数理解析研究所講究録 1263 巻 2002 年 1-12

Stochastic optimal weighting problem

九州大学大学院経済学研究院 岩 本 誠 一 (Seiichi Iwamoto)
九州大学大学院経済学府 植 野 貴 之 (Takayuki Ueno)
Graduate School of Economics, Kyushu University

1 Introduction

In this paper, we consider a class of optimal weighting problems. R. Bellman [1, p.136] has proposed a threshold probability optimization problem. We study the problem and its related problems through final state approach in dynamic programming [1, p.82], [11, p.71], [6–9]. Introducing weighted sum and weighted minimum for Bernoulli sequence, we optimize expected value, variance and threshold probability over the total unit sum.

In section 2, we consider the optimization problem of expected value and variance for the additive statistics. In Section 3, we solve the corresponding threshold probability problem. Section 4 discusses the minimum criterion. We transform the three stochastic problems into equivalent deterministic ones. Further the stochastic problems are solved by one-dimensional state-expansion in dynamic programming. The last section concludes that the final state approach is valid for corresponding problems on Markov chain.

2 Expected value and variance

A sequence of random variables $Y_1,\ Y_2, \ldots,\ Y_n, \ldots$ is called *Bernoulli*, if it is independent and identically distributed with

$$P(Y_n = 1) = p, \quad P(Y_n = 0) = q \qquad (0 \le p \le 1,\ p + q = 1).$$

Then the expected value and variance are :

$$E[Y_n] = p, \qquad V[Y_n] = pq.$$

Given a finite Bernoulli sequence $\{Y_1,\ Y_2, \ldots,\ Y_N\}$, we consider expected value, variance and threshold probability of two weighted statistics — additive and minimum — :

$$x_1Y_1 + x_2Y_2 + \cdots + x_NY_N, \qquad x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_NY_N$$

where $x = (x_1, x_2, \ldots, x_N)$ is a weight. The weight x is called *feasible* if it satisfies the two constraints :

$$\begin{array}{ll} \text{(i)} & x_1 + x_2 + \cdots + x_N = 1 \\ \text{(ii)} & x_n \in [0,1] \quad n = 1, 2, \ldots, N. \end{array}$$

The problem is to find a feasible weight which optimizes a criterion function (expected value, variance or threshold probability). We show that dynamic programming method supplies such an optimal weight.

2.1 Maximizing expected value

First we consider an optimal weighting problem as follows :

$$\mathsf{E}(1) \qquad \begin{array}{ll} \text{Max} & E[\,x_1Y_1 + x_2Y_2 + \cdots + x_NY_N\,] \\ \text{s.t.} & \text{(i)} \;\; x_1 + x_2 + \cdots + x_N = 1 \\ & \text{(ii)} \;\; x_n \in [0,1] \quad n = 1,2,\ldots,N. \end{array}$$

The linearity of expectation and condition (i) imply

$$E[\,x_1Y_1 + \cdots + x_NY_N\,] = p.$$

Thus any feasible $x = (x_1, \ldots, x_N)$ yields the value p. Therefore, the maximum value is p and all feasible points are maximum point.

Let us now consider dynamic programming approach. Let $f_n(d_n)$ be the maximum value of

$$\mathsf{E}_n(d_n) \qquad \begin{array}{ll} \text{Max} & E[\,x_nY_n + x_{n+1}Y_{n+1} + \cdots + x_NY_N\,] \\ \text{s.t.} & \text{(i)}_{\text{n}} \;\; x_n + x_{n+1} + \cdots + x_N = d_n \\ & \text{(ii)}_{\text{n}} \;\; x_m \in [0,1] \quad m = n, n+1, \ldots, N \end{array}$$

$$0 \le d_n \le 1, \qquad 1 \le n \le N.$$

Thus the maximum value of $\mathsf{E}(1)$ is given by $f_1(1)$. The sequence of maximum value functions $\{f_n\}$ satisfies the backward recursive formula :

$$\begin{cases} f_N(d) \;=\; pd & 0 \le d \le 1 \\ f_n(d) \;=\; \underset{0 \le x \le d}{\text{Max}}\,[px + f_{n+1}(d-x)] & 0 \le d \le 1, \;\; 1 \le n \le N-1. \end{cases} \tag{1}$$

Let $\pi_n^*(d)$ be the maximizer in (1). Then the sequence $\pi^* = \{\pi_n^*\}$ is an optimal policy. In fact, solving (1), we have the sequence of maximum value functions $\{f_n\}$ and an optimal policy π^*, where

$$f_n(d) = pd \quad 1 \le n \le N, \qquad \pi_n^*(d) = \text{any} \in [0,\, d]. \tag{2}$$

The pair of sequence of maximum value functions and an optimal policy yields the optimal solution (maximum value and maximum point) of expectation problem $\mathsf{E}(1)$:

$$f_1(1) = p, \qquad x^* \text{ is any feasible point.} \tag{3}$$

2.2 Minimizing variance

Second we consider the optimal weighting problem for variance :

$$\mathsf{V}(1) \qquad \begin{array}{ll} \text{min} & V[\,x_1Y_1 + x_2Y_2 + \cdots + x_NY_N\,] \\ \text{s.t.} & \text{(i)} \;\; x_1 + x_2 + \cdots + x_N = 1 \\ & \text{(ii)} \;\; x_n \in [0,1] \quad n = 1,2,\ldots,N. \end{array}$$

From the independence and Schwarz's inequality we have

$$V[\,x_1Y_1 + \cdots + x_NY_N\,] \;\ge\; \frac{pq}{N}.$$

Therefore the minimum value $\hat{u}$ is $\hat{u} = pq/N$ and the minimum point $\hat{x}$ is $\hat{x} = (1/N, \ldots, 1/N)$.

In fact, the linear-quadratic minimization problem

$$\min \quad x_1^2 + \cdots + x_N^2 \quad \text{s.t.} \quad \text{(i)}, \text{(ii)},$$

is solved through dynamic programming [1–4]. Letting

$$f_N(c) := \min[\, x_1^2 + \cdots + x_N^2 \,|\, x_1 + \cdots + x_N = c, \; x_n \geq 0 \;\; 1 \leq n \leq N \,] \quad c \geq 0, N \geq 1.$$

Then we have the recursive equation

$$f_N(c) = \min_{0 \leq x \leq c} [\, x^2 + f_{N-1}(c - x) \,] \quad c \geq 0, N \geq 2, \qquad f_1(c) = c^2.$$

Successively solving the equation we have the sequence of minimum value functions $\{f_1, \ldots, f_N\}$ and the optimal policy (sequence of optimal decision functions) $\hat{\sigma} = \{\hat{\sigma}_1, \ldots, \hat{\sigma}_N\}$:

$$f_n(c) = c^2/n \qquad \hat{\sigma}_n(c) = c/n.$$

Hence $\mathsf{V}(1)$ has the minimum value $f_N(1) = 1/N$. The minimum point $\hat{x} = (\hat{x}_1, \ldots, \hat{x}_N) = (1/N, \ldots, 1/N)$ is calculated through the optimal policy $\hat{\sigma}$ with the deterministic transformation $T(c; x) = c - x$ [4, p.13].

Finally we apply dynamic programming method. Let $h_n(d_n)$ be the minimum value of

$$\mathsf{V}_n(d_n) \qquad \begin{array}{ll} \min & V[\, x_n Y_n + \cdots + x_N Y_N \,] \\ \text{s.t.} & \text{(i)}_n \;\; x_n + \cdots + x_N = d_n \\ & \text{(ii)}_n \;\; x_m \in [0, 1] \quad m = n, n+1, \ldots, N \end{array}$$

$$0 \leq d_n \leq 1, \qquad 1 \leq n \leq N.$$

Then we have

$$\begin{cases} h_N(d) \;=\; pqd^2 & 0 \leq d \leq 1 \\ h_n(d) \;=\; \displaystyle\min_{0 \leq x \leq d} [pqx^2 + h_{n+1}(d - x)] & 0 \leq d \leq 1, \;\; 1 \leq n \leq N-1. \end{cases} \tag{4}$$

Then we have the minimum value functions $\{h_n\}$ and an optimal policy $\pi^* = \{\pi_n^*\}$, where

$$h_n(d) = \frac{pqd^2}{N - n + 1} \quad 1 \leq n \leq N, \qquad \pi_n^*(d) = \frac{d}{N - n + 1}. \tag{5}$$

This pair yields the desired optimal solution ; minimum value $\hat{u} = h_1(1)$ and minimum point $\hat{x}$.

3 Maximizing threshold probability

First we describe the probability function of random variable

$$Z := Z_N := x_1 Y_1 + x_2 Y_2 + \cdots + x_N Y_N.$$

Let us define the range Z takes

$$\mathcal{Z} := \{\, z = x_1 y_1 + \cdots + x_N y_N \,|\, (y_1, \cdots, y_N) \in \{0, 1\}^N \,\} \subset [0.1].$$

Then the probability function is defined by

$$P(Z=z)=\sum_{y:*}p^{y_1}q^{1-y_1}\cdots p^{y_N}q^{1-y_N} \qquad z\in\mathcal{Z}$$

where $y:*$ denotes the summation over all $(y_1,\cdots,y_N)\in\{0,1\}^N$ satisfying

$$x_1y_1+\cdots+x_Ny_N=z.$$

Then for any given constant upper level value $c\in[0,1]$ we consider the threshold probability problem as follows [1, p.136,137]:

$$\mathrm{P}(1) \qquad \begin{array}{ll} \text{Max} & P(\,x_1Y_1+x_2Y_2+\cdots+x_NY_N\geq c\,) \\ \text{s.t.} & \text{(i)}\;\; x_1+x_2+\cdots+x_N=1 \\ & \text{(ii)}\;\; x_n\in[0,1] \quad n=1,2,\ldots,N. \end{array}$$

We remark that the threshold probability is expressed in terms of multiple sum :

$$P(Z_N\geq c)=\sum_{y:**}p^{y_1}q^{1-y_1}\cdots p^{y_N}q^{1-y_N}$$

where $y:**$ denotes the summation over all $(y_1,\cdots,y_N)\in\{0,1\}^N$ satisfying

$$x_1y_1+\cdots+x_Ny_N\geq c.$$

Further we note that the thresold probability depends on the weight $x=(x_1,\ldots,x_N)$ which corresponds to the sequence of decisions :

$$P(Z_N\geq c)=P^x(Z_N\geq c).$$

In particular, the two-variable problem

$$\begin{array}{ll} \text{Max} & P(\,x_1Y_1+x_2Y_2\geq c\,) \\ \text{s.t.} & \text{(i)}\;\; x_1+x_2=1 \\ & \text{(ii)}\;\; x_n\in[0,1] \quad n=1,2. \end{array}$$

has the maximum value function $g_2=g_2(c)$ and the maximum point $x^*(c)=(x_1^*(c),x_2^*(c))$ as follows :

$$g_2(c)=\begin{cases} p^2+2pq+q^2 & \text{if } \;c=0 \\ p^2+2pq & \text{if } \;0<c\leq 1/2 \\ p^2+pq & \text{if } \;1/2<c<1 \\ p^2 & \text{if } \;c=1. \end{cases}$$

$$(x_1^*(c),x_2^*(c))=(\lambda\,,1-\lambda) \quad \text{where} \begin{cases} 0\leq\lambda\leq 1 & \text{if } \;c=0 \\ c\leq\lambda\leq 1-c & \text{if } \;0<c\leq 1/2 \\ 0\leq\lambda\leq 1-c\,,\;c\leq\lambda\leq 1 & \text{if } \;1/2<c<1 \\ 0<\lambda<1 & \text{if } \;c=1. \end{cases}$$

Let us now consider the N-variable problem. We use a simple notation

$$Z_N := x_1Y_1 + x_2Y_2 + \cdots + x_NY_N \qquad N \geq 1.$$

First we introduce an additional state parameter $d \in [0,1]$ and define

$$\begin{aligned} f_N(c,d) \;&:=\; \mathrm{Max}[\,P(Z_N \geq c)\,|\,x_1 + \cdots + x_N = d,\; x_n \geq 0 \;\; 1 \leq n \leq N] \\ &\qquad\qquad\qquad\qquad\qquad 0 \leq c, d \leq 1, \;\; N \geq 1. \end{aligned}$$

Then we have the recursive equation

$$\begin{aligned} f_N(c,d) \;&=\; \mathop{\mathrm{Max}}_{0\leq x\leq d}[p \cdot f_{N-1}(c-x, d-x) + q \cdot f_{N-1}(c, d-x)] \\ &\qquad\qquad\qquad\qquad 0 \leq c, d \leq 1, \;\; N \geq 2 \end{aligned} \tag{6}$$

$$f_1(c,d) \;=\; \begin{cases} 1 \\ p \\ 0 \end{cases} \qquad \pi_1^*(c,d) \;=\; \begin{cases} d & \text{if } c = d = 0, \\ d & \text{if } d \geq c > 0, \\ d & \text{if } c > d > 0, \end{cases}$$

Second let us define

$$g_N(c) := \mathrm{Max}[\,P(Z_N \geq c)\,|\,x_1 + \cdots + x_N = 1,\; x_n \geq 0 \;\; 1 \leq n \leq N] \quad N \geq 1.$$

Bellman [1, p.137] derives the recursive equation :

$$\begin{aligned} g_N(c) \;&=\; \mathop{\mathrm{Max}}_{0\leq x\leq 1}\left[p \cdot g_{N-1}\left(\frac{c-x}{1-x}\right) + q \cdot g_{N-1}\left(\frac{c}{1-x}\right)\right] \\ &\qquad\qquad\qquad\qquad 0 \leq c \leq 1, \;\; N \geq 2 \end{aligned} \tag{7}$$

$$g_1(c) \;=\; \begin{cases} 1 \\ p \end{cases} \qquad \pi_1^*(c) \;=\; \begin{cases} 1 & \text{if } c = 0, \\ 1 & \text{if } 0 < c \leq 1. \end{cases}$$

4 Minimum criterion

Let us consider the following three stochastic optimization problems :

$$\begin{array}{lll} & \mathrm{Max} & P(x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_NY_N \geq c) \\ \mathsf{P}(1) & \text{s.t.} & \text{(i)} \;\; x_1 + x_2 + \cdots + x_N = 1 \\ & & \text{(ii)} \;\; x_n \in [0,1] \quad n = 1, 2, \ldots, N, \end{array}$$

$$\mathsf{E}(1) \qquad \mathrm{Max} \quad E[x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_NY_N] \quad \text{s.t.} \quad \text{(i)}, \text{(ii)},$$

$$\mathsf{V}(1) \qquad \min \quad V[x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_NY_N] \quad \text{s.t.} \quad \text{(i)}, \text{(ii)}.$$

4.1 Deterministic problems

In this subsection we reduce the three stochastic problems to equivalent deterministic ones. First of all we describe the probability function of random variable

$$W := W^x := x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_NY_N \quad \text{for } x = (x_1, \ldots, x_N).$$

When x satisfies $x_n = 0$ for some n $(1 \le n \le N)$, $W = 0$ w.p. 1. Otherwise

$$W = \begin{cases} x_1 \wedge x_2 \wedge \cdots \wedge x_N \\ 0 \end{cases} \quad \text{for } (Y_1, Y_2, \ldots, Y_N) = \begin{cases} (1, 1, \ldots, 1) \\ \text{otherwise.} \end{cases}$$

Thus we have for any x satisfying $x_n > 0$ for all n

$$P(W = x_1 \wedge \cdots \wedge x_N) = p^N, \quad P(W = 0) = 1 - p^N. \tag{8}$$

4.1.1 Expected value

First we consider the expectation problem $\mathsf{E}(1)$. The expected value is

$$E[W] = \begin{cases} 0 \\ (x_1 \wedge x_2 \wedge \cdots \wedge x_N)p^N \end{cases} \quad \text{for} \quad \begin{cases} \text{some } x_n = 0 \\ \text{all } x_n > 0. \end{cases}$$

Thus $\mathsf{E}(1)$ is reduced to the deterministic optimization problem :

$$\widetilde{\mathsf{E}}(1) \qquad \begin{array}{ll} \text{Max} & (x_1 \wedge x_2 \wedge \cdots \wedge x_N)p^N \\ \text{s.t.} & \text{(i)} \;\; x_1 + x_2 + \cdots + x_N = 1 \\ & \text{(ii)} \;\; x_n \in [0, 1] \quad n = 1, 2, \ldots, N. \end{array}$$

This has the maximum value $\dfrac{p^N}{N}$ at the equal weight $x^* = \left(\dfrac{1}{N}, \ldots, \dfrac{1}{N}\right)$ [2–4].

4.1.2 Variance

Second we consider the variance problem $\mathsf{V}(1)$. The second-order moment is

$$E[W^2] = \begin{cases} 0 \\ (x_1 \wedge x_2 \wedge \cdots \wedge x_N)^2 p^N \end{cases} \quad \text{for} \quad \begin{cases} \text{some } x_n = 0 \\ \text{all } x_n > 0. \end{cases}$$

Since $V[W] = E[W^2] - E^2[W]$, $\mathsf{V}(1)$ is reduced to :

$$\widetilde{\mathsf{V}}(1) \qquad \begin{array}{ll} \text{Max} & (x_1 \wedge x_2 \wedge \cdots \wedge x_N)^2 p^N (1 - p^N) \\ \text{s.t.} & \text{(i)} \;\; x_1 + x_2 + \cdots + x_N = 1 \\ & \text{(ii)} \;\; x_n \in [0, 1] \quad n = 1, 2, \ldots, N. \end{array}$$

Therefore, the variance problem $\mathsf{V}(1)$ has the maximum value $\dfrac{p^N(1 - p^N)}{N^2}$ at the equal weight $x^* = \left(\dfrac{1}{N}, \ldots, \dfrac{1}{N}\right)$, too [2–4]. On the other hand, the variance problem has the *minimum* value 0 at any feasible weight $\hat{x}$ with $\hat{x}_n = 0$ for some n $(1 \le n \le N)$.

4.1.3 Threshold probability

Third let us consider the threshold probability maximimization problem $\mathsf{P}(1)$. When $c = 0$, any feasible point x attains the maximum value 1.

Hereafter we assume that $0 < c \leq 1$. We note that

$$\begin{aligned} & P(x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_NY_N \geq c) \\ = \; & P(x_1Y_1 \geq c)P(x_2Y_2 \geq c) \cdots P(x_NY_N \geq c) \\ = \; & p(x_1)p(x_2)\cdots p(x_N) \end{aligned}$$

where $p(x_n) := p_c(x_n)$ is defined by

$$p(x_n) \;:=\; P(x_nY_n \geq c) \;=\; \begin{cases} p \\ 0 \end{cases} \text{for} \; \begin{cases} x_n \geq c \\ x_n < c. \end{cases} \tag{9}$$

Therefore, the threshold probability problem $\mathsf{P}(1)$ is :

$$\widetilde{\mathsf{P}}(1) \qquad \begin{array}{ll} \text{Max} & p(x_1)p(x_2)\cdots p(x_N) \\ \text{s.t.} & \text{(i)} \;\; x_1 + x_2 + \cdots + x_N = 1 \\ & \text{(ii)} \;\; x_n \in [0,1] \quad n = 1, 2, \ldots, N. \end{array}$$

Let us consider the following two cases :

1. $0 < c \leq 1/N$:
   Then we can take any feasible x^* satisfying $x_n^* \geq c$ for all n. This implies that $p(x_1^*)\cdots p(x_N^*) = p^N$. If any feasible x satisfies $x_n < c$ for some n, then $p(x_1)\cdots p(x_N) = 0$. Thus x^* attains the maximum value p^N.

2. $1/N < c \leq 1$:
   Then any feasible x satisfies $x_n < c$ for some n. Hence $p(x_1)\cdots p(x_N) = 0$. Therefore, any feasible point yields the maximum value (and minimum value) 0.

4.2 Dynamic programming

In this section we solve the preceding three stochastic optimization problems through dynamic programming approach. This dynamic programming approach is final state model [1, 6–11]. First of all we introduce the sequence of random variables $\{\widetilde{\Lambda}_n\}$ and the sequence of sets $\{\Lambda_n\}$ defined by

$$\widetilde{\Lambda}_n := x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_{n-1}Y_{n-1}$$

and

$$\begin{aligned} \Lambda_n \;:=\; & \{\lambda_n \mid \lambda_n = x_1y_1 \wedge \cdots \wedge x_{n-1}y_{n-1} \quad x_i \in [0,1], \\ & \qquad y_i = 0, 1 \in [0,1] \quad i = 1, \ldots, n-1\} \end{aligned}$$

for $n = 2, 3, \ldots, N+1$, respectively. Then we have

$$\begin{aligned} & x_1Y_1 \wedge x_2Y_2 \wedge \cdots \wedge x_NY_N = \widetilde{\Lambda}_{N+1} \\ & \widetilde{\Lambda}_{n+1} = \widetilde{\Lambda}_n \wedge x_nY_n \qquad 2 \leq n \leq N-1 \\ & \Lambda_n = [0,1] \qquad 2 \leq n \leq N+1. \end{aligned}$$

On the other hand we introduce the sequence of variables $\{d_n\}$ defined by

$$d_n := x_n + x_{n+1} + \cdots + x_N \quad n = 1, \ldots, N, \qquad d_{N+1} := 0.$$

Then we see that the system of simultaneous constraints (i), (ii) is equivalent to the sequential one

$$d_1 = 1, \quad \begin{cases} d_{n+1} = d_n - x_n \\ x_n \in [0,\ d_n] \end{cases} \quad n = 1, 2, \ldots, N, \quad d_{N+1} = 0$$

In particular, we note that $x_n \in [0,\ d_n]$ for $n = N$ becomes $x_N \in \{d_N\}$ or $x_N = d_N$.

4.2.1 Expected value

Thus the expectation problem is transliterated to the dynamic programming problem with terminal function :

$$\begin{array}{llll} & \text{Max} & E[\widetilde{\Lambda}_{N+1}] & \\ \mathsf{DE}(1) & \text{s.t.} & \text{(i)}' \begin{cases} d_2 = d_1 - x_1 \\ \widetilde{\Lambda}_2 = x_1 Y_1 \\ x_1 \in [0,\ d_1] \end{cases}, \begin{cases} d_{n+1} = d_n - x_n \\ \widetilde{\Lambda}_{n+1} = \lambda_n \wedge x_n Y_n \\ x_n \in [0,\ d_n] \end{cases} & n = 2, \ldots, N \\ & & \text{(ii)}'' \ d_{N+1} = 0 & \end{array}$$

where $d_1 = 1$ is the initial state at time $n = 1$. Thus we have an alternating sequence of states and decisions as follows :

$$\begin{array}{l} d_1 = 1 \xrightarrow{x_1} (d_2, \lambda_2) \xrightarrow{x_2} (d_3, \lambda_3) \xrightarrow{x_3} (d_4, \lambda_4) \cdots \\ \xrightarrow{x_{n-1}} (d_n, \lambda_n) \xrightarrow{x_n} (d_{n+1}, \lambda_{n+1}) \xrightarrow{x_{n+1}} \cdots \\ \xrightarrow{x_{N-1}} (d_N, \lambda_N) \xrightarrow{x_N} (d_{N+1}, \lambda_{N+1}) \ \text{where} \ d_{N+1} = 0. \end{array}$$

We note that the first state is $d_1 = 1$ and the n-th decision is x_n. Both are one-variable. All the remaining states $\{(d_n, \lambda_n)\}$ are two-variable. The terminal condition $d_{N+1} = 0$ requires that only the final decision x_N has no choice at (d_N, λ_N) : it must be the first component d_N. Any other decision has a *continuous* choice : $x_n \in [0, d_n]$ $1 \le n \le N-1$.

First let $u_1(d_1)$ be the maximum value of DE(1). Second let $u_n(d_n, \lambda_n)$ be the maximum value of

$$\begin{array}{llll} & \text{Max} & E[\widetilde{\Lambda}_{N+1}] & \\ \mathsf{DE}_n(d_n, \lambda_n) & \text{s.t.} & \text{(i)}' \begin{cases} d_{m+1} = d_m - x_m \\ \widetilde{\Lambda}_{m+1} = \lambda_m \wedge x_m Y_m \\ x_m \in [0,\ d_m] \end{cases} & m = n, \ldots, N \\ & & \text{(ii)}'' \ d_{N+1} = 0 & \end{array}$$

for $(d_n, \lambda_n) \in [0, 1]^2$, $n = 2, \ldots, N$. Finally let $u_{N+1}(d_{N+1}, \lambda_{N+1})$ be as follows :

$$u_{N+1}(d_{N+1}, \lambda_{N+1}) = \lambda_{N+1} \qquad d_{N+1} \in \{0\}, \quad \lambda_{N+1} \in [0, 1].$$

Then we have the recursive formula

$$\begin{cases} u_{N+1}(0,\lambda) = \lambda & 0 \le \lambda \le 1 \\ u_N(d,\lambda) = p\cdot(\lambda \wedge d) & 0 \le d,\lambda \le 1 \\ u_n(d,\lambda) = \underset{0\le x\le d}{\mathrm{Max}}[\,p\cdot u_{n+1}(d-x,\lambda\wedge x) + q\cdot u_{n+1}(d-x,0)\,] & \\ \qquad\qquad\qquad 0 \le d,\lambda \le 1,\ \ 2\le n\le N-1 & \\ u_1(1) = \underset{0\le x\le 1}{\mathrm{Max}}[\,p\cdot u_2(1-x,\lambda\wedge x) + q\cdot u_2(1-x,0)\,] & \end{cases} \tag{10}$$

Let $\pi_n^*(d,\lambda)$ be the maximizer in (10). Then the sequence $\pi^* = \{\pi_n^*\}$ is an optimal policy. In fact, solving (10), we have the sequence of maximum value functions $\{u_n^*\}$ and an optimal policy π^*, where

$$u_{N+1}(0,\lambda) = \lambda, \quad u_n(d,\lambda) = p^{N-n+1}\left(\lambda \wedge \frac{d}{N-n+1}\right) \quad 2\le n\le N, \quad u_1(1) = \frac{p^N}{N}. \tag{11}$$

$$\pi_n^*(d,\lambda) = \frac{d}{N-n+1} \quad 2\le n\le N, \qquad \pi_1^*(1) = \frac{1}{N}. \tag{12}$$

Thus we see that the pair of sequence of maximum value functions and an optimal policy yields the optimal solution of the expectation problem $\mathsf{E}(1)$:

$$u_1(1) = \frac{p^N}{N}, \qquad x^* = \left(\frac{1}{N},\frac{1}{N},\cdots,\frac{1}{N}\right). \tag{13}$$

4.2.2 Threshold probability

Second, let us now consider the threshold probability maximization problem $\mathsf{P}(1)$:

$$\mathsf{P}(1) \qquad \begin{array}{ll} \mathrm{Max} & P(\,x_1Y_1\wedge\cdots\wedge x_NY_N \ge c\,) \\ \mathrm{s.t.} & \text{(i)}\ \ x_1+x_2+\cdots+x_N = 1 \\ & \text{(ii)}\ \ x_n\in[0,1] \quad n=1,2,\ldots,N. \end{array}$$

It is well known that any threshold probability is expressed as an expected value through characteristic function :

$$P(\,x_1Y_1\wedge\cdots\wedge x_NY_N \ge c\,) = E[\chi(x_1Y_1\wedge\cdots\wedge x_NY_N)] \tag{14}$$

where $\chi(\cdot)$ is the characteristic function of interval $[c,\infty)$

$$\chi(w) = \begin{cases} 1 & c\le w<\infty \\ 0 & w<c. \end{cases}$$

Thus the threshold probability problem $\mathsf{P}(1)$ becomes the expected value problem :

$$\mathsf{EP}(1) \qquad \mathrm{Max}\ \ E[\,\chi(x_1Y_1\wedge\cdots\wedge x_NY_N)\,] \ \ \text{s.t.}\ \ \text{(i), (ii)}$$

The preceding analysis which generates an equivalent dynamic programming problem DE(1) works well for the expectation problem EP(1). This problem is formulated into the dynamic programming problem with terminal function :

$$\text{DP}(1) \qquad \begin{array}{ll} \text{Max} & E[\,\chi(\widetilde{\Lambda}_{N+1})\,] \\ \text{s.t.} & \text{(i)}' \begin{cases} d_2 = d_1 - x_1 \\ \widetilde{\Lambda}_2 = x_1 Y_1 \\ x_1 \in [0,\, d_1] \end{cases} , \begin{cases} d_{n+1} = d_n - x_n \\ \widetilde{\Lambda}_{n+1} = \lambda_n \wedge x_n Y_n \\ x_n \in [0,\, d_n] \end{cases} \qquad n = 2, \ldots, N \\ & \text{(ii)}'' \;\; d_{N+1} = 0 \end{array}$$

where

$$d_1 = 1.$$

Then we have the corresponding recursive equation as follows :

$$\begin{cases} v_{N+1}(0,\lambda) = \chi(\lambda) & 0 \le \lambda \le 1 \\ v_N(d,\lambda) = p \cdot \chi(\lambda \wedge d) & 0 \le d, \lambda \le 1 \\ v_n(d,\lambda) = \underset{0 \le x \le d}{\text{Max}} [\,p \cdot v_{n+1}(d-x, \lambda \wedge x) + q \cdot v_{n+1}(d-x, 0)\,] & \\ \qquad\qquad\qquad\qquad\qquad\qquad 0 \le d, \lambda \le 1, \;\; 2 \le n \le N-1 & \\ v_1(1) = \underset{0 \le x \le 1}{\text{Max}} [\,p \cdot v_2(1-x, \lambda \wedge x) + q \cdot v_2(1-x, 0)\,] & \end{cases} \tag{15}$$

Thus we have obtained the sequence of maximum value functions $\{v_n^*\}$ and an optimal policy σ^*. The pair yields the optimal solution of thresold probability problem P(1) :

$$v_1(1) = \begin{cases} p^N \\ 0 \end{cases} \qquad x^* = \begin{cases} (x_1^*, \ldots, x_N^*) & 0 < c \le \frac{1}{N} \\ \text{any feasible point} & \frac{1}{N} < c \le 1 \end{cases} \tag{16}$$

where $x^* = (x_1^*, \ldots, x_N^*)$ is any feasible point satisfying $x_n^* \ge c$ for all n.

5 Markov chain

Now, as a summary, we consider a general problem. We have assumed that the finite sequence $\{Y_1, Y_2, \ldots, Y_N\}$ is independent. We remove the independence. Insteads, we take a Markov chain $\{Y_n\}_1^{N+1}$ with transition probability law $p = \{p(\cdot|\cdot)\}$ on finite state space Y:

$$p(z|y) \ge 0 \quad y, z \in Y, \qquad \sum_{z \in Y} p(z|y) = 1 \quad y \in Y.$$

Further we assume that a *reward function* $r : Y \to R^1$, an associative *aggregator* $\circ : R^1 \times R^1 \to R^1$:

$$(r \circ s) \circ t = r \circ (s \circ t)$$

and a *utility function* $\psi : R^1 \to R^1$ are given [5].

We consider an optimal weighting problem for Markov chain as follows :

$$\mathsf{G}_N(y_1,1) \qquad \begin{array}{ll} \text{Max} & E[\,\psi(x_1r(Y_1)\circ x_2r(Y_2)\circ\cdots\circ x_Nr(Y_N))\,] \\ \text{s.t.} & \text{(i)}\;\; x_1+x_2+\cdots+x_N=1 \\ & \text{(ii)}\;\; x_n\in[0,1] \qquad\qquad\qquad n=1,2,\ldots,N. \\ & \text{(iii)}\;\; Y_{n+1}\sim p(\cdot|y_n) \end{array}$$

The preceding dynamic programming method transforms $\mathsf{G}_N(y_1,1)$ to the equivalent sequential optimization problem :

$$\mathsf{DG}_N(y_1,d_1) \qquad \begin{array}{ll} \text{Max} & E[\,\psi(\widetilde{\Lambda}_{N+1})\,] \\ \text{s.t.} & \text{(i)}'\;\; \begin{cases} d_2=d_1-x_1 \\ \lambda_2=x_1r(y_1) \\ x_1\in[0,\,d_1] \\ Y_2\sim p(\cdot|y_1) \end{cases},\;\; \begin{cases} d_{n+1}=d_n-x_n \\ \lambda_{n+1}=\lambda_n\circ x_nr(y_n) \\ x_n\in[0,\,d_n] \\ Y_{n+1}\sim p(\cdot|y_n) \end{cases} \quad n=2,\ldots,N \\ & \text{(ii)}''\;\; d_{N+1}=0 \end{array}$$

where

$$d_1=1.$$

Here we take

$$\begin{array}{lll} \widetilde{\Lambda}_n & := & x_1r(Y_1)\circ x_2r(Y_2)\circ\cdots\circ x_{n-1}r(Y_{n-1}) \\ \Lambda_n & := & \{\lambda_n|\lambda_n=x_1r(y_1)\circ x_2r(y_2)\circ\cdots\circ x_{n-1}r(y_{n-1}) \\ & & \quad 0\le x_m\le 1,\; y_m\in Y \qquad 1\le m\le n-1\} \end{array} \qquad 2\le n\le N+1$$

Thus we have the corresponding recursive equation :

$$\begin{cases} w_{N+1}(y,0;\lambda)=\psi(\lambda) & \lambda\in\Lambda_{N+1} \\ w_N(y,d;\lambda)=\psi(\lambda\circ dr(y)) & y\in Y,\;\; 0\le d\le 1,\;\; \lambda\in\Lambda_N \\ w_n(y,d;\lambda)=\underset{0\le x\le d}{\text{Max}}\sum_{z\in Y}w_{n+1}(z,d-x;\lambda\circ xr(y))p(z|y) & \\ \qquad\qquad y\in Y,\;\; 0\le d\le 1,\;\; \lambda\in\Lambda_2,\;\; 2\le n\le N-1 & \\ w_1(y,1)=\underset{0\le x\le 1}{\text{Max}}\sum_{z\in Y}w_2(z,1-x;xr(y))p(z|y) & y\in Y. \end{cases} \tag{17}$$

References

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

[2] S. Iwamoto, Inverse theorem in dynamic programming I, II, III, J. Math. Anal. Appl. **58**(1977), 113–134, 247–279, 439–448.

[3] S. Iwamoto, Dynamic programming approach to inequalities, J. Math. Anal. Appl. **58**(1977), 687–704.

[4] S. Iwamoto, *Theory of Dynamic Program: Japanese,* Kyushu Univ. Press, Fukuoka, 1987.

[5] S. Iwamoto, Associative dynamic programs, J. Math. Anal. Appl., **201**(1996), 195-211.

[6] S. Iwamoto, Maximizing threshold probability through invariant imbedding, Ed. H.F. Wang and U.P. Wen, Proceedings of The Eighth BELLMAN CONTINUUM, National Tsing Hua University, Hsinchu, ROC, Dec., 2000, 17-22.

[7] S. Iwamoto, Fuzzy decision-making through three dynamic programming approaches, d. H.F. Wang and U.P. Wen, Proceedings of The Eighth BELLMAN CONTINUUM, National Tsing Hua University, Hsinchu, ROC, Dec., 2000, 23-27.

[8] S. Iwamoto, Recursive method in stochastic optimization under compound criteria, Advances in Mathematical Economics **3**(2001), 63-82.

[9] S. Iwamoto, K. Tsurusaki and T. Fujita, On Markov policies for minimax decision processes, J. Math. Anal. Appl. **253**(2001), 58-78.

[10] S. Iwamoto and T. Fujita, Stochastic decision-making in a fuzzy environment, J. Operations Res. Soc. Japan **38**(1995), 467-482.

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