Some Studies on Multistage Decision Making Under Fuzzy Dynamic Programming

Prashanta Kumar Parida prashantaparida_77@rediffmail.com
Department of mathematics,
Eastern Academy of Science and Technology
Bhubaneswar, Orissa, India.

S.K.Sahoo sahoosk1@rediffmail.com
Institute of Mathematics & Applications,
Bhubaneswar-751003, Orissa, India.

K.C.Sahoo kishorechandra_sahoo@rediffmail.com
Department of mathematics,
Eastern Academy of Science and Technology
Bhubaneswar, Orissa, India.

Abstract

Studies has been made in this paper, on multistage decision problem, fuzzy dynamic programming (DP). Fuzzy dynamic programming is a promising tool for dealing with multistage decision making and optimization problems under fuzziness. The cases of deterministic, stochastic, and fuzzy state transitions and of the fixed and specified, implicity given, fuzzy and infinite times, termination times are analyzed.

Keywords: Multistage decision process, multistage decision making under fuzziness, multistage optimization problem under fuzziness, fuzzy dynamic programming.

1. INTRODUCTION

Multistage decision problems usually arise when decisions are made in the sequential manner over time where earlier decisions may affect the feasibility and performance of later decisions. Dynamic programming approach is used in such multistage decision problems which are based on Bellman's Principle of Optimality.

The common elements of dynamic programming models include decision stages, a set of possible states in each stage, transitions from states in one stage to states in the next, value functions that measure the best possible objective values that can be achieved starting from each state, and finally the recursive relationships between value functions and different states.

A decision made at a given stage, and at a given state, induces a change in the output of the process. The performance of the process is measured over some planning horizon, and is expressed by an aggregate of partial scores that express the performance of the particular stage decisions. An optimal sequence of decisions or controls at the consecutive stages over the planning horizon is then sought.

Dynamic programming is a powerful formal tool for dealing with a large spectrum of multistage decision making and control problems. In mid 1950's (cf. Bellman, 1957), dynamic programming has become a standard tool in many fields including operation research, control theory and engineering, engineering, computer science, etc.

Dynamic programming has been one of the earliest general techniques to which fuzzy sets theory has been applied (cf. Chang, 1969; Bellman and Zadeh 1970, Esogbue and Ramesh, 1970).

2. MULTISTAGE DECISION MAKING PROCESS

The multistage decision making process can be separated into a number of sequential steps, or stages, which is completed in one or more ways. The options for completing stages are known as decisions. The condition of the process at a given stage is known as state at that stage; each decision effects a transition from the current state to a state associated with the next stage.

The multistage decision making process is finite if there are only a finite number of stages in the process and a finite number of states associated with each stage. Many multistage decision processes have returns (cost or benefits) associated with each decision, and these returns may very with both the stage and state of the process. The multistage decision process is deterministic if the outcome of each decision is known exactly.

Mathematical Program

The mathematical program

$$\text{optimize} \quad Z = \sum_{i=1}^{n} f_i(x_i)$$

$$\text{subject to} \quad \sum_{i=1}^{n} x_i \le b \qquad (1)$$

$$x_i \ge 0, \quad i = 1,2,......,n$$

in which $f_i(x_i)$, $i = 1,2,......,n$ are known (non-linear) functions of a single variable and b is a non-negative integer.

2.1. PRINCIPLE OF OPTIMALITY IN DYNAMIC PROGRAMMING

Dynamic programming is an approach for optimizing multistage decision making processes. It is based on Bellman's principle of optimality. An optimal policy has the property that, "regardless of the decisions taken to enter a particular state in a particular stage, the remaining decisions must constitute an optimal policy for leaving that state". This principle, begin with the last stage of an n-stage process and determine for each state the best policy for leaving that state and completing the process, assuming that all preceding stages have been completed.

Let $u \equiv$ the state variable, whose values specify the states

$m_j(u) \equiv$ optimum return from completing the process beginning at stage j in state u

$d_j(u) \equiv$ decision taken at stage j that achieves $m_j(u)$

The entries corresponding to the last stage of the process, $m_n(u)$ and $d_n(u)$ are generally straightforward to compute. The remaining entries are obtained recursively; i.e. the entries for j th stage $(j = 1,2,......, n-1)$ are determined as functions of the entries for the $(j+1)$th stage.

The dynamic programming approach is well suited processes modeled by system Eq.(1) in which each decision pays off separately, independent of other decisions. For system Eq.(1), the value of $m_n(u)$ for $u = 0,1,2,......,b$ are given by

$$m_n(u) = \underset{0 \le x \le u}{optimum} \{f_n(x)\} \qquad (2)$$

The recursion formula is

$$m_j(u) = \underset{0 \le x \le u}{optimum}\{f_j(x) + m_{j+1}(u-x)\} \tag{3}$$

for $j = n-1, n-2, \ldots\ldots, 1$. In Eq.(2), the decision variable x (which is denoted x_n in Eq.(1)) runs through integral values, as x $(\equiv x_j)$ in Eq.(3). That value of x which yields the optimum in Eq.(2) is taken as $d_n(u)$ and that value of x which yields the optimum in Eq.(3) is taken as $d_j(u)$. If more than one values of x yields either optimum, arbitrarily choose one as the optimal decision. The optimal solution to program Eq.(1) is $Z^* = m_1(b)$ the optimal return from completing the process beginning at state1 with b units available for allocation. With Z^* determined, the optimal decisions $x_1^*, x_2^*, \ldots\ldots, x_n^*$ are found sequentially from

$$\left.\begin{aligned} x_1^* &= d_1(b) \\ x_2^* &= d_2(b - x_1^*) \\ x_3^* &= d_3(b - x_1^* - x_2^*) \\ &\ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \\ x_n^* &= d_n(b - x_1^* - x_2^* - \ldots\ldots - x_{n-1}^*) \end{aligned}\right\}$$

(4) **2.2. DYNAMIC PROGRAMMING WITH DISCOUNTING**

If money earns interest at the rate of i per period, an amount $p(n)$ due n periods in the future has the present (or discounted) value

$$p(0) = \alpha^n p(n), \quad \text{where} \quad \alpha \equiv \frac{1}{1+i} \tag{5}$$

The multistage decision making processes, the stages represent time periods and the objective is to optimize a monetary quantity. The solution by dynamic programming, the recurrence formula for $m_j(u)$, the best return beginning in stage j and state u, involves terms of the form $m_{j+c}(y)$, the best return beginning in the stage $j+c$ (c time periods after stage j) and state y. If $m_{j+c}(y)$ is multiplied by α^c, where α is the above-defined discount factor, then $m_{j+c}(y)$ is discounted to its present value at the beginning of stage j. So $m_1(u)$ will be discounted to the beginning of stage 1, which is the start of the process.

2.3. STOCHASTIC MULTISTAGE DECISION PROCESSES

The multistage decision making process is stochastic if the return associated with at least one decision in the process is random. This randomness generally two ways: either the states are uniquely determined by the decisions but the returns associated with one or more states are uncertain or the returns are uniquely determined by the states arising from one or more decisions are uncertain.

If the probability distributions governing the random events are known and if the number of stages and the number of states are finite, then the dynamic programming approach is useful for optimizing a stochastic multistage decision making process. The general procedure is to optimize the expected value of the return. In those cases, where the randomness occurs exclusively in the returns associated with the states and not in the states arising from the decisions, this procedure has the effect of transforming a stochastic process into a deterministic one.

3. MULTISTAGE DECISION MAKING UNDER FUZZINESS

3.1. DECISION MAKING UNDER FUZZINESS

The basic elements of [19] general approach are: a fuzzy goal G in X , a fuzzy constraint C in X , and a fuzzy decision D in X ; X is a (non-fuzzy) space of options. The fuzzy relation

$$\mu_D(x) = \mu_C(x) \wedge \mu_G(x), \quad \forall \ x \in X \tag{6}$$

which expresses the "goodness" of an $x \in X$ as a solution to the decision making problem. Considered, 1 for definitely good (perfect) to 0 for definitely bad (unacceptable), through all intermediate values. The "$a \wedge b = \min(a,b)$" operation is commonly used and is assumed throughout this paper though many other operations are also employed ([11] and [12]).

For an optimal (non-fuzzy) solution, an $x \in X$ such that

$$\mu_D(x^*) = \sup_{x \in X} \mu_D(x) = \sup_{x \in X} (\mu_C(x) \wedge \mu_G(x)) \tag{7}$$

is a natural (but not the only possible [11]) choice.

The multiple fuzzy constraints and fuzzy goals is defined in different spaces. Suppose that the fuzzy constraint C is defined as a fuzzy set in $X = \{x\}$ the fuzzy goal G is defined as a fuzzy set in $Y = \{y\}$ and a function $f : X \to Y$, $y = f(x)$ is known. X and Y may be sets of decisions and their outcomes, respectively.

Now, the induced fuzzy goal, both G' and C are defined as

$$\mu_{G'}(x) = \mu_G[f(x)] \text{ for each } x \in X \tag{8}$$

and the (min-type) fuzzy decision is

$$\mu_D(x) = \mu_{G'}(x) \wedge \mu_C(x) = \mu_G[f(x)] \wedge \mu_C(x), \text{ for each } x \in X \tag{9}$$

The n fuzzy constraints defined in X , $C_1, C_2, ..., C_n$, m fuzzy goals defined in Y , $G_1, G_2, ..., G_m$, and a function $y = f(x)$, then the (min-type) fuzzy decision is

$$\mu_D(x) = \mu_{G_1}[f(x)] \wedge .... \wedge \mu_{G_m}[f(x)] \wedge \mu_{C_1}(x) \wedge .... \wedge \mu_{C_n}(x) \quad \text{for each } x \in X \tag{10}$$

Now we proceed the multistage decision making case within the above general framework.

3.2. MULTISTAGE DECISION MAKING (CONTROL) UNDER FUZZINESS

By [11], [12] and [19], we assume, the dynamics of the deterministic dynamic system under control is described by a state transition equation

$$x_{t+1} = f(x_t, u_t), \qquad t = 0,1,.......$$

(11)

where $x_t, x_{t+1} \in X = \{x\} = \{s_1, s_2, ..., s_n\}$ are the states at time (stage) t and $t+1$, respectively and $u \in U = \{u\} = \{c_1, c_2, ..., c_m\}$ is the decision (control or input) at t; X and U, are assumed finite.

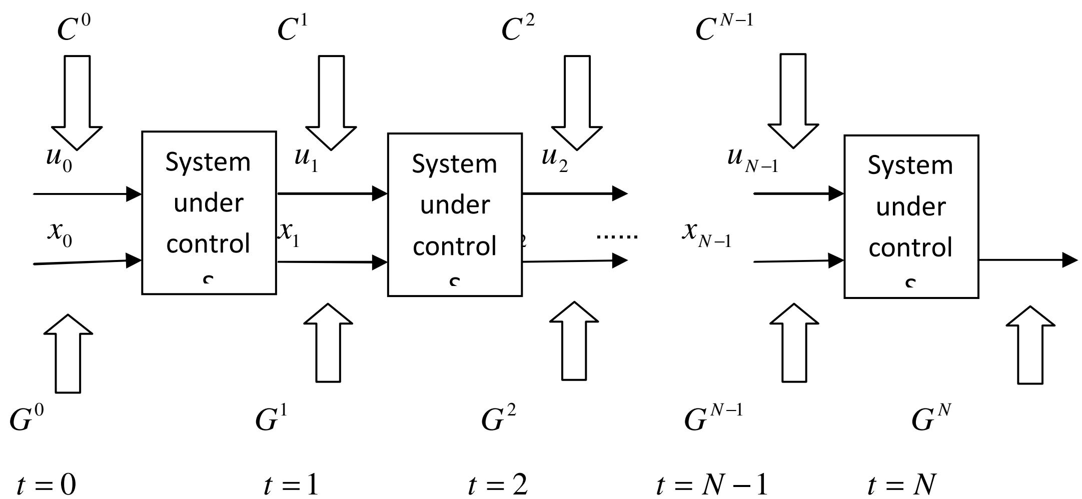

FIGURE 1: A General Framework For Multistage Decision Making Under Fuzziness

The multistage decision making (control) under fuzziness may be depicted in Fig.1. We start from an initial state at stage (time) $t=0$, x_0 make a decision u_0 attain a state, at time $t=1$, x_1 make a decision u_1,...... Finally, being at $t=N-1$ in x_{N-1}, we apply u_{N-1} and attain the final state x_N.

The state transitions are given Eq.(11), the consecutive decision u_t are subjected to fuzzy constraints C^t and on the states x_{t+1} fuzzy goals G^{t+1} are imposed, $t=0,1,.......,N-1.$ The multistage decision making process is evaluated by the fuzzy decision which is assumed a decomposable fuzzy set in $U\times X\times.......\times U\times X$, is

$$\mu_D(u_0,u_1,.....,u_{N-1}|x_0)=\mu_{C^0}(u_0)\wedge\mu_{G^1}(x_1)\wedge..........\wedge\mu_{C^{N-1}}(u_{N-1})\wedge\mu_{G^N}(x_N)$$
$$=\overset{N-1}{\underset{t=0}{\wedge}}\left[\mu_{C^t}(u_t)\wedge\mu_{G^{t+1}}(x_{t+1})\right]$$

(12)

where N is some termination time is fixed. The basic case, the optimal sequence of decisions $u_0^*,u_1^*,.....,u_{N-1}^*$, such that

$$\mu_D(u_0^*,u_1^*,.....,u_{N-1}^*|x_0)=\max_{u_0,u_1,...,u_{N-1}}\left[\mu_{C^0}(u_0)\wedge\mu_{G^1}(x_1)\wedge..........\wedge\mu_{C^{N-1}}(u_{N-1})\wedge\mu_{G^N}(x_N)\right]$$

$$=\max_{u_0,u_1,...,u_{N-1}}\overset{N-1}{\underset{t=0}{\wedge}}\left[\mu_{C^t}(u_t)\wedge\mu_{G^{t+1}}(x_{t+1})\right]$$

(13)

For simplicity, a fuzzy goal, $\mu_{G^N}(x_N)$ is only imposed on the final state x_N. Then the fuzzy decision is

$$\mu_D(u_0,u_1,.....,u_{N-1}|x_0)=\mu_{C^0}(u_0)\wedge\mu_{G^1}(x_1)\wedge..........\wedge\mu_{C^{N-1}}(u_{N-1})\wedge\mu_{G^N}(x_N)$$

(14)

and the problem is to find $u_0^*,u_1^*,.....,u_{N-1}^*$, such that

$$\mu_D\left(u_0^*, u_1^*, \ldots\ldots, u_{N-1}^* | x_0\right) = \max_{u_0, u_1, \ldots, u_{N-1}} \left[\mu_{C^0}(u_0) \wedge \mu_{G^1}(x_1) \wedge \ldots\ldots\ldots\wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(x_N)\right]$$
(15)

This general problem formulation may be extended, mainly with respect to ([11] and [12]):

- The type of termination time: fixed and specified in advance, fuzzy implicitly given infinite.
- The type of the dynamic system: deterministic, stochastic, fuzzy and fuzzy stochastic.
- The type of objective function: cost minimization, profit/benefit maximization and fuzzy criterion set based satisfactory degree maximization, and virtually all cases a dynamic-programming-type algorithm can be devised.

4. FUZZY DYNAMIC PROGRAMMING FOR THE CASE OF A FIXED AND SPECIFIED TERMINATION TIME

The case of a fixed and specified (in advance) termination time is basic, and provides a suitable point of departure for extensions. Here we discussed a deterministic, stochastic, fuzzy dynamic system, and fuzzy criterion set dynamic program.

4.1. THE CASE OF A DETERMINISTIC DYNAMIC SYSTEM

A deterministic system is described by its state transition Eq.(11), i.e. $x_{t+1} = f(x_t, u_t)$; $x_t, x_{t+1} \in X = \{s_1, s_2, \ldots\ldots, s_n\}$; $u_t \in U = \{c_1, c_2, \ldots\ldots\ldots, c_m\}$; $t = 0,1,\ldots\ldots, N-1$, $x_0 \in X$ is the initial state, and $N < \infty$ is a fixed and specified termination time. The fuzzy constraints are $\mu_{C^0}(u_0), \ldots\ldots\ldots, \mu_{C^{N-1}}(u_{N-1})$ and the fuzzy goal is $\mu_{G^N}(x_N)$. The fuzzy decision is

$$\mu_D\left(u_0, u_1, \ldots\ldots, u_{N-1} | x_0\right) = \mu_{C^0}(u_0) \wedge \ldots\ldots\ldots\wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(x_N) \tag{16}$$

and to find $u_0^*, u_1^*, \ldots\ldots, u_{N-1}^*$, such that

$$\mu_D\left(u_0^*, u_1^*, \ldots\ldots, u_{N-1}^* | x_0\right) = \max_{u_0, u_1, \ldots, u_{N-1}} \left[\mu_{C^0}(u_0) \wedge \ldots\ldots\ldots\wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(x_N)\right] \tag{17}$$

Clearly, the last two terms are $\mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(f(x_{N-1}, u_{N-1}))$ depend on u_{N-1}, hence Eq.(17) can be rewritten as

$$\mu_D\left(u_0^*, u_1^*, \ldots\ldots, u_{N-1}^* | x_0\right) = \max_{u_0, u_1, \ldots, u_{N-1}} \left[\mu_{C^0}(u_0) \wedge \ldots\ldots\ldots\wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(x_N)\right]$$

$$= \max_{u_0, u_1, \ldots, u_{N-1}} \left[\mu_{C^0}(u_0) \wedge \ldots\ldots \wedge \mu_{C^{N-2}}(u_{N-2}) \wedge \max_{u_{N-1}}\left(\mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(f(x_{N-1}, u_{N-1}))\right)\right] \tag{18}$$

On repeating this backward iteration for $u_{N-1}, u_{N-2}, ...,u_0$, we obtain the set of dynamic programming recurrence equations

$$\begin{cases} \mu_{G^{N-i}}(x_{N-i}) = \max_{u_{N-i}} \left[\mu_{C^{N-i}}(u_{N-i}) \wedge \mu_{G^{N-i+1}}(x_{N-i+1}) \right] \\ x_{N-i+1} = f(x_{N-i}, u_{N-i}), \qquad i = 1,2,........,N \end{cases} \tag{19}$$

where $\mu_{G^{N-i}}(\cdot)$ is a fuzzy goal at $t = N - i$ induced by a fuzzy goal at $t = N - i + 1$.

An optimal sequence of decisions sought, $u_0^*, u_1^*, ..., u_{N-1}^*$ is given by the successive maximizing values of u_{N-i} in Eq.(19). It is convenient to represent the solution, u_t^*, by an optimal policy $a_t^* : X \to U$, such that $u_t^* = a_t^*(x_t)$, $t = 0,1,........,N-1$, i.e., relating an optimal decision to the current state.

4.2. THE CASE OF A STOCHASTIC DYNAMIC SYSTEM

The stochastic system is assumed to be a Markov chain whose temporal evolution is described by a conditional probability $P(x_{t+1} | x_t, u_t)$ such that $x_t, x_{t+1} \in X = \{s_1, s_2, ..., s_n\}$, $u_t \in U = \{c_1, c_2, ..., c_m\}$; $x_0 \in X$ is an initial state, $t = 0,1,......,N-1$ and $N < \infty$ is a fixed and specified termination time. Now the following two problem formulations:

- due to [19]: find an optimal sequence of decisions $u_0^*, u_1^*, ...,u_{N-1}^*$ to maximize the probability of attainment of the fuzzy goal, subject to the fuzzy constraints, i.e.

$$\mu_D(u_0^*, u_1^*, ...,u_{N-1}^* | x_0) = \max_{u_0, u_1, ..., u_{N-1}} \left[\mu_{C^0}(u_0) \wedge .... \wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(x_N) \right] \tag{20}$$

where the fuzzy goal is viewed to be a fuzzy event in X whose (non-fuzzy) probability is [17],

$$E\mu_{G^N}(x_N) = \sum_{x_N \in X} P(x_N | x_{N-1}, u_{N-1}) . \mu_{G^N}(x_N) \tag{21}$$

- due to [14]: find an optimal sequence of decisions $u_0^*, u_1^*, ...,u_{N-1}^*$ to maximize the expectation of the fuzzy decisions membership function, i.e.

$$\mu_D(u_0^*, u_1^*, ...,u_{N-1}^* | x_0) = \max_{u_0, u_1, ..., u_{N-1}} E\left[\mu_{C^0}(u_0) \wedge .... \wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \mu_{G^N}(x_N) \right] \tag{22}$$

These formulations are clearly not equivalent.

Bellman and Zadeh's approach [19]

Since in Eq.(20), $\mu_{C^{N-1}}(u_{N-1}) \wedge E\mu_{G^N}[f(x_{N-1}, u_{N-1})]$ depend only on u_{N-1}, the next two right-most terms depend only on u_{N-2} etc., the structure of Eq.(20) is essentially the same as that of Eq.(18), and the set of fuzzy dynamic programming recurrence equation is

$$\begin{cases} \mu_{G^{N-i}}(x_{N-i}) = \max_{u_{N-i}} \left[\mu_{c^{N-i}}(u_{N-i}) \wedge E\mu_{G^{N-i+1}}(x_{N-i+1}) \right] \\ E\mu_{G^{N-i+1}}(x_{N-i+1}) = \sum_{x_{N-i+1} \in X} P(x_{N-i+1} | x_{N-1}, u_{N-1}) \ \mu_{G^{N-i+1}}(x_{N-i+1}); \ i = 1,2,....,N \end{cases} \tag{23}$$

and we consecutively obtain u^*_{N-i} or optimal policies a^*_{N-i}, such that $u^*_{N-i} = a^*_{N-i}(x_{N-i})$, $i = 1,2,.......,N$.

Kacprzyk and Staniewski's approach [14]

To solve problem Eq.(22), first introduce a sequence of functions $h_i : X \times \underset{j=1}{\overset{i}{X}} U \to [0,1]$ and $g_j : X \times \underset{l=1}{\overset{j+1}{X}} U \to [0,1]$; $i = 0,1.2,......,N$; $j = 1,2,.......,N-1$, such that

$$\begin{cases} h_N(x_N, u_0,........,u_{N-1}) = \mu_{C^0}(u_0) \wedge ..... \wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \mu_D(u_0,.....,u_{N-1} | x_0) \\ .... \\ g_k(x_k, u_0,........,u_k) = \sum_{i=1}^{n} h_{k+1}(s_i, u_0,........,u_k).P(s_i | x_k, u_k) \\ h_k(x_k, u_0,........,u_{k-1}) = \max_{u_k} g_k(x_k, u_0,........,u_k) \\ .... \\ h_0(x_0) = \max_{u_0} g_0(x_0, u_0) \end{cases} \quad (24)$$

The consecutive decisions and states are $u_0, u_1,........,u_j$ and $x_0, x_1,........,x_j$, respectively, then g_j is the expected value of $\mu_D(\cdot | x_0)$ provided that the next decisions are optimal, i.e. $u^*_{j+1}, u^*_{j+2},........,u^*_{N-1}$. It can be shown ([11], [12] and [14]), that there exist functions $\omega_k : X \times \underset{j=1}{\overset{k}{X}} U \to U$, such that

$$h_k(x_k, u_0,........,u_{k-1}) = g_k(x_k, u_0,........,\omega_{k-1}, \omega_k(x_k, u_0,........,u_{k-1})).$$

Then, an optimal policy sought, a_t^*, $t = 0,1,2,.......,N-1$ is given by

$$\begin{cases} a_0^* = \omega_0(x_0) \\ a_1^*(x_0, x_1) = \omega_1(x_1, a_0^*(x_0)) \\ .... \\ a^*_{N-1}(x_0, x_1,......,x_{N-1}) = \omega_{N-1}(x_{N-1}, a^*_{N-2}(x_0,.......,x_{N-2},........,a_0^*(x_0)),......) \end{cases} \quad (25)$$

It is depends not only on the current state but also on the trajectory.

4.3. THE CASE OF A FUZZY DYNAMIC SYSTEM

In this case the system is fuzzy and its dynamics is governed by a state transitions equation

$$X_{t+1} = F(X_t, U_t), \quad t = 0,1,2,......... \quad (26)$$

where X_t, X_{t+1} are fuzzy states at time (stage) t and $t+1$, and U_t is a fuzzy decision at t characterized by their membership functions $\mu_{X_t}(x_t)$, $\mu_{X_{t+1}}(x_{t+1})$ and $\mu_{U_t}(u_t)$, respectively. Eq.(26) is equivalent to a conditioned fuzzy set $\mu_{X_{t+1}}(x_{t+1}|x_t, u_t)$ ([11] and [12]).

Baldwin and Pilsworth [13] proposed a dynamic programming scheme. First, for each $t = 0,1,2,........,N-1$ a fuzzy relation $\mu_{R^t}(u_t, x_{t+1}) = \mu_{C^t}(u_t) \wedge \mu_{G^{t+1}}(x_{t+1})$ is constructed. The degree to which U_t and X_{t+1} satisfy C^t and G^{t+1} is

$$\mu_T(u_t, \mu_{R^t}(u_t, x_{t+1}), x_{t+1}) = \max_{u_t}\left[(\mu_{U_t}(u_t) \wedge \mu_{C^t}(u_t)) \wedge \max_{x_{t+1}}(\mu_{X_{t+1}}(x_{t+1}) \wedge \mu_{G^{t+1}}(x_{t+1}))\right] \quad (27)$$

The fuzzy decision is

$$\mu_D(U_0,........,U_{N-1}|X_0)$$

$$= \max_{u_0}\left[(\mu_{U_0}(u_0) \wedge \mu_{C^0}(u_0)) \wedge ....\wedge \max_{u_{N-1}}(\mu_{U_{N-1}}(u_{N-1}) \wedge \mu_{C^{N-1}}(u_{N-1})) \wedge \max_{x_N}(\mu_{X_N}(x_N) \wedge \mu_{G^N}(x_N))\right] \quad (28)$$

and an optimal sequence of fuzzy decisions $U_0^*,........,U_{N-1}^*$, such that

$$\mu_D(U_0^*,........,U_{N-1}^*|X_0) = \max_{U_0,......,U_{N-1}} \mu_D(U_0,........,U_{N-1}|X_0)$$

$$= \max_{U_0}\max_{u_0}\left[(\mu_{U_0}(u_0) \wedge \mu_{C^0}(u_0)) \wedge ....\wedge \max_{U_{N-1}}\max_{u_{N-1}}\begin{pmatrix}\mu_{U_{N-1}}(u_{N-1}) \wedge \mu_{C^{N-1}}(u_{N-1}) \wedge \\ \max_{X_N}\max_{x_N}(\mu_{X_N}(x_N) \wedge \mu_{G^N}(x_N))\end{pmatrix}\right] \quad (29)$$

Hence the set of dynamic programming recurrence equation is

$$\begin{cases}\mu_{\overline{G}^N}(X_N) = \max_{x_N}[\mu_{X_N}(x_N) \wedge \mu_{G^N}(x_N)] \\ \mu_{\overline{G}^{N-i}}(X_{N-i}) = \max_{U_{N-i}}\left[\max_{u_{N-1}}(\mu_{U_{N-i}}(u_{N-i}) \wedge \mu_{C^{N-i}}(u_{N-i})) \wedge \mu_{\overline{G}^{N-i+1}}(X_{N-i+1})\right] \\ \mu_{X_{N-i+1}}(x_{N-i+1}) = \max_{x_{N-i}}\left[\max_{u_{N-1}}(\mu_{U_{N-i}}(u_{N-i}) \wedge \mu_{X_{N-i+1}}(x_{N-i+1}|x_{N-i}, u_{N-i}))\right] \wedge \mu_{X_{N-i}}(x_{N-i}) \\ \qquad i = 1,2,........,N-1\end{cases} \quad (30)$$

They redefine the problem formulation in terms of the reference fuzzy states and fuzzy decisions to finally make Eq.(30) solvable ([11], [12] and [13]).

5. FUZZY DYNAMIC PROGRAMMING FOR THE CASE OF A FUZZY TERMINATION TIME

In many real-world problems it may be more adequate (sufficient) to assume a fuzzy termination time as more or less 5 years, a couple of days, ten years or so [7].

Let $R = \{0,1,......,k-1,k,k+1,.......,N\}$ be the set of decision making states. At each $t \in R$, a fuzzy constraint $\mu_{C^t}(u_t)$, and a fuzzy goal $\mu_{G^v}(x_v)$, $v \in R$ is imposed on the final state. The fuzzy termination time is given by $\mu_T(v)$, $v \in R$, which is a termination time v.

The fuzzy decision

$$\mu_D(u_0, u_1, ..., u_{v-1} | x_0) = \mu_{C^0}(u_0) \wedge .... \wedge \mu_{C^{v-1}}(u_{v-1}) \wedge \mu_T(v).\mu_{G^v}(x_v) \quad (31)$$

and to find an optimal termination time v^* and an optimal sequence of decisions $u_0^*, u_1^*, ..., u_{v-1}^*$ such that

$$\mu_D(u_0^*, u_1^*, ..., u_{v-1}^* | x_0) = \max_{u_0, u_1, ..., u_{N-1}} \left[\mu_{C^0}(u_0) \wedge .... \wedge \mu_{C^{v-1}}(u_{v-1}) \wedge \mu_T(v).\mu_{G^v}(x_v)\right] \quad (32)$$

5.1. THE CASE OF A DETERMINISTIC DYNAMIC SYSTEM

Eq.(32) was formulated and solved by Kacprzyk ([7] and [8]). Then, Stein [23] presented a computationally more efficient model and solution.

Kacprzyk's approach

In Kacprzyk's ([7] and [9]) formulation the set of possible termination times is $\{v \in R : \mu_T(v) > 0\} = \{k, k+1, ..., N\} \subseteq R$, hence an optimal sequence of decisions is $u_0^*, u_1^*, ..., u_{k-1}^*, u_k^*, ..., u_{v-1}^*$. The part $u_{k-1}^*, u_k^*, ..., u_{v^*-1}^*$ is determined by solving

$$\begin{cases} \mu_{G^{v-i}}(x_{v-i}, v) = \max_{V_{v-i}} \left[\mu_{C^{v-i}}(u_{v-i}) \wedge \mu_{G^{v-i+1}}(x_{v-i+1}, v)\right] \\ x_{v-i+1} = f(x_{v-i}, u_{v-i}) \\ i = 1,2,......, v-i+1; \; v = k, k+1, ..., N-1 \end{cases} \quad (33)$$

where $\mu_{G^v}(x_v, v) = \mu_T(v).\mu_{G^v}(x_v)$. An optimal termination time v^*, then found by the maximizing v in

$$\mu_{G^{k-1}}(x_{k-1}) = \max_v \mu_{G^{k-1}}(x_{k-1}, v) \quad (34)$$

The part $u_0^*, u_1^*, ..., u_{K-2}^*$ is then determined by solving

$$\begin{cases} \mu_{G^{k-i+1}}(x_{k-i+1}) = \max_{u_{k-i-1}} \left[\mu_{C^{k-i-1}}(u_{k-i-1}) \wedge \mu_{G^{k-i}}(x_{k-i})\right] \\ x_{k-i} = f(x_{k-i-1}, u_{k-i-1}); \quad i = 1,2,......., k-1 \end{cases}$$

(35)

Stein's approach

Stein [23] presented a computationally more efficient dynamic programming approach. At $t = N-1$, $i \in \{1,2,......, N-1\}$, and attain

$\mu_{\overline{G}^{N-i}}(x_{N-i}) = \mu_T(N-i).\mu_{G^{N-i}}(x_{N-i})$ or apply u_{N-i} and attain $\mu_{C^{N-i}}(u_{N-i}) \wedge \mu_{G^{N-i+1}}(x_{N-i+1})$.

The set of recurrence equation is therefore

$$\begin{cases}\mu_{G^{N-i}}(x_{N-i}) = \mu_{\overline{G}_{N-i}}(x_{N-i}) \max_{u_{N-i}} \left[\mu_{C^{N-i}}(u_{N-i}) \wedge \mu_{G^{N-i+1}}(x_{N-i+1})\right] \\ x_{N-i+1} = f(x_{N-i}, u_{N-i}); \quad i = 1,2,.......,N\end{cases} \tag{36}$$

and an optimal termination time is such a $t = N - i$ at which terminating decision $u^*_{v^*-1}$, occurs, i.e. when

$$\mu_{G^{N-i}}(x_{N-i}) > \max_{u_{N-i}} \left[\mu_{C^{N-i}}(u_{N-i}) \wedge \mu_{G^{N-i+1}}(x_{N-i+1})\right]$$

(37)

5.2. THE CASE OF A STOCHASTIC DYNAMIC SYSTEM

This stochastic dynamic system was first formulated and solved in Kacprzyk ([8] and [9]) by combining the section 4.2 and 5.1. An optimal termination time v^* and an optimal sequence of decisions $u_0^*, u_1^*, ...,u_{v-1}^*$ such that

$$\mu_D\left(u_0^*, u_1^*, ...,u_{v-1}^* \middle| x_0\right) = \max_{v,u_0,u_1,...,u_{v-1}} \left[\mu_{C^0}(u_0) \wedge .... \wedge \mu_{C^{v-1}}(u_{v-1}) \wedge E\mu_{G^v}(x_v)\right] \tag{38}$$

where $\mu_{G^v}(x_v) = \mu_T(v).\mu_{G^v}(x_v)$; and $\{v : \mu_T(v) > 0\} = \{k, k+1, ..., N\}$. As in section 4.2, we determine v^* and $u_{k-1}^*, u_k^*, ...,u_{v^*-1}^*$ by solving

$$\begin{cases}\mu_{G^{v-i}}(x_{v-i}) = \max_{u_{v-i}} \left[\mu_{C^{v-i}}(u_{v-i}) \wedge E\mu_{G^{v-i+1}}(x_{v-i+1})\right] \\ E\mu_{G^{v-i+1}}(x_{v-i+1}) = \sum_{x_{v-i+1} \in X} P\left(x_{v-i+1} \middle| x_{v-1}, u_{v-1}\right) \times \mu_{G^{v-i+1}}(x_{v-i+1}); \quad i = 1,2,.....,N\end{cases} \tag{39}$$

and v^* is given by the maximizing v in

$$\mu_{G^{k-1}}(x_{k-1}) = \max_v \; \mu_{G^{k-1}}(x_{k-1}, v) \tag{40}$$

The remaining part $u_{k-2}^*, u_{k-3}^*, ...,u_0^*$ is obtained by solving

$$\begin{cases}\mu_{G^{k-1-i}}(x_{k-1-i}) = \max_{u_{k-1-i}} \left[\mu_{C^{k-1-i}}(u_{k-1-i}) \wedge E\mu_{G^{k-i}}(x_{k-i})\right] \\ E\mu_{G^{k-i}}(x_{k-i}) = \sum_{x_{k-i} \in X} P\left(x_{k-i} \middle| x_{k-1-i}, u_{k-1-i}\right) \times \mu_{G^{k-1-i}}(x_{k-1-i}); i = 1,2,.......,k-1\end{cases} \tag{41}$$

In the later Stein's [23], formulation the problem is solved by the following set of recurrence equations

$$\begin{cases}\mu_{G^{N-i}}(x_{N-i}) = \mu_{\overline{G}_{N-i}}(x_{N-i}) \wedge \max_{u_{N-i}} \left[\mu_{C^{N-i}}(u_{N-i}) \wedge E\mu_{G^{N-i+1}}(x_{N-i+1})\right] \\ E\mu_{G^{N-i+1}}(x_{N-i+1}) = \sum_{x_{N-i} \in X} P\left(x_{N-i+1} \middle| x_{N-i}, u_{N-i}\right) \times \mu_{G^{N-i+1}}(f(x_{N-i}, u_{N-i})); \; i = 1,2,.......,N\end{cases} \tag{42}$$

where v^* occurs when

$$\mu_{G^{N-i}}(x_{N-i}) > \max_{u_{N-i}} \left[\mu_{C^{N-i}}(u_{N-i}) \wedge E\mu_{G^{N-i+1}}(x_{N-i+1})\right] \tag{43}$$

5.3. THE CASE OF A FUZZY DYNAMIC SYSTEM

In this case, fix some (finite and relatively small) number of reference fuzzy states (and possibly decisions), and obtain an auxiliary approximate system whose state transitions are of deterministic system type ([12] and [16]). Then, Stein's [23] approach can be employed.

6. MULTISTAGE DECISION MAKING (CONTROL) WITH AN IMPLICIT SPECIFIED TERMINATION TIME

Now the process terminates when the state enters for the first time a termination set of states $W = \{ s_{p+1}, s_{p+2}, ..., s_n \} \subset X$. We determine an optimal sequence of decisions $u_0^*, u_1^*, ..., u_{\overline{N}-1}^*$, such that

$$\mu_D \left(u_0^*, u_1^*, ..., u_{\overline{N}-1}^* | x_0\right) = \max_{u_0, u_1, ..., u_{N-1}} [\mu_C(u_0|x_0) \wedge .... \wedge \mu_C(u_{\overline{N}-1}|x_{\overline{N}-1}) \wedge \mu_{G^{\overline{N}}}(x_{\overline{N}})] \tag{44}$$

where $x_0, x_1, ..., x_{\overline{N}-1} \in X \setminus W$, and $x_{\overline{N}} \in W$

The solution of Eq.(44) may proceed by using:

- Bellman and Zadeh's [19] iterative approach,
- Komolov's et al. [22] graph theoretic approach and
- Kacprzyk's ([5] and [8]) branch and bound approach.

7. MULTISTAGE DECISION MAKING (CONTROL) WITH AN INFINITE TERMINATION TIME

In all the problems considered the solution process required some iteration over consecutive stages. This may be justified the number of stages is not too high, and when the process itself exhibits a sufficient variability over time. In the fuzzy setting, the multistage decision making (control) problem with an infinite termination time was first formulated and solved by Kacprzyk and Staniewski ([15] and [16]).

For the deterministic dynamic system Eq.(11), the fuzzy decision is

$$\mu_D(u_0, u_1, ... | x_0) = \mu_C(u_0|x_0) \wedge \mu_G(x_1) \wedge \mu_C(u_1|x_1) \wedge \mu_G(x_2) \wedge ....$$

$$= \lim_{N \to \infty} \bigwedge_{t=0}^{N} [\mu_C(u_t|x_t) \wedge \mu_G(x_{t+1})] \tag{45}$$

and to find an optimal stationary strategy $a_\infty^* = (a^*, a^*, ...)$, such that

$$\mu_D(a_\infty^* | x_0) = \max_{a_\infty} \mu_D(a_\infty | x_0)$$

$$= \max_{a_\infty} \lim_{N \to \infty} \bigwedge_{t=0}^{N} [\mu_C(a(x_t)|x_t) \wedge \mu_G(x_{t+1})] \tag{46}$$

Eq.(46) may be solved in a finite number of steps by using a policy iteration algorithm whose essence is a step-by-step improvement of stationary policies. A policy iteration type algorithm was also proposed for the stochastic system by [16].

8. COMPUTATIONAL COMPLEXITY

The development of efficient algorithms for processing various aspects of fuzzy dynamic programs is an important research area in dynamic programming. In this section, we will summarize main results of the computational complexity analysis provided by (1997). Esogbue [2] using two algorithms of Kacprzyk [7] and Stein [23] for the fuzzy termination time discussed.
Esogbue showed that the dynamic programming approach presented by Kacprzyk requires $N(N+1)/2$ iterations while the one proposed by Stein [23] requires only N and it is computationally more efficient. Let, the time and space complexity analysis for both.

Here, consisting of n equations and u_{N-1} controls each assuming m values, and time $t = N-1, \ldots\ldots, k$, the total number of operations involved is

$$n(2mt - 2mk - 2m - t + k) + n(N - k + 1) + (k-1).n.(2m-1) \tag{47}$$

Let $m = k = t$, then Eq.(47) is of order $O(2k^3)$.

The basic dynamic programming formulation of the problem proposed by Stein [23] is the same expect for the structure of the recurrence equations that require N only iterations of the optimizing process as opposed to $N(N+1)/2$ required is Kacprzyk's algorithm.

The space and the time complexities are of order $O(n)$ and $O(mn)$, respectively. Essentially, the time complexity is of order $O((2m-1)(N-k))$. This is of order $O(k^2)$. The total storage demand is $n(N-k+1)(N-k+2)/2 + 2n + n(k-1)$. This is order $O(k)$. It was pointed out, the difference between the earliest and the latest possible termination times is an important factor in the computationally burden of this optimization process.

So, the model proposed by Stein [23] is computationally more efficient, taking $O(k)$ memory spaces in $O(k^2)$ operations, as opposed to $O(k^2)$ memory spaces in $O(k^3)$ operations in Kacprzyk's (1977) model. The superiority is exhibited both from the space and time complexity considerations.

The case of stochastic system, it was shown that the computational burden and storage requirements are identical for both algorithms. If in the dynamic programming formulation, the objective is to maximize the probability of attainment of the fuzzy goal G subject to non fuzzy constraints.

9. CONCLUSION

We have presented a brief exposition of main aspects of fuzzy dynamic programming, including main problem classes and major applications in a variety of fields. We have studied dynamic programming problems in fuzzy environments. Some basic problems have been studied and development of fuzzy dynamic programming is given. It is shown that the fuzzy dynamic programming may be a promising tool for dealing with multistage decision making and optimization problems under fuzziness. For other surveys of fuzzy dynamic programming and its applications, we refer the reader to [1] and then the fundamental presentation by [6] and [11].

REFERENCE

[1] A.O. Esogbue and R.E. Bellman. "Fuzzy dynamic programming and its extensions". TIMS/ Studies in the Management Sciences, 20, 147-167, 1984.

[2] A.O. Esogbue and V. Ramesh. "Dynamic Programming and fuzzy allocation processes". Technical Memo No. 202, Dept. of Operation Research Case Western University, Cleveland, OH., (1970).

[3] G. Conrnuejds and R. Tutuncu. “Optimization Methods in Finance”. Cambride University Press, (2007).

[4] H.-J. Zimmermann. “Fuzzy Sets, Decision Makings, and Experts Systems”. Kluwer, Dordrecht (1987).

[5] J. Kacprzyk. “A branch-and-bound algorithm for the multistage control of a non fuzzy system in a fuzzy environment”. Control and Cybernetices, 7, 51-64, 1978b.

[6] J. Kacprzyk. “A generalization of fuzzy multistage decision making and control via linguistic quantifiers”. Int. J. of Control, 38, 1249-1270, 1978b.

[7] J. Kacprzyk. “Control of non-fuzzy system in a fuzzy environment with fuzzy termination time”. Systems Sciences, 3, 325-341, 1977.

[8] J. Kacprzyk. “Control of a stochastic system in a fuzzy enviroment with fuzzy termination time”. Systems Science, 4, 291-300, 1978b.

[9] J. Kacprzyk. “Decision making in a fuzzy environment with fuzzy termination time”. Fuzzy Sets and Systems, 1, 169-179, 1978b.

[10] J. Kacprzyk. “Fuzzy dynamic programming: A new quality through fuzzy sets”. In Da Ruan (Ed). Fuzzy Sets Theory and Advanced Mathematical applications, Kluwer, Dordrecht, 137-154, 1996.

[11] J. Kacprzyk. “Multistage control: A model based approach to control and decision making”. Wiley, Chichester (1997).

[12] J. Kacprzyk. “Multistage decision making under fuzziness”. Verlag TUV, Rheinland, Cologne (1978b).

[13] J.F. Baldwin and B.W. Pilsworth. “Dynamic programming for fuzzy systems with fuzzy environment”. J. Math. Anal. and Apples., 85, 1-23, 1982.

[14] J. Kacprzyk and P. Staniewski. “A new approach to the control of stochastic systems in a fuzzy environment”. Arch. Automat, Telemech, XXV, 443-444, 1980.

[15] J. Kacprzyk and P. Staniewski. “Control of a deterministic system in a fuzzy environment over an infinite planning horizon”. J. of Fuzzy Sets and Systems 10, 291-298, 1983.

[16] J. Kacprzyk and P. Staniewski. “Long-term inventory policy-making through fuzzy decision-making models”. J. of Fuzzy Sets and Systems, 8, 117-132, 1982.

[17] L.A. Zadeh. “Probability measures of fuzzy events”. J. of math. Anal. And Appls., 23, 421-427, 1968.

[18] R.E. Bellman. “Dynamic programming”. Princeton Univ. Press, Princeton, NJ.

[19] R.E. Bellman and L.A. Zadeh. "Decision-making in a fuzzy environment". Management Sci., 17, 141-164, 1970.

[20] R. Brorson and G. Naadimuthu. "Theory and Problems of Operations Research". (2006).

[21] S.S.L. Chang. "Fuzzy dynamic programming and the decision making process". Proc. 3rd Princeton Conf. of Inf. Sci. and Syst. (Princeton, NJ, USA), 200-203, 1969.

[22] S.V. Komolov et al. "On the problem of optimal control of a finite automaton with fuzzy constraints and fuzzy goal (in Russian)". Kybernetika (Kiev), 6, 30-34, 1979.

[23] W.E. Stain. "Optimal stopping in a fuzzy environment". J. of Fuzzy Sets and Systems, 3, 252-259, 1968.

[24] Y. Yoshida. "Dynamical Aspects in Fuzzy Decision Making". pp.1-25, (2001).