<table>
<thead>
<tr>
<th>Title</th>
<th>Maximizing Order Probabilities on Controlled Markov Chains</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author(s)</td>
<td>Ueno, Takayuki; Iwamoto, Seiichi</td>
</tr>
<tr>
<td>Citation</td>
<td>数理解析研究所講究録 (2003), 1306: 201-209</td>
</tr>
<tr>
<td>Issue Date</td>
<td>2003-02</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://hdl.handle.net/2433/42823">http://hdl.handle.net/2433/42823</a></td>
</tr>
<tr>
<td>Type</td>
<td>Departmental Bulletin Paper</td>
</tr>
<tr>
<td>Textversion</td>
<td>publisher</td>
</tr>
</tbody>
</table>

Kyoto University
Maximizing Order Probabilities on Controlled Markov Chains

1 Introduction

It is natural to consider expectation criteria in stochastic decision problems. In particular, both
discounted expected value of total reward and average value per stage are well studied criteria
in Markov decision processes.

However, in some situation such as in economy of lower growth rate, growth of economy is in
itself a preferable criterion to total amount of production such as the gross national production.
How can we measure the growth of decision process? We take an order that sequence of earned
rewards is nondecreasing in stage direction.

In this paper we consider a probability criterion that the reward is nondecreasing in time—an
order probability—. We maximize the order probability on finite-horizon controlled Markov
chains. We show that the policy class for maximization depends upon reward function’s de-
pendence on today’s state, today’s decision and tomorrow’s state.

In Section 2, we formulate an optimization problem with order probability criterion on finite-
stage controlled Markov chains. Section 3 derives a recursive equation for decision process where
reward function is independent of today’s decision. Section 4 considers process with reward
function of today’s state, today’s decision and tomorrow’s state. It is shown that a recursive
relation is derived through imbedding method by expanding the original state space.

2 Decision Process with Order Probability

Throughout the paper, the following data is given:

\( N \geq 2 \) is an integer; the total number of stages
\( X = \{s_1, s_2, \ldots, s_l\} \) is a finite state space
\( U = \{a_1, a_2, \ldots, a_k\} \) is a finite action space
\( r_n : X \times U \to R^1 \) is an n-th reward function \((0 \leq n \leq N-1)\)
\( k : X \to R^1 \) is a terminal function
\( p = \{p(\cdot|\cdot, \cdot)\} \) is a Markov transition law

\[ p(y|x, u) \geq 0 \quad \forall (x, u, y) \in X \times U \times X, \quad \sum_{y \in X} p(y|x, u) = 1 \quad \forall (x, u) \in X \times U \]

\( y \sim p(\cdot|x, u) \) denotes that next state \( y \) conditioned on state \( x \) and action \( u \)
appears with probability \( p(y|x, u) \).
Let an $N$-stage controlled Markov chain \{(X_{n}, U_{n})\} on finite state space $X$ and finite decision space $U$ be under a Markov transition law $p$. We maximize the order probability that the reward (random variable) will appear in ascent order

\[ r_0(X_0, U_0) \leq r_1(X_1, U_1) \leq \cdots \leq r_{N-1}(X_{N-1}, U_{N-1}) \leq k(X_N). \]

The problem is to how to find an optimal policy which maximizes the order probability $P(r_0 \leq r_1 \leq \cdots \leq r_{N-1} \leq k)$. We focus our attention on policy class where the optimization should be taken.

The order probability

\[ P(r_0(X_0, U_0) \leq r_1(X_1, U_1) \leq \cdots \leq r_{N-1}(X_{N-1}, U_{N-1}) \leq k(X_N)) \]

depends not only upon initial state but also upon when, where and what the decision maker will choose. We maximize the order probability in Markov class:

Maximize $P^\pi_{x_0}(r_0 \leq r_1 \leq \cdots \leq r_{N-1} \leq k_N)$

subject to \( i \), \( \dot{u} \)

\[ p(x_{n+1}|x_n, u_n) \quad 1 \leq n \leq N \]

\[ u_n \in U \]

Here $P^\pi_{x_0}$ is the (discrete) probability measure on the product space $X^N$ induced from the transition law $p$, a Markov policy $\pi = \{\pi_0, \pi_1, \ldots, \pi_{N-1}\} \in \Pi$, and an initial state $x_0 \in X$. Thus the probability is expressed by the partial multiple summation:

\[ P^\pi_{x_0}(r_0 \leq \cdots \leq k) = \sum_{(x_1, x_2, \ldots, x_N) \in (*)} \sum_{(x_1, x_2, \ldots)} p(x_1|x_0, u_0)p(x_2|x_1, u_1) \cdots p(x_N|x_{N-1}, u_{N-1}) \]

where the domain (*) in which the partial multiple summation is taken denotes the set of all $(x_1, x_2, \ldots, x_N) \in X \times X \times \cdots \times X$ satisfying

\[ r_0(x_0, u_0) \leq r_1(x_1, u_1) \leq \cdots \leq r_{N-1}(x_{N-1}, u_{N-1}) \leq r(x_N). \]

Here we note that the sequence of intermediate decisions $\{u_0, u_1, \ldots, u_{N-1}\}$ is determined through the Markov policy $\pi = \{\pi_0, \ldots, \pi_{N-1}\}$ as follows:

\[ u_0 = \pi_0(x_0), \quad u_1 = \pi_1(x_1), \quad \ldots, \quad u_{N-1} = \pi_{N-1}(x_{N-1}). \]

Thus the order probability control problem is written as follows:

Maximize $P^\pi_{x_0}(r_0 \leq \cdots \leq k)$

subject to \( i \), \( \dot{u} \)

\[ 0 \leq n \leq N - 1. \]

We call this problem the order problem. Let $v_0(x_0)$ denote the maximum value of problem $P_0(x_0)$:

\[ v_0(x_0) \:= \max_{\pi \in \Pi} P^\pi_{x_0}(r_0 \leq \cdots \leq k) \quad x_0 \in X. \]

Then our problem is to find an optimal policy $\pi^*$ in Markov class $\Pi$:

\[ v_0(x_0) = P^*_{x_0}(r_0 \leq \cdots \leq k) \quad \forall x_0 \in X. \]
3 Subproblems

In this section we derive a recursive equation for process $P_0(x_0)$. Let $\pi = \{\pi_n, \pi_{n+1}, \ldots, \pi_{N-1}\}$ be a Markov policy for subprocess from $n$-th stage on. We denote by $\Pi(n)$ the set of all such Markov policies.

Now let us take $n (0 \leq n \leq N-1)$, $x_n \in X$ and a policy $\pi \in \Pi(n)$. We consider the order probability

$$P_{x_n}^\pi (r_n \leq r_{n+1} \leq \cdots \leq k)$$

$:= P_{x_n}^\pi (r_n(X_n, U_n) \leq r_{n+1}(X_{n+1}, U_{n+1}) \leq \cdots \leq r_{N-1}(X_{N-1}, U_{N-1}) \leq k(X_N))$.

for the process which starts at state $x_n \in X$ and is governed by $\pi \in \Pi(n)$. Formally we set $P_{x_n}(k(X_N)) := 1$.

Then we have the recursive relation:

**Lemma 3.1** For $n \leq N-2$ we have

$$P_{x_n}^\pi (r_n \leq r_{n+1} \leq \cdots \leq k) = \sum_{*} P_{x_{n+1}}^{\pi'} (r_{n+1} \leq \cdots \leq k) p(x_{n+1}|x_n, u_n)$$

$\leftrightarrow \ *: x_{n+1} \in X; r_n(x_n, u_n) \leq r_{n+1}(x_{n+1}, u_{n+1})$

where $\pi' = \{\pi_{n+1}, \ldots, \pi_{N-1}\}$, $u_m = \pi_m(x_m) \ m = n, n+1$.

Further for $\pi = \{\pi_{N-1}\}$ we have

$$P_{x_{N-1}}^\pi (r_{N-1} \leq k) = \sum_{*} P_{x_N}(k) p(x_N|x_{N-1}, u_{N-1})$$

$\leftrightarrow \ *: x_N \in X; r_{N-1}(x_{N-1}, u_{N-1}) \leq k(x_N)$

where $u_{N-1} = \pi_{N-1}(x_{N-1})$.

We see that Eq. (5) states

$$P_{x_{N-1}}^\pi (r_{N-1} \leq k) = \sum_{*} p(x_N|x_{N-1}, u_{N-1})$$

Now we consider the family of subproblems:

maximize $P_{x_n}^\pi (r_n \leq r_{n+1} \leq \cdots \leq r_{N-1} \leq k)$

subject to

(i) $X_{m+1} \sim p(\cdot|x_m, u_m) \quad n \leq m \leq N$

(ii) $u_m \in U$

where $0 \leq n \leq N-1$, $x_n \in X$. Let $v_n(x_n)$ denote the maximum value of problem $P_n(x_n)$:

$$v_n(x_n) := \max_{\pi \in \Pi(n)} P_{x_n}^\pi (r_n \leq \cdots \leq k) \quad x_n \in X.$$ (6)

where we set

$$v_N(x_N) := 1 \quad x_n \in X.$$ (7)

Then we have the recursive equation:
Theorem 3.1

\begin{align*}
v_N(x) &= 1 \quad x \in X \\
v_{N-1}(x) &= \max_{u \in U} \sum_{*} v_N(y)p(y|x, u) \quad x \in X \\
&\quad \mapsto \quad * \colon y \in X; r_{N-1}(x, u) \leq k(y) \\
v_n(x) &= \max_{u \in U} \sum_{*} v_{n+1}(y)p(y|x, u) \quad x \in X, \ n = N-2, \ldots, 1, 0. \\
&\quad \mapsto \quad * : y \in X; r_n(x, u) \leq r^*_n(y)
\end{align*}

where \( \pi^*_{N-1}, \pi^*_{N-2}, \ldots, \pi^*_1, \pi^*_0 \) are calculated backward; the first \( \pi^*_{N-1}(x) \) is a maximizer for (8) and the subsequent \( \pi^*_n(x) \) is a maximizer for (9). Further, \( r^*_{N-1}, r^*_{N-2}, \ldots, r^*_1 \) are successively defined through \( \pi^*_{N-1}, \pi^*_{N-2}, \ldots, \pi^*_1 \):

\begin{align*}
r^*_n(x) &= r_n(x, \pi^*_n(x)) \quad n = N-1, N-2, \ldots, 1.
\end{align*}

3.1 Decision-free Reward System

Now we consider the special case where the reward function \( r : X \times U \to R^1 \) is independent of decision variable \( n, u \):

\[ r(x, u) = r(x). \]

Thus the order probability is

\[ P^\pi_{x_0}(r_0 \leq r_1 \leq \cdots \leq k) := P^\pi_{x_0}(r(X_0) \leq r(X_1) \leq \cdots \leq r(X_{N-1}) \leq k(X_N)). \]

Then we have simplified results.

Corollary 3.1 For \( n \leq N-2 \) we have

\begin{align*}
P^\pi_{x_n}(r_n \leq r_{n+1} \leq \cdots \leq k) &= \sum_{*} P^\pi_{x_{n+1}}(r_{n+1} \leq \cdots \leq k)p(x_{n+1}|x_n, u_n) \\
&\quad \mapsto \quad * : x_{n+1} \in X; r(x_n) \leq r(x_{n+1})
\end{align*}

where \( u_n = \pi_n(x_n), \ \pi' = \{ \pi_{n+1}, \ldots, \pi_{N-1} \} \).

Further for \( \pi = \{ \pi_{N-1} \} \) we have

\begin{align*}
P^\pi_{x_{N-1}}(r_{N-1} \leq k) &= \sum_{*} P^\pi_{x_N}(k)p(x_N|x_{N-1}, u_{N-1}) \\
&\quad \mapsto \quad * : x_N \in X; r(x_{N-1}) \leq k(x_N)
\end{align*}

where \( u_{N-1} = \pi_{N-1}(x_{N-1}) \).

Corollary 3.2

\begin{align*}
v_N(x) &= 1 \quad x \in X \\
v_{N-1}(x) &= \max_{u \in U} \sum_{y \in X} v_N(y)p(y|x, u) \quad x \in X \\
&\quad \mapsto \quad \pi(x) \leq k(y) \\
v_n(x) &= \max_{u \in U} \sum_{y \in X} v_{n+1}(y)p(y|x, u) \quad x \in X, \ n = N-2, \ldots, 0.
\end{align*}
4 Reward Functions Depend on Tomorrow

In this section we treat the reward function which depends not only today but also on tomorrow. We consider both evaluation problem of order probability and optimization problem.

4.1 Evaluation

First we consider a recursive evaluation of order probability. Now let a sequence of reward functions

\[ r_n : X \times X \to \mathbb{R}^1 \quad (0 \leq n \leq N-1), \quad k : X \to \mathbb{R}^1 \]

be given. We note that the reward functions depend on next state.

Evaluate \[ P_0(r_0 \leq r_1 \leq \cdots \leq r_{N-1} \leq k) \]
under (i) \[ X_{n+1} \sim p(\cdot | x_n) \quad n = 0, \ldots, N-1 \]

Thus we evaluate the order probability

\[ v_0(x_0) := P_0(r_0 \leq r_1 \leq \cdots \leq r_{N-1} \leq k) \] (15)
under the Markov chain \( \{X_n\}_{0}^{N} \) with transition probability law \( p = \{p(\cdot | \cdot)\} \).

Let us consider the family of subproblems \( \{P_n(x_n, x_{n+1})\} \):

Evaluate \[ P_n(r_n \leq r_{n+1} \leq \cdots \leq r_{N-1} \leq k) \]
under (i) \[ X_{m+1} \sim p(\cdot | x_m) \quad m = n+1, \ldots, N-1 \]
\( (x_n, x_{n+1}) \in X \times X \), \( n = 0, \ldots, N-1 \).

Here we note that the evaluated order probability is the conditional probability :

\[
P_{x_n, x_{n+1}}(r_n \leq r_{n+1} \leq \cdots \leq r_{N-1} \leq k) = P(r_n(X_n, X_{n+1}) \leq r_{n+1}(X_{n+1}, X_{n+2}) \leq \cdots \leq r_{N-1}(X_{N-1}, X_N) \leq k(X_N) | (X_n, X_{n+1}) = (x_n, x_{n+1}))
\]

\[ = \sum_{\star} \sum_{x_{n+2}} \cdots \sum_{x_{N}} p(x_{n+2} | x_{n+1}) p(x_{n+3} | x_{n+2}) \cdots p(x_N | x_{N-1}, u_{N-1}) \]
\[ \leq \star \quad : r_n(x_n, x_{n+1}) \leq r_{n+1}(x_{n+1}, x_{n+2}) \leq \cdots \leq r_{N-1}(x_{N-1}, x_N) \leq k(x_N). \]

Let \( w_n(x_n, x_{n+1}) \) denote the probability of \( P_n(x_n, x_{n+1}) \), where

\[ w_N(x_N) := 1. \]

Then we have the recursive equation

Lemma 4.1

\[
w_N(x) = 1 \quad x \in X
\]

\[
w_{N-1}(x, y) = \begin{cases} 1 & \text{if } r_{N-1}(x, y) \leq k(y) \\ 0 & \text{otherwise} \end{cases} \quad (x, y) \in X \times X
\]

\[
w_n(x, y) = \sum_{\star} w_{n+1}(y, z)p(z | y) \quad (x, y) \in X \times X, \quad 0 \leq n \leq N-1
\] (16)
The desired probability (15) is given by
\[ v_0(x_0) = \sum_{x_1 \in X} w_0(x_0, x_1)p(x_1|x_0) \]

4.2 Optimization

Second we consider a recursive optimization of order probability. Let reward functions
\[ r_n : X \times U \times X \to R^1 \quad (0 \leq n \leq N-1), \quad k : X \to R^1 \]
be dependent on well next state as current decision. We optimize the order probability
\[ P_x(r_0 \leq r_1 \leq \cdots \leq r_{N-1} \leq k) \] (17)
under the controlled Markov chain \{(X_n, U_n)\} with transition probability law \( p = \{p(\cdot|\cdot, \cdot)\} \). The problem is which class we optimize in and how we can find an optimal policy. The preceding discussion on evaluation enables us to choose an policy class where any policy is a sequence of decision functions:
\[ \gamma_n : X \times \Lambda_n \to U \quad 2 \leq n \leq N-1. \]

Let us introduce the sequence of yesterday (last) reward sets \( \{\Lambda_n(x_n)\} \) to current state \( x_n \):
\[ \Lambda_0(x_0) \triangleq \{\lambda_0 | \lambda_0 = -\infty\} \]
\[ \Lambda_n(x_n) \triangleq \left\{ \lambda_n \left| \begin{array}{c}
\lambda_n = r_{n-1}(x_{n-1}, u_{n-1}, x_n) \\
(x_{n-1}, u_{n-1}) \in X \times U
\end{array} \right. \right\} \] (18)
\[ x_n \in X, \quad n = 1, \ldots, N. \]

Further we define yesterday (last) reward set
\[ \Lambda_n := \bigcup_{x \in X} \Lambda_n(x) \]
and the sequence of expanded state spaces \( \{Y_n\} \):
\[ Y_n := \{y_n = (x_n, \lambda_n) | x_n \in X, \lambda_n \in \Lambda_n(x_n)\} \quad n = 0, \ldots, N. \]

Let us define the corresponding random variable \( \tilde{\Lambda}_n \) by
\[ \tilde{\Lambda}_0 \triangleq \lambda_0 \]
\[ \tilde{\Lambda}_n \triangleq r_{n-1}(X_{n-1}, U_{n-1}, X_n) \quad n = 1, \ldots, N. \]

which takes values in \( \Lambda_n \).

Now we introduce a new controlled Markov chain on the expanded state spaces \( \{Y_n\}_0^N \). Here the state variables \( \{(X_n; \tilde{\Lambda}_n)\} \) behave such that the first component \( \{X_n\} \) obeys the original Markov transition law \( p \) and the second \( \{\tilde{\Lambda}_n\} \) follows the stochastic dynamics \( \tilde{\Lambda}_{n+1} := r_n(x_n, u_n, X_{n+1}) \). When the decision-maker chooses a decision \( u_n(\in U) \) on \( (x_n; \lambda_n)(\in Y_n) \) at \( n \)-th stage, the next state random variable \( (X_{n+1}; \tilde{\Lambda}_{n+1}) \) will take \( (x_{n+1}; \lambda_{n+1}) \) with probability
\[ p(x_{n+1}|x_n, u_n) \] at \((n+1)\)-st stage, where \( \lambda_{n+1} = r_n(x_n, u_n, x_{n+1}) \). Thus this is expressed by a coupled dynamics \( (i)_n, (i)'_n \) \( 0 \leq n \leq N-1 \).

Thus we maximize the order probability over the new controlled chain on expanded state spaces as follows:

\[
\begin{align*}
\text{maximize} & \quad P_{x_0}^* (\lambda_0 \leq r_0 \leq r_1 \leq \cdots \leq r_{N-1} \leq k) \\
\text{subject to} & \quad (i)_n x_{n+1} \sim p(\cdot|x_n, u_n) \\
& \quad (i)'_n \tilde{\Lambda}_{n+1} = r_n(x_n, u_n, X_{n+1}) \quad 0 \leq n \leq N-1 \\
& \quad (ii)_n u_n \in U
\end{align*}
\]

where \( \lambda_0 = -\infty \).

We imbed \( P_0(x_0; \lambda_0) \) into the family of subproblems \( \{ P_n(x_n; \lambda_n) \} \):

\[
\begin{align*}
\text{maximize} & \quad P_{x_n}^* (\lambda_n \leq r_n \leq r_{n+1} \leq \cdots \leq r_{N-1} \leq k) \\
\text{subject to} & \quad (i)_m x_{m+1} \sim p(\cdot|x_m, u_m) \\
& \quad (i)'_m \tilde{\Lambda}_{m+1} = r_m(x_m, u_m, X_{m+1}) \quad n \leq m \leq N-1 \\
& \quad (ii)_m u_m \in U
\end{align*}
\]

where \( x_n \in X, \lambda_n \in \Lambda_n(x_n) \) and \( 0 \leq n \leq N-1 \).

Let \( v_n(x_n, \lambda_n) \) denote the maximum value of \( P_n(x_n; \lambda_n) \), where we set

\[ v_N(x_N, \lambda_N) := P_{x_N} (\lambda_N \leq k(x_N)) \]

Then we have the recursive equation:

**Theorem 4.1**

\[
v_N(x; \lambda) = \begin{cases} 
1 & \text{if } \lambda \leq k(x) \\
0 & \text{otherwise}
\end{cases} \quad x \in X, \lambda \in \Lambda_N(x)
\]

\[
v_{N-1}(x; \lambda) = \max_{x \in U} \sum_y v_N(y; r_{N-1}(x, u, y)) p(y|x, u) \quad x \in X, \lambda \in \Lambda_{N-1}(x) \quad (19)
\]

\[
v_n(x; \lambda) = \max_{x \in U} \sum_y v_{n+1}(y; r_n(x, u, y)) p(y|x, u) \quad x \in X, \lambda \in \Lambda_n(x) \quad n = N-2, \ldots, 0 \quad (20)
\]

where \( \pi_{N-1}^*, \pi_{N-2}^*, \ldots, \pi_1^*, \pi_0^* \) are calculated backward; the first \( \pi_{N-1}^*(x; \lambda) \) is a maximizer for (19) and the subsequent \( \pi_n^*(x; \lambda) \) is a maximizer for (20).

**References**


