Intelligent Switching Control of the Pneumatic Artificial Muscle Manipulators

Kyoung Kwan AHN*, and TU Diep Cong Thanh**

* School of Mechanical and Automotive Engineering, University of Ulsan, Ulsan, Korea
(Tel : +82-52-222-1404; E-mail: kkahn@ulsan.ac.kr)
** Department of Mechanical and Automotive Engineering, University of Ulsan, Ulsan, Korea
(Tel : +82-52-222-1404; E-mail: tdcthanh@yahoo.com)

Abstract: Problems with the control, oscillatory motion and compliance of pneumatic systems have prevented their widespread use in advanced robotics. However, their compactness, power/weight ratio, ease of maintenance and inherent safety are factors that could be potentially exploited in sophisticated dexterous manipulator designs. These advantages have led to the development of novel actuators such as the McKibben Muscle, Rubber Actuator and Pneumatic Artificial Muscle Manipulators. However, some limitations still exist, such as a deterioration of the performance of transient response due to the changes in the external inertia load in the pneumatic artificial muscle manipulator.
To overcome this problem, a switching algorithm of the control parameter using a learning vector quantization neural network (LVQNN) is newly proposed. This estimates the external inertia load of the pneumatic artificial muscle manipulator. The effectiveness of the proposed control algorithm is demonstrated through experiments with different external inertia loads.

Keywords: pneumatic artificial muscle, neural network, switching control, intelligent control

1. INTRODUCTION

Problems with the control and compliance of pneumatic systems have prevented their widespread use in advanced robotics. However their compactness, power/weight ratio, low cost, ease of maintenance, cleanliness, ready availability, cheap power source and inherent safety because of the compliance of compressed air, are factors that could be potentially exploited in sophisticated dexterous manipulator designs. These advantages have led to the development of novel actuators such as the McKibben Muscle, Rubber Actuator and Pneumatic Artificial Muscle Manipulators. Thus a pneumatic artificial muscle (which is abbreviated as PAM) manipulator has been applied to construct a therapy robot for cases in which high level of safety for humans is required. However, the air compressibility and the lack of damping ability of the pneumatic muscle actuator bring a dynamic delay of the pressure response and cause oscillatory motion. Therefore it is not easy to realize motion with high accuracy, high speed and with respect to various external inertial loads in order to realize a human-friendly therapy robot.

As the PAM manipulator is one of the well-known systems for safety with humans, it is preferable in contacting tasks with humans and many control strategies have been proposed. As a result, a considerable amount of research has been devoted to the development of various position control systems for the PAM manipulator. A Kohonen-type neural network was used for the position control of the robot end-effector within 1 cm after learning[1]. Recently, the authors have developed a feedforward neural network controller, where the joint angle and pressure of each chamber of the pneumatic muscle were used as learning data and the accurate trajectory following was obtained, with an error of 1[°][2].

However, for widespread use of these actuators in the field of manipulators, a high speed, precise control of the PAM manipulators is required. Among previous control approaches, PID control[3], fuzzy PD+I learning control[4], fuzzy + PID control[5], robust control[6-8], feedback linearization control[9], feedforward control + fuzzy logic[10], phase plane switching control[11], variable structure control algorithm[12] and H infinity control[13-14] have been applied to control the PAM manipulator. Though these systems were successful in addressing smooth actuator motion in response to step inputs, many of these systems used expensive servo valves and the external inertia load were also assumed to be constant or slowly varying. The external inertia load is not always exactly known and the contact force with humans is different in each case when the manipulator will be used as a therapy robot in the near future. Therefore, it is necessary to propose a new intelligent control algorithm, which is applicable to a very compressible pneumatic muscle system with various loads.

In order to overcome these problems, a learning vector quantization neural network (LVQNN) was applied as a supervisor of the traditional PID controller, which estimated the external inertia load and switched the gain of the PID controller. It was already proven by experiment on the position control of the pneumatic rodless cylinder that the LVQNN was an appropriate algorithm for the recognition of quickly-changing external loads and it had little computation time as well as an easy application to the PAM system[15].

The object of this paper is to implement proportional valves, rather than expensive servo valves, to develop a fast, accurate, inexpensive and intelligent PAM control system without regard for the changes in external inertia loads. The proposed control algorithm was verified to be very effective by experimenting with different loads.

2. EXPERIMENTAL SETUP

2.1 Experimental apparatus

The schematic diagram of the pneumatic artificial muscle manipulator is shown in Fig. 1. The hardware includes an IBM-compatible personal computer (Pentium 1 GHz), which calculated the control input and controlled the proportional valve (FESTO, MPYE-5-1/8HF-710 B) through D/A board (Advantech, PCI 1720), and two pneumatic artificial muscles (FESTO, MAS-10-N-220-AA-MCFK). The structure of the artificial muscle is shown in Fig. 2. The pressure difference between the antagonistic artificial muscles produced a torque and rotated the joint as a result. (Fig. 4) A joint angle θ was detected by a rotary encoder (METRONIX, H40-8-3600ZO) and the air pressure into each chamber was also measured by the pressure sensors (FESTO, SDE-10-10) and fed back to the computer through a 24-bit digital counter board (Advantech, PCL 833) and A/D board (Advantech, PCI 1711), respectively. The external inertia load could be changed from 20kg·cm^2 to 620 kg·cm^2, which is a 3,000% change with respect to the minimum inertia load condition. The experiments were conducted under the pressure of 4 [bar] and

all control software was coded in C program language. A photograph of the experimental apparatus is shown in Fig 3.

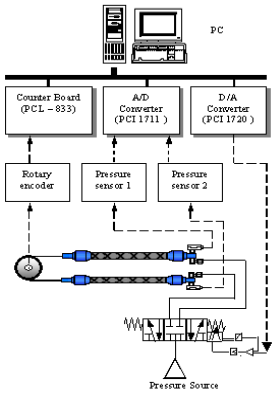


Fig. 1 Schematic diagram of pneumatic artificial muscle manipulator.

Fig. 2 Structure of the pneumatic artificial muscle.

2.2 Characteristics of The PAM Manipulator

The PAM is a tube clothed with a sleeve made of twisted fiber-cords, and fixed at both ends by fixtures. The muscle is expanded to the radial direction and constricted to the vertical direction by raising the inner pressure of the muscle through a power-conversion mechanism of the fiber-cords. The PAM has the property of a spring, and can change its own compliance by inner pressure. A few sliding parts and a little friction are favorable for a delicate power control. But the PAM has the characteristics of hysteresis, non-linearity and low damping. Particularly, the system dynamics of the PAM changes drastically by the compressibility of air in cases of changing external loads. In our experiments, the external load changed about 3,000[%] with respect to the minimum inertia condition.

When using the PAM for the control of a manipulator, it is necessary to understand the characteristics of hysteresis, nonlinearity and so on. Therefore, the following experiments were performed to investigate the characteristics of the PAM. Figs. 5~6 demonstrate the hysteresis characteristics for the joint. This hysteresis can be shown by rotating a joint along a pressure trajectory from $P_1=P_{max}$, $P_2=0$ to $P_1=0$, $P_2=P_{max}$ and back again by incrementing and decrementing the pressures by controlling the proportional valve. The hysteresis of the PAM is shown in Fig. 6. The width of the gap between the two curves depended on how fast the pressures were changed; the slower the change in the pressures, the narrower the gap. The trajectory, control input to the proportional valve, velocity, and pressure of each chamber of the PAM are depicted in Fig. 5. The velocity is numerically computed from the position. Near the extreme values, the joint velocity decreased since the increase in exerted force for a constant change in pressure was less.

3. INTELLIGENT SWITCHING CONTROL ALGORITHM

3.1 The overall control system

The control performance of a PAM manipulator depends on the pressure responses of the pneumatic artificial muscle. Therefore, the pressure should be controlled as rapidly and accurately as possible. To handle these problems, several research works have been concerned with such factors as pressure control systems with a compensation of pressure delay using a 7 PCM digital control[16], valve systems for the flow rate using piezo-electric valves[17], pressure control systems using servo valves[18] and the servo systems using electro-magnetic valves[19]. Though these pressure control systems are satisfactory in their response, the cost of the flow control system is very expensive and some of these systems require another sub-controller to satisfy set-point controls.

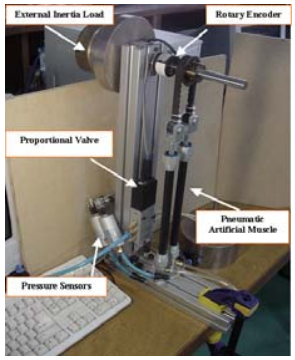


Fig. 3 Photograph of the experimental apparatus.

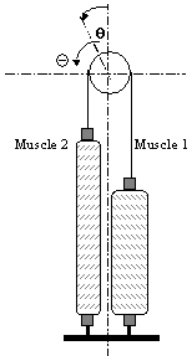


Fig. 4 Working principle of pneumatic artificial muscle manipulator.

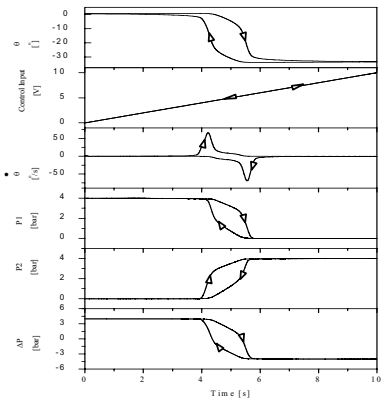


Fig. 5 Characteristics of pneumatic artificial muscle manipulator.

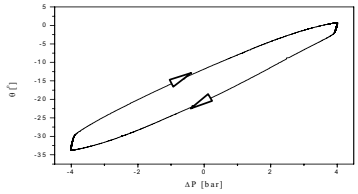


Fig. 6 Hysteresis of pneumatic artificial muscle manipulator.

On the contrary, Hildebrandt and his team used an electronic proportional directional 5/3-way control valve in order to control pressure and flow rates[20]. With this valve, the stroke of the valve-spool is controlled proportionally to a specified set point. In addition, a fuzzy PID-type tracking controller with learning ability has good results with accurate positioning of the pneumatic muscle after a few seconds of operation[4]. However, some limitations still exist because the necessary time for learning is quite long and the controller output functions properly after about 30~45 seconds according to the input signal. And the problem of changes of external inertia load is not mentioned in the above system. Thus the goal of this paper is to develop a fast, accurate, inexpensive and intelligent pneumatic servo system for the PAM without regard to changes of external inertia loads.

To cope with the 30 times change of external inertia load with respect to the base inertia load, the control performance cannot be guaranteed by using a fixed gain controller and the external inertia load condition must be recognized using the dynamic information of the PAM manipulator in an on-line manner. Here we propose the learning vector quantization neural network (which is abbreviated as LVQNN) as a supervisor, which classifies 3 typical external inertia loads (20, 290, 570 kg·cm^2). The structure of the newly-proposed switching control algorithm is shown in Fig. 7.

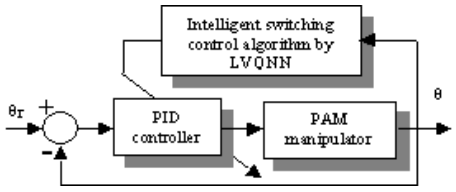


Fig. 7 Structure of the newly proposed control algorithm.

To control this PAM manipulator, a conventional PID control algorithm was applied in this paper as the basic controller. The controller output can be expressed in the time domain as:

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(t)dt + K_p T_d \frac{de(t)}{dt} \quad (1)$$

Taking the Laplace transform of Eq. (1) yields:

$$U(s) = K_p E(s) + \frac{K_p}{T_i s} E(s) + K_p T_d E(s) \quad (2)$$

and the resulting PID controller transfer function of:

$$\frac{U(s)}{E(s)} = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \quad (3)$$

A typical real-time implementation at sampling sequence k can be expressed as:

$$u(k) = K_p e(k) + u(k-1) + \frac{K_p T}{T_i} e(k) + K_p T_d \frac{e(k) - e(k-1)}{T} \quad (4)$$

where u(k), e(k) are the control input to the control valve and the error between the desired set point and the output of joint, respectively.

3.2 Recognition the external load condition by using the LVQNN

The external load must be recognized for an intelligent control of the PAM manipulator. Here the LVQNN is newly-proposed as a supervisor of the switching controller.

3.2.1 Structure of the neural classifier

According to the learning process, neural networks are divided into two kinds: supervised and unsupervised. The difference between them lies in how the networks are trained to recognize and categorize objects. The LVQNN is a supervised learning algorithm, which was developed by Kohonen and is based on the self-organizing map (SOM) or Kohonen feature map. The LVQNN methods are simple and effective adaptive learning techniques. They rely on the nearest neighbor classification model and are strongly related to condensing methods, where only a reduced number of prototypes are kept from a whole set of samples. This condensed set of prototypes is then used to classify unknown samples using the nearest neighbor rule. The LVQNN has a competitive and linear layer in the first and second layer, respectively. The competitive layer learns to classify the input vectors and the linear layer transforms the competitive layer's classes into the target classes defined by the user. Figure 8 shows the architecture of the LVQNN, where P, y, W1, W2, R, S1, S2, and T denote input vector, output vector, weight of the competitive layer, weight of the linear layer, number of neurons of the input layer, competitive layer, linear and target layer, respectively. In the learning process, the weights of the LVQNN are updated by the following Kohonen learning rule if the input vector belongs to the same category.

$$\Delta W_1(i,j) = \lambda a_1(i)(p(j) - W_1(i,j)) \quad (5)$$

If the input vector belongs to a different category, the weights of the LVQNN are updated by the following rule

$$\Delta W_1(i,j) = -\lambda a_1(i)(p(j) - W_1(i,j)) \quad (6)$$

where λ is the learning ratio and a_1(i) is the output of the competitive layer.

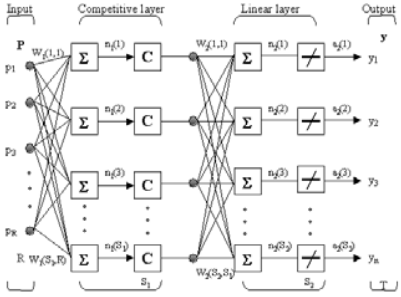


Fig. 8 Structure of the LVQNN.

3.2.2 Data generation for the training of the LVQNN

In the design of the LVQNN, it was very important to identify what input to select and how many sequences of data to use. Generally the training result was better according to the increase of the number of input vectors, but it took more calculation time and the starting time of the recognition of inertia load was later. In our experiment, we prepared 2 cases of input vectors as shown in Fig. 9(a) and (b). In Case 1, the input vectors into the LVQNN were set for the control input, angular velocity, and pressures of each chamber. Meanwhile, the input vectors into the LVQNN are set for the control input, angular velocity and pressure difference in Case 2. In each case, the output of the LVQNN was an integer value between 1 and 3, where 3 cases could be classified according to the external inertial load, i.e. for example, class 1 meant that the range of the external inertia load was approximately between 20 and 45 kg·cm^2 as shown in Table 1. To obtain the learning data for the LVQNN, a series of experiments were conducted under 9 different external inertial load conditions, as shown in Table 1. The experimental results of the generation of training data are shown in Fig. 10(a)~(e), which correspond to the control input to the proportional valve, angular velocity of joint, pressures in the chamber 1 and chamber 2, and pressure difference, respectively. In each figure, the number * and # in Inertia*# means the class and the inertia change in that class, respectively. In the experiments of the generation of training data, the reference angle is set to 15 [°] and the PID controller with fixed gain was used.

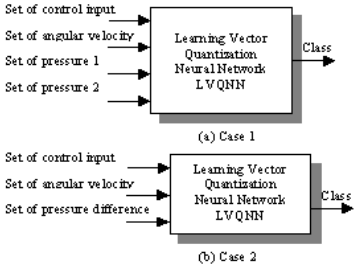


Fig. 9 Learning data for LVQNN.

Table 1 Classification of external inertia load

Initial *# [kg·cm^2]	Class 1	Class 2	Class 3
# \ *	1	2	3
1	20	28	56
2	33	31	59
3	45	34	62

3.2.3 Training process of the LVQNN

The learning vector quantization neural network (LVQNN) is a method for training competitive layers in a supervised manner. A competitive layer will automatically learn to classify input vectors. However, the classes that the competitive layer finds are dependent only on the distance between input vectors. If two input vectors are very similar, the competitive layer probably will put them into the same class. Thus, the LVQNN can classify any set of input vectors, not just linearly-separable sets of input vectors. The only requirement is that the competitive layer must have enough neurons, and each class must be assigned enough competitive neurons.

A total of 9 experimental cases were carried out to prepare for the generation of training data for the LVQNN. In the training stage of LVQNN, the number of input vectors were adjusted from 5 to 21 with 5 steps and the number of neurons in the competitive layer were adjusted from 8 to 26 with 10 steps in each case, as shown in Table 2, in order to obtain the optimal weight of the LVQNN. To investigate the classification ability of the LVQNN, the same input vectors, which were used in the learning stage, were re-entered into the LVQNN and the learning success rate was calculated. Here, the learning success rate defines the percentage of success of the LVQNN learning, where success means that the output of the LVQNN was equal to the target class with respect to the same input vectors.

As the LVQNN classified input vectors into target classes by using a competitive layer and the classes that the competitive layer found were dependent only on the distance between input vectors, a high learning success rate was realized when the input vectors were distributed widely.

From Fig. 10, both pressures of each chamber of the muscle were used as learning data in Case 1 and the difference pressure in Case 2. From Fig. 10, It was understood that the input vectors in Case 1 were distributed more widely than those in Case 2. Therefore, it was concluded that the training result of Case 1 was better than the training result of Case 2.

From Fig. 11 and Table 2, it was also understood that the optimal number of input vectors and neurons of the competitive layer were 13 and 18, respectively and the maximum training success rate was 87[%], which was enough for a recognition of the external inertia load condition.

3.3 Proposition of the smooth switching algorithm

If the external inertial load condition was different from the previous training condition, the output of the LVQNN may have belonged to the mixed classes with different ratios in each case (i.e. if the external inertia load was between the inertia of Class 1 and Class 2 it may have belonged to 1 or 2 class). Therefore the following switching algorithm was proposed to apply to the abrupt change of class recognition result. The switching algorithm is described by the following equation:

$$class(k) = \chi.class(k-1) + (1-\chi).class(k) \quad (7)$$

where k is the discrete sequence, χ is the forgetting factor and class (k) is the output of the LVQNN at the k time sequence. The optimal parameters of PID controller with respect to each inertia condition were obtained by trial-and-error through experiments, which are shown in Table 3. These PID parameters seemed too small because the sampling time was not included in the derivation of the PID controller and they had the magnitude of the sampling time. From Table 3, it was understood that the proportional, integral and derivative control gains were decreasing in accordance with an increase in the external inertia load.

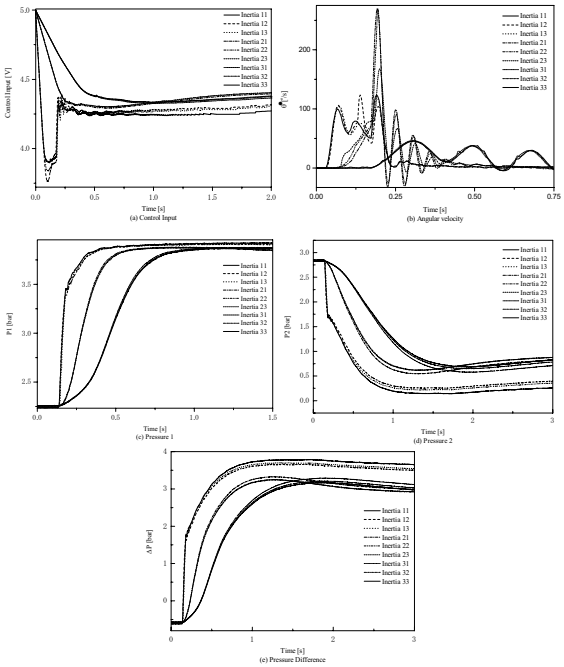
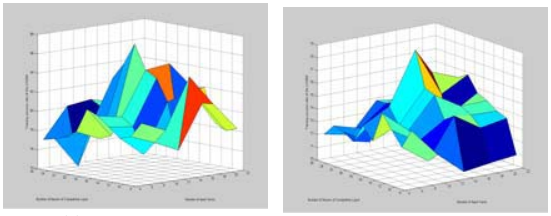


Fig. 10 Experimental results for learning data generation.



(a) Case 1 (b) Case 2

Fig. 11 Training success rate of the LVQNN.

Table 2. Training success rate of the LVQNN (%)
(NIV: Number of Input Vector)
(NCL: Number of Neuron of Competitive Layer)

NIV/ NCL	5	9	13	17	21
8	80.48	79.38	84.59	81.38	78.47
10	78.86	77.88	76.73	81.28	78.02
12	78.51	76.43	81.58	82.32	78.64
14	79.48	79.16	83.83	76.47	82.13
16	80.42	79.97	84.50	84.78	81.83
18	80.92	77.80	87.02	77.70	79.56
20	78.37	77.28	84.77	79.71	78.73
22	77.60	78.06	83.06	81.38	77.92
24	77.24	74.00	79.80	82.33	77.96
26	76.96	80.09	80.42	79.70	76.86

(a) Case 1

NIV/ NCL	5	9	13	17	21
8	74.54	72.30	70.57	70.88	71.12
10	73.01	71.52	74.67	73.37	73.73
12	74.06	72.81	74.12	74.10	74.02
14	75.13	73.44	75.78	74.53	74.57
16	72.71	73.56	76.07	70.93	74.91
18	73.64	74.77	78.80	74.96	76.09
20	72.14	73.89	75.05	76.42	75.61
22	72.66	71.43	74.21	73.56	75.51
24	72.86	72.26	73.84	74.81	75.37
26	71.93	71.49	73.79	73.47	75.31

(b) Case 2

Table 3. Optimal parameters of PID controller.

Class No.	K_p	K_i	K_d
1	$5x10^{-3}$	$0.1x10^{-3}$	$0.65x10^{-3}$
2	$0.25x10^{-3}$	$0.01x10^{-3}$	$0.06x10^{-3}$
3	$0.1x10^{-3}$	$0.001x10^{-3}$	$0.036x10^{-3}$

4. EXPERIMENT RESULTS

Figure 12 shows the experimental results of position control with different external inertia loads (20, 280 and 560 [kg·cm^2]), where the control gains were fixed and the same as that of the minimum external inertia load condition. From Fig. 12, it was understood that the system response became more oscillatory according to the increase of the external inertial load and it was requested that the control parameters be adjusted according to the change of the external inertia load.

Next, experiments were carried out to verify the effectiveness of the proposed switching algorithm by the LVQNN. The experimental results are shown in Fig. 13, 14 and 15, which correspond to the minimum external inertial load condition (Class 1), medium inertia load condition (Class 2), and maximum inertia load condition (Class 3), respectively. In these figures, we show angle of joint, control input, angular velocity, pressures in chamber 1 and 2, output of the LVQNN and filtered output of the LVQNN, respectively. As the number of the input vector was 13, which included 4 control inputs, 3 angular velocities, and 3 pressures of each chamber, the output of the LVQNN started to function after 3 sampling time (i.e. at least 4 control inputs must be prepared for the calculation of LVQNN). From these experimental results, particularly in the filtered output of the LVQNN, it was verified that the external inertial load was almost exactly recognized to the correct class and an accurate position control was realized with a steady error of 0.05 [o].

The experimental results with an external inertia load of 420[kg.cm2] are shown in Fig.16. This load condition corresponded to Class 2 and Class 3. The class number calculated from the output of the LVQNN was between 2 and 3, which proved that the external inertia load was between 280 and 560[kg·cm2]. In Fig. 17, experiments were conducted to compare the system response with respect to 2 different weight conditions (280, 560[kg·cm2]) with and without the proposed switching algorithm by the LVQNN. From the experimental results, it was found that the system response became oscillatory according to an increase in the external inertial load. On the contrary, the system response was almost the same and the steady state error was within 0.1 [°] in any case by using the proposed switching algorithm with the LVQNN. It was also verified that the proposed method was very effective in the accurate position control of the PAM manipulator.

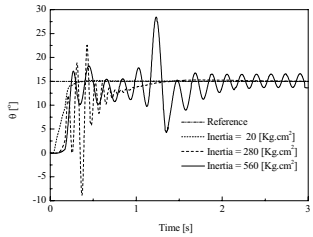


Fig. 12 Experiment results of the pneumatic artificial muscle manipulator without switching control in the case of three different external inertia load.
(PID parameters: K_p=5x10^{-3}, K_i=0.1x10^{-3}, K_d=0.65x10^{-3})

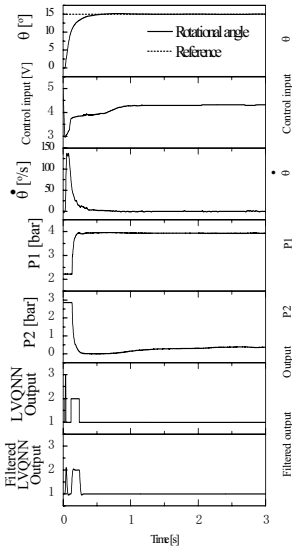


Fig. 13 Experimental results with no external inertia load (class 1)

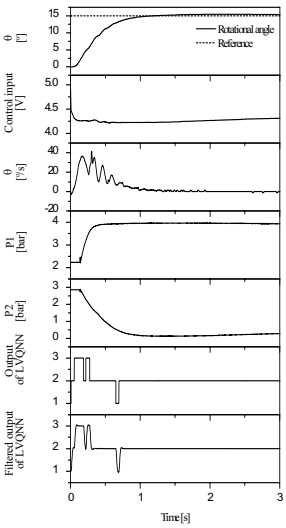


Fig. 14. Experimental results when external inertia load is 280 [kg.cm^2] (class 2)

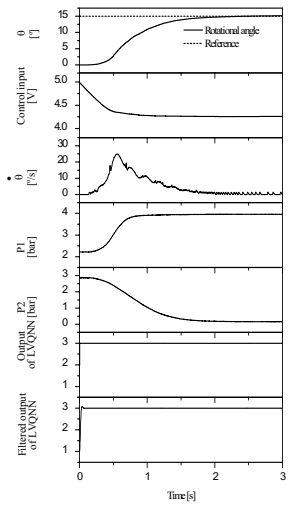


Fig. 15. Experimental results when external inertia load is 560 [kg.cm^2] (class 3)

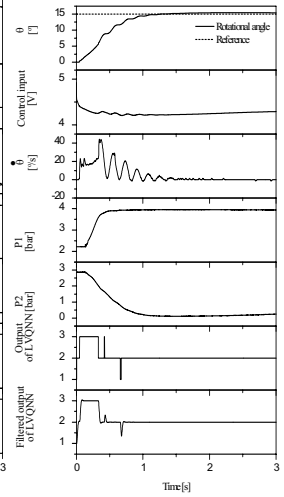


Fig. 16 Experimental results when external inertia load is 420 [kg·cm^2] (between class 2 and class 3)

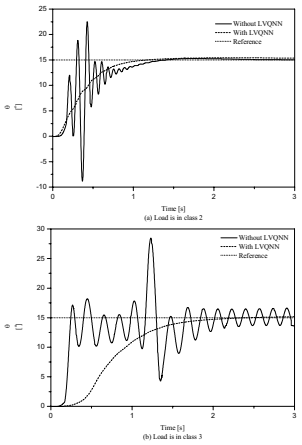


Fig. 17 Comparison of experimental results between with and without LVQNN

5. CONCLUSION

In this study, a fast, accurate, inexpensive and external inertial load independent pneumatic artificial muscle manipulator that may be applied to a variety of practical positioning applications was developed. The position control was successfully implemented using a proportional valve instead of an expensive servo valve. And the steady state error was reduced within 0.1 [º].

The second contribution of this paper is to propose a learning vector quantization neural network (LVQNN) as a supervisor of the switching controller in the pneumatic artificial muscle manipulator, where the LVQNN functions to recognize the condition of the weight of an external inertial load and to select suitable gains for each load condition.

From the experiments of the position control of an pneumatic artificial muscle manipulator, it was verified that the smooth switching algorithm is very effective to overcome the deterioration of control performances of transient responses even if the external inertia load changed for 3,000[%].

ACKNOWLEDGEMENT

This work was partly supported by the Korea Science and Engineering Foundation (KOSEF) through the Research Center for Machine Parts and Materials Processing (ReMM) at the University of Ulsan

REFERENCES

[1] T. Hesselroth, K. Sarkar, P. Patrick van der Smagt, and K. Schulten, "Neural network control of a pneumatic robot arm," *IEEE Trans Syst., Man., Cybernetics*, Vol. 24, No 1, pp. 28-38, 1994.

[2] P. Patrick van der Smagt, F. Groen, and K. Schulten, "Analysis and control of a Rubbertuator arm," *Biol. Cybernetics.*, Vol. 75, pp. 433-440, 1996.

[3] N. Tsagarakis, D.G. Caldwell, G.A. Medrano-Cerda, "A 7 DOF pneumatic muscle actuator (pMA) powered exoskeleton," *IEEE International Workshop on Robot and Human Interaction*, Pisa, Italy, pp. 327-333, 1999.

[4] S.W. Chan, J.H. Lilly,D.W. Repperger, J.E. Berlin, "Fuzzy PD+I learning control for a pneumatic muscle," *IEEE Int., Conf., Fuzzy Systems*, Vol. 1, pp. 278-283, 2003.

[5] K. Balasubramanian, K.S. Rattan, "Fuzzy logic control of a pneumatic muscle system using a linearizing control scheme," *Int. Conf., North American Fuzzy Information Processing Society*, pp 432-436, 2003.

[6] D. Cai, H. Yamaura, "A robust controller for manipulator driven by artificial muscle actuator," *in Proc., IEEE Int., Conf., Control Applications*, pp. 540-545, 1996.

[7] M. Guihard, P. Gorce, "Dynamic control of an artificial muscle arm," *in Proc., IEEE Int., Conf., Systems, Man and Cybernetics*, France, Vol. 4, pp. 813-818, 1999.

[8] P. Carbonell, Z.P. Jiang, D.W. Repperger, "Nonlinear control of a pneumatic muscle actuator: backstepping vs. sliding-mode," *in Proc., IEEE Int., Conf., Control Applications* , Mexico City, Mexico, pp. 167-172, 2001.

[9] T. Kimura, S. Hara, T. Fujita, T. Kagawa, "Control for pneumatic actuator systems using feedback linearization with disturbance rejection," *in Proc., Conf., American Control*, Vol. 1, pp. 825-829, 1995.

[10] K. Balasubramanian, K.S. Rattan, "Feedforward control of a non-linear pneumatic muscle system using fuzzy logic," *in IEEE Int., Conf., Fuzzy Systems*, Vol. 1, pp. 272-277, 2003.

[11] T. Noritsugu, Y. Tsuji, K. Ito, "Improvement of control performance of pneumatic rubber artificial muscle manipulator by using electrorheological fluid damper," *in Proc., IEEE Int., Conf., Systems, Man and Cybernetics*, Washington, Vol. 4, pp 788-793, 1999.

[12] M. Hamerlain, "An anthropomorphic robot arm driven by artificial muscles using a variable structure control," *in IEEE/RSJ Int, Conf. Intelligent Robots and Systems*, Vol. 1, pp. 550-555, 1995.

[13] K. Osuka, T. Kimura, T. Ono, "H∞ control of a certain nonlinear actuator," *in Proc., IEEE Int., Conf., Decision and Control*, Honolulu, Hawai, Vol. 1, pp. 370-371, 1990.

[14] Ahn, K., K., Lee, B., R., Yang, S., Y., " Design and experimental evaluation of a robust force controller for a 6-link electro-hydraulic manipulator via H infinity control theory," *in KSME, Int., Jour.*, Vol. 17, No. 7, pp. 999~1010, 2003.

[15] Ahn, K., K., Pyo, S., M., Yang, S., Y., Lee, B., R., "Intelligent control of pneumatic actuator using LVQNN," *in Proc., KORUS 2003, Symposium on Science and Technology*, pp. 260~266, 2003.

[16] T. Noritsugu, T. Tanaka, "Application of rubber artificial muscle manipulator as a rehabilitation robot," *in IEEE/ASME Trans., Mechatronics*, Vol. 2, pp. 259-267, 1997.

[17] G.A. Medrano-Cerda, C. J. Bowler, and D. G. Caldwell, "Adaptive position control of antagonistic pneumatic muscle actuators," *in Proc. EEE/RSJ Int., Conf., Intelligent Robots and Systems, Human Robot Interaction and Cooperative Robots*, Pittsburgh, PA, pp. 378-383, 1995.

[18] B. Tondu, V. Boitier, P. Lopez, "Naturally compliant robot-arms actuated by McKibben artificial muscles," *in IEEE Int., Conf., Systems, Man and Cybernetics*, Vol. 3, pp. 2635-2640, 1994.

[19] Y.K. Lee, I. Shimoyama, "A skeletal framework artificial hand actuated by pneumatic artificial muscles," *in IEEE Int., Conf., Robotics and Automation*, Vol. 2, pp. 926-931, 1999.

[20] A. Hildebrandt, O. Sawodny, R. Neumann, A. Hartmann, "A Flatness Based Design for Tracking Control of Pneumatic Muscle Actuators," *in Int., Conf., Control Automation Robotics and Vision*, Singapore, pp. 1156-1161, 2002.