Journal of Intelligent & Fuzzy Systems 33 (2017) 125–136
DOI:10.3233/JIFS-161211
IOS Press

An intelligent controller for ionic polymer metal composites using optimized fuzzy reinforcement learning

Masoud Goharimanesh[a], Elyas Abbasi Jannatabadi[a], Hossein Moeinkhah[c], Mohammad Bagher Naghibi-Sistani[b] and Ali Akbar Akbari[a,*]
[a]*Department of Mechanical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran*
[b]*Department of Electrical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran*
[c]*Department of Mechanical Engineering, University of Sistan and Baluchestan, Zahedan, Iran*

Abstract. This paper proposes an optimized fuzzy reinforcement-learning algorithm to control ionic polymer metal composites. The IPMC has been made by thin polymer membrane with metal electrodes plated chemically on the both faces. Its application is widely and may be used as the artificial muscle due to the large bending strain at low voltages. Although there are some controllers designed in the literature, most of them are model-based and for this reason are not used widely. In this study, a free model controller based on fuzzy is considered. The fuzzy rule making is not straightforward and must be taken by an expert, so an algorithm based on the reinforcement learning is employed to make the rule sets strongly. After learning the fuzzy sets, firstly, the reinforcement learning parameters have been optimized using the Taguchi method and then an optimized algorithm based on the genetic is started to tune up the configuration of membership functions for controller designing. The effectiveness of the reported controller for the IPMC actuator is confirmed by simulation and experimental results.

Keywords: Ionic polymer metal composite, fuzzy, reinforcement leaning, genetic algorithm

1. Introduction

Ionic polymer metal composites (IPMCs) have attracted attention of many researchers for their potential applications in a large variety of engineering areas [1–12]. Shahinpoor et al. introduced an electroactive polymer material which shows very large deformations in response to the low input voltages [13, 14]. They clarified several advantages of IPMC actuators including large strokes with low voltages, soft surface and low stiffness and good performance in wet and underwater environments. In 1994 Kanno et al. developed the first empirical models of the IPMC actuator by using the data from the step response data of an IPMC strip [15]. They proposed a fourth-degree transfer function by using this data.

Movement manipulating of an IPMC based materials is one the most challenging issues which has attracted many scholars. In 2004, Bhat et al. designed a feedback controller for an IPMC actuator which was implemented in a cantilever configuration to reduce settling time, percentage overshoot and the steady state error [16]. Five years later, Andres Hunt et al. used an IPMC actuator to stabilize an inverted pendulum for about 5 minutes [17]. In their controller algorithm, a state-space model is developed for the system, in which a linear quadratic regulator (LQR)

*Corresponding author. Ali Akbar Akbari, Department of Mechanical Engineering, Ferdowsi University of Mashhad, P.O. Box 9177948974, Mashhad, Iran. Tel.: +98 9151115611; E-mail: akbari@um.ac.ir.

controller is coupled with an observer. In addition, one year later, D. Liu et al. designed a controller for micro–manipulation by exploiting an IPMC actuated rotary linkage [18]. In particular, a Proportional, Integral (PI) controller was initially developed to control the tip displacement of the mechanism. Then for tuning the performance of the PI controller an adaptive nonlinear tuning method called Iterative Feedback Tuning was developed. In 2012, Lee et al. unveiled a novel controlling strategy for the target mobile robot units by using an ionic polymer–metal composite actuator and a microwave link [19].

Since smart materials like IPMC have been employed for improving the function of an underwater applications like biomimetic robots, a variety of algorithms were suggested to develop the control performance in the underwater conditions for various reference motions [20, 21]. In recent years, several studies have been performed to optimize motion control of an IPMC actuators with adaptive models [22, 23]. A control algorithm for manipulating an IPMC with different shapes and dimensions was presented by Lina Hao et al. In this study a semi-physical sliding mode control is proposed for controlling both deflection and force of Multi–IPMCs without changing any parameter in the control system [24]. Nevertheless, controlling this kind of smart material is not straightforward and usually deals with nonlinearities and approximate physical models [5, 6, 8, 9]. In all of the proposed methods mentioned in the literature, control methods are using a linear or a nonlinear model for the IPMC [6, 9]. Although modeling of these polymers is not completely defined, they respond properly in the linear regime [6]. Recently an accurate model was generated to simulate the helix IPMC behavior [25]. While defining a comprehensive model is a bewildering task, the uncertainties and lack of a reliable controller for nonlinear dynamics are the most challenging problems. Due to these problems, we decided to use a control method which is needless of having a model. This method is based on the learning by the reinforcements [26–32]. Reinforcement Learning (RL) is a powerful tool for finding the optimum policy for a process [31, 33–36]. RL uses the environment feedback and make a signal named reinforcement. This signal may be a reward or a punishment. Agent is same as process and action is as controller signal. RL aims to find the best action for each state which agent (process) wants to move. Q-learning is a simple algorithm which is used in this concept. This algorithm has a lookup table named Q table [37, 38]. It estimates the discounted future rewards for taking actions from given states. When reinforcement learning integrated with the fuzzy logic, it can be more reliable because of the continuous behavior of the fuzzy [39]. RL helps fuzzy controller to find the rules in the best way, and fuzzy sets aid the RL to have a full domain state-action approximation. Therefore, fuzzy-RL is used to set the rules of a process without the existence of the model [39]. Previously, we built a fuzzy-RL toolbox which is capable of taking into account any model in MATLAB-SIMULINK [40]. After building the fuzzy rules using reinforcement learning, the second round of the optimization will be started. In this stage, the membership functions will be tuned by an evolutionary procedure such as the genetic algorithm.

2. Modeling

Although we established the power of fuzzy reinforcement learning for the free model, a model which is developed in [6], as shown in Equation (1), is used for the proposed simulation.

$$\frac{P(s)}{V(s)} = \frac{p_2 s^2 + p_1 s + p_0}{s^4 + q_3 s^3 + q_2 s^2 + q_1 s + q_0} \tag{1}$$

Where the uncertainties of the present parameters (pi, qi) are shown in (2). These parameters describe the Laplace transfer function of position (P(s)) to voltage (V(s)) for an IPMC.

$$\begin{aligned} p_0 &\in \left[0.0527,\ 0.1582\right], \quad p_1 = 0.0774; \\ p_2 &\in \left[1.647 \times 10^{-4},\ 4.941 \times 10^{-4}\right] \\ q_0 &\in \left[73.5376,\ 661.8387\right] \\ q_1 &\in \left[335.6968,\ 1.0263 \times 10^{3}\right] \\ q_2 &\in \left[39.0918,\ 95.8073\right] \\ q_3 &\in \left[16.6336\ 48.0608\right] \end{aligned} \tag{2}$$

3. Fuzzy reinforcement learning strategy

Learning can be served as an apparatus for forming the best arrangements in a procedure [33, 41]. Reinforcement learning interacts with its surroundings and produces a reinforcement signal. This signal may be considered as a reward or a punishment based on

the evaluation of a state. In RL method, a controller signal is often called action and all reinforcement-learning agents (they are known as process or decision maker) affect their environment by the means of the actions. The main aim of RL is to discover what actions for each state direct the system to best performance [33]. In this paper, we used Q-learning which is one of the famous methods of reinforcement learning [37]. In Q-learning, the agent provides a table of expected discounted reward for each state-action pair [42]. The agent will then learned from these rewards what to do in order to maximize the reward for each state and lead the system to the best controlling policy.

The algorithm and detailed procedure of the suggested method is depicted in Fig. 1 and Table 1, respectively. This algorithm has a lookup table named Q table. It tries to estimate the discounted future rewards for taking actions from given states.

In traditional reinforcement learning, states must be discretized because the agent deals with its environment in discrete time steps [33]. Because of the state discretization, traditional RL requires a lot of memory and cannot be applied when dealing with continuous- state problems [40]. In this situation, states should be approximated using function approximators such as fuzzy inference systems (FIS) [32, 43] or gradient methods [39]. However solutions which provided with these methods, suffers from slow convergence. The proposed method in this paper is fuzzy Q-learning, where each state is the result of a rule set [43, 44].

In this case, all of the states are considered as the inputs of fuzzy and action is defined as the output. Figure 2 presents the Q-Learning principals with their equations. In this method, the Takagi-Sugeno fuzzy inference system (TS-FIS) is used and all of the rules between the input membership functions and the constant outputs, are generated by Q-learning algorithm [44].

$$a(x) = \sum_{i=1}^{N} (\alpha_i(x) \times a_i) / \sum_{i=1}^{N} \alpha_i(x)$$

$$Q(x,a) = \left(\sum_{i=1}^{N} \alpha_i(x) \times q\left[i.i^{\dagger}\right] \right) / \sum_{i=1}^{N} \alpha_i(x)$$

$$V(x) = \sum_{i=1}^{N} \left(\alpha_i(x) \times q\left[i.i^{*}\right]\right) / \sum_{i=1}^{N} \alpha_i(x)$$

$$\Delta Q = r + \gamma V(y) - Q(x,a)$$

$$\Delta q\left[i.i^{\dagger}\right] = \alpha \Delta Q \alpha_i(x) / \sum_{i=1}^{N} \alpha_i(x)$$

In the above equations, α and α_i are learning rate and truth-value, respectively. Figures 3–4 show membership functions related to output/desired

Initialize Q(s,a) arbitrarily
Repeat for each episode:
Initialize s
Repeat for each time step:
Choose a from s using policy derived from Q(s,a) (e.g., epsilon-greedy)
Take action a, observe r, s'

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha_t(s_t,a_t) \times \left[R_{t+1} + \gamma \times \max Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)\right]$$

s←s'
Until s is terminal

Fig. 1. Q-learning Algorithm [33].

Table 1
Learning terms definition

Symbol	Description
$Q(s_t, a_t)$	Old value
$\alpha_t(s_t, a_t)$	Learning rate
R_{t+1}	Reward
γ	Discount factor
$\max Q(s_{t+1}, a_{t+1})$	Maximum of future value
$R_{t+1} + \gamma \times \max Q(s_{t+1}, a_{t+1})$	Learned value

Fuzzy system is launched by type of Takagi-Sugeno. For j^{th} input, the number of i^{th} membership functions is $n\left(mf_i^j\right)$

Qtable $(n \times m)$ is generated initially where n is the all of relations between mf_i^j and m is the discrete number of actions.

For each episode:

- Observe the state (s)
- The truth value is defined
- The reward value is calculated by reward function
- Exploring is running by ε - greedy
- An action is calculated by (3)
- The next state is observed by the previous action (s')
- Qtable is updated by (4-6)
- Let (s') be the new state

Fig. 2. Fuzzy Q-learning Algorithm.

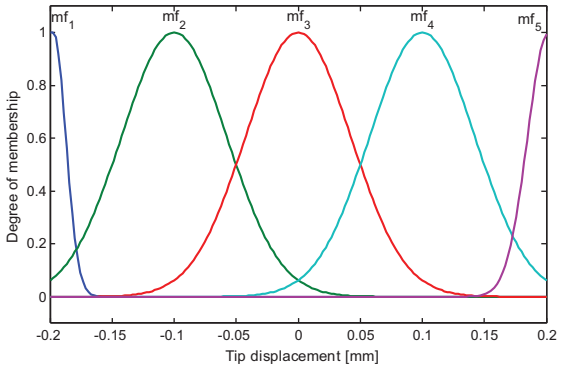


Fig. 3. Membership functions for tip displacement (output and desired).

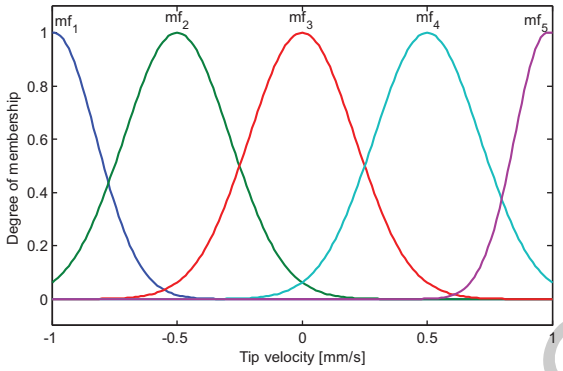


Fig. 4. Membership functions for tip velocity (output and desired).

displacement and output/desired velocity. The number of states and actions in this study are 625 and 1, respectively. The relationship between these functions are established with the help of reinforcement learning.

In this paper, the action is a control voltage in the domain of –5 to 5 volts. In addition, Q-learning which is one the robust methods of RL algorithm is exploited and a policy is generated with the help of reward function. The reward function describes the quality of every performance that the system makes in each transition. The proposed reward function can be seen in (8).

$$R = \left[\, |p_o - p_d| \;\; |v_o - v_d| \,\right] \times \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix} \times \left[\, |p_o - p_d| \;\; |v_o - v_d| \,\right]^T \quad (3)$$

Where *po*, *pd* and *vo, vd* are observed and desired tip position and velocity, respectively. The inner matrix in this reward function insists on the tip position due to the greater weight of this factor (10>1). Learning rate and discount factor are considered 0.1 and 0.9, respectively. The procedure for obtaining the optimum value for learning rate, discount factor and epsilon greedy parameter are discussed more in Section 5.

4. Control process

Fuzzy system is a powerful method for interpreting human's language and dealing with decision-making problems encompassing uncertainties [41, 42]. There are two types of fuzzy interface systems including Mamdani-type FIS and Takagi-Sugeno-type FIS. Output membership functions in Sugeno's fuzzy interface method are either linear or constant and this differentiates the Mamdani method from Sugeno. The main drawback of fuzzy controllers is the arrangement of basic rule base that would fulfill desire control targets. One method that can be used to tackle this issue and find an optimal solution is RL. Herein, the efficient functional rules for Takagi-Sugeno-type fuzzy are generated and tuned through Q-learning. In particular, we use Takagi-Sugeno FIS with four inputs and one output which is the voltage of IPMC. The rules between these parameters are constructed through trial-error interactions with the Q-learning algorithm. Fuzzy control system is organized as shown in Fig. 5.

The controller and the implementation of the proposed model in Simulink is demonstrated in Fig. 6.

Using fuzzy-RL, a powerful policy is extracted as shown in Fig. 7. The mentioned policy depicts the relationship of the voltage (–5 to 5 volt) versus output and desired tip displacement, which is the goal of the control problem.

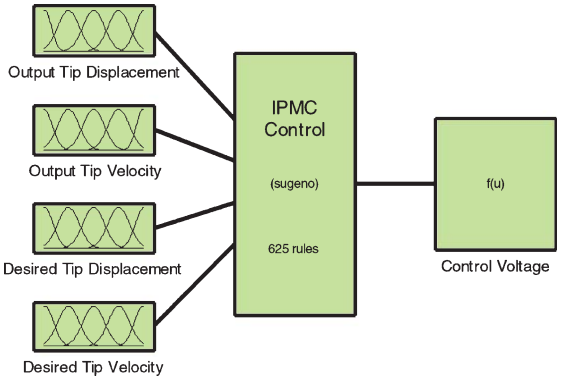


Fig. 5. Fuzzy set for IPMC control system.

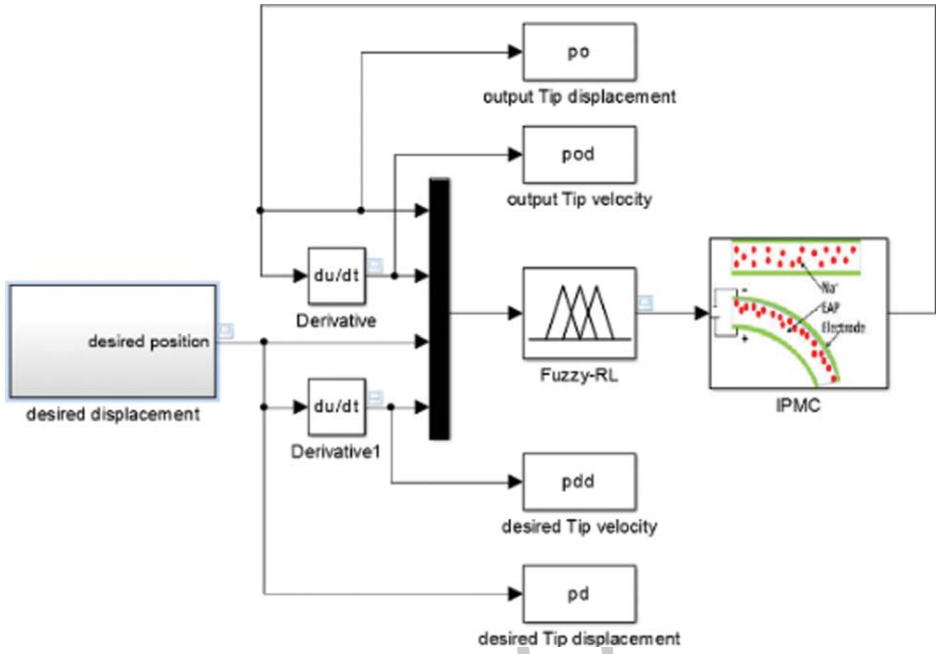


Fig. 6. SIMULINK environment for implementing the model and controller.

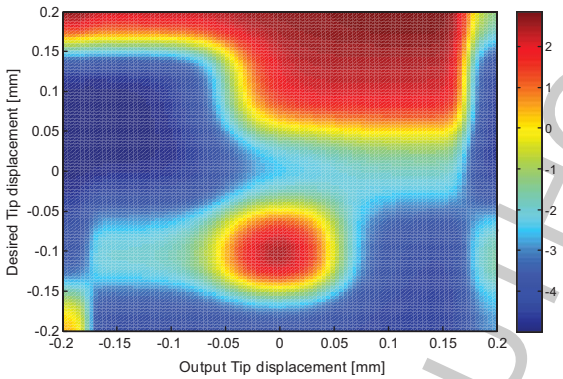


Fig. 7. Policy shows the voltage versus output and desired tip displacement.

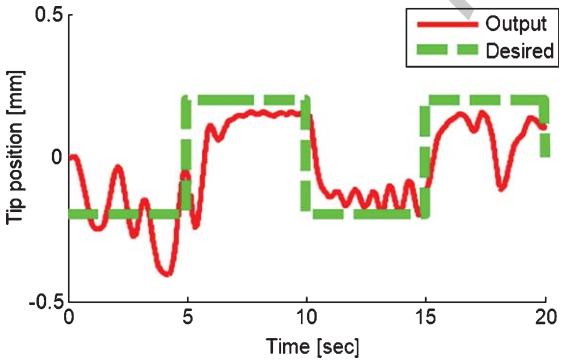


Fig. 8. A basic controller during the first of the learning process (Tip displacement in [mm] versus the time [sec]).

Figure 8 shows the control of the desired signal using fuzzy reinforcement learning in the earlier stage. As it is clear in this graph, the controller cannot find the desired signal well.

Although the fuzzy rules can be made by reinforcement learning, better result will be achieved by an optimizing procedure. First, maximizing the rewards can be done by finding optimum Q-learning parameters like learning rate, discount value and the main parameter of exploration algorithm. Second, changing the structure of fuzzy membership functions using an evolutionary procedure like genetic algorithm [45]. The layout of this procedure is shown in Fig. 9.

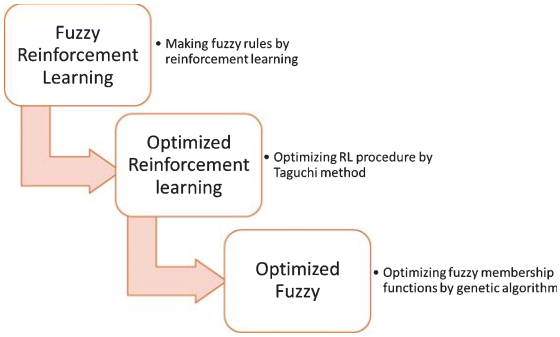


Fig. 9. Procedure diagram of proposed optimized fuzzy reinforcement learning algorithm.

5. Optimization of fuzzy reinforcement learning

In this section an optimization procedure is considered to improve fuzzy reinforcement learning result. After generating rules by reinforcement learning, three involving parameters in Q-learning are optimized using the Taguchi method. Then fuzzy-RL will continue with these new parameters value to reach to an acceptable intelligence. This status occurs when Q-table converges to a fixed pattern in high iteration process. After that, learning will be stopped and fuzzy membership functions will be tuned up using the genetic algorithm.

5.1. RL parameter optimization

In order to identify optimum solution for Q-learning procedure, three involved parameters, epsilon parameter, learning rate and discount factor are considered to get the maximum reward in a period time of learning. In addition, a classical reinforcement learning which was considered in a new environment in [42] is employed. These three parameters can change in an interval between 0 and 1. It is time consuming to investigate most of the possible values and obtain the best parameters. In this case, a design of experiment based on the Taguchi method is employed. In this method a clear insight on all possible values could be gained by the minimum number of experiments in an orthogonal table.

The Taguchi method is an industrial optimization method to obtain the optimum levels for some involving parameters [46–49]. This technique decreases the number of experiments due to the orthogonal array, therefore it can be suitable for the studies which are not able to cover all of the possible experiments [50, 51]. This method has seven steps which are demonstrated in Fig. 10.

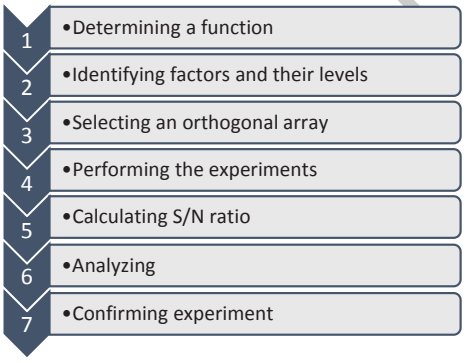


Fig. 10. Seven steps of designing experiments based on Taguchi method.

Three parameters proposed in this study and their levels are given in Table 2.

Table 2
Experiments parameters and their levels

Level	Epsilon parameter A	Learning rate B	Discount factor C
1	0.01	0.01	0.01
2	0.05	0.05	0.05
3	0.1	0.1	0.1
4	0.5	0.5	0.5
5	0.9	0.9	0.9

Table 3
Orthogonal table of experiments, parameters levels and their associated reward

No.	A	B	C	Reward
1	1	1	1	1.427844
2	1	2	2	2.138186
3	1	3	3	1.951669
4	1	4	4	2.799058
5	1	5	5	3.894814
6	2	1	2	2.545467
7	2	2	3	2.155887
8	2	3	4	2.53877
9	2	4	5	3.027735
10	2	5	1	1.809457
11	3	1	3	1.676096
12	3	2	4	2.492116
13	3	3	5	2.839745
14	3	4	1	1.649241
15	3	5	2	1.66965
16	4	1	4	1.679834
17	4	2	5	1.496886
18	4	3	1	1.445647
19	4	4	2	1.499083
20	4	5	3	1.553908
21	5	1	5	1.616732
22	5	2	1	1.588492
23	5	3	2	1.597852
24	5	4	3	1.61183
25	5	5	4	1.621279

Table 4
Signal to noise ratio values

Level	A	B	C
1	7.251	4.875	3.962
2	7.531	5.746	5.351
3	6.056	6.048	4.988
4	3.711	6.114	6.733
5	4.121	5.888	7.635
Delta	3.820	1.240	3.673
Rank	1	3	2

Table 5
Signal to noise ratio values

	DF	SeqSS	AdjSS	AdjMs	F	P
A	1	2.8181	2.8181	2.8181	16.48	0.001
B	1	0.1701	0.1701	0.1701	0.99	0.330
C	1	2.8677	2.8677	2.8677	16.77	0.001
Error	21	3.5914	3.5914	0.1710		
Total	24	9.4473				

An orthogonal table with 25 rows is given in Table 3. All of the three parameters are interacting with each other by their levels described in Table 2. The last column in Table 3 is the reward value obtained for each experiment.

The "Larger Better" criterion was used in this investigation. Using LB, the description of the loss function (L) for RMS output, y_i of n repeated experiments and The S/N ratio η_{ij} can be expressed as (9) and (10), respectively.

$$L_{SB} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{y_i^2} \tag{4}$$

$$\eta_{ij} = -\log(L_{ij}) \tag{5}$$

Where two indices i and j represent ith performance characteristic and jth experiment, respectively. The S/N ratio for each experiment of L_{25} are shown in Table 4 and Fig. 11.

As Fig. 11 shows the optimum values for epsilon greedy parameter, learning rate and discount factor are 0.05, 0.5 and 0.9, respectively. For further clarification, analysis of variance is used. As **Error! Reference source not found.** shows the p-value for learning rate parameter is not as low as the other parameters. As a result, it could be concluded that learning rate parameter is not as effective during the simulation as other parameters.

To identify the interaction of these parameters, a full quadratic model is employed. In this model, $R^2 = 88.76\%$, R^2 (pre) $= 58.32\%$, R^2 (Adj) $= 82.02\%$. Two informative graphs are shown in Fig. 12 using response surface method. These results confirm how the low value of epsilon parameter, the medium level of learning rate and the highest value of discount rate can result in the maximum level of reward.

5.2. *Fuzzy membership function optimization*

Genetic algorithm (GA) are family members of computational models that are inspired from evolutionary events, such as mutation and crossover. The performance of GA has been evaluated in seeking an efficient solution in a large search space, and the consequence is that GA could be integrated effectively with intensive search procedure, such as optimal fuzzy rules searching [43]. These solutions are selected according to a fitness function; then new members are generated over the administration of crossover operator. This operation is continued routinely until the cost function reaches to its minimum value. Figure 13 illustrates a framework for GA pro-

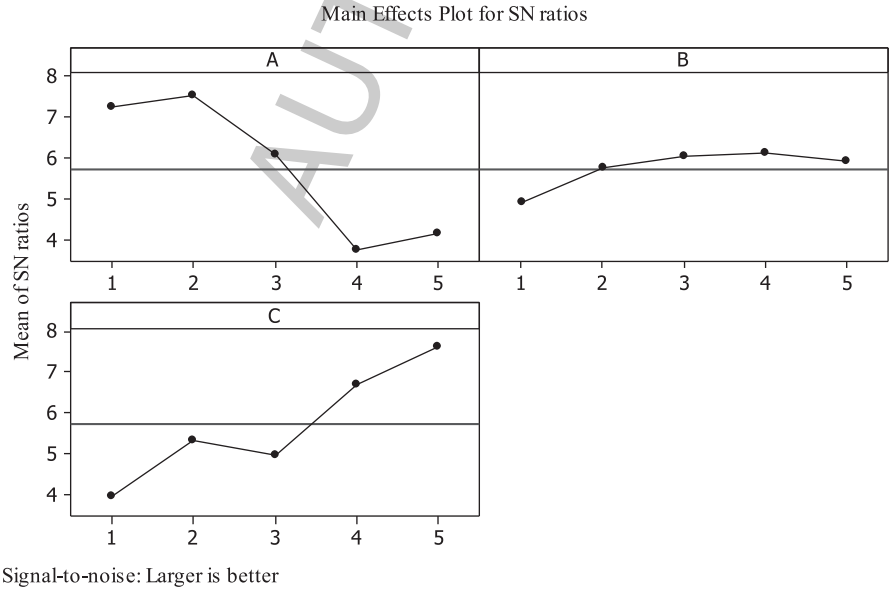


Fig. 11. Signal to noise ratio graph.

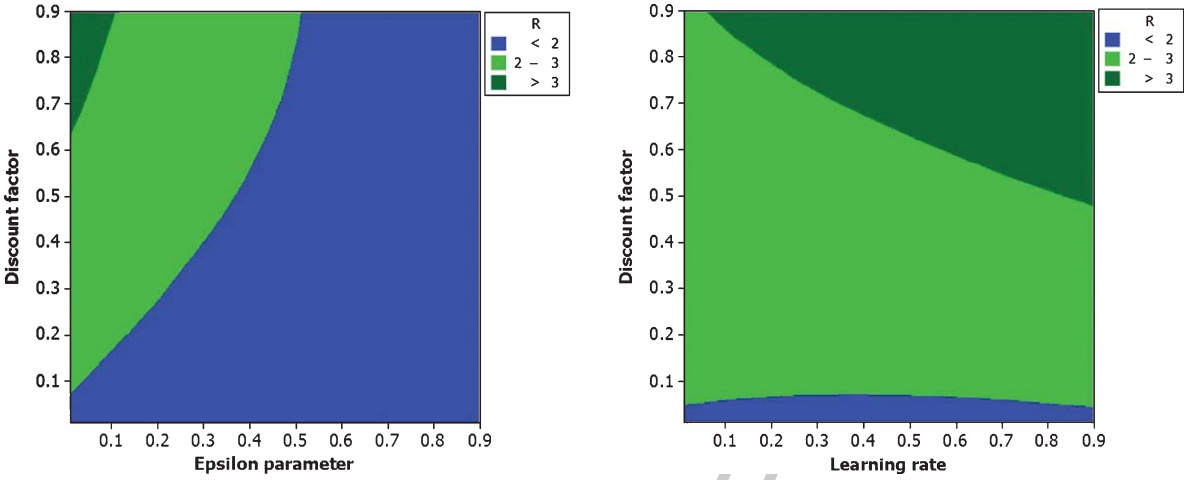


Fig. 12. Results of response surface method.

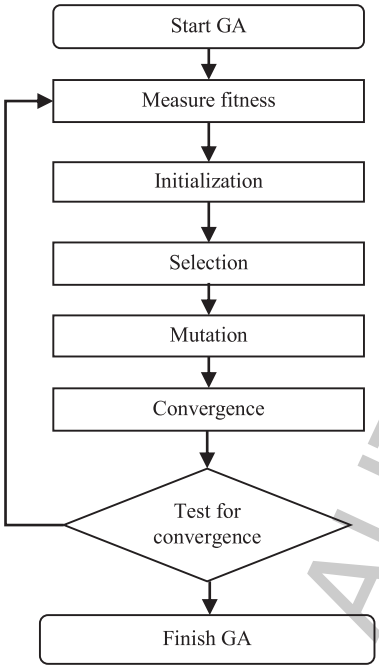


Fig. 13. Biological genetic algorithm process flow.

cess; also, the assigned variables to implement the method in MATLAB are available in Table 6.

6. Simulation and results

Exploitng fuzzy Q-learning reduces the search area for the GA and results in a faster running time for optimization. In the proposed method, GA has been employed to investigate optimal relationships between inputs and output in membership functions. The final membership functions by applying this searching technique are shown in Fig. 14 which illustrates four membership functions before and after the optimizing. For the fourth input, desired velocity, we don't see any difference in changing membership functions. It is because the desired curve is constant in the period of time and so the first differentiation of the position is zero. This causes to see no difference between the memberships functions of the desired velocity even the genetic algorithm was implemented.

After the fine-tuning, we can see a more reliable response of the mentioned controller.

As Fig. 15 shows, the Fuzzy-RL optimized by the genetic algorithm could follow the desired signal. Moreover, to illustrate the applicability of the proposed controller we have considered a variable desired signal. To show the error of the output and the desired signal, two criteria are used. The Root

Table 6
Properties of the conducted genetic algorithm

Option	Value
Crossover function	Heuristic
Crossover fraction	0.8
Elite number	2
Initial penalty	10
Mutation function	Adaptive feasible
Penalty factor	100
Population initial range	[–1,1]
Population size	100
Population type	Bit string
Selection function	Stochastic uniform

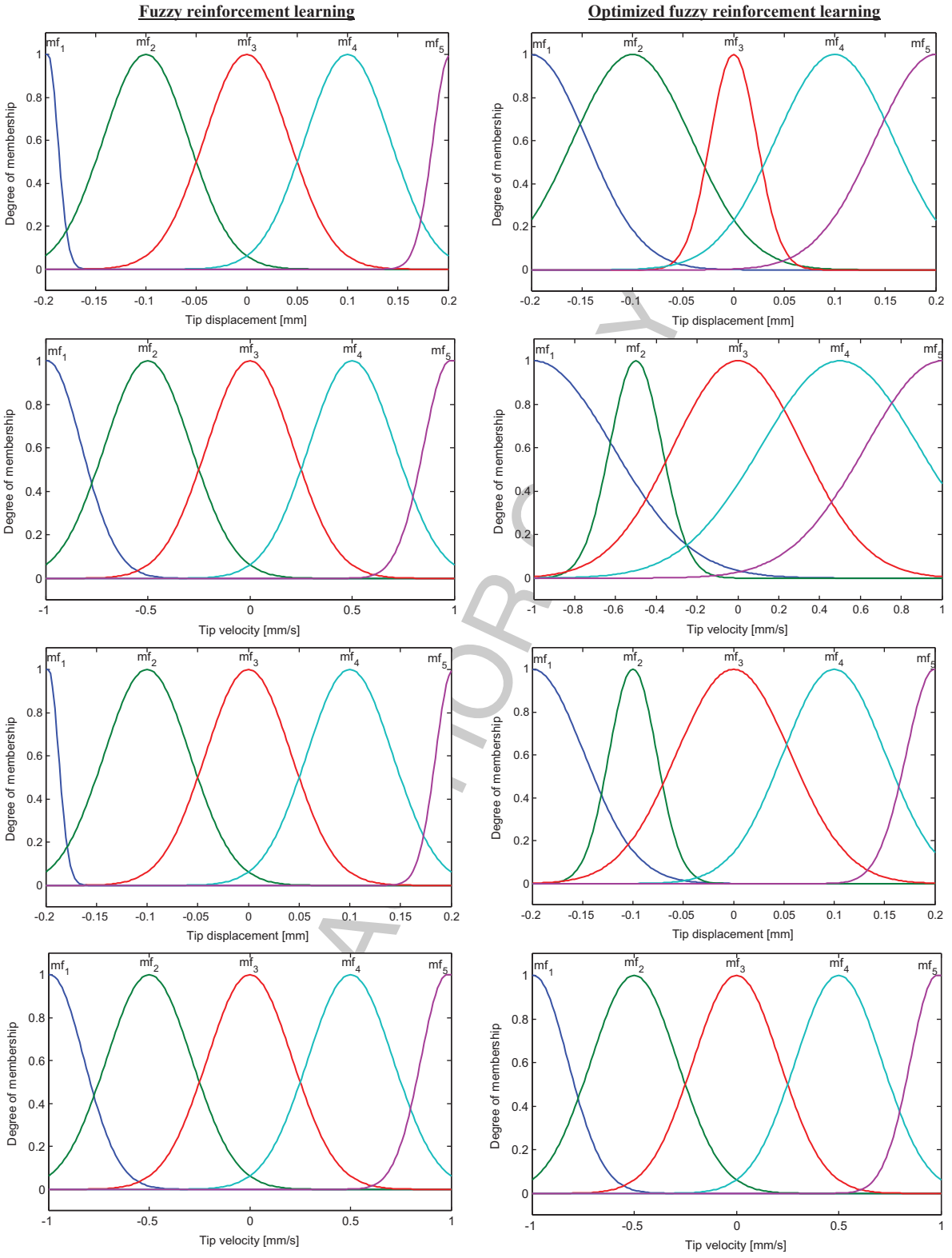
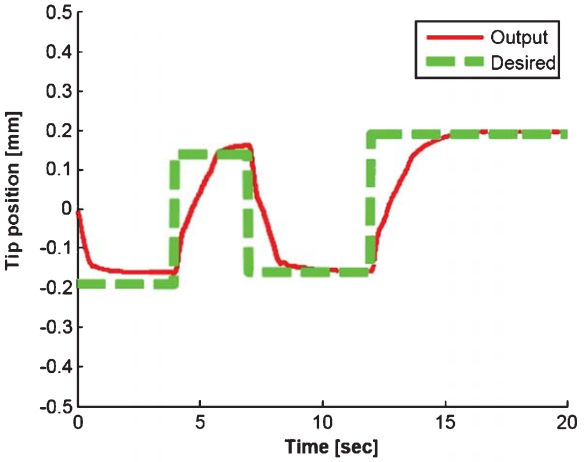
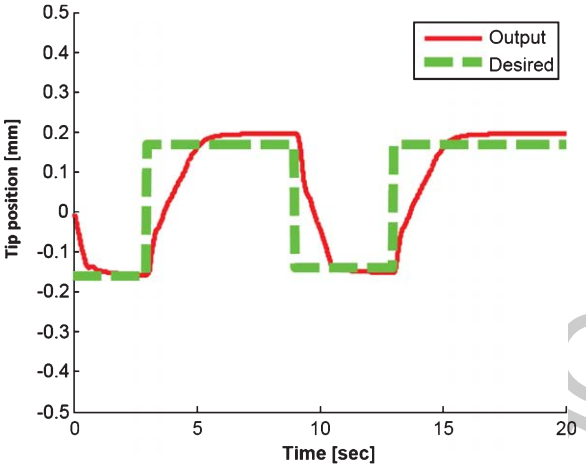


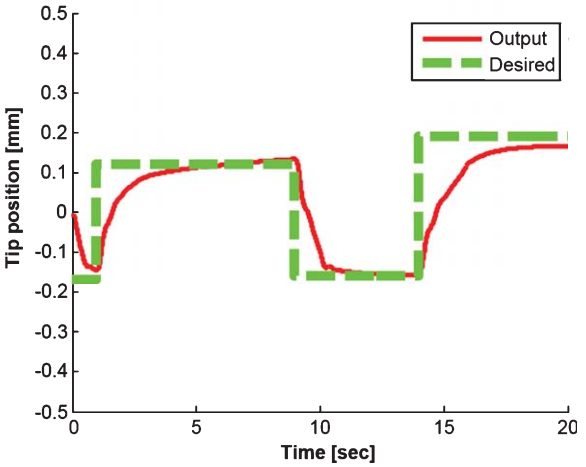
Fig. 14. The past and tuned membership functions optimized by genetic algorithm after being expert by reinforcement learning.



RMSE=0.0910, ITAE=19.6569



RMSE= 0.0913, ITAE=23.1516



RMSE=0.0815, ITAE = 25.8409

Fig. 15. An expert controller after learning process (tip displacement in [mm] versus the time [sec]) to track the random desired position.

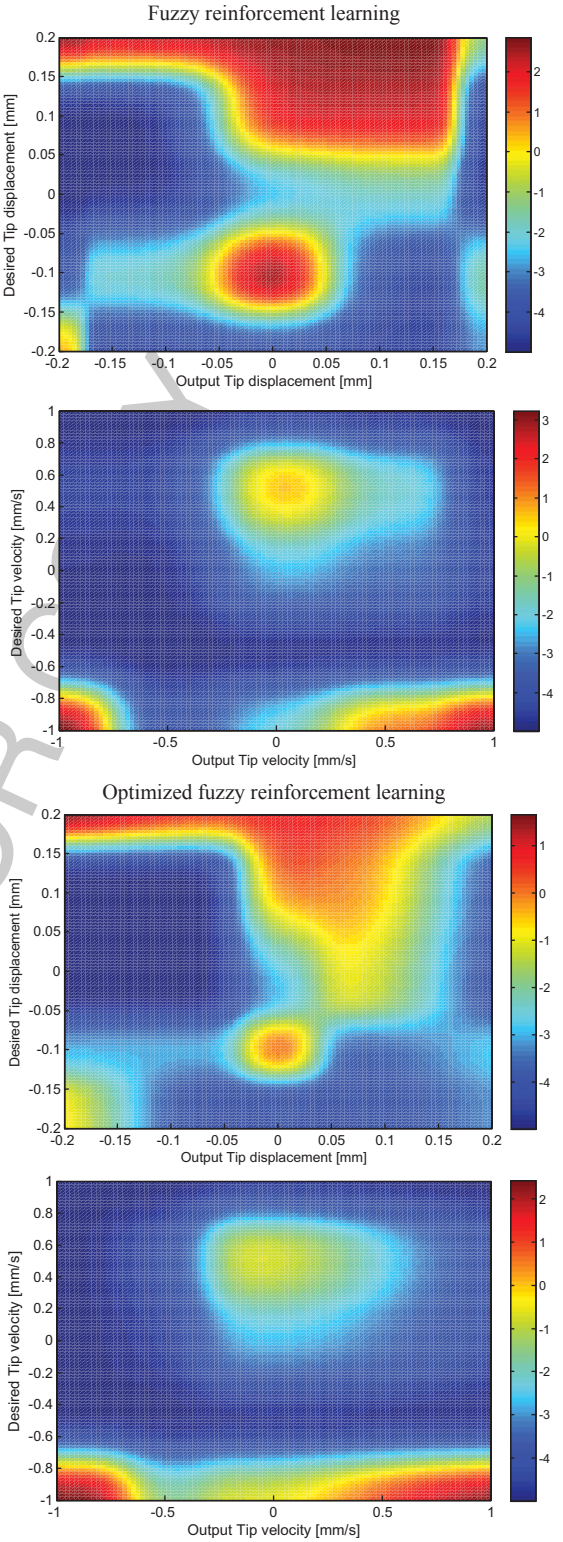


Fig. 16. Policy achieved for Fuzzy-RL and optimized Fuzzy-RL.

Mean Square Error (RMSE) is usually used to measure the difference between values predicted by a model and the values actually observed from the environment that is being modelled. These individual differences are also called residuals, and the RMSE serves to aggregate them into a single measure of predictive power. The RMSE in this study is described as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (p_o - p_d)^2}{n}} \tag{6}$$

Where P_o is observed or output values and P_d is desired values at time/place i.

The integral of time-weighted absolute error (ITAE) integrates the absolute error multiplied by the time over time. This weighs errors which exist after a long time much more heavily than those at the start of the response. This criterion is described as (12).

$$ITAE = \int_0^{20} |e| t dt \tag{7}$$

The policy of fuzzy reinforcement learning and optimized reinforcement learning are compared with all of the inputs in Fig. 16. This shows a dramatic change of variable behavior between the desired tip displacement versus the tip displacement and the desired tip velocity versus the velocity which are compared in these policies.

7. Conclusion

In this paper, we established an intelligent method to control the smart materials like IPMC. As mentioned, the IPMC models are not thoroughly exploited and for this reason, there is not a comprehensive controller, which is reliable for the uncertainties and many disturbing conditions. Fuzzy controller can be employed in these conditions but the main problem in this method is finding the most efficient IF-THEN rules. As discussed, reinforcement learning can aid the fuzzy to set the rules in the suboptimal policy. In this research, we found that the fine-tuning for the membership functions could strengthen the performance of the proposed controller in which it can let the system follow the desired signal.

References

[1] Y. Xiao, K.K. Poornesh, L. Wendling and C. Cho, Ex-situ temperature & humidity aging effect on the tensile behavior of Nafion N117 membrane used in Ionic Polymer-Metal Composite actuators, *Materials Letters* (2014).

[2] M. Farid, Z. Gang, T. Linh Khuong and Z.Z. Sun, Grasshopper knee joint-torque analysis of actuators using ionic polymer metal composites (IPMC), *Journal of Biomimetics, Biomaterials, and Tissue Engineering* **19** (2014), 13–23.

[3] A.A.A. Moghadam, W. Hong, A. Kouzani, A. Kaynak, R. Zamani and R. Montazami, Nonlinear dynamic modeling of ionic polymer conductive network composite actuators using rigid finite element method, *Sensors and Actuators A: Physical* (2014).

[4] R. Caponetto, S. Graziani, F.L. Pappalardo and F. Sapuppo, Experimental characterization of ionic polymer metal composite as a novel fractional order element, *Advances in Mathematical Physics* **2013** (2013).

[5] R. Caponetto, S. Graziani, F.L. Pappalardo and M.G. Xibilia, A Comparison between Robust and Parameterized Controllers for Fractional Order Modeled Ionic Polymeric Metal Composite Actuator, in *Fractional Differentiation and its Applications*, 2013, pp. 905–910.

[6] H. Moeinkhah, A. Akbarzadeh and J. Rezaeepazhand, Design of a robust quantitative feedback theory position controller for an ionic polymer metal composite actuator using an analytical dynamic model, *Journal of Intelligent Material Systems and Structures* (2013), 1045389X13512906.

[7] H. Moeinkhah, J.-Y. Jung, J.-H. Jeon, A. Akbarzadeh, J. Rezaeepazhand, K. Park, et al., How does clamping pressure influence actuation performance of soft ionic polymer–metal composites? *Smart Materials and Structures* **22** (2013), 025014.

[8] K.K. Ahn, D.Q. Truong, D.N.C. Nam, J.I. Yoon and S. Yokota, Position control of ionic polymer metal composite actuator using quantitative feedback theory, *Sensors and Actuators A: Physical* **159** (2010), 204–212.

[9] Z. Chen and X. Tan, A control-oriented and physics-based model for ionic polymer–metal composite actuators, *Mechatronics, IEEE/ASME Transactions on* **13** (2008), 519–529.

[10] B. Kim, D.-H. Kim, J. Jung and J.-O. Park, A biomimetic undulatory tadpole robot using ionic polymer–metal composite actuators, *Smart Materials and Structures* **14** (2005), 1579.

[11] S. Nemat-Nasser, Micromechanics of actuation of ionic polymer-metal composites, *Journal of Applied Physics* **92** (2002), 2899–2915.

[12] S. Nemat-Nasser and J.Y. Li, Electromechanical response of ionic polymer-metal composites, *Journal of Applied Physics* **87** (2000), 3321–3331.

[13] M. Shahinpoor, Conceptual design, kinematics and dynamics of swimming robotic structures using ionic polymeric gel muscles, *Smart Materials and Structures* **1** (1992), 91.

[14] M. Shahinpoor and K.J. Kim, Ionic polymer-metal composites: I. Fundamentals, *Smart Materials and Structures* **10** (2001), 819.

[15] R. Kanno, A. Kurata, M. Hattori, S. Tadokoro, T. Takamori and K. Oguro, Characteristics and modeling of ICPF actuator, in *Proceedings of the Japan-USA Symposium on Flexible Automation*, 1994, pp. 691–698.

[16] N. Bhat and W. Kim, Precision force and position control of an ionic polymer metal composite, *Proceedings of*

the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering **218** (2004), 421–432.

[17] A. Hunt, Z. Chen, X. Tan and M. Kruusmaa, Control of an inverted pendulum using an Ionic Polymer-Metal Composite actuator, in *Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on*, 2010, pp. 163–168.

[18] D. Liu, A. McDaid, K. Aw and S. Xie, Position control of an ionic polymer metal composite actuated rotary joint using iterative feedback tuning, *Mechatronics* **21** (2011), 315–328.

[19] J. Lee, W. Yim, C. Bae and K. Kim, Wireless actuation and control of ionic polymer–metal composite actuator using a microwave link, *International Journal of Smart and Nano Materials* **3** (2012), 244–262.

[20] Q. Shen, T. Wang, L. Wen and J. Liang, Modelling and fuzzy control of an efficient swimming ionic polymer-metal composite actuated robot, *Int J Adv Robot Syst* **10**(350) (2013).

[21] H.-L. Xing, J.-H. Jeon, K. Park and I.-K. Oh, Active disturbance rejection control for precise position tracking of ionic polymer–metal composite actuators, *Mechatronics, IEEE/ASME Transactions on* **18** (2013), 86–95.

[22] X. Chen, Adaptive Control for Ionic Polymer-Metal Composite Actuator Based on Continuous-Time Approach, in *World Congress*, 2014, pp. 5073–5078.

[23] N.T. Thinh and D.T. Dung, Adaptive Neuro-Fuzzy Control for Ionic Polymer Metal Composite Actuators, in *Robot Intelligence Technology and Applications 2*, ed: Springer, 2014, pp. 939–947.

[24] L. Hao, Y. Chen and Z. Sun, The sliding mode control for different shapes and dimensions of IPMC on resisting its creep characteristics, *Smart Materials and Structures* **24** (2015), 045040.

[25] H. Moeinkhah, J. Rezaeepazhand, A. Akbarzadeh and I.K. Oh, Accurate dynamic modeling of helical ionic polymer-metal composite actuator based on intrinsic equations, *IEEE/ASME Transactions on Mechatronics* **20** (2015), 1680–1688.

[26] F. Wang, K. Xu, Q.S. Zhang, Y.W. Wang and X.X. Zheng, A multi-step neural control for motor brain-machine interface by reinforcement learning, *Applied Mechanics and Materials* **461** (2014), 565–569.

[27] F.L. Lewis and D. Liu, *Reinforcement learning and approximate dynamic programming for feedback control*, vol. 17, John Wiley & Sons, 2013.

[28] F.L. Lewis and D. Vrabie, Reinforcement learning and adaptive dynamic programming for feedback control, *Circuits and Systems Magazine, IEEE* **9** (2009), 32–50.

[29] J. Valasek, J. Doebbler, M.D. Tandale and A.J. Meade, Improved adaptive-reinforcement learning control for morphing unmanned air vehicles, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* **38** (2008), 1014–1020.

[30] J. Valasek, M.D. Tandale and J. Rong, A reinforcement learning - Adaptive control architecture for morphing, *Journal of Aerospace Computing, Information and Communication* (2005), 174–195.

[31] R.S. Sutton, A.G. Barto and R.J. Williams, Reinforcement learning is direct adaptive optimal control, *Control Systems, IEEE* **12** (1992), 19–22.

[32] H.R. Berenji, A reinforcement learning—based architecture for fuzzy logic control, *International Journal of Approximate Reasoning* **6** (1992), 267–292.

[33] R.S. Sutton and A.G. Barto, *Reinforcement learning: An introduction*, vol. 1: Cambridge Univ Press, 1998.

[34] R.S. Sutton, Generalization in reinforcement learning: Successful examples using sparse coarse coding, *Advances in Neural Information Processing Systems*, 1996, pp. 1038–1044.

[35] V. Derhami, Similarity of learned helplessness in human being and fuzzy reinforcement learning algorithms, *Journal of Intelligent & Fuzzy Systems* **24** (2013), 347–354.

[36] L.M. Friske and C.H. Ribeiro, Experiments on the use of option policies in reinforcement learning, *Journal of Intelligent & Fuzzy Systems* **13** (2002), 123–132.

[37] C.J. Watkins and P. Dayan, Q-learning, *Machine Learning* **8** (1992), 279–292.

[38] C.J.C.H. Watkins, Learning from delayed rewards, University of Cambridge, 1989.

[39] H.R. Berenji and P. Khedkar, Learning and tuning fuzzy logic controllers through reinforcements, *Neural Networks, IEEE Transactions on* **3** (1992), 724–740.

[40] A. A. Akbari and M. Goharimanesh, Yaw Moment Control Using Fuzzy Reinforcement Learning, in *Advanced Vehicle Control (AVEC14)*, 2014.

[41] L.P. Kaelbling, M.L. Littman and A.W. Moore, Reinforcement learning: A survey, *Journal of Artificial Intelligence Research* **4** (1996), 237–285.

[42] M. Goharimanesh, A.A. Akbari and M.-B. Naghibi-Sistani, Combining the principles of fuzzy logic and reinforcement learning for control of dynamic systems, *Journal of Applied and Computational Sciences in Mechanics* **27** (2015), 1–14.

[43] H.R. Berenji, Fuzzy Q-learning for generalization of reinforcement learning, in *Fuzzy Systems, 1996, Proceedings of the Fifth IEEE International Conference on*, 1996, pp. 2208–2214.

[44] A. Bonarini, A. Lazaric, F. Montrone and M. Restelli, Reinforcement distribution in fuzzy Q-learning, *Fuzzy Sets and Systems* **160** (2009), 1420–1443.

[45] M. Goharimanesh, A. Lashkaripour, S. Shariatnia and A. Akbari, Diabetic control using genetic fuzzy-PI controller, *International Journal of Fuzzy Systems* **16** (2014), 133.

[46] G. Taguchi, *Introduction to Quality Engineering: Designing Quality into Products and Processes*. Tokyo: The Organization, 1986.

[47] G. Taguchi, L.W. Tung and D. Clausing, *System of Experimental Design: Engineering Methods to Optimize Quality and Minimize Costs*. vol. 2: UNIPUB/Kraus International Publications, 1987.

[48] G. Taguchi, E.A. Elsayed and T.C. Hsiang, *Quality Engineering in Production Systems*. McGraw-Hill College, 1989.

[49] G.I. Taguchi and Y. Yokoyama, *Taguchi Methods*. vol. 2: ASI, 1994.

[50] M. Goharimanesh, A. Akbari and A.A. Tootoonchi, More efficiency in fuel consumption using gearbox optimization based on Taguchi method, *Journal of Industrial Engineering International* **10** (2014), 1–8.

[51] M. Goharimanesh and A. Akbari, Optimum parameters of nonlinear integrator using design of experiments based on Taguchi method, *Journal of Applied Mechanics* **46** (2015), 233–241.