

AN OPTIMAL STOPPING PROBLEM ON TREE

Hisano, Hiroshi
Department of Information, Tohwa University

<https://doi.org/10.5109/16773>

出版情報 : Bulletin of informatics and cybernetics. 39, pp.45-57, 2007-12. Research Association of Statistical Sciences

バージョン :

権利関係 :

AN OPTIMAL STOPPING PROBLEM ON TREE

by

Hiroshi HISANO

*Reprinted from the Bulletin of Informatics and Cybernetics
Research Association of Statistical Sciences, Vol.39*



FUKUOKA, JAPAN
2007

AN OPTIMAL STOPPING PROBLEM ON TREE

By

Hiroshi HISANO*

Abstract

This paper deals with an optimal stopping problem where both transition weights and terminal rewards are given on tree with a root. It is shown that the optimal stopping rule can be constructed by backward induction. It is also clarified that there is no need to take into the consideration the condition that not only the terminal rewards but also transition weights should be “ positive”.

Key Words and Phrases: Node, Optimal stopping, Principle of alternate optimality, Stopping rule, Tree.

1. Introduction

A general theory of optimal stopping has been established by Snell (1952), Chow, Robbins and Siegmund (1971), Shiryaev (1978) and others. A main concern was a class of infinite-horizon problems. Recently a finite-horizon problem has been frequently applied to mathematical finance, in particular to pricing of American option, e.g. Shiryaev (1999). On the other hand, the theory of optimal stopping is closely related to dynamic programming, e.g. Bellman (1957) and to Markov decision process, e.g. Howard (1960). A construction of optimal stopping rule for finite horizon problem is performed through backward induction, which is a fundamental idea of dynamic programming/Markov decision process. These results are mainly constructed on stochastic dynamic systems.

In this paper we consider a class of optimal stopping problems on finite trees. Thus systems we consider are not necessarily stochastic. Our tree has a transition weight on arc, which may take negative values. The transition weight is a generalized transition probability such as in stochastic dynamic programming or in Markov decision processes. This report, which is continued from Iwamoto(1977), Iwamoto(2002) and Hisano (2002), presents a recursive result under a most relaxed condition.

2. Tree, node and stopping rules

In this paper we are concerned with optimal stopping on finite trees. For the sake of simplicity, we choose a typical tree and present our results there. Our results are easily transliterated onto a general tree, which takes much space for definitions and notations.

Let us now consider a typical tree T with one root s_0 (a on the graph below).

* Department of Information, TOHWA University Fukuoka 815-8510, Japan. tel +81-92-541-1152
his@tohwa-u.ac.jp

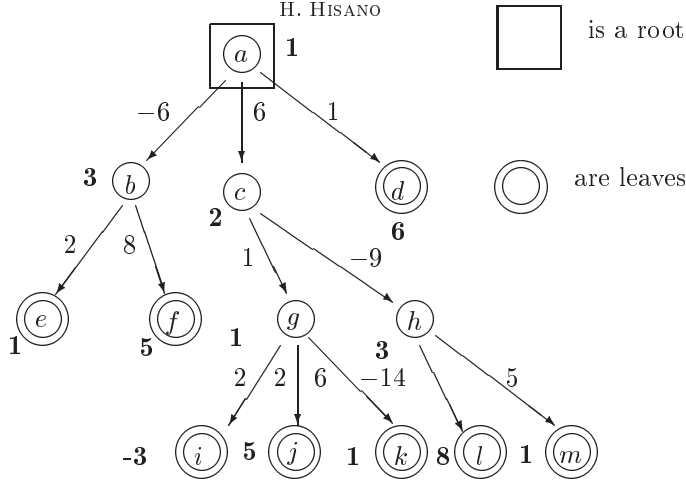


Fig1: An illustration

For $x, y \in T$, when node x is adjacent to node y , we describe it $x \leftarrow y$ or $y \rightarrow x$ and furthermore such an adjacent path itself shall be described with the same sign.

Example 2.1 On Fig 1, the whole adjacent paths are composed of $a \rightarrow b, a \rightarrow c, a \rightarrow d, b \rightarrow e, b \rightarrow f, c \rightarrow g, c \rightarrow h, g \rightarrow i, g \rightarrow j, g \rightarrow k, h \rightarrow l, h \rightarrow m$.

We call the nodes on the bottom row, e, f, i, j, k, l, m , as well as d , leaves, and a path from the root to a leaf (a line of adjacent paths) a full path. On Fig 1, there are eight full paths: $a \rightarrow b \rightarrow e, a \rightarrow b \rightarrow f, a \rightarrow c \rightarrow g \rightarrow i, a \rightarrow c \rightarrow g \rightarrow j, a \rightarrow c \rightarrow g \rightarrow k, a \rightarrow c \rightarrow h \rightarrow l, a \rightarrow c \rightarrow h \rightarrow m, a \rightarrow d$, which corresponds with the number of the leaves. A part below the node x of tree T is called a sub tree and indicated as $T(x)$.

Example 2.2 On Fig 1, $T = T(a), T = \{a\} \cup T(b) \cup T(c) \cup T(d)$, and $T(b), T(c)$ and $T(d)$ are sub trees. A sub tree with two levels is called a branch. $T(b), T(g)$ and $T(h)$ are branches of T .

2.1. Stopping rule

Definition 2.1 A stopping rule, which is a sub set of T , is constructed inductively as follows: If $T(x)$ is two levels tree then the stopping rule of $T(x)$ is $\{x\}, \{s_1, s_2, \dots, s_k\}$. We denote the set of stopping rules for T , $S(T(x))$. This means $S(T(x)) = \{\{x\} \cup \{s_1, s_2, \dots, s_k\}\} = \{\{x, s_1, s_2, \dots, s_k\}\}$, and, in other words, it is either a point set of the root or a set of all leaves. When a sub tree with three levels $T(x)$ is constructed of several branches and it is described as $T(x) = \{x\} \cup \{T(s_1), T(s_2), \dots, T(s_k)\}$, the stopping rule for $T(x)$ is $S(T(x)) = \{\{x\}, \bigcup_{i=1}^l S(T(s_i))\}$. In this context $\bigcup_{i=1}^l$ means an union of some $S(T(s_i))$.

Example 2.3 On Fig 1, $S(T(c)) = \{\{c\} \cup \{S(T(g)) \cup S(T(h))\}\}$.

Because $S(T(g)) = \{g\} \cup \{i, j, k\}$ and $S(T(h)) = \{h\} \cup \{l, m\}$,

$S(T(g)) \cup S(T(h)) = \{\{g, h\}, \{g, l, m\}, \{i, j, k, h\}, \{i, j, k, l, m\}\}$.

Consequently we have $S(T(c)) = \{\{c\}, \{g, h\}, \{g, l, m\}, \{i, j, k, h\}, \{i, j, k, l, m\}\}$.

In this manner, the stopping rule Γ for $T = T(a)$ is defined by backward induction.

Note that Γ is a sub set of the power set of T (e.g. $\Gamma \subset 2^T$).

Otherwise, the stopping rule can also be defined as follows:

Definition 2.2 *The stopping rule for $T(a)$ is a set of subsets of $T(a)$ and gives only one common node among full paths.*

This definition of the stopping rule clarifies the following:

The total number of the stopping rule

Theorem 2.1

$\#S(T(x)) = 2$, if $T(x)$ is two levels tree.

$$\#S(T(x)) = 1 + \prod_{i=1}^l \#S(T(s_i))$$

PROOF. It is obvious that the total number of the stopping rule $S(T(x))$ for a branch $T(x)$ is 2. The total number of the stopping rule $S(T(x))$ for a sub tree

$$T(x) = \{x\} \cup \{T(s_1), T(s_2), \dots, T(s_l)\}$$

is

$$1 + \prod_{i=1}^l \#S(T(s_i))$$

because

$$S(T(x)) = \{\{x\}, \bigcup_{i=1}^l S(T(s_i))\}.$$

□

2.2. Recurrence reward function

For any stopping rule γ , the reward $u(\gamma)$ is defined as follows:

$$u(\gamma) = \sum_{s \in \gamma} g(a, x_1)g(x_1, x_2) \cdots g(x_k, s)t(s),$$

where x_1, x_2, \dots, x_k are the nodes from a to s , and $g(\cdot, \cdot)$ are transition weights.

Definition 2.3 For $T(x) = \{x\} \cup \{T(s_1), T(s_2), \dots, T(s_l)\}$, let

$$\begin{aligned} V(x) &= \text{Max} \left\{ \sum_{g(x,s) \geq 0} g(x,s)V(s) + \sum_{g(x,s) < 0} g(x,s)v(s), t(x) \right\}, \\ v(x) &= \text{min} \left\{ \sum_{g(x,s) \geq 0} g(x,s)v(s) + \sum_{g(x,s) < 0} g(x,s)V(s), t(x) \right\}. \end{aligned}$$

where $t(x)$ is a terminal reward at x , and $g(x, s)$ is a transition weight in the from x to s (where $x \rightarrow s$). For a leaf $x \in T$, let $V(x) = v(x) = t(x)$.

Then, the following holds:

Theorem 2.2

$$V(a) \geq u(\gamma) \geq v(a) \text{ for } \forall \gamma \in \Gamma.$$

PROOF.

$$\begin{aligned} V(a) &= \text{Max}[g(a, b)v(b) + g(a, c)V(c) + g(a, d)V(d), t(a)] \\ &\geq g(a, b)v(b) + g(a, c)V(c) + g(a, d)V(d) \\ &\geq g(a, b)t(b) + g(a, c)t(c) + g(a, d)t(d) \\ V(b) &= \text{Max}[g(b, e)V(e) + g(b, f)V(f), t(b)] \\ &\geq g(b, e)V(e) + g(b, f)V(f) \\ &\geq g(b, e)t(e) + g(b, f)t(f) \\ V(c) &= \text{Max}[g(c, g)V(g) + g(c, h)v(h), t(c)] \\ &\geq g(c, g)V(g) + g(c, h)v(h) \\ &\geq g(c, g)t(g) + g(c, h)t(h). \end{aligned}$$

As these inequalities are held, for $\gamma = \{b, c, d\}$, for example,

$$V(a) \geq g(a, b)t(b) + g(a, c)t(c) + g(a, d)t(d) = u(\gamma).$$

Thus, $V(a) \geq u(\gamma)$ is provable for $\forall \gamma \in \Gamma$.

The essential point of this proof lies with the fact that $V(a)$ can be evaluated with a formula solely with the function t , as the following are held:

$$\begin{aligned} g(a, b) &< 0 \text{ implies } g(a, b)v(b) \geq g(a, b)t(b) \\ \text{and} \\ g(a, c) &> 0 \text{ implies } g(a, c)V(c) \geq g(a, c)t(c). \end{aligned}$$

$u(\gamma) \geq v(a)$ can be proved exactly in the similar manner, because

$$\begin{aligned} v(a) &= \min[g(a, b)V(b) + g(a, c)v(c) + g(a, d)v(d), t(a)] \leq g(a, b)V(b) + g(a, c)v(c) + \\ &g(a, d)v(d) \leq g(a, b)t(b) + g(a, c)t(c) + g(a, d)t(d) = u(\gamma). \end{aligned}$$

Theorem 2.3

$$V(a) = u(\gamma^*) \text{ for } \exists \gamma^* \in \Gamma, \quad v(a) = u(\gamma^*) \text{ for } \exists \gamma^* \in \Gamma. \quad (2.1)$$

PROOF. The whole nodes are divided into the first-row node $\{a\}$, the second-row nodes $\{b, c, d\}$, the third $\{e, f, g, h\}$ and the fourth $\{i, j, k, l, m\}$. The stopping rule γ^*

can be constructed for in the following procedure:

We define a stopping rule γ^* as follows.

(1) For the first-row node $\{a\}$, If $V(a) = t(a)$ then set $\gamma^* = \{a\}$ else next.

(2) Apply the following to the second-row nodes $\{b, c, d\}$ respectively:

(2-1) If $\{g(a, b) \geq 0 \text{ and } V(b) = t(b)\}$ or $\{g(a, b) < 0 \text{ and } v(b) = t(b)\}$, add b to γ^* .

(2-2) If $\{g(a, c) \geq 0 \text{ and } V(c) = t(c)\}$ or $\{g(a, c) < 0 \text{ and } v(c) = t(c)\}$, add c to γ^* .

(2-3) If $\{g(a, d) \geq 0 \text{ and } V(d) = t(d)\}$ or $\{g(a, d) < 0 \text{ and } v(d) = t(d)\}$, add d to γ^* .

In case of Fig 1, $V(d) = v(d) = t(d)$ and consequently it is evident that the node d is included in γ^* .

If $\{b, c, d\}$ are all added to γ^* in this procedure, $\gamma^* = \{b, c, d\}$. Otherwise, go to the next procedure (3). (Henceforth suppose that $\gamma^* \supset \{b, d\}$.)

(3) For the third $\{e, f, g, h\}$ belonging to a sub tree starting from a node which has not been added to γ^* by the procedure (2) (g and h are such nodes provided that $\gamma^* \supset \{b, d\}$ in the procedure (2)), apply the following respectively:

If $\{g(a, c)g(c, g) \geq 0 \text{ and } V(g) = t(g)\}$ or $\{g(a, c)g(c, g) < 0 \text{ and } v(g) = t(g)\}$, add g to γ^* .

If $\{g(a, c)g(c, h) \geq 0 \text{ and } V(h) = t(h)\}$ or $\{g(a, c)g(c, h) < 0 \text{ and } v(h) = t(h)\}$, add h to γ^* .

If both g and h are added to γ^* in this procedure, $\gamma^* = \{b, d\} \cup \{g, h\}$. Otherwise go to the next procedure (4).

(4) Since the fourth are all leaves, it is understood that: Then $g \notin \gamma^*$ implies $\{i, j, k\} \in \gamma^*$ and $h \notin \gamma^*$ implies $\{l, m\} \in \gamma^*$.

γ^* , constructed in this manner, satisfies $V(a) = u(\gamma^*)$

Now supposing that $\gamma^* = \{b, d, g, l, m\}$, the following formulae is held based on the foregoing construction:

- (i) $\{g(a, b) \geq 0 \text{ and } V(b) = t(b)\}$ or $\{g(a, b) < 0 \text{ and } v(b) = t(b)\}$
- (ii) $\{g(a, d) \geq 0 \text{ and } V(d) = t(d)\}$ or $\{g(a, d) < 0 \text{ and } v(d) = t(d)\}$
- (iii) $\{g(a, c)g(c, g) \geq 0 \text{ and } V(g) = t(g)\}$ or $\{g(a, c)g(c, g) < 0 \text{ and } v(g) = t(g)\}$
- (iv) $\{g(a, c)g(c, h) < 0 \text{ or } V(h) \neq t(h)\}$ and $\{g(a, c)g(c, h) \geq 0 \text{ or } v(h) \neq t(h)\}$

Out of these, (i) and (ii) are so self-explanatory that the explanation is omitted.

In addition, in consideration of Fig 1, when

$$g(a, b) < 0, \quad g(a, c)g(c, g) > 0, \quad g(a, c)g(c, h) < 0,$$

these conditions are equal to

$$v(b) = t(b), \quad V(g) = t(g), \quad v(h) \neq t(h).$$

Now that,

$$\begin{aligned}
u(\gamma^*) &= g(a, b)t(b) + g(a, d)t(d) + g(a, c)g(c, g)t(g) \\
&\quad + g(a, c)g(c, h)g(h, l)t(l) + g(a, c)g(c, h)g(h, m)t(m)u(\gamma^*) \\
&= g(a, b)v(b) + g(a, d)V(d) + g(a, c)g(c, g)V(g) + g(a, c)g(c, h)g(h, l)t(l) \\
&\quad + g(a, c)g(c, h)g(h, m)t(m)g(a, c)V(c) \\
&= g(a, c)g(c, g)V(g) + g(a, c)g(c, h)g(h, l)t(l) + g(a, c)g(c, h)g(h, m)t(m)
\end{aligned}$$

are given, we obtain ,

$$u(\gamma^*) = g(a, b)v(b) + g(a, d)V(d) + g(a, c)V(c) = V(a).$$

In other words, it is sufficient to prove,

$$V(c) = g(c, g)V(g) + g(c, h)g(h, l)t(l) + g(c, h)g(h, m)t(m).$$

Meanwhile, we learn ,

$$v(h) = g(h, l)t(l) + g(h, m)t(m)$$

since

$$v(h) \neq t(h),$$

and finally we have proven

$$v(c) = g(c, g)v(g) + g(c, h)V(h) = g(c, g)v(g) + g(c, h)g(h, l)t(l) + g(c, h)g(h, m)t(m).$$

Now we know that γ^* is always constructible with this logic. □

3. Dynamic programming

This subject can also be rehashed as a problem of dynamic programming with the terminal rewards, as illustrated below. On the following Fig 3, each node has an indication of an aggregate of the rewards on the path and the terminal, where

$$\begin{aligned}
t(a) &= 1, \quad t(b) = 3, \quad t(c) = 2, \quad t(d) = 6, \quad t(e) = 1, \quad t(f) = 5, \quad t(g) = 1, \\
t(h) &= 3, \quad t(i) = -3, \quad t(j) = 5, \quad t(k) = 1, \quad t(l) = 8, \quad t(m) = 1 \text{ (bold characters in Fig 3)}
\end{aligned}$$

e.g. the reward at e is computed by $g(e) = g(a, b)g(b, e)t(e) = (-6) \times 2 \times 1 = -12$

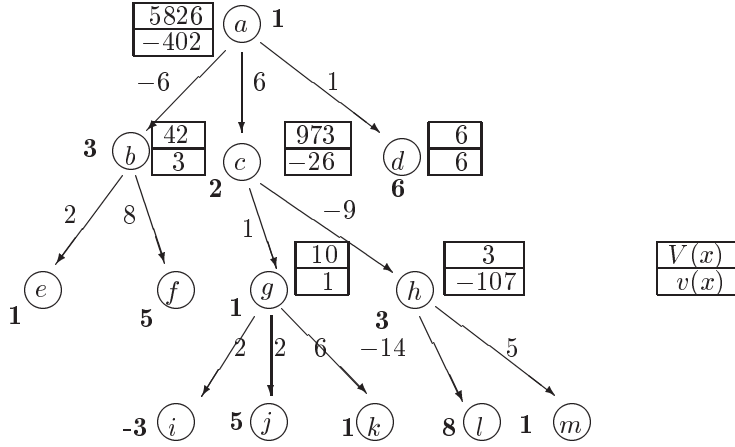


Fig 2: An original problem (bold : original terminal rewards)

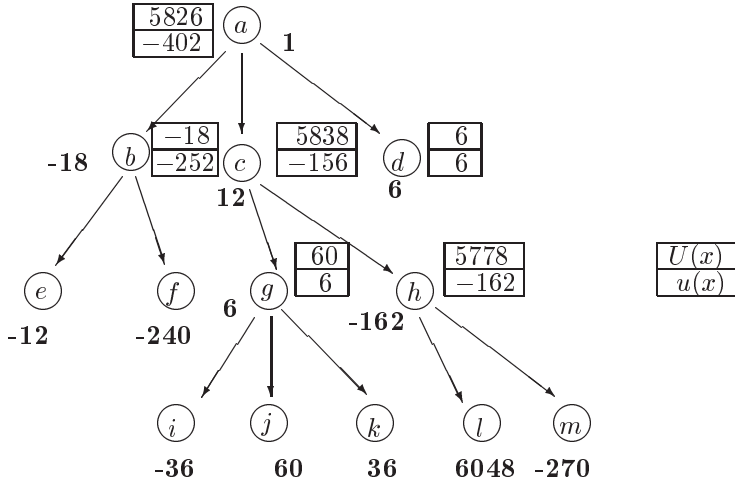


Fig 3: A rehashed dynamic programming (bold: aggregate terminal rewards)

Those two figures show that $U(a) = V(a), u(a) = v(a)$.

The label of each node shows the maximum (above) and the minimum (below) of the process reward up to that point. (In Fig 3 valuation is the sum of the labels of the adjacent nodes.) Thinking this way, we understand that the optimal stopping rule is a set of x which satisfies $U(x) = t(x)$. In Fig 3 the optimal stopping rule is $\{b, d, i, j, k, l, m\}$. As an aggregate terminal reward including the path information is given at each node, it is obvious that the maximum (or the minimum) of this problem of dynamic programming is identical to the maximum (or the minimum) of the original problem. The optimal stopping rule of a sub problem with a root of any node composing the optimal stopping

rule of the problem rehased as dynamic programming as above, is the node itself (namely a point set of root). (This is the principle of optimality.) Now we illustrate that the solution for the optimal stopping rule for original problem and that in the rehased problem are substantially the same and thus the optimal stopping rules in these two problems are identical.

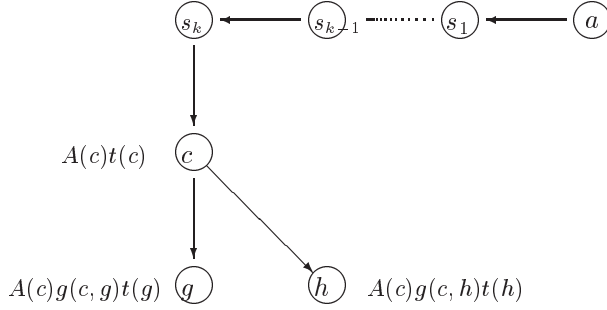


Fig 4: A rehased rewards

In the original problem, let the terminal reward at nodes c, g, h be $t(c), t(g), t(h)$ respectively, and let the product of path rewards from a to c , $g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)$, be $A(c)$. Then, the problem rehased as dynamic programming has the following terminal rewards:

Let that the label at the node c in the original problem be $\frac{V(c)}{v(c)}$ and that at the node c in the rehased problem be $\frac{U(c)}{u(c)}$, the following formulae hold between these labels:

Lemma 3.1 *If $g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c) > 0$, then*

$$\begin{aligned} V(c) &= \frac{U(c)}{g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c)} = \frac{U(c)}{A(c)}, \\ v(c) &= \frac{u(c)}{g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c)} = \frac{u(c)}{A(c)}. \end{aligned}$$

If $g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c) < 0$, then

$$\begin{aligned} V(c) &= \frac{u(c)}{g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c)} = \frac{u(c)}{A(c)}, \\ v(c) &= \frac{U(c)}{g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c)} = \frac{U(c)}{A(c)}. \end{aligned}$$

< Notation > Suppose $g(s_i, s_j) \neq 0$.

PROOF. We make a proof with backward induction, depending on the location of node c .

(1) In case node c is a leaf.

Since $U(c) = u(c) = g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c)t(c)$, $V(c) = v(c) = t(c)$, Lemma 3.1 holds.

(2) In case node c is not a leaf.

Suppose that for the nodes adjacent to node c , p_1, p_2, \dots, p_l , $g(c, p_1) > 0, g(c, p_2) > 0, \dots, g(c, p_l) > 0$, and for $q_1, q_2, \dots, q_m, g(c, q_1) < 0, g(c, q_2) < 0, \dots, g(c, q_m) < 0$.

When $g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c) > 0$,

$V(c)$

$$\begin{aligned} &= \text{Max} \left[\sum_{i=1}^l g(c, p_i)V(p_i) + \sum_{j=1}^m g(c, q_j)v(q_j), t(c) \right] \\ &= \text{Max} \left[\sum_{i=1}^l \frac{g(c, p_i)U(p_i)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, p_j)} + \sum_{j=1}^m \frac{g(c, q_j)U(q_j)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, q_j)}, t(c) \right] \end{aligned}$$

because

if $g(a, s_1)g(a, s_2) \cdots g(s_k, c) > 0$, $g(c, q_j) < 0$ we obtain $g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, q_j) < 0$ and thus

$V(q_j)$

$$\begin{aligned} &= \frac{U(q_j)}{g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c)g(c, q_j)} \\ &= \text{Max} \left[\sum_{i=1}^l \frac{U(p_i)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} + \sum_{j=1}^m \frac{U(q_j)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)}, t(c) \right] \\ &= \frac{1}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} \text{Max} \left[\sum_{i=1}^l U(p_i) + \sum_{j=1}^m U(q_j), g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)t(c) \right] \\ &= \frac{U(c)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)}. \end{aligned}$$

$v(c)$

$$\begin{aligned} &= \min \left[\sum_{i=1}^l g(c, p_i)v(p_i) + \sum_{j=1}^m g(c, q_j)V(q_j), t(c) \right] \\ &= \min \left[\sum_{i=1}^l \frac{g(c, p_i)u(p_i)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, p_j)} + \sum_{j=1}^m \frac{g(c, q_j)u(q_j)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, q_j)}, t(c) \right] \\ &= \min \left[\sum_{i=1}^l \frac{u(p_i)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} + \sum_{j=1}^m \frac{u(q_j)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)}, t(c) \right] \\ &= \frac{1}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} \min \left[\sum_{i=1}^l u(p_i) + \sum_{j=1}^m u(q_j), g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)t(c) \right] \end{aligned}$$

$$= \frac{u(c)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)},$$

thus Lemma 3.1 holds.

When $g(a, s_1) \cdots g(s_{k-1}, s_k)g(s_k, c) < 0$,

$$\begin{aligned} V(c) &= \text{Max} \left[\sum_{i=1}^l g(c, p_i)V(p_i) + \sum_{j=1}^m g(c, q_j)v(q_j), t(c) \right] \\ &= \text{Max} \left[\sum_{i=1}^l \frac{g(c, p_i)u(p_i)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, p_j)} + \sum_{j=1}^m \frac{g(c, q_j)u(q_j)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, q_j)}, t(c) \right] \\ &= \text{Max} \left[\sum_{i=1}^l \frac{u(p_i)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} + \sum_{j=1}^m \frac{u(q_j)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)}, t(c) \right] \\ &= \frac{1}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} \min \left[\sum_{i=1}^l u(p_i) + \sum_{j=1}^m u(q_j), g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)t(c) \right] \\ &= \frac{u(c)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} \end{aligned}$$

$$\begin{aligned} v(c) &= \min \left[\sum_{i=1}^l g(c, p_i)v(p_i) + \sum_{j=1}^m g(c, q_j)V(q_j), t(c) \right] \\ &= \min \left[\sum_{i=1}^l \frac{g(c, p_i)U(p_i)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, p_j)} + \sum_{j=1}^m \frac{g(c, q_j)U(q_j)}{g(a, s_1)g(a, s_2) \cdots g(s_k, c)g(c, q_j)}, t(c) \right] \\ &= \min \left[\sum_{i=1}^l \frac{U(p_i)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} + \sum_{j=1}^m \frac{U(q_j)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)}, t(c) \right] \\ &= \frac{1}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} \text{Max} \left[\sum_{i=1}^l U(p_i) + \sum_{j=1}^m U(q_j), g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)t(c) \right] \\ &= \frac{U(c)}{g(a, s_1)g(s_1, s_2) \cdots g(s_k, c)} \end{aligned}$$

and thus Lemma 3.1 holds. □

Lemma 3.2

$$U(a) = V(a), \quad u(a) = v(a)$$

PROOF. Let p_1, p_2, \dots, p_l be the nodes adjacent to node a such that

$$g(a, p_1) > 0, g(a, p_2) > 0, \dots, g(a, p_l) > 0,$$

and q_1, q_2, \dots, q_m , be the nodes adjacent to node a such that

$$g(a, q_1) < 0, g(a, q_2) < 0, \dots, g(a, q_m) < 0.$$

Since,

$$\begin{aligned} V(p_i) &= \frac{U(p_i)}{g(a, p_i)}, \quad v(p_i) = \frac{v(p_i)}{g(a, p_i)} \quad (1 \leq \forall i \leq l) \\ V(q_j) &= \frac{U(q_j)}{g(a, q_j)}, \quad v(q_j) = \frac{v(q_j)}{g(a, q_j)} \quad (1 \leq \forall j \leq m) \end{aligned}$$

It follows

$$\begin{aligned} V(a) &= \text{Max} \left[\sum_{i=1}^l g(a, p_i) V(p_i) + \sum_{j=1}^m g(a, q_j) v(q_j), \quad t(a) \right] \\ &= \text{Max} \left[\sum_{i=1}^l g(a, p_i) \frac{U(p_i)}{g(a, p_i)} + \sum_{j=1}^m g(a, q_j) \frac{U(q_j)}{g(a, q_j)}, \quad t(a) \right] \\ &= \text{Max} \left[\sum_{i=1}^l U(p_i) + \sum_{j=1}^m U(q_j), \quad t(a) \right] = U(a) \\ v(a) &= \min \left[\sum_{i=1}^l g(a, p_i) v(p_i) + \sum_{j=1}^m g(a, q_j) V(q_j), \quad t(a) \right] \\ &= \min \left[\sum_{i=1}^l g(a, p_i) \frac{u(p_i)}{g(a, p_i)} + \sum_{j=1}^m g(a, q_j) \frac{u(q_j)}{g(a, q_j)}, \quad t(a) \right] \\ &= \min \left[\sum_{i=1}^l u(p_i) + \sum_{j=1}^m u(q_j), \quad t(a) \right] = u(a) \end{aligned}$$

This completes the proof that the optimal rewards in the original problem and those in the rehashed problem are identical. \square

Theorem 3.1 When $g(a, s_1) \cdots g(s_{k-1}, s_k) g(s_k, c) > 0$, the following two are equivalent:

$$\begin{aligned} V(c) &= t(c) \text{ in the original problem} \\ U(c) &= g(a, s_1) \cdots g(s_{k-1}, s_k) g(s_k, c) t(c) \text{ in the rehashed problem} \end{aligned}$$

Also, when $g(a, s_1) \cdots g(s_{k-1}, s_k) g(s_k, c) < 0$, the following two are equivalent:

$$\begin{aligned} v(c) &= t(c) \text{ in the original problem} \\ u(c) &= g(a, s_1) \cdots g(s_{k-1}, s_k) g(s_k, c) t(c) \text{ in the rehashed problem} \end{aligned}$$

PROOF. It is evident from Lemma 3.1. \square

With this Theorem 3.1, the nodes included in the optimal stopping rule for original problem are included in the optimal stopping rule for rehashed problem and *vice versa*. Therefore, it holds that the optimal stopping rules in both problem are completely identical.

3.1. Principle of alternate optimality

Theorem 3.2 *In the original problem, principle of alternate optimality holds in terms of the following:*

If $\{s_1, s_2, \dots, s_k\}$ is the optimal stopping rule for the original problem, then the following hold:

Given that the nodes precedent to s_i ($1 \leq i \leq k$) be $a, b_{1(i)}, \dots, b_{m(i)}$ ($1 \leq i \leq k$).

- (1) *If $g(a, b_{1(i)})g(b_{1(i)}, b_{2(i)}) \cdots g(b_{m(i)}, s_i) > 0$,
 s_i is the maximum optimal stopping rule for the sub problem $T(s_i)$,*
- (2) *If $g(a, b_{1(i)})g(b_{1(i)}, b_{2(i)}) \cdots g(b_{m(i)}, s_i) < 0$,
 s_i is the minimum optimal stopping rule for the sub problem $T(s_i)$.*

PROOF. If Theorem 3.2 is not true, the total optimal value may increase or decrease by replacing the optimal stopping rule for the sub tree $T(s_i)$ with another, which is contradictory to the fact that $\{s_1, s_2, \dots, s_k\}$ is the optimal stopping rule for the original problem. \square

Theorem 3.3 *We construct the optimal stopping rule for the original problem as follows:*

For each node x , let

$$V(x) = \text{Max} \left(\sum_{g(x,s) \geq 0} g(x,s)V(s) + \sum_{g(x,s) < 0} g(x,s)v(s), t(x) \right],$$

$$v(x) = \text{min} \left(\sum_{g(x,s) \geq 0} g(x,s)v(s) + \sum_{g(x,s) < 0} g(x,s)V(s), t(x) \right],$$

by backward induction, Consider the whole set of node c satisfying

$$g(a, s_1)g(a, s_2) \cdots g(s_k, c) > 0 \text{ and } V(c) = t(c) \ (v(c) = t(c)),$$

$$g(a, s_1)g(a, s_2) \cdots g(s_k, c) < 0 \text{ and } v(c) = t(c) \ (V(c) = t(c)),$$

and, let the factors of such a set which are arranged to exist once on each full path be $\{s_1, s_2, \dots, s_k\}$.

Then, $\{s_1, s_2, \dots, s_k\}$ is the maximum (minimum) optimal stopping rule.

PROOF. We shall prove for the maximum case here, since the minimum case can be demonstrated in the same manner. In accordance with Theorem 3.2, each node s_i ($1 \leq i \leq k$) is the optimal stopping rule for sub tree $T(s_i)$.

Now we consider replacement of one node (e.g. s_1) out of $\{s_1, s_2, \dots, s_k\}$ to the set

$\{p_1, p_2, \dots, p_m\}$ such that $s_1 \rightarrow p_1, s_1 \rightarrow p_2, \dots, s_1 \rightarrow p_m$. Then, the reward of the sub problem $T(s_i)$ given by s_1 is greater than the reward given by $\{p_1, p_2, \dots, p_m\}$, and consequently we find that such replacement does not contribute to increase of the reward of the original problem $T(a)$.

On the other hand, consider replacement of s_1 with a precedent node p_1 (e.g. $s_1 \leftarrow p_1$). As s_1 is not necessarily the sole node among $\{s_1, s_2, \dots, s_k\}$ which is subsequent to p_1 , we assume that $\{s_1, s_2, \dots, s_l\}$ is subsequent to p_1 (e.g. $p_1 \rightarrow s_1, p_1 \rightarrow s_2, \dots, p_1 \rightarrow p_l$) and that $\{s_{l+1}, s_{l+2}, \dots, s_k\}$ is not subsequent to p_1 . Then, as $\{s_1, s_2, \dots, s_l\}$ is the optimal stopping rule for the sub problem $T(p_1)$, the reward of the sub problem $T(p_1)$ given by $\{p_1\}$ is not greater than the reward given by $\{s_1, s_2, \dots, s_l\}$, and consequently we find that such replacement does not contribute to increase of the reward of the original problem $T(a)$. From the above, we find that the stopping rule $\{s_1, s_2, \dots, s_k\}$ constructed in this manner is the maximum optimal stopping rule. \square

Acknowledgement

The author would like to thank an anonymous referee for usefull suggestions.

References

- Bellman, R.E. (1957). *Dynamic programming*, NJ:Princeton Univ. Press.
- Chow, Y.S., Robbins, H. and Siegmund, D. (1971). *Great Expectations: The Theory of Optimal Stopping*, Boston: Houghton Mifflin Company.
- Hisano, H. (2002). Optimal stopping problem on finite Markov chain, *Bull. Inform. Cybernet.* **34**, no. 2, 97–104.
- Howard, R.A. (1960). *Dynamic programming and Markov Processes*, Mass.: MIT Press.
- Iwamoto, S. (1977). The second principal of optimality, *Bull. Math. Statist.* **17**, no.3-4, 101–114.
- Iwamoto, S. (2002). *Optimal stopping in fuzzy environment*, Proc. 9-th Bellman Continuum, Beijing, 264-269.
- Shiryaev, A.N. (1978). *Optimal Stopping Rules*, New York: Springer-Verlag.
- Shiryaev, A.N. (1999). *Essentials of Stochastic Finance*, Singapore: World Scientific.
- Snell, J.L. (1952). Applications of martingale system theorems, *Transactions of the American Mathematical Society* **73**, 171–176.

Received November 15, 2006

Revised August 24, 2007