Design of Multiobjective Satisfactory Fuzzy Logic Controller using Reinforcement Learning

Dong-Oh Kang and Zeungnam Bien

Dept. of Electrical Engineering, KAIST
373-1, Kusong-dong, Yusong-gu, Taejon 305-701, Korea
Tel: +82-42-869-5419, Fax: +82-42-869-8750
E-mail: dongoh@ctrsys.kaist.ac.kr, zbien@ee.kaist.ac.kr

Abstract: The technique of reinforcement learning algorithm is extended to solve the multiobjective control problem for uncertain dynamic systems. A multiobjective adaptive critic structure is proposed in order to realize a max-min method in the reinforcement learning process. Also, the proposed reinforcement learning technique is applied to a multiobjective satisfactory fuzzy logic controller design in which fuzzy logic subcontrollers are assumed to be derived from human experts. Some simulation results are given in order to show effectiveness of the proposed method.

1. Introduction

In daily life, we often confront with various forms of decision making situations under which more than one goal must be fulfilled. It is considered a challenging issue to investigate decision making problems with multiple objectives for efficient, satisfactory solutions. Similarly, in many practical control problems, for example, a overhead crane, an automatic control system of a train, and a refuse incinerator plant, a number of objectives need to be simultaneously achieved, which may conflict or compete with each other [1,2,3,4,5]. And, this kind of control problem is called a **multiobjective control problem**. For these multiobjective control problems, any control strategy based on a single-objective optimization technique can hardly provide a desired performance. In case of large scale systems and/or ill-defined systems, the control problem with multiple objectives is more difficult to handle due to the uncertainty in the system models. Lately, various intelligent system methods have been employed to deal with multiobjective control problems for ill-defined and/or uncertain plants. The fuzzy controller is a typical example.

Yasunobu proposed a predictive fuzzy controller that uses the rules based on skilled human operators' experience and applied it to an automatic container crane and an automatic train operation system [1,2] K. Kim and J. Kim proposed a design method to assign the certainty factors in a heuristic manner to the obtained rules and apply them to calculate control inputs [6]. Ginsberg sorted out those rules which have smaller number of antecedent conditions based on the traditional AI approach [7]. Pedrycz considered a design scheme to delete less confident rules via measure of inconsistency. He defined index of inconsistency and level of inconsistency, and eliminated the rules which have low level of inconsistency [8]. Yu and Bien proposed a new measure of inconsistency between rules, and proposed a control method based on the definition [9]. Lim and Bien also proposed a rule modification scheme via pre-determined satisfaction degree function [5]. Recently, Yang and Bien proposed a programming approach using a fuzzy predictive model, and applied it to a MAGLEV ATO control problem [4]. To solve the multiobjective optimization problem, they assumed that a fuzzy predictive model of the plant is available and applied the max-min approach. But, because their method is dependent on a model, it is hardly applicable for uncertain systems. Also, it is found that, when the prediction horizon is long, the result may require too much computation for real-time control, and the programming approach often makes it difficult to utilize human heuristics and experience in the scheme.

Recently, it is known that the **reinforcement learning** technique can solve this kind of difficult situation [10,11,12]. Reinforcement learning is very similar to the dynamic programming plus a direct adaptive control technique [11]. It uses expected utility-like information about environment to decide the action, and update the information via interaction with the environment without using any model. In this sense, reinforcement learning can be a potential solution to the control problem for which information about the plant is not complete. Furthermore, different from the programming optimization process, the method can include heuristics and experience of human experts in its scheme by modifying its policy.

2. Multiobjective Reinforcement Learning

In this section, we introduce some modified concepts for the multiobjective reinforcement learning. As in the conventional reinforcement learning scheme, a policy π for the multiobjective optimization problem is defined as a mapping from a set S of states to a set A of possible actions. On the other hand, different from the ordinary reinforcement learning, a reward $\mathbf{r}_{t+1}$ is defined as a mapping from the Cartesian product of a set of states and a set of possible action to the M dimensional real-valued vector space R^M as in (1). A state-value function V(s) of the multiobjective problem is also a vector since the state-value function is usually the discounted sum of the immediate rewards over time;

$$\begin{aligned} &\pi : S \to A,\ a = \pi(s) \\ &\mathbf{r}_{t+1} = \vec{\Re}_{t+1}(s,a) : S \times A \to R^M, \\ &= \left[r_{t+1}^1 \quad r_{t+1}^2 \quad \cdots \quad r_{t+1}^M \right]^T, \end{aligned} \tag{1}$$

$$\mathbf{V}^{\pi}(s) = \sum_{t=0}^{\infty} \gamma^t \mathbf{r}_{t+1} = \left[V_1^{\pi} \quad V_2^{\pi} \quad \cdots \quad V_M^{\pi}\right]^T,$$

where $\mathbf{r}_t$ is a vector reward at time t given that the agent follows the policy π, s is a state, a is a action, M is the number of objectives, and $0 \le \gamma \le 1$ is the discount rate.

For the multiobjective reinforcement learning problem under consideration, we define the concepts of **Pareto optimal vector state-value functions** and **Pareto optimal policies** as follows:

Definition 1. Domination [13,14]: A vector $\mathbf{x} \in R^M$ is said to dominate a vector $\mathbf{y} \in R^M$ if every element of **x** is larger than or equal to the corresponding element of **y**, and there exists at least one element of **x** that is larger than the corresponding element of **y**. Formally, we write as follows:

$$\mathbf{x} >_p \mathbf{y} \Leftrightarrow (\forall i, x_i \ge y_i) \; and \; (\exists i, x_i > y_i), i = 1, \cdots, M, \tag{2}$$

where $\mathbf{x} = [x_1 \; x_2 \; \cdots \; x_M]^T$, $\mathbf{y} = [y_1 \; y_2 \; \cdots \; y_M]^T$.

Definition 2. Pareto optimal vector state-value function: A vector state-value function $\mathbf{V}^p$ is called Pareto optimal if there is no other vector state-value function that dominates $\mathbf{V}^p$ among the possible vector state-value functions. The set Ξ_{pareto} of Pareto optimal vector state-value functions is as follows:

$$\Xi_{pareto} = \{\mathbf{V}^p \in \Xi_{\mathbf{T}} \mid \text{There is no } \mathbf{V}(s) \in \Xi_{\mathbf{T}} \text{ such that } \mathbf{V}(s) >_p \mathbf{V}^p(s), \text{for } \forall s \in S\}, \tag{3}$$

where $\Xi_{\mathbf{T}}$ is the set of all possible state-value functions.

Definition 3. Pareto optimal policy: A policy π^p is called Pareto optimal if and only if the resulting vector state value function is a Pareto optimal state-value function.

Let $\prod_p$ denote Pareto optimal policies, that is;

$$\prod^p = \{\pi^p \in \prod \mid \text{There is no } \mathbf{V}^{\pi}(s) >_p \mathbf{V}^{\pi^p}(s), \text{ for } \forall s \in S, \text{and for } \forall \pi \in \prod\}, \tag{4}$$

where $\mathbf{V}^{\pi}$ is a vector state-value function when a policy π is adopted, and $\prod$ is a set of possible policies. It is remarked that the Pareto optimal set of the vector state-value functions may contain more than one element.

From the perspective of Pareto optimality, we may say that the main issue of the **multiobjective reinforcement learning** is how we get a Pareto optimal policy and a Pareto optimal vector state-value function. In this paper, we adopt the max-min approach for a solution of the multiobjective reinforcement learning. In case of the max-min multiobjective optimization, we get a scalarized state-value function as follows:

$$V^*(s) = \max_{\pi \in \prod} \min_{k=1,\cdots,M} V_k^{\pi}(s), \text{ for each } s, \tag{5}$$

Using the function, we can get a Pareto optimal policy just as in the max-min optimization. To realize the max-min optimization process in reinforcement learning, we propose a new reinforcement learning scheme called **multiobjective adaptive critic**. The original adaptive heuristic critic proposed by Barto uses only one adaptive critic to estimate the state-value function of states [10]. In the multiobjective adaptive critic structure, multiple adaptive critics are used for estimating the state-value functions of the corresponding objectives. The structure of the algorithm for two-critic case is depicted in Fig 1.

For each critic, its temporal difference δ_t^i is calculated and used for its update of the parameters as follows:

$$\begin{aligned} &\delta_t^i = r_t^i + \gamma \tilde{V}_{w_{t-1}^i}^i(s_t) - \tilde{V}_{w_{t-1}^i}^i(s_{t-1}), \\ &w_{t+1}^i \leftarrow w_t^i + \alpha \times \delta_{t+1}^i. \end{aligned} \tag{6}$$

Here, $i = 1, \cdots, M$ is the index for a critic, corresponding to the index for an objective, and M is the number of objectives. s_t is the state after an action is taken, s_{t-1} is the state before an action is taken, γ is the discount rate, r_t^i is a reward corresponding to the objective at time t, and $\tilde{V}_{w_{t-1}^i}^i$ is the estimated state-value function and the output of the ith critic with parameters w_{t-1}^i. w_t^i is the parameters of the ith adaptive critic at time t, and α is a learning rate.

To the associative search element, one temporal difference is selected among the temporal differences δ_t^i of the critics and it is used to update the parameters of the critics and the policy as follows:

$$\begin{aligned} &k = \arg \min_{i=1,\cdots,M} \tilde{V}^i, \\ &\delta_t^p = \delta_t^k = r_t^k + \gamma \tilde{V}_{w_{t-1}^k}^k(s_t) - \tilde{V}_{w_{t-1}^k}^k(s_{t-1}), \\ &w_{t+1}^p \leftarrow w_t^p + \beta \times \delta_{t+1}^p, \end{aligned} \tag{7}$$

where δ_t^p is the temporal difference for the associative search element at time t, w_t^p is the parameters of the policy at time t, and β is a learning rate.

We may call the proposed process as an implicit max-min optimization, because the maximization process is not directly adopted and is performed by the temporal difference learning. To realize an adaptive structure with variable parameters for the adaptive critic and associative search element, we choose an adaptive fuzzy inference system in this paper [12].

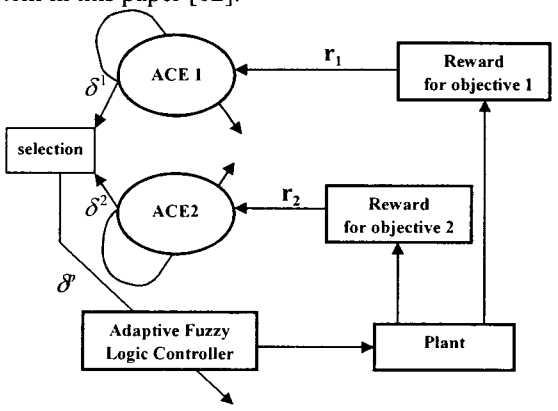

Fig. 1. Structure of multiobjective reinforcement learning (the case of two objectives).

3. Application to Multiobjective Control

Problem

In real control practice, experts may be available about the plant to be controlled. It would be better if we use their expertise in controller design. But, because the controller from the knowledge may not be Pareto optimal, we apply the proposed multiobjective reinforcement learning to the controller. We can also make the reinforcement learning to converge fast using the previously obtained knowledge. The overall procedure to apply the proposed multiobjective reinforcement learning approach to the multiobjective control problem is as follows:

Step 1. Construct a fuzzy controller using knowledge from human experts.

We can utilize the knowledge or heuristics of experts about the plant to control in this step. The obtained knowledge is easily converted into the rule base of fuzzy controller using the fuzzy inference system.

Step 2. Derive the vector state-value function of the current fuzzy controller via temporal difference learning.

In this step, the policy obtained in Step 1 is not modified, while learning of the vector state-value function is performed using the rewards defined for the objective.

Step 3. Determine the satisfaction degrees to compare between the state-value of each objective.

Because the rewards are determined without taking the other objectives into consideration, the state-values of objectives should be normalized using satisfaction degree values from human designer in order to be compared.

Step 4. Derive a Pareto optimal fuzzy controller via the proposed multiobjective reinforcement learning.

The proposed learning method is applied to the adaptive fuzzy controller with the real plant. In the step, the policy, that is, the adaptive fuzzy controller is changed into a Pareto optimal policy for the real plant through the proposed multiobjective reinforcement learning.

4. Design of Satisfactory Multiobjective Fuzzy Logic Controller

It is reported in [5] that the fuzzy controller obtained from human experts is easily optimized for one objective. Therefore, for the multiobjective control problem, it may result in multiple fuzzy subcontrollers each of which is optimized for only one objective as proposed by Lim and Bien [5]. We may consider a supervisory controller, which coordinates the outputs of the subcontrollers. The overall structure of the multiobjective fuzzy logic controller is shown in Fig. 2. In this paper, the weighted sum of the fuzzy logic subcontrollers for the final output is used. The multiobjective adaptive critic architecture is used for the fuzzy logic tuner of the supervisory fuzzy logic controller. And, for the rewards to the controller, the predefined satisfaction degrees of the objectives are used [15]. The max-min multiobjecitve reinforcement learning approach is adopted for the proposed multiobjective adaptive critic structure, and the Pareto optimal policy is found in terms of satisfaction degree. The total controller is described to learn a satisfactory solution.

[Problem Formulation]

Given satisfaction degree functions $P_k(V_k), k=1,\cdots,M$ for the multiple objectives $V_k, k=1,\cdots,M$, determine the weights $w_k, k=1,\cdots,M$ for each output of the fuzzy logic subcontrollers so that the control result is satisfactory.

$$u_{final} = \frac{\sum_{k=1}^{M} w_k u_k}{\sum_{k=1}^{M} w_k}, \qquad u_k = Fuzzy_k(\mathbf{x}), \tag{8}$$

where x is the state variable of the plant and $Fuzzy_k(\mathbf{x})$ is the output of the fuzzy subcontroller.

The rules of the supervisory fuzzy logic controller are in the form of MIMO fuzzy controller as follows:

$$R_\ell: \text{If } x_1 \text{ is } L_{x_1}^{(\ell)} \text{ and } x_2 \text{ is } L_{x_2}^{(\ell)} \text{ and } \ldots \text{ and } x_N \text{ is } L_{x_N}^{(\ell)}, \text{ then } \omega_1 \text{ is } L_{\omega_1}^{(\ell)}, \ldots, \omega_M \text{ is } L_{\omega_M}^{(\ell)}; \tag{9}$$

where N is the number of states and M is the number of objectives

4.1 Simulation

To show the effectiveness of the proposed method, simulation is conducted for an overhead crane control system [5]. For the system, two objectives are defined: positioning and antiswing. These objectives are conflicting with each other and thus some form of compromise is need.

The dynamics of the plant is as follows [5]:

$$\begin{aligned} x(k+1) &= x(k) + T\dot{x}(k), \\ \dot{x}(k+1) &= \dot{x}(k) + T\frac{f}{M}, \\ \theta(k+1) &= \theta(k) + T\dot{\theta}(k), \\ \dot{\theta}(k+1) &= \dot{\theta}(k) + T\frac{-g\sin(\theta(k)) + f\cos(\theta(k))/M}{\ell}, \end{aligned} \tag{10}$$

where f is the input force, x is the trolley position, θ is the angle of the load, the mass M of the trolley is 1 (kg), the gravity constant g is 9.8 (m/sec^2), the length ℓ of the rope is 1 (m), and the sampling time T is 0.01 (sec).

Initial value of the plant is given as x=1.0 (m), θ= 0.7 (rad). The rule base for the weight decision fuzzy inference system is as follows:

$$\text{If } x \text{ is } L_x^{(l)}, \text{ and } \theta \text{ is } L_\theta^{(l)}, \text{ then } \omega_1 \text{ is } L_{\omega_1}^{(l)}, \ \omega_2 \text{ is } L_{\omega 2}^{(l)}; \tag{11}$$

Satisfaction degrees for the objectives are given by

$$P_i(t_{elapsed}) = \begin{cases} 1 & \text{for } t_{elapsed} \le t_{\min} \\ 1 - \dfrac{t_{elapsed} - t_{\min}}{t_{\max} - t_{\min}} & \text{for } t_{\min} \le t_{elapsed} \le t_{\max}, \ i = 1,2, \\ 0 & \text{for } t_{\max} \le t_{elapsed} \end{cases} \tag{12}$$

And its parameters are $t_{\min}=10$ (sec), $t_{\max}=20$ (sec), where $t_{elapsed}$ is the time period to arrive the goal state, and the goal state is $|x| \le 0.05(m)$, and $|\theta| \le 2.9°$ within 20 seconds. The satisfaction degrees are fed back as

rewards after each control action. If the satisfaction degrees of both control objectives exceed the certain desired level, the proposed learning mechanism stops. When learning stops, the supervisory fuzzy logic controller does not vary. But, if one of the satisfaction degrees goes back below the desired level again, the proposed learning mechanism will revive. In this simulation, we set the desired satisfaction level at 0.8.

In the fuzzy subcontroller, 7×7=49 rules are used for each. Fig. 3 and Fig. 4 show that the control results by the proposed method are satisfactory after 4 failures. We find that the learning stops after both of the satisfaction degrees are above the desired level 0.8.

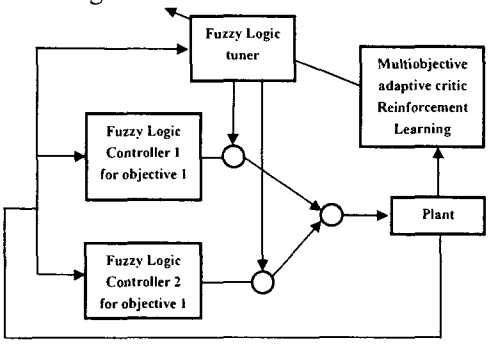

Fig. 2 Overall structure.

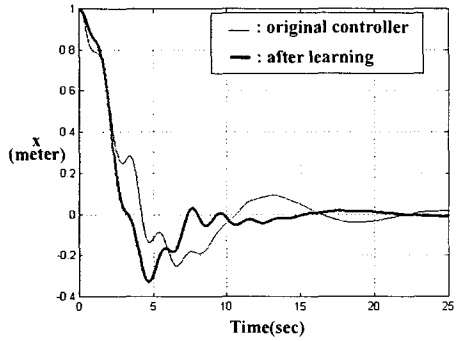

Fig. 3. Control result (position).

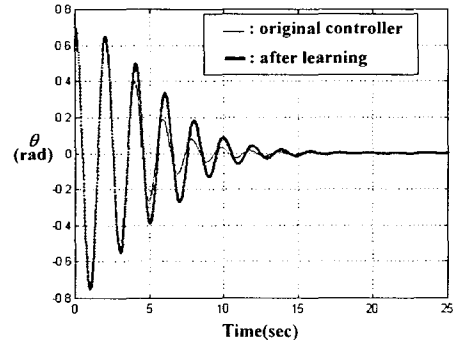

Fig. 4. Control result (angle).

5. Concluding Remarks

It is shown that a modified reinforcement learning method can be effectively used for a multiobjective optimization problem. For this, the Pareto optimal policy for multiobjective reinforcement learning was defined and a multiobjective adaptive critic method was proposed to get the Pareto optimal policy. The proposed method can produce a solution of Pareto optimal type for implicit max-min optimization in case of large scale systems and/or ill-defined systems. Furthermore, the method can be used for obtaining an on-line multiobjective controller because the calculation cost is relatively low. The proposed algorithm was applied to the multiobjective satisfactory fuzzy logic controller. Some simulation results were given to show the effectiveness of the proposed method. For the future research, some theoretical analyses of the convergence property of the proposed method and stability of the multiobjective satisfactory fuzzy logic controller are needed.

References

[1] S. Yasunobu and T. Hasegawa, "Evaluation of an automatic container crane operation system based on predictive fuzzy control," *Control Theory and Advanced Technology*, vol. 2, pp. 419-432, 1986.

[2] S. Yasunobu and S. Miyamoto, "Automatic train operation system by predictive fuzzy control," in *Industrial Application of fuzzy control* (M.Sugeno eds.), pp.1-18, North-Holland: Elsevier Science Publishers, 1985.

[3] Y. S. Song, "Design of Fuzzy Sensor-based Fuzzy Combustion Control System for Refuse Incinerator," *Master Thesis paper in KAIST*, Dept. of Automation and Design Engineering, 1997.

[4] Z. Bien, D. Kang, and S. Yang, "Programming Approach for fuzzy model-based Multiobjective Control Systems," *International Journal on Fuzziness, Uncertainty, and Knowledge-Based Reasoning*, Vol. 7, No. 4, pp. 289-292, 1999.

[5] T. Lim and Z. Bien, "FLC Design for Multi-Objective System," *Journal of Applied Mathematics and Computer Science*, vol. 6, no. 3, pp. 565-580, 1996.

[6] K. Kim and J. Kim, "Multicriteria fuzzy control," *Journal of Intelligent and Fuzzy Systems*, Vol. 2, pp. 279-288, 1994.

[7] Ginsberg, S. Weiss, and P. Politakis, "Automatic knowledge base refinement for classification systems," *Artificial Intelligence*, Vol. 35, pp. 197-226, 1988.

[8] W.Pedrycz, ed., *Fuzzy control and fuzzy systems*, Research Studies Press, 1993.

[9] Z. Bien and W. Yu, "Extracting core information from inconsistent fuzzy control rules," *Fuzzy Sets and System*, vol. 71, no. 1, pp. 95-111, April. 1995

[10] R. S. Sutton and A. G. Barto, *Reinforcement Learning An Introduction*, MIT Press, 1998.

[11] R. S. Sutton, A. G. Barto, and R. J. Williams, "Reinforcement Learning is Direct Adaptive Optimal control," *IEEE Control Sys*tems, pp. 19-22, 1992.

[12] L. Jouffe, "Fuzzy Inference System Learning by Reinforcement Methods," *IEEE Transaction on Systems, Man, and Cybernetics-Part C*, Vol. 28, No. 3, pp. 338-355, 1998.

[13] Y. J. Lai and C. L. Hwang, *Fuzzy Multiple Objective Decision Making*, Springer-Verlag, Berlin, 1994.

[14] M. Sakawa, *Fuzzy Sets and Interactive Multiobjective Optimization*, Plenum Press, New York, 1993.

[15] H. -J. Zimmermann, ed., *Fuzzy Set Theory - And Its Applications*, Dordrecht: Kluwer Academic Publishers, 1991.