数理解析研究所講究録
1079 巻 1999 年 153-163

分数型評価のマルコフ決定過程

九大・経済　岩本誠一
九工大・工　藤田敏治

Abstract

We consider how to optimize a ratio of two expected values of additive statistics on a finite-state controlled Markov chain. We present an algorithm for finding an optimal policy by use of both stochastic dynamic programming and fractional programming.

1 Introduction

We are concerned with finding an optimal policy which maximizes a ratio of two expected values of additive rewards over a controlled Markov decision process ([7],[8]).

2 Fractional Expectation Problem

Throughout the paper, the following data is given:

$$\begin{array}{l} N \geq 1 \text{ is an integer; the } \textit{total number of stages} \\ S = \{s_1,\, s_2, \ldots, s_p\} \text{ is a finite } \textit{state space} \\ A = \{a_1,\, a_2, \ldots, a_k\} \text{ is a finite } \textit{action space} \\ r : S \times A \to R^1, \quad R : S \times A \to (0,\infty) \text{ are two } \textit{n-th reward functions} \\ k : S \to R^1, \;\; K : S \to (0,\infty) \text{ are two } \textit{terminal reward functions} \\ \beta \text{ is a } \textit{discount factor} \;:\; 0 < \beta < 1 \\ p \text{ is a } \textit{Markov transition law} \\ \quad : \;\; p(y|x,u) \geq 0 \;\; \forall(x,u,y) \in S \times A \times S, \quad \sum_{y \in S} p(y|x,u) = 1 \;\; \forall(x,u) \in S \times A \\ \quad y \sim p(\cdot\,|x,u) \text{ denotes that next state } y \text{ conditioned on state } x \text{ and action } u \\ \quad \text{appears with probability } p(y|x,u). \end{array} \tag{1}$$

We use the following simple notations:

$$\begin{array}{l} r_n := r(X_n,U_n), \quad R_n := R(X_n,U_n) \quad 1 \leq n \leq N \\ r_{N+1} := k(X_{N+1}), \quad R_{N+1} := K(X_{N+1}) \\ E_{x_n}[Y] := E[Y|X_n = x_n]. \end{array} \tag{2}$$

Let $c \in R^1$ be a given constant (level). Then we consider how to maximize the ratio of the expected value of one *additive* statistics

$$r(X_1,U_1) + r(X_2,U_2) + \cdots + r(X_N,U_N) + k(X_{N+1})$$

to that of the other

$$R(X_1,U_1) + R(X_2,U_2) + \cdots + R(X_N,U_N) + K(X_{N+1}).$$

A *Markov* policy $\pi = \{\pi_1, \pi_2, ..., \pi_N\}$ is a finite sequence of decision functions:

$$\pi_n : S \to A \quad 1 \le n \le N. \tag{3}$$

The set of all Markov policies is denoted by Π. Given an initial state $x_1 \in S$, let us consider the maximization problem:

$$\mathrm{F}(x_1) \qquad \text{Maximize} \quad \frac{E^{\pi}_{x_1}\left[\sum_{n=1}^{N+1} r_n\right]}{E^{\pi}_{x_1}\left[\sum_{n=1}^{N+1} R_n\right]} \quad \text{subject to} \quad \text{(i)} \quad \pi \in \Pi. \tag{4}$$

By introducing the Lagrange multiplier λ, the fractional optimization problem (4) is transformed into the standard stochastic optimization problem with the following additive criteria:

$$\begin{array}{lll} & \text{Maximize} & E^{\pi}_{x_1}\left[\sum_{n=1}^{N+1}(r_n - \lambda R_n)\right] \qquad (5) \\ \mathrm{P}(x_1;\lambda) & \text{subject to} & \text{(i)} \;\; x_{n+1} \sim p(\cdot|x_n, u_n) \quad 1 \le n \le N \\ & & \text{(ii)} \;\; u_n \in A \quad 1 \le n \le N \\ & & \qquad x_1 \in S, \;\; \lambda \in R^1, \;\; 1 \le n \le N+1. \end{array}$$

Let $u_n(x_n;\lambda)$ be the maximum value of the subproblem:

$$\begin{array}{lll} & \text{Maximize} & E^{\pi}_{x_n}\left[\sum_{m=n}^{N+1}(r_m - \lambda R_m)\right] \qquad (6) \\ \mathrm{P}_n(x_n;\lambda) & \text{subject to} & \text{(i)} \;\; x_{m+1} \sim p(\cdot|x_m, u_m) \quad n \le m \le N \\ & & \text{(ii)} \;\; u_m \in A \quad n \le m \le N \\ & & \qquad x_n \in S, \;\; \lambda \in R^1, \;\; 1 \le n \le N+1. \end{array}$$

Then we have the recursive equation([4]):

THEOREM 2.1

$$\begin{aligned} u_n(x;\lambda) &= \operatorname*{Max}_{u \in A}[r(x,u) - \lambda R(x,u) + \sum_{y \in S} u_{n+1}(y;\lambda) p(y|x,u)] \qquad (7) \\ & \qquad\qquad x \in S, \;\; \lambda \in R^1, \;\; 1 \le n \le N \\ u_{N+1}(x;\lambda) &= k(x) - \lambda K(x) \qquad x \in S, \;\; \lambda \in R^1. \end{aligned}$$

3 Infinite-stage Problem

In this section we consider an optimization problem of the ratio of one total discounted expected value over an infinite-stage to the other as follows:

$$\mathrm{F}'(x_1) \qquad \text{Maximize} \quad \frac{E^{\pi}_{x_1}\left[\sum_{n=1}^{\infty} \beta^{n-1} r_n\right]}{E^{\pi}_{x_1}\left[\sum_{n=1}^{\infty} \beta^{n-1} R_n\right]} \quad \text{subject to} \quad \text{(i)} \quad \pi \in \Pi \tag{8}$$

where

$$\sum_{n=1}^{\infty} \beta^{n-1} r_n = r(X_1, U_1) + \beta r(X_2, U_2) + \cdots + \beta^{n-1} r(X_n, U_n) + \cdots$$
$$\sum_{n=1}^{\infty} \beta^{n-1} R_n = R(X_1, U_1) + \beta R(X_2, U_2) + \cdots + \beta^{n-1} R(X_n, U_n) + \cdots.$$

Here Π is the set of all Markov policies, whose element $\pi = \{\pi_1, \pi_2, \ldots, \pi_n, \ldots\}$ is an infinite sequence of decision functions :

$$\pi_n : S \to A \quad n = 1,\ 2, \ldots. \tag{9}$$

An introduction of Lagrange multiplier λ reduces the fractional optimization problem (8) to a standard discounted dynamic programming problem ([3],[5],[6],[9]) as follows:

$$\begin{array}{lll} & \text{Maximize} & E_{x_1}^{\pi}\left[\sum_{n=1}^{\infty} \beta^{n-1}(r_n - \lambda R_n)\right] \\ \mathrm{P}'(x_1;\lambda) & \text{subject to} & \text{(i)}\ \ x_{n+1} \sim p(\cdot|x_n, u_n) \quad n = 1,\ 2, \ldots \\ & & \text{(ii)}\ \ u_n \in A \quad n = 1,\ 2, \ldots \\ & & \qquad x_1 \in S, \ \ \lambda \in R^1. \end{array} \tag{10}$$

Let $u(x_1;\lambda)$ be the maximum value of the problem (10). Then we have the recursive equation:

THEOREM 3.1

$$u(x;\lambda) = \operatorname*{Max}_{u\in A}[r(x,u) - \lambda R(x,u) + \beta \sum_{y\in S} u(y;\lambda) p(y|x,u)] \tag{11}$$
$$x \in S, \ \ \lambda \in R^1.$$

4 Fractional Programming Approach

In this section we solve the fractional expectation problems (4) and (8) through both fractional programming and dynamic programming.

4.1 Fractional Programming

Let us review two fundamental results on fractional programming. We consider the following problem:

$$\text{Fr} \qquad \text{Maximize} \quad \frac{f(z)}{g(z)} \quad \text{subject to} \quad z \in Z \tag{12}$$

where Z is a nonempty set and $f : Z \to R^1, \ \ g : Z \to (0, \infty)$. It is well-known that the fractional programming problem Fr is associated with the following parametric problem:

$$\mathrm{Pr}(\lambda) \qquad \text{Maximize} \quad f(z) - \lambda g(z) \quad \text{subject to} \quad z \in Z. \tag{13}$$

THEOREM 4.1 *([11]) The fractional problem* Fr *has an optimal solution $z^* \in Z$ if and only if the parametric problem* $\mathrm{Pr}(\lambda)$ *has the optimal solution $z^* \in Z$ for some parameter λ and the optimal value vanishes.*

Let us consider Dinkelbach's Algorithm:

- Step 1. Select some $z \in Z$ and set $n = 1$, $z_{(1)} = z$ and $\lambda_{(1)} = \dfrac{f(z)}{g(z)}$.
- Step 2. Solve $\mathrm{Pr}(\lambda_{(n)})$ and select some optimal solution $z \in Z$.
- Step 3. If $f(z) - \lambda_{(n)}g(z) = 0$, set $z' = z$ and $\lambda' = \dfrac{f(z)}{g(z)}$, and stop. Otherwise, set $z_{(n+1)} = z$ and $\lambda_{(n+1)} = \dfrac{f(z)}{g(z)}$.
- Step 4. Set $n = n + 1$ and go to Step 2.

THEOREM 4.2 *([11]) Either Dinkelbach's Algorithm terminates in some finite n-th iteration, in which case z' is an optimal solution and λ' is a maximum value of* Fr, *or else the sequence $\{\lambda_{(n)}\}$ converges strict-monotonically to the maximum value of* Fr. *Termination is assured if Z is finite.*

We remark that the convergence is in fact superlinear. If Dinkelbach's Algorithm generates a finite sequence $\{\lambda_{(k)}\}_{1\le k\le n}$ with properties

$$\text{(i)} \quad \lambda_{(1)} < \lambda_{(2)} < \cdots < \lambda_{(n-1)} < \lambda_{(n)},$$

(ii) $f(z) - \lambda_{(n)}g(z) = 0$ for some optimal solution $z \in Z$ of $\mathrm{Pr}(\lambda_{(n)})$, (iii) $z' = z$, and (iv) $\lambda' = \dfrac{f(z)}{g(z)}$, and terminates, then the z is an optimal solution and $\lambda_{(n)}$ is the maximum value of Fr.

4.2 Fractional Expectation Problems

First let us consider the fractional expectation problem (4) by use of fractional programming ([1]) and dynamic programming. The problem (4) is formulated as the following fractional programming problem:

$$\mathrm{Fr}(x_1) \qquad \text{Maximize } \frac{f(\pi;x_1)}{g(\pi;x_1)} \quad \text{subject to } \pi \in \Pi \tag{14}$$

where Π is the set of N-stage Markov policies and

$$\begin{aligned} f(\pi;x_1) &= E^{\pi}_{x_1}\left[\sum_{n=1}^{N+1} r_n\right] \\ g(\pi;x_1) &= E^{\pi}_{x_1}\left[\sum_{n=1}^{N+1} R_n\right]. \end{aligned}$$

Then the corresponding parametric problem reduces to:

$$\mathrm{Pr}(x_1)(\lambda) \qquad \text{Maximize } f(\pi;x_1) - \lambda g(\pi;x_1) \quad \text{subject to } \pi \in \Pi. \tag{15}$$

THEOREM 4.3 *For each initial state $x_1 \in X$, Dinkelbach's Algorithm yields a Markov policy π^*, which is optimal at x_1:*

$$\frac{E_{x_1}^{\pi^*}\left[\sum_{n=1}^{N+1} r_n\right]}{E_{x_1}^{\pi^*}\left[\sum_{n=1}^{N+1} R_n\right]} \geq \frac{E_{x_1}^{\pi}\left[\sum_{n=1}^{N+1} r_n\right]}{E_{x_1}^{\pi}\left[\sum_{n=1}^{N+1} R_n\right]} \quad \forall \pi \in \Pi. \tag{16}$$

Proof Since Π is finite, Theorems 4.1 and 4.2 apply. □

Second we consider the infinite-stage problem (8). By taking in turn

$$f(\pi; x_1) = E_{x_1}^{\pi}\left[\sum_{n=1}^{\infty} \beta^{n-1} r_n\right]$$
$$g(\pi; x_1) = E_{x_1}^{\pi}\left[\sum_{n=1}^{\infty} \beta^{n-1} R_n\right],$$

we have a stationary policy which is optimal at a given initial state.

THEOREM 4.4 *For each state $x_1 \in X$, Dinkelbach's Algorithm yields a stationary policy $\pi^* = h^{(\infty)}$, which is optimal at x_1:*

$$\frac{E_{x_1}^{\pi^*}\left[\sum_{n=1}^{\infty} \beta^{n-1} r_n\right]}{E_{x_1}^{\pi^*}\left[\sum_{n=1}^{\infty} \beta^{n-1} R_n\right]} \geq \frac{E_{x_1}^{\pi}\left[\sum_{n=1}^{\infty} \beta^{n-1} r_n\right]}{E_{x_1}^{\pi}\left[\sum_{n=1}^{\infty} \beta^{n-1} R_n\right]} \quad \forall \pi \in \Pi \tag{17}$$

where $h : S \to A$ is a stage-free decision function of π^*:

$$h^{(\infty)} = \{h, h, \ldots, h, \ldots\}.$$

Proof Let Π_{st} be the set of all stationary policies. Then we see that $\Pi_{st} \subset \Pi$ and Π_{st} is finite. We restrict the fractional problem (14) to Π_{st}. Then Theorems 4.1 and 4.2 apply. In fact, the corresponding parametric problem (15) is a discounted dynamic programming problem in the sense of D. Blackwell ([3]). Thus it has an optimal stationary policy. □

5 A 2-2 Decision Models

In this section, we illustrate a two-state and two-action decision model.

5.1 A 2-2-2 Decision Model

As an illustrative example we consider the following two-stage problem:

$$\mathrm{F}(x_1) \quad \begin{array}{ll} \text{Maximize} & \dfrac{E_{x_1}^{\pi}[r(x_1, u_1) + r(X_2, U_2) + k(X_3)]}{E_{x_1}^{\pi}[R(x_1, u_1) + R(X_2, U_2) + K(X_3)]} \\ \text{subject to} & \text{(i)} \ \ x_{n+1} \sim p(\cdot|x_n, u_n) \quad 1 \leq n \leq 2 \\ & \text{(ii)} \ \ u_n \in A \quad 1 \leq n \leq 2 \end{array} \tag{18}$$

on the following data:

stage rewards : $(r(x_t,u_t),\ R(x_t,u_t))$

$x_t \diagdown u_t$	a_1	a_2
s_1	(0, 2)	(1, 1)
s_2	(−1, 3)	(2, 2)

terminal rewards

x_3	$(k(x_3),\ K(x_3))$
s_1	(1 , 2)
s_2	(0 , 1)

transition law

$P(a_1) = \{p(x_{t+1}|x_t, a_1)\}$

$x_t \backslash x_{t+1}$	s_1	s_2
s_1	1/2	1/2
s_2	0	1

$P(a_2) = \{p(x_{t+1}|x_t, a_2)\}$

$x_t \backslash x_{t+1}$	s_1	s_2
s_1	1	0
s_2	1/4	3/4

Thus we have the following parametric data:

stage reward : $r(x_t,u_t) - \lambda R(x_t,u_t)$

$x_t \diagdown u_t$	a_1	a_2
s_1	$0-2\lambda$	$1-\lambda$
s_2	$-1-3\lambda$	$2-2\lambda$

terminal reward

x_3	$k(x_3) - \lambda K(x_3)$
s_1	$1-2\lambda$
s_2	$0-\lambda$

Then the recursive equation

$$\begin{aligned} u_3(x;\lambda) &= k(x) - \lambda K(x) \\ u_2(x;\lambda) &= \operatorname*{Max}_{u\in A}\left[r(x,u) - \lambda R(x,u) + \sum_{y\in S} u_3(y;\lambda)p(y|x,u)\right] \\ u_1(x;\lambda) &= \operatorname*{Max}_{u\in A}\left[r(x,u) - \lambda R(x,u) + \sum_{y\in S} u_2(y;\lambda)p(y|x,u)\right] \\ & \qquad\qquad x \in S,\quad \lambda \in R^1 \end{aligned} \tag{19}$$

together with the suffixed notations

$$u_n(\lambda) := u_n(s_1;\lambda), \qquad v_n(\lambda) := u_n(s_2;\lambda)$$

$$k_i := k(s_i), \quad K_i := K(s_i), \quad r_i^k := r(s_i, a_k), \quad R_i^k := R(s_i, a_k), \quad p_{ij}^k := p(s_j|s_i, a_k)$$

reduces to:

$$\begin{aligned} u_3(\lambda) &= k_1 - \lambda K_1 \\ v_3(\lambda) &= k_2 - \lambda K_2 \\ u_n(\lambda) &= \left[r_1^1 - \lambda R_1^1 + p_{11}^1 u_{n+1}(\lambda) + p_{12}^1 v_{n+1}(\lambda)\right] \\ &\qquad \vee \left[r_1^2 - \lambda R_1^2 + p_{11}^2 u_{n+1}(\lambda) + p_{12}^2 v_{n+1}(\lambda)\right] \\ v_n(\lambda) &= \left[r_2^1 - \lambda R_2^1 + p_{21}^1 u_{n+1}(\lambda) + p_{22}^1 v_{n+1}(\lambda)\right] \\ &\qquad \vee \left[r_2^2 - \lambda R_2^2 + p_{21}^2 u_{n+1}(\lambda) + p_{22}^2 v_{n+1}(\lambda)\right] \qquad n = 1,\ 2. \end{aligned} \tag{20}$$

Then Eq.(20) becomes:

$$\begin{aligned} u_3(\lambda) &= 1-2\lambda \\ v_3(\lambda) &= 0-\lambda \\ u_n(\lambda) &= \left[0-2\lambda+\frac{1}{2}u_{n+1}(\lambda)+\frac{1}{2}v_{n+1}(\lambda)\right]\vee\left[1-\lambda+u_{n+1}(\lambda)\right] \\ v_n(\lambda) &= \left[-1-3\lambda+v_{n+1}(\lambda)\right]\vee\left[2-2\lambda+\frac{1}{4}u_{n+1}(\lambda)+\frac{3}{4}v_{n+1}(\lambda)\right] \quad n=1,\,2. \end{aligned}$$

Thus we have

$$\begin{aligned} u_2(\lambda) &= \left[\frac{1}{2}-\frac{7}{2}\lambda\right]\vee[2-3\lambda] = \begin{cases} \frac{1}{2}-\frac{7}{2}\lambda, & -\infty<\lambda\le-3 \\ 2-3\lambda, & -3\le\lambda<\infty \end{cases} \\ v_2(\lambda) &= [-1-4\lambda]\vee\left[\frac{9}{4}-\frac{13}{4}\lambda\right] = \begin{cases} -1-4\lambda, & -\infty<\lambda\le-\frac{13}{3} \\ \frac{9}{4}-\frac{13}{4}\lambda, & -\frac{13}{3}\le\lambda<\infty \end{cases} \end{aligned}$$

$$u_1(\lambda) = \begin{cases} -\frac{1}{4}-\frac{23}{4}\lambda, & -\infty<\lambda\le-\frac{13}{3} \\ \frac{11}{8}-\frac{43}{8}\lambda, & -\frac{13}{3}\le\lambda\le-3 \\ \frac{17}{8}-\frac{41}{8}\lambda, & -3\le\lambda\le-\frac{7}{9} \\ 3-4\lambda, & -\frac{7}{9}\le\lambda<\infty \end{cases}$$

$$v_1(\lambda) = \begin{cases} -2-7\lambda, & -\infty<\lambda\le-\frac{13}{3} \\ \frac{5}{4}-\frac{25}{4}\lambda, & -\frac{13}{3}\le\lambda\le-\frac{47}{17} \\ \frac{67}{16}-\frac{83}{16}\lambda, & -\frac{47}{17}\le\lambda<\infty \end{cases}$$

Then the desired optimal policy $\pi^*(\lambda)=\{\pi_1^*(\lambda),\pi_2^*(\lambda)\}$ where

$$\pi_n^*(\lambda)=\begin{bmatrix} \pi_n^*(s_1;\lambda) \\ \pi_n^*(s_2;\lambda) \end{bmatrix} \tag{21}$$

is specified as follows :

$$\pi_2^*(\lambda)=\begin{cases} \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}, & -\infty<\lambda\le-\frac{13}{3} \\ \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, & -\frac{13}{3}\le\lambda\le-3 \\ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, & -3\le\lambda<\infty \end{cases} \qquad \pi_1^*(\lambda)=\begin{cases} \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}, & -\infty<\lambda\le-\frac{47}{17} \\ \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, & -\frac{47}{17}\le\lambda\le-\frac{7}{9} \\ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, & -\frac{7}{9}\le\lambda<\infty \end{cases} \tag{22}$$

By applications of Dinkelbach's Algorithm from $\pi=\left\{\begin{bmatrix} a_1 \\ a_1 \end{bmatrix},\begin{bmatrix} a_1 \\ a_1 \end{bmatrix}\right\}$, we have optimal solutions as follows:

CASE(I) Algorithm I for $x_1 = s_1$.

1. Select $\pi_1 = \left\{ \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}, \begin{bmatrix} a_1 \\ a_1 \end{bmatrix} \right\} \in \Pi$. Then $\lambda_{(1)} = \dfrac{f(\pi_1; s_1)}{g(\pi_1; s_1)} = \dfrac{-1/4}{23/4} = -\dfrac{1}{23}$.

2. Solve $\Pr\left(-\dfrac{1}{23}\right)$ and select unique optimal solution $\pi_2 = \left\{ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, \begin{bmatrix} a_2 \\ a_2 \end{bmatrix} \right\} \in \Pi$. Then $f(\pi_2; s_1) - \lambda_{(1)} g(\pi_2; s_1) = 3 - \left(-\dfrac{1}{23}\right) \cdot 4 = \dfrac{72}{23} \neq 0$. Hence $\lambda_{(2)} = \dfrac{f(\pi_2; s_1)}{g(\pi_2; s_1)} = \dfrac{3}{4}$.

3. Solve $\Pr\left(\dfrac{3}{4}\right)$ and select unique optimal solution $\pi^* = \left\{ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, \begin{bmatrix} a_2 \\ a_2 \end{bmatrix} \right\} \in \Pi$. Then $f(\pi^*; s_1) - \lambda_{(2)} g(\pi^*; s_1) = 3 - \dfrac{3}{4} \cdot 4 = 0$. Thus $\pi^* = \pi_2$ is an optimal at s_1 and $\lambda_{(2)} = \dfrac{3}{4}$ is the desired maximum value.

CASE(II) Algorithm I for $x_1 = s_2$.

1. Select $\pi_1 = \left\{ \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}, \begin{bmatrix} a_1 \\ a_1 \end{bmatrix} \right\} \in \Pi$. Then $\lambda_{(1)} = \dfrac{f(\pi_1; s_2)}{g(\pi_1; s_2)} = \dfrac{-2}{7}$.

2. Solve $\Pr\left(-\dfrac{2}{7}\right)$ and select unique optimal solution $\pi_2 = \left\{ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, \begin{bmatrix} a_2 \\ a_2 \end{bmatrix} \right\} \in \Pi$. Then $f(\pi_2; s_2) - \lambda_{(1)} g(\pi_2; s_2) = \dfrac{67}{16} - \left(-\dfrac{2}{7}\right) \cdot \dfrac{83}{16} = \dfrac{635}{112} \neq 0$. Hence $\lambda_{(2)} = \dfrac{f(\pi_2; s_2)}{g(\pi_2; s_2)} = \dfrac{67/16}{83/16} = \dfrac{67}{83}$.

3. Solve $\Pr\left(\dfrac{67}{83}\right)$ and select unique optimal solution $\pi^* = \left\{ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, \begin{bmatrix} a_2 \\ a_2 \end{bmatrix} \right\} \in \Pi$. Then $f(\pi^*; s_2) - \lambda_{(1)} g(\pi^*; s_2) = \dfrac{67}{16} - \dfrac{67}{83} \cdot \dfrac{83}{16} = 0$. Thus $\pi^* = \pi_2$ is also optimal at s_2 and $\lambda_{(2)} = \dfrac{67}{83}$ is the desired maximum value.

Therefore, the resulting stationary policy $\pi^* = \left\{ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, \begin{bmatrix} a_2 \\ a_2 \end{bmatrix} \right\}$ is optimal (for both states) and the optimal ratio vectors is $\begin{pmatrix} 3/4 \\ 67/83 \end{pmatrix}$.

5.2 A 2-2-∞ Decision Model

Now we consider the corresponding infinite-stage problem on the two-state and two-action model:

$$\mathrm{F}'(x_1) \qquad \text{Maximize} \quad \frac{E^{\pi}_{x_1}\left[\sum_{n=1}^{\infty} \beta^{n-1} r_n\right]}{E^{\pi}_{x_1}\left[\sum_{n=1}^{\infty} \beta^{n-1} R_n\right]} \quad \text{subject to} \quad \text{(i)} \quad \pi \in \Pi \tag{23}$$

where $\beta = 0.8$. Then the recursive equation for the corresponding parametric problem

$$u(x; \lambda) = \max_{u \in A} \left[r(x, u) - \lambda R(x, u) + \beta \sum_{y \in S} u(y; \lambda) p(y|x, u) \right] \tag{24}$$
$$x \in S, \quad \lambda \in R^1$$

together with the suffixed notations

$$u(\lambda) := u(s_1;\lambda), \quad v(\lambda) := u(s_2;\lambda)$$

$$r_i^k := r(s_i, a_k), \quad R_i^k := R(s_i, a_k), \quad p_{ij}^k := p(s_j|s_i, a_k)$$

reduces to:

$$\begin{aligned} u(\lambda) &= \left[r_1^1 - \lambda R_1^1 + \beta(p_{11}^1 u(\lambda) + p_{12}^1 v(\lambda))\right] \vee \left[r_1^2 - \lambda R_1^2 + \beta(p_{11}^2 u(\lambda) + p_{12}^2 v(\lambda))\right] \\ v(\lambda) &= \left[r_2^1 - \lambda R_2^1 + \beta(p_{21}^1 u(\lambda) + p_{22}^1 v(\lambda))\right] \vee \left[r_2^2 - \lambda R_2^2 + \beta(p_{21}^2 u(\lambda) + p_{22}^2 v(\lambda))\right]. \end{aligned} \tag{25}$$

Then Eq.(25) reduces to:

$$\begin{aligned} u(\lambda) &= \left[0 - 2\lambda + \frac{4}{5}\left(\frac{1}{2}u(\lambda) + \frac{1}{2}v(\lambda)\right)\right] \vee \left[1 - \lambda + \frac{4}{5}u(\lambda)\right] \\ v(\lambda) &= \left[-1 - 3\lambda + \frac{4}{5}v(\lambda)\right] \vee \left[2 - 2\lambda + \frac{4}{5}\left(\frac{1}{4}u(\lambda) + \frac{3}{4}v(\lambda)\right)\right] \end{aligned}$$

namely

$$\begin{aligned} [-10\lambda - 3u(\lambda) + 2v(\lambda)] \vee [5 - 5\lambda - u(\lambda)] &= 0 \\ [-5 - 15\lambda - v(\lambda)] \vee [10 - 10\lambda + u(\lambda) - 2v(\lambda)] &= 0. \end{aligned}$$

This system of two function equations has the following unique solution:

$$u(\lambda) = \begin{cases} -\frac{10}{3} - \frac{40}{3}\lambda, & -\infty < \lambda \leq -\frac{5}{2} \\ 5 - 10\lambda, & -\frac{5}{2} \leq \lambda \leq 0 \\ 5 - 5\lambda, & 0 \leq \lambda < \infty \end{cases} \qquad v(\lambda) = \begin{cases} -5 - 15\lambda, & -\infty < \lambda \leq -\frac{5}{2} \\ \frac{15}{2} - 10\lambda, & -\frac{5}{2} \leq \lambda \leq 0 \\ \frac{15}{2} - \frac{15}{2}\lambda, & 0 \leq \lambda < \infty. \end{cases}$$

Then the desired optimal policy $\pi^*(\lambda) = h^{(\infty)}(\lambda)$ where

$$h(\lambda) = \begin{bmatrix} h(s_1;\lambda) \\ h(s_2;\lambda) \end{bmatrix} \tag{26}$$

is specified as follows :

$$h(\lambda) = \begin{cases} \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}, & -\infty < \lambda \leq -\frac{5}{2} \\ \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, & -\frac{5}{2} \leq \lambda \leq 0 \\ \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}, & 0 \leq \lambda < \infty \end{cases} \tag{27}$$

By applications of Dinkelbach's Algorithm from $\pi = h^{(\infty)}$ with $h = \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}$, we have the following optimal solutions:

CASE(I) Algorithm II for $x_1 = s_1$.

1. Select $\pi_1 = h_1^{(\infty)} \in \Pi_{st}$ with $h_1 = \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}$. Then $\lambda_{(1)} = \dfrac{f(\pi_1; s_1)}{g(\pi_1; s_1)} = \dfrac{-15/3}{-40/3} = -\dfrac{3}{8}$.

2. Solve $\Pr\left(-\dfrac{3}{8}\right)$ and select optimal solution $\pi_2 = h_2^{(\infty)} \in \Pi_{st}$ with $h_2 = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$. Then $f(\pi_2; s_1) - \lambda_{(1)} g(\pi_2; s_1) = 5 - \left(-\dfrac{3}{8}\right) \cdot 10 = \dfrac{35}{4} \neq 0$. Hence $\lambda_{(2)} = \dfrac{f(\pi_2; s_1)}{g(\pi_2; s_1)} = \dfrac{5}{10} = \dfrac{1}{2}$.

3. Solve $\Pr\left(\dfrac{1}{2}\right)$ and select optimal solution $\pi_3 = h_3^{(\infty)} \in \Pi_{st}$ with $h_3 = \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}$. Then $f(\pi_3; s_1) - \lambda_{(2)} g(\pi_3; s_1) = 5 - \dfrac{1}{2} \cdot 5 = \dfrac{5}{2} \neq 0$. Hence $\lambda_{(3)} = \dfrac{f(\pi_3; s_1)}{g(\pi_3; s_1)} = \dfrac{5}{5} = 1$.

4. Solve Pr(1) and select optimal solution $\pi^* = h_*^{(\infty)} \in \Pi_{st}$ with $h_* = \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}$.

5. Then $f(\pi^*; s_1) - \lambda_{(3)} g(\pi^*; s_1) = 5 - 1 \cdot 5 = 0$. Thus $\pi^* = \pi_3$ is an optimal at s_1 and $\lambda_{(3)} = 1$ is the desired maximum value.

CASE(II) Algorithm II for $x_1 = s_2$.

1. Select $\pi_1 = h_1^{(\infty)} \in \Pi_{st}$ with $h_1 = \begin{bmatrix} a_1 \\ a_1 \end{bmatrix}$. Then $\lambda_{(1)} = \dfrac{f(\pi_1; s_2)}{g(\pi_1; s_2)} = \dfrac{-5}{15} = -\dfrac{1}{3}$.

2. Solve $\Pr\left(-\dfrac{1}{3}\right)$ and select optimal solution $\pi_2 = h_2^{(\infty)} \in \Pi_{st}$ with $h_2 = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$. Then $f(\pi_2; s_2) - \lambda_{(1)} g(\pi_2; s_2) = \dfrac{15}{2} - \left(-\dfrac{1}{3}\right) \cdot 10 = \dfrac{65}{6} \neq 0$. Hence $\lambda_{(2)} = \dfrac{f(\pi_2; s_2)}{g(\pi_2; s_2)} = \dfrac{15/2}{10} = \dfrac{3}{4}$.

3. Solve $\Pr\left(\dfrac{1}{2}\right)$ and select optimal solution $\pi_3 = h_3^{(\infty)} \in \Pi_{st}$ with $h_3 = \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}$. Then $f(\pi_3; s_2) - \lambda_{(2)} g(\pi_3; s_2) = \dfrac{15}{2} - \dfrac{3}{4} \cdot \dfrac{15}{2} = \dfrac{15}{8} \neq 0$. Hence $\lambda_{(3)} = \dfrac{f(\pi_3; s_2)}{g(\pi_3; s_2)} = \dfrac{15/2}{15/2} = 1$.

4. Solve Pr(1) and select optimal solution $\pi^* = h_*^{(\infty)} \in \Pi_{st}$ with $h_* = \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}$.

5. Then $f(\pi^*; s_2) - \lambda_{(3)} g(\pi^*; s_2) = \dfrac{15}{2} - 1 \cdot \dfrac{15}{2} = 0$. Thus $\pi^* = \pi_3$ is also an optimal at s_2 and $\lambda_{(3)} = 1$ is also the desired maximum value.

On the other hand, applications from $\pi = h^{(\infty)}$ with $h = \begin{bmatrix} a_2 \\ a_1 \end{bmatrix}$ yields the following results:

CASE(III) Algorithm II for $x_1 = s_1$.

1. Select $\pi_1 = h_1^{(\infty)} \in \Pi_{st}$ with $h_1 = \begin{bmatrix} a_2 \\ a_1 \end{bmatrix}$. Then $\lambda_{(1)} = \dfrac{f(\pi_1; s_1)}{g(\pi_1; s_1)} = \dfrac{5}{5} = 1$. From CASE(I), the desired maximum value is 1. Thus the policy π_1 is also optimal at s_1.

2. Solve Pr(1) and select optimal solution $\pi_2 = h_2^{(\infty)} \in \Pi_{st}$ with $h_2 = \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}$. Then $f(\pi_2; s_1) - \lambda_{(1)} g(\pi_2; s_1) = 5 - 1 \cdot 5 = 0$. Thus $\pi^* = \pi_2$ is optimal at s_1 and $\lambda_{(2)} = 1$ is the desired maximum value.

CASE(IV) Algorithm II for $x_1 = s_2$.

1. Select $\pi_1 = h_1^{(\infty)} \in \Pi_{st}$ with $h_1 = \begin{bmatrix} a_2 \\ a_1 \end{bmatrix}$. Then $\lambda_{(1)} = \dfrac{f(\pi_1; s_2)}{g(\pi_1; s_2)} = \dfrac{-5}{15} = -\dfrac{1}{3}$.

2. Solve $\Pr\left(-\dfrac{1}{3}\right)$ and select unique optimal solution $\pi_2 = h_2^{(\infty)} \in \Pi_{st}$ with $h_2 = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$.
Hereafter CASE(II) follows. Thus, $\pi^* = \pi_3$ is also an optimal at s_2 and $\lambda_{(3)} = 1$ is also the desired maximum value. Thus the policy π_1 is not optimal at s_2.

Therefore, the resulting stationary policy $\pi^* = h_1^{(\infty)}$ with $h_1 = \begin{bmatrix} a_2 \\ a_2 \end{bmatrix}$ is optimal (for both states) and the optimal ratio vectors is $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$. Furthermore, the stationary policy $\pi^{**} = h_2^{(\infty)}$ with $h_2 = \begin{bmatrix} a_2 \\ a_1 \end{bmatrix}$ is optimal at s_2.

References

[1] A.I. Barros, *Discrete and Fractional Programming Techiniques for Location Models*, Kluwer Academic Publishers, Dordrecht, 1998.

[2] R.E. Bellman, *Dynamic Programming*, Princeton Univ. Press, NJ, 1957.

[3] D. Blackwell, Discounted dynamic programming, Ann. Math. Stat. **36**(1965), 226-235.

[4] T. Fujita, Re-examination of Markov Policies for Additive Decision Process, Bull. Infor. Cyber., 29(1997), 51-65.

[5] K. Hinderer, *Foundations of Non-Stationary Dynamic Programming with Discrete Time Parameter*, Lect. Notes in Operation Research and Mathematical Systems, Vol. 33, Springer-Verlag, Berlin, 1970.

[6] R. A. Howard, *Dynamic Programming and Markov Processes*, MIT Press, Cambridge, Mass., 1960.

[7] S. Iwamoto, On expected values of Markov statistics, Bull. Infor. Cyber., 30(1998), 1-24.

[8] S. Iwamoto and T. Fujita, On conditional expected values of Markov statistics, under preparation.

[9] M.L. Puterman, *Markov Decision Processes : discrete stochastic dynamic programming*, Wiley & Sons, New York, 1994.

[10] M. Sniedovich, Analysis of a class of fractional programming problems, Math. Prog. **43**(1989), 329-347.

[11] M. Sniedovich, *Dynamic Programming*, Marcel Dekker, Inc. NY, 1992.