Journal of Robotics and Control (JRC)
Volume 4, Issue 4, 2023
ISSN: 2715-5072, DOI: 10.18196/jrc.v4i4.18464

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

Thanh Quyen Ngo [1*], Tong Tan Hoa Le [2], Binh Minh Lam [3], Trung Kien Pham [4]
[1, 2, 3, 4] Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh, Vietnam
Email: [1] ngothanhquyen@iuh.edu.vn, [2] letongtanhoa@iuh.edu.vn, [3] lambinhminh@iuh.edu.vn, [4] phamtrungkien@iuh.edu.vn
*Corresponding Author

***Abstract*—The control of any robotic system always faces many great challenges in theory and practice. Because between theory and reality, there is always a huge difference in the uncertainty components in the system. That leads to the accuracy and stability of the system not being guaranteed with the set requirements. 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 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 the 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 from the gradient-descent learning method and Lyapunov function so that the stability of the system can be guaranteed. The simulation and experimental results of the novel three-link De-icing robot manipulator are provided to verify the effectiveness of the proposed control methodology. The results indicate that the proposed model has superior accuracy compared to that of the Standalone CMAC Controller. The parameters of the average squared error in the S-RWCMAC - based 3 robot joints are lower than those of the Standalone CMAC Controller by 0.023%, 0.029%, and 0.032%, respectively.**

Keywords*—Wavelet; Recurrent Wavelet Cerebellar Model Articulation Controller (RWCMAC); De-icing Robot Manipulator; Supervisory Control; Transmission Line; Obstacle Crossing; Path Planning.

I. INTRODUCTION

Fuzzy Logic Control (FLCs) are deployed in complex, high-order systems that require the use of all states. These state variables are used to represent the contents of the rule premises. Therefore, creating a complex network of rules becomes a significant challenge in designing and implementing FLCs. This creates distinct difficulty when requiring many control rules and requires considerable effort from experts. However, in complex higher-order systems, we often must increase the number of state variables. That means increasing the number of control rules. This not only takes time and effort in the process of creating rules but also requires a deep understanding of the system and control rules on the part of experts. Therefore, deploying FLCs in complex, high-order systems become a difficult task. To overcome this challenge, a careful assessment of the effectiveness and complexity of creating and implementing control rules should be conducted. Research and development of advanced methods to reduce the number of rules and effort required to implement FLCs is an important step to overcome the limitations of this approach. To address these issues, single-input Fuzzy Logic controllers (S-FLC) were proposed for the identification and control of complex dynamical systems [1]-[5]. As a result, the number of fuzzy rules is greatly reduced compared to the case of conventional FLCs, but its control performance is almost the same as conventional FLCs.

Cerebellar model articulation controller (CMAC) was proposed by Albus in 1975 [6] 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 [7]-[10]. The conventional CMACs are regarded as non-fully connected perceptron-like associative memory networks 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 [11] 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 networks in many applications were well documented [12]-[15].

Most control systems are developed in the literature based on the CMAC which is used to approach the nonlinear mapping [16]-[20], in which the input dimension of CMAC includes 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, other control schemes are also proposed in [21]-[25] by combination with conventional control techniques (sliding model control, etc.) so that the dimension of the input space is reduced. Recently, there are some researchers [26]-[30] who have reported very good results based on the CMAC control system. In [31]-[34], 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 disadvantage of the proposed CMAC control system adopts two learning stages, an off-line learning stage, and an on-line learning stage. In [35], the later proposed controller is developed the same in [32], but this control scheme not only overcomes disadvantages in [33] but also the stability of the control system can be guaranteed. However, the major drawback of the above single-input CMACs is they belong to static networks.

Many studies have been conducted on the application of neural networks (NNs) to predict, recognize, and control dynamic systems [36]-[40]. One of the most outstanding advantages of NN is its ability to approximate arbitrary linear or nonlinear systems through learning. Based on that advantage, NNs are used to approximate mathematical models of control systems. Structurally, NNs can be classified primarily into transmission neural networks (FNNs) [41], [42] and regression neural networks (RNNs) [43], [44]. RNNs have capabilities superior to FNNs, such as status response and information storage capabilities [43], [44]. Because the RNN has a feedback loop, it captures the state of the system through error, so the RNN represents better control performance. However, regardless of FNN or RNN, the learning process is slow because all weights are updated in each study cycle. Therefore, the effectiveness of NN is limited in math problems that need to be learned online. The cerebellar model articulation controller (CMAC) is widely applied to control the closed loop of complex dynamic systems due to its fast-learning characteristics, good generalization capabilities, and simple computations compared to multi-user perceptrons with back-propagation algorithms [45], [46]. The CMAC is a perceptron-like associative memory network that is not fully bound to overlapping receiving fields. The application of CMAC is not limited to control problems but also to the modeler's function approximation. The CMAC network has proven that it can approximate a nonlinear function with any desire. The advantages of using CMAC over conventional NN in many practical applications have been presented in recent literature [47]-[50]. However, the main disadvantage of current CMACs is that their application domain is limited to static. Over the past decade, much research has been conducted on the applications of wavelet neural networks, combining the ability to learn from the processes of artificial neural networks and wavelet separation capabilities [51]-[53] to identify and control dynamic systems [54]-[59]. In [54], wavelet networks were proposed as an alternative to FNN for approximate arbitrary nonlinear functions based on wavelet transform theory, and the backstream algorithm was adapted to train wavelet networks. Many successful applications have been implemented based on neural wavelet networks (WNNs) that combine network learning and wavelet dissociation [55]-[58]. Unlike conventional NN networks, WNN member functions are spatially localized wavelet functions. Therefore, WNN networks are capable of learning more efficiently than conventional NN networks for system control and identification as demonstrated in [55]-[57]. Therefore, WNN networks have been of considerable interest in applications to handle control systems with uncertain and non-linear mathematical models as presented in [57], [58]. Zhang et al. in [59] described wavelet-based neural networks for learning and estimating mathematical models. The structure of this network is like that of the radial base function network except that the radial functions are replaced by orthogonal scaling functions. From a function representation point of view, traditional radial base function networks can be represented under any function in extended space. However, it is redundant. That means that a given function's radial base function lattice representation is not unique and probably not the most efficient.

In this paper, by combining the fast learning property of CMAC, the 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 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 the gradient-descent learning method which can be caused by 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, otherwise is not.

- The use of S-RWCMAC in de-icing robot systems on power lines has several important contributions as follows: High performance: The S-RWCMAC is specifically designed to work on power lines and ice breakers. It can reach and work in difficult and dangerous positions that humans cannot perform. The use of this robot increases work efficiency and reduces the time required to restore the electrical system.
- In addition, the supervisory and the adaptive robust controllers are proposed to append to the S-RWCMAC in order to get rid of the approximation error and maintain the system stability.
- Safety for humans: The use of S-RWCMAC minimizes the hazards and risks associated with humans having to work on high voltage poles and in extreme weather conditions. This robot can perform icebreaking tasks automatically and accurately, thereby reducing the risk of work accidents.
- Flexibility and remote controllability: The S-RWCMAC can be controlled remotely, allowing remote management and operation of icebreaker

operations on power lines. This minimizes human intervention and provides flexibility in performing various tasks.

In summary, the use of S-RWCMAC in ice breaker robots on power lines brings many important contributions such as increasing work efficiency, ensuring human safety, saving costs and providing flexibility during control.

This paper is organized as follows: System description is described in Section II. Section III presents the 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 in the equation (1).

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

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 the dynamic properties given in section IV. By rewriting (1), the dynamic equation of the robot manipulator can be obtained as (2).

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

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

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 a 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 a vector that represents the joint position and velocity. Finally, the system (2) can be rewritten as (3).

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

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 (4).

$$e = q_d(t) - q(t) \quad (4)$$

and the system-tracking error vector is defined as (5).

$$E \underline{\Delta} \ [e^T \quad \dot{e}^T \quad \cdots \quad e^{n-1^T}]^T \in R^{nm} \quad (5)$$

Where n is the order of the 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 (6).

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

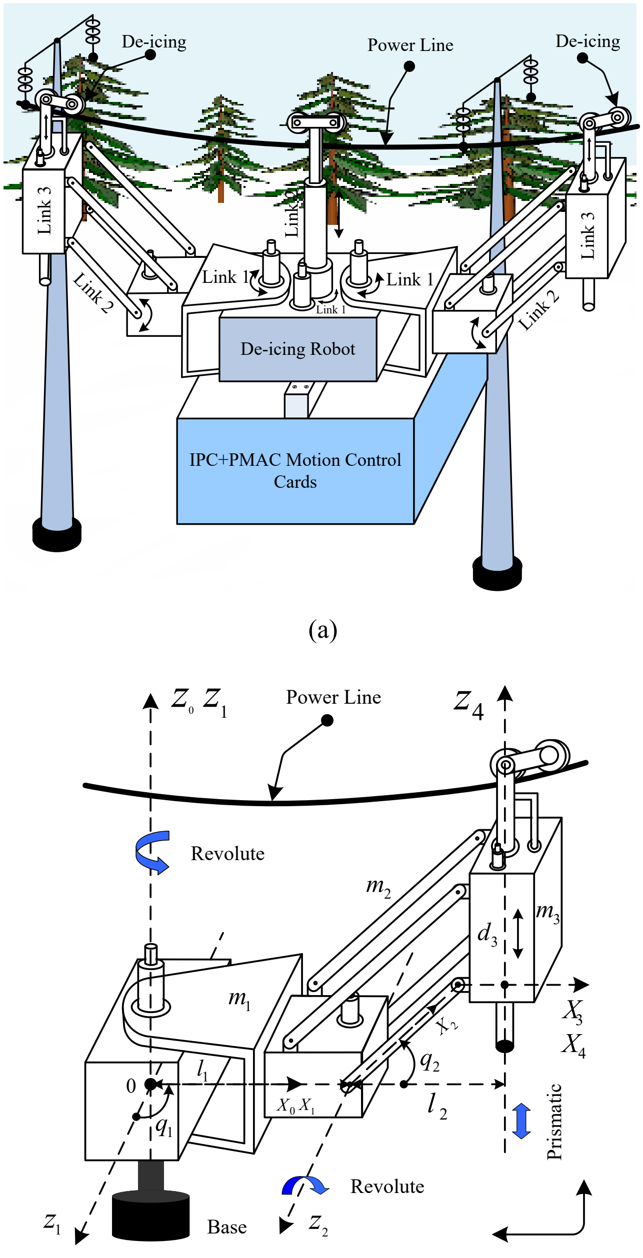

Fig. 1. Architecture of three-link De-icing robot manipulator

By substituting the ideal controller (6) into (3), the error dynamic equation is given as (7).

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

It is obvious that errors will 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 the complex plane. However, the ideal controller in (6) can not determine, because of $L(x,t)$ is exactly unknown for practical applications. So, to this problem, a proposed adaptive S-

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

III. ADAPTIVE S-RWCMAC-BASED SUPERVISORY CONTROL SYSTEM

The architecture of the adaptive S-RWCMAC-based supervisory control system is shown in Fig. 2 which consists of adaptive S-RWCMAC.

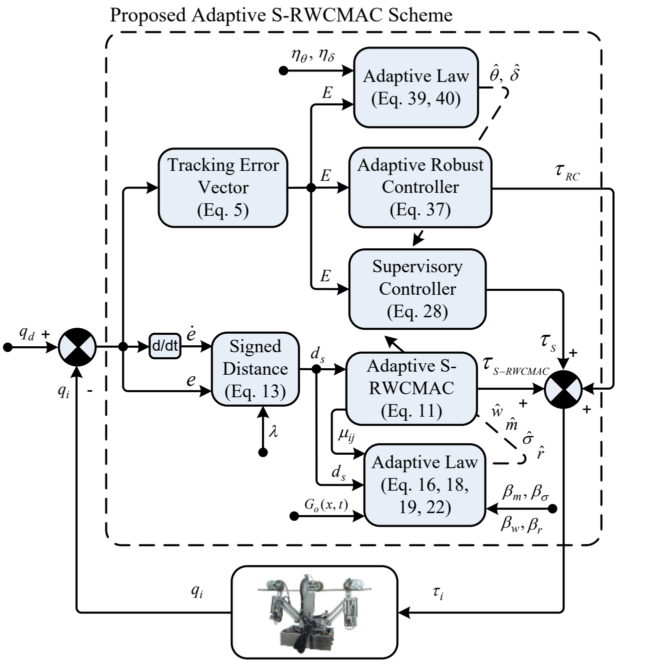

Fig. 2. Block diagram of proposed adaptive S-RWCMAC-based supervisory control system

The supervisory controller, and the adaptive robust controller with the equation (8).

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

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 to 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 distributed and 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 the basic Gaussian function for each block is given here as a mother wavelet which can be represented as (9).

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

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

Where $F_{ij}(d_{rsi}, m_{ij}, \sigma_{ij}, k) = \frac{(d_{rsi}(k) - m_{ij})}{\sigma_{ij}}$, 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 (10).

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

Where r_{ij} is the recurrent gain, k denotes the time step and $\mu_{ij}(k-1)$ denotes the value of $\mu_{ij}(k)$ through a time delay. 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 (11).

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

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.

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 2-D input.

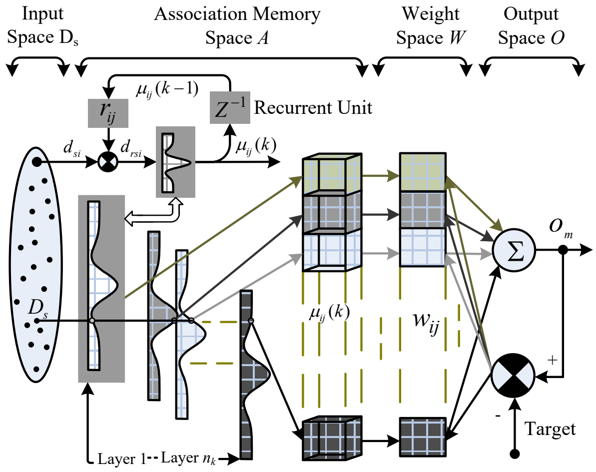

Fig. 3. Architecture of a single-input RWCMAC

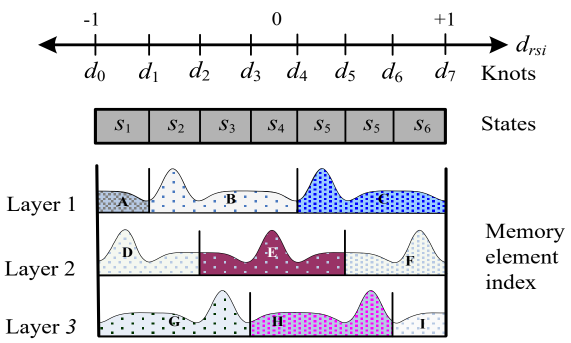

Fig. 4. Block division of S-RWCMAC with wavelet function

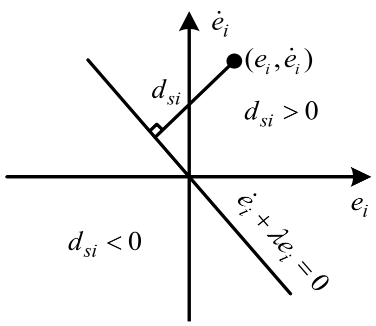

Fig. 5. Derivation of a signed distance

The switching line is defined as (12).

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

Where λ_{n-1} is a constant. Then, the signed distance between the switching line and the operating point $(e_i, \dot{e}_i) \in R^n$ can be expressed by the equation (13).

$$\begin{aligned} d_{si} &= \frac{e_{ni}^{n-1} + \lambda_{n-1} e_{(n-1)i}^{n-2} + \cdots + \lambda_2 e_{2\iota} + \dot{\lambda}_1 e_{1\iota}}{\sqrt{1 + \lambda_{n-1}^2 + \cdots + \lambda_2^2 + \lambda_1^2}} \\ &= \Gamma(e_{ni}^{n-1} + \lambda_{n-1} e_{(n-1)i}^{n-2} + \cdots + \lambda_2 e_{2\iota} + \dot{\lambda}_1 e_{1\iota}) \end{aligned} \tag{13}$$

Where $\Gamma = 1/\left(\sqrt{1 + \lambda_{n-1}^2 + \cdots + \lambda_2^2 + \lambda_1^2}\right)$ is a positive constant. By taking the time derivative of (13) for $n = 2$ and using (3), (8). We have (14).

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

Define a cost function $V(d_s(t)) = \frac{1}{2} d_s^2(t)$; 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 equation (15).

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

With this representation of the S-RWCMAC system, it becomes straightforward to apply the backpropagation 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 (16).

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

Where β_w is the positive learning rate for the output weight memory w_{ij}, the connective weight can be updated according to the equation (17).

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

2. The translations and dilations of the jth mother wavelet function can be also updated according to (18).

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

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

Where β_m, β_σ are positive learning rates for the translation and the dilation parameters. The translation and dilation can be updated as (20) and (21).

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

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

3. Finally, the updating law for recurrent gain can be derived as (22).

$$\Delta r_{ij} = -\beta_r \frac{\partial d_s(t)\dot{d}_s(t)}{\partial \tau_{S-RWCMAC}} \frac{\partial \tau_{S-RWCMAC}}{\partial \mu_{ij}} \frac{\partial \mu_{ij}}{\partial F_{ij}} \frac{\partial F_{ij}}{\partial r_{ij}} = \beta_r \Gamma d_{si}(t)\, G_0(x,t) a_{ij} w_{ij} b_{ij} \frac{1-F_{ij}^2}{(d_{rsi}-m_{ij})} \mu_{ij}(k-1) \quad (22)$$

Where β_r is the learning rate, the recurrent gain can be updated by the equation (23).

$$r_{ij}(t+1) = r_{ij}(t) + \Delta r_{ij} \quad (23)$$

C. Supervisory Controller

The design of a supervisory controller is necessary in case of divergence of states. This controller is used to pull the states back to a predefined constraint set. The supervisory controller fires only when the system states leave the predefined constraint set. When the system states stay within the constraint set, only the adaptive S-RWCMAC will be utilized to approximate the ideal control law. The supervisory controller is presented as follows. From (3) and (6) and using (8), an error equation is obtained as (24).

$$\dot{E} = \Lambda E + B_m(\tau^* - \tau) = \Lambda E + B_m(\tau^* - \tau_{S-RWCMAC} - \tau_{RC} - \tau_S) \quad (24)$$

Where,

$$\Lambda = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \\ -k_n & -k_{n-1} & -k_{n-2} & -k_{n-3} & \cdots & -k_2 & -k_1 \end{pmatrix} \in R^{nm\times nm},$$

$$B_m = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 \\ g_{m1} & g_{m2} & \cdots & g_{mm} \end{pmatrix} \in R^{nm\times m}$$

The Lyapunov function is defined as (25).

$$V_s = \frac{1}{2} E^T P E \quad (25)$$

Where $P \in R^{nm\times nm}$ is a symmetric positive definite matrix which satisfies the Lyapunov equation (26).

$$\Lambda^T P + P\Lambda = -Q \quad (26)$$

And $Q \in R^{nm\times nm}$ is a positive definite matrix. Take the derivative of the Lyapunov function and use (24) and (26), then obtained equation (27).

$$\begin{aligned} \dot{V}_s(E,t) &= \frac{1}{2}\dot{E}^T P E + \frac{1}{2} E^T P \dot{E} \\ &= \frac{1}{2} E^T \Lambda^T P E + \frac{1}{2} E^T P \Lambda E + E^T P B_m \times \\ &\quad \left(\tau^* - \tau_{S-RWCMAC} - \tau_S\right) \\ &= -\frac{1}{2} E^T Q E + E^T P B_m \left(\tau^* - \tau_{S-RWCMAC} - \tau_S\right) \\ &\le -\frac{1}{2} E^T Q E + \left|E^T P B_m\right| \left(\tau^* + \left|\tau_{S-RWCMAC}\right| + \left|\tau_{RC}\right|\right) \\ &\quad - E^T P B_m \tau_S \end{aligned} \quad (27)$$

Assumption 1: we assume that actual value $f(x,\ t)$ and $g(x,\ t)$ can be separated as nominal part denoted by $F_0(x,\ t)$, $G_0(x,\ t)$ are known, in which $G_0(x,\ t)$ is assumed to be positive, differentiable and $G_0^{-1}(x,\ t)$. The unknown lumped uncertainly $L(x,\ t)$ is bounded by $|L(x,\ t)| \le L^U$.

Based on Assumption 1 and observing (27), the supervisory controller is designed as (28).

$$\tau_S = I \operatorname{sgn}\left(E^T P B_m\right)\left[\left|\tau_{S-RWCMAC}\right| + \left|\tau_{RC}\right| + \frac{1}{G_0(x)} \times \left(-F_0(x) + L^U + \ddot{q}_d + K^T E\right)\right] \quad (28)$$

Where sgn(□) is a sign function, and the operator index as in (29).

$$I = \begin{cases} 1, & \text{if } V_s \ge \bar{V} \\ 0, & if \ V_s < \bar{V} \end{cases} \quad (29)$$

In which $\bar{V}$ is a preset positive constant. Substituting (6) and (28) into (27) and considering the case $I = 1$, it is obtained that equation (30).

$$\begin{aligned} \dot{V}_s(E,t) &= -\frac{1}{2} E^T Q E + |E^T P B_m| \\ &\quad \times \Big[\frac{1}{G_0(x)}(\ddot{q}_d - F_0(x) - L(x,\ t) + K^T E) \\ &\quad + |\tau_{S-RWCMAC}| \\ &\quad + |\tau_{RC}| - \frac{1}{G_0(x)}(-F_0(x) + L^U + \ddot{q}_d + K^T E) \\ &\quad + |\tau_{S-RWCMAC}| + |\tau_{RC}|\Big] \\ &\le -\frac{1}{2} E^T Q E + |E^T P B_m| \frac{1}{G_0(x)} |L(x,\ t)| - \frac{1}{G_0(x)} L^U \\ &= -\frac{1}{2} E^T Q E - |E^T P B_m| \frac{1}{G_0(x)} (L^U - |L(x,\ t)|) \\ &\le 0. \end{aligned} \quad (30)$$

Using the supervisory controller τ_S shown in (28), the inequality $\dot{V}_S \le 0$ can be obtained for nonzero value of the tracking error vector E when $V_S > \bar{V}$. As the results from (30), the supervisory controller is capable to drive the tracking error to zero.

A. Adaptive Robust Controller

The S-RWCMAC approximation (11) is used to approximate the ideal controller (6) through learning. By the universal theorem, there exists an S-RWCMAC $\tau_{S-RWCMAC}$ to approximate τ^* (31) [60].

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

Where ε denotes the approximation error which is assumed to be bounded by $0 \leq \|\varepsilon\| \leq D$, if D is assumed to be a positive constant during the observer. Then, the conventional robust controller is designed as (32).

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

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. In 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 (33).

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

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 shown in Fig. 6.

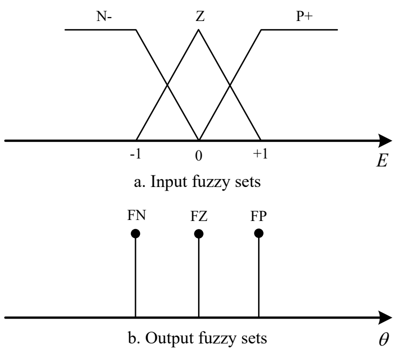

Fig. 6. Fuzzy rule membership functions

In this paper, the defuzzification of the output is obtained by the height method as (34).

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

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 [60], there exists 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 $|\alpha| \leq \delta$. Replacing D by $\hat{D}$ in (32), the robust controller can be represented as (36).

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

To compensate for the approximation error of the FLC τ_{RC} is developed as (37).

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

Where $\hat{\delta}$ is the estimated value of δ. From (3), (6) and using (8), (31), then, the error equation becomes (38).

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

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) to (40), then the stability of the proposed S-RWCMAC-based supervisory control system can be ensured.

$$\dot{\hat{\theta}} = \beta_\theta b |E^T P B_m| \tag{39}$$

$$\dot{\hat{\delta}} = \beta_\delta |E^T P B_m| \tag{40}$$

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

Where Ω is a positive constant.

Proof: Define a Lyapunov function candidate as in (42).

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

Where $\tilde{\theta} = \theta^* - \hat{\theta}$, $\tilde{\delta} = \delta - \hat{\delta}$, β_θ and β_δ are approximation error of fuzzy compensation, estimation error, and 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 equation (43).

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

From (28) and using (39) to (41), (37) and (43) can be rewritten as (44).

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

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 $h \equiv E^TQE/2 \le -\dot{V}(E,\tilde{\theta},\tilde{\delta},t)$ and integrate function $h(t)$ with respect to time (45).

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

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 (46).

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

In addition, $\dot{h}(t)$ is bounded; thus, 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 be 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 positiond_3 (m).

The parameters for the equation of motion (1) can be represented as (47).

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

$$M_{11} = 9/4m_1l_1 + m_2\left(1/4c_2l_2 + l_1^2 + l_2l_1(c_1^2 - s_1^2)\right) + m_3(c_2l_2^2 + 2c_2l_1l_2)$$

$$M_{22} = 1/4m_2l_2^2 + m_3l_2^2 + 4/3m_1l_1^2$$

$$M_{23} = M_{32} = m_3c_2l_2$$

$$M_{33} = m_3$$

$$M_{12} = M_{13} = M_{21} = M_{31} = 0$$

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

$$C_{11} = -8m_2l_1l_2c_1s_1\dot{q}_1 + (-1/2m_2s_2c_2l_2^2 + m_3(-2s_2c_2l_2^2 - 2s_2l_1l_2)\dot{q}_2$$

$$C_{21} = (-1/2m_2s_2c_2l_2^2 + m_3(-2s_2c_2l_2^2 - 2s_2l_1l_2)\dot{q}_1$$

$$C_{22} = -m_3s_2l_2\dot{d}_3$$

$$C_{23} = -2m_3s_2l_2\dot{q}_2$$

$$C_{32} = -m_3s_2l_2\dot{q}_2$$

$$C_{12} = C_{13} = C_{31} = C_{33} = 0$$

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

Where $q = [q_1,\ q_2,\ d_3] \in R^3$and the shorthand otations$c_1 = cos(q_1)$, $c_2 = cos(q_2)$, $s_1 = sin(q_1)$ 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 wave with amplitude ±0.5 and period 2π. The desired reference model is defined as (48).

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

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 $\bar{V} = 4$.

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

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

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, 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.

a) 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).

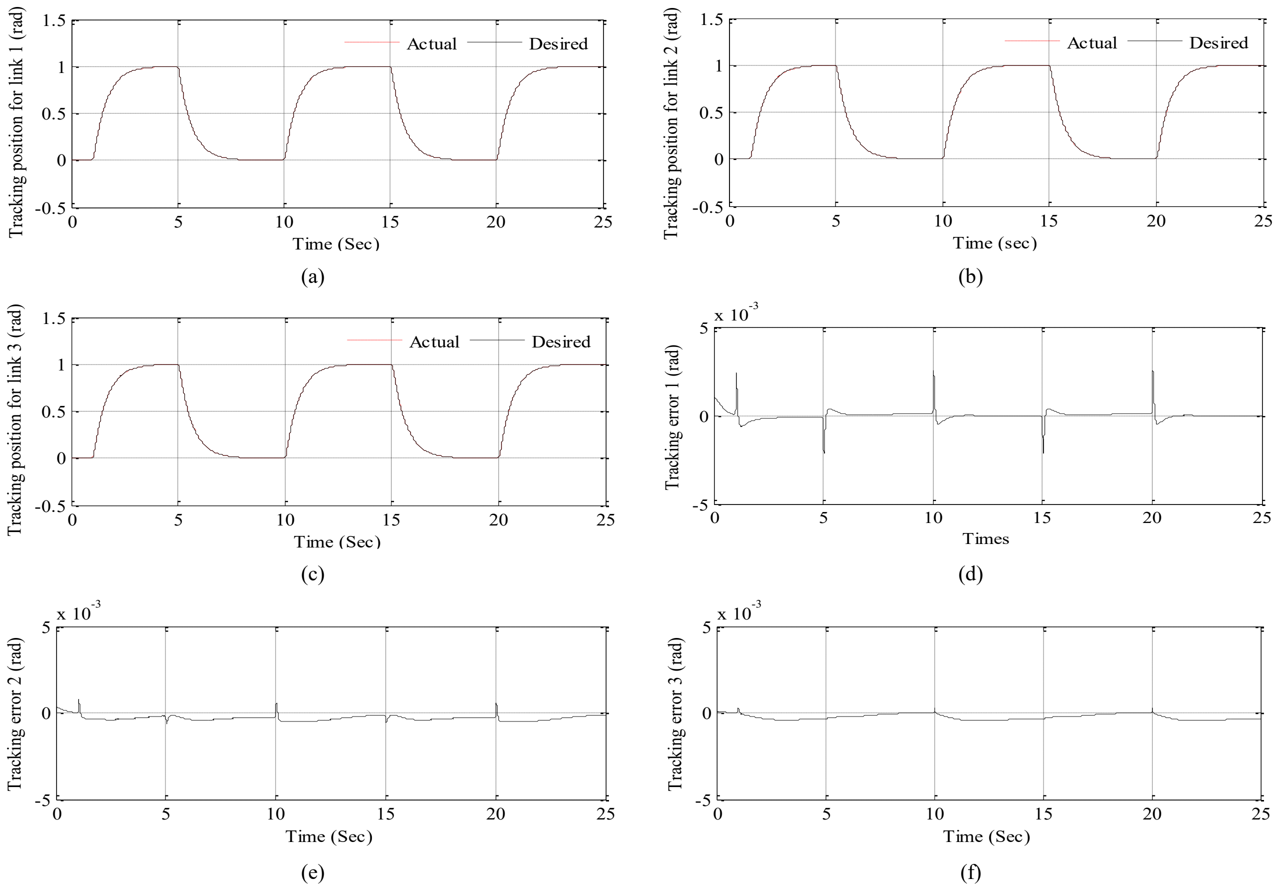

Fig. 7. Simulation position responses and tracking error of the proposed adaptive S-RWCMAC-based supervisory control system at links 1, 2 and 3

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] is also shown in Table I. 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 means 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.

Based on the given data, we can analyze some important points as follows:

1. Control performance evaluation: RMS is a measurement commonly used to evaluate the accuracy and efficiency of a control system. The lower the RMS value, the smaller the error and the better the control performance.

2. Comparison between Standalone CMAC Controller and S-RWCMAC Controller: Based on the given data, we see that the S-RWCMAC Controller has a lower RMS value than the Standalone CMAC Controller for all links. This shows that the S-RWCMAC Controller has better control in reducing errors and achieving the desired output position. At the same time, it shows that the S-RWCMAC Controller can achieve a more accurate response and reducing errors in controlling links.

TABLE I. 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%

However, for a more comprehensive and accurate assessment of the performance of these controllers, additional factors such as stability, response speed, and ability to respond to interference and fluctuations in the control system should be considered.

B. *Experimental Results*

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). 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 operating. The manipulator has only two joints a revolute joint and a prismatic joint. It only works when the De-icing robot avoids 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.

Each joint of the manipulator is derived by the "EC—**" type MAXON DC servo motors, which are 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 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 the model is DMC2610 with a PCI interface connected to the IPC. The DMC2610 implements the proposed program and executes it in 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 each joint angle 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), (i).

According to these experimental results in Fig. 9 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] is also shown in Table II. 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.

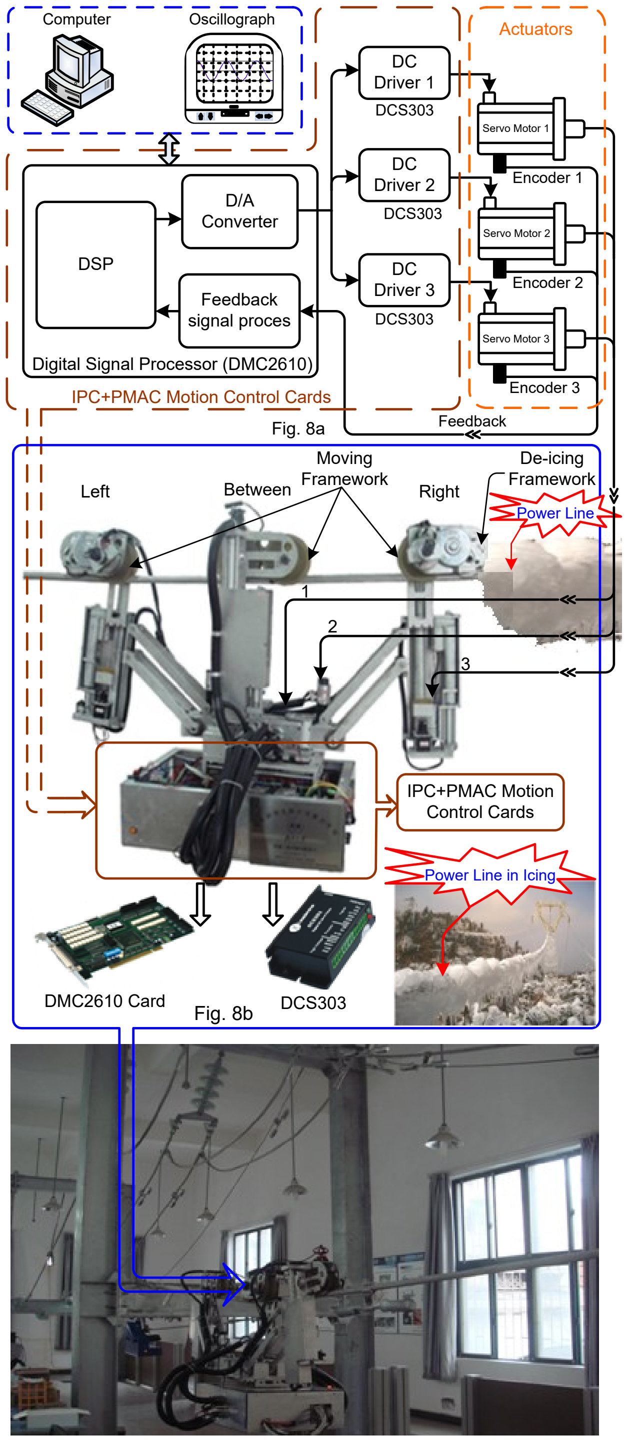

Fig. 8c

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

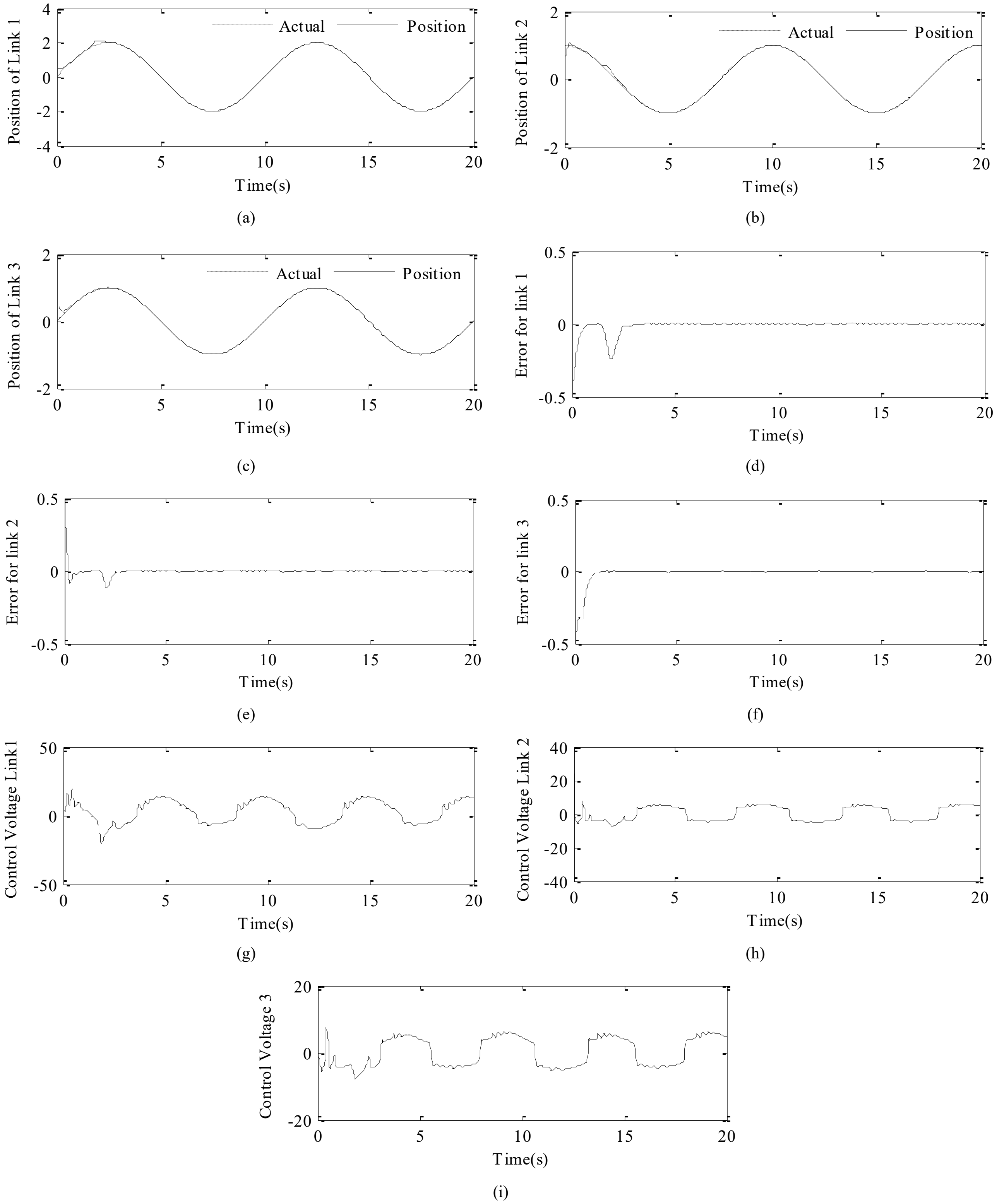

Fig. 9. Experimental position responses, tracking errors of the proposed adaptive S-RWCMAC-based supervisory control system at joints 1, 2 and 3

Moreover, comparing the proposed S-RWCMAC control system with the standalone CMAC control system, the root means 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 II. 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 And Discussion

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, which plays a very crucial role in the field of robotics. By using the S-RWCMAC structure, the system is capable of learning and adapting quickly to the environment. At the same time, the system is also capable of adjusting the parameters of the model to achieve optimal results.

One of the outstanding advantages of the system is the combination of wavelet neural networks and the fast-learning capabilities of CMAC. The use of decay properties? (locality) and fast learning capabilities helps reduce data size and increase the processing speed of S-RWCMAC, creating a more flexible and effective control method.

The adoption of the adaptive controller and the durable controller is also shown to ensure safety, avoidunwanted states, and minimize the impact of errors on system performance. In addition, the online tuning laws of S-RWCMAC parameters and error estimation of the adaptive robust controller which are derived from the gradient-descent learning method and the Lyapunov function guarantee the stability of the system. The simulation and experimental results indicate that the proposed S-RWCMAC-based supervisory control system can achieve favorable tracking performance for different reference commands.

With its good tracking performance, this monitoring control system is promising to a variety of automated systems such as mobile robots, AC servo systems, etc.

For a more comprehensive evaluation of the proposed adaptive S-RWCMAC-based monitoring control system, more research of the detailed structure of S-RWCMAC can be conducted.

The S-RWCMAC controller is a high-level algorithm in the field of control for dealing with uncertain components in nonlinear systems. These algorithms can learn from system errors and adjust parameters to optimize control performance. Thereby bringing adaptability and flexibility to the system.

By dealing with uncertain components, the S-RWCMAC has the potential to improve the control system's performance. However, like other algorithms, they also face some limitations and challenges.

First, calculating and processing data in the control system requires a lot of resources to calculate. Ensuring high performance and fast system response time requires good and efficient data processing. This can pose challenges in terms of complex algorithms.

Second, environmental conditions may create other limitations and challenges to the performance of the control system. Interference from outside and inside the system can significantly affect the stability and operation of the system. Building control systems to deal with these factors is no small challenge.

Despite the limitations and challenges, the advancement and application of the S-RWCMAC control system remain very promising. The extension and application of S-RWCMAC can be adapted to handle complex systems and meet control system requirements.

However, the research and development of new algorithms are necessary to improve the performance and application of control systems. Continued investment in control technology research and development is critical to making progress in this area. Especially exploiting the potential of the S-RWCMAC control system.

In summary, CMAC and S-RWCMAC are important and promising technologies in the field of control. However, to overcome limitations and challenges, it is necessary to invest in further research and development and to create optimal hardware and software architecture solutions for the effective implementation of CMAC and S-RWCMAC control systems in practice.

References

[1] B. J. Choi, S. W. Kwak, and B. K. Kim, "Design of single-input fuzzy logic controller and its properties," *Fuzzy Sets and Systems*, vol. 106, no. 3, pp. 299-308, 1999.

[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] K. Ishaque, S. S. Abdullah, S. M. Ayob, and 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] T. Q. Ngo, Y. N. Wang, T. L. Mai, M. H. Nguyen, and J. Chen, "Robust adaptive neural-fuzzy network tracking control for robot manipulator," *International Journal of Computers Communications & Control*, vol. 7, no. 2, 2012.

[5] T. L. Mai, Y. N. Wang, and T. Q. Ngo, "Adaptive tracking control for robot manipulators using fuzzy wavelet neural networks," *International Journal of Robotics and Automation*, vol. 30, no. 1, pp. 26-39, 2015.

[6] 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.

[7] 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.

[8] T. Q. Ngo and T. V. Phuong, "Robust adaptive self-organizing wavelet fuzzy CMAC tracking control for de-icing robot manipulator," *International Journal of Computers Communications & Control*, vol. 10, no. 4, pp. 567–578, 2015.

[9] V. -P. Ta, X. -K. Dang, and T. -Q. Ngo, "Adaptive tracking control based on CMAC for nonlinear systems," *2017 International Conference on System Science and Engineering (ICSSE)*, pp. 494-498, 2017, doi: 10.1109/ICSSE.2017.8030923.

[10] T. Q. Ngo, M. K. Duong, D. C. Pham, and D. N. Nguyen, "Adaptive Wavelet CMAC Tracking Control for Induction Servomotor Drive System," *Journal of Electrical Engineering & Technology*, vol. 14, no. 1, pp. 209-218, 2019.

[11] S. Jagannathan, S. Commuri, and F. L. Lewis, "Feedback linearization using CMAC neural networks," *Automatica*, vol. 34, no. 3, pp. 547–557, 1998.

[12] 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.

[13] C. T. Chiang and C. S. Lin, "CMAC with general basis functions," *J. Neural Netw.*, vol. 9, no. 7, pp. 1199–1211, 1996.

[14] Y. N. Wang, T. Q. Ngo, T. L. Mai, and C. Z. Wu, "Adaptive recurrent wavelet fuzzy CMAC tracking control for de-icing robot manipulator," *in Proceedings of the World Congress on Engineering and Computer Science*, vol. 1, pp. 372–379, 2012.

[15] T. Q. Ngo and Y. N. Wang, "Self-Structured Organizing Single-Input CMAC Control for Robot Manipulator," *International Journal of Advanced Robotic Systems*, vol. 8, pp. 110-119, Sep 2011.

[16] H. C. Lu, C. Y. Chuang, and 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.

[17] C. -M. Lin and T. -Y. Chen, "Self-Organizing CMAC Control for a Class of MIMO Uncertain Nonlinear Systems," in *IEEE Transactions on Neural Networks*, vol. 20, no. 9, pp. 1377-1384, Sept. 2009, doi: 10.1109/TNN.2009.2013852.

[18] M. Hwang, Y. J. Chen, M. Y. Ju, and W. C. Jiang, "A fuzzy CMAC learning approach to image based visual servoing system," *Information Sciences*, vol. 576, pp. 187-203, 2021.

[19] C. -M. Lin and Y. -F. Peng, "Adaptive CMAC-based supervisory control for uncertain nonlinear systems," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 2, pp. 1248-1260, April 2004, doi: 10.1109/TSMCB.2003.822281.

[20] S. H. Lane, D. A. Handelman, and J. J. Gelfand, "Theory and development of higher-order CMAC neural networks," in *IEEE Control Systems Magazine*, vol. 12, no. 2, pp. 23-30, April 1992, doi: 10.1109/37.126849.

[21] S. -Y. Wang, C. -L. Tseng, and C. -C Yeh, "Adaptive supervisory Gaussian-cerebellar model articulation controllers for direct torque control induction motor drive," *IET Electr. Power Appl.*, vol. 5, no. 3, pp. 295-306, June 2011.

[22] Y. H. Kim and F. L. Lewis, "Optimal design of CMAC neural-network controller for robot manipulators," in *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 30, no. 1, pp. 22-31, Feb. 2000, doi: 10.1109/5326.827451.

[23] 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.

[24] C. -T. Chiang and C. -S. Lin, "CMAC with general basis functions," *Neural Networks*, vol. 9, no. 7, pp. 1199–1211, 1996.

[25] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *in IEEE Transactions on Neural Networks*, vol. 7, no. 2, pp. 388-399, March 1996, doi: 10.1109/72.485674.

[26] C. -M. Lin, L. -Y. Chen, and C. -H. Chen, "RCMAC Hybrid Control for MIMO Uncertain Nonlinear Systems Using Sliding-Mode Technology," in *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 708-720, May 2007, doi: 10.1109/TNN.2007.891198.

[27] H. -M. Lee, C. -M. Chen, and Y. -F. Lu, "A self-organizing HCMAC neural-network classifier," in *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 15-27, Jan. 2003, doi: 10.1109/TNN.2002.806607.

[28] W. Yu, F. O. RodrÍguez, and M. A. Moreno-Armendariz, "Hierarchical Fuzzy CMAC for Nonlinear Systems Modeling," in *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 5, pp. 1302-1314, Oct. 2008, doi: 10.1109/TFUZZ.2008.926579.

[29] M. N. Nguyen, D. Shi, and C. Quek, "FCMAC-BYY: Fuzzy CMAC Using Bayesian Ying–Yang Learning," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 5, pp. 1180-1190, Oct. 2006, doi: 10.1109/TSMCB.2006.874691.

[30] J. Sim, W. L. Tung, and C. Quek, "FCMAC-Yager: A Novel Yager-Inference-Scheme-Based Fuzzy CMAC," in *IEEE Transactions on Neural Networks*, vol. 17, no. 6, pp. 1394-1410, Nov. 2006, doi: 10.1109/TNN.2006.880362.

[31] M. -F. Yeh, "Single-input CMAC control system," *Neurocomputing*, vol. 70, no. 16-18, pp. 2638-2644, Apr. 2007.

[32] M. -F. Yeh, H. -C. Lu, and J. -C. Chang, "Single-Input CMAC Control System with Direct Control Ability," *2006 IEEE International Conference on Systems, Man and Cybernetics*, pp. 2602-2607, 2006, doi: 10.1109/ICSMC.2006.385256.

[33] T. Tao, H. C. Lu, C. Y. Hsu, and T. H. Hung, "The one-time learning hierarchical CMAC and the memory limited CA-CMAC for image data compression," *J. Chin. Inst. Eng.*, vol. 26. no.2, pp. 133-145, 2003.

[34] F. Xu, J. Xu, J. Zhang, C. Zhang, and Z. Wang, "Research on parallel control of CMAC and PD based on U model," *Automatika*, vol. 62, pp. 331-338, 2021.

[35] M. -F. Yeh and C. -H. Tsai, "Standalone CMAC Control System With Online Learning Ability," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 1, pp. 43-53, Feb. 2010, doi: 10.1109/TSMCB.2009.2030334.

[36] M. Agarwal, "A systematic classification of neural-network-based control," in *IEEE Control Systems Magazine*, vol. 17, no. 2, pp. 75-93, April 1997, doi: 10.1109/37.581297.

[37] J. G. Kuschewski, S. Hui, and S. H. Zak, "Application of feedforward neural networks to dynamical system identification and control," in *IEEE Transactions on Control Systems Technology*, vol. 1, no. 1, pp. 37-49, March 1993, doi: 10.1109/87.221350.

[38] C. -M. Lin and C. -F. Hsu, "Neural-network-based adaptive control for induction servomotor drive system," in *IEEE Transactions on Industrial Electronics*, vol. 49, no. 1, pp. 115-123, Feb. 2002, doi: 10.1109/41.982255.

[39] C. -C. Ku and K. Y. Lee, "Diagonal recurrent neural networks for dynamic systems control," in *IEEE Transactions on Neural Networks*, vol. 6, no. 1, pp. 144-156, Jan. 1995, doi: 10.1109/72.363441.

[40] T. W. S. Chow and Yong Fang, "A recurrent neural-network-based real-time learning control strategy applying to nonlinear systems with unknown dynamics," in *IEEE Transactions on Industrial Electronics*, vol. 45, no. 1, pp. 151-161, Feb. 1998, doi: 10.1109/41.661316.

[41] Z. Xu, C. Sun, T. Ji, J. H. Manton, and W. Shieh, "Feedforward and Recurrent Neural Network-Based Transfer Learning for Nonlinear Equalization in Short-Reach Optical Links," in *Journal of Lightwave Technology*, vol. 39, no. 2, pp. 475-480, 2021, doi: 10.1109/JLT.2020.3031363.

[42] C. Fu, Q. -G. Wang, J. Yu, and C. Lin, "Neural Network-Based Finite-Time Command Filtering Control for Switched Nonlinear Systems With Backlash-Like Hysteresis," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 3268-3273, July 2021, doi: 10.1109/TNNLS.2020.3009871.

[43] K. -S. Hwang and C. -S. Lin, "Smooth trajectory tracking of three-link robot: a self-organizing CMAC approach," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 28, no. 5, pp. 680-692, Oct. 1998, doi: 10.1109/3477.718518.

[44] F. J. G-. Serrano, A. R. F-. Vidal, and A. A-. Rodriguez, "Generalizing CMAC architecture and training," in *IEEE Transactions on Neural Networks*, vol. 9, no. 6, pp. 1509-1514, Nov. 1998, doi: 10.1109/72.728400.

[45] J. C. Jan and S. -L. Hung, "High-order MS CMAC neural network," in *IEEE Transactions on Neural Networks*, vol. 12, no. 3, pp. 598-603, May 2001, doi: 10.1109/72.925562.

[46] Y. C. Pati and P. S. Krishnaprasad, "Analysis and synthesis of feedforward neural networks using discrete affine wavelet transformations," in *IEEE Transactions on Neural Networks*, vol. 4, no. 1, pp. 73-85, Jan. 1993, doi: 10.1109/72.182697.

[47] B. Delyon, A. Juditsky, and A. Benveniste, "Accuracy analysis for wavelet approximations," in *IEEE Transactions on Neural Networks*, vol. 6, no. 2, pp. 332-348, March 1995, doi: 10.1109/72.363469.

[48] T. Lindblad and J. M. Kinser, "Inherent features of wavelets and pulse coupled networks," in *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 607-614, May 1999, doi: 10.1109/72.761719.

[49] Q. Zhang and A. Benveniste, "Wavelet networks," in *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 889-898, Nov. 1992, doi: 10.1109/72.165591.

[50] J. Zhang, G. G. Walter, Y. Miao, and W. N. W. Lee, "Wavelet neural networks for function learning," in *IEEE Transactions on Signal Processing*, vol. 43, no. 6, pp. 1485-1497, June 1995, doi: 10.1109/78.388860.

[51] R. -J. Wai, "Development of new training algorithms for neuro-wavelet systems on the robust control of induction servo motor drive," in *IEEE Transactions on Industrial Electronics*, vol. 49, no. 6, pp. 1323-1341, Dec. 2002, doi: 10.1109/TIE.2002.804986.

[52] 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, no. 1, pp. 113-132, 2011.

[53] C. -H. Lu, "Design and Application of Stable Predictive Controller Using Recurrent Wavelet Neural Networks," in *IEEE Transactions on Industrial Electronics*, vol. 56, no. 9, pp. 3733-3742, Sept. 2009, doi: 10.1109/TIE.2009.2025714.

[54] F. -J. Lin, S. -Y. Chen, and Y. -C. Hung "Field-programmable gate array-based recurrent wavelet neural network control system for linear ultrasonic motor," *IET Electr. Power Appl.*, vol. 3, no. 4, pp. 289-312, 2009.

[55] Q. Zhang, "Using wavelet network in nonparametric estimation," in *IEEE Transactions on Neural Networks*, vol. 8, no. 2, pp. 227-236, March 1997, doi: 10.1109/72.557660.

[56] Y. Qussar, I. Rivals, L. Personnaz, and G. Dreyfus, "Training wavelet networks for nonlinear dynamic input-output modeling," *Neurocomputing*, vol. 20, pp. 173-188, 1998.

[57] L. M. Reyneri, "Unification of neural and wavelet networks and fuzzy systems," IEEE Trans. Neural Networks, vol. 10, pp. 801-814, 1999.

[58] H. Y. Dalkiliç and S. A. Hashimi, "Prediction of daily streamflow using artificial neural networks (ANNs), wavelet neural networks (WNNs), and adaptive neuro-fuzzy inference system (ANFIS) models," *Water Supply*, vol. 20, no. 4, pp. 1396-1408, 2020.

[59] C. -J. Lin and C. -C. Chin, "Prediction and identification using wavelet-based recurrent fuzzy neural networks," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 5, pp. 2144-2154, Oct. 2004, doi: 10.1109/TSMCB.2004.833330.

[60] L. X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1994.