

Direct Adaptive Control Based on Improved RBF Neural Network for Omni-directional Mobile Robot

Fan Jinhui

College of Electronic Information and Control
Engineering
Beijing University of Technology
Beijing, China;
School of Electrical Engineering
Hebei University of Science and Technology
Shijiazhuang, China
fanjinhui@sina.cn

Jia Songmin, Li Xiuzhi

College of Electronic Information and Control
Engineering
Beijing University of Technology
Beijing, China;
jiasongmin@sina.com

Abstract—A direct adaptive controller based on improved radial basis function (RBF) neural networks (NN) is proposed for an omni-directional mobile robot (OMR). The OMR is a multi-input and multi-output (MIMO), unmodeled and uncertain nonlinear system which is difficult to be modeled due to a large number of immeasurable and uncertain variables. To model the system exactly and increase the real-time performance, a novel direct adaptive control approach based on improved RBF-NN is designed to approximate the OMR, which needs no explicit knowledge of the uncertain nonlinear MIMO system. Besides the kinematics, the dynamics of the OMR are considered to perform tasks with heavy load transportations or high speed movements. A stable on-line adaptive law is derived and proved using Lyapunov stability theory. The proposed controller is applied the OMR trajectory tracking and shows excellent robustness and stability. The simulation results demonstrate the feasibility and validity of proposed scheme.

Keywords—direct adaptive controller; radial basis function neural network; omni-directional mobile robot; nonlinear and uncertain nonlinear system; Lyapunov stability theory

I. INTRODUCTION

It is well known that factors on unmodeled uncertainties and nonlinear disturbance widely exist in most practical control systems. The existence of those uncertain and nonlinear factors usually make the stability control more difficult than conventional certain and linear one. In recent years, a great deal of research progress has been achieved on nonlinear and uncertain SISO/MIMO feedback control systems by some researchers[1,2,3]. Meanwhile, many neural network-based adaptive control methods are studied and developed extensively[4,5,6,7].

The radial basis function (RBF) neural network (NN) is a local approximation network, and it has been proved that the RBF-NN can approximate any continuous function in a definite precision [8,9,10]. RBF-NN can be applied in many relative control fields. Lewis et al.[4] presented a multilayer neural network controller for a general serial-link rigid robot arms using a filtered error approach. Huang et al.[11] proposed an adaptive control method using RBF-NN for tracking control of mechanical systems in the presence of friction and periodic disturbances. Kim and Inman[12] proposed a NN-based learning control

scheme for the motion control of autonomous underwater vehicles. Wang et al.[6] developed a neural network-based robust adaptive control design scheme for nonlinear systems with an unknown nonlinear function and unmodeled dynamics. Boquete et al.[13] applied two recurrent RBF-NNs for the identification and control of a wheelchair. Chen et al.[14] discussed the stability of a genetic algorithm-based adaptive neural network controller for a nonlinear plant using the structure of a RBF-NN.

The main contribution of this paper is to provide a direct adaptive control method based on improved double-model structured RBF neural network for a nonlinear and unmodeled uncertain MIMO omni-directional mobile robot (OMR) system. The proposed method guarantees the stability, real-time and robustness of the nonlinear system with unmodeled uncertainties, random external disturbances simultaneously.

The organization of this paper is as follows. In section2, the problem statement is described on the basis of kinematics and dynamics of the OMR system. In section 3, improved adaptive RBF-NN controller is obtained. In section 4, analysis of stability is proved by Lyapunov theory. In section 5, simulation results are shown. Finally, the conclusion is provided in section 6.

II. PROBLEM STATEMENT

We consider dynamics of an OMR based on literature [15]. The rigid body model in the body-fixed reference frame can be represented by

$$M\dot{v} + C(v)v + D(v)v + g(q) + \tau_d = \tau \quad (1)$$

$$\dot{q} = J(q)v$$

where $v = [v_x, v_y, \omega]^T$, and $q = [x, y, \phi]^T$. Here, v represents the position and attitude vector of in the body-fixed reference frame, q denotes the position and attitude vector of OMR in the earth-fixed reference frame, and τ denotes the control forces and moments generated by motors on the OMR in the body-fixed reference frame. $J(q)$ denotes a transition matrix, by which the velocity vector in body-fixed reference frame can be transformed into the earth-fixed reference frame. M is a symmetric, positive definite inertia matrix including the mass of the

OMR and its user, $C(v)$ is the centripetal and Coriolis matrix, $D(v)$ is the friction matrix, $g(q)$ is the gravitational vector of forces and moments, and τ_d represents the forces and moments of nonlinear random disturbances. Here, we consider $g(q)=0$ since the movement of OMR is two-dimensional planar motion. Therefore, the equation of an OMR can be transformed into the earth-fixed reference frame and written as follows

$$M_q(q)\ddot{q} + C_q(q, v)\dot{q} + D_q(q, v)\dot{q} + \tau_{dq} = J^{-T}(q)\tau \quad (2)$$

where $M_q(q) = J^{-T}(q)MJ^{-1}(q)$,

$$C_q(q, v) = J^{-T}(q)[C(q, \dot{q}) - MJ^{-1}(q)\dot{J}(q)]J^{-1}(q) \quad ,$$

$$D_q(q, v) = J^{-T}(q)D(q, \dot{q})J^{-1}(q) \quad , \text{ and } \tau_{dq} = J^{-T}(q)\tau_d \quad .$$

Here, $M_q(q)$ is symmetric and positive definite, and $C_q(q, v)$ satisfies the skew symmetric relationship as $x^T[\dot{M}_q(q) - 2C_q(q, v)]x = 0$. The friction matrix $D_q(q, v)$ is positive definite. Figure 1 shows the coordinate system of OMR.

In (2), the friction matrix $D_q(q, v)$ and the forces and moments of disturbances τ_d are relevant to some random nonlinear factors, such as the ground condition, the load of OMR, the current posture of user's. As mentioned above, it is very difficult to build a conventional mathematical model of an OMR in practical applications. The frictions coefficients between the four wheels and ground are not only difficult to be obtained exactly, but also variable with the motion state of the OMR. Even if each element was measurable in (2), some unmodeled uncertainties would be still in existence for the disturbances from the circumstances. Therefore, the design of an adaptive controller is necessary for this class of nonlinear systems with unmodeled uncertainties and random disturbances.

III. DESIGN OF RBF-NN-BASED ADAPTIVE CONTROLLER

The RBF-NN neural networks can be used as a direct adaptive controller for a class of nonlinear dynamic systems[8], for which an explicit linear parametrization of the uncertainties in the dynamics is either unknown or impossible. In this study, the RBF-NN is used to approximate the nonlinear dynamics of OMR to a specified degree of accuracy. If an enough number of hidden layer neurons are employed, even though the OMR nonlinear dynamics is completely unknown.

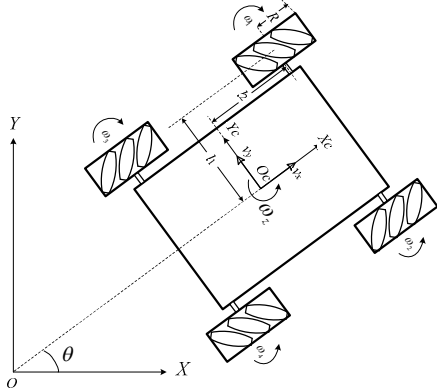


Figure 1. Coordinate system of OMR (planform)

A. RBF-NN Architecture

The conventional RBF-NN consists of three layers, which are input layer, hidden layer and output layer. The connections between layers are multiplied by weights of appropriate dimensions[16]. The RBF-NN structure is shown in Figure 2. In RBF-NN, we denote by $\|\cdot\|$ any proper vector norm, and we also denote the p-norm by $\|\cdot\|_p$. Each output of hidden layer can be represented by

$$\phi_i(X) = \exp\left(\frac{-\|X - C_i\|^2}{2\sigma_i^2}\right), \quad (i = 1, 2, \dots, k) \quad (3)$$

where $X = [x_1 \ x_2 \ \dots \ x_m]^T$ denotes the input vector, and m denotes the number of input layer neurons. Here, k denotes the number of hidden layer neurons with $i = 1, 2, \dots, k$, $C_i \in \mathbb{R}^{k \times 1}$ denotes the centre position of Gaussian function in hidden layer neurons, and σ_i denotes the width of Gaussian function in hidden layer neurons.

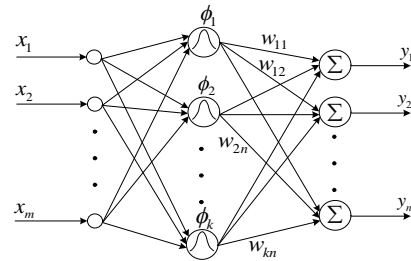


Figure 2. RBF neural network structure

The output of output-layer can be written as

$$y_j = \sum_{i=1}^k w_{ij}\phi_i(X), \quad (i = 1, 2, \dots, k; j = 1, 2, \dots, n) \quad (4)$$

where $j = 1, 2, \dots, n$ is the number of output-layer neurons, w_{ij} denotes the weights between hidden-layer neurons and output-layer neurons, and y_j denotes the output of the RBF-NN. Thus, the RBF-NN equation can be easily written in terms of vectors as

$$Y = W^T \Phi \quad (5)$$

where $Y = [y_1 \ y_2 \ \dots \ y_n]^T$, it is the output vector of RBF-NN, $W \in \mathbb{R}^{k \times n}$ is the weight vector between hidden-layer and output-layer, and $\Phi(X) = [\phi_1(X), \phi_2(X), \dots, \phi_k(X)]^T$ is the output vector of the hidden layer neurons.

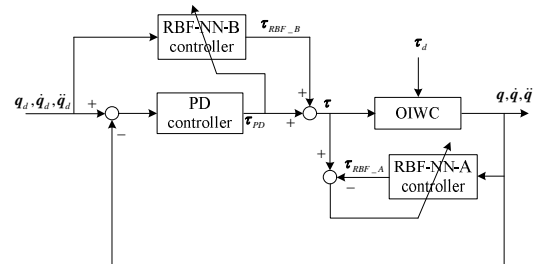


Figure 3. Schematic diagram of the nonlinear control system

B. Improved RBF-NN

Considering real-time performance of the online adaptive controller, the conventional RBF-NN is improved

on dynamic training algorithm. Direct adaptive control laws based on RBF-NN are derived as follows. The control structure with two RFB-NNs is shown in Figure 3.

In Figure 3, the RBF-NN-A is used to learn and approximate the inverse dynamic model to generate the compensated control signals for self-learning in offline mode. The dynamics of the OMR are obtained by RBF-NN-A using dynamic training algorithm. In off-line learning mode, the model learning is efficient because actual operation is unnecessary. After the RBF-NN-A is training offline, it can be trained to approximate the nonlinear and uncertainties online.

The RBF-NN-B is only a duplicate of RBF-NN-A, but as the controller is working, its weights will be adjusted in response to the adaptive law as discussed in the next subsection. The RBF-NN-B is mainly used to compensate for approximation errors and uncertain disturbance in the OMR system. Compared with conventional RBF-NN, the number of hidden layer neurons of RBF-NN-B is dynamic, since each neuron can be set in a sleeping state and be easily awakened if need be. In the dynamic weight vector adjustment process, the neurons of hidden layer are not cut but in dormancy. Considering few samples obtained at the beginning of learning period, a parallel proportion derivative (PD) controller is running for better dynamic performance in the initial learning stage online.

C. Controller Design based on Improved RBF-NN

In this section, our main goal is to design an adaptive controller that can guarantee the boundedness of all variables in the closed-loop control system. At the same time, a desired trajectory tracking can be assured by the output of the RBF-NN. Therefore, some assumptions are provided throughout the paper as follows.

Assumption 1: During motion process of the OMR, the desired position and attitude are known bounded functions, satisfying the condition as

$$\left\| \begin{bmatrix} \mathbf{q}_d^T & \dot{\mathbf{q}}_d^T & \ddot{\mathbf{q}}_d^T \end{bmatrix}^T \right\| \leq q_B \quad (6)$$

where \mathbf{q}_d denotes the desired position and attitude of the OMR in the earth-fixed reference frame, and we assume $\dot{\mathbf{q}}_d$ and $\ddot{\mathbf{q}}_d$, the time derivatives of \mathbf{q}_d , can be obtained from a trajectory planner. Here, q_B is a positive constant.

The filtered errors can be written as

$$\mathbf{s} = \ddot{\mathbf{q}} + \lambda \dot{\mathbf{q}} \quad (7)$$

where $\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_d - \ddot{\mathbf{q}}$, $\dot{\mathbf{q}} = \dot{\mathbf{q}}_d - \dot{\mathbf{q}}$, and λ is a positive constant.

Let us define $\dot{\mathbf{q}}_r$ as vector of the virtual reference trajectory in the earth-fixed reference frame, and (8) is satisfied as

$$\dot{\mathbf{q}}_r = \mathbf{J}(\mathbf{q})\mathbf{v}_r \quad (8)$$

where $\mathbf{v}_r = \mathbf{J}^{-1}(\mathbf{q})[\ddot{\mathbf{q}}_r - \dot{\mathbf{J}}(\mathbf{q})\mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{q}}_r]$. Therefore, (7) can be rewritten as

$$\mathbf{s} = \dot{\mathbf{q}}_r - \dot{\mathbf{q}} \quad (9)$$

The first deviation of (9) can be obtained as

$$\dot{\mathbf{s}} = \ddot{\mathbf{q}}_r - \ddot{\mathbf{q}} \Rightarrow \dot{\mathbf{q}}_r = \dot{\mathbf{q}}_d + \lambda \dot{\mathbf{q}} \quad (10)$$

Considering (2) and (9), (2) is easily rewritten as

$$\mathbf{M}_q \ddot{\mathbf{q}}_r + \mathbf{C}_q \dot{\mathbf{q}}_r + \mathbf{D}_q \dot{\mathbf{q}}_r = \mathbf{J}^{-T}(\boldsymbol{\tau} - \boldsymbol{\tau}_d) \quad (11)$$

With (1) and (11), we can obtain

$$\mathbf{M}_q \ddot{\mathbf{q}}_r + \mathbf{C}_q \dot{\mathbf{q}}_r + \mathbf{D}_q \dot{\mathbf{q}}_r = \mathbf{J}^{-T}(\mathbf{M}\dot{\mathbf{v}}_r + \mathbf{C}\mathbf{v}_r + \mathbf{D}\mathbf{v}_r) \quad (12)$$

From (10) and (12), taking the time derivative of \mathbf{s} , we can write the dynamics in terms of \mathbf{s} as follows.

$$\begin{aligned} \mathbf{M}_q \dot{\mathbf{s}} &= \mathbf{M}_q(\ddot{\mathbf{q}}_r - \ddot{\mathbf{q}}) = \mathbf{M}_q \ddot{\mathbf{q}}_r - \mathbf{M}_q \ddot{\mathbf{q}} \\ &= \mathbf{J}^{-T}(\mathbf{M}\dot{\mathbf{v}}_r + \mathbf{C}\mathbf{v}_r + \mathbf{D}\mathbf{v}_r + \boldsymbol{\tau}_d - \boldsymbol{\tau}) - (\mathbf{C}_q + \mathbf{D}_q)\mathbf{s} \end{aligned} \quad (13)$$

Because the accurate model is impossible to be obtained for unmodeled and certain factors as mentioned above, the (13) cannot be applied in actual system. Let us define function $f(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q})$ as the output of the proposed improved RBF-NN, and this function is used to approximate the real dynamics of the OMR to a specified degree of accuracy. Since the unstructured or unmodeled dynamics usually cannot be described exactly, we define the function as follows.

$$f(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) = \mathbf{W}^T \boldsymbol{\Phi} + \boldsymbol{\varepsilon} \quad (14)$$

where nonlinear function is defined as

$$f(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) = \mathbf{M}\dot{\mathbf{v}}_r + \mathbf{C}\mathbf{v}_r + \mathbf{D}\mathbf{v}_r \quad (15)$$

here, \mathbf{W} , $\boldsymbol{\Phi}$ are the same as defined above.

Assumption 2: For the designed RBF-NN above, the weight vector \mathbf{W} is bounded as $\|\mathbf{W}\|_F \leq W_{\max}$ with $W_{\max} > 0$, where $\|\mathbf{W}\|_F^2 = \text{tr}(\mathbf{W}^T \mathbf{W}) = \sum_{i,j} w_{ij}^2$, $\|\cdot\|_F$ denotes the Frobenius norm, and $\text{tr}(\cdot)$ is the trace operation of a matrix or a vector.

Assumption 3: The RBF-NN approximation errors $\boldsymbol{\varepsilon}$ is bounded as $\|\boldsymbol{\varepsilon}\| \leq \varepsilon_N$, where ε_N is a positive constant.

Because the RBF-NN is used to approximate the OMR dynamics with high precision, the system error will be small enough. Therefore, we have

$$\hat{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) \approx f(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) \quad (16)$$

where $\hat{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q})$ is the approximation of the OMR dynamics. In this paper, we choose a control input of the RBF-NN as

$$\boldsymbol{\tau} = \hat{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) + \mathbf{J}^T \mathbf{K}_d \mathbf{s} + \boldsymbol{\alpha} \quad (17)$$

where $\mathbf{K}_d = \mathbf{K}_d^T > 0$, and it denotes the control gain matrix. Here, $\boldsymbol{\alpha}$ denotes the robustness control term, which is used to enhance the robustness of the RBF-NN controller when external disturbances or approximation errors exist. We define

$$\hat{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) = \hat{\mathbf{W}}^T \boldsymbol{\Phi} \quad (18)$$

here, $\hat{\mathbf{W}}$ is the approximation of weight vector between hidden layer and output layer of RBF-NN, and the time derivative of $\hat{\mathbf{W}}$ can be written as

$$\dot{\hat{\mathbf{W}}} = \boldsymbol{\Gamma} \boldsymbol{\Phi} (\mathbf{J}^{-1} \mathbf{s})^T - \zeta \boldsymbol{\Gamma} \|\mathbf{s}\| \hat{\mathbf{W}} \quad (19)$$

where $\boldsymbol{\Gamma}$ is a positive constant matrix with $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^T > 0$, and ζ is a constant with $\zeta > 0$. Taking a control input as (17), the closed-loop system described in (13) can be rewritten as

$$\mathbf{M}_q \dot{\mathbf{s}} = \mathbf{J}^{-T}[\tilde{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) + \boldsymbol{\varepsilon} + \boldsymbol{\tau}_d - \boldsymbol{\alpha}] - \mathbf{C}_q \mathbf{s} - (\mathbf{K}_d + \mathbf{D}_q)\mathbf{s} \quad (20)$$

where $\tilde{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q})$ is the difference between the practical dynamics and the approximation one as

$$\tilde{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) = \mathbf{W}^T \boldsymbol{\Phi} - \hat{f}(\dot{\mathbf{v}}_r, \mathbf{v}_r, \mathbf{v}, \mathbf{q}) = \tilde{\mathbf{W}}^T \boldsymbol{\Phi} \quad (21)$$

where the weight estimated error is defined as

$$\tilde{\mathbf{W}} = \mathbf{W} - \hat{\mathbf{W}} \quad (22)$$

IV. ANALYSIS OF SYSTEM STABILITY

Theorem 1: If Assumptions 1-3 are satisfied, the control input satisfies (17), the adaptive control law of RBF-NN weight vector is (19), and the robustness control term is defined as

$$\boldsymbol{\alpha} = (\varepsilon_N + d_B) \frac{\mathbf{J}^{-1} \mathbf{s}}{\|\mathbf{J}^{-1} \mathbf{s}\|} \quad (23)$$

then both the filtered tracking errors $s(t)$ and the weight estimated errors $\tilde{\mathbf{W}}$ of the closed-loop system are uniformly ultimately bounded stable.

Proof: The Lyapunov function is selected as

$$\mathbf{V} = \frac{1}{2} \mathbf{s}^T \mathbf{M}_q \mathbf{s} + \frac{1}{2} \text{tr} \{ \tilde{\mathbf{W}}^T \boldsymbol{\Gamma}^{-1} \tilde{\mathbf{W}} \} \quad (24)$$

Taking derivative along (13) and (17), we have

$$\dot{\mathbf{V}} = \frac{1}{2} (\dot{\mathbf{s}}^T \mathbf{M}_q \mathbf{s} + \mathbf{s}^T \dot{\mathbf{M}}_q \mathbf{s} + \mathbf{s}^T \dot{\mathbf{M}}_q \mathbf{s}) + \text{tr} \{ \tilde{\mathbf{W}}^T \boldsymbol{\Gamma}^{-1} \dot{\tilde{\mathbf{W}}} \} \quad (25)$$

with considering that $\dot{\mathbf{s}}^T \mathbf{M}_q \mathbf{s} = \mathbf{s}^T \dot{\mathbf{M}}_q \mathbf{s}$, (13), (17), (20), (21), and with using diagonal matrix $\mathbf{X}^T [\dot{\mathbf{M}}_q(q) - 2\mathbf{C}_q(q, \dot{q})] \mathbf{X} = 0$, (25) can be rewritten as

$$\begin{aligned} \dot{\mathbf{V}} &= \mathbf{s}^T (\mathbf{M}_q \dot{\mathbf{s}} + \mathbf{C}_q \mathbf{s}) + \text{tr} \{ \tilde{\mathbf{W}}^T \boldsymbol{\Gamma}^{-1} \dot{\tilde{\mathbf{W}}} \} \\ &= -\mathbf{s}^T (\mathbf{D}_q + \mathbf{K}_d) \mathbf{s} + (\mathbf{J}^{-1} \mathbf{s})^T [\tilde{\mathbf{W}}^T \boldsymbol{\Phi} + \boldsymbol{\varepsilon} + \boldsymbol{\tau}_d - \boldsymbol{\alpha}] + \\ &\quad \text{tr} \{ \tilde{\mathbf{W}}^T \boldsymbol{\Gamma}^{-1} \dot{\tilde{\mathbf{W}}} \} \end{aligned} \quad (26)$$

Using (23), we can rewrite (26) as

$$\begin{aligned} \dot{\mathbf{V}} &\leq -\mathbf{s}^T (\mathbf{D}_q + \mathbf{K}_d) \mathbf{s} + \text{tr} \{ \tilde{\mathbf{W}}^T [\boldsymbol{\Phi} (\mathbf{J}^{-1} \mathbf{s})^T + \boldsymbol{\Gamma}^{-1} \dot{\tilde{\mathbf{W}}}] \} + \\ &\quad (\mathbf{J}^{-1} \mathbf{s})^T [\boldsymbol{\varepsilon} + \boldsymbol{\tau}_d - \boldsymbol{\alpha}] \\ &\leq -\mathbf{s}^T (\mathbf{D}_q + \mathbf{K}_d) \mathbf{s} + \zeta \|\mathbf{s}\| \text{tr} \{ \tilde{\mathbf{W}}^T (\mathbf{W} - \tilde{\mathbf{W}}) \} + \\ &\quad (\mathbf{J}^{-1} \mathbf{s})^T [\boldsymbol{\varepsilon} + \boldsymbol{\tau}_d - \boldsymbol{\alpha}] \end{aligned} \quad (27)$$

Using (23), we have

$$\begin{aligned} \dot{\mathbf{V}} &\leq -\mathbf{s}^T (\mathbf{D}_q + \mathbf{K}_d) \mathbf{s} + \zeta \|\mathbf{s}\| \text{tr} \{ \tilde{\mathbf{W}}^T (\mathbf{W} - \tilde{\mathbf{W}}) \} + \\ &\quad (\mathbf{J}^{-1} \mathbf{s})^T [\boldsymbol{\varepsilon} + \boldsymbol{\tau}_d] - (\mathbf{J}^{-1} \mathbf{s})^T (\varepsilon_N + d_B) \frac{\mathbf{J}^{-1} \mathbf{s}}{\|\mathbf{J}^{-1} \mathbf{s}\|} \\ &= -\mathbf{s}^T (\mathbf{D}_q + \mathbf{K}_d) \mathbf{s} + \zeta \|\mathbf{s}\| \text{tr} \{ \tilde{\mathbf{W}}^T (\mathbf{W} - \tilde{\mathbf{W}}) \} + \\ &\quad (\mathbf{J}^{-1} \mathbf{s})^T [\boldsymbol{\varepsilon} + \boldsymbol{\tau}_d] - \|\mathbf{J}^{-1} \mathbf{s}\| (\varepsilon_N + d_B) \end{aligned} \quad (28)$$

Then (28) can be rewritten as

$$\begin{aligned} \dot{\mathbf{V}} &\leq -(\mathbf{D}_{q\min} + \mathbf{K}_{d\min}) \|\mathbf{s}\|^2 + \zeta \|\mathbf{s}\| \|\tilde{\mathbf{W}}\|_F (W_{\max} - \|\tilde{\mathbf{W}}\|_F) \\ &= -\|\mathbf{s}\| \left[(\mathbf{D}_{q\min} + \mathbf{K}_{d\min}) \|\mathbf{s}\| + \zeta \|\tilde{\mathbf{W}}\|_F (\|\tilde{\mathbf{W}}\|_F - W_{\max}) \right] \end{aligned} \quad (29)$$

where $\mathbf{D}_{q\min}$ and $\mathbf{K}_{d\min}$ are the minimum eigenvalues of vector \mathbf{D}_q and vector \mathbf{K}_d , $\|\tilde{\mathbf{W}}\|_F$ denotes the Frobenius norm of $\tilde{\mathbf{W}}$, and W_{\max} is the maximum eigenvalue of the weight vector \mathbf{W} . Therefore, we can obtain (30) as

$$\left[(\mathbf{D}_{q\min} + \mathbf{K}_{d\min}) \|\mathbf{s}\| + \zeta \|\tilde{\mathbf{W}}\|_F (\|\tilde{\mathbf{W}}\|_F - W_{\max}) \right] \quad (30)$$

$$= \zeta (\|\tilde{\mathbf{W}}\|_F - W_{\max} / 2)^2 - \zeta W_{\max}^2 / 4 + (\mathbf{D}_{q\min} + \mathbf{K}_{d\min}) \|\mathbf{s}\|$$

If the closed-loop control system is stable, then (29)

should be negative definite, meaning that (30) must be positive definite. Therefore, we have

$$\|\mathbf{s}\| > \frac{\zeta W_{\max}^2 / 4}{(\mathbf{D}_{q\min} + \mathbf{K}_{d\min})} \quad \text{or} \quad \|\tilde{\mathbf{W}}\|_F > W_{\max} \quad (31)$$

From (31), it implies that $\dot{\mathbf{V}}$ decreases as $\|\mathbf{s}\|$ and $\|\tilde{\mathbf{W}}\|_F$ converges toward the boundary and remains in its vicinity. By choosing arbitrarily large \mathbf{K}_d , a small tracking error can be obtained.

Theorem 2: If Assumption 1 is satisfied, while the OMR dynamics RBF-NN approximation errors is $\varepsilon=0$, the external disturbances becomes zero, the control input satisfies (32), and the adaptive control law of RBF-NN weight vector is defined as (33)

$$\boldsymbol{\tau} = \hat{f}(\dot{\mathbf{v}}, \mathbf{v}, \mathbf{q}) + \mathbf{J}^T \mathbf{K}_d \mathbf{s} \quad (32)$$

$$\dot{\tilde{\mathbf{W}}} = \boldsymbol{\Gamma} \boldsymbol{\Phi} (\mathbf{J}^{-1} \mathbf{s})^T \quad (33)$$

then both the filtered tracking errors $s(t) \rightarrow 0$, and the weight estimated errors $\tilde{\mathbf{W}}$ of the closed-loop system are uniformly ultimately bounded stable.

Theorem 2 can be concluded and proved referencing to literature [17].

Theorem 3: If Assumptions 1-2 are satisfied, the control input satisfies (32), and the adaptive control law of RBF-NN weight vector satisfies (19), then both the filtered tracking errors $s(t)$ and the weight estimated errors $\tilde{\mathbf{W}}$ of the closed-loop system are uniformly ultimately bounded stable.

Referencing to the proof process of Theorem 1, we can obtain that the time derivative of Lyapunov function is negative as long as (34) is satisfied.

$$\begin{aligned} \|\mathbf{s}\| &> \frac{\zeta W_{\max}^2 / 4 + (\varepsilon_N + d_B)}{(\mathbf{D}_{q\min} + \mathbf{K}_{d\min})} \quad \text{or} \\ \|\tilde{\mathbf{W}}\|_F &> W_{\max} / 2 + \sqrt{W_{\max}^2 / 4 + (\varepsilon_N + d_B) / \zeta} \end{aligned} \quad (34)$$

By comparing (31) with (34), it is easy to find the steady-state tracking error decreased through adding the robustness control term or increasing control gain matrix.

V. SIMULATION RESULTS

We present the simulation results in order to verify that the proposed controller can compensate the unmodeled and uncertain disturbances. The desired velocities and positions are generated as a reference trajectory in our simulation study.

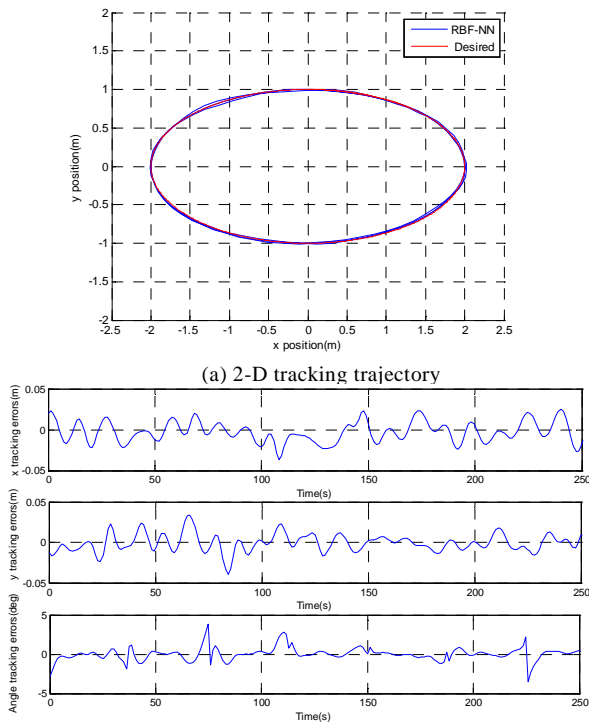
A. Parameters Initialization of Adaptive Controller

The trajectory tracking process is shown in Figure 4, when the OMR system is under the bounded and random disturbance. The number of RBF-NN hidden layer is 25, the maximum training time is 300, the final train error is 0.01, the initialized width of Gaussian function in hidden layer is 1, and the Gaussian function center of is initialized randomly. Considering the actual driving ability, the input gain matrix is $\mathbf{K}_d = [200, 200, 100]$, $\lambda = 0.5$, the adaptive learning coefficient of weight vector is $\boldsymbol{\Gamma} = \text{diag}\{[200, 200, 100]\}$, and the learning rate is set as 0.3. The desired tracking trajectory is defined as

$$\mathbf{x} = \sin(2\pi t/100), \quad \mathbf{y} = 2 \sin(2\pi t/100) \quad (35)$$

B. Results of Simulation

The simulation result shows the proposed adaptive controller is efficient under bounded and random disturbance in Figure 4.



(b) Tracking errors with random bounded disturbances
Figure 4. Simulation results with random bounded disturbances

The initial condition of the RBF-NN is the same as the above simulation. The disturbance is decreased to zero, and the tracking errors of positions in two directions can be obtained as shown in Figure 5. The proposed controller is also verified under zero disturbance condition.

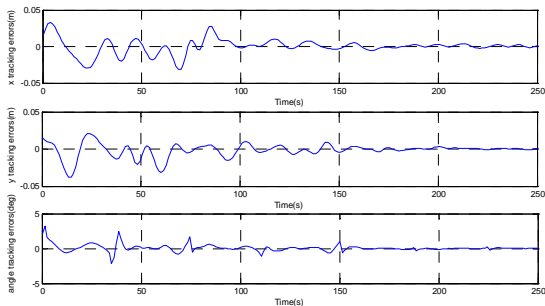


Figure 5. Tracking errors with damped disturbances

VI. CONCLUSIONS

In this paper, an adaptive RBF-NN-based controller is proposed for an OMR with uncertain model and nonlinear dynamic disturbances. The proposed RBF-NN is used to approximate the real OMR model. The control scheme can obtain significant trajectory tracking performance without the explicit prior mathematical model, and guarantees the control bounded in the closed-loop system simultaneously. In addition, the adaptive controller is robust and adaptive for nonlinear and uncertain system, which is verified by

the simulation results through being trained and applied online. The results demonstrate the feasibility of the proposed direct adaptive controller.

ACKNOWLEDGMENT

The research work is financially supported by Key Program of Beijing Natural Science Foundation (B), National Natural Science Foundations (61175087, 61105033) of China, Scientific Research Staring Foundation for the Returned Overseas Chinese Scholars, Ministry of Education of China, and General Aviation Platform Foundation of Hebei University of Science and Technology.

REFERENCES

- [1] CS. Chen. "Dynamic structure adaptive neural fuzzy control for MIMO uncertain nonlinear systems," *Information Sciences*, 2009, vol. 179(15): pp.2676-2688.
- [2] M. Chen, SS. Ge and B. Ren. "Adaptive tracking control of uncertain MIMO nonlinear systems with input constraints," *Automatica*, 2011, vol. 47(3): pp.452-465.
- [3] MC. Pai. "Robust discrete-time sliding mode control for multi-input uncertain time-delay systems," *Proceeding of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 2012, vol. 226(7): pp.927-935.
- [4] FL. Lewis, A. Yesildirek and K. Liu. "Multilayer neural-net robot controller with guaranteed tracking performance," *IEEE Transactions on Neural Networks*, 1996, vol. 7(2): pp.388-399.
- [5] TP. Zhang and SS. Ge. "Adaptive neural control of MIMO nonlinear state time-varying delay systems with unknown dead-zones and gain signs," *Automatic*, 2007, vol. 43(6): pp.1021-1033.
- [6] D. Wang, J. Huang, W. Lan and X. Li. "Neural network-based robust adaptive control of nonlinear systems with unmodeled dynamics," *Mathematics and Computers in Simulation*, 2009, vol.79(5): pp.1745-1753.
- [7] D. Vrabie and F. Lewis. "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Networks*, 2009, vol. 22(3): pp.237-246.
- [8] K. Hornik, M. Stinchcombe and H. White. "Multilayer feedforward networks are universal approximators," *Neural Networks*, 1989, vol. 2(5): pp.359-366.
- [9] J. Park and IW. Sandberg. "Universal approximation using Radial-Basis-Function networks," *Neural Computation*, 1991, vol. 3(2): pp.246-257.
- [10] M. Leshno and VY. Lin, A. Pinkus, et al. "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural Networks*, 1993, vol. 6(6): pp.861-867.
- [11] SN. Huang, KK. Tan and TH. Lee. "Adaptive motion control using neural network approximations," *Automatica*, 2002, vol. 38(2): pp.227-233.
- [12] MH. Kim and DJ. Inman. "Direct Adaptive Control of Underwater Vehicles using Neural Networks". *Journal of Vibration and Control*, 2003, vol. 9(5): pp.605-619.
- [13] L. Boquete, R. Barea, R. Garcia, M. Mazo and F.Espinosa. "Identification and control of a wheelchair using recurrent neural networks," *Engineering Applications of Artificial Intelligence*, 1999, vol. 12(4): pp.443-452.
- [14] CW. Chen, PC. Chen and WL. Chiang. "Modified intelligent genetic algorithm-based adaptive neural network control for uncertain structural systems," *Journal of Vibration and Control*, 2013, vol. 19(9): pp.1333-1347.
- [15] R. Fierro and FL. Lewis. "Control of a nonholonomic mobile robot using neural networks," *IEEE Transactions on Neural Networks*, 1998, vol. 9(4): pp.589-600.
- [16] FL. Lewis, A. Yesildirek and K. Liu. "Neural net robot controller: Structure and stability proofs," *Journal of Intelligent and Robotic System*, 1995, vol. 12(3): pp.277-299.
- [17] JE. Slotine and W. Li. "Applied nonlinear control Englewood Cliffs," NJ: Prentice Hall, 1991.