Adaptive Single-Input Recurrent WCMAC-Based Supervisory Control for De-icing Robot Manipulator

***Abstract*—This paper presents a novel adaptive single-input recurrent wavelet differentiable cerebellar model articulation controller (S-RWCMAC)-based supervisory control system for an *m*-link robot manipulator to achieve the precision trajectory tracking. This adaptive S-RWCMAC-based supervisory control system consists of a main adaptive S-RWCMAC, a supervisory controller and an adaptive robust controller. The S-RWCMAC incorporates the advantages of the wavelet decomposition property with a CMAC fast learning ability, dynamic response and input space dimension of RWCMAC can be simplified; and it is used to control the plant. The supervisory controller is appended to the adaptive S-RWCMAC to force the system states within a predefined constraint set and the adaptive robust controller is developed to dispel the effect of approximate error. In this scheme, if the adaptive S-RWCMAC can not maintain the system states within the constraint set. Then, the supervisory controller will work to pull the states back to the constraint set and otherwise is idle. The online tuning laws of S-RWCMAC and the robust controller parameters are derived in gradient-descent learning method and Lyapunov function, so that the stability of the system can be guaranteed. Finally, the simulation and experimental results of novel three-link De-icing robot manipulator are provided to verify the effectiveness of the proposed control methodology.**

Keywords*—Wavelet, recurrent Wavelet Cerebellar model articulation controller (RWCMAC), De-icing robot manipulator, supervisory control.

I. INTRODUCTION

Fuzzy logic control (FLCs) has found extensive applications for plants that are complex and ill-defined which is suitable for simple second order plants. However, in case of complex higher order plants, all process states are required as fuzzy input variables to implement state feedback FLCs. All the state variables must be used to represent contents of the rule antecedent. So, it requires a huge number of control rules and much effort to create. To address these issues, single-input Fuzzy Logic controllers (S-FLC) was proposed for the identification and control of complex dynamical systems [1–3]. As a result, the number of fuzzy rules is greatly reduced compared to the case of the conventional FLCs, but its control performance is almost the same as conventional FLCs.

Cerebellar model articulation controller (CMAC) was proposed by Albus in 1975 [4] for the identification and control of complex dynamical systems, due to its advantage of fast learning property, good generalization capability and ease of implementation by hardware [5–7]. The conventional CMACs, regarded as non-fully connected perceptron-like associative memory network with overlapping receptive fields which used constant binary or triangular functions. The disadvantage is that their derivative information is not preserved. For acquiring the derivative information of input and output variables, Chiang and Lin [8] developed a CMAC network with a differentiable Gaussian receptive-field basis function and provided the convergence analysis for this network. The advantages of using CMAC over neural network in many applications were well documented [9]–[11].

The most control systems are developed in the literatures based on the CMAC which is used to approach the nonlinear mapping [12–13], in which the input dimension of CMAC includes the all state variables of the system. As a result, many adaptive approaches are also rejected as being overly computationally intensive because of the real-time parameter identification and required control design. To deal with these problems, another control schemes are also proposed in [14–16] by combination with conventional control techniques (sliding model control, etc.) so that the dimension of the input space is reduced. Recently, there are some researches [17–19] which have reported very good results based on the CMAC control system. In [17–18], the proposed single-input CMAC controllers are not only solely used to control the plant, so the input space dimension can be simplified and no conventional controller. However, the disadvantaged of proposed CMAC control system adopts two learning stages, an off-line learning stage and an on-line learning stage. In [19], the later proposed controller is developed the same in [18], but in this control scheme was not only overcome disadvantages in [18] but also the stability of the control system can be guarantied. However, the major drawback of the above single-input CMACs is they belong to static networks.

Recently, many applications have also been implemented quite successfully based on wavelet neural networks (WNNs) which combine the learning ability of network and capability of wavelet decomposition property [20–23]. Different from conventional NNs, the membership functions of WNN is wavelet functions which are spatially localized, so, the WNNs are capable of learning more efficiently than conventional NNs for control and system identification as has been demonstrated in [20, 22]. As a result, WNNs has been considerable interest in the applications to deal with uncertainties and nonlinearity control system as is shown in [22–23].

In this paper, by combining the fast learning property of CMAC, capability of the wavelet decomposition property, by including a delayed self-recurrent unit in the association memory space, , and based on [1–2], the system tracking error $E \in R^n$ is transformed into a single variable, termed the signed distance $d_{si} \in R^m$, we propose a novel adaptive single-input recurrent wavelet CMAC (S-RWCMAC)-based supervisory control system which is presents a dynamic S-RWCMAC with single-input for three-link De-icing robot manipulator to achieve the precision trajectory tracking. This control system consists of an adaptive S-RWCMAC, a supervisory controller and an adaptive robust controller. The S-RWCMAC is the main controller which is used to mimic the ideal controller through learning and the adaptive robust controller is developed to dispel the effect of the approximation error. The online tuning laws of S-RWCMAC parameters are derived in gradient-descent learning method which can be caused the instability controlled system, especially in the transient period. So, the supervisory controller is appended to the adaptive S-RWCMAC to force the system states within a predefined constraint set, if the adaptive S- RWCMAC can not maintain the system states within the constraint set. Then, the supervisory controller will work to pull the states back to the

constraint set and otherwise is not. This is the first contribution of this paper. The second contribution proposes novel architecture and mathematical model of De-icing robot manipulator which can be effectiveness application in practical.

This paper is organized as follows: System description is described in Section II. Section III presents S-RWCMAC-based supervisory control system. Numerical simulation and experimental results of a three-link De-icing robot manipulator under the possible occurrence of uncertainties are provided to demonstrate the tracking control performance of the proposed S-RWCMAC system in Section IV. Finally, conclusions are drawn in Section V.

II. SYSTEM DESCRIPTION

In general, the dynamic of an *m*-link robot manipulator may be expressed in the Lagrange following form:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{1}$$

Where $q, \dot{q}, \ddot{q} \in R^m$ are the joint position, velocity and acceleration vectors, respectively, $M(q) \in R^{m\times m}$ denotes the inertia matrix, $C(q,\dot{q}) \in R^{m\times m}$ expresses the matrix of centripetal and Coriolis forces, $G(q) \in R^{m\times 1}$ is the gravity vector, $\tau \in R^{m\times 1}$ is the torque vectors exerting on joints. In this paper, a new three-link De-icing robot manipulator, as shown in Fig.1 (b), is utilized to verify dynamic properties are given in section IV. By rewriting (1), the dynamic equation of robot manipulator can be obtained as follows:

$$\begin{aligned}\ddot{q} &= -M^{-1}(q)\big(C(q,\dot{q})\dot{q} + G(q)\big) + M^{-1}(q)\tau \\ &= f(x) + g(x)\tau\end{aligned} \tag{2}$$

$$f(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{pmatrix} = -M^{-1}(q)\big(C(q,\dot{q})\dot{q} + G(q)\big) \in R^{m\times 1},$$

$$g(x) = \begin{pmatrix} g_{11}(x) & \cdots & g_{1m}(x) \\ \vdots & \ddots & \vdots \\ g_{m1}(x) & \cdots & g_{mm}(x) \end{pmatrix} = M^{-1}(q) \in R^{m\times m}.$$

Where $f(x)$, $g(x)$ are nonlinear dynamic functions which are difficult to determine exactly or can not even obtain. So, we can not establish model-based control system. In order to with this problem, here we assume that actual value $f(x)$ and $g(x)$ can be separated as nominal part denoted by $F_0(x)$, $G_0(x)$, in which $G_0(x)$ is assumed to be positive, differentiable and $G_0^{-1}(x)$ exists for all q. $L(x,t)$ is represented as the unknown lumped uncertainly and $x = [q^T, \dot{q}^T]^T$ is vector which represents the joint position and velocity. Finally, the system (2) can be rewritten as follows:

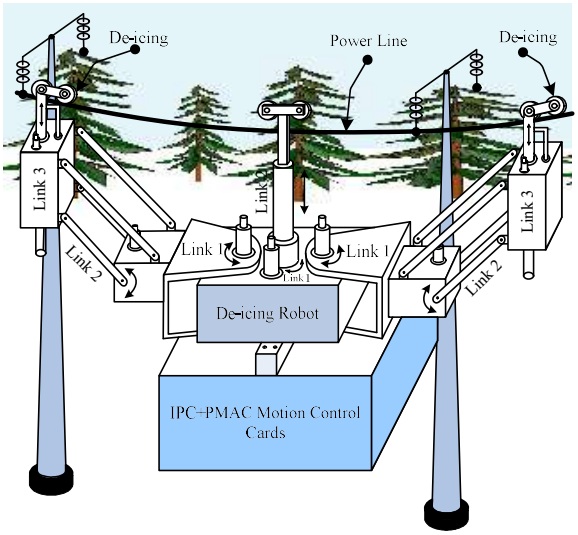
$$\ddot{q}(t) = F_0(x) + G_0(x)\tau + L(x,t), \tag{3}$$

The control problem is to force $q(t) \in R^n$, to track a given bounded reference input signal $q_d(t) \in R^n$. Let $e(t) \in R^n$ be the tracking error as follows:

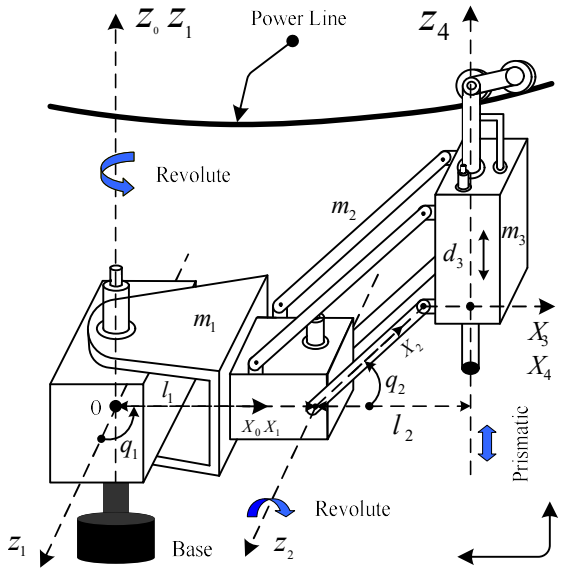
$$e = q_d(t) - q(t) \tag{4}$$

and the system tracking error vector is defined as

$$E \triangleq [e^T \quad \dot{e}^T \quad \cdots \quad e^{n-1^T}]^T \in R^{nm} \tag{5}$$



(a)



(b)

Fig.1. Architecture of three-link De-icing robot manipulator.

Where n is order of nonlinear system. If the nominal parts $F_0(x)$, $G_0(x)$ and the uncertainly $L(x,t)$ are exactly known, then an ideal controller can be designed as follows:

$$\tau^*(t) = \frac{1}{G_0(x)}\big[\ddot{q}_d(t) - F_0(x) - L(x,t) + KE\big] \tag{6}$$

By substituting the ideal controller (6) into (3), the error dynamic equation is given as follows:

$$\ddot{e}(t) + K^T E = 0 \tag{7}$$

It is obvious that errors will be asymptotically tend to zero if the gain matrices of $K = [k_1, \cdots, k_{nm}]^T \in R^{nm}$ is determined so that the roots of the characteristic polynomial $P(\lambda) = I\ddot{\lambda} + k_1\dot{\lambda} + k_2\lambda + k_3$ lie strictly in the open left haft of complex plane. However, the ideal controller in (6) can not determine, because of $L(x,t)$ is exactly unknown for practical applications. So, in order to this problem, a proposed

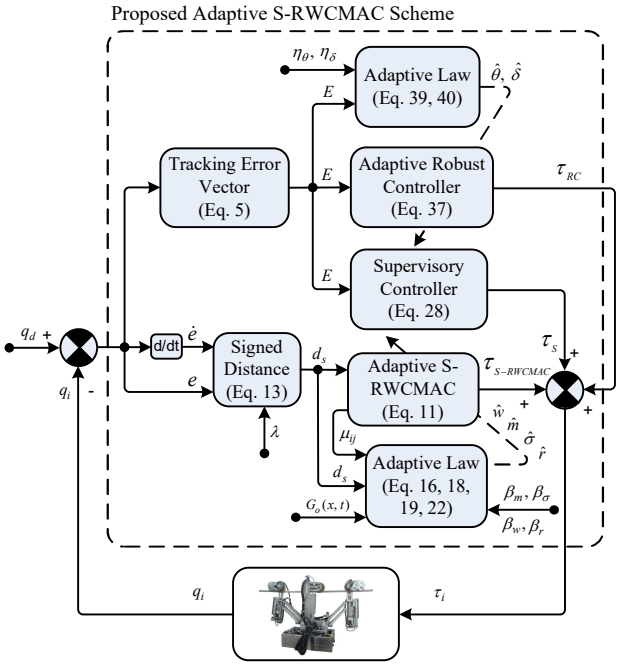


Fig.2. Block diagram of proposed adaptive S-RWCMAC-based supervisory control system.

adaptive S-RWCMAC-based supervisory control system is shown in Fig. 2 which is described in the following sections.

III. ADAPTIVE S-RWCMAC-BASED SUPPERVISORY CONTROL SYSTEM

The architecture of the adaptive S-RWCMAC-based supervisory control system is shown in Fig. 2 which consists of adaptive S-RWCMAC, the supervisory controller and the adaptive robust controller with the following form:

$$\tau = \tau_{S-RWCMAC} + \tau_S + \tau_{RC} \tag{8}$$

Where $\tau_{S-RWCMAC}$ is the main controller based on the S-RWCMAC which is used to mimic the ideal controller in (6) and τ_S is the supervisory controller, which can be used to stabilize the states of the controlled system within a predefined constrain set and the adaptive robust controller τ_{RC} is utilized to compensate for the approximation error between the ideal controller and $\tau_{S-RWCMAC}$.

A. Brief of the S-RWCMAC

An S-RWCMAC is proposed based on [19] and is depicted in Fig. 3. This S-RWCMAC is composed of an input space, an association memory space, a weight memory space, an output. The signal propagation and the basic function in each space are introduced as follows.

1. Input space D_s ; assume that each input state variable d_{rsi} can be quantized into N_{si} discrete states and that the information of a quantized state is distributive stored in N_e $(N_e \geq 2)$ memory elements. Therefore, there exist $N_{si}+1$ individual points on the d_{rsi} - axis. Fig. 4 depicts the schematic diagram of one dimensional S-RWCMAC operations with $N_e = 3$ and $N_{si} = 7$ for $i = 1, 2 \cdots m$ where the equidistant quantization scheme is used to partition the input space $[-1, +1]$. This simple example shows that the input space is quantized into three discrete regions, called blocks. For instance, there are three blocks, namely, A, B, and C, in the first layer. By shifting each block a small interval, different blocks can be obtained. For example, D, E and F in the second layer are possible shifted regions. With this kind of decomposition, one can imagine that there are $N_e = 3$ layers of blocks. Each state is covered by $N_e = 3$ different blocks. The S-RWCMAC associates each block to a physical memory element. Information for a quantized state is distributive stored in memory elements associated with blocks that cover this state. Note that a state will share some memory elements with its neighboring states, but any two states will not correspond to the same set of blocks. Moreover, the entire memory size that is equal to the number of blocks and denoted by N_h is determined by $N_h = N_{si} + N_e - 1$. See the block division shown in Fig. 4 for instance; the total number of memory elements is 9. In this space, each block performs a receptive-field basis function, which can be defined mother wavelet. The first derivative of basic Gaussian function for each block is given here as a mother wavelet which can be represented as follows:

$$\mu_{ij}(d_{rsi}, m_{ij}, \sigma_{ij}, k) = -\frac{(d_{rsi}(k) - m_{ij})}{\sigma_{ij}} \exp\left[-\frac{\left(\frac{(d_{rsi}(k) - m_{ij})}{\sigma_{ij}} \right)^2}{2} \right]$$

$$= -F_{ij} \exp\left(-\frac{F_{ij}^2}{2} \right)$$

$$i = 1, 2, \cdots, m, \quad j = 1, 2, \cdots, N_h \tag{9}$$

Where $F_{ij}(d_{rsi}, m_{ij}, \sigma_{ij}, k) = (d_{rsi}(k) - m_{ij})/\sigma_{ij}$, μ_{ij} represents the reception-field basic function for jth block of the ith input, m_{ij} is a translation parameter and σ_{ij} is dilation. In addition, the input of this block can be represented as

$$d_{rsi}(k) = d_{si}(k) + r_{ij}\mu_{ij}(k-1) \tag{10}$$

Where r_{ij} is the recurrent gain, k is denotes the time step, and $\mu_{ij}(k-1)$ denotes the value of $\mu_{ij}(k)$ through a time delay. Clearly, the input of this block contains the memory term $\mu_{ij}(k-1)$, which stores the past information of the network and presents the dynamic mapping.

2. Output space O: The output of S-RWCMAC is the algebraic sum of the firing element with the weight memory, and is expressed as

$$\tau_{S-RWCMAC} = \sum_{j=1}^{N_h} a_{ij} w_{ij} \mu_{ij}, \quad i = 1, 2, \cdots, m. \tag{11}$$

Where w_{ij} denotes the weight of the jth block, $a_{ij} = a_{ij}(d_{rsi})$, $j = 1, 2, \cdots N_h$ is the index indicating whether the jth memory element is addressed by the state involving d_{rsi} . Since each state addressed exactly N_e memory elements, only those addressed a_{ij}'s are one, and the others are zero.

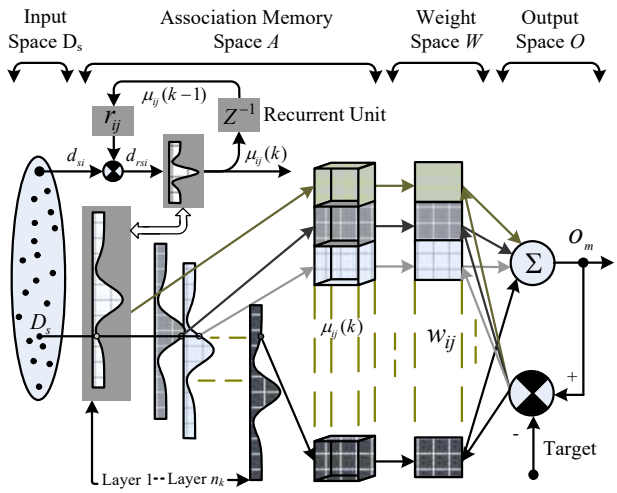


Fig.3. Architecture of a single-input RWCMAC.

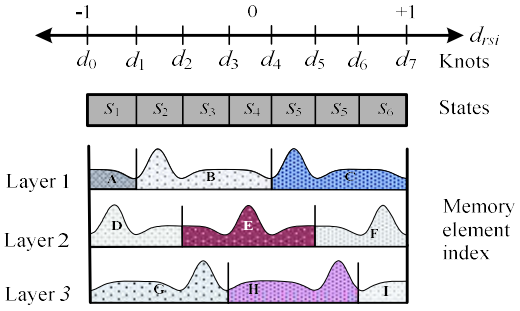


Fig.4. Block division of S-RWCMAC with wavelet function

B. On-line learning algorithm

The central part of the learning algorithm for an S-RWCMAC is how to choose the weight memory w_{ij}, m_{ij} is a translation parameter, σ_{ij} is dilation of the wavelet functions and r_{ij} is a recurrent gain. For achieving effective learning, an on-line learning algorithm, which is derived using the supervised gradient descent method, is introduced so that it can in real-time adjust the parameters of S-RWCMAC.

According to [1], [2], the system tracking error $(e_i, \dot{e}_i) \in R^n$ is transformed into a single variable, termed the signed distance $d_{si} \in R^m$, which is the distance from an actual state $(e_i, \dot{e}_i) \in R^n$ to the switching line as shown in Fig. 5 for a

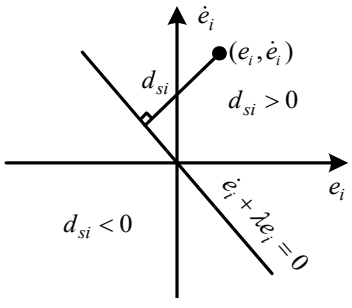


Fig.5. Derivation of a signed distance.

2-D input. The switching line is defined as follows:

$$e_i^{n-1} + \lambda_{n-1} e_i^{n-2} + \cdots + \lambda_2 \dot{e}_i + \lambda_1 e_i = 0 \tag{12}$$

Where λ_{n-1} is a constant. Then, the signed distance between the switching line and operating point $(e_i, \dot{e}_i) \in R^n$ can be expressed by the following equation:

$$\begin{aligned} d_{si} &= \frac{e_{ni}^{n-1} + \lambda_{n-1} e_{(n-1)i}^{n-2} + \cdots + \lambda_2 \dot{e}_{2i} + \lambda_1 e_{1i}}{\sqrt{1 + \lambda_{n-1}^2 + \cdots + \lambda_2^2 + \lambda_1^2}} \\ &= \Gamma(e_{ni}^{n-1} + \lambda_{n-1} e_{(n-1)i}^{n-2} + \cdots + \lambda_2 \dot{e}_{2i} + \lambda_1 e_{1i}) \end{aligned} \tag{13}$$

Where $\Gamma = 1 \Big/ \left(\sqrt{1 + \lambda_{n-1}^2 + \cdots + \lambda_2^2 + \lambda_1^2} \right)$ is a positive constant

By taking the time derivative of (13) for $n = 2$ and using (3), (8). We have

$$\begin{aligned} \dot{d}_s &= \Gamma(\ddot{e} + \lambda_1 \dot{e}) \\ &= \Gamma(-F_0(x,t) - G_0(x,t)(\tau_{S-RWCMAC} + \tau_{RC} + \tau_S) \\ &\quad + \ddot{q}_d(t) - L(x,t) + \lambda_1 \dot{e}) \end{aligned} \tag{14}$$

Define a cost function $V(d_s(t)) = 1/2\, d_s^2(t)$; then, $\dot{V}(d_s(t)) = d_s(t)\dot{d}_s(t)$. The tuning of S-RWCMAC parameter aims to speed up the convergence of $V(d_s(t))$, i.e., minimize $\dot{V}(d_s(t))$ with respect to the tuned parameters. By multiplying both sides of (14) by $d_s(t)$, yields

$$\begin{aligned} d_s(t)\dot{d}_s(t) = -\Gamma(&d_s(t)F_0(x,t) - d_s(t)G_0(x,t)(\tau_{S-RWCMAC} \\ &\tau_{RC} + \tau_S) + d_s(t)(\ddot{q}_d(t) - L(x,t) + \lambda_1 \dot{e})) \end{aligned} \tag{15}$$

With this representation of the S-RWCMAC system, it becomes straightforward to apply the back propagation idea to adjust the parameters. The weight memory w_{ij} and the translations m_{ij} and dilations σ_{ij} of the mother wavelet function are updated by the following:

1. The updating law for the jth weight memory can be derived according to

$$\begin{aligned} \Delta w_{ij} &= -\beta_w \frac{\partial d_s(t)\dot{d}_s(t)}{\partial \tau_{S-RWCMAC}} \frac{\partial \tau_{S-RWCMAC}}{\partial w_{jk}} \\ &= \beta_w \Gamma\, d_s(t)\, G_0(x,t)\, a_{ij} \mu_{ij}(F_{ij}) \end{aligned} \tag{16}$$

Where β_w is positive learning rate for the output weight memory w_{ij}, the connective weight can be updated according to the following equation:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij} \tag{17}$$

2. The translations and dilations of the jth mother wavelet function can be also updated according to

$$\begin{aligned} \Delta m_{ij} &= -\beta_m \frac{\partial d_s(t)\dot{d}_s(t)}{\partial \tau_{S-RWCMAC}} \frac{\partial \tau_{S-RWCMAC}}{\partial \mu_{ij}} \frac{\partial \mu_{ij}}{\partial F_{ij}} \frac{\partial F_{ij}}{\partial m_{ij}} \\ &= -\beta_m \Gamma d_{si}(t)\, G_0(x,t)\, a_{ij} w_{ij} \mu_{ij} \frac{1 - F_{ij}^2}{(d_{rsi} - m_{ij})} \end{aligned} \tag{18}$$

$$\begin{aligned} \Delta \sigma_{ij} &= -\beta_\sigma \frac{\partial d_s(t)\dot{d}_s(t)}{\partial \tau_{S-RWCMAC}} \frac{\partial \tau_{S-RWCMAC}}{\partial \mu_{ij}} \frac{\partial \mu_{ij}}{\partial F_{ij}} \frac{\partial F_{ij}}{\partial \sigma_{ij}} \\ &= -\beta_\sigma \Gamma d_{si}(t)\, G_0(x,t)\, a_{ij} w_{ij} \mu_{ij} \frac{1 - F_{ij}^2}{\sigma_{ij}} \end{aligned} \tag{19}$$

Where β_m, β_σ are positive learning rates for the translation and the dilation parameters. The translation and dilation can be updated as follows:

$$m_{ij}(t+1) = m_{ij}(t) + \Delta m_{ij} \tag{20}$$

$$\sigma_{ij}(t+1) = \sigma_{ij}(t) + \Delta \sigma_{ij} \tag{21}$$

3.

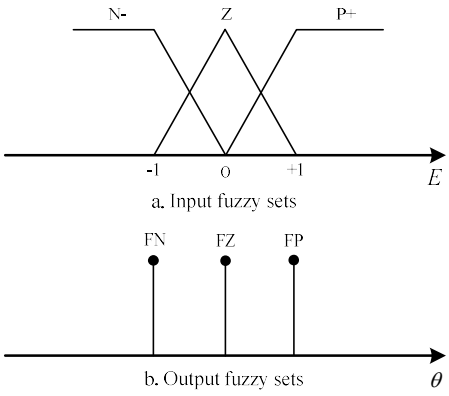


Fig.6. Fuzzy rule membership functions.

$$\tau^* = \tau_{S-RWCMAC} + \varepsilon \tag{31}$$

Where ε denotes the approximation error which is assumed to be bounded by $0 \le \|\varepsilon\| \le D$, if D is assumed to be a positive constant during the observer. Then, the conventional robust controller is designed as follows:

$$\tau_{RC} = D\,\mathrm{sgn}(E^T PB_m) \tag{32}$$

However, the parameter variations of the controlled system are difficult to measure and the exact value of the load disturbance is also difficult to know in advance for practical applications. Therefore the error bound estimation needs a continuous prediction by the proposed fuzzy logic controller (FLC) bound observer. The general FLC as shown in Fig. 6, every fuzzy rule is composed of an antecedent and a consequent part, a general form of the fuzzy rules can be represented as follows:

$$\begin{aligned} &R_o: \text{ if } E_1 \text{ is } \Gamma_{1kl} \text{ and } E_2 \text{ is } \Gamma_{2kl} \cdots E_{n_i} \text{ is } \Gamma_{ikl} \text{ Then} \\ &\quad D_{kl} \text{ is } \theta_{kl} \text{ for } i = 1,2\cdots n_i,\ k = 1,2\cdots n_k,\ l = 1,2\cdots n_l \end{aligned} \tag{33}$$

Where n_i is the input dimension, n_k and n_l are the rule for ith input variable, θ_{kl} is the output weight in the consequent part and $n_o = n_k n_l$ is the number of the fuzzy rules. Here, the membership functions for the input are chosen to be a triangular type as show in Fig. 6. In this paper, the defuzzification of the output is obtained by the height method.

$$\hat{D} = \frac{\sum_{o=1}^{n_o} \theta_o \prod_{i=1}^{n_i} \Gamma_i}{\sum_{o=1}^{n_o} \prod_{i=1}^{n_i} \Gamma_i} = \theta^T b \tag{34}$$

Where $b = [b_1 \quad b_2 \quad \cdots \quad b_{n_o}]^T \in R^{n_o}$ is a firing strength vector of rule and $\theta = [\theta_1 \quad \theta_2 \quad \cdots \quad \theta_{n_o}]^T \in R^{n_o}$ is the consequent parameter vector which is adjusted by the adaptive rule. By the universal theorem [24], there exist an optimal FLC in the form of (34) such that

$$D = \theta^{*T} b + \alpha \tag{35}$$

Where θ^* is the optimal weighting vector that achieves the minimum approximation error and α is the approximation error of the FLC and assumed to be bounded by $|\alpha| \le \delta$.

Replacing D by $\hat{D}$ in (32), the robust controller can be represented as

$$\tau_{RC} = \hat{\theta}^T b\,\mathrm{sgn}(E^T PB_m) \tag{36}$$

To compensate the approximation error of the FLC τ_{RC} is developed as follows:

$$\tau_{RC} = D\,\mathrm{sgn}(E^T PB_m) + \hat{\delta}\,\mathrm{sgn}(E^T PB_m) \tag{37}$$

Where $\hat{\delta}$ is the estimated value of δ . From (3), (6) and using (8), (31), then, the error equation becomes:

$$\dot{E} = \Lambda E + B_m(\tau^* - \tau) = \Lambda E + B_m(-\tau_{RC} - \tau_S + \varepsilon) \tag{38}$$

Theorem 1: Consider an n-link robot manipulator expressed (2). If the adaptive S-RWCMAC-based supervisory control law is designed as (8) in which the supervisory control law is designed in (28), the S-RWCMAC is presented in (11) with the adaptive laws of the S-RWCMAC are designed as (16), (18), (19) and (22), and the robust controller is developed as (37) with the estimation law in (34), (39-40), then the stability of the proposed S-RWCMAC-based supervisory control system can be ensured

$$\dot{\hat{\theta}} = \beta_\theta b \left| E^T PB_m \right| \tag{39}$$

$$\dot{\hat{\delta}} = \beta_\delta \left| E^T PB_m \right| \tag{40}$$

$$\theta^* = \left\{ \left| \varepsilon(t) \right| / norm(b) \right\} + \Omega \tag{41}$$

Where Ω is a positive constant.

Proof: Define a Lyapunov function candidate as

$$V(E, \tilde{\theta}, \tilde{\delta}, t) = \frac{1}{2} E^T PE + \frac{\tilde{\theta}^T \tilde{\theta}}{2\beta_\theta} + \frac{\tilde{\delta}^2}{2\beta_\delta} \tag{42}$$

Where $\tilde{\theta} = \theta^* - \hat{\theta}$, $\tilde{\delta} = \delta - \hat{\delta}$ β_θ and β_δ are approximation error of fuzzy compensation, estimation error, learning rates for the fuzzy compensator and the error estimator, respectively. By differentiating (42) with respect to time and using (26) and (38), we can obtain.

$$\begin{aligned} \dot{V}(E, \tilde{\theta}, \tilde{\delta}, t) &= \frac{1}{2} E^T PE + \frac{\tilde{\theta}^T \tilde{\theta}}{2\beta_\theta} + \frac{\tilde{\delta}^2}{2\beta_\delta} \\ &= \frac{1}{2} \dot{E}^T PE + \frac{1}{2} E^T P\dot{E} - \frac{1}{\beta_\theta} \tilde{\theta}^T \dot{\hat{\theta}} - \frac{1}{\beta_\delta} \tilde{\delta}\dot{\hat{\delta}} \\ &= \frac{1}{2} E(\Lambda^T P + P\Lambda)E + \frac{1}{2}(B_m^T PE + E^T PB_m) \times \\ &\quad (-\tau_{RC} - \tau_S + \varepsilon) - \frac{1}{\beta_\theta} \tilde{\theta}^T \dot{\hat{\theta}} - \frac{1}{\beta_\delta} \tilde{\delta}\dot{\hat{\delta}} \\ &= -\frac{1}{2} E^T QE + E^T PB_m(-\tau_{RC} - \tau_S + \varepsilon) \\ &\quad - \frac{1}{\beta_\theta} \tilde{\theta}\dot{\hat{\theta}} - \frac{1}{\beta_\delta} \tilde{\delta}\dot{\hat{\delta}} \end{aligned} \tag{43}$$

From (28) and using (39)–(41), (37) and (43) can be rewritten as follows.

$$
\begin{aligned}
\dot{V}\left(E,\tilde{\theta},\tilde{\delta},t\right) &= -\frac{1}{2}E^T QE + E^T PB_m\left(-\tau_{RC}-\tau_S+\varepsilon\right) \\
&\quad -\frac{1}{\beta_\theta}\tilde{\theta}\dot{\hat{\theta}} - \frac{1}{\beta_\delta}\tilde{\delta}\dot{\hat{\delta}} \\
&= -\frac{1}{2}E^T QE + E^T PB_m\varepsilon - \left|E^T PB_m\right|\hat{\delta} \\
&\quad -\left|E^T PB_m\right|\hat{\theta}^T b - \left|E^T PB_m\right|\alpha \\
&\quad -E^T PB_m\tau_S - \frac{1}{\beta_\theta}\left(\theta^* - \hat{\theta}\right)\dot{\hat{\theta}} - \frac{1}{\beta_\delta}\left(\delta-\hat{\delta}\right)\dot{\hat{\delta}} \\
&\le -\frac{1}{2}E^T QE + \left|E^T PB_m\right|\left|\varepsilon\right| + \left|E^T PB_m\right|\left|\alpha\right| \\
&\quad -\theta^* b\left|E^T PB_m\right| - \delta\left|E^T PB_m\right| \\
&= -\frac{1}{2}E^T QE - \left|E^T PB_m\right|\left(b\theta^* - |\varepsilon|\right) \\
&\quad -\left|E^T PB_m\right|\left(\delta - |\alpha|\right) \\
&\le -\frac{1}{2}E^T QE - \left|E^T PB_m\right| norm(b)\,\Omega \\
&\le -\frac{1}{2}E^T QE \le 0
\end{aligned} \tag{44}
$$

Since $\dot{V}(E,\tilde{\theta},\tilde{\delta},t)\le 0$ is a negative semi-definite function, i.e. $\dot{V}(E,\tilde{\theta},\tilde{\delta},t)\le \dot{V}(E,\tilde{\theta},\tilde{\delta},0)$, it implies that E, $\tilde{\theta}$ and $\tilde{\delta}$ is bounded functions. Let function $h\equiv E^T QE/2 \le -\dot{V}(E,\tilde{\theta},\tilde{\delta},t)$ and integrate function $h(t)$ with respect to time

$$
\int_0^t h(\tau)d\tau \le V(E,\tilde{\theta},\tilde{\delta},0) - V(E,\tilde{\theta},\tilde{\delta},t) \tag{45}
$$

Because $V(E,\tilde{\theta},\tilde{\delta},0)$ is a bounded function, and $V(E,\tilde{\theta},\tilde{\delta},t)$ is a non-increasing and bounded function, the following result can be concluded:

$$
\lim_{t\to\infty}\int_0^t h(\tau)d\tau < \infty \tag{46}
$$

In addition, $\dot{h}(t)$ is bounded; thus, by Barbalat's lemma can be shown that $\lim_{t\to\infty} h(t)=0$. It can imply that E will be converging to zero as time tends to infinite. As a result, the stability of the proposed adaptive S-RWCMAC-based supervisory control system can be guaranteed.

IV. SIMULATION AND EXPERIMENTAL RESULTS

A. Simulation results

A three-link De-icing robot manipulator as shown in Fig.1 is utilized in this paper to verify the effectiveness of the proposed control scheme. The detailed system parameters of this robot manipulator are given as: link mass $m_1, m_2, m_3\ (kg)$, lengths $l_1, l_2\ (m)$, angular positions $q_1, q_2\ (rad)$ and displacement position $d_3\ (m)$.

The parameters for the equation of motion (1) can be represented as follow:

$$
M(q) = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}
$$

$$
\begin{aligned}
M_{11} &= 9/4m_1l_1 + m_2(1/4c_2l_2 + l_1^2 + l_2l_1(c_1^2 - s_1^2)) + \\
&\quad m_3(c_2l_2^2 + l_2^2 + 2c_2l_1l_2) \\
M_{22} &= 1/4m_2l_2^2 + m_3l_2^2 + 4/3m_1l_1^2 \\
M_{23} &= M_{32} = m_3c_2l_2 \\
M_{33} &= m_3 \\
M_{12} &= M_{13} = M_{21} = M_{31} = 0
\end{aligned}
$$

$$
C(\dot{q}) = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}
$$

$$
\begin{aligned}
C_{11} &= -8m_2l_1l_2c_1s_1\dot{q}_1 + (-1/2m_2s_2c_2l_2^2 + m_3(-2s_2c_2l_2^2 \\
&\quad - 2s_2l_1l_2)\dot{q}_2 \\
C_{21} &= (-1/2m_2s_2c_2l_2^2 + m_3(-2s_2c_2l_2^2 - 2s_2l_1l_2)\dot{q}_1 \\
C_{22} &= -m_3s_2l_2\dot{d}_3 \\
C_{23} &= -2m_3s_2l_2\dot{q}_2 \\
C_{32} &= -m_3s_2l_2\dot{q}_2 \\
C_{12} &= C_{13} = C_{31} = C_{33} = 0
\end{aligned}
$$

$$
G(q) = \begin{bmatrix} (1/2c_1c_2l_2 + c_1l_1)m_2g \\ (-1/2s_1s_2l_2m_2 + c_2l_2m_3)g \\ m_3g \end{bmatrix} \tag{47}
$$

Where $q=[q_1, q_2, d_3]\in R^3$ and the shorthand notations $c_1=\cos(q_1)$, $c_2=\cos(q_2)$, $s_1=\sin(q_1)$ and $s_2=\sin(q_2)$ are used.

For the convenience of the simulation, the nominal parameters of the robotic system are given as $[m_1, m_2, m_3, l_1, l_2, g]=[3, 2, 2.5, 0.14, 0.32, 9.8]$ and the initial conditions $[q_1, q_2, d_3, \dot{q}_1, \dot{q}_2, \dot{d}_3, 0]=[0, 0, 0, 0, 0, 0]$ and the unknown lumped uncertainly in (3) is $L(x,t)$ which is assumed to be a square ware with amplitude ± 0.5 and period 2π. The desired reference model is defined as

$$
\begin{pmatrix} \dot{x}_{d1} \\ \dot{x}_{d2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -30 & -20 \end{pmatrix}\begin{pmatrix} x_{d1} \\ x_{d2} \end{pmatrix} + \begin{pmatrix} 0 \\ 50 \end{pmatrix} r(t) \tag{48}
$$

Where $[x_{d1}(0), x_{d2}(0)]^T=[0, 0]^T$ and $r(t)$ is a periodic rectangular signal. The adaptive S-RWCMAC-based supervisory control system implemented here needs to know the actual values $F_0(x,t)=1$, $G_0(x,t)=1$, the bound of the unknown lumped uncertainly $L^U(x,t)=5$ and $\overline{V}=4$.

Furthermore, the input variables of S-RWCMAC are d_{s1}, d_{s2} and d_{s3}, the translation and dilation of mother wavelet functions are selected to cover the input space $\{[-0.5, 0.5]\}_3$, the initial value of mother wavelet functions, the recurrent gain and weight memory are defined as: $\sigma_{ij}=0.15$, $r_{ij}=1.0$, $w_{ij}=0$, $\lambda_i=1$, for $j=1, 2, \cdots 9$, $i=1, 2, 3$ and

$$
\begin{aligned}
&[m_{i1}, m_{i2}, m_{i3}, m_{i4}, m_{i5}, m_{i6}, m_{i7}, m_{i8}, m_{i9}] \\
&\quad = [-0.4, -0.3, -0.2, -0.1, 0, 0.1, 0.2, 0.3, 0.4]
\end{aligned}
$$

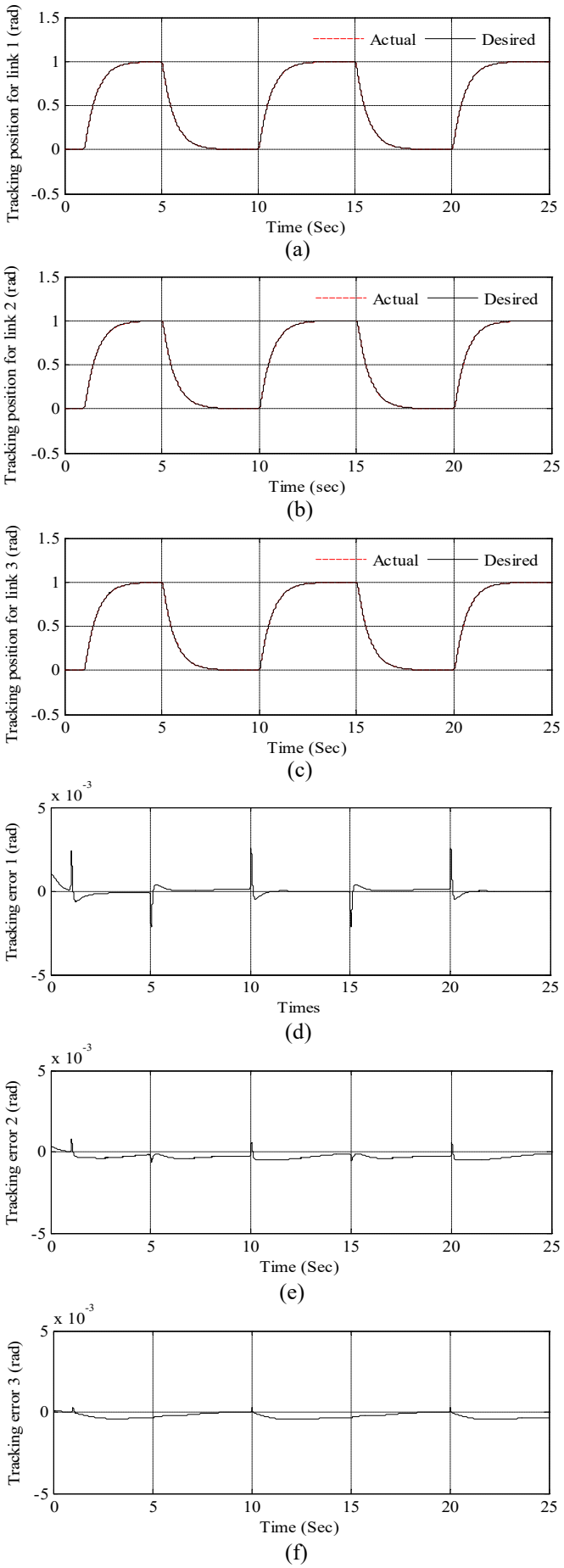


Fig.7. Simulation position responses and tracking error of the proposed adaptive S-RWCMAC-based supervisory control system at links 1, 2 and 3.

. Finally, the learning rates of S-RWCMAC are chosen such as: $\beta_{wi} = 0.05$, $\beta_{mi} = 0.02$, $\beta_{\sigma i} = 0.02$, $\beta_{ri} = 0.02$. For the adaptive robust controller, the each input variable $e_{ik}, \dot{e}_{il}$ was divided into three fuzzy subsets within $\{[-0.5, 0.5]\}_3$ along with each input dimension by using the triangular membership function for $k = 1, 2, \cdots, 3$, $l = 1, 2, \cdots, 3$, the learning rate and the initial output weight in the consequent part are selected as $\beta_{\theta i} = 0.5$, $\beta_{\delta i} = 0.2$ and $\theta_{oi} = 1$ for $o = 1, 2, \cdots, 9$, $i = 1, 2, 3$ respectively.

According to the simulation results of the S-WFCMAC-based supervisory control system due to periodic as shown in Fig. 7. The joint-position tracking responses and tracking error are shown in Fig. 7(a-c) and Fig. 7(d-f). The simulation results indicate that the high-accuracy trajectory tracking responses can be achieved by using the proposed adaptive S-RWCMAC-based supervisory control system for periodic step reference trajectory. However, the performance measure comparison of the proposed S-RWCMAC control system with the standalone CMAC control system which has been proposed in [19] are also shown in Table 1. This table shows that, for the adaptive S-RWCMAC control system, the root mean square errors are 0.038%, 0.023% and 0.012% for periodic step commands for each link, respectively. Moreover, comparing the proposed S-RWCMAC control system with the standalone CMAC control system, the root mean square errors have been reduced by about 0.023%, 0.029% and 0.032% for periodic step commands for each link, respectively. This indeed confirms the performance improvement of the proposed S-RWCMAC control system.

Table 1: Performance measure of S-RWCMAC and the Standalone CMAC approximation

Control System	Standalone CMAC Controller [19]	S-RWCMAC Controller
	RMS (rad)	RMS (rad)
Link 1	0.061%	0.038%,
Link 2	0.052%	0.023%
Link 3	0.044%	0.012%

B. Experimental results

An image of a practical experimental control system for De-icing robot consists of three manipulators and is shown in Fig 8(b). The left and right manipulators have three-link with two revolute joints and a prismatic joint. End-effectors of each manipulator have attached the motion structure to move the De-icing robot on the power line and the de-icing device. During normal operating conditions, the left and right manipulators are only operation. The between manipulator has only two joints with a revolute joint and a prismatic joint. It only works when the De-icing robot voids obstacles on the power line. In general, the operation of De-icing robot is very complex. In this paper, we consider only the three-link De-icing robot manipulator for proposed methodologies while the other manipulator is the same.

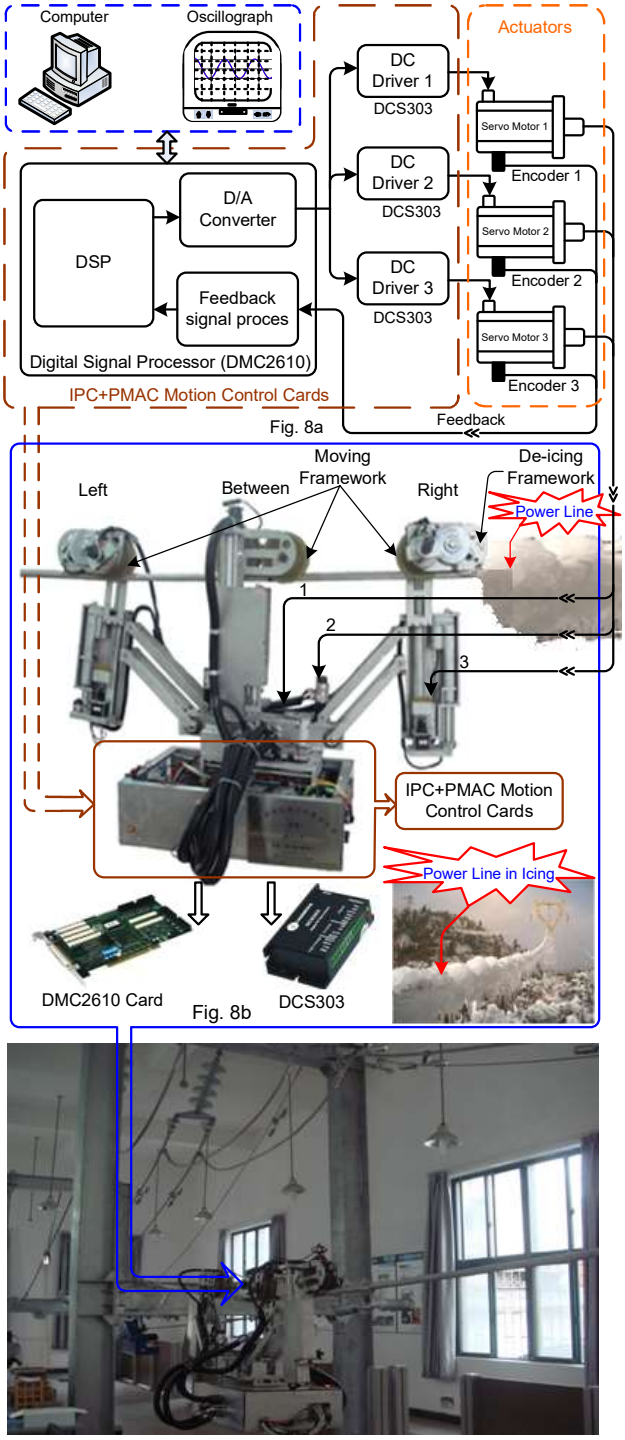
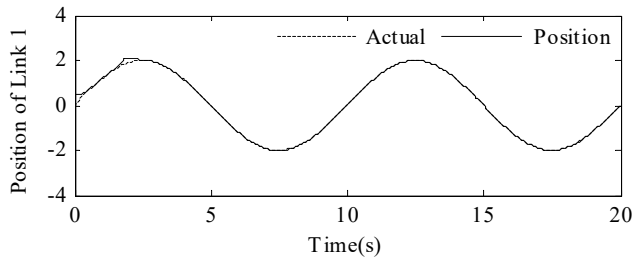


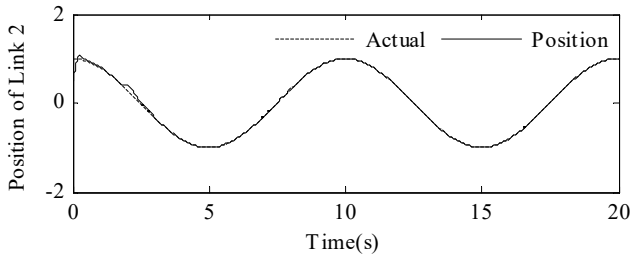
Fig. 8c

Fig.8. IPC-based De-icing robot position control system a) Block diagram of three-link De-icing robot manipulator control system, b) image of practical control system, c) image of special robot laboratory of power industry.

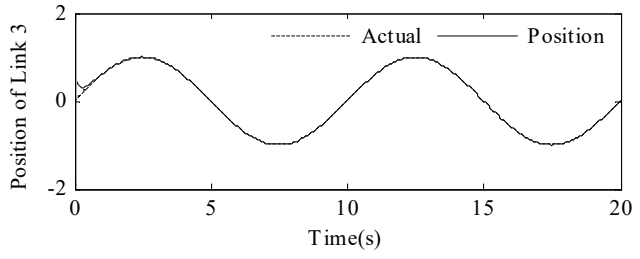
The hardware block diagram of the control system is implemented to verify the effectiveness of the proposed methodologies and is shown in Fig. 8 (a). Each joint of manipulator is derived by the "EC—**" type MAXON DC servo motors, which is designed by Switzerland Company, and each this motor contains an encoder. Digital filter and frequency multiplied by circuits are built into the encoder interface circuit to increase the precision of position feedback. The DCS303 is a digital DC servo driver developed with DSP to control the DC servo motor. The DCS303 is a micro-size



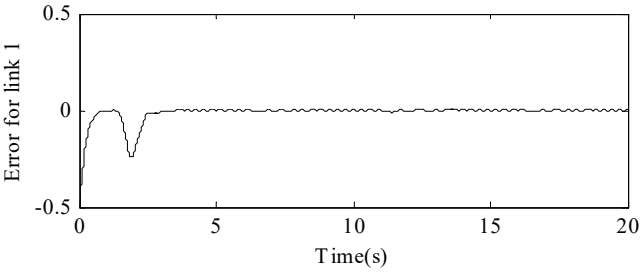
(a)



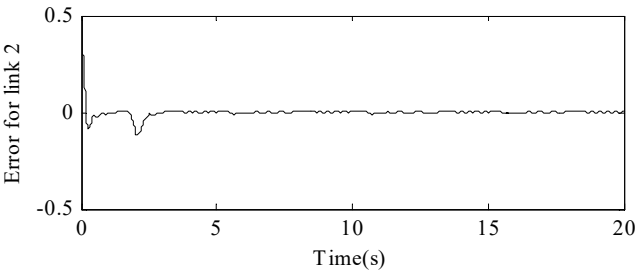
(b)



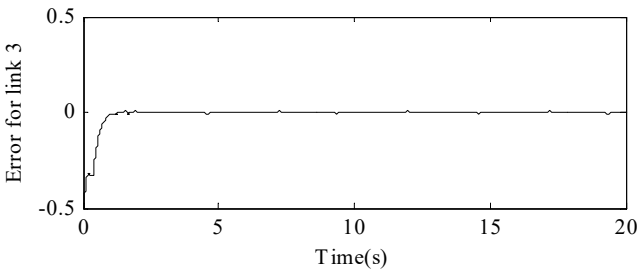
(c)



(d)



(e)



(f)

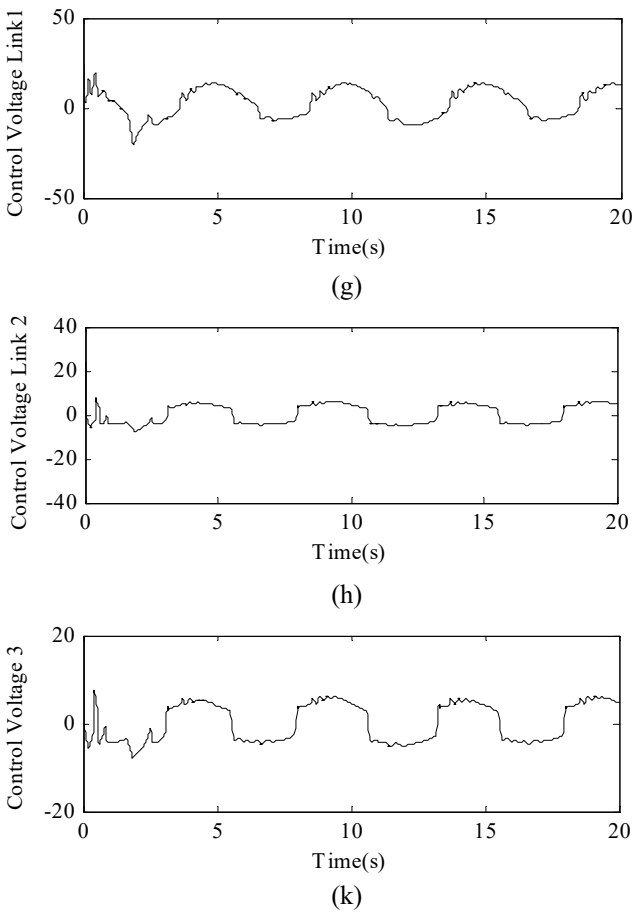


Fig.9. Experimental position responses, tracking errors of the proposed adaptive S-RWCMAC-based supervisory control system at joints 1, 2 and 3.

brush DC servo drive. It is an ideal choice for this operating environment. Two DC servo motor motion control cards are installed in the industrial personal computer, in which, a 6-axis DC servo motion control card is used to control the joint motors and a 4-axis motion control card is used to control the drive motors. Each card includes multi-channels of digital/analog and encoder interface circuits. The name of model is DMC2610 with a PCI interface connected to the IPC. The DMC2610 implements the proposed program and execute in the real time. Considering that the control sampling rate Ts = 1 ms is too demanding for the hardware implementation, Ts = 10 ms is thus considered here.

The experimental parameters of the proposed S-RWCMAC-based supervisory control system are selected in the same simulation. In this section, the control objective is to control the each joint angles of the three-link De-icing robot manipulator to move different sinusoidal commands $[q_{d1},\ q_{d2},\ q_{d3}]=[2\sin(0.2\pi t),\ \cos(0.2\pi t),\ \sin(0.2\pi t),]$ and the initial conditions of system are given as $[q_1, q_2, d_3, \dot{q}_1, \dot{q}_2, \dot{d}_3, 0]=[0.5, 0.5, 0.5, 0, 0, 0]$. Finally, the experimental position responses, tracking errors and control effort results of the proposed S-RWCMAC-based supervisory control system are depicted in Fig. 9 (a), (b), (c), Fig. 9 (d), (e), (f) and Fig. 9 (g), (h), (k). According to these experimental results of proposed S-RWCMAC-based supervisory control system due to sinusoidal reference trajectories; it is shown that high-accuracy tracking performance of proposed S-RWCMAC-based supervisory control system can also be archived for sinusoidal reference commands. For the experimental results, the performance measure comparison of the proposed S-RWCMAC control system with the standalone CMAC control system which has been proposed in [19] are also shown in Table 2. This table shows that, for the adaptive S-RWCMAC control system, the root mean square errors are 0.078%, 0.055% and 0.043% for sinusoidal commands for each link, respectively. Moreover, comparing the proposed S-RWCMAC control system with the standalone CMAC control system, the root mean square errors have been reduced by about 0.023%, 0.027% and 0.011% for sinusoidal commands for each link, respectively. This indeed confirms the performance improvement of the proposed S-RWCMAC control system.

Table 2: Performance measure of S-RWCMAC and the Standalone CMAC approximation

Control System	Standalone CMAC Controller [19]	S-RWCMAC Controller
	RMS (rad)	RMS (rad)
Link 1	0.091%	0.068%
Link 2	0.072%	0.045%
Link 3	0.054%	0.043%

V. CONCLUSIONS

This study has successfully implemented an adaptive S-RWCMAC-based supervisory control system for the three-link De-icing robot manipulator to achieve high-precision position tracking performance. Due to in the proposed scheme consists of the adaptive S-RWCMAC, the supervisory and the adaptive robust controllers, in which, the adaptive S-RWCMAC controller incorporates the advantages of the wavelet decomposition property with a CMAC fast learning ability, dynamic response and input space dimension of S-RWCMAC can be simplified, The supervisory controller is appended to the adaptive S-RWCMAC to force the system states within a predefined constraint set and the adaptive robust controller is developed to dispel the effect of approximate error. The online tuning laws of S-RWCMAC parameters and error estimation of adaptive robust controller are derived in gradient-descent learning method and the Lyapunov function so that the stability of the system can be guaranteed. Finally, through the simulation and experimental results indicate that the proposed S-RWCMAC-based supervisory control system can achieve favorable tracking performance for difference reference commands. This proposed system can also be applied to other systems, such as mobile robotic, AC servo system and so on.

REFERENCES

[1] B. J. Choi, S. W. Kwak, B. K. Kim, "Design of single-input fuzzy logic controller and its properties," *Fuzzy Sets Syst*., 106(1999), 299-308.

[2] B. J. Choi, S. W. Kwak and B. K. Kim, "Design and stability analysis of single-input fuzzy logic controller," *IEEE Syst. Man Cybers. B*, vol. 30, no. 2, pp. 303-309, Apr. 2000.

[3] Kashif Ishaque, S. S. Abdullah, S. M. Ayob, Z. Salam, "Single input fuzzy logic controller for unmanned underwater vehicle," *J Intell Robot Syst*, vol. 59, no. 3, pp. 87-100, Feb. 2010.

[4] J. S. Albus, "A new approach to manipulator control: The cerebellar model articulation controller (CMAC)," *J. Dyn. Syst. Meas. Control*, vol. 97, no. 3, pp. 220–227, 1975.

[5] H. Shiraishi, S. L. Ipri, and D. D. Cho, "CMAC neural network controller for fuel-injection systems," *IEEE Trans. Control Syst. Technol.*, vol. 3, no. 1, pp. 32–38, Mar. 1995.

[6] S. Jagannathan, S. Commuri, and F. L. Lewis, “Feedback linearization using CMAC neural networks,” *Automatica*, vol. 34, no. 3, pp. 547–557, 1998.
[7] Y. H. Kim and F. L. Lewis, “Optimal design of CMAC neural-network controller for robot manipulators,” *IEEE Trans. Syst. Man Cybern. C, Appl. Rev*., vol. 30, no. 1, pp. 22–31, Feb. 2000.
[8] C. T. Chiang and C. S. Lin, “CMAC with general basis functions,” *J. Neural Netw*., vol. 9, no. 7, pp. 1199–1211, 1996.
[9] H. C. Lu, C. Y. Chuang, M. F. Yeh, “Design of hybrid adaptive CMAC with supervisory controller for a class of nonlinear system,” *Neurocomputing*, vol. 72, no. 7-9, pp. 1920-1933, Aug. 2009.
[10] C. M. Lin, T. Y. Chen, “Self-Organizing CMAC control for a class of MIMO uncertain nonlinear systems,” *IEEE Neural Nets*. Vol. 20, no. 9, pp. 1377-1384, Sep. 2009.
[11] S.-Y. Wang, C.-L. Tseng, C.-C Yeh, “Adaptive supervisory Gaussian-cerebellar model articulation controllers for direct torque control induction motor drive,” *IET Electr. Power Appl.*, vol. 5, Iss. 3, pp. 295-306, June 2011.
[12] Young H. Kim, and Frank L. Lewis, “Optimal Design of CMAC Neural-Network Controller for Robot Manipulators,” *IEEE Trans. On Syst. Man Sybs. Part C,* vol. 30, no. 1, Feb. 2000.
[13] H.-J. Uang and C.-C. Lien, “Mixed H_2/H_∞ PID tracking control design for uncertain spacecraft systems using a cerebellar model articulation controller,” *IEE Proc.-Control Theory Appl.*, vol. 153, no. 1, pp. 1-13, Jan. 2006.
[14] C. M. Lin, L. Y. Chen and C. H. Chen, “RCMAC Hybrid Control for MIMO Uncertain Nonlinear System Using Sliding-Model Technology,” *IEEE Trans. Neural Network*, vol. 18, no. 3, pp. 708-720, May. 2006.
[15] Hahn-Ming Lee, Chih-Ming Chen and Yung-Feng Lu, “A Self Organizing HCMAC Neural Network Classifier,” *IEEE Trans. Neural Network*, vol. 14, no. 1, pp. 15-27, Jan. 2003.
[16] Wen Yu, Floriberto Ortiz Rodriguez, and Marco A. Moreno-Armendariz, “Hierarchical Fuzzy CMAC for Nonlinear Systems Modeling,” *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 5, pp. 1302-1314, Oct. 2008.
[17] Ming-Feng Yeh, “Single-input CMAC control system,” *Neurocomputing*, vol. 70, no. 16-18, pp. 2638-2644, Apr. 2007.
[18] M. F. Yeh, H. C. Lu and J. C. Chang, “Single-input CMAC control system with direct control ability,” *IEEE International Conf. Syst. Man Cybern.*, vol. 3, Oct. 2006.
[19] M. F. Yeh and C. H. Tsai, “Standalone CMAC control systems with online learning ability,” *IEEE Trans. Syst. Man Cybern. B*, vol. 40, no. 1 pp. 43-53, Feb. 2010
[20] R. J. Wait, “Development of new training algorithms for neuro-wavelet systems on the robust control of induction servo motor drives,” *IEEE Trans. Ind. Electron.*, vol. 49, no. 6, pp. 1323-1341, Dec. 2002.
[21] F.F.M. EI-Sousy, “Robust wavelet-neural network sliding-mode control system for permanent magnet synchronous motor drives,” *IET Electr. Power Appl.*, vol. 5, Iss. 1, pp. 113-132, 2011.
[22] C. H. Lu, “ Design and application of stable predictive controller using recurrent wavelet neural networks,” *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3733-3742, Sep. 2009.
[23] F.-J. Lin, S.-Y. Chen, Y.-C. Hung “Field-programmable gate array-based recurrent wavelet neural network control system for linear ultrasonic motor,” *IET Electr. Power Appl.*, vol. 3, Iss. 4, pp. 289-312, 2009.
[24] L. X. Wang, Adaptive Fuzzy Systems and Control: Design and Stability Analysis. Englewood Cliffs, NJ: Prentice-Hall, 1994.