Model predictive control for the tracking of autonomous mobile robot combined with a local path planning

Jianhua Li¹, Jianfeng Sun¹, Liqun Liu² and Jiasheng Xu¹

Abstract
This article presents a model predictive control (MPC) coupled with an artificial potential field (APF) to resolve the trajectory tracking while considering the obstacle avoidance. In this article, the obstacle avoidance problem is solved by a local path planning based on the artificial potential field by constructing a virtual goal. A virtual goal is generated to produce an attractive force to guide the mobile robot to a collision-free space. The planned path is controlled by a proportional–integral–derivative (PID) controller to avoid collision. After arriving at the virtual goal, an off-line explicit MPC is calculated to obtain the optimal control inputs to track the reference trajectory. The simulation results show that the proposed method can be applied to control the mobile robot in the environment with one obstacle.

Keywords
Artificial potential function, model predictive control, obstacle avoidance, trajectory tracking

Introduction
The wheeled mobile robots (WMRs) have many applications in various fields, such as exploration in risk environment, industrial transportation for goods and military field. In many applications, the wheeled mobile robot is required to have intelligence ability to make an action to potential obstacle and track the desired trajectory. The study of trajectory tracking with obstacle problem has attracted more attention in recent years.

Model predictive control (also called receding horizon) is one of the most commonly used optimal control technologies in control theory. It is used to calculate a cost function over the finite time interval in the future (prediction horizon) to find the optimal control sequence based on the prediction of the future behaviours of system. Compared to some conventional controller design, it can overcome uncertainties or disturbances existing in a dynamics model. The cost function can be considered as a Lyapunov function by adding a terminal cost to guarantee the stability of the system. It can also solve some difficulties of designing a time-invariant feedback controller in the point stabilization problem instead of designing a simultaneous method for point stabilization and tracking problems. In Refs. 6 and 7, an explicit model predictive control solving the optimization problem in an off-line way is used to track the desired trajectory. In Ref. 8, the safety region associated with the obstacle was defined as the state constraints, and the obstacle avoidance and trajectory tracking were implemented in a nonlinear MPC method. The environment where the mobile moves always is filled with potential obstacles. The automated vehicle is required to have the ability to avoid collision. Thus, the obstacle avoidance problem is widely investigated in recent years. The collision avoidance is studied to produce a collision-free path by utilizing some path planning algorithms, such as A* heuristic search, evolutionary algorithms including genetic algorithms. In some work, a real-time obstacle avoidance and tracking method is used to avoid the potential obstacle. Beyond this method, the artificial potential field can generate a local collision-free path by attracting the vehicle for goal and generating repulsive force for the obstacle. In the literature, the artificial potential function was used to generate a collision-free path, and the planned path is tracked by an MPC. In Ref. 13, a dynamical window approach (DWA) was used to find the optimal control inputs that will

¹School of Mechanical and Electrical Engineering, Lanzhou University of Technology, Lanzhou, China
²Zhuzhou Smelter Group Co., Ltd, Hunan, China

Corresponding author: Jianfeng Sun, School of Mechanical and Electrical Engineering, Lanzhou University of Technology, Lanzhou 730050, China. Email: jianfeng.sun.lut@gmail.com

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
not collide with the obstacle. A collision-free path in the environment filled with obstacles was found by constructing a multi-objective function considering artificial potential field for the obstacles. Due to the obstacle existing in the trajectory, the tracking problem with obstacle is still one of the challenging problems. The obstacle avoidance problem was solved by adding the geometric constraints for the position of the mobile robot to a cost function designed by MPC, and the optimal inputs for the mobile robot were solved by a gradient descent method. In Refs. 15 and 16, the trajectory tracking with obstacle problem was solved by designing a nonlinear controller using Lyapunov function method. However, the local motion of obstacle avoidance when the mobile robot is tracking is not well taken into account in above articles.

In this article, a model predictive control coupled with a local controller for trajectory tracking with existing obstacles. To solve the obstacle avoidance problem, an ellipse defined by a positive definite matrix is used to produce the region to detect the obstacle. The ellipse region can be adjusted by a rotation angle and a diagonal matrix. When an obstacle is detected, a virtual goal away from the obstacle is generated by the vision sensor of the mobile robot. Meanwhile, an artificial potential function is generated to produce an attractive force for the obstacle and a repulsive force for the obstacle, guiding the robot to the desired direction of negative gradient of artificial potential function. The generated local path is controlled by a PID controller. After arriving at the virtual goal, a cost function associated with tracking errors and auxiliary control variables is defined to evaluate the control performance. The constraints of inputs are also considered to avoid actuator saturation. An explicit MPC is utilized to solve the optimization problem by linearizing the nonlinear tracking error dynamics. The trajectory tracking with obstacles problem is solved by an explicit MPC coupled with a local path planning. Finally, the simulation results show that the proposed methodology is valid for the mobile robot.

The rest of the article is designed as follows. The problem statement is described in the section “Problem Formulation.” The potential function for obstacle avoidance and MPC controller for tracking is designed in the section “Methodology.” The simulated results are shown in the section “Simulation and Results.” Finally, the conclusion is drawn in the section “Conclusion.”

**Problem formulation**

The mobile robot starts tracking the leader mobile robot. The local path planning is performed under the assumption that the position of the obstacle is detected when the obstacle is in the range of the sensors. Once an obstacle is detected, then the local collision-free path is derived by the defined artificial potential field. The mobile robot plant calculates the optimal control inputs in terms of PID technology. After updating the pose of mobile robot, the new inputs are calculated again. When the mobile robot arrives at a virtual destination defined by a collision-free point, the tracking problem is going on and solved by an explicit MPC.

**Methodology**

In this section, we will introduce the kinematic model of the mobile robot, and an artificial potential field for obstacle avoidance is proposed. An explicit model prediction control is used to track the reference trajectory.

**Obstacle avoidance**

In this section, a potential field is proposed to avoid the obstacles. We define that a safe region is calculated by a positive definite matrix, then a rotation matrix is used to control the orientation of the region. Potential function is introduced to create repulsive and attractive force for the vehicle and obstacle, respectively. Assumption that the position of the obstacle is \((x_0, y_0)\). The region can be expressed by the following formula

\[
X^T A X = 1
\]  

where equation (1) is positive definite function. \(A\) is the symmetric positive definite matrix \(R^{2\times2}\). \(X \in R^2\), denoted by \([x_0 - x, y_0 - y]^T\) is a vector with two components in \(O_2\) coordinate system. The equation (1) can be rewritten as the following expression by diagonalizing the positive definite matrix \(A\)

\[
(QX)^T \Lambda (QX) = 1
\]  

where \(Q\) is the eigenvector matrix of \(A\), and \(\Lambda\) is the diagonal matrix associated with the eigenvalues of \(A\), denoted by \(\Lambda = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix}\). Equation (2) defines an ellipse in a two-dimensional space. In order to rotate the ellipse, a rotation matrix is defined as follows

\[
Q = \begin{bmatrix}
\cos(\theta) & \sin(\theta) \\
-\sin(\theta) & \cos(\theta)
\end{bmatrix}
\]  

where \(\theta\) is the counter-clockwise rotation angle. We want to adjust the region by constructing the eigenvalue of the matrix \(\Lambda\) using \(\frac{1}{2}\) and \(\frac{1}{2}\).

As shown in Figure 1, the components in eigenvalue vector \(\Lambda\) denoted by \(a\) and \(b\) are used to adjust the region relative distances to the obstacle. To avoid the obstacle collision, a repulsive function \(P_r\) is used to a repulsive force around the obstacle. The repulsive function is defined the same as in Ref. 14

\[
P_r = \begin{cases}
\frac{1}{\gamma + \epsilon(x,y)} & \{ (x,y) \notin \Omega | \Omega : (QX)^T \Lambda (QX) \leq 1 \} \\
0 & \text{else}
\end{cases}
\]  

where \(\lambda\) and \(\gamma\) are constant values to affect the strength of the repulsive energy, and \(\Omega\) is the ellipse region defined by equation (2). \(\epsilon(x, y)\) determines the value and shape of repulsive function associated with the distance between robot and obstacle. \(\epsilon(x, y)\) can be defined as

\[
\epsilon(x,y) = \frac{(x - x_0)^2}{R_x} + \frac{(y - y_0)^2}{R_y}
\]  

where \(R_x\) and \(R_y\) are parameters to determine the effect of horizontal and lateral distances from the obstacle.
Assumption that values of $a$ and $b$ are the same, the three-dimensional diagram of repulsive force for the obstacle is shown in Figure 2.

We also define a safety region into which the mobile robot cannot move to avoid collision. When the obstacle is detected by the sensors attached to the mobile robot, a vector associated with the obstacle is defined as follows

$$f_o = \begin{bmatrix} x_0 - x \\ y_0 - y \end{bmatrix} \tag{6}$$

to drive the robot to a desired goal, a virtual goal generating attractive potential energy for the robot can guide the robot to the desired goal, as depicted in Figure 3. The position of virtual relative to the obstacle can be expressed as

$$f_g = \begin{bmatrix} x_g - x \\ y_g - y \end{bmatrix} \tag{7}$$

the virtual goal can be obtained according to the following relations

$$f_g = wRf_o \tag{8}$$

where $w$ is a positive definite weighting matrix, and the rotation matrix $R$ can be described as

$$R = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \tag{9}$$

where $\alpha$ is the direction of virtual goal relative to the position of the mobile robot. The attraction energy for the robot can be written as

$$P_a = \frac{1}{2}k_a\|X - X_g\|^2 \tag{10}$$

where parameter $k_a$ is a scale factor used to adjust the attractive potential energy. The attractive force for the virtual goal $(0,0)$ is shown in Figure 4.

Combining (4) and (10) yields the artificial potential for the robot as

$$P = P_a + P_r \tag{11}$$

the coordinates of virtual target can be calculated using equation (8). The combination of repulsive force and attractive force is depicted in Figure 5.
Potential with respect to that is, the vector composed of negative gradient of total situation, the mobile robot moves along the desired direction, \( \frac{1}{2} \) The trajectory tracking is a problem that mobile robot can be calculated as \( f = [-\nabla P_x - \nabla P_y] \) (12)

where \( f \) is the vector that guides the mobile robot to move away from the obstacle and get close to the virtual goal. The steering direction of the mobile robot can be written as

\[ \theta_s = \text{atan2}(-\nabla P_y, -\nabla P_x) \] (13)

Assuming that the linear velocity of mobile robot is constant, the direction of motion controlled using PID controller\(^{17} \) can be determined by equation (13). In this situation, the mobile robot moves along the desired direction, that is, the vector composed of negative gradient of total potential with respect to \( x \) and \( y \). With a small constant linear velocity, the next reference orientation can be derived by taking small step size in the direction of the vector \( f \).\(^{18} \) The mobile robot will escape the obstacle and move to a safe region.

**Trajectory tracking problem**

The trajectory tracking is a problem that mobile robot tracks a desired trajectory generated by a virtual mobile robot. The pose of the mobile robot is represented as \( [x, y, \theta]^T \). The kinematic equation of the mobile robot can be written as

\[
\begin{bmatrix}
  \dot{x} \\
  \dot{y} \\
  \dot{\theta}
\end{bmatrix} =
\begin{bmatrix}
  \cos \theta & 0 & v \\
  \sin \theta & 0 & 0 \\
  0 & 1 & w
\end{bmatrix}
\] (14)

where \( v \) and \( \omega \) are the control inputs of the mobile robot. The tracking error dynamics is given by

\[
\begin{bmatrix}
  \dot{x}_e \\
  \dot{y}_e \\
  \dot{\theta}_e
\end{bmatrix} =
\begin{bmatrix}
  \cos \theta & \sin \theta & 0 \\
  -\sin \theta & \cos \theta & 0 \\
  0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  x_e \\
  y_e \\
  \theta_e
\end{bmatrix}
\] (15)

The tracking problem can be formulated as finding appropriate inputs to make the tracking errors to zero when time goes infinitly.

To convert the nonlinear error model, auxiliary variables are defined as

\[
\begin{bmatrix}
  u_1 \\
  u_2
\end{bmatrix} =
\begin{bmatrix}
  v \cos \epsilon_3 - v \\
  w_r - w
\end{bmatrix}
\] (16)

substituting this into equation (15), the error dynamics can be rewritten as

\[
\begin{bmatrix}
  \dot{\epsilon}_1 \\
  \dot{\epsilon}_2 \\
  \dot{\epsilon}_3
\end{bmatrix} =
\begin{bmatrix}
  0 & 0 & 0 \\
  -\omega_r & 0 & \epsilon_3 \\
  0 & 0 & -1
\end{bmatrix}
\begin{bmatrix}
  \epsilon_1 \\
  \epsilon_2 \\
  \epsilon_3
\end{bmatrix} +
\begin{bmatrix}
  1 & 0 & 0 \\
  0 & 0 & 0 \\
  0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
  u_1 \\
  u_2
\end{bmatrix}
\] (17)

according to equation (16), inputs for the mobile robot can be considered as a feedforward term that adds the optimal term which can be written as follows

\[
\begin{bmatrix}
  v \\
  w
\end{bmatrix} =
\begin{bmatrix}
  v \cos (\epsilon_3 - \epsilon_1) \\
  w_r - u_2
\end{bmatrix}
\] (18)

**MPC**

For a general nonlinear control system denoted by the following formula

\[ \dot{x} = f(x, u) \] (19)

with constraints

\[
\begin{bmatrix}
  x(0) = x_0 \\
  u(t) \in U
\end{bmatrix}
\] (20)

where \( x \in \mathbb{R}^n \) is the state vector of the system, \( u(t) \) is the control input vector for the system, \( U \) is a convex set of input constraints, and \( f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p \) is a \( p \) dimensional vector-valued continuous function mapping the control \( u \in \mathbb{R}^m \) and the state \( x \) into the variation of the states of system. The goal of model predictive control similar to other control methods is to find control \( u \) to make the state to the equilibrium point, i.e., \( f(x, u_k) = 0 \).

Considering the state constraints of inputs and states, a total cost function associated with the state of system and control inputs is defined as

\[
J(x,u,T) = \min_u \int_0^T L(x,u)dt + F(x(T))
\]

s.t. \( y_{min} \leq y_{ik}, k = 1, \ldots, N \)

\[
y_{max} \leq y_{ik}, k = 1, \ldots, N
\]

\[
u_{min} \leq u_{ik}, u_{max}, k = 0, 1, \ldots, N
\]

\[
x_{it} = x(t)
\]

\[
y_{it+k+1} = Ax_{it+k} + Bu_{it+k}, k \geq 0
\]

\[
y_{it+k} = Cx_{it+k}, k \geq 0
\]

where \( N \) is the prediction horizon, and \( y \) is the measurement of the system. \( L(x, u) = x^TQx + u^TRu \) is used to represent the control performance. In order to guarantee the stability of the control system, a terminal cost function \( F(x(T)) \) is used to penalize the terminal state into a terminal region used to guarantee the stability of MPC, which can be represented by \( F(x(T)) = x(T)^TPx(T) \). \( Q \) and \( R \) are diagonal matrices that penalize the state performance and actuator effect, respectively.
At any time $t \geq 0$, the optimal solution of cost function (21) is solved. When the sampling time $\delta$ is very small. At every time interval $\tau \in [t, t + \delta]$. The optimal solution can be denoted by $\mathbf{u}(\tau, \mathbf{x}(t))$. The real control inputs at time $t = \tau$ can be chosen as

$$
\mathbf{u}(\tau, \mathbf{x}(t)) = \mathbf{u}(\tau, \mathbf{x}(t)) \quad \tau \in [t, t + \delta]
$$

(22)

The prediction horizon $\tau$ can be presented by $N$ in a discrete way when the sampling time of the system is $\sigma$. Hence, at any prediction serial $k$, the constraints for the cost function are shown in equation (21). During the process of each sampling time, the optimal control inputs in equation (22) are applied to the mobile robot. In the next sampling time duration, equation (21) is solved again, and the new state measurement is updated. The whole process is repeated in this way to track the desired trajectory.

In practice, the cost function is always converted into a discrete formula. The above objective can be considered as a multi-parameter optimization problem, which can be calculated by using an explicit solution. Compared to the interior-point method for solving the convex optimization problem, an explicit solution of the MPC can be expressed in the form of look-up table to reduce the running time of the algorithm.

**Implementation**

In order to avoid the potential obstacle existing in the environment, the local path is planned based on the artificial field and controlled by a PID controller. In addition, an off-line MPC is used to track the reference trajectory. The summary of the proposed algorithm in this article is described as follows:

**Algorithm 1.** Summary of MPC algorithm in the environment with obstacles

**Input:** Parameters for explicit controller and PID controller, artificial function, and the position of obstacles

**Output:** Control inputs $v, w$

1: The explicit MPC is initialized under the convex state constraints, and the locations of obstacles are sensed

2: The explicit MPC is computed off-line

3: loop

4: The trajectory errors are updated and the environment is sensed

5: if The mobile enters the region then

6: The virtual goal is generated, and PID controller is used based on an artificial potential function

7: repeat

8: The PID controller is executed

9: until The mobile robot arrives at the virtual goal

10: else

11: The off-line explicit MPC is executed

12: end if

13: Applying the control inputs to the mobile robot

14: The position of the mobile robot is updated

15: end loop

---

**Simulation and result**

**Simulation configuration**

In this section, a circular reference trajectory is simulated to validate the proposed method. The circular reference trajectory is utilized by considering an obstacle is located in the reference path. In this example, the sampling time is 50 ms, and the prediction horizon is set to 10. In this example, the weighting matrices $Q$, $P$, and $R$ are diagonal matrices

$$
Q = \text{diag}(40, 1000, 20)
$$

$$
P = \text{diag}(0.01, 0.01)
$$

$$
R = \text{diag}(40, 40, 20)
$$

(23)

The circular reference trajectory is generated by $v_r = 0.45 \text{ m/s}$ and $w_r = -0.3 \text{ rad/s}$. The initial postures of the reference trajectory and the mobile robot are $[4.5 \ 6.5 \ 0.7]^T$ and $[4.4 \ 6.45 \ 0.7]^T$, respectively. The constraints of auxiliary control inputs are set to 0 m/s $\leq v \leq 5$ m/s and $-30 \text{ rad/s} \leq w \leq 30 \text{ rad/s}$. The errors state constraints are set to $-0.4 \text{ m} \leq e_1 \leq 0.4 \text{ m}$, $-0.4 \text{ m} \leq e_2 \leq 0.4 \text{ m}$, and $-2 \text{ rad} \leq e_3 \leq 2 \text{ rad}$. After finding the optimal auxiliary control inputs, the real optimal control inputs used to control the motion of the mobile robot can be converted by using equation (16).

The obstacle is located in (6.9, 6.2). To avoid collision occurred in the tracking, a example is performed to validate the effectiveness of the proposed method. When the sensors attached to the mobile robot detect the obstacle in the range, the major and minor axes of the ellipse are 0.2 m and 0.25 m, respectively. For simplicity, in this article, the rotation angle used to rotate the ellipse to generate potential region is aligned with the orientation of the mobile robot. The other parameters for simulating are expressed in Table 1.

Finally, the prediction horizon is $N = 10$, and the control horizon is the same as prediction horizon. The sampling time $t = 0.02$ s is used to discrete the tracking errors model represented by equation (15).

**Simulation result**

As shown in Figure 6, the mobile robot tracks the reference trajectory under the environment filled with one obstacle. The mobile robot tracks the reference trajectory based on explicit MPC controller. When the obstacle is in the range of

| Parameters | Description                  | Value  |
|------------|------------------------------|--------|
| $a$        | The major axis of ellipse    | 0.2 m  |
| $b$        | The minor axis of ellipse    | 0.25 m |
| $\alpha$   | The rotation angle for virtual goal | $\pi/2$ |
| $\lambda$  | The strength for the repulsive energy | 400 |
| $\omega$   | The weight for simulating the virtual goal | 1 |
| $\epsilon$ | The strength for the repulsive energy | 20 |
| $R_x$      | The effect of longitudinal distance | 20 |
| $R_y$      | The effect of lateral distance | 20 |
| $K_a$      | The scalar factor for the attractive potential | 6 |

---

**Table 1.** The parameters for simulation.
the ellipse region for obstacle avoidance, a repulsive force for the obstacle and an attraction force for the virtual goal are generated. The virtual goal is generated by a safe position which is far away from the obstacle, which makes efforts to guide the mobile robot to escape the obstacle. In practice, the virtual goal can be generated by some vision sensors. During this process, the mobile robot is controlled by a PID controller. The mobile robot moves at a constant linear velocity, and the angle velocity is calculated based on PID (see Figure 7). When the robot arrives at the virtual goal, the mobile robot is controlled utilizing a PID controller. The tracking problem is solved using MPC combined with a local path planning controlled by a PID controller. Finally, the simulated results show that the proposed methodology can be applied to the tracking and obstacle avoidance problem in the application of the wheeled mobile robot.

Future work will focus on the moving obstacle in the environment, since the proposed method only considers the static environment. Furthermore, the multi-robot formation control in complex environment filled with potential obstacles and experimental validation are also interesting works to be investigated.

Conclusion

In this article, an ellipse potential region for obstacle avoidance defined by a positive definite function is proposed, and an artificial potential field is proposed to generate repulsive force around the obstacle and attractive force for the vehicle. In addition to those, an off-line explicit model predictive control is used to find the optimal control inputs for tracking. When the position of obstacle is in the defined ellipse region, a local path is planned by the proposed artificial potential field, and the mobile robot is controlled utilizing a PID controller. The tracking problem is solved using MPC combined with a local path planning controlled by a PID controller. Finally, the simulated results show that the proposed methodology can be applied to the tracking and obstacle avoidance problem in the application of the wheeled mobile robot.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The authors would like to thank the support of National Key R&D Program of China (grant no. 2018YFB1309005), the New Equipment Non-ferrous Metallurgy Engineering Research Center of the Ministry of Education, Lanzhou University of Technology.

References

1. Nascimento TP, Dórea CE and Gonçalves LMG. Non-holonomic mobile robots’ trajectory tracking model predictive control: a survey. *Robotica* 2018; 36(5): 676.
2. Chen H and Allgöwer F. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica* 1998; 34(10): 1205–1217.
3. Gu D and Hu H. Receding horizon tracking control of wheeled mobile robots. *IEEE Trans Control Syst Technol* 2006; 14(4): 743–749.

4. Jadabaie A, Yu J and Hauser J. Stabilizing receding horizon control of nonlinear systems: a control lyapunov function approach. In: Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251), San Diego, CA, USA, 2–4 June 1999, 3. IEEE, pp. 1535–1539.

5. Kühne F, Lages WF and da Silva JMG. Point stabilization of mobile robots with nonlinear model predictive control. In: IEEE International Conference Mechatronics and Automation 2005, Niagara Falls, ON, Canada, 29 July–1 Aug 2005, 3, pp. 1163–1168. DOI:10.1109/ICMA.2005.1626717.

6. Maurović I, Baotić M and Petrović I. Explicit model predictive control for trajectory tracking with mobile robots. In: 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Budapest, Hungary, 3–7 July 2011, IEEE, pp. 712–717.

7. Bemporad A, Morari M, Dua V et al. The explicit linear quadratic regulator for constrained systems. *Automatica* 2002; 38(1): 3–20.

8. Lim H, Kang Y, Kim C et al. Nonlinear model predictive controller design with obstacle avoidance for a mobile robot. In: 2008 IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, Beijing, China, 12–15 October 2008, IEEE, pp. 494–499.

9. Li J, Sun J and Chen G. A multi-switching tracking control scheme for autonomous mobile robot in unknown obstacle environments. *Electronics* 2020; 9(1): 42.

10. Rostami SMH, Sangaiah AK, Wang J et al. Obstacle avoidance of mobile robots using modified artificial potential field algorithm. *EURASIP J Wireless Commun Networking* 2019; 2019(1): 70.

11. Ji J, Khajepour A, Melek WW et al. Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints. *IEEE Trans Vehicular Technology* 2016; 66(2): 952–964.

12. Rasekhipour Y, Khajepour A, Chen SK et al. A potential field-based model predictive path-planning controller for autonomous road vehicles. *IEEE Trans Intell Transportation Syst* 2016; 18(5): 1255–1267.

13. De Lima DA and Pereira GAS. Navigation of an autonomous car using vector fields and the dynamic window approach. *J Control Automation Electr Syst* 2013; 24(1–2): 106–116.

14. Tran N, Prodan I, Grotli E et al. Potential-field constructions in an mpc framework: application for safe navigation in a variable coastal environment. *IFAC-PapersOnLine* 2018; 51(20): 307–312.

15. Yang H, Fan X, Shi P et al. Nonlinear control for tracking and obstacle avoidance of a wheeled mobile robot with nonholonomic constraint. *IEEE Trans Control Syst Technology* 2015; 24(2): 741–746.

16. Cui M, Liu H, Liu W et al. An adaptive unscented kalman filter-based controller for simultaneous obstacle avoidance and tracking of wheeled mobile robots with unknown slipping parameters. *J Intell Robotic Syst* 2018; 92(3–4): 489–504.

17. Knospe C. Pid control. *IEEE Control Syst Mag* 2006; 26(1): 30–31.

18. Xie S, Wu P, Peng Y et al. The obstacle avoidance planning of usv based on improved artificial potential field. In: 2014 IEEE International Conference on Information and Automation (ICIA), Hailar, China, 28–30 July 2014, IEEE, pp. 746–751.

19. Boyd S, Boyd SP and Vandenberghe L. *Convex Optimization*. Cambridge University Puniiversity press, 2004.

20. Li Y, Ang KH and Chong GC. Pid control system analysis and design. *IEEE Control Syst Mag* 2006; 26(1): 32–41.