WILEY | Hindawi

Research Article

Motion Planning for Omnidirectional Wheeled Mobile Robot by Potential Field Method

Weihao Li,[1] Chenguang Yang,[1,2] Yiming Jiang,[1] Xiaofeng Liu,[3,4,5] and Chun-Yi Su[1]

[1]*Key Lab of Autonomous Systems and Networked Control, Ministry of Education, South China University of Technology, Guangzhou 510640, China*
[2]*Zienkiewicz Centre for Computational Engineering, Swansea University, Swansea SA1 8EN, UK*
[3]*College of IoT Engineering, Hohai University, Nanjing, China*
[4]*Jiangsu Key Laboratory of Special Robots, Hohai University, Nanjing, China*
[5]*Changzhou Key Laboratory of Robotics and Intelligent Technology, Changzhou, China*

Correspondence should be addressed to Chenguang Yang; cyang@ieee.org

In this paper, potential field method has been used to navigate a three omnidirectional wheels' mobile robot and to avoid obstacles. The potential field method is used to overcome the local minima problem and the goals nonreachable with obstacles nearby (GNRON) problem. For further consideration, model predictive control (MPC) has been used to incorporate motion constraints and make the velocity more realistic and flexible. The proposed method is employed based on the kinematic model and dynamics model of the mobile robot in this paper. To show the performance of proposed control scheme, simulation studies have been carried to perform the motion process of mobile robot in specific workplace.

1. Introduction

In recent decades, omnidirectional mobile robot (OMR) has attracted increasing attention and investigation from the research communities [1–3]. One of advantages of OMR using omnidirectional wheels is that it does not have nonholonomic constraint which exists in differentially driven mobile robot [4–6]. With the input of the rotating speed of each omnidirectional wheel, the mobile robot can easily move wherever the user wants. This simplifies the control law which can be achieved easily. As it is shown in Figure 1, omnidirectional wheel consists of wheel and rollers, which means that the speed of the whole omnidirectional wheel is the combination of wheel speed and roller speed. Robot's control is very complicated, and sometimes it is necessary to consider state constraint of the robot to complete the control design [7, 8].

Since the path planning problem has been put forward, it has been studied by numbers of researchers [9]. A large number of research results have been proposed. The path planning algorithm develops from the earliest grid method, artificial potential field method [10], visibility graph [11] to

C-space method [12], A^* algorithm, and D^* algorithm [13]. Now, it is also studied to combine fuzzy logic algorithm [14, 15], adaptive algorithm [16, 17], and neural network algorithm [18–21]. In recent years, potential field method is more and more mature and widely used in omnidirectional mobile robots, because of its logical simplicity and obstacle avoidance capability. Many researches have proved the excellent capability of navigation and obstacle avoidance [22–24]. Hence, potential field method is utilized in this paper for the motion planning of omnidirectional wheeled mobile robot. To overcome the local minima problem and the goals nonreachable with obstacles nearby (GNRON) problem, the repulsive potential functions for motion planning contain the distance between robot and obstacle.

A popular way to control a mobile robot is to design the kinematic control based only on the kinematics equation [25–27]. Since 1995, people have put forward an integral dynamics model of a mobile robot [4]. Using dynamics model to control robot's motion is a common way [28–31]. This paper combines the kinematics as well as dynamics equation of the omnidirectional wheel and potential field method, to
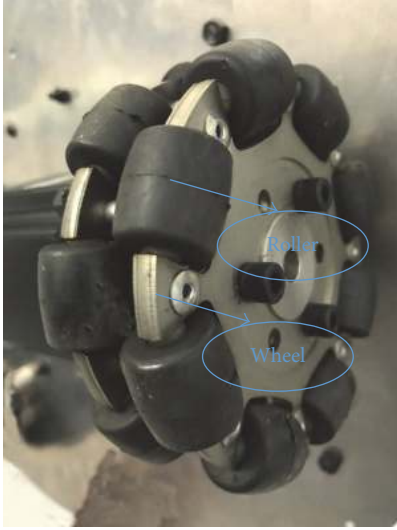
FIGURE 1: The structure of i omnidirectional wheel.

FIGURE 2: The motion relationship between system center and the i omnidirectional wheel.

control and navigate the mobile robot. In addition to these contributions in this paper, model predictive control (MPC) is utilized in motion planning for robust controller performance [32–35]. This paper is organized as follows: in Section 2, the kinematics equation of omnidirectional wheel and how the mobile robot built with 3 omnidirectional wheels can achieve the omnidirectional motion are discussed. In Section 3, the dynamics model of 3-omnidirectional wheel mobile robot is explained. In Section 4, a novel potential field method, which can overcome the GNRON problem, is introduced. In Section 5, MPC has been introduced. Both kinematics model and dynamics model have been applied in MPC. In Section 6, the simulations illustrating the effectiveness of the proposed method are presented. Finally, conclusion is given in Section 7.

2. The Kinematics Equation of Omnidirectional Wheel

From Figure 2, the following equation can be obtained:

$$\begin{bmatrix} v'_{ix} \\ v'_{iy} \end{bmatrix} = \begin{bmatrix} 0 & \sin \alpha_i \\ r & \cos \alpha_i \end{bmatrix} \begin{bmatrix} \omega_i \\ v_{ir} \end{bmatrix} = K_{i1} \begin{bmatrix} \omega_i \\ v_{ir} \end{bmatrix}, \qquad (1)$$

where v_{ix}, v_{iy}, and ω_i are generalized velocity of point O'_i in Cartesian coordinate system and v'_{ix}, v'_{iy}, and ω'_i are generalized velocity of point O'_i in $x'O'y'$ coordinate system. v_{ir} is the ith roller's central velocity vector.

When the ith omnidirectional wheel's central speed is mapped to Cartesian coordinate system, then

$$\begin{bmatrix} v_{ix} \\ v_{iy} \end{bmatrix} = \begin{bmatrix} \cos \eta_i & -\sin \eta_i \\ \sin \eta_i & \cos \eta_i \end{bmatrix} \begin{bmatrix} v'_{ix} \\ v'_{iy} \end{bmatrix} = K_{i2} K_{i1} \begin{bmatrix} \omega_i \\ v_{ir} \end{bmatrix}. \qquad (2)$$
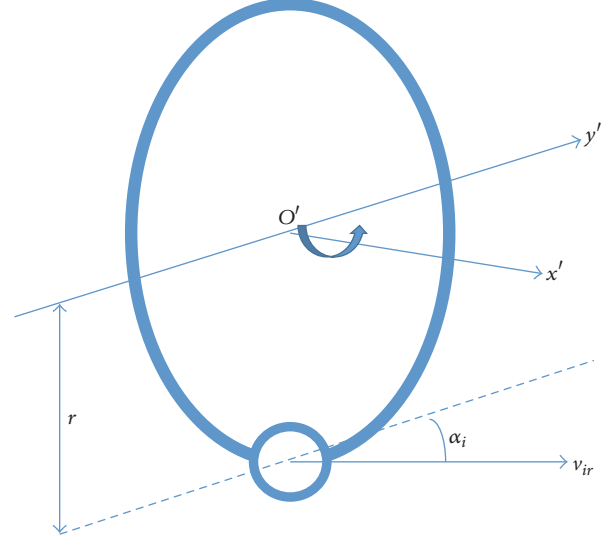
The system moves in two-dimensional space, so, according to geometric relationship and [36], the wheels' speed can be represented by

$$\begin{bmatrix} v_{ix} \\ v_{iy} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -l_{iy} \\ 0 & 1 & l_{ix} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = K_{i3} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \omega \end{bmatrix}, \qquad (3)$$

where l_{ix} and l_{iy} are the position of the ith wheel's mass point in xOy coordinate. According to (2)-(3), the system inverse kinematics equations can be defined as

$$K_{i2} K_{i1} \begin{bmatrix} \omega_i \\ v_{ir} \end{bmatrix} = K_{i3} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \omega \end{bmatrix}, \quad i = 1, 2, 3, \qquad (4)$$

where $\det(K_{i1})$ is not zero, so as the $\det(K_{i2})$. Define $K_i = [K_{i2}]^{-1}[K_{i1}]^{-1}K_{i3}$, $l_{ix} = l_i \cos \beta_i$, and $l_{iy} = l_i \sin \beta_i$. β_i is the angle between OO'_i and x-axis. Then the inverse kinematics equation of ith omnidirectional wheel is

$$\begin{bmatrix} \omega_i \\ v_{ir} \end{bmatrix} = [K_{i2}]^{-1}[K_{i1}]^{-1}K_{i3} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \omega \end{bmatrix}; \qquad (5)$$

define $\gamma_i = \eta_i - \alpha_i$; then

$$K_i = \frac{1}{-r \sin \alpha_i}$$

$$\cdot \begin{bmatrix} \cos(\gamma_i) & \sin(\gamma_i) & -l_{iy} \cos(\gamma_i) + l_{ix} \sin(\gamma_i) \\ r \cos \eta_i & -r \sin \eta_i & l_{iy} \cdot r \cos \eta_i - l_{ix} \cdot r \sin \eta_i \end{bmatrix}. \qquad (6)$$
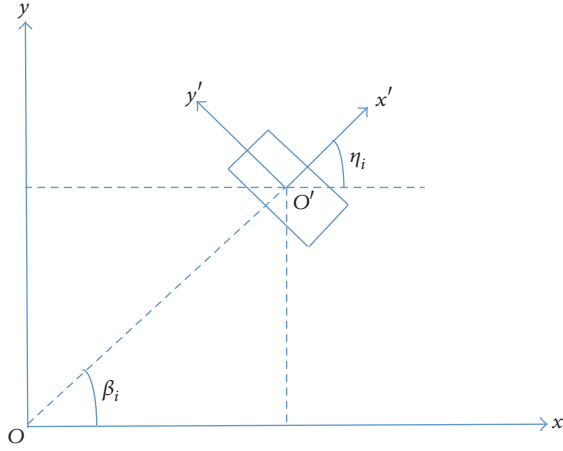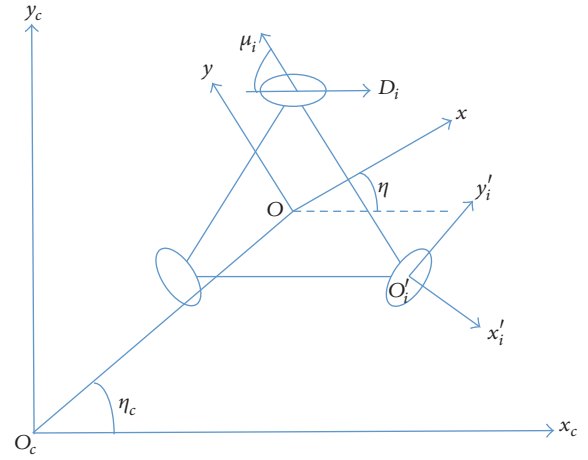
FIGURE 3: The structure of ith omnidirectional wheel.

Define

$$A_1 = \begin{bmatrix} \dfrac{\cos(\gamma_1)}{\sin \alpha_1} \\ \dfrac{\cos(\gamma_2)}{\sin \alpha_2} \\ \dfrac{\cos(\gamma_3)}{\sin \alpha_3} \end{bmatrix}$$

$$A_2 = \begin{bmatrix} \dfrac{\sin(\gamma_1)}{\sin \alpha_1} \\ \dfrac{\sin(\gamma_2)}{\sin \alpha_2} \\ \dfrac{\sin(\gamma_3)}{\sin \alpha_3} \end{bmatrix} \qquad (7)$$

$$A_3 = \begin{bmatrix} \dfrac{l_1 \sin(\gamma_1 - \beta_1)}{\sin \alpha_1} \\ \dfrac{l_2 \sin(\gamma_2 - \beta_2)}{\sin \alpha_2} \\ \dfrac{l_3 \sin(\gamma_3 - \beta_3)}{\sin \alpha_3} \end{bmatrix}.$$

The inverse kinematics solution of wheel speed to system center is

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{-r} \begin{bmatrix} A_1 & A_2 & A_3 \end{bmatrix} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \omega \end{bmatrix}. \qquad (8)$$

The Jacobian matrix systems inverse kinematics equation is

$$R = \frac{1}{-r} \begin{bmatrix} A_1 & A_2 & A_3 \end{bmatrix}. \qquad (9)$$

According to Figures 2–4, since this paper discusses a mobile robot built by three omnidirectional wheels, α_i, β_i, and η_i, are fixed. The actual values are shown in Table 1.

FIGURE 4: All three coordinates in OMR system.

TABLE 1: A table with notes.

Parameter	β_1	β_2	β_3	η_1	η_2	η_3	α_1	α_2	α_3
Value	60°	180°	300°	60°	180°	300°	90°	90°	90°

From Table 1 and specification that $r = 50.67$ mm and $l = 118.18$ mm, the actual parameter of OMR can be substituted into (8); then

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} -0.0170 & 0.00987 & 2.3323 \\ 0 & -0.0197 & 2.3323 \\ 0.0171 & 0.00987 & 2.3323 \end{bmatrix} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \omega \end{bmatrix}. \qquad (10)$$

From (10), rank$(R) = 3$, which means that this robot can achieve omnidirectional movement.

3. The Dynamics Equation of OMR

Three coordinates $x_c O_c y_c$, xOy, and $x'_i O'_i y'_i$ are constructed as in Figure 4. O_c is a specific point in the workspace, and O is the central point of the mobile robot while O'_i is the central point of each wheel. η is the angle between the front of OMR and X_c. The vector $S_c = [x_c, y_c]^T$ indicates the position of O. According to Newton's second law,

$$\begin{aligned} m\ddot{x}_c &= F_{x_c} \\ m\ddot{y}_c &= F_{y_c}, \end{aligned} \qquad (11)$$

equivalent to

$$M\ddot{S}_c = F_c. \qquad (12)$$

$F = [F_{x_c}, F_{y_c}]^T$ is a force vector on the central point of mobile robot in the Cartesian coordinate. $M = \text{diag}(m, m)$ is a symmetric positive definite matrix, and m is the mass of mobile robot.

The transfer matrix which transfers coordinate $x_c O_c y_c$ into coordinate xOy is

$$T_c = \begin{bmatrix} \cos \eta & -\sin \eta \\ \sin \eta & \cos \eta \end{bmatrix}. \qquad (13)$$

Consequently,

$$\dot{S}_c = T_c \dot{s}$$
$$\dot{F}_c = T_c \dot{f}$$
(14)

$s = [x, y]^T$ and $f = [f_x, f_y]$, respectively, mean the central point O's displacement and force vector in mobile coordinate xOy. Therefore, (12) becomes

$$M \left(T_c^T \dot{T}_c \dot{s} + \ddot{s} \right) = f.$$
(15)

Finally, the dynamics of the omnidirection mobile robot can be described as

$$m \left(\ddot{x} - \ddot{y} \dot{\eta} \right) = f_x$$
$$m \left(\ddot{y} + \ddot{x} \dot{\eta} \right) = f_y$$
$$I_v \ddot{\eta} = M_I,$$
(16)

where I_v and M_I are the moment of inertia of mobile robot around its central axis and the corresponding torque, respectively. Among them, f_x, f_y, and M_I can be drawn by

$$f_x = - \left(N_1^w + N_2^w \right) \sin \mu + N_3$$
$$f_y = \left(N_1^w - N_2^w \right) \cos \mu,$$
(17)

where μ is the angle between wheel and y-axis, and $\mu_1 = 30°$, $\mu_2 = 30°$, and $\mu_3 = 90°$.

$$M_I = l_1 N_1^w + l_2 N_2^w + l_3 N_3^w.$$
(18)

According to [37], the dynamics model of drive system of each wheel is assumed as

$$I_w \dot{\omega}_i + \xi \omega_i = h \tau_i - r N_i^w \quad (i = 1, 2, 3),$$
(19)

where N_i^w is the drive power of each wheel. I_ω is the moment of inertia of wheel around its central axis. ξ is viscous friction constant between wheel and ground. $\dot{\omega}_i$ is angular acceleration of each wheel. r is radius of the wheel. h is drive factor. τ_i is the input torque of each wheel.

The speed of each omnidirectional mobile robot's wheel v_i can be described as $r \omega_i$. According to [38], the dynamics model of OMR can be described as the following equation:

$$\ddot{q} = A(q) \dot{q} + B(q) \tau,$$
(20)

where

$$A(q)$$
$$= \begin{bmatrix} \dfrac{-3\xi}{3I_w + 2Mr^2} & \dfrac{-3I_w \dot{\eta}}{3I_w + 2Mr^2} & 0 \\[2mm] \dfrac{3I_w \dot{\eta}}{3I_w + 2Mr^2} & \dfrac{3\xi}{3I_w + 2Mr^2} & 0 \\[2mm] 0 & 0 & \dfrac{-3\xi l^2}{3I_w l^2 + I_v r^2} \end{bmatrix}$$
(21)

$$B(q)$$
$$= \begin{bmatrix} \dfrac{-hr\kappa_1}{3I_w + 2Mr^2} & \dfrac{hr\kappa_1}{3I_w + 2Mr^2} & \dfrac{2hr \cos \eta}{3I_w + 2Mr^2} \\[2mm] \dfrac{hr\kappa_2}{3I_w + 2Mr^2} & \dfrac{hr\kappa_3}{3I_w + 2Mr^2} & \dfrac{2hr \sin \eta}{3I_w + 2Mr^2} \\[2mm] \dfrac{hrl}{3I_w l^2 + I_v r^2} & \dfrac{hrl}{3I_w l^2 + I_v r^2} & \dfrac{hrl}{3I_w l^2 + I_v r^2} \end{bmatrix}.$$

Let us define $\kappa_1 = \sqrt{3} \sin \eta - \cos \eta$, $\kappa_2 = -\sqrt{3} \cos \eta - \sin \eta$, and $\kappa_3 = \sqrt{3} \cos \eta - \sin \eta$.

4. Potential Field for OMR's Motion Planning

Using the potential field algorithm for OMR's path planning will be modified to produce a virtual force to navigate mobile robot and obstacle avoidance. For simple theoretical analysis, mobile robot is considered as a mass point and moves in two-dimensional space whose position can be denoted by $s = [x, y]^T$. The distance as well as the angle between robot and goal, robot and obstacles can be detected by ultrasonic sensors. Inspired by [23], then the attractive potential function caused by goal can be calculated by the following equation:

$$U_{\text{att}}(s) = \frac{1}{2} a_{\text{att}} d^c \left(s, s_{\text{goal}} \right),$$
(22)

where a_{att} is a positive scaling factor, $d^c(s, s_{\text{goal}}) = \|(s_{\text{goal}} - s)\|$ is the distance between the OMB's mass point and the goal s_{goal}, and $c = 1$ or 2. For $c = 2$, the attractive force is

$$F_{\text{att}}(s) = -\nabla U_{\text{att}}(s) = a_{\text{att}} \left(s_{\text{goal}} - s \right) = F_{\text{att}} n_{\text{RG}}.$$
(23)

The repulsive potential function is

$$U_{\text{rep}}(s)$$
$$= \begin{cases} \dfrac{1}{2} a_{\text{rep}} \left(\dfrac{1}{d(s, s_{\text{obs}})} - \dfrac{1}{d_0} \right)^2 d^n \left(s, s_{\text{goal}} \right), & \text{if } d(s, s_{\text{obs}}) \leq d_0 \\[3mm] 0, & \text{if } d(s, s_{\text{obs}}) > d_0. \end{cases}$$
(24)

where a_{rep} is a positive scaling factor, $d(s, s_{\text{obs}})$ is the minimal distance between the OMB's mass point q and the hindrance, d_0 denotes the level of the influence of the hindrance to the robot and it is defined as a positive constant, and n is a positive constant. The repulsive force is

$$F_{\text{rep}}(s) = -\nabla U_{\text{rep}}(s)$$
$$= \begin{cases} F_{\text{rep}1} n_{\text{OR}} + F_{\text{rep}2} n_{\text{RG}}, & \text{if } d(s, s_{\text{obs}}) \leq d_0 \\[2mm] 0, & \text{if } d(s, s_{\text{obs}}) > d_0. \end{cases}$$
(25)

$n_{OR} = \nabla d(s, s_{obs})$ and $n_{RG} = -\nabla d(s, s_{goal})$ are two unit vectors pointing from the obstacle to the OMR and from the OMR to the goal. According to (23) and (25), F_{total} is drawn by the following equation:

$$F_{total} = F_{att}(s) + F_{rep}(s)$$
$$= F_{att}n_{RG} + F_{rep1}n_{OR} + F_{rep2}n_{RG}. \tag{26}$$

According to [23], for the robot with 3 omnidirectional wheels, the real input is the 3 angular velocities of the omnidirectional wheels, ω_1, ω_2, and ω_3, which satisfies (8), and v_{x_c} and v_{y_c} have the following relationship with v and ω:

$$v_{x_c} = v \cos \eta$$
$$v_{y_c} = v \sin \eta. \tag{27}$$

The mobile robot needs to decelerate as soon as it nears the obstacle, while its velocity will be higher when it is far from the obstacle, so the robot's velocity is chosen by the distance between OMB and obstacle. Thus the velocity of the OMR in $x_c O_c y_c$ can be determined by

$$v = \begin{cases} v_o \\ \dfrac{d(s, s_{obs})}{d_0} v_o, \end{cases} \tag{28}$$

where v_o is the optimal velocity of the robot.

As the total force F_{total} can be calculated, its angle η_f is known. The difference angle between η_f and the orientation of the robot η_c is

$$\eta_e = \eta_f - \eta_c. \tag{29}$$

Thus the angular velocities ω can be ensured by

$$\omega = K\eta_e; \tag{30}$$

define K as a positive gain.

5. Model Predictive Control for Omnidirection Mobile Robot

In recent years, MPC has been widely used in motion control Internet of things applications [35, 39]. MPC has low requirements for model's accuracy and it is suitable for step response model and linear and nonlinear model. The control problem is described as a cost function's optimization problem. The input which is constrained by some specific conditions and minimizes the cost function is the optimal input. One of MPC's advantage is its rolling optimization [40] that means, according to its reference, it can optimize a cost function to get an optimal input vector at every sample time. According to the MPC method introduced by [41], because the reference is produced by potential field, what we need is a discrete-time model with constraints. In the following section, according to [35] two discrete-time controllers, kinematic controller, and dynamics controller are proposed.

5.1. Kinematic Controller. With (8) and $v = r\omega$, then the kinematics model of mobile robot can be transformed into

$$v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = -\begin{bmatrix} A_1 & A_2 & A_3 \end{bmatrix} \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \omega \end{bmatrix}. \tag{31}$$

Define $q = [x_c, y_c, \eta]^T$ as the state of mobile robot in $x_c O_c y_c$ and $S(q) = (rR)^{-1}$. Therefore

$$\dot{q} = -S(q) v. \tag{32}$$

With the help of zero-order hold (ZOH), a continuous-time system can be described as a discrete-time form

$$q(j + 1) = q(j) + \dot{q}(t) T. \tag{33}$$

with a sampling period T. According to (31), (32), and (33) can be rewritten as

$$q(j+1) = \begin{bmatrix} x_c(j+1) \\ y_c(j+1) \\ \eta(j+1) \end{bmatrix}$$
$$= \begin{bmatrix} x_c(j) - 1.1547Tv_1 - 0.5785Tv_2 + 0.5777Tv_3 \\ y_c(j) - 1.002Tv_2 + 0.334Tv_3 \\ \eta(j) + 0.0028Tv_3 \end{bmatrix} \tag{34}$$

The cost function for the MPC can be defined as

$$J(q, u) = \sum_{k=j}^{j+N-1} L_K(q(k), u(k) + F(q(j+N))), \tag{35}$$

where $L_K(q, u)$ is the stage cost.

$$L_K(q, u) = \sum_{k=1}^{N} q^T(k+j \mid j) Q_K q(k+j \mid j)$$
$$+ \sum_{k=0}^{N_u-1} \Delta u^T(k+j \mid j) R_K \Delta u(k+j \mid j), \tag{36}$$

where N is prediction horizon where $N \geq 1$ and N_u is control horizon where $1 \leq N_u \leq N$. Q_K and R_K are appropriate weighting matrices. $q(k + j \mid j)$ means the predicted state of the OMR and $\Delta u(k + j \mid j)$ means the input increment of the controller. $L_K(q, u)$ is the most used standard quadratic form in practice. By way of solving the following finite-horizon optimal control problem (FHOCQ) online:

$$u^* = \arg \min_u \{J(q \cdot u)\}. \tag{37}$$

The current control $u(j) = [v(j), \omega(j)]^T$ can be ensured at the instant time j. Because the torques generated by motors are limited by the performance of the motors, $u(j)$ has upper bound and lower bound and the change of $u(j)$ is also constrained. Thus

$$u_{min} \leq u(j) \leq u_{max}$$
$$\Delta u_{min} \leq \Delta u(j) \leq \Delta u_{max} \tag{38}$$
$$q_{min} \leq q(j) \leq q_{max}.$$

According to [42], (38) can be transformed into

$$h_i^- - \widehat{P}_i \leq \widehat{u}(j) \leq h_i^+ - \widehat{P}_i$$
$$h_i^- \leq q(j) \leq h_i^+. \tag{39}$$

The kinematic equation (34) can be described into the following form:

$$q(j+1) = g_1(q(j)) + g_2(q(j))u(j), \tag{40}$$

where g_1 and g_2 are the continuous nonlinear function, $g_1(0) = 0$, $q = [q_1, q_2, q_3]^T = [x_c, y_c, \eta]^T$ is the state vector, $u = [u_1, u_2]^T = [v, \omega]^T$ is the input vector, and

$$g_1(q) = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix},$$

$$g_2 = T \begin{bmatrix} -1.1547 & -0.5785 & 0.5777 \\ 0 & -1.002 & 0.334 \\ 0 & 0 & 0.0028 \end{bmatrix}. \tag{41}$$

Define the following vectors:

$$\overline{q} = [q(j+1 \mid j), \ldots, q(j+N \mid j)]^T \in R^{3N}$$

$$\overline{u}(j) = [u(j+1 \mid j), \ldots, u(j+N \mid j)]^T \in R^{2N_u} \tag{42}$$

$$\Delta \overline{u}(j) = [\Delta u(j+1 \mid j), \ldots, \Delta u(j+N \mid j)]^T$$
$$\in R^{2N_u}.$$

The predicated output can be determined by the following form:

$$\overline{x} = G\Delta\overline{u}(j) + \widetilde{g}_1 + \widetilde{g}_2, \tag{43}$$

where

$$G = \begin{bmatrix} g_2(q(\overline{N}_1)) & \cdots & 0 \\ g_2(q(\overline{N}_2)) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ g_2(q(\overline{N}_N)) & \cdots & g_2(q(\overline{N}_N)) \end{bmatrix}$$

$$\widetilde{g}_1 = \begin{bmatrix} g_1(q(\overline{N}_1)) \\ g_1(q(\overline{N}_2)) \\ \vdots \\ g_1(q(\overline{N}_N)) \end{bmatrix} \in R^{3N} \tag{44}$$

$$\widetilde{g}_2 = \begin{bmatrix} g_2(q(\overline{N}_1))u(j-1) \\ g_2(q(\overline{N}_2))u(j-1) \\ \vdots \\ g_2(q(\overline{N}_N))u(j-1) \end{bmatrix} \in R^{3N};$$

define $\overline{N}_i = j - 1 + i \mid j - 1$.

Hence, the original optimization problem (35) can be transformed into

$$\min \|\overline{q}(j)\|_{Q_K}^2 + \|\Delta \overline{u}(j)\|_{R_K}^2$$
$$= \min \|G\Delta\overline{u}(j) + \widetilde{g}_1 + \widetilde{g}_2\|_{Q_K}^2 + \|\Delta\overline{u}(j)\|_{R_K}^2 \tag{45}$$

subject to

$$\Delta\overline{u}_{\min} \leq \Delta\overline{u}(j) \leq \Delta\overline{u}_{\max}$$
$$\overline{u}_{\min} \leq \overline{u}(j-1) \leq \Delta\overline{u}_{\max}$$
$$\overline{u}_{\min} \leq \overline{u}(j-1) + I\Delta\overline{u}(j) \leq \Delta\overline{u}_{\max} \tag{46}$$
$$\overline{x}_{\min} \leq G\Delta\overline{u}(j) + \widetilde{g}_1 + \widetilde{g}_2 \leq \overline{x}_{\max}.$$

Problem (45) can be rewritten as a QP problem

$$\min \frac{1}{2}\Delta\overline{u}^T\overline{W}\Delta\overline{u} + \overline{H}^T\Delta\overline{u} \tag{47}$$

subject to

$$E\Delta\overline{u} \leq \widetilde{b}$$
$$\Delta\overline{u}_{\min} \leq \Delta\overline{u} \leq \Delta\overline{u}_{\max}, \tag{48}$$

where the coefficients are

$$\overline{W} = 2\left(G^T Q_K G + R_K\right)$$
$$\overline{H} = -2G^T Q_K \left(\widetilde{g}_1 + \widetilde{g}_2\right) \tag{49}$$
$$E = \begin{bmatrix} -I & I & -G & G \end{bmatrix}^T.$$

5.2. Dynamics Controller. According the dynamics model of OMR, then

$$\ddot{x}_c = \frac{-3I_w \dot{y}_c \dot{\eta} - 3\xi \dot{x}_c - hr\kappa_1\tau_1 + hr\kappa_1\tau_2 + 2hr\tau_3}{2Mr^2 + 3I_w}$$

$$\ddot{y}_c = \frac{3I_w \dot{x}_c \dot{\eta} + 3\xi \dot{y}_c - hr\kappa_2\tau_1 + hr\kappa_3\tau_2 + 2hr\tau_3}{2Mr^2 + 3I_w} \tag{50}$$

$$\ddot{\eta} = \frac{-3\xi l^2 \eta^2 + hlr(\tau_1 + \tau_2 + \tau_3)}{I_v r^2 + 3I_w r^2}.$$

Applying (33) into (50) and $[\ddot{x}, \ddot{y}, \ddot{\eta}] = [\dot{v}_{x_c}, \dot{v}_{y_c}, \dot{\omega}]$ the discrete-time dynamics model of the OMR can be described as

$$v_x(j+1) = v_x(j) + \ddot{x}T$$
$$v_y(j+1) = v_y(j) + \ddot{y}T \tag{51}$$
$$\omega(j+1) = \omega(j) + \ddot{\eta}T.$$

According to $u = [v_{x_c}, v_{y_c}, \omega]^T$ and $q = [x_c, y_c, \eta]^T$, then

$$u(j+1) = u(j) + \ddot{q}T \tag{52}$$

Applying (52) into (40), then, we can draw

$$q(j+2)$$
$$= g_1(q(j)) + g_2(q(j))u(j) \quad (53)$$
$$+ g_2(g_1(q(j)) + g_2(q(j))u(j))u(j+1)$$

subject to

$$u_{min} \leqslant u(j) \leqslant u_{max}$$
$$\tau_{min} \leqslant \tau(j) \leqslant \tau_{max} \quad (54)$$
$$\ddot{q}_{min} \leqslant \ddot{q}(j) \leqslant \ddot{q}_{max},$$

where, respectively, min and max means the lower bounds and upper bounds. τ is the motor inputs. u is the velocity. $\ddot{q}$ is the acceleration.

According to (53), the predictive model can be formulated as

$$q(k+j \mid j) = g_1(q(k+j-2 \mid j))$$
$$+ g_2(q(k+j-2 \mid j))u(k+j-2 \mid k)$$
$$+ g_2(g_1(q(k+j-2 \mid j)) \quad (55)$$
$$+ g_2(q(k+j-2 \mid j))u(k+j-2 \mid k))u(k+j$$
$$-1 \mid k),$$

where $k \subseteq [1, \overline{N}]$.

The quadratic objective function (QBF) of the robot's state and the motor input under a predictive horizon $\overline{N}$ and a control horizon $\overline{N}_u$ can be determined by the following equation:

$$L_D(k) = \sum_{k=1}^{\overline{N}} q^T(k+j \mid j)\overline{Q}_D q(k+j \mid j)$$
$$\quad (56)$$
$$+ \sum_{k=1}^{\overline{N}_u} \tau^T(k+j-2 \mid j)\overline{R}_D \tau(k+j-2 \mid j),$$

where $\overline{Q}_D$ and $\overline{R}_D$ are appropriate weighting matrices.

Hence, the dynamics predictive motor torque can be obtained by

$$\tau^* = \arg\min_\tau L_D(j) \quad (57)$$

subject to

$$\tau_{min} \leqslant \tau(k+j \mid j) \leqslant \tau_{max}, \quad k \subseteq [0, \overline{N}-2]$$
$$\quad (58)$$
$$\ddot{q}_{min} \leqslant \ddot{q}(k+j \mid j) \leqslant \ddot{q}_{max}, \quad k \in [1, \overline{N}].$$

6. Experimental Part

In this section, we design a 100 m × 100 m workplace within 6 obstacles (O_i, $i = 1, 2, 3, 4, 5, 6$). We define the velocity of the OMR as 1 m/s and the start point of the OMR as (3, 90) while the destination is (70, 28). We define the prediction horizon $N = 3$ and the control horizon $N_u = 2$. As the inputs of kinematic controller and dynamics controller are different, we set up a different constraint. The save distance between the OMR and obstacle is adjustable and the accurate value of the OMR's model is based on the OMR we built physically. The simulation is carried out in MATLAB.

6.1. Simulation of Kinematics Controller. In this section, simulation is carried out based on kinematics model. The OMR is navigated by potential field method combining MPC. In this simulation, we assume the goal position is already known, and the start position and start velocity are defined in advance.

The process of this simulation is shown by the following flow chart in Figure 5, where O_i means the ith obstacle. First, we define the originate position of the mobile robot and the originate velocity. Then, we sent the goal position to the mobile robot. With the help of potential field method, mobile robot can draw the next status, which is used as the reference for the MPC. After the MPC process, the predictive status is sent to mobile robot for navigation and to the previous process for the next reference.

The area of workplace is within 100 m × 100 m and there are some obstacles randomly distributed in it. With the help of potential field and MPC, the OMR can adjust its velocity and finally reach the goal which is shown by Figures 6 and 7.

According to Figures 6 and 7, we can see that the OMR successfully reaches the goal and can smoothly self-avoid the obstacle. The velocity of the OMR is shown by Figures 8, 9, and 10.

At the beginning of simulation, we define the originate velocity of the robot as 1. When the mobile robot starts moving, the velocity (v_t) is modified by MPC as v_t on the left of simulation.

Connecting Figures 6, 8, and 9, we can see that the moment the change of v_t is rapid is the moment that mobile robot encounters an obstacle and it needs to change its direction. Figure 10 is the velocities of three omnidirectional wheels; the composition of all three velocities is v_t.

According to potential field method, we can get the total force angle, and then we can determine η. The change of η is shown by Figure 11.

The result of potential field method in navigation is outstanding. The OMR can smoothly avoid obstacles and successfully guide itself to the goal. With supplement of MPC, the motion of mobile robot is constrained and the velocity becomes more realistic and flexible. The robustness of system is enhanced.

6.2. Simulation of Dynamics Controller. In this section, similar to the last section, we change kinematics model into dynamics model and simulate a similar process which OMR is navigated by potential field and MPC. In this simulation, we assume the goal position is already known, and the start position and start velocity are defined in advance too.

According Figures 12 and 13, it shows that, with the control of dynamics controller and navigation of potential field, the OMR achieves its goal and successfully avoids the
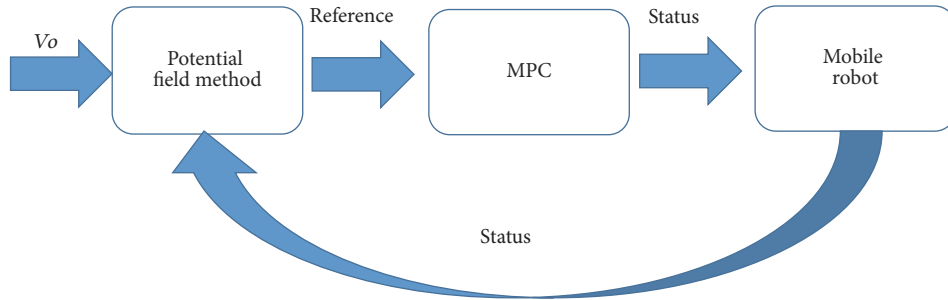
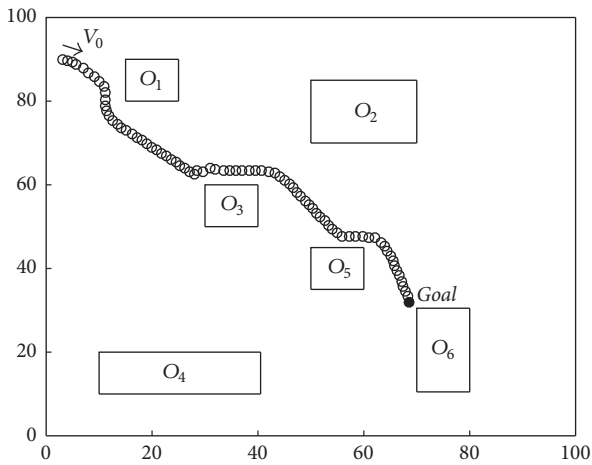FIGURE 5: The velocity of the OMR during the simulation.

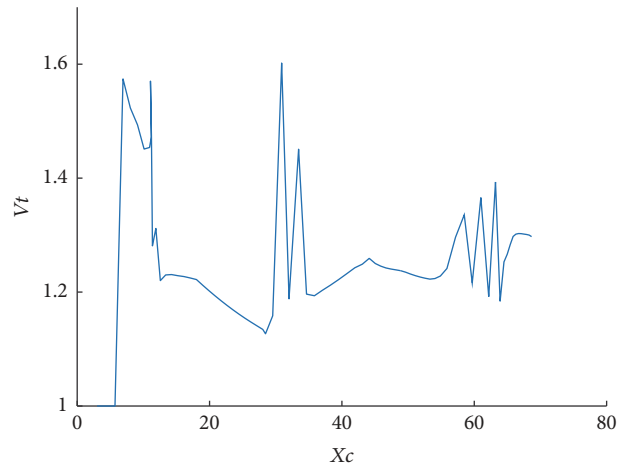FIGURE 6: The simulation of OMR controlled by kinematic controller moving in the workplace.

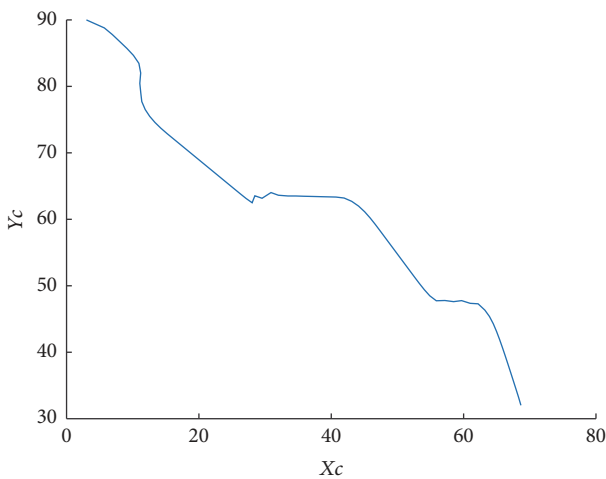FIGURE 8: The velocity of the OMR controlled by kinematic controller during the simulation.

FIGURE 7: The trajectory of the OMR controlled by kinematic controller in $X_cO_cY_c$ coordinate.
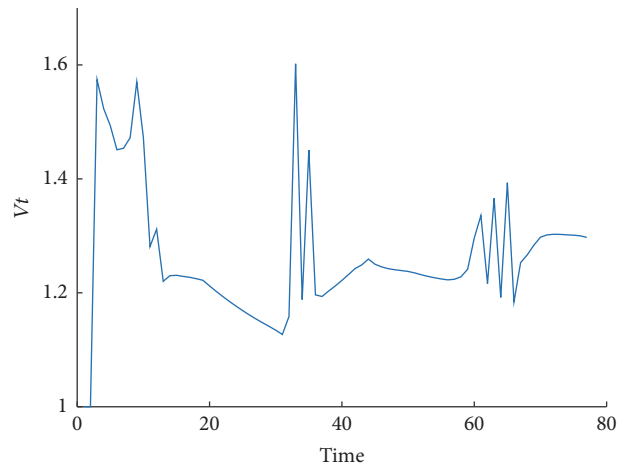

FIGURE 9: The change of velocity of the OMR controlled by kinematic controller.

obstacles. When the OMR approaches an obstacle, potential field algorithm generates a repulsive force which makes the OMR turn around and prevent itself from hitting the obstacle. With the help of MPC, the velocity is constrained, which can

prevent some value that is beyond the real produced by the potential field method from navigating the OMR.

Comparing Figures 14 and 15 to Figure 12, it is easy to see that when the velocity changes sharply is when the OMR is too close to the obstacle and it needs to slow down to avoid possible collision.
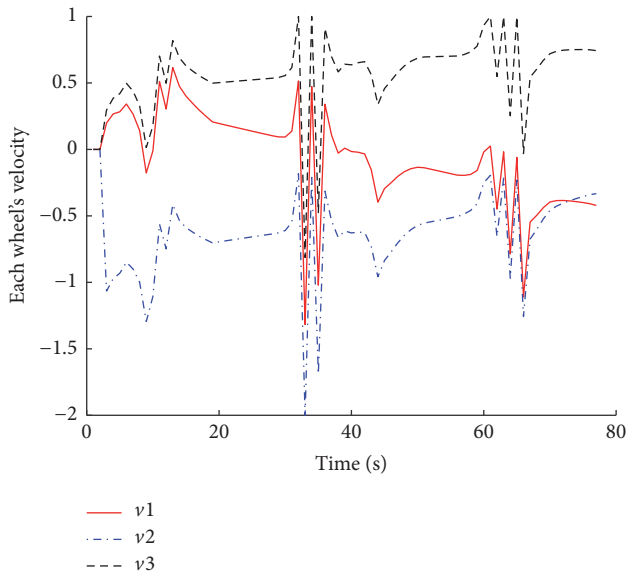
v1
v2
v3

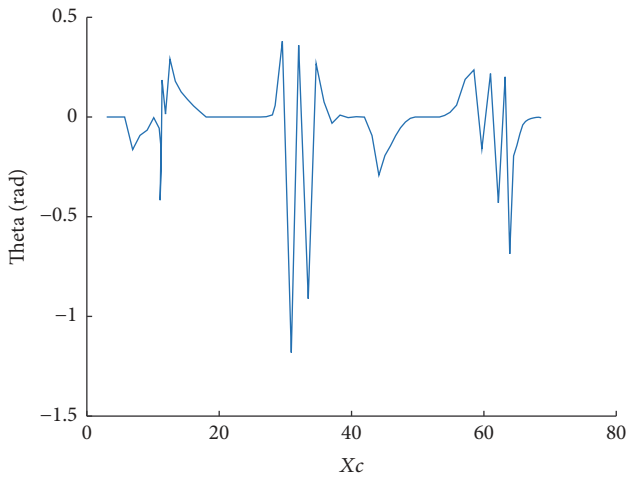FIGURE 10: Three omnidirectional wheels' velocity.

FIGURE 11: The angle with the change of robot's position is determined by potential field.
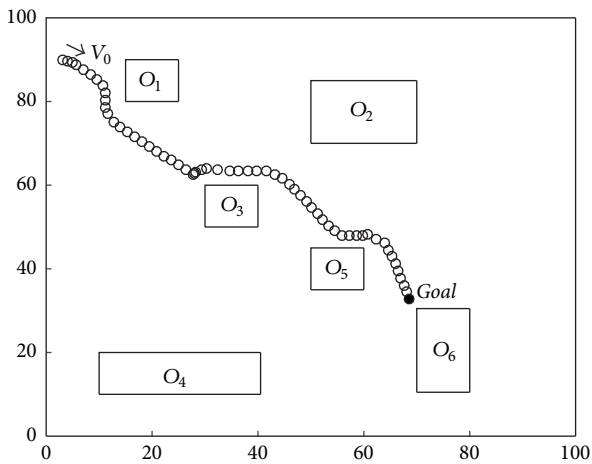

FIGURE 12: The simulation of the OMR controlled by dynamics controller moving in the workplace.
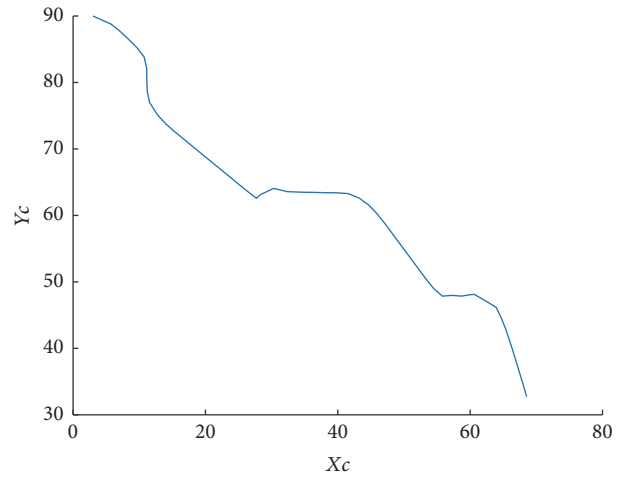

FIGURE 13: The trajectory of the OMR controlled by dynamics controller in $X_cO_cY_c$ coordinate.
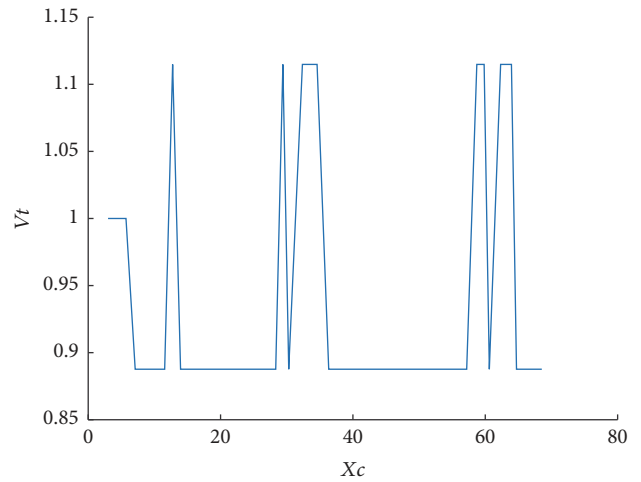

FIGURE 14: The velocity of the OMR controlled by dynamics controller during the simulation.
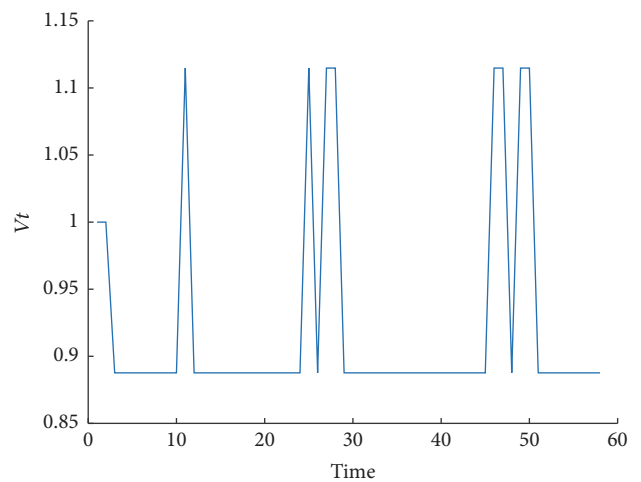

FIGURE 15: The change of velocity of the OMR controlled by dynamics controller.
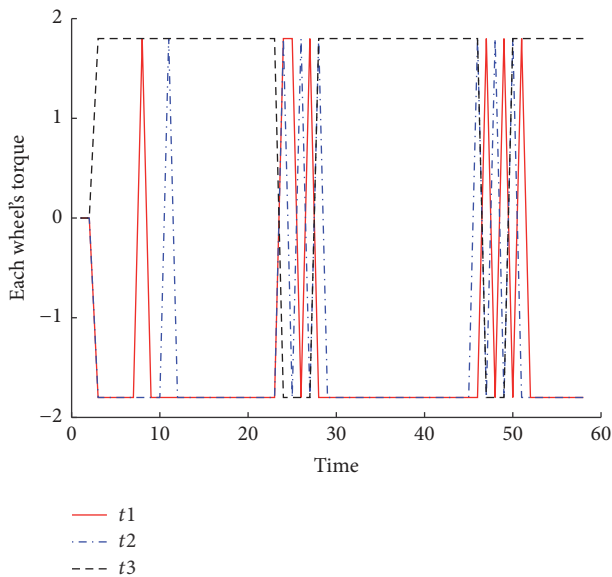
Figure 16: The change of torque of the OMR.

According to Figure 16, when the OMR needs to change its direction, it changes three torques, respectively, to generate a combined effort to steer its motion.

7. Conclusion

In this paper, a novel potential field method has been used to navigate a class of OMR. With the consideration of distance between the robot and the goal, robot, and obstacle, the GNRON problem is solved. In addition, it discusses the kinematic as well as dynamics model of mobile robot. For improving system's robustness, it combines potential field and MPC, so the motion planning is more complete. Finally, simulation results show that the proposed control scheme is more appropriate for omnidirectional mobile robot's navigation.

Disclosure

Chun-Yi Su is on leave from Concordia University, Canada.

Competing Interests

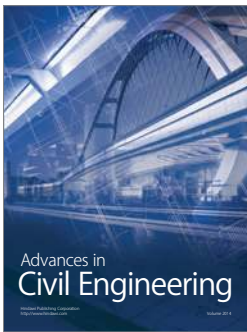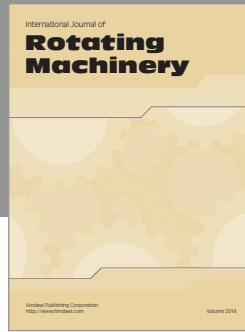The authors declare that they have no competing interests.

Acknowledgments

References

[1] R. L. Williams II, B. E. Carter, P. Gallina, and G. Rosati, "Dynamic model with slip for wheeled omnidirectional robots," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 3, pp. 285–293, 2002.

[2] J. E. Mohd Salih, M. Rizon, S. Yaacob, A. H. Adom, and M. R. Mamat, "Designing omni-directional mobile robot with mecanum wheel," *American Journal of Applied Sciences*, vol. 3, no. 5, pp. 1831–1835, 2006.

[3] Y. Liu, X. Wu, J. J. Zhu, and J. Lew, "Omni-directional mobile robot controller design by trajectory linearization," *Dynamics*, vol. 2, pp. 2–2, 2003.

[4] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot: backstepping kinematics into dynamics," in *Proceedings of the 34th IEEE Conference on Decision and Control*, pp. 3805–3810, IEEE, December 1995.

[5] C. Canudas de Wit and O. J. Sordalen, "Exponential stabilization of mobile robots with nonholonomic constraints," *IEEE Transactions on Automatic Control*, vol. 37, no. 11, pp. 1791–1797, 1992.

[6] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos, "Nonholonomic navigation and control of cooperating mobile manipulators," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 53–64, 2003.

[7] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 620–629, 2016.

[8] W. He, Y. Dong, and C. Sun, "Adaptive neural impedance control of a robotic manipulator with input saturation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 3, pp. 334–344, 2016.

[9] R. Cui, Y. Li, and W. Yan, "Mutual information-based multi-AUV path planning for scalar field sampling using multidimensional RRT∗," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 7, pp. 993–1004, 2016.

[10] M. C. Lee and M. G. Park, "Artificial potential field based path planning for mobile robots using a virtual obstacle concept," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '03)*, pp. 735–740, July 2003.

[11] L. E. Kavraki, P. Švestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[12] B. K. Choi, D. H. Kim, and R. B. Jerard, "C-space approach to tool-path generation for die and mould machining," *CAD Computer Aided Design*, vol. 29, no. 9, pp. 657–669, 1997.

[13] D. Ferguson and A. Stentz, "Using interpolation to improve path planning: the field D∗ algorithm," *Journal of Field Robotics*, vol. 23, no. 2, pp. 79–101, 2006.

[14] A. M. C. Smith, C. Yang, H. Ma, P. Culverhouse, A. Cangelosi, and E. Burdet, "Novel hybrid adaptive controller for manipulation in complex perturbation environments," *PLoS ONE*, vol. 10, no. 6, Article ID e0129281, 2015.

[15] Y. Jiang, C. Yang, and H. Ma, "A review of fuzzy logic and neural network based intelligent control design for discrete-time systems," *Discrete Dynamics in Nature and Society*, vol. 2016, Article ID 7217364, 11 pages, 2016.

[16] C. Yang, K. Huang, H. Cheng, Y. Li, and C.-Y. Su, "Haptic identification by ELM controlled uncertain manipulator," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, In press.

[17] C. Yang, Z. Li, and J. Li, "Trajectory planning and optimized adaptive control for a class of wheeled inverted pendulum vehicle models," *IEEE Transactions on Cybernetics*, vol. 43, no. 1, pp. 24–36, 2013.

[18] C. Yang, Y. Jiang, Z. Li, W. He, and C.-Y. Su, "Neural control of bimanual robots with guaranteed global stability and motion precision," *IEEE Transactions on Industrial Informatics*, 2017.

[19] C. Yang, X. Wang, Z. Li, Y. Li, and C. Su, "Teleoperation control based on combination of wave variable and neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017.

[20] C. Yang, X. Wang, L. Cheng, and H. Ma, "Neural-learning-based telerobot control with guaranteed performance," *IEEE Transactions on Cybernetics*, 2017.

[21] L. Zhang, Z. Li, and C. Yang, "Adaptive neural network based variable stiffness control of uncertain robotic systems using disturbance observer," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2236–2245, 2017.

[22] Z. Li, S. Zhao, J. Duan, C. Su, C. Yang, and X. Zhao, "Human cooperative wheelchair with brain machine interaction based on shared control strategy," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 1, pp. 185–195, 2017.

[23] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, pp. 615–620, 2000.

[24] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1398–1404, Sacramento, Calif, USA, April 1991.

[25] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, MIT Press, 2011.

[26] C.-C. Tsai, F.-C. Tai, and Y.-R. Lee, "Motion controller design and embedded realization for Mecanum wheeled omnidirectional robots," in *Proceedings of the World Congress on Intelligent Control and Automation (WCICA '11)*, pp. 546–551, Taipei, Taiwan, June 2011.

[27] R. Cui, X. Zhang, and D. Cui, "Adaptive sliding-mode attitude control for autonomous underwater vehicles with input nonlinearities," *Ocean Engineering*, vol. 123, pp. 45–54, 2016.

[28] R. Cui, J. Guo, and Z. Mao, "Adaptive backstepping control of wheeled inverted pendulums models," *Nonlinear Dynamics*, vol. 79, no. 1, pp. 501–511, 2015.

[29] C. S. Chin and S. H. Lum, "Rapid modeling and control systems prototyping of a marine robotic vehicle with model uncertainties using xPC Target system," *Ocean Engineering*, vol. 38, no. 17-18, pp. 2128–2141, 2011.

[30] C. S. Chin, M. W. Lau, E. Low, and G. G. Seet, "A robust controller design method and stability analysis of an underactuated underwater vehicle," *International Journal of Applied Mathematics and Computer Science*, vol. 16, no. 3, pp. 345–356, 2006.

[31] C. S. Chin, M. W. S. Lau, and E. Low, "Supervisory cascaded controller design: experiment test on a remotely operated vehicle," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 225, no. 3, pp. 584–603, 2011.

[32] Z. Li, H. Xiao, C. Yang, and Y. Zhao, "Model predictive control of nonholonomic chained systems using general projection neural networks optimization," *IEEE Transactions on Systems Man and Cybernetics: Systems*, vol. 45, no. 10, pp. 1313–1321, 2015.

[33] A. S. K. Annamalai, R. Sutton, C. Yang, P. Culverhouse, and S. Sharma, "Robust adaptive control of an uninhabited surface vehicle," *Journal of Intelligent and Robotic Systems*, vol. 78, no. 2, pp. 319–338, 2015.

[34] H. Xiao, Z. Li, C. Yang et al., "Robust stabilization of a wheeled mobile robot using model predictive control based on neurodynamics optimization," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 1, pp. 505–516, 2017.

[35] Z. Li, C. Yang, C.-Y. Su, J. Deng, and W. Zhang, "Vision-based model predictive control for steering of a nonholonomic mobile robot," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 2, pp. 553–564, 2016.

[36] G. Campion, G. Bastin, and B. D'Andréa-Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots," *IEEE Transactions on Robotics & Automation*, vol. 12, no. 1, pp. 47–62, 1996.

[37] D. Xu, D. Zhao, J. Yi, and X. Tan, "Trajectory tracking control of omnidirectional wheeled mobile manipulators: robust neural network-based sliding mode approach," *IEEE Transactions on Systems, Man & Cybernetics, Part B: Cybernetics*, vol. 39, no. 3, pp. 788–799, 2009.

[38] K. Watanabe, "Control of an omnidirectional mobile robot," in *Proceedings of the 2nd International Conference on knowledge-Based Intelligent Electronic Systems (KES '98)*, vol. 3, pp. 51–60, Adelaide, Australia, April 1998.

[39] Z. Li and C. Yang, "Neural-adaptive output feedback control of a class of transportation vehicles based on wheeled inverted pendulum models," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 6, pp. 1583–1591, 2012.

[40] J. M. Maciejowski, *Predictive Control with Constraints*, Prentice-Hall, 2002.

[41] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB*, Springer, London, UK, 2009.

[42] M. Li, Y. Li, S. S. Ge, and T. H. Lee, "Adaptive control of robotic manipulators with unified motion constraints," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 184–194, 2017.