$$+(\log r_1(s_1, a_1, s_2))(\log r_2(s_2, a_2, s_3))$$
$$\cdots(\log r_N(s_N, a_N, s_{N+1}))t_{N+1}(d(s_{N+1}))\,.$$

4.12. Stationary logarithmic additive DP.

A DP has a *stationary logarithmic additive* RS $\{\{r_n\}, t, d\}$ if $t_n=t$, $1\leqq n\leqq N+1$ in the non-stationary logarithmic additive RS.

4.13. Non-stationary exponential additive DP.

A DP has a *non-stationary exponential additive* RS $\{\{r_n\}, \{t_n\}, d\}$ if $\beta_n=e^{r_n}$, $1\leqq n\leqq N$ in the non-stationary recursive additive RS, where $r_n\in\mathscr{B}(S\times A\times S)$. Then the OF is

$$R_N(h)=t_1(r_1(s_1, a_1, s_2))+e^{r_1(s_1,a_1,s_2)}t_2(r_2(s_2, a_2, s_3))$$
$$+\cdots+e^{r_1(s_1,a_1,s_2)+r_2(s_2,a_2,s_3)+\cdots+r_{N-1}(s_{N-1},a_{N-1},s_N)}t_N(r_N(s_N, a_N, s_{N+1}))$$
$$+e^{r_1(s_1,a_1,s_2)+r_2(s_2,a_2,s_3)+\cdots+r_N(s_N,a_N,s_{N+1})}t_{N+1}(d(s_{N+1}))\,.$$

4.14. Stationary exponential aditive DP.

A DP has a *stationary exponential additive* RS $\{\{r_n\}, t, d\}$ if $t_n=t$, $1\leqq n\leqq N+1$ in the non-stationary exponential additive RS.

4.15. Multiplicative DP.

A DP has a *multiplicative* RS $\{\{r_n\}, d\}$ if

$$g(r_n(s, a, s')\,;\, l(s'))=r_n(s, a, s')\times l(s') \tag{4.2}$$

in the stationary recursive RS with stage-wise reward r_n, where $r_n \in \mathscr{B}(S\times A\times S)$, $r_n \geqq 0$, $d \geqq 0$. Then the OF is

$$R_N(h)=r_1(s_1,\ a_1,\ s_2)r_2(s_2,\ a_2,\ s_3)\cdots r_N(s_N,\ a_N,\ s_{N+1})d(s_{N+1})\,.$$

Clearly the RS is stochastically recursive and monotone. Further $r_n>0$, $1\leqq n\leqq N$ implies stochastically strict monotonicity and strict monotonicity. Condition (L) is satisfied provided $0\leqq r_n\leqq K_n$, where K_n is constant.

4.16. Terminal DP.

A DP has a *terminal* RS$\{d\}$ if $r_n\equiv 0$, $1\leqq n\leqq N$ in the additive RS. Then the terminal OF is

$$R_N(h)=d(s_{N+1})\,.$$

It is natural that this RS satisfies all the conditions stated in the additive DP. Note that this OF is also the case where $r_n\equiv 1$, $1\leqq n\leqq N$ in the multiplicative OF.

The DPs in § 4.17-4.21 are restricted only to the deterministic case, because they do not, in general, satisfy the stochastic recursiveness.

4.17(18). Maximum (Minimum) DP.

A DP has a *maximum* (*minimum*) RS$\{\{r_n\},\ d\}$ if

$$g(r_n(s,\ a,\ s')\,;\ l(s'))=\max\,(r_n(s,\ a,\ s'),\ l(s'))$$
$$(=\min\,(r_n(s,\ a,\ s'),\ l(s')))$$

in the stationary recursive RS with stage-wise reward r_n, where $r_n\in\mathscr{B}(S\times A\times S)$. Then the OF is

$$R_N(h)=\max\,(r_1(s_1,\ a_1,\ s_2),\ r_2(s_2,\ a_2,\ s_3),\ \cdots,\ r_N(s_N,\ a_N,\ s_{N+1}),\ d(s_{N+1}))$$
$$(=\min\,(r_1(s_1,\ a_1,\ s_2),\ r_2(s_2,\ a_2,\ s_3),\ \cdots,\ r_N(s_N,\ a_N,\ s_{N+1}),\ d(s_{N+1})))\,.$$

(see [2, p. 57], [9, p. 56, p. 89]). The RS is recursive and monotone. It is easily shown that the maximum (minimum) RS satisfies Condition (L).

4.19. Backward power DP.

A DP has a *backward power* RS$\{\{r_n\},\ d\}$ if

$$g(r_n(s,\ a,\ s')\,;\ l(s'))=r_n(s,\ a,\ s')^{l(s')}$$

in the stationary recursive RS with stage-wise reward r_n, where $r_n\in\mathscr{B}(S\times A\times S)$. Then the OF is

$$R_N(h)=r_1(s_1,\ a_1,\ s_2)^{r_2(s_2,\ a_2,\ s_3)^{\cdot^{\cdot^{\cdot^{r_N(s_N,\ a_N,\ s_{N+1})^{d(s_{N+1})}}}}}}$$

where $x^{y^{\cdot^{\cdot^{\cdot^{z^w}}}}}=\left(x^{\left(y^{\cdot^{\cdot^{\cdot^{(z^w)}}}\cdot\cdot\right)}\right)$. This RS is recursive. Note that $\min r_n(\cdot)>1$

implies strict monotonicity. Furthermore, $1<r_n\leqq K$, $1\leqq n\leqq N$, K constant implies Condition (L′) (see Remark 1 in § 2.2).

4.20. Forward power DP.

A DP has a *forward power* RS$\{\{r_n\}, d\}$ if

$$g(r_n(s, a, s'); l(s'))=l(s')^{r_n(s, a, s')}$$

in the stationary recursive RS with stage-wise reward r_n, where $r_n\in\mathscr{B}(S\times A\times S)$. Then the OF is

$$R_N(h)=d(s_{N+1})^{r_N(s_N, a_N, s_{N+1})^{\cdot^{\cdot^{r_2(s_2, a_2, s_3)^{r_1(s_1, a_1, s_2)}}}}}$$

where in this case $w^{z^{\cdot^{\cdot^{y^x}}}}=\left(\left(\cdot^{\cdot^{(w^z)}}\right)^{\cdot^{\cdot^{y}}}\right)^x$. This RS is recursive. Note that $\min r_n(\cdot)>0$ implies strict monotonicity. Furthermore, $\min r_n(\cdot)>1$ and $r_n\leqq K$, $1\leqq n\leqq N$, K constant Condition (L′) (see Remark 1 in § 2.2).

5. Examples.

5.1. Deterministic case.

The mathematical programming problems in Examples 1-4 may be reduced to our DPs and analytically solved by calculating and using the corresponding optimal policies and optimal returns. They have continuous state and action spaces. Note that $u^n(s)$ or $U^n(s)$ is attained by the maximum or minimum in each examples (see Remark 2 in § 2.2).

EXAMPLE 1. Minimize $\left(\frac{1}{x_1}+2x_1^2\right)\times(\max(x_2, x_3))$

subject to (i) $x_1(x_2+x_3)=c\ (>0)$

(ii) $x_1, x_2, x_3>0$.

This reduces to a non-stationary recursive DP $(S, \{A_n\}_{1\leqq n\leqq 3}, \{T_n\}_{1\leqq n\leqq 3}, \{g_n\}_{1\leqq n\leqq 3}, d)$, where

$$N=3,\quad s=c,\quad a=x,\quad S=(0, \infty),\quad A_1(c)=(0, \infty),\quad A_2(c)=(0, c],$$

$$A_3(c)=\{c\},\quad T_1(c, x)=\frac{c}{x},\quad T_2(c, x)=T_3(c, x)=c-x,$$

$$g_1(c, x, c'; l(c'))=\left(\frac{1}{x}+2x^2\right)\times l(c'),\quad g_2(c, x, c'; l(c'))=\max(x, l(c')),$$

$$g_3(c, x, c'; d(c'))=x+d(c'),\quad d(c)=0.$$

Then by solving the recursive equations

$$u^n(c)=\underset{x_n\in A_n(c)}{\mathrm{Min}}\, g_n(c, x_n, T_n(c, x_n); u^{n-1}(T_n(c, x_n)))\qquad 1\leqq n\leqq 3,$$

$$u^0(c)=d(c)=0\,,$$

we have the optimal policy $\{f_1^*, f_2^*, f_3^*\}$ and optimal returns u^0, u^1, u^2, u^3 as follows:

$$f_1^*(c)=1,\quad f_2^*(c)=\frac{c}{2},\quad f_3^*(c)=c,$$

$$u^0(c)=0,\quad u^1(c)=c,\quad u^2(c)=\frac{c}{2},\quad u^3(c)=\frac{3}{2}c\,.$$

Therefore, when

$$x_1^*=f_1^*(c)=1,\qquad x_2^*=f_2^*(T_1(c,\ x_1^*))=\frac{c}{2},$$

$$x_3^*=f_3^*(T_2(T_1(c,\ x_1^*),\ x_2^*))=\frac{c}{2},$$

the given problem attains the minimum $u^3(c)=\frac{3}{2}c$.

EXAMPLE 2. Maximize $\max(x_1,\ x_2+x_3\cdot x_4)$

subject to (i) $x_1+x_2+x_3+x_4=c\quad(\geqq 0)$

(ii) $x_n\geqq 0\qquad 1\leqq n\leqq 4$.

This reduces to a non-stationary recursive DP whose elements are specified as follows:

$$N=4,\quad s=c,\quad a=x,\quad S=[0,\infty),\quad A_n(c)=[0,\ c]\qquad 1\leqq n\leqq 3,\quad A_4(c)=\{c\},$$

$$T_n(c,\ x)=c-x\qquad 1\leqq n\leqq 4,\quad g_1(c,\ x,\ c'\ ;\ l(c'))=\max(x,\ l(c')),$$

$$g_2(c,\ x,\ c'\ ;\ l(c'))=x+l(c'),\quad g_3(c,\ x,\ c'\ ;\ l(c'))=x\times l(c'),$$

$$g_4(c,\ x,\ c'\ ;\ d(c'))=x+d(c'),\quad d(c)=0\,.$$

We have the optimal policy

$$f_1^*(c)=\begin{cases}0 \text{ or } c & 0\leqq c\leqq 4\\ 0 & c>4,\end{cases}\qquad f_2^*(c)=\begin{cases}c & 0\leqq c\leqq 4\\ 0 & c>4,\end{cases}$$

$$f_3^*(c)=\frac{c}{2},\qquad f_4^*(c)=c\,,$$

and the optimal returns

$$u^0(c)=0,\quad u^1(c)=c,\quad u^2(c)=\frac{c^2}{4},\quad u^3(c)=u^4(c)=\begin{cases}c & 0\leqq c\leqq 4\\ \dfrac{c^2}{4} & c>4\,.\end{cases}$$

Hence, if $0\leqq c\leqq 4$ (resp. $c>4$) the problem yields the maximum c $\left(\text{resp. } \frac{c^2}{4}\right)$ at the point $(x_1^*,\ x_2^*,\ x_3^*,\ x_4^*)=(0,\ c,\ 0,\ 0)$ or $(c,\ 0,\ 0,\ 0)$ $\left(\text{resp, } \left(0,\ 0,\ \frac{c}{2},\ \frac{c}{2}\right)\right)$. This completes Example 2.

Before we proceed to discuss the further examples, it should be noted that any function composed recursively by virtue of the operations "addition", "multiplication", "multiplicative addition", "maximum" ··· may, in some appropriate condition, be used as both objective and constraint functions in such problems as Examples 1, 2. Moreover, note that the constraint consisting of p $(\geqq 2)$ inequalities whose left hand sides are such functions composed above is also expressed as our state transformation [7]. For example, the constraint

$$x_1+x_2x_3 \leqq c_1 \qquad (\geqq 0)$$

$$x_1+x_2+x_3 \leqq c_2 \qquad (\geqq 0)$$

is expressed as follows:

$$N=3,\quad s=(c_1,\, c_2),\quad a=x,\quad S=R^2_+,\quad A_1(c_1,\, c_2)=[0,\, \min(c_1,\, c_2)]\,,$$

$$A_2(c_1,\, c_2)=(0,\, c_2],\quad A_3(c_1,\, c_2)=A_1(c_1,\, c_2),\quad T_1((c_1,\, c_2),\, x)=(c_1-x,\, c_2-x),$$

$$T_2((c_1,\, c_2),\, x)=\left(\frac{c_1}{x},\, c_2-x\right),\quad T_3((c_1,\, c_2),\, x)=(c_1-x,\, c_2-x)\,.$$

EXAMPLE 3. Let's now consider the unconditional maximum problem stated in [2, p. 102]:

$$\text{Maximize } (1-x_1)e^{x_1}+(1-x_2)e^{x_1+x_2}+ \cdots +(1-x_N)e^{x_1+x_2+\cdots+x_N}\,.$$

This, however, reduces to a stationary exponential additive DP whose elements are specified as follows: N, $s=c$, $a=x$, $S=\{c_0\}$, i. e., one point set, $A_n(c_0)=R^1$, $T_n(c_0,\, x)=c_0$, $r_n(c_0,\, x,\, c_0)=x$ $1\leqq n \leqq N$, $t(r)=(1-e)e^r$, $d(c_0)=1$. Then the optimal policy is

$$f_1^*(c_0)=e^{e^{\cdot^{\cdot^{e}}}}_{(N-2)\text{-factors}},\ \cdots,\ f_{N-3}(c_0)=e^e,\quad f_{N-2}(c_0)=e\,,\quad f_{N-1}(c_0)=1,\quad f_N(c_0)=0\,,$$

and the optimal returns are

$$u^0(c_0)=0,\quad u^1(c_0)=1,\quad u^2(c_0)=e,\quad u^3(c_0)=e^e,\ \cdots,\ u^N(c_0)=e^{e^{\cdot^{\cdot^{e}}}}_{(N-1)\text{-factors}}\,.$$

Hence the problem attains the maximum $u^N(c_0)=e^{e^{\cdot^{\cdot^{e}}}}$ ($(N-1)$-factors) at the point $(x_1^*,\, x_2^*,\, \cdots,\, x_N^*)=(e^{e^{\cdot^{\cdot^{e}}}}_{(N-2)\text{-factors}},\, e^{e^{\cdot^{\cdot^{e}}}}_{(N-3)\text{-factors}},\, \cdots,\, e,\, 1,\, 0)$. Here, $x^{y^{\cdot^{\cdot^{z^{w}}}}} = x^{\left(y^{\left(\cdot^{\cdot^{(z^{w})}}\right)}\right)}$.

EXAMPLE 4. We find the problem:

$$\text{Minimize } g(a_1)+\frac{g(a_2)}{a_1}+\frac{g(a_3)}{a_1a_2}+ \cdots +\frac{g(a_m)}{a_1a_2\cdots a_{m-1}}$$

$$\text{subject to (i) } a_i \geqq 1, \qquad i=1,\, 2,\, \cdots,\, m$$

$$\text{(ii) } a_1a_2\cdots a_m=x$$

in [2, p. 58]. This is rather extended to

$$\text{Minimize } g(x_1)+\frac{g(x_2)}{x_1}+\frac{g(x_3)}{x_1x_2}+ \cdots +\frac{g(x_m)}{x_1x_2\cdots x_{m-1}}$$

$$\text{subject to (i) } x_n>0, \qquad 1\leqq n \leqq m$$

$$\text{(ii) } x_1x_2\cdots x_n=c \quad (>0)\,.$$

Then the latter reduces to a stationary divided additive DP such that

$$N=m,\quad s=c,\quad a=x,\quad S=A_n(c)=(0,\, \infty)\ 1\leqq n \leqq m-1,\quad A_m(c)=\{c\},$$

$$T_n(c,\, x)=\frac{c}{x},\quad r_n(c,\, x,\, c')=x\quad 1\leqq n \leqq m,\ d(c)=0,\quad t(r)=g(r)\,.$$

The following Examples 5-11 have, on the turn, finite state and action spaces.

EXAMPLE 5. Maximize max $\{r(s_1, a_1, s_2), r(s_2, a_2, s_3), \cdots, r(s_{100}, a_{100}, s_{101}), d(s_{101})\}$ subject to the following stationary state transformation T,

s \ a	1	2	3
1	1	2	2
2	3	1	3
3	1	1	1

$T(s, a)$

, $S=A_n(s)=\{1, 2, 3\}$

$s=1, 2, 3$

where $d(s)=0$ and $r(s, a, s')$ is

s	a	$r(s, a, 1)$	$r(s, a, 2)$	$r(s, a, 3)$
1	1	10	4	8
	2	8	2	4
	3	4	6	4
2	1	13	0	18
	2	6	16	8
	3	−5	−5	−5
3	1	10	2	8
	2	6	4	2
	3	4	0	8

This is the maximum DP whose elements are specified above. We have the optimal policy $\{f_1^*, f_2^*, \cdots, f_{100}^*\}$, where

$$f_1^*=\begin{bmatrix}1\\1\\1\end{bmatrix},\quad f_2^*=\begin{bmatrix}2 \text{ or } 3\\1\\1\end{bmatrix},\quad f_3^*=\begin{bmatrix}2 \text{ or } 3\\1 \text{ or } 2\\1, 2 \text{ or } 3\end{bmatrix},$$

$$f_4^*=f_5^*=\cdots=f_{100}^*=\begin{bmatrix}1, 2 \text{ or } 3\\1, 2 \text{ or } 3\\1, 2 \text{ or } 3\end{bmatrix}$$

and the optimal returns $u^1 \to u^2 \to \cdots \to u^{100}$, where

$$\begin{bmatrix}10\\18\\10\end{bmatrix}\longrightarrow\begin{bmatrix}18\\18\\10\end{bmatrix}\longrightarrow\begin{bmatrix}18\\18\\18\end{bmatrix}\longrightarrow\begin{bmatrix}18\\18\\18\end{bmatrix}\longrightarrow\cdots\longrightarrow\begin{bmatrix}18\\18\\18\end{bmatrix}.$$

Here (resp. thereafter) we have used (resp. will use) the conventional notations, $f_n^*=\begin{bmatrix}f_n^*(1)\\f_n^*(2)\\f_n^*(3)\end{bmatrix}$ and $u^n=\begin{bmatrix}u^n(1)\\u^n(2)\\u^n(3)\end{bmatrix}$. For example, $f_2^*(1)=2$ or 3, $u^2(3)=10$.

EXAMPLE 6. Maximize $(x^{(y^z)})$ subject to (i) $x+y+z=9$, (ii) $x, y, z \geqq 2$, integers.

This reduces to a backward power DP whose elements are as follows:

$$N=3,\quad s=c,\quad a=x,\ y \text{ or } z,\quad S=\{2,3,\cdots,9\},\quad A_1(c)=\{2,3,\cdots,c-4\}$$

$$c=6,7,8,9,\quad A_2(c)=\{2,3,\cdots,c-2\}\quad c=4,5,\cdots,9,\quad A_3(c)=\{c\}$$

$$c=2,3,\cdots,9,\quad T_n(c,a)=c-a,\quad r_n(c,a,c')=a\quad 1\leqq n\leqq 3,\quad d(c)\equiv 1\,.$$

Then we have

c	$u^0(c)$,	$u^1(c)$	$f_3^*(c)$,	$u^2(c)$	$f_2^*(c)$,	$u^3(c)$	$f_1^*(c)$
2	1	2	2				
3	1	3	3				
4	1	4	4	2^2	2		
5	1	5	5	3^2	3		
6	1	6	9	3^3	3	2^4	2
7	1	7	7	3^4	3	2^8	2
8	1	8	8	4^4	4	2^{27}	2
9	1	9	9	4^5	4	2^{81}	2

Therefore, the problem attains the maximum $u^3(9)=2^{81}$ at the point $(x^*, y^*, z^*)=(2,3,4)$.

EXAMPLE 7. Maximize $(z^y)^x$ subject to (i) $x+y+z=9$, (ii) $x, y, z\geqq 2$, integers. This reduces to a forward power DP whose elements are the same as those of Example 6 except for $d(c)\equiv 0$. We have

c	$u^0(c)$,	$u^1(c)$	$f_3^*(c)$,	$u^2(c)$	$f_2^*(c)$,	$u^3(c)$	$f_1^*(c)$
2	0	2	2				
3	0	3	3				
4	0	4	4	2^2	2		
5	0	5	5	3^2	2		
6	0	6	6	3^3	3	2^4	2
7	0	7	7	3^4	4	3^4	2
8	0	8	8	4^4	4	3^6	2
9	0	9	9	4^5	5	3^9	3

Therefore, the problem attains the maximum $u^3(9)=3^9$ at the point $(x^*, y^*, z^*)=(3,3,3)$.

5.2. Stochastic case.

EXAMPLE 8. Maximize $E^\pi(\prod_{k=1}^{5} r(s_k, a_k, s_{k+1}))$ subject to the following stationary transition law q, where $S=A_n(s)=\{1,2,3\}$ $s=1,2,3$, $n=1,2,\cdots,5$, and the stationary stage-wise reward r is also given as follows:

s	a	$q(1\|s,a)$	$q(2\|s,a)$	$q(3\|s,a)$	$r(s,a,1)$	$r(s,a,2)$	$r(s,a,3)$
	1	0.5	0.25	0.25	0.5	0.75	1.5
1	2	0.0625	0.75	0.1875	1.5	2.0	0.33···
	3	0.25	0.125	0.625	2.0	1.5	0.5
	1	0.5	0.0	0.5	1.0	2.0	1.5
2	2	0.0625	0.875	0.0625	0.5	1.5	2.0
	3	0.33···	0.33···	0.33···	2.0	1.0	0.5
	1	0.25	0.25	0.5	2.0	0.5	1.0
3	2	0.125	0.75	0.125	1.0	2.5	0.5
	3	0.75	0.0625	0.1875	0.333···	0.5	2.0

This problem reduces to a multiplicative DP with the terminal reward function $d(s)=1$ $s=1, 2, 3$. Then we have the optimal policy $\{f_1^*, f_2^*, \cdots, f_5^*\}$ and the sequence of optimal returns $U^1 \rightarrow U^2 \cdots \rightarrow U^5$ as follows:

$$f_1^*=f_3^*=\begin{bmatrix}2\\2\\2\end{bmatrix}, \qquad f_2^*=f_4^*=f_5^*=\begin{bmatrix}2\\1\\2\end{bmatrix},$$

$$\begin{bmatrix}1.656\\1.469\\2.063\end{bmatrix}\longrightarrow\begin{bmatrix}2.487\\2.375\\3.090\end{bmatrix}\longrightarrow\begin{bmatrix}3.989\\3.581\\4.957\end{bmatrix}\longrightarrow\begin{bmatrix}6.055\\5.712\\7.523\end{bmatrix}\longrightarrow\begin{bmatrix}9.606\\8.670\\11.937\end{bmatrix}.$$

EXAMPLE 9. Further consider the problem:

Maximize $E^{\pi}(\sqrt{r_1}+r_1\sqrt{r_2}+r_1r_2\sqrt{r_3}+\cdots+r_1r_2\cdots r_{99}\sqrt{r_{100}}+r_1r_2\cdots r_{100}\sqrt{d})$ subject to the same transition law q in Example 8, where $S=A_n(s)=\{1, 2, 3\}$ $s=1, 2, 3$, $r_n=r(s_n, a_n, s_{n+1})$ $1\leqq n\leqq 100$, $d=d(s_{101})$. In this case, r and d are given as follows:

s	$d(s)$	a	$r(s,a,1)$	$r(s,a,2)$	$r(s,a,3)$
		1	1.5	0.5	0.8
1	1.0	2	0.8	1.2	2.0
		3	1.2	1.5	0.6
		1	1.0	2.0	1.5
2	1.0	2	2.4	0.6	0.8
		3	1.0	0.8	0.5
		1	0.5	1.0	0.8
3	0.8	2	2.0	0.8	0.5
		3	0.75	2.0	2.0

This is a stationary multiplicative additive DP with $t(r)=\sqrt{r}$. The optimal policy is stationary one specified by $f^*=\begin{bmatrix}2\\1\\3\end{bmatrix}$, and the sequence of optimal returns $U^1 \rightarrow U^2 \rightarrow \cdots \rightarrow U^{100}$ is

$$\begin{bmatrix}2.51\\2.28\\2.04\end{bmatrix}\to\begin{bmatrix}4.09\\3.90\\3.47\end{bmatrix}\to\begin{bmatrix}6.16\\5.76\\5.09\end{bmatrix}\to\cdots\to\begin{bmatrix}4.382\times10^8\\4.088\times10^8\\3.613\times10^8\end{bmatrix}\to\begin{bmatrix}5.25\times10^8\\4.90\times10^8\\4.33\times10^8\end{bmatrix}.$$

EXAMPLE 10. Maximize $E^{\pi}[d(s_6)]$ subject to the same transition law as q in Example 8, where $S=A_n(s)=\{1, 2, 3\}$ and $d(1)=0.5$, $d(2)=0.0$, $d(3)=-0.2$. This is a terminal DP. Then we have the optimal policy $\{f_1^*, f_2^*, \cdots, f_5^*\}$ and the sequence of optimal returns $U^1\to U^2\to\cdots\to U^5$ as follows:

$$f_1^*=f_3^*=f_5^*=\begin{bmatrix}1\\2\\3\end{bmatrix},\quad f_2^*=f_4^*=\begin{bmatrix}3\\1\\1\end{bmatrix},$$

$$\begin{bmatrix}0.200\\0.150\\0.338\end{bmatrix}\longrightarrow\begin{bmatrix}0.280\\0.269\\0.256\end{bmatrix}\longrightarrow\begin{bmatrix}0.271\\0.269\\0.274\end{bmatrix}\longrightarrow\begin{bmatrix}0.273\\0.273\\0.272\end{bmatrix}\longrightarrow\begin{bmatrix}0.273\\0.273\\0.273\end{bmatrix}.$$

EXAMPLE 11. Finally we consider the problem: Maximize $E^{\pi}[r_1+r_2r_3+r_2r_3r_4+r_2r_3d]$ subject to the following stationary transition law q, where $S=A_n(s)=\{1, 2\}$, $r_n=r(s_n, a_n, s_{n+1})$ $s=1, 2$, $n=1, 2, 3$, $d=d(s_5)$, and r and d are also given as follows:

s	$d(s)$	a	$q(1\mid s, a)$	$q(2\mid s, a)$	$r(s, a, 1)$	$r(s, a, 2)$
1	1.0	1	$\frac{2}{3}$	$\frac{1}{3}$	2	3
		2	$\frac{1}{2}$	$\frac{1}{2}$	4	2
2	1.0	1	0	1	1	1
		2	$\frac{1}{2}$	$\frac{1}{2}$	2	0

We have

s	$u^0(s)$	$U^1(s)$	$f_4^*(s)$	$U^2(s)$	$f_3^*(s)$	$U^3(s)$	$f_2^*(s)$	$U^4(s)$	$f_1^*(s)$
1	1	4	2	13	2	31	2	$\frac{82}{3}$	1
2	1	2	1 or 2	5	2	13	2	23	2

Note that this is a non-stationary recursive DP such that

$$g_1(s, a, s'; l(s'))=r_1(s, a, s')+l(s')$$

$$g_2(s, a, s'; l(s'))=r_2(s, a, s')l(s')$$

$$g_3(s, a, s'; l(s'))=r_3(s, a, s')+r_3(s, a, s')l(s')$$

$$g_4(s, a, s'; l(s'))=r_4(s, a, s')+l(s').$$

References

[1] ARSENIN, W. J. and LJAPUNOV, A. A. *Die Theorie der A-Mengen*, Arbeiten zur Deskriptiven Mengenlehre. Deutcher Verlag, Berlin. (1955), 35–93.

[2] BELLMAN, R. Dynamic Programming. Princeton University Press, Princeton, New Jersey, 1957.

[3] FURUKAWA, N. and IWAMOTO, S. *Markovian decision processes with recursive reward functions*. Bull. Math. Statist., **15**, No. 3-4 (1973), 79–91.

[4] FURUKAWA, N. and IWAMOTO, S. Correction to "*Markovian decision processes with recursive reward functions*". Bull. Math. Statist., **16**, No. 1-2 (1974), 127.

[5] GOŚCIŃSKI, A. and JAKUBOWSKI, R. *A stochastic automata theoretical approach to dynamic programming*. Bull. Acad. Polon. Sci. Ser. Sci. Techn. **20** (1972), 425–429.

[6] IWAMOTO, S. *Finite horizon Markov games with recursive payoff systems*. Mem. Fac. Sci. Ser. Math. Kyushu Univ., **29** (1975), 123–147.

[7] IWAMOTO, S. *Inverse theorem in dynamic programming I, II*. To appear in J. Math. Anal. Appl.

[8] KARP, R. M. and HELD, M. *Finite-state processes and dynamic programming*. SIAM J. Appl. Math. **15** (1967), 693–718.

[9] NEMHAUSER, G. L. Introduction to Dynamic Programming. John Wiley, New York, 1966.

[10] STRAUCH, R. E. *Negative dynamic programming*. Ann. Math. Statist. **37** (1966), 871–890.