# Adaptive Sliding Mode Controller with Single Parameter Approximation for Four-wheel Omnidirectional Mobile Robots

Jianbin Wang, Jianping Chen and Sijie Ouyang

***Abstract*—**Due to the parametric uncertainties and the disturbances in dynamic model, the traditional control methods are not suit for the motion control of four-wheel omnidirectional mobile robot. An adaptive sliding mode controller is presented, the RBF neural network is applied to estimate the function of the parameter uncertainty and the disturbance. To suit the real-time control demand, single parameter estimation is used instead of the weights of RBF neural network, and then a Lyapunov function is exploited to prove the stability of the improved closed-loop system. The numerical simulations results show that the controlled system has better robust stability and ability of restraining disturbance.

***Keywords*—**four-wheel drive, omnidirectional mobile robot, sliding mode, single parameter approximation, neural network.

## I. INTRODUCTION

OMNIDIRECTIONAL mobile robot can move in any direction without changing any position and posture. The omnidirectional mobile robot has been widely applied to the human production and life practice in recent years with the special motion advantage. The control problems of motion and regulation have been extensively studied and attracted the interest of many control researchers in the field of omnidirectional mobile robotics [1].

In preliminary studies, it is common that the motion control problems of the omnidirectional mobile robotics are addressed taking into account the kinematic equations [2]. Considering, only its kinematic model, several control strategies have been proposed [3, 4]. However, these studies all suppose that the robot worked in an ideal environment or there was no uncertainty in the kinematic model. In fact, a good motion control for mobile robot with its kinematic model needs it to track the designed velocity perfectly, which is impossible in practical application. Then a number of contributions have been focused on the dynamic representation of the omnidirectional mobile robot [5]. A fuzzy controller was designed to realize tracking control for mobile robots [6]. In the same spirit, the dynamic model of the mobile robot is considered in order to study the slipping effects between the wheels of the vehicle and the working surface [7]. To deal with the parameter uncertainty and the disturbance existing in the dynamic model, sliding mode control approach and its improvements are presented [8]-[11]. A sliding mode control scheme was proposed to deal with the disturbance, and an adaptive controller was designed to solve the parameter uncertainty in path tracking problem [8]. In [10], based on the linearized system, an integral sliding mode control is designed for trajectory tracking control of an omnidirectional mobile robot. But it has inherent deficiency, which needs computing the upper boundedness of the system dynamics, and may cause high noise amplification. A robust neural network based sliding mode controller, which uses an NN to identify the unstructured system dynamics directly, is further proposed to overcome the disadvantages of the integral sliding mode control and reduce the online computing burden of conventional NN adaptive controllers [11]. However, it is not suit for the real-time control.

To deal with the model uncertainties and the disturbances in dynamic model of the four-wheel omnidirectional mobile robot, an adaptive sliding mode control with single parameter approximation is presented in this paper. And the RBF neural network is applied to estimate the function of the parameter uncertainty and the disturbance, single parameter estimation is used instead of the weights of RBF neural network to fit real-time control. And then, the stability of the closed-loop system and the convergence of the adapting process are strictly demonstrated by Lyapunov stable theory. Finally, a series of simulation results show that the control system has good tracking performance, better robust stability and ability of restraining disturbances.

## II. DYNAMIC MODEL OF FOUR-WHEEL OMNIDIRECTIONAL MOBILE ROBOT

### A. Kinematic Model

Four-wheel robots are one of the models of robots, which are

This work was supported by the National Natural Science Foundation of China(No.61402400), the Natural Science Foundation of Guangdong Province (No. 2016A030313015), the Key Basic Research Project form Department of Education of Guangdong Province, the Youth Innovation Foundation form Department of Education of Guangdong Province (No. 2016KQNCX175), the Science and Technology Innovation Guidance Project of Zhaoqing City (No. 201624030902), and the Science Foundation for the Youth Scholars of Zhaoqing University (No. 201623).

Jianbin Wang is with the School of Computer Science and Software, Zhaoqing University, Zhaoqing 5260615, Guangdong, China.

Jianping Chen is with the School of Computer Science and Software, Zhaoqing University, Zhaoqing 5260615, Guangdong, China. (corresponding author; e-mail: jpchen @zqu.edu.cn)

Sijie Ouyang is with the School of Mechanical and Automotive Engineering, Zhaoqing University, Zhaoqing 5260615, Guangdong, China.

used in many domains.They are omnidirectional with four wheels that have the ability of moving to any direction at any time (they are holonomic mobile robots, in other words). Fig. 1 shows the schematic of the four-wheel robot used in our laboratory.

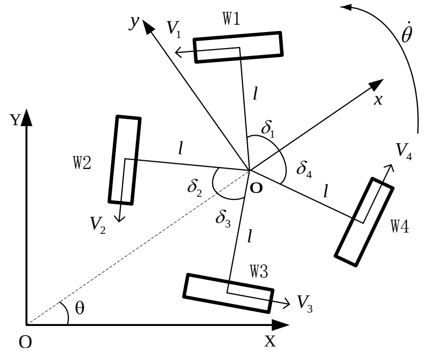



**Fig. 1.** Wheel placement of mobile robot

According to the geometric relationship of Fig. 1, the robot posture (position and orientation) in the robot coordinate frame is expressed as , and the robot posture can be presented in the world coordinate frame as , the relationship between and is as (1).

$$\mathbf{X}_m = \mathbf{R}(\theta)\cdot\mathbf{X}_w = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}\cdot\mathbf{X}_w \tag{1}$$

Where $XOY$ is the world coordinate frame for robot, $xoy$ is the robot own coordinate frame, denotes the moving direction of robot, $W_k$ $(k=1,2,3,4)$ denotes every wheel, $\delta_k$ denotes the angle between wheel and $x$ axis respectively, $V_k$ denotes the velocity of each wheel, its positive direction is anticlockwise, $l$ is the distance between the center of robot-body and that of each wheel.

The kinetic model of mobile robot can be constructed as (2).

$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{pmatrix} = r\cdot\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \begin{pmatrix} -\sin(\delta_1+\theta) & \cos(\delta_1+\theta) & l \\ -\sin(\delta_2-\theta) & -\cos(\delta_2-\theta) & l \\ \sin(\delta_2+\theta) & -\cos(\delta_2+\theta) & l \\ \sin(\delta_1-\theta) & \cos(\delta_1-\theta) & l \end{pmatrix}\begin{pmatrix} \dot{X}_w \\ \dot{Y}_w \\ \dot{\theta} \end{pmatrix} = g(\theta)\dot{\mathbf{X}}_w \tag{2}$$

Where $v_k$ denotes the linear velocity and $\omega_k$ is its angular velocity of each wheel, $r$ is the radius of each wheel.

### *B. Dynamic Model*

The force acting on the robot can be derived by applying the Newton's second law in the robot coordinate frame. Where $\mathbf{M} = diag\{m,m,J\}$, $m$ is the total mass of robot, $J$ is the total inertia for robot rotation, and $\mathbf{f} = (f_1\ f_2\ f_3\ f_4)^T$ is the tangential force generated by DC motors at each wheel.

$$\mathbf{M}\ddot{\mathbf{X}}_w = \mathbf{g}^{\mathrm{T}}(\theta)\cdot\mathbf{f} . \tag{3}$$

The dynamics of armature current of each DC motor can be described as (4). Where $V_s$ is battery voltage, $u\in[-1,1]$ is normalized control input, and $L_a$ is reactance of the motor, $R_a$ is armature resistance, $K_b$ is back-emf constant, $n$ is gear ratio and $\varphi$ is angular of the each wheel.

$$L_a\frac{di_a}{dt} + R_a i_a = V_s u - K_b n\dot{\varphi} . \tag{4}$$

Since the electrical time constant of the motor is very small compared to the mechanical time constant, we can neglect the inductance of the motor electric circuit and describe the generated torque of each motor $\tau$, with $K_t$ is the torque constant.

$$\tau = K_t n i_a = \frac{1}{R_a}K_t n(V_s u - K_b n\dot{\varphi}) . \tag{5}$$

Dynamic equation of velocity for each wheel is as (6). Where $J_w$ is the inertia at center of wheel about vertical axis and $F_v$ is the viscous friction factor in drive axis.

$$J_w\ddot{\varphi}_k + F_v\dot{\varphi}_k = \tau_k - rf_k . \tag{6}$$

Then the dynamic formulation produces the system representation,

$$\mathbf{D}\ddot{\mathbf{q}} + \mathbf{C}(\dot{q})\dot{\mathbf{q}} = \mathbf{B}\boldsymbol{\tau} . \tag{7}$$

Where $\mathbf{q} = \mathbf{X}_w$, $\boldsymbol{\tau} = (\tau_1\ \ \tau_2\ \tau_3\ \tau_4)^T$, $\mathbf{B} = \mathbf{R}(\theta)\mathbf{g}^{\mathrm{T}}(0)$,

$$\mathbf{D} = \begin{bmatrix} \frac{3J_w}{2r^2}+m & 0 & 0 \\ 0 & \frac{5J_w}{2r^2}+m & \frac{J_w}{r^2}(\sqrt{2}-\sqrt{3})l \\ 0 & \frac{J_w}{r^2}(\sqrt{2}-\sqrt{3})l & \frac{4J_w l^2}{r^2}+J \end{bmatrix},$$

$$\mathbf{C}(\dot{q}) = \begin{bmatrix} \frac{3}{2}-\frac{J_w\dot{\theta}}{2r^2}\sin 2\theta & \dot{\theta}(m+\frac{4J_w}{r^2}) & 0 \\ \dot{\theta}(m+\frac{4J_w}{r^2}) & \frac{5}{2}+\frac{J_w\dot{\theta}}{2r^2}\sin 2\theta & (\sqrt{3}-\sqrt{2})l \\ \frac{J_w\dot{\theta}}{r^2}(\sqrt{3}-\sqrt{2})l\cos\theta & (\sqrt{3}-\sqrt{2})l(\frac{J_w\dot{\theta}}{r^2}\sin\theta+1) & 4l^2 \end{bmatrix}$$

For describing conveniently and computing simply, the dynamic formula (7) can be rewritten as the standard form,

$$\mathbf{D}\ddot{\mathbf{q}} + \mathbf{C}(\dot{q})\dot{\mathbf{q}} + \mathbf{T}_d = \mathbf{T} . \tag{8}$$

Where the virtual control $\mathbf{T} = \mathbf{B}\boldsymbol{\tau}$, $\mathbf{T}_d$ is the parameter uncertainties and exogenous disturbances, which is bounded..

## III. Controller Design

### *A. Sliding Mode Control based on RBF NN*

Considering the tracking error of trajectory is

$$\mathbf{e}(t) = \mathbf{q}_d(t) - \mathbf{q}(t) . \tag{9}$$

When $t\to\infty$, $\mathbf{e}(t)\to 0$, $\dot{\mathbf{e}}(t)\to 0$ .

Supposing that there is a feasible smooth bounded reference trajectory $\mathbf{q}_d(t)$, for all trajectories $\mathbf{q}(t_0) = \mathbf{q}_d(t_0)$, according to the theory of sliding mode control, we have designed the sliding mode switching function as,

$$\mathbf{s} = \dot{\mathbf{e}} + \boldsymbol{\Lambda}\mathbf{e} \tag{10}$$

Where $\mathbf{\Lambda} = \mathbf{\Lambda}^{\mathrm{T}} > 0$, then we get

$$\dot{\mathbf{q}} = -\mathbf{s} + \dot{\mathbf{q}}_d + \mathbf{\Lambda e} \tag{11}$$

Plug equation (11) into the equation (8), we have

$$\begin{aligned}\mathbf{D}\dot{\mathbf{s}} &= \mathbf{D}(\ddot{\mathbf{q}}_d - \ddot{\mathbf{q}} + \mathbf{\Lambda}\dot{\mathbf{e}}) = \mathbf{D}(\ddot{\mathbf{q}}_d + \mathbf{\Lambda}\dot{\mathbf{e}}) - \mathbf{D}\ddot{\mathbf{q}} \\ &= \mathbf{D}(\ddot{\mathbf{q}}_d + \mathbf{\Lambda}\dot{\mathbf{e}}) + \mathbf{C}(\dot{q})\dot{\mathbf{q}} + \mathbf{T}_d - \mathbf{T} \\ &= \mathbf{D}(\ddot{\mathbf{q}}_d + \mathbf{\Lambda}\dot{\mathbf{e}}) - \mathbf{Cs} + \mathbf{C}(\dot{\mathbf{q}}_d + \mathbf{\Lambda e}) + \mathbf{T}_d - \mathbf{T} \\ &= -\mathbf{Cs} - \mathbf{T} + \mathbf{F} + \mathbf{T}_d\end{aligned} \tag{12}$$

Where $\mathbf{F} = \mathbf{D}(\ddot{\mathbf{q}}_d + \mathbf{\Lambda}\dot{\mathbf{e}}) + \mathbf{C}(\dot{\mathbf{q}}_d + \mathbf{\Lambda e})$, is the model uncertainty.

The model uncertainty $\mathbf{F}$ is usually difficult to obtain in practical application, if the parameters had been changed for some reasons, the control performance might get worse. We use the RBF NN to estimate the model uncertainty $\mathbf{F}$.

For each axis $i$, $i = 1, 2, 3$, the idea formula of RBF NN is designed as

$$h_{ij} = \exp(\left\|\mathbf{x}_i - \mathbf{c}_{ij}\right\| / \sigma^2{}_{ij}),\ j = 1, 2, \cdots, m \tag{13}$$

$$F_i = w_i{}^{\mathrm{T}}\mathbf{h}_i + \varepsilon_i \tag{14}$$

Where $\mathbf{x}_i = [e_i \ \ \dot{e}_i \ \ q_{di} \ \ \dot{q}_{di} \ \ \ddot{q}_{di}]$,is the input of RBF NN for axis $i$; $\mathbf{h}_i = [h_{i1} \ \ h_{i2} \ \ \cdots \ \ h_{im}]^{\mathrm{T}}$; $\varepsilon_i$ is the estimation error of RBF NN; $w_i$ is the idea weight value for axis $i$.

From function (14) of $\mathbf{F}$, we have

$$\mathbf{x} = [\mathbf{e} \quad \dot{\mathbf{e}} \quad \mathbf{q}_d \quad \dot{\mathbf{q}}_d \quad \ddot{\mathbf{q}}_d] \tag{15}$$

Then

$$\begin{aligned}\mathbf{F} &= [F_1 \ \ F_2 \ \ \cdots \ \ F_i \ \ \cdots \ \ F_n]^{\mathrm{T}} \\ &= \begin{bmatrix} w_1{}^{\mathrm{T}}\mathbf{h}_1 + \varepsilon_1 \\ \vdots \\ w_i{}^{\mathrm{T}}\mathbf{h}_i + \varepsilon_i \\ \vdots \\ w_n{}^{\mathrm{T}}\mathbf{h}_n + \varepsilon_n \end{bmatrix} = \begin{bmatrix} w_1{}^{\mathrm{T}}\mathbf{h}_1 \\ \vdots \\ w_i{}^{\mathrm{T}}\mathbf{h}_i \\ \vdots \\ w_n{}^{\mathrm{T}}\mathbf{h}_n \end{bmatrix} + \boldsymbol{\varepsilon}\end{aligned} \tag{16}$$

Where $\boldsymbol{\varepsilon} = [\varepsilon_1 \ \ \varepsilon_2 \ \ \cdots \ \ \varepsilon_i \ \ \cdots \ \ \varepsilon_n]^{\mathrm{T}}$, $\|\boldsymbol{\varepsilon}\| \le \varepsilon_N$.

The control law contains formula (12) and (16) is the sliding mode control law based on RBF NN estimation. The model uncertainty of each axis can be estimated well. However, it can't fit the real-time control.

### *B. Adaptive Approach with single parameter Approximation*

Instead of the weights of RBF NN, an estimated single parameter is used. Defining the estimated value of weight is $\hat{w}_i$, the estimated error is

$$\tilde{w}_i = w_i - \hat{w}_i,\ \|w_i\| \le w_{i\max}. \tag{17}$$

And assuming that the single parameter is $\phi > 0$, and $\phi = \max_{1\le i\le n}\{\|w_i\|^2\}$, its estimation is $\hat{\phi}$, then

$$\tilde{\phi} = \hat{\phi} - \phi \tag{18}$$

Defining $\mathbf{W} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}$, $\mathbf{H} = \begin{bmatrix} h_1 \\ \vdots \\ h_n \end{bmatrix}$, $\tilde{\mathbf{W}} = \mathbf{W} - \hat{\mathbf{W}}$, according the GL operator in lecture [12], we get

$$\mathbf{W} \circ \mathbf{H} = \begin{bmatrix} w_1{}^{\mathrm{T}}h_1 \\ \vdots \\ w_n{}^{\mathrm{T}}h_n \end{bmatrix}, \mathbf{s} \circ \mathbf{s} = \begin{bmatrix} s_1{}^{\mathrm{T}}s_1 \\ \vdots \\ s_n{}^{\mathrm{T}}s_n \end{bmatrix},\ \mathbf{H} \circ \mathbf{H} = \begin{bmatrix} h_1{}^{\mathrm{T}}h_1 \\ \vdots \\ h_n{}^{\mathrm{T}}h_n \end{bmatrix}, \text{ then}$$

$$\mathbf{F} = \mathbf{W} \circ \mathbf{H} + \boldsymbol{\varepsilon} \tag{19}$$

By the sliding mode and Lyapunov theory, if the closed-loop system is stably, the sliding control law is designed as follows,

$$\mathbf{T} = \frac{1}{2}\hat{\phi}\mathbf{s} \circ (\mathbf{H} \circ \mathbf{H}) + \mathbf{Ks} + (\varepsilon_N + b_d)\operatorname{sgn}(\mathbf{s}) \tag{20}$$

Where, $\|\mathbf{T}_d\| \le b_d$, $\mathbf{K} = diag\{K_1, K_2, \cdots, K_n\}$, $K_i > 0$.

The adaptive law of $\hat{\phi}$ is

$$\dot{\hat{\phi}} = \frac{\gamma}{2}\sum_{i=1}^{n} s_i^2 \|\mathbf{h}_i\|^2 \tag{21}$$

Plug equation (20) into the equation (12), we have

$$\begin{aligned}\mathbf{D}\dot{\mathbf{s}} = &-(\mathbf{K} + \mathbf{C})\mathbf{s} - \frac{1}{2}\hat{\phi}\mathbf{s} \circ (\mathbf{H} \circ \mathbf{H}) \\ &+ \mathbf{F} + \mathbf{T}_d - (\varepsilon_N + b_d)\operatorname{sgn}(\mathbf{s})\end{aligned} \tag{22}$$

### *C. Stability Analysis*

Considering the Lyapunov function candidate as follows,

$$\mathbf{V} = \frac{1}{2}\mathbf{s}^{\mathrm{T}}\mathbf{D}\mathbf{s} + \frac{1}{2\gamma}\tilde{\phi}^2,\ \gamma > 0. \tag{23}$$

The time derivative of $\mathbf{V}$ is given as,

$$\begin{aligned}\dot{\mathbf{V}} &= \mathbf{s}^{\mathrm{T}}\mathbf{D}\dot{\mathbf{s}} + \frac{1}{2}\mathbf{s}^{\mathrm{T}}\dot{\mathbf{D}}\mathbf{s} + \frac{1}{\gamma}\tilde{\phi}\dot{\hat{\phi}} \\ &= \mathbf{s}^{\mathrm{T}}[-(\mathbf{K} + \mathbf{C})\mathbf{s} - \frac{1}{2}\hat{\phi}\mathbf{s} \circ (\mathbf{H} \circ \mathbf{H}) + \mathbf{F} + \mathbf{T}_d - (\varepsilon_N + b_d)\operatorname{sgn}(\mathbf{s})] \\ &\quad + \frac{1}{2}\mathbf{s}^{\mathrm{T}}\dot{\mathbf{D}}\mathbf{s} + \frac{1}{\gamma}\tilde{\phi}\dot{\hat{\phi}} \\ &= \mathbf{s}^{\mathrm{T}}[-\mathbf{Ks} - \frac{1}{2}\hat{\phi}\mathbf{s} \circ (\mathbf{H} \circ \mathbf{H}) + \mathbf{W} \circ \mathbf{H} + \boldsymbol{\varepsilon} + \mathbf{T}_d - (\varepsilon_N + b_d)\operatorname{sgn}(\mathbf{s})] \\ &\quad + \frac{1}{2}\mathbf{s}^{\mathrm{T}}(\dot{\mathbf{D}} - 2\mathbf{C})\mathbf{s} + \frac{1}{\gamma}\tilde{\phi}\dot{\hat{\phi}}\end{aligned} \tag{24}$$

Because that $\mathbf{s}^{\mathrm{T}}[\boldsymbol{\varepsilon}+\mathbf{T}_d-(\varepsilon_N+b_d)\operatorname{sgn}(\mathbf{s})]\le 0$, and $\mathbf{s}^{\mathrm{T}}(\dot{\mathbf{D}}-2\mathbf{C})\mathbf{s}\le 0$, we get

$$\dot{\mathbf{V}}\le \mathbf{s}^{\mathrm{T}}[-\frac{1}{2}\hat{\phi}\mathbf{s}\circ(\mathbf{H}\circ\mathbf{H}+\mathbf{W}\circ\mathbf{H})]-\mathbf{s}^{\mathrm{T}}\mathbf{K}\mathbf{s}+\frac{1}{\gamma}\tilde{\phi}\dot{\hat{\phi}} \quad (25)$$

Note the formula as follows,

$$\begin{aligned}\mathbf{s}^{\mathrm{T}}[\mathbf{W}\circ\mathbf{H}]&=[s_1 \ \cdots \ s_n]\begin{bmatrix} w_1{}^{\mathrm{T}}h_1 \\ \vdots \\ w_n{}^{\mathrm{T}}h_n\end{bmatrix}\\ &=s_1w_1{}^{\mathrm{T}}h_1+\cdots+s_nw_n{}^{\mathrm{T}}h_n \\ &=\sum_{i=1}^{n}s_iw_i{}^{\mathrm{T}}h_i\end{aligned} \quad (26)$$

$$\begin{aligned}\mathbf{s}^{\mathrm{T}}[-\frac{1}{2}\hat{\phi}\mathbf{s}\circ(\mathbf{H}\circ\mathbf{H})]&=-\frac{1}{2}\hat{\phi}[s_1 \ \cdots \ s_n]\left[\begin{bmatrix}s_1\\ \vdots \\ s_n\end{bmatrix}\circ\begin{bmatrix}\mathbf{h}_1{}^{\mathrm{T}}\mathbf{h}_1\\ \vdots \\ \mathbf{h}_n{}^{\mathrm{T}}\mathbf{h}_n\end{bmatrix}\right]\\ &=-\frac{1}{2}\hat{\phi}[s_1 \ \cdots \ s_n]\begin{bmatrix}s_1\mathbf{h}_1{}^{\mathrm{T}}\mathbf{h}_1\\ \vdots \\ s_n\mathbf{h}_n{}^{\mathrm{T}}\mathbf{h}_n\end{bmatrix}_1\\ &=-\frac{1}{2}\hat{\phi}(s_1^2\left\|\mathbf{h}_1\right\|^2+\cdots+s_n^2\left\|\mathbf{h}_n\right\|^2)\\ &=-\frac{1}{2}\hat{\phi}\sum_{i=1}^{n}s_i^2\left\|\mathbf{h}_i\right\|^2\end{aligned} \quad (27)$$

$$s_i^2\phi\mathbf{h}_i{}^{\mathrm{T}}\mathbf{h}_i+1\ge s_i^2\left\|w_i\right\|^2\left\|\mathbf{h}_i\right\|^2+1\ge 2s_iw_i{}^{\mathrm{T}}\mathbf{h}_i \quad (28)$$

$$s_iw_i{}^{\mathrm{T}}\mathbf{h}_i\le\frac{1}{2}s_i^2\phi\mathbf{h}_i{}^{\mathrm{T}}\mathbf{h}_i+\frac{1}{2} \quad (29)$$

Then we have

$$\mathbf{s}^{\mathrm{T}}[\mathbf{W}\circ\mathbf{H}]=\sum_{i=1}^{n}s_iw_i{}^{\mathrm{T}}\mathbf{h}_i\le\frac{1}{2}\phi\sum_{i=1}^{n}s_i^2\mathbf{h}_i{}^{\mathrm{T}}\mathbf{h}_i+\frac{n}{2} \quad (30)$$

$$\begin{aligned}\dot{\mathbf{V}}&\le-\frac{1}{2}\hat{\phi}\sum_{i=1}^{n}s_i^2\left\|\mathbf{h}_i\right\|^2+\frac{1}{2}\phi\sum_{i=1}^{n}s_i^2\mathbf{h}_i{}^{\mathrm{T}}\mathbf{h}_i+\frac{n}{2}-\mathbf{s}^{\mathrm{T}}\mathbf{K}\mathbf{s}+\frac{1}{\gamma}\tilde{\phi}\dot{\hat{\phi}}\\ &=-\frac{1}{2}\tilde{\phi}\sum_{i=1}^{n}s_i^2\left\|\mathbf{h}_i\right\|^2+\frac{n}{2}-\mathbf{s}^{\mathrm{T}}\mathbf{K}\mathbf{s}+\frac{1}{\gamma}\tilde{\phi}\dot{\hat{\phi}}\\ &=\tilde{\phi}(-\frac{1}{2}\tilde{\phi}\sum_{i=1}^{n}s_i^2\left\|\mathbf{h}_i\right\|^2+\frac{1}{\gamma}\dot{\hat{\phi}})+\frac{n}{2}-\mathbf{s}^{\mathrm{T}}\mathbf{K}\mathbf{s}\\ &=\frac{n}{2}-\mathbf{s}^{\mathrm{T}}\mathbf{K}\mathbf{s}\le 0\end{aligned} \quad (31)$$

Where $\left\|\mathbf{s}\right\|\le\sqrt{\dfrac{n}{2\mathbf{K}}}$.

It means that considering the system (8) in closed loop with the sliding mode controller (20) and the adapting law of single parameter $\hat{\phi}$ in (21), then the tracking error $\mathbf{e}(t)$ is globally asymptotically stabilized to zero, even though the upper bound of the parameter uncertainty and the disturbance is not exactly confirmed.

## IV. Numerical Simulation

We carried out numerical simulations to assess the performance of the controller given in (20) and (21). The values of the parameters correspond to a laboratory prototype built in our institution and they are founded in Table I .

TABLE I. PARAMETER VALUES OF THE ROBOT

| Parameters | Value | Parameters | Value |
|---|---|---|---|
| $F_v$ | 1.86[Nm/(rad/sec)] | $J_w$ | 0.8[gcm$^2$] |
| $K_t$ | 0.0259 [Nm/A] | $V_s$ | 24[V] |
| $K_b$ | 0.0259 [V/(rad/sec)] | m | 23[kg] |
| n | 22 | L | 0.225[m] |
| $R_a$ | 0.611[Ω] | r | 0.1[m] |
| J | 33.3[gcm$^2$] | | |

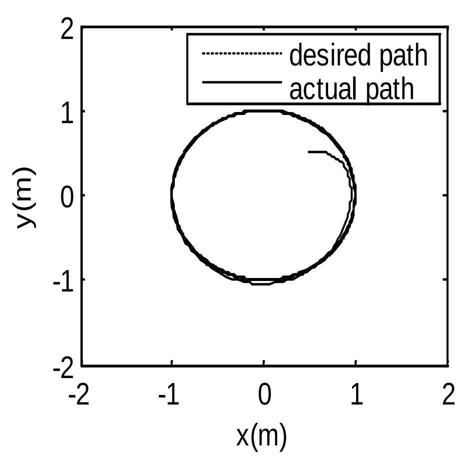
In the numerical simulations, the control parameters are as follows, the initial posture is $\mathbf{q}_d=[0.5,0.5,0.5]^{\mathrm{T}}$, $\mathbf{K}=diag\{100,100,100\}$, $\boldsymbol{\Lambda}=diag\{45,45,45\}$, the adapting weight is $\gamma=300$, $\varepsilon_N=0.5$, $b_d=0.2$, according to formula(15),

$$\mathbf{c}=0.1\times\begin{bmatrix}-1.5 & -1 & -0.5 & 0 & 0.5 & 1 & 1.5\\ -1.5 & -1 & -0.5 & 0 & 0.5 & 1 & 1.5\\ -1.5 & -1 & -0.5 & 0 & 0.5 & 1 & 1.5\\ -1.5 & -1 & -0.5 & 0 & 0.5 & 1 & 1.5\\ -1.5 & -1 & -0.5 & 0 & 0.5 & 1 & 1.5\end{bmatrix}, \quad b_i=0.5,$$

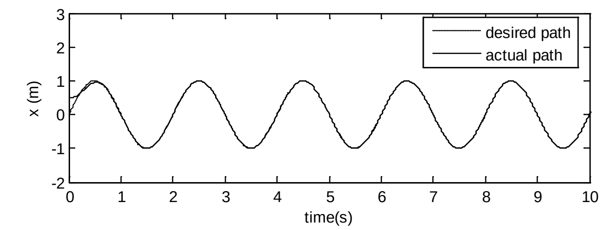
$i=1,\cdots,7$.

### A. The Model Uncertainty Has Been Known Exactly

The robot tracked the circle path with traditional sliding mode controller (TSMC) in [10] and the controller with single parameter approximation (SPAC), if the model uncertainty has been known exactly, the results of path tracking are shown in Fig. 2 and Fig.4, and the control torque inputs are in Fig. 3 and Fig.5.

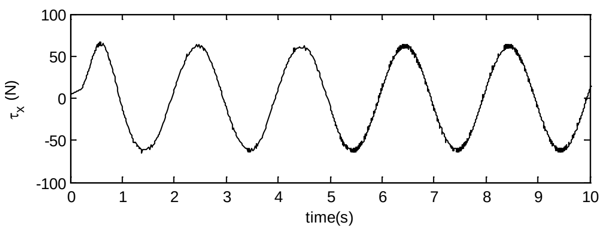



(a) Result of path tracking

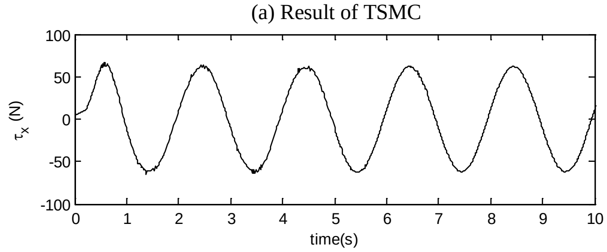



(b) Tracking result in $x$ axis

**Fig. 2.** Tracking results




(a) Result of TSMC




(b) Result of SPAC

**Fig. 3.** Curve of the control torque input $\tau_x$

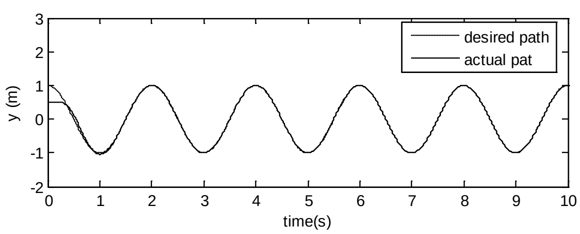
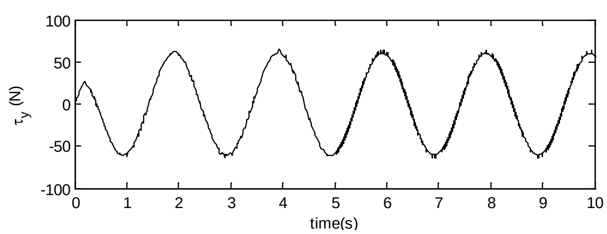
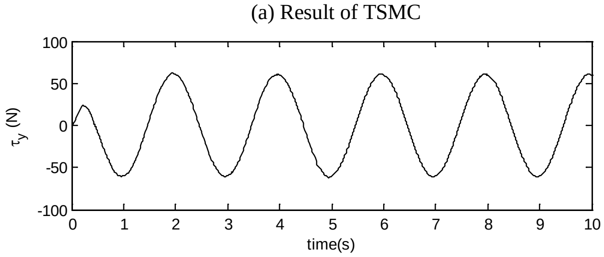



**Fig. 4.** Tracking results in $y$ axis




(a) Result of TSMC



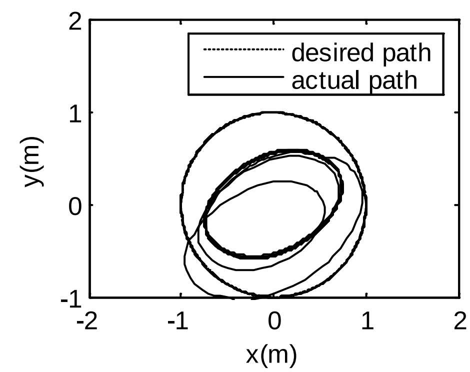

(b) Result of SPAC

**Fig. 5.** Curve of the control torque input $\tau_y$

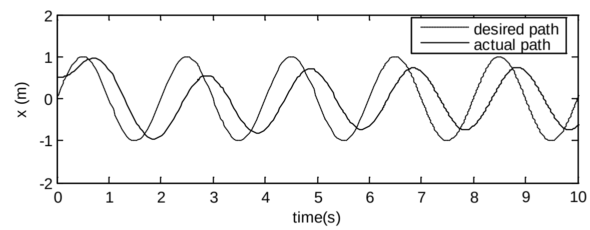
When the parameters are exactly confirmed, from Fig. 2 and Fig. 4, the SPAC can make the robot track the path quickly with smaller tracking error. In Fig. 3 (a) and Fig. 5(a), the control torque inputs $\tau_x$ and $\tau_y$ produced from TSMC has chattered obviously after $t = 5$ s, however, in Fig. 3 (b) and Fig. 5(b), the inputs produced from SPAC are smoother, because the RBF NN can reduce the high-frequency chatters.

*B. The Model Uncertainty Has not Been Confirmed Exactly*

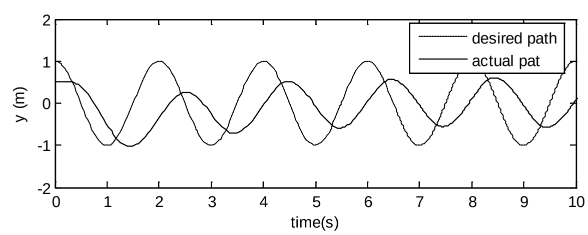
If the model uncertainty **F** has been changed from **F** to 1.5**F**, the results of path tracking with TSMC are shown in Fig. 6 and the results of path tracking with SPAC are shown in Fig. 7.



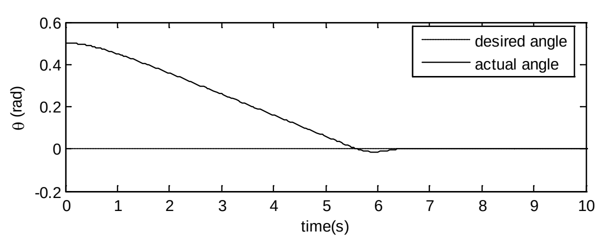

(a)Result of path tracking




(b) Tracking result in $x$ axis

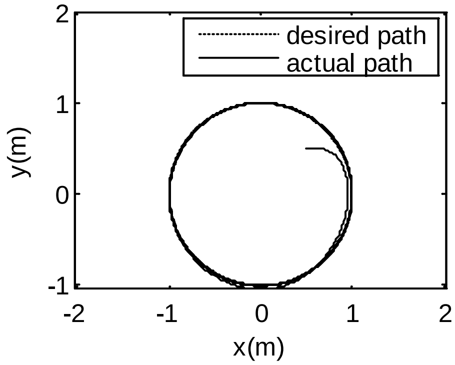



(c) Tracking result in $y$ axis

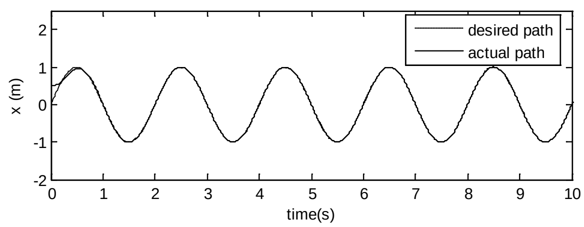



(d) Tracking result of angle

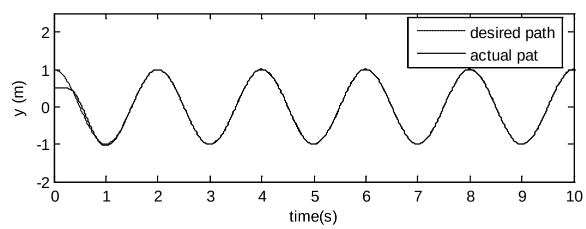
**Fig. 6.** Tracking results of TSMC



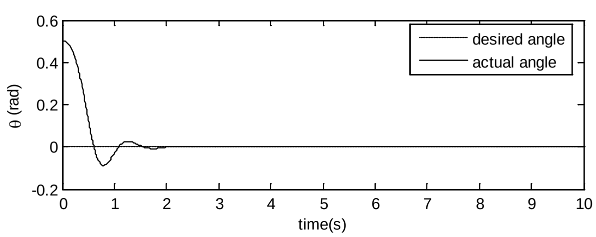

(a)Result of path tracking




(b) Tracking result in $x$ axis




(c) Tracking result in $y$ axis




(d) Tracking result of angle

**Fig. 7.** Tracking results of SPAC

From Fig. 6, when the model uncertainty is not exactly confirmed, for example, $\mathbf{F}$ has been changed from $\mathbf{F}$ to $1.5\mathbf{F}$, in other words , the actual control model has bigger errors compared with the nominal control model in formula (8). And then the TSMC cannot track the reference path quickly, sometimes loses the desired path.

But in Fig. 7 the SPAC can make the robot track its desired path with good performance. It is because that the adapting law can estimate the value timely and effective. When the parameters of control model have been changed, the adapting law could estimate the parameters timely and changed the control values to suit the changed model. In other words, the adapting law can keep the control system with better robust stability and ability of restraining disturbance.

## V. Conclusions

As the model uncertainties and disturbances in the dynamic model of four-wheel omnidirectional mobile robot are usually difficult to obtain, an adaptive sliding mode control is presented, and a RBF neural network is used to estimate the model uncertainty. To fit the real-time control, an approach with single parameter approximation is designed. The asymptotic stability of the closed loop system is formally proved. The trajectory tracking results of SPAC have shown that the properties of the closed-loop system have a better performance than TSMC, whether or not the model uncertainty is constant or changed. The inherent chattering of SPAC has been eliminated effectively, however, TSMC has caused high noise amplifications and high control costs.

## References

[1] S. J.C. Lins Barreto, A.G. Scolari Conceicao, C. E. T. Dorea, et al. Design and implementation of model-predictive control with friction compensation on an omnidirectional mobile robot[J]. IEEE/ASME Transactions on Mechatronics, 2014, 19(2): 467-476.

[2] K. N. Tamas, D. A. Raffaello, and G. Pritam, “Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle,” Robotics and Autonomous System, vol. 46(1), pp. 47-64, February 2004.

[3] Fukao T, Nakagawa H, Adachi N. Adaptive tracking control of a nonholonomic mobile robot. IEEE Transactions on Robotics and Automation, 2000, 16(5): 609-615.

[4] S. H. Chen, J. C. Juang, and S. H. Su, “Backstepping control with sum of squares design for omni-directional mobile robots,” Proceddings of ICCAS-SICE, pp. 545-550, August 2009.

[5] Park B S, Yoo S J, Park J B, et al. Adaptive neural sliding mode control of nonholonomic wheeled mobile robots with model uncertainty. IEEE Transactions on Control Systems Technology, 2009, 17(1): 207-214.

[6] Das T, Kar I N. Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots. IEEE Transactions on Control Systems Technology, 2006, 14(3): 501-510.

[7] de Sousa Jr C, Hemerly E M, Galvao R K H. Adaptive control for mobile robot using wavelet networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2002, 32(4): 493-504.

[8] Shojaei K, Shahri A M. Adaptive robust time-varying control of uncertain nonholonomic robotic systems. IET Control Theory and Applications, 2012, 6(1): 90-102.

[9] H. C. Huang, C. C. Tsai, and S. C. Lin, Adaptive polar-space motion control for embedded omnidirectional mobile robots with parameter variations and uncertainties,” Journal of Intelligent & Robotic Systems, vol. 62(1), pp. 81-102, 2011.

[10] S. T. Kao, W. J. Chiou, and M. T. Ho, “Integral sliding mode control for trajectory tracking control of an omnidirectional mobile robot,” Proceedings of the 8th Asian Control Conference, pp. 765-770, 2011.

[11] D. Xu, D. B. Zhao, J. Q. Yi, and X. M. Tan, “Trajectory tracking control of omnidirectional wheeled mobile manipulators: robust neural network-based sliding mode approach,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 39(3), pp. 788-799, June 2009.

[12] S.S.Ge, C.C.Hang, T.H.Lee, et al. “Stable Adaptive Neural Network Control,” Boston, MA: Kluwer,2001.