INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 10(4):567-578, August, 2015.

Robust Adaptive Self-Organizing Wavelet Fuzzy CMAC Tracking Control for Deicing Robot Manipulator

T.Q. Ngo, T.V. Phuong

ThanhQuyen Ngo*
Faculty of Electrical Engineering,
Industrial University of HCM City , HCM City, Vietnam
*Corresponding author: thanhquyenngo2000@yahoo.com

TaVan Phuong
Faculty of Electrical and Electronics Engineering,
HCMC University of Technology And Education, Vietnam
tavphuong@yahoo.com; phuongtv@hcmute.edu.vn

Abstract:
In this paper, a robust adaptive self-organizing control system based on a novel wavelet fuzzy cerebellar model articulation controller (WFCMAC) is developed for an n-link robot manipulator to achieve the high-precision position tracking. This proposed controller consists of two parts: one is the WFCMAC approach which is implemented to cope with nonlinearities, due to the novel WFCMAC not only incorporates the wavelet decomposition property with fuzzy CMAC fast learning ability but also it will be self-organized; that is, the layers of WFCMAC will grow or prune systematically. Therefore, dimension of WFCMAC can be simplified. The second is the order which is the adaptive robust controller which is designed to achieve robust tracking performance of the system. The adaptive tuning laws of WFCMAC parameters and error estimation of adaptive robust controller are derived through the Lyapunov function so that the stability of the system can be guaranteed. Finally, the simulation and experimental results of novel three-link deicing robot manipulator are applied to verify the effectiveness of the proposed control methodology.
Keywords: Wavelet, CMAC, Deicing robot manipulator.

1 Introduction

In general, robotic manipulators have to face various uncertainties in their dynamics, such as friction and external disturbance. It is difficult to establish an exactly mathematical model for the design of a model-based control system. In order to deal with this problem, the fuzzy logic control (FLC) has found extensive applications for complex and ill-defined plants [1-2], and 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. Therefore, it requires a huge number of control rules and much effort to create them. The neural networks (NNs) are a model-free approach, which can approximate a nonlinear function to arbitrary accuracy [3–4]. However, the learning speed of the NNs is slow, since all the weights are updated during each learning cycle. In addition, the fully connected NNs are sensitive to training data. Therefore, the ability of function approximation of a general multiplayer NN is restricted to requiring online learning.

Based on the advantages of fuzzy and neural networks, the fuzzy neural network (FNN) which incorporates advantages of fuzzy inference and neuron-learning has been developed and its effectiveness is demonstrated in solving control problems [5–6]. Recently, many applications have been implemented quite successfully based on wavelet neural networks (WNNs) which combine the learning ability of network and capability of wavelet decomposition property [7–8]. Different from conventional NNs, the membership functions of WNN are spatially localized wavelet functions; and therefore, the WNNs are capable of learning more efficiently than conventional NNs for control and system identification as has been demonstrated in [7]. As a result, WNNs have been used to deal with uncertainties and nonlinearity in [8].

To overcome the disadvantages of NNs, cerebellar model articulation controller (CMAC) was proposed by Albus in 1975 [9] 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 [10–11]. Conventional CMACs are regarded as nonfully 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. To acquire the derivative information of input and output variables, Chiang and Lin [12] 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 [13–14]. However, in the above CMAC studies, the structure of CMAC are not merited of the high-level human knowledge representation and thinking of fuzzy theory.

In this paper, we propose a novel robust adaptive self-organizing wavelet fuzzy CMAC (RASOWFCM) control system for three-link deicing robot manipulator to achieve the high-precision position tracking. This control system combines the advantages of fuzzy inference system with CMAC and wavelet decomposition capability and it does not require prior knowledge of a certain amount of memory space, and a self-organizing approach of this control system demonstrates the properties of generating and pruning the input layers automatically The developed self-organizing rule of WFCMAC is clearly and easily used for real-time systems and the adaptive robust controller which are designed to achieve robust tracking performance of the system. This is the first contribution of this paper. The second contribution is that it proposes novel architecture and mathematical model of deicing robot which can be effective in practical applications. The adaptive tuning laws of WFCMAC parameters and error estimation are derived in Lyapunov method.

This paper is organized as follows: system description is given in Section 2. The structure of self-organizing WFCMAC and RASOWFCM control system are presented in Sections 3 and 4. Numerical simulation and experimental results of a three-link deicing robot manipulator under the possible occurrence of uncertainties are provided to demonstrate the tracking control performance of the proposed RASOWFCM system in Section 5. Finally, conclusions are drawn in Section 6.

2 System description

2.1 Robotic dynamic model

In general, the dynamics of an n-link robot manipulator may be expressed in the following Lagrange form:

$$M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_{\mathrm{d}} = \tau, \tag{1}$$

where $q, \dot{q}, \ddot{q} \in \mathbb{R}^n$ are the joint position, velocity and acceleration vectors, respectively, $M(q) \in \mathbb{R}^{n\times n}$ denotes the inertia matrix, $V(q,\dot{q}) \in \mathbb{R}^{n\times n}$ expresses the matrix of centripetal and Coriolis forces, $G(q) \in \mathbb{R}^{n\times 1}$ is the gravity vector, $\tau_{\mathrm{d}} \in \mathbb{R}^{n\times 1}$, $F(\dot{q})$ represents the vector of external disturbance, friction term, respectively, and $\tau \in \mathbb{R}^{m\times 1}$ is the torque vectors exerting on joints.

In this paper, a novel three-link deicing robot manipulator, as shown in Fig.1, is utilized to verify dynamic properties which are given in Section 5.

In our controller design, the following properties will be used [15].

Property 1: The inertia matrix $M(q)$ is symmetric and positive definite. It is also bounded as a function of $q : m_1 I \leqslant M(q) \leqslant m_2 I, m_1, m_2 > 0$.

Property 2: $\dot{M}(q)-2V(q,\dot{q})$ is a skew symmetric matrix. Therefore, $y^{\mathrm{T}}[\dot{M}(q)-2V(q,\dot{q})]y = 0$, where y is a $n \times 1$ nonzero vector. The control problem is to force $q(t) \in \mathbb{R}^n$ to track a given bounded reference input signal $q_{\mathrm{d}}(t) \in \mathbb{R}^n$. we have the tracking error as follows:

$$e = q_{\mathrm{d}}(t) - q(t), \tag{2}$$

and the filtered system tracking error vector is defined as

$$r = \dot{e} + \Lambda e, \tag{3}$$

where $\Lambda = \Lambda^{\mathrm{T}} > 0$, by the differentiating $r(t)$ with respect to e and using (1), the arm dynamics can be written in terms of the filtered tracking error vector as follows:

$$\begin{aligned} M(q)\dot{r} &= -V(q,\dot{q})r - \tau + M(q)(\ddot{q}_{\mathrm{d}} + \Lambda\dot{e}) + V(q,\dot{q})(\dot{q}_{\mathrm{d}} + \Lambda e) + F(\dot{q}) + G(q) + \tau_{\mathrm{d}} \\ &= -V(q,\dot{q})r - \tau + f(x), \end{aligned} \tag{4}$$

where the nonlinear function $f(x)$ is defined as follows:

$$f(x) = M(q)(\ddot{q}_{\mathrm{d}} + \Lambda\dot{e}) + V(q,\dot{q})(\dot{q}_{\mathrm{d}} + \Lambda e) + G(q) + F(\dot{q}) + \tau_{\mathrm{d}}, \tag{5}$$

where $x \triangleq [e^{\mathrm{T}} \quad \dot{e}^{\mathrm{T}} \quad q_{\mathrm{d}}^{\mathrm{T}} \quad \dot{q}_{\mathrm{d}}^{\mathrm{T}} \quad \ddot{q}_{\mathrm{d}}^{\mathrm{T}}]$ denotes the variables of the nonlinear function.

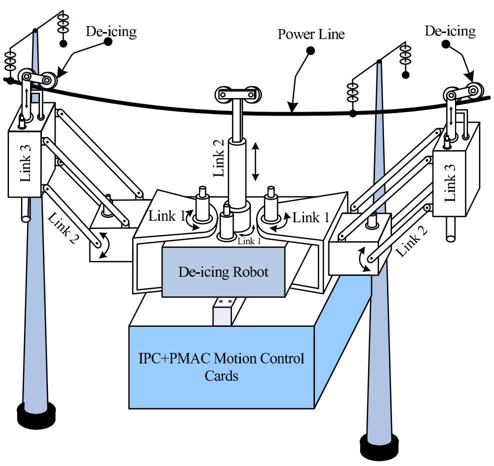

Figure 1: Architecture of three-link deicing robot manipulator.

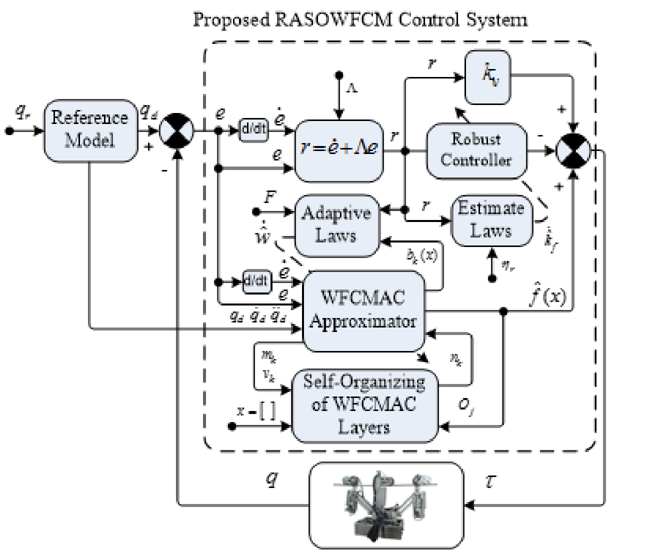

Figure 2: Block diagram of RASOWFCM control system.

2.2 Defined control law

Now, we define a control input torque as follows:

$$\tau_0 = \hat{f}(x) + K_{\mathrm{v}} r \tag{6}$$

with $\hat{f}(x)$ an estimate of $f(x)$ and a gain matrix $K_{\rm v} = K_{\rm v}{}^{\rm T} > 0$. From (4), the closed-loop system becomes

$$M(q)\,\dot{r} = -(K_{\rm v} + V(q,\dot{q}))\,r + \tilde{f}(x), \tag{7}$$

where functional estimated error is given by

$$\tilde{f}(x) = f(x) - \hat{f}(x). \tag{8}$$

This is a system error wherein filtered tracking error is driven by this estimated error function. The control τ_0 incorporates a proportional plus derivative (PD) term in $K_{\rm v}r = K_{\rm v}(\dot{e}+\Lambda e)$. In the remainder of the paper we shall use (7) to focus on selecting WFCMAC weight tuning algorithms that guarantee the stability of the filtered tracking error $r(t)$. Then, since (3), with the input considered as $r(t)$ and the output as $e(t)$, describes a stable system, standard techniques [16] guarantee that $e(t)$ exhibits stable behaviour. In fact, $\|e\|_2 \leqslant \|r\|_2/\sigma_{\min}(\Lambda)$, $\|\dot{e}\|_2 \leqslant \|r\|_2$ with $\sigma_{\min}(\Lambda)$ the minimum singular value of Λ. Generally, Λ is diagonal so that $\sigma_{\min}(\Lambda)$ is the smallest element of Λ.

3 Structure of WFCMAC

3.1 Brief of the WFCMAC

The main difference between the FCMAC and the original CMAC is that the association layer in the FCMAC is the rule layer which is represented as follows:

$$\begin{array}{ll}\mathbb{R}^{\lambda}: & \text{if } x_1 \text{ is } \mu_{1k}, \text{ and } x_2 \text{ is } \mu_{2k},\ldots, x_i \text{ is } \mu_{ik}, \text{Then } o_{ij} = w_{ij} \\ & \text{for } k=1,2,\ldots,n_k, i=1,2,\ldots,n, \lambda=1,2,\ldots,n_\lambda \text{ and } j=1,2,\ldots,m_j,\end{array} \tag{9}$$

where n is the input dimension, n_k is the number of layers for each input dimension, the number of rules n_λ equals the number of layers and m_j is the output dimension, μ_{ik} is the fuzzy set for the ith input, the kth layer, and w_{jk} is the output weight in the consequent part. Based on [17], a novel WFCMAC is represented and shown in Fig.4. It is combined a wavelet function with the FCMAC consists of input, association memory, receptive field, and output spaces, is proposed to implement the WFCMAC estimate in RASOWFCM control system which is shown in Fig.2. The signal propagation is introduced according to functional mapping as follows:

1. The first mapping X: $X \rightarrow A$ we assume that each input state variable $x = [\ x_1 \quad x_2 \quad \cdots \quad x_n\ ]$ can be quantized into n_e discrete states and that the information of a quantized state is regarded as a wavelet receptive-field basic function for each layer. The mother wavelet is a family of wavelets. The first derivative of basic Gaussian function for each layer is given here as the mother wavelet which can be represented as follows:

$$\mu_{ik}(x_i) = -\frac{x_i - m_{ik}}{\sigma_{ik}} \exp\left[-\frac{(x_i - m_{ik})^2}{2\sigma_{ik}^2}\right], \qquad i = 1,2,\cdots, n\,, k = 1,2,\cdots, n_k, \tag{10}$$

where μ_{ik} represents the kth layer of the input with m_{ik} is a translation parameter and σ_{ik} is dilation.

2. The second mapping A: $A \rightarrow R$ the information μ_{ik} of each kth layer relates to each location of receptive field space. Fig.4 illustrates a structure of two-dimension WFCMAC with wavelet basic function and $n_e = 7$ case. Areas of receptive field space are formed by multiple-input regions are called hypercube; i.e in the fuzzy rules in (9), the product is used as the “and" computation in the consequent part. The firing of each state in each layer can obtain the weight

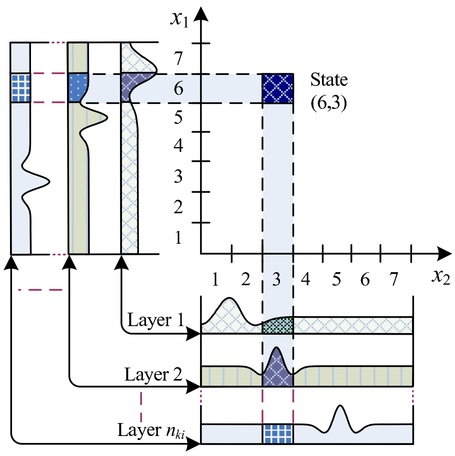

Figure 3: Two-dimension WFCMAC with wavelet function and $n_e = 7$.

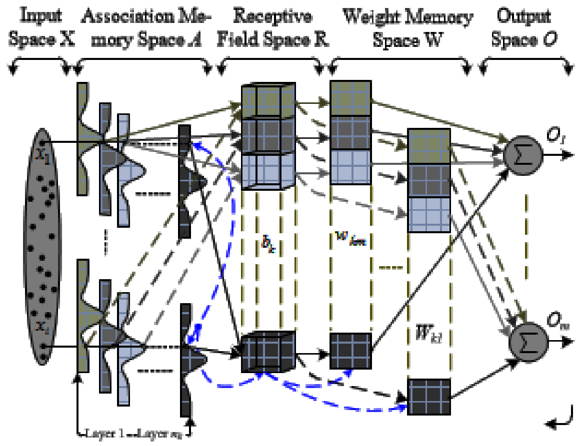

Figure 4: Block diagram of WFCMAC.

of each corresponding layer. Assuming that in 2-D WFCMAC case shown in Fig.3, input state vector is (6,3), the content of kth hypercube can be obtained as follows:

$$b_k(x,\, m_k,\, \sigma_k) = \prod_{i=1}^{n} \mu_{ik}(x_i). \tag{11}$$

3. Output mapping O: The WFCMAC output is an algebraic sum of activated weighs with hybercube elements. The jth output mathematic form can be expressed as follows:

$$O_j = \begin{bmatrix} w_{j1} & w_{j2} & \cdots & w_{jn_k} \end{bmatrix} \begin{bmatrix} b_1(x) \\ b_2(x) \\ \vdots \\ b_{n_k}(x) \end{bmatrix} = \sum_{k=1}^{n_k} w_{jk} \prod_{i=1}^{n} \mu_{ik}(x_i) \quad j = 1, 2, \cdots, m_j. \tag{12}$$

The block diagram shows in Fig.2, in which the WFCMAC plays a major role in the nonlinear function estimation. Because, there is a trade-off between a designed performance and a tedious computation so we must choose a reasonable number of layers. However, if the number of layers chosen is too small, the learning process may be insufficient to achieve a desired performance. Otherwise, if the number of layers chosen is too large, the calculation burden becomes too heavy, so it is not suitable for real-time applications. To deal with this problem, a self-organizing WFCMAC is proposed includes structure and parameter learning.

3.2 Self-organizing WFCMAC

In this section, It is necessary to determine a structure learning in order to add a new layer whether to add a new layer in membership layers depends on the firing strength $b_k \in \mathbb{R}^{n_k}$ of each layer for each incoming data x_i. If the firing strength $b_k \in \mathbb{R}^{n_k}$ of each layer for new input data x_i falls outside the bounds of the threshold, then, the WFCMAC approach will generate a new layer. The self-organizing WFCMAC can be also summarized based on [17] as follows:

1. Calculate a distance of mean MD_k, $k = 1, 2, \cdots, n_k$ in association memory A as follows:

$$MD_k(x) = \|x - |m_k|\|_2 \quad Where \quad m_k = \begin{bmatrix} m_{1k} & m_{2k} & \cdots & m_{nk} \end{bmatrix} \tag{13}$$

2. Using Max-Min method is proposed for layer growing. Find

$$\hat{k} = \arg \min_{1 \leqslant k \leqslant n_k} MD_k(x), k = 1, 2, \cdots\ n_k m, \quad if \quad \max_i MD_{\hat{k}}(x) > K_g, \tag{14}$$

Here K_g is a threshold value of adaptation with $0 < K_g \leqslant 1$. In our case $K_g = 0.1$ and a new layer is generated. This means that for a new input data, the exciting value of existing basic function is too small. In this case, number of layers increased as follows:

$$n_k(t+1) = n_k(t) + 1, \tag{15}$$

Where n_k is the number of layers at time t. Thus, a new layer will be generated and then the corresponding parameters in the new layer such as the initial mean and variance of Gaussian basic function in association memory space and the weight memory space will be defined as

$$m_{in_k} = x_i, \quad \sigma_{in_k} = \sigma_{\hat{k}}, \quad w_{in_k} = 0. \tag{16}$$

Another self-organizing learning process is considered to determine whether to delete existing layer, which is inappropriate. A Max-Min method is proposed for layer pruning.

Considering the jth output of WFCMAC in (12), the ratio of the kth component to jth output is defined as

$$MM_{jk} = \frac{v_{jk}}{O_j}\ , \quad k = 1, 2, \cdots, n_k. \tag{17}$$

Where $v_{jk} = w_{jk} b_k(x)$, Then, the minimum ratio of the kth component is defined as follows:

$$\tilde{k} = \arg \min_{1 \leqslant k \leqslant n_k} \max_{1 \leqslant j \leqslant m} MM_{jk}, \qquad if \quad MM_{\tilde{k}} \leqslant K_c. \tag{18}$$

Here K_c is a predefined deleting threshold. In our case $K_c = 0.03$ and the $\tilde{k}$th layer will be deleted. This means that for an output data, if the minimum contribution of a layer is less than the deleting threshold, then this layer will be deleted.

4 RASOWFCM Design

In the RASOWFCM scheme is shown in Fig.2, the WFCMAC is used to estimate an unmodelled nonlinear function, Moreover, the RASOWFCM law and adaptive tuning algorithms for WFCMAC are introduced from the stability analyses of the closed-loop system by using Lyapunov method Input of the WFCMAC estimator are the elements in the filtered error vector and joint positions signal, Output of the WFCMAC estimator are nonlinear dynamic function vectors in the local models.

Based on the powerful approximation ability [18], there exists an optimal WFCMAC estimator to approximate the nonlinear dynamic function in (5) such that

$$f(x) = W^{\mathrm{T}} b(x) + \varepsilon(x) \tag{19}$$

With W the ideal weight matrix and the estimated error vector $\varepsilon(x) \in \mathbb{R}^{n \times 1}$ are assumed to be given by

$$\tilde{W} = \arg \min_{\hat{W} \in M_w} [\sup_{x \in M_x} \| f(x) - \hat{W}^{\mathrm{T}} b(x) \|], 0 \leqslant \| \varepsilon \|_1 \leqslant k_f. \tag{20}$$

In which $\|\bullet\|$ is the Euclidean norm, M_x, M_w and k_f are the predefined compact sets of x and $\hat{W}$, and the positive constant

Define the WFCMAC functional estimate by

$$\hat{f}(x) = \hat{W}^{\mathrm{T}} b(x), \tag{21}$$

With $\hat{W}$ being the current values of the WFCMAC weight provided by the tuning algorithm. With the ideal weights required in (19) define the weight deviations or weight estimation errors as

$$\tilde{W} = W - \hat{W}. \tag{22}$$

With τ_o defined by (6), we design the control law as follow:

$$\tau = \tau_0 - \tau_r. \tag{23}$$

Where τ_0 is the main controller in (6), which consists of two terms: one is the PD controller and the other is WFCMAC controller is used to estimate the $f(x)$ nonlinear function and the adaptive robust controller τ_r is utilized to compensate for the approximation error between uncertainly model and WFCMAC approach.

We assume that the error bound is a constant during the observation. However, in practical application it is difficult to determine. Therefore, a bound estimation is developed to estimate this error bound. The estimation of error bound is defined as follows:

$$\tilde{k}_f = k_f - \hat{k}_f. \tag{24}$$

Where $\hat{k}_f$ is the estimated value of k_f. The adaptive robust controller is designed to compensate for the effect of the approximation error is selected as follows:

$$\tau_r = -\hat{k}_f \mathrm{sgn}(r). \tag{25}$$

Substituting (23) into (4) and using (19), (21) and (25), then, the closed-loop filtered error dynamics becomes:

$$M\dot{r} = -\left(K_{\mathrm{v}} + V(q, \dot{q})\right) r + \tilde{W}^{\mathrm{T}} b(x) + \varepsilon - \hat{k}_f \mathrm{sgn}(r). \tag{26}$$

Theorem 1. *Consider an n-link robot manipulator represented (1). If the control law of RASOWFCM is designed as (23), and the weight update law of WFCMAC and the adaptive law of the error bound are designed as (7), then the stability of the proposed RASOWFCM system can be ensured*

$$\dot{\hat{W}} = F\, b(x)\, r^{\mathrm{T}}, \quad \dot{\tilde{k}}_f = -\dot{\hat{k}}_f = -\eta_r \parallel r \parallel. \tag{27}$$

Proof: Define a Lyapunov function candidate as

$$L(r(t), \tilde{W}) = \frac{1}{2} r^{\mathrm{T}} M r + \frac{1}{2} \mathrm{Tr}(\tilde{W}^{\mathrm{T}} F^{-1} \tilde{W}) + \frac{\tilde{k}_f^2}{2\eta_r}. \tag{28}$$

Where F, η_r are the learning rate for the WFCMAC memory contents, error bound, respectively. By differentiating (28) with respect to time and using (26), (27), and using properties of the robot dynamics are introduced in Section 2, we can obtain.

$$\begin{aligned}
\dot{L} &= r^{\mathrm{T}} M\dot{r} + \tfrac{1}{2} r^{\mathrm{T}} \dot{M} r + \mathrm{Tr}(\dot{\tilde{W}}^{\mathrm{T}} F^{-1} \tilde{W}) + \tfrac{\tilde{k}_f \dot{\tilde{k}}_f}{\eta_r} \\
&= -r^{\mathrm{T}} K_{\mathrm{v}} r + \tfrac{1}{2} r^{\mathrm{T}} (\dot{M} - 2V) r + \mathrm{Tr}\tilde{W}(F^{-1} \dot{\tilde{W}}^{T} + b r^{\mathrm{T}}) + r^{\mathrm{T}} (\varepsilon - \hat{k}_f) \mathrm{sgn}(r) + \tfrac{\tilde{k}_f \dot{\tilde{k}}_f}{\eta_r} \\
&= -r^{\mathrm{T}} K_{\mathrm{v}} r + \tfrac{1}{2} r^{\mathrm{T}} (\dot{M} - 2V) r + \mathrm{Tr}\tilde{W}(F^{-1} \dot{\tilde{W}}^{T} + b r^{\mathrm{T}}) + \varepsilon\, r^{\mathrm{T}} - \hat{k}_f \parallel r \parallel - (k_f - \hat{k}_f) \parallel r \parallel \\
&= -r^{\mathrm{T}} K_{\mathrm{v}} r + \varepsilon\, r^{\mathrm{T}} - k_f \parallel r \parallel \\
&\leqslant -r^{\mathrm{T}} K_{\mathrm{v}} r + \parallel \varepsilon \parallel \parallel r \parallel - k_f \parallel r \parallel = -r^{\mathrm{T}} K_{\mathrm{v}} r - (k_f - \parallel \varepsilon \parallel) \parallel r \parallel \leqslant -r^{\mathrm{T}} K_{\mathrm{v}} r \leqslant 0.
\end{aligned} \tag{29}$$

Since $\dot{L}(r(t), \tilde{W}, \tilde{k}_f)$ is a negative semi-definite function, i.e. $\dot{L}(r(t), \tilde{W}(t), \tilde{k}_f(t)) \leqslant \dot{L}(r(0), \tilde{W}(0), \tilde{k}_f(0))$, it implies that r, $\tilde{W}$ and $\tilde{k}_f$ is bounded functions. Let function $h \equiv (k_f - \|\varepsilon\|)r \leqslant (k_f - \|\varepsilon\|)\,\|r\| \leqslant -\dot{L}(r, \tilde{W}, \tilde{k}_f)$ and integrate function $h(t)$ with respect to time

Because $\dot{L}(r(0), \tilde{W}(0), \tilde{k}_f(0))$ is a bounded function, and $\dot{L}(r(t),\, \tilde{W}(t),\, \tilde{k}_f(t))$ is a nonincreasing and bounded function, the following result can be achieved:

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 r will be converging to zero as time tends to infinity. □

5 Simulation and experimental results

5.1 Simulation results

A three-link deicing 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

$$\begin{cases} M\left(q\right) = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}, V(\dot{q}) = \begin{bmatrix} V_{11} & V_{12} & V_{13} \\ V_{21} & V_{22} & V_{23} \\ V_{31} & V_{32} & V_{33} \end{bmatrix}, \\ V_{11} = -8m_2l_1l_2c_1s_1\dot{q}_1 + (-1/2m_2s_2c_2l_2^2 + m_3(-2s_2c_2l_2^2 - 2s_2l_1l_2)\dot{q}_2, \\ M_{11} = 9/4m_1l_1 + m(1/4c_2l_2 + l_1^2 + l_2l_1(c_1^2 - s_1^2)) + m_3(c_2l_2^2 + l_2^2 + 2c_2l_1l_2), \\ M_{22} = 1/4m_2l_2^2 + m_3l_2^2 + 4/3m_1l_1^2, M_{22} = 1/4m_2l_2^2 + m_3l_2^2 + 4/3m_1l_1^2, \\ M_{33} = m_3, M_{12} = M_{13} = M_{21} = M_{31} = 0, \\ V_{21} = (-1/2m_2s_2c_2l_2^2 + m_3(-2s_2c_2l_2^2 - 2s_2l_1l_2)\dot{q}_1, V_{22} = -m_3s_2l_2\dot{d}_3, \\ V_{23} = -2m_3s_2l_2\dot{q}_2, V_{32} = -m_3s_2l_2\dot{q}_2, V_{12} = V_{13} = V_{31} = V_{33} = 0, \\ G(q) = \begin{bmatrix} (1/2c_1c_2l_2 + c_1l_1)m_2g \\ (-1/2s_1s_2l_2m_2 + c_2l_2m_3)g \\ m_3g \end{bmatrix}. \end{cases} \tag{30}$$

Where $q \in \mathbb{R}^3$ and the shorthand notations $c_1 = \cos(q_1)$, $c_2 = \cos(q_2)$, and $c_1 = \sin(q_1)$, $c_2 = \sin(q_2)$ are used.

For convenience of simulation, nominal parameters of the robotic system are given as $m_1 = 3\,(\text{kg})$, $m_2 = 2\,(\text{kg})$, $m_3 = 2.5\,(\text{kg})$, $l_1 = 0.14\ (\text{ m})$, $l_2 = 0.32\ (\text{ m})$ and and the initial conditions $q_1\left(0\right) = 1$, $q_2\left(0\right) = 0$, $d_3\left(0\right) = 0$, $\dot{q}_1\left(0\right) = 0$, $\dot{q}_2\left(0\right) = 0$, $\dot{d}_3\left(0\right) = 0$. The desired reference trajectories are $q_{d1}(t) = \sin(t)$, $q_{d2}(t) = \cos(t)$, $d_{d2}(t) = \cos(t)$, respectively.

For recording respective control performance, the mean-square-error of the position-tracking response is defined as:

$$\text{MSE}_i = \frac{1}{T}\sum_{j=1}^{\text{T}} \left[\, q_{di}(j) - q_i(j)\,\right]^2, \quad i = 1, 2, 3. \tag{31}$$

Where T is the total sampling instant, and q_i and q_{di} are the elements in the vector q_i and q_{di}. In this paper, the numerical simulation results carried out by using Matlab software.

For the proposed RASOWFCM control system, the parameters are chosen in the following:

$$\Lambda = \text{diag}\left\{10, 10, 10\right\}, K_{\text{v}} = \text{diag}\left\{50, 50, 50\right\}, F = 15, \eta_r = 0.2. \tag{32}$$

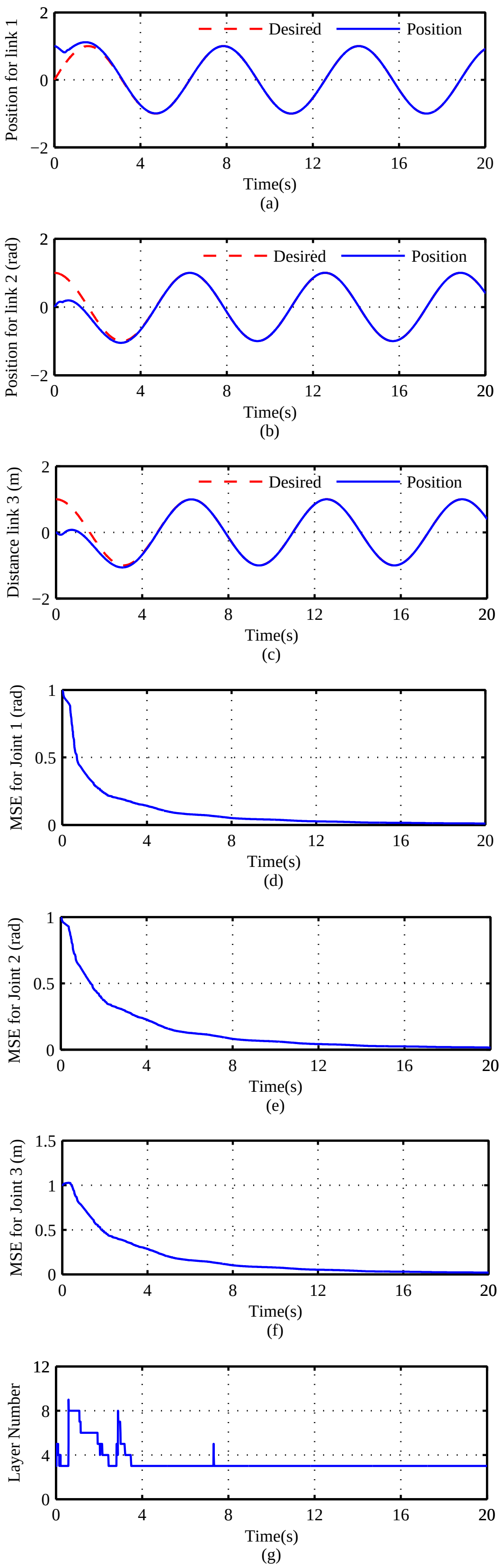

Figure 5: Simulated position responses, MSEs and layer number of the RA-SOWFCM control system at joints.

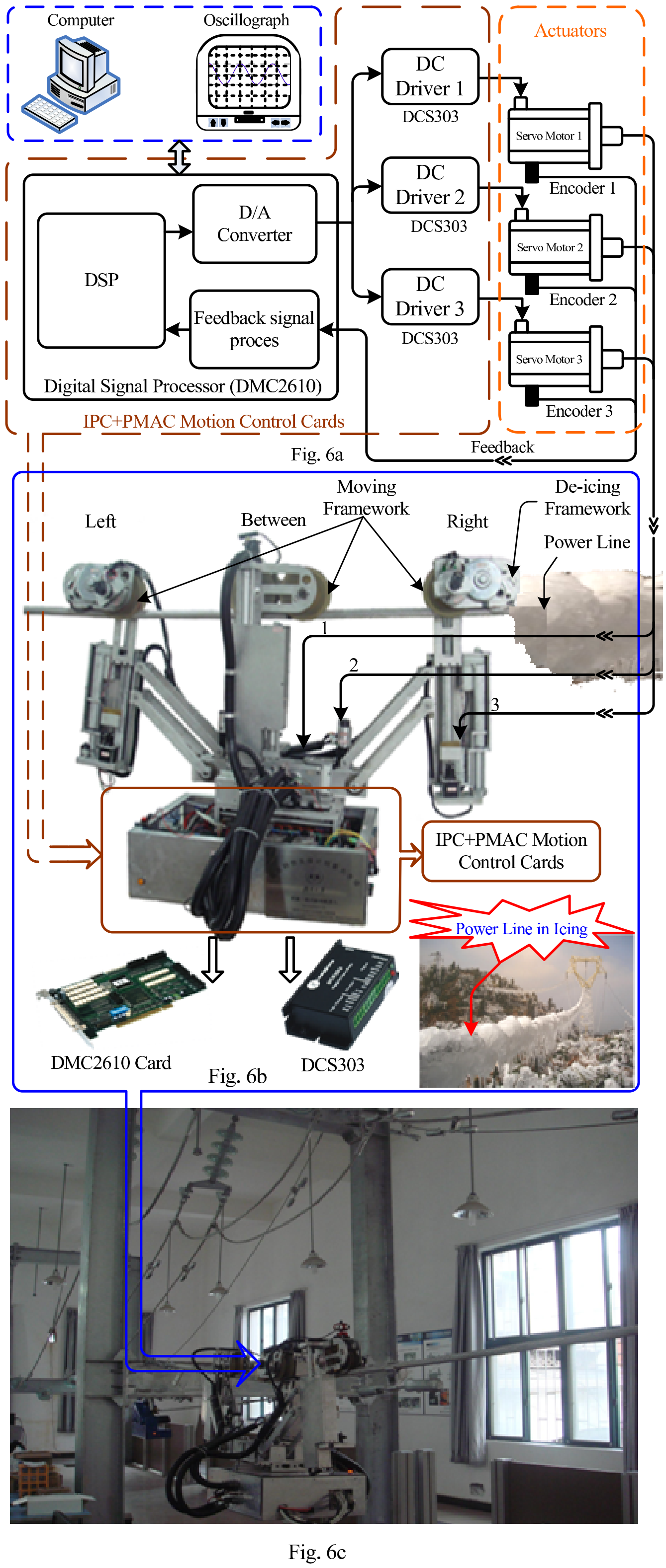

Figure 6: IIPC-based deicing robot position control system a) Block diagram of three-link deicing robot manipulator c) image of practical control system. control system, b) Image of practical control system.

and the initial values of system parameters are given as $n_k = 2$, $\hat{k}_f = 1$, the inputs of WFC-MAC, the translation and dilation of wavelet functions are selected to cover the input space $\{[-1 \ \ 1][-1 \ \ 1][-1 \ \ 1]\}$. The threshold value of K_g is set as 0.1; K_c is set as 0.01. The simulation results of the proposed RASOWFCM system, the responses of joint position MSE and layer number are depicted in Fig.5(a), (b), (c); (d), (e), (f) and (g), respectively. The simulation results show that the proposed RASOWFCM control system can achieve a favorable tracking performance with self-organizing of WFCMAC, and the layers of WFCMAC will be converge to three layers.

5.2 Experimental results

An image of a practical experiment control system for deicing robot consists of three manipulators and is shown in Fig 6(b). The left and right manipulators have three links with two revolute joints and a prismatic joint. End-effectors of each manipulator have attached the motion structure to move the deicing robot on the power line and the deicing device. Under normal operating conditions, the left and right manipulators are only in operation. The middle manipulator has only two joints with a revolute joint and a prismatic joint. It only works when the deicing robot voids obstacles on the power line. In general, the operation of deicing robot is very complex. In this paper, we consider only the three-link deicing robot manipulator for proposed methodologies while the other manipulator is the same. The hardware block diagram of the control system is implemented to verify the effectiveness of the proposed methodologies and is shown in Fig.6(a). Each joint of manipulator is derived by the "EC**" type MAXON DC servo motors, and each 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 model DMC2610 has a PCI interface connected to the IPC. The DMC2610 implements and excutes the proposed program in the real time. Considering that the control sampling rate $T_s = 1ms$ is too demanding for the hardware implementation, $T_s = 10ms$ is thus chosen here. The experimental parameters of the proposed RASOWFCM control system are selected as simulation: In this section, the control objective is to control each joint angle of a three-link deicing robot manipulator to move periodically for a periodic step commands and initial conditions of the system are given as $q_1\,(0) = 0\,(\text{rad})$, $q_2\,(0) = 0(\text{rad})$, $d_3\,(0) = 0\ (\text{m})$. Finally, the experimental position and tracking error response results of the proposed RASOWFCM control system are depicted in Fig.7 (a), (b) and (c). According to these experiment and simulation results of proposed RASOWFCM control system due to sinusoidal and periodic step reference commands indicate that the high-accuracy tracking position responses can be achieved by using the proposed RASOWFCM control system for difference reference commands under a wide range of external disturbance.

6 Conclusion

In this paper, a RASOWFCM control system is proposed to control the joint position of a three-link deicing robot manipulator. In the proposed RASOWFCM system, dynamical system is completely unknown in the control process. The adaptive tuning laws of WFCMAC parameters and error estimation are derived according to Lyapunov function so that a stability of the

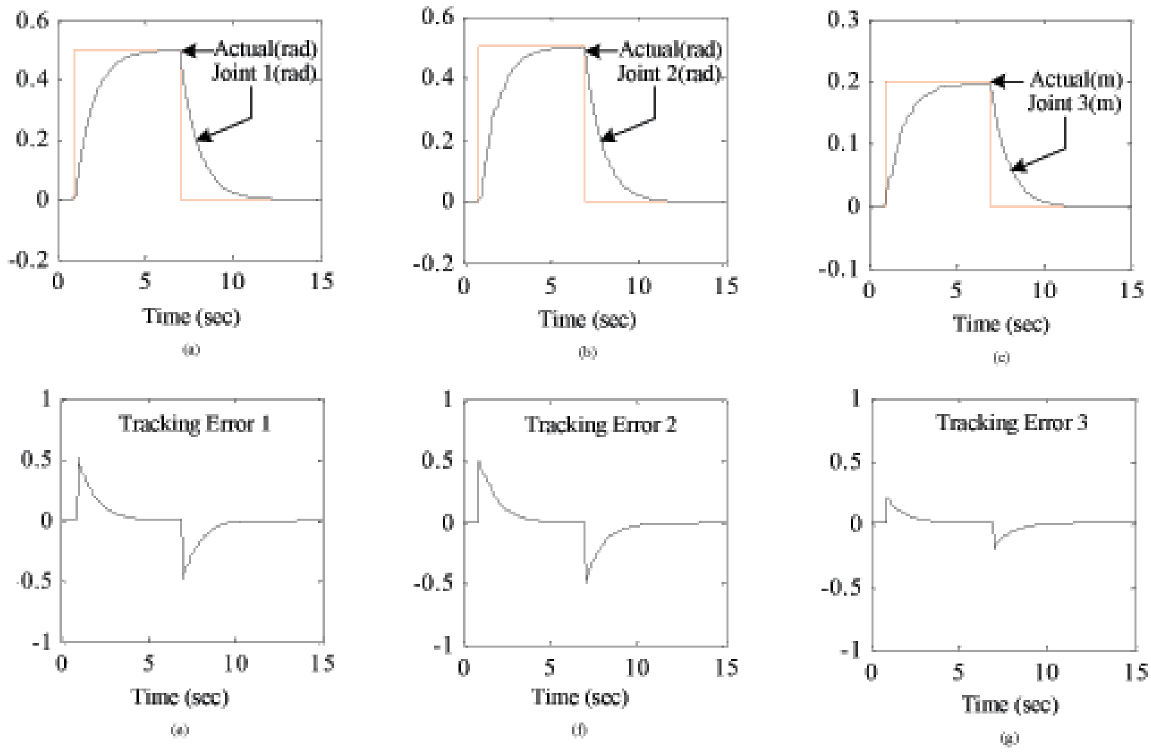

Figure 7: Experimental position responses, tracking errors of the proposed RASOWFCM control system at joints 1, 2 and 3

system can be guaranteed. This paper has not only successfully developed the RASOWFCM control system for a three-link deicing robot manipulator based on the novel WFCMAC requires low memory with online parameter tuning algorithm, but also provided novel architecture and mathematical model of deicing robot, which is verified the effectiveness through the practical application. Simulation and experimental results of the proposed RASOWFCM system can achieve favorable tracking performance for the proposed three-link deicing robot manipulator.

Bibliography

[1] M. Baban, C.F. Baban, C. Bungau, G. Dragomir, R.M. Pancu (2014); Estimation of the Technical State of Automotive Disc Brakes Using Fuzzy Logic, *International Journal of Computers Communications & Control*, 9(5): 531-538.

[2] C.R. Costea, H.M. Silaghi, D. Zmaranda, M.A. Silaghi (2015); Control System Architecture for a Cement Mill Based on Fuzzy Logic, *International Journal of Computers Communications & Control*, 10(2): 165-173.

[3] Y. Zou, Y. N. Wang, X. Z. Liu (2010); Neural network robust H_∞ tracking control strategy for robot manipulators. *Applied Mathematical Modelling*, 34(7): 1823-1838.

[4] Y. Feng, W. Yao-nan, Y. Yi-min (2012); Inverse Kinematics Solution for Robot Manipulator based on Neural Network under Joint Subspace, *International Journal of Computers Communications & Control*, 7(3): 459-472.

[5] C. Zhu, Y. F. Fang (2007); Adaptive control of parallel manipulators via fuzzy-neural network algorithm. *Journal Control Theory & Application*, 5(3): 295-300.

[6] T. Ngo, Y. Wang, T.L. Mai, M.H. Nguyen, J. Chen (2012); Robust Adaptive Neural-Fuzzy Network Tracking Control for Robot Manipulator. *International Journal of Computers Communications & Control*, 7(2): 341-352.

[7] C. F. Hsu, C. M. Lin, T. T. Lee (2006); Wavelet adaptive backstepping control for a class of nonlinear Systems, *IEEE Transcation Neural Network*, 17(5): 1175-1183.

[8] C. H. Lu (2009); Design and application of stable predictive controller using recurrent wavelet neural networks, *IEEE Transactions On Industrial Electronics*, 56(9): 733 – 3742.

[9] J. S. Albus (1975); A new approach to manipulator control: The cerebellar model articulation controller (CMAC), *J. Dyn. Syst. Meas. Control*, 97(3): 220 – 227.

[10] S. Jagannathan, S. Commuri, F. L. Lewis (1998); Feedback linearization using CMAC neural networks, *Automatica*, 34(3): 547 – 557.

[11] Y. H. Kim, F. L. Lewis (2000); Optimal design of CMAC neural-network controller for robot manipulators, *IEEE Transcation System Man Cybernation C, Application Revision*, 30(1): 22 – 31.

[12] C. T. Chiang, C. S. Lin (1996); CMAC with general basis functions, *Journal of Neural Network*, 9(7): 1199 – 1211.

[13] H. C. Lu, C. Y. Chuang, M. F. Yeh (2009); Design of hybrid adaptive CMAC with supervisory controller for a class of nonlinear system. *Neurocomputing*, 72(7-9): 1920 – 1933.

[14] Y. F. Peng, C. M. Lin (2004); Intelligent hybrid control for uncertain nonlinear systems using a recurrent cerebellar model articulation controller. *IEEE Proceedings Control Theory Application*, 151(5): 589 – 600.

[15] Y.C. Hsu, G. Chen, H.X. Li (2001); A Fuzzy adaptive Variable Structure Controller with Application to Robot Manipulators, *IEEE Transcation System Man Cybernation*, 31(3): 331 – 340.

[16] H. K. Khalil (1996); *Nonlinear systems*, Englewood Cliffs, NJ: Prentice-Hall, 1996.

[17] C. M. Lin, T. Y. Chen (2009); Self-organizing CMAC control for a class of MIMO uncertain nonlinear systems, *IEEE Neural Nets*, 20(9): 1377 – 1384.

[18] C.T. Lin, C. S. George Lee (1996); *Neural fuzzy systems*, Englewood Cliffs, NJ: Prentice-Hall.