2018-Design of Adaptive Sliding Mode Controller for Four-Mecanum Wheel Mobile Robot

Proceedings of the 37th Chinese Control Conference
July 25-27, 2018, Wuhan, China

Design of Adaptive Sliding Mode Controller for Four-Mecanum Wheel Mobile Robot

Xingyang Lu, Xiangying Zhang, Guoliang Zhang, Songmin Jia

[1] Faulty of Information Technology, Beijing University of Technology, Beijing, China

[2] Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing, China

[3] Engineering Research Center of Digital Community, Ministry of Education, Beijing, China

E-mail: xy_zhang@bjut.edu.cn

Abstract: In this paper, an adaptive sliding mode controller is designed for the omnidirectional mobile robot with four Mecanum wheels to track the trajectory. The Mecanum wheel mobile robot has been widely used in factories, hospitals, families and other workspaces because of its omnidirectional ability. Consider the dynamic model of the omnidirectional mobile robot based on Mecanum wheel with uncertainties and external disturbances, a neural network adaptive sliding mode control (NNASMC) strategy is proposed, which combines the techniques of Radial Basis Function (RBF) neural network adaptive control method and sliding mode control method. Finally, some simulation experiments are conducted in the Matlab environment. Simulation results demonstrate the effectiveness of the proposed control method.

Key Words: adaptive sliding mode control, Mecanum wheel, mobile robot, Radial Basis Function (RBF)

1 Introduction

In the recent years, the Mecanum wheel mobile robot with the omnidirectional ability has been widely applied in many fields, such as factories [1], soccer robots [2], hospitals, etc. The Mecanum wheel was invented in 1973 by BengtIlon with the Swedish company, Mecanum A.B., named Ilon, which consists of a number of passive rollers that are mounted at 45 degrees around a solid axis [3]. Due to the flexible motion ability, the Mecanum wheel mobile robot can move to any position without changing direction from the initial position and have superior steering capability in narrow workspace. Owing to the advantages, the Mecanum wheel mobile robot has aroused wide interest in research. Among various types of Mecanum wheel mobile robots, four Mecanum wheels mobile robot is one of these types, which possesses three degrees of freedom in a horizontal plane.

Many researchers have designed and analyzed the structure of the robot from the kinematics and dynamics of the omnidirectional mobile robot with four Mecanum wheels [4-6]. The kinematics depicts the mathematical relationship between the position and velocity of the mobile robot, which is a direct model. Dynamics can clearly describe the role of the force on the position and speed of the mobile robot, which is an essential model. In order to do better research, we get the kinematic model and obtain the dynamic model by using the Lagrange method, which just like the model in [4-7].

The omnidirectional mobile robot with four Mecanum wheels is derived by four DC motors. The performance of omnidirectional movement through the collaboration of four Mecanum wheels. In order to obtain better trajectory tracking control effect, a number of control method such as the Proportion-Integral-Derivative (PID) control method, fuzzy control method [8], sliding mode control method and others are widely used in a lot of research. In practice, however, uncertainties and external disturbances often exist in the omnidirectional motion system and sometimes the bounds of uncertainties keeps changing. The sliding mode control method has the robustness in the dynamic model control which exists the uncertainties, and when the uncertainties is bounded [9-11]. Therefore, the neural network adaptive sliding mode control (NNASMC) method for the tracking control of the four-Mecanum wheel mobile robot is proposed in this paper. The NNASMC method can take advantage of the approximation performance of Radial Basis Function (RBF) neural networks to estimate the uncertainties. To demonstrate the effectiveness of the proposed control scheme for trajectory tracking in presence of uncertainties and external disturbances, simulations are conducted in Matlab/Simulink environment.

The rest of this paper is organized as follows. Section 2 introduces the kinematic and dynamic model of the four Mecanum wheels omnidirectional mobile robot. In Section 3, the proposed NNASMC control scheme is described in details. Simulation experiments and analysis are given in Section 4 and Section 5 concludes our work finally.

2 Model Description

An omnidirectional mobile robot that consists of four Mecanum wheels (Fig. 1) in which four wheels are arranged symmetrically on the geometric center of the body. Figure 1 shows the kinematic geometry of an omnidirectional robot with four Mecanum wheels.

*This work is supported by National Natural Science Foundation of China under Grants 61703012, 81471770 and 61563011, Beijing Natural Science Foundation under Grant 4182010, and the BJUT united grand scientific research program on intelligent manufacturing.

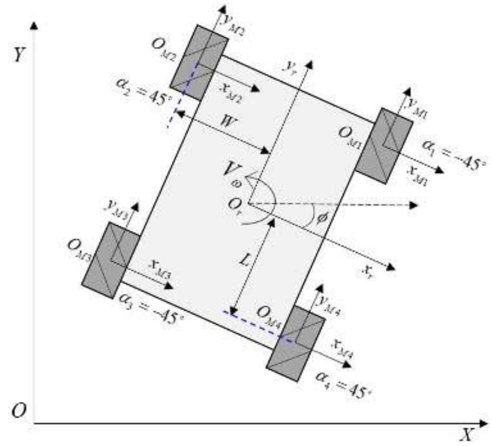

Fig. 1: Kinematic geometry of the robot

2.1 Kinematics

Suppose $q_{Mi}=[x_{Mi}, y_{Mi}]^T$ (i = 1, 2, 3, 4) as the position vector of wheel in O_{Mi} (i = 1, 2, 3, 4), the velocity vector is given as:

$$\dot{q}_{Mi}=\begin{bmatrix}\dot{x}_{Mi}\\ \dot{y}_{Mi}\end{bmatrix}=\begin{bmatrix}0 & \sin(\alpha_i)\\ R_i^w & -\cos(\alpha_i)\end{bmatrix}\begin{bmatrix}\omega_i\\ v_{Mi}^r\end{bmatrix}=J_1^T\dot{q}_i^{\theta} \quad (1)$$

where ω_i (i = 1, 2, 3, 4) is the wheel angular velocity, v_{Mi}^r (i = 1, 2, 3, 4) is the velocity vector at the center of the ground roller, R_i^w (i = 1, 2, 3, 4) is the wheel radius, and α_i (i = 1, 2, 3, 4) is the roller slope angle of each wheel. Using the rotation matrix in frame O_r, the robot velocity vector can be written as:

$$\dot{q}_i^r=\begin{bmatrix}\dot{x}_i^r\\ \dot{y}_i^r\end{bmatrix}=\begin{bmatrix}cos(\phi_{Mi}^r) & -\sin(\phi_{Mi}^r)\\ \sin(\phi_{Mi}^r) & cos(\phi_{Mi}^r)\end{bmatrix}\begin{bmatrix}\dot{x}_{Mi}\\ \dot{y}_{Mi}\end{bmatrix}=J_2^T\dot{q}_{Mi} \quad (2)$$

where ϕ_{Mi}^r (i = 1, 2, 3, 4) is the rotational angle of O_{Mi} (i = 1, 2, 3, 4) with respect to O_r . The vehicle is moving on flat ground, and thus,

$$\dot{q}_i^r=\begin{bmatrix}\dot{x}_i^r\\ \dot{y}_i^r\\ \dot{\varphi}_i^r\end{bmatrix}=\begin{bmatrix}1 & 0 & -d_{iy}\\ 0 & 1 & d_{ix}\end{bmatrix}\begin{bmatrix}v_x\\ v_y\\ v_\omega\end{bmatrix}=J_3^T\dot{q}^v \quad (3)$$

where $\dot{q}^v=\begin{bmatrix}v_x & v_y & v_\omega\end{bmatrix}^T$ is the velocity vector of wheel in O, d_{ix} and d_{iy} are translational distance between O_r and O_{Mi} in x and y direction respectively. Using (1) (2) and (3), we can obtain the following equation:

$$\dot{q}_i^{\theta}=(J_2^T)^{-1}(J_1^T)^{-1}J_3^T\dot{q}^v=R_i^T\dot{q}^v \quad (4)$$

Consider that Mecanum wheels are identical, the corresponding kinematic parameters can be determined as $R_i^w=R, \phi_{M1}^r=\phi_{M2}^r=\phi_{M3}^r=\phi_{M4}^r=0, |d_{ix}|=W, |d_{iy}|=L,$ $\alpha_1=\alpha_3=-45°, \alpha_2=\alpha_4=45°$.

Using (4) and Jacobian matrices for each wheel, the inverse kinematic model of the mobile robot is obtained as:

$$\begin{bmatrix}\omega_1\\ \omega_2\\ \omega_3\\ \omega_4\end{bmatrix}=\frac{1}{R}\begin{bmatrix}-1 & 1 & L+W\\ 1 & 1 & -(L+W)\\ -1 & 1 & -(L+W)\\ 1 & 1 & L+W\end{bmatrix}\begin{bmatrix}v_x\\ v_y\\ v_\omega\end{bmatrix} \quad (5)$$

with the kinematics model can be obtained as follows:

$$\begin{bmatrix}v_x\\ v_y\\ v_\omega\end{bmatrix}=\frac{R}{4}\begin{bmatrix}-1 & 1 & -1 & 1\\ 1 & 1 & 1 & 1\\ \frac{1}{L+W} & -\frac{1}{L+W} & -\frac{1}{L+W} & \frac{1}{L+W}\end{bmatrix}\begin{bmatrix}\omega_1\\ \omega_2\\ \omega_3\\ \omega_4\end{bmatrix} \quad (6)$$

2.2 Dynamics

The dynamic model of the omnidirectional robot with uncertainties moving on a plane flat surface is developed using Lagrange method [4-7].

The kinetic energy equation of the robot is calculated as follows：

$$\mathrm{E}_k=\frac{1}{2}m(v_x{}^2+v_y{}^2)+\frac{1}{2}I_z\omega^2+\frac{1}{2}I_\omega\sum_{i=1}^{4}\omega_i^2 \quad (7)$$

where m is the total mass of the robot, and I_z, I_w denote the moment of inertia of the robot's body and the wheels respectively. Furthermore, the loss energy equation is as follows:

$$F=\frac{1}{2}D_M\sum_{i=1}^{4}\omega_i^2 \quad (8)$$

where D_w is the coefficient of the wheel's viscous friction. Lagrange equation is as follows [6]:

$$\frac{d}{dt}(\frac{\partial L}{\partial \dot{q}})-\frac{\partial L}{\partial q}+\frac{\partial F}{\partial q}=Q \quad (9)$$

where q is the generalized coordinates, L is the Lagrangian function, which is equal to the kinetic energy of the system minus the potential energy, F is for the system dissipation function, and Q is the generalized force. Make a synthesis of the above equations and we get:

$$M(q)\ddot{q}+D_w(q)\dot{q}=u(t)-\tau_d \quad (10)$$

where $u(t)$=$[u_1(t) \quad u_2(t) \quad u_3(t) \quad u_4(t)]^T$, $u_i(t)$, i= 1, 2,3,4 represents control input of each wheel, τ_d represents external disturbance, $\dot{q}$=$[\omega_1 \quad \omega_2 \quad \omega_3 \quad \omega_4]^T$ is the angular velocity of the Mecanum wheel. By considering the parameter variations in the robot, in which the Positive definite system inertia matrix $M(q)$, and the coefficient of the wheel's viscous friction $D_M(q)$ can be respectively expressed as $M(q)=M_0(q)+\Delta M$, $D_M(q)=D_M^0(q)+\Delta D_M$, where the ΔM and ΔD_M are the perturbed terms portion, $M_0(q)$, and $D_M^0(q)$ are the nominal values respectively. Hence, the dynamic model of the robot we obtained as follows:

$$\begin{aligned}M_0(q)\ddot{q}+D_M^0(q)\dot{q}&=u(t)-\tau_d-\Delta M-\Delta D_M\\ &=u(t)-E(q)\end{aligned} \quad (11)$$

where $M_0(q)=\begin{bmatrix} C & -B & B & D \\ -B & C & D & B \\ B & D & C & -B \\ D & B & -B & C \end{bmatrix}, A=\frac{mr^2}{8},$

$$B=\frac{I_z r^2}{16(W+L)^2}, C=A+B+I_\omega, D=A-B$$

3 Controller Design Based on NNASMC

In the control scheme of this paper, it is divided into two parts: the kinematics control of the outer layer and the dynamic control of the inner layer. The outer loop adopts PID controller, and in the inner loop, the NNASMC is adopted which is described in details in the following section.

3.1 Sliding Mode Control

In this subsection, we utilize the sliding mode control method to design a dynamic tracking controller, so that the actual velocities of the wheel converges to the control velocities.

Before designing the sliding mode controller, we rewrite the dynamic model of the robot as follows:

$$\ddot{q}=M_0^{-1}(q)u(t)-M_0^{-1}(q)E(q)-H_0(q)\dot{q} \qquad (12)$$

where $H_0(q)=M_0^{-1}(q)D_M^0(q)$, and we let the uncertainty function of $g(z_{in})=M_0^{-1}(q)E(q)=0$,thus, the following equation is obtained:

$$\ddot{q}=M_0^{-1}(q)u(t)-H_0(q)\dot{q} \qquad (13)$$

The control objective in this subsection is to make the actual velocities $\dot{q}$ of the wheel follow the reference velocities $\dot{q}_d$ precisely, that is, the tracking error $e_M(t)=\dot{q}_d-\dot{q}$ is as close to zero as possible. To achieve precise and fast control objective, the following sliding surfaces is defined as [12]:

$$s(t)=\dot{e}_M(t)+\Lambda e_M(t) \qquad (14)$$

where $e_M(t)=\begin{bmatrix} e_1^M(t) & e_2^M(t) & e_3^M(t) & e_4^M(t)\end{bmatrix}^T$ and $\Lambda=diag(\lambda_1 \quad \lambda_2 \quad \lambda_3 \quad \lambda_4)$ is a design constant positive matrix. In this paper, the following reaching law is adopted [11]:

$$\dot{s}(t)=-\lambda_1^s s(t)-\lambda_2^s \operatorname{sgn}(s(t)) \qquad (15)$$

where $\lambda_1^s=diag(\lambda_{11}^s \quad \lambda_{12}^s \quad \lambda_{13}^s \quad \lambda_{14}^s)$, $\lambda_2^s=diag(\lambda_{21}^s \quad \lambda_{22}^s \quad \lambda_{23}^s \quad \lambda_{24}^s)$ are constant matrices and positive definite, and $\operatorname{sgn}(s(t))$ is a discontinuous function, it can be given as follows [13]:

$$\operatorname{sgn}(s(t))=\begin{cases}\dfrac{s(t)}{\|s(t)\|}, & \text{when} \quad \|s(t)\|>0 \\ 0, & \text{when} \quad \|s(t)\|=0\end{cases} \qquad (16)$$

Finally, according to (12)–(16), through a series of substitutions and transformations, the sliding mode controller based on approaching law can be expressed as follows:

$$u_s(t)=M_0(q)(\lambda_1^s s(t)+\lambda_2^s \operatorname{sgn}(s(t))+\Lambda(\dot{q}_d-\dot{q}) + \ddot{q}_d+H_0(q)\dot{q}) \qquad (17)$$

3.2 RBF Neural Network Adaptive Control

In practice, many of the parameters in the dynamic system are difficult to measure accurately, and the existence of external disturbances makes it difficult to get a precise mathematical model. In this paper, RBF neural network will be utilized to emulate the uncertain nonlinear function $E(q)$ in Equation (11) by creating an adaptation control law. In the case of the mathematical model with uncertainty, the sliding mode control law is obtained as follows:

$$\begin{aligned} u_s(t)&=M_0(q)(\lambda_1^s s(t)+\lambda_2^s \operatorname{sgn}(s(t))+\Lambda(\dot{q}_d-\dot{q}) \\ &\quad +\ddot{q}_d+H_0(q)\dot{q}+g(z_{in})) \\ &=M_0(q)(\lambda_1^s s(t)+\lambda_2^s \operatorname{sgn}(s(t))+f(z_{in}))\end{aligned} \qquad (18)$$

We use $\hat{f}(z_{in})$ to estimation $f(z_{in})$ the unknown nonlinear function. And the function $f(z_{in}): R^p \to R$ can be represented as follows [14]:

$$f(z_{in})=W^T h(z_{in})+\varepsilon \qquad (19)$$

where $z_{in}=[x_1,x_2,\cdots,x_n]^T\in R^n$ is the input vector, $W=[W_1,W_2\cdots W_m]^T\in R^m$ represents a vector of updated network weights, $h(z_{in})=[h_1(z_{in}),h_2(z_{in})\cdots h_p(z_{in})]^T\in R^p$ represents a vector of basis function, is approximation error of network, which is bounded with $\|\varepsilon\|\le\varepsilon_N$,and ε_N is an unknown positive constant. For RBF network, the basis function $h_j(z_{in}),(j=1,2,3,4)$ is a particular network architecture with the form of Gaussian functions as [14]:

$$h_j(z_{in})=\exp\left(-\frac{\|z_{in}-c_j\|^2}{2b_j^2}\right) \qquad (20)$$

where

$c=[c_{ij}]^T\in R^{n\times m},(i=1,2,\cdots,n, j=1,2,\cdots,m)$ is the coordinate vector of the center point of the Gauss basis function of the j neuron of the hidden layer, $b=[b_1,b_2,\cdots,b_m]^T$, b_j is the width of the center point of the Gauss basis function of the *j* neuron of the hidden layer.

Using the approximation ability of RBF neural network, the uncertainty nonlinear function can be approximated by [14]:

$$\hat{f}(z_{in})=\hat{W}^T h(z_{in}) \qquad (21)$$

where $\hat{W}^T\in R^m$ represents an updated weight matrix. And the weight of neural network is updated as follows:

$$\dot{\hat{W}}=\Gamma h(z_{in})s^T(t) \qquad (22)$$

We need to construct an adjustment mechanism for eliminate the unknown nonlinear function $f(z_{in})$, and the state inputs of RBF neural network is defined as:

$$z_{in}=[e_M(t) \quad \dot{e}_M(t) \quad \dot{q} \quad \ddot{q}]^T \qquad (23)$$

4 Numeral Experiment

In this section, a simulation for trajectory tracking of the omnidirectional mobile robot are performed to illustrate the effectiveness of the proposed control scheme in MATLAB/Simulink environment.

In addition, the performance of the proposed control method in this paper are compared with the SMC control method and conventional PID control method, and we consider the uncertainties and external disturbances of the model in the simulations. The physical parameters of the omnidirectional mobile robot are selected as: M_0=98kg, R=0.09m, W=0.36m, L=0.52m, I_z =4.1kg.m², I_ω = 0.029 kg.m², D_M^0 =0.4. The parameters of the designed control law are set as follows:

$$\lambda_1^s = diag(15 \quad 15 \quad 15 \quad 15)$$

$$\lambda_2^s = diag(10 \quad 10 \quad 10 \quad 10)$$

$$\Lambda = diag(10 \quad 10 \quad 10 \quad 10)$$

The equation of the circular trajectory is given as:

$$\begin{cases} x = 2\sin(0.25t) \\ y = 2\cos(0.25t) \\ \varphi_\omega = 0 \end{cases}$$

where t is the simulation time in seconds and satisfies $t \geq 0$ .The initial posture of the mobile robot is $[x \quad y \quad \varphi_\omega]^T = [0 \quad 1 \quad 0]^T$.

In this simulation, the uncertainties and external disturbances are taken as:

$$\delta(t) = \begin{cases} 2\sin(t) & 10 \leq t \leq 16 \\ 0 & \text{otherwise} \end{cases}$$

Figure 2 shows simulation results of the circular trajectory tracking, the black line represents the reference curve, the red line represents the response curve when using the PID control method, the red line represents the response curve when using the sliding mode control method, and the red line represents the response curve when using the control method that proposed in this paper.

Figure 3 shows simulation results of the red dotted line frame in Fig.2 and shows the response curve in the presence of external disturbances.

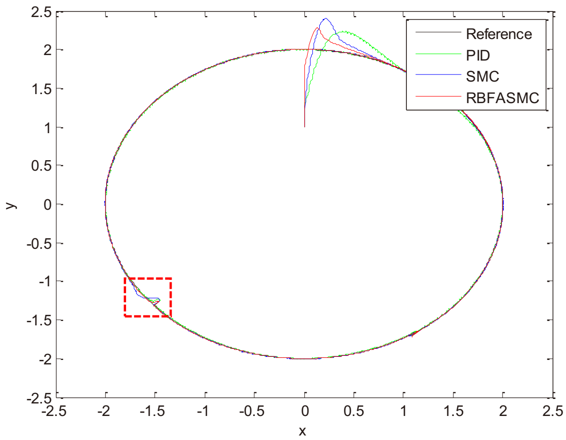

Fig. 2: Simulated Result of the circular trajectory tracking

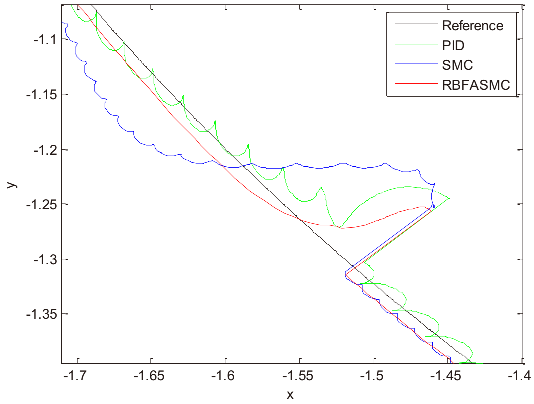

Fig. 3: Simulated Result of the red dotted line frame in Fig.2

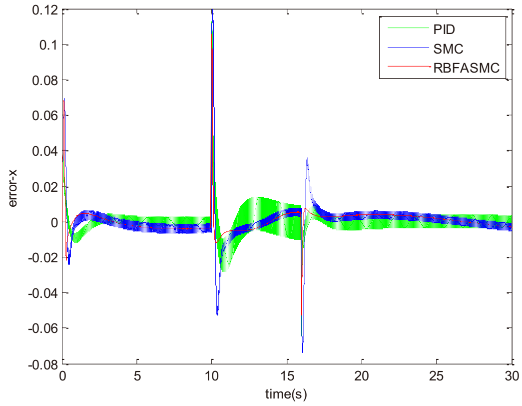

Fig. 4: Tracking error in the direction of x

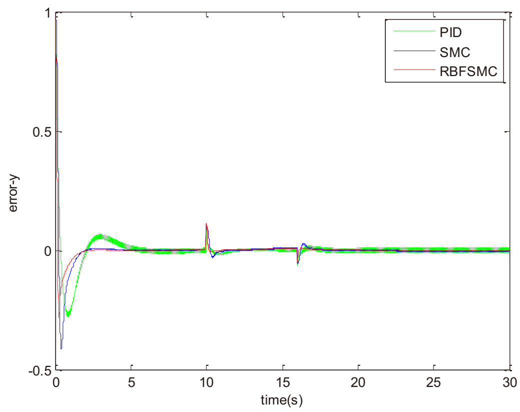

Fig. 5: Tracking error in the direction of y

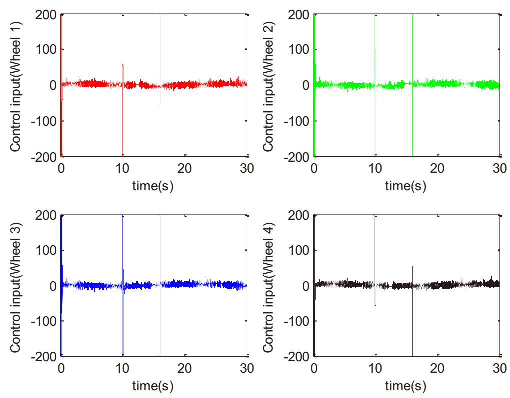

Fig. 6: Control input of the wheels

As shown in Fig.3 and Fig.4, the response curve of simulation experimental results illustrate that the control scheme proposed in this paper has better performance with the dynamic system in presence of external disturbances and uncertainties. Figure.4 and Figure.5 show the tracking error of the X direction and the Y direction in the trajectory tracking simulation experiment, respectively, and demonstrate the proposed control method has a better ability to deal with external disturbances. All of the simulation results show that the adaptive sliding mode control scheme proposed in this paper improved the trajectory tracking performance for the omnidirectional mobile robot with four Mecanum wheels compared with PID control method and sliding mode control method.

5 Conclusion

In this paper, an adaptive sliding mode control scheme for the omnidirectional mobile robot with four Mecanum wheels in presence of external disturbances and uncertainties is proposed. Simulation experiments are conducted in Matlab environments. Simulation results compared with PID control method and sliding mode control method demonstrate the effectiveness of the proposed control method. We will further improve the control strategy and analyze the theory In the future work.

References

[1] J. Kang, B. Kim, and M. Chung, Development of omnidirectional mobile robots with mecanum wheels assisting the disabled in a factory environment, in *Proceedings of 8th International Conference on Control,, Automation and Systems*, 2008: 2070–2075.

[2] K. Tu, and S. Luo, Design and implementation of omni-directiona1 soccer robots for RoboCup, in *Proceedings of 2006 IEEE International Conference on Systems*, 2006: 2000–2005.

[3] B. Ilon, Wheels for a course stable self propelling vehicle movable in any desired direction on the ground or some other base. *U. S. A. Patent*, 1975.

[4] N. Tlale, and M. Villiers, Kinematics and dynamics modelling of a mecanum wheeled mobile platform, in *Proceedings of 15th International Conference on Mechatronics and Machine Vision in Practice*, 2008: 657–662.

[5] Y. Wang, and D. Chang, Motion performance analysis and layout selection for motion system with four mecanum wheels, *Journal of Mechanical Engineering*, 45(5): 307-310, 2009.

[6] Y. Jia, X. Song, and S. Xu, Modeling and motion analysis of four-Mecanum wheel omni-directional mobile platform, *in Proceedings of 2014 Automatic Control Conference*, 2014: 328-333

[7] C. Tsai, H. Wu, F. Tai, and Y. Chen, Distributed consensus formation control with collision and obstacle avoidance for uncertain networked omnidirectional multi-robot systems using fuzzy wavelet neural networks, *International Journal of Fuzzy Systems*, 19(5): 1-17, 2016.

[8] C. Chen, C. Ren, and T. Du, Fuzzy observed-based adaptive consensus tracking control for second-order multi-agent systems with heterogeneous nonlinear dynamics, *IEEE Transactions on Fuzzy Systems*, 24(4): 906-915, 2016.

[9] V. Utkin, and J. Shi, Integral sliding mode in systems operating under uncertainty conditions, *IEEE Conference on Decision & Control*, 4(4): 4591-4596, 2002.

[10] Y. Feng, X. Yu, and Z. Man, Nonsingular terminal sliding mode control of rigid manipulators, *Automatica*, 38: 2159-2167, 2006.

[11] Y. Wang. L. Gu, Y Xu, and X. Cao, Practical tracking control of robot manipulators with continuous fractional-order nonsingular terminal sliding mode, *IEEE Transactions on Industrial Electronics*, 63(10): 6194-6204, 2016.

[12] J. Baek, M. Jin, and S. Han, A new adaptive sliding-mode control scheme for application to robot manipulators, *IEEE Transactions on Industrial Electronics*, 63(6): 3628–3637, 2016.

[13] J. Niu, Q. Yang, X Wang, and R. Song, Sliding mode tracking control of a wire-driven upper-limb rehabilitation robot with nonlinear disturbance observer, *Frontiers in Neurology,* 8: 1-10, 2017.

[14] C. Yang, T. Teng, B. Xu, Z. Li, J. Na, and C. Su, Global adaptive tracking control of robot manipulators using neural networks with finite-time learning convergence, *International Journal of Control, Automation and Systems,* 15(4): 1916–1924, 2017.