

1-1-2015

A Cooperative Path Planning Algorithm for a Multiple Mobile Robot System in a Dynamic Environment

Wentao Yu

Jun Peng

Xiaoyong Zhang

Kuo-chi Lin

University of Central Florida

Find similar works at: <https://stars.library.ucf.edu/facultybib2010>

University of Central Florida Libraries <http://library.ucf.edu>

This Article is brought to you for free and open access by the Faculty Bibliography at STARS. It has been accepted for inclusion in Faculty Bibliography 2010s by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

Recommended Citation

Yu, Wentao; Peng, Jun; Zhang, Xiaoyong; and Lin, Kuo-chi, "A Cooperative Path Planning Algorithm for a Multiple Mobile Robot System in a Dynamic Environment" (2015). *Faculty Bibliography 2010s*. 6340.
<https://stars.library.ucf.edu/facultybib2010/6340>

A Cooperative Path Planning Algorithm for a Multiple Mobile Robot System in a Dynamic Environment

Regular Paper

Wentao Yu¹, Jun Peng^{1,*}, Xiaoyong Zhang¹ and Kuo-chi Lin²

¹ School of Information Science and Engineering, Central South University, China

² Department of Mechanical and Aerospace Engineering, University of Central Florida, USA

* Corresponding author E-mail: pengj@csu.edu.cn

Received 21 Nov 2013; Accepted 27 Jun 2014

DOI: 10.5772/58832

© 2014 The Author(s). Licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract A practical path planning method for a multiple mobile robot system (MMRS) requires handling both the collision-free constraint and the kinematic constraint of real robots, the latter of which has to date been neglected by most path planning methods. In this paper, we present a practical cooperative path planning algorithm for MMRS in a dynamic environment. First, each robot uses an analytical method to plan an obstacle-avoidance path. Then, a distributed prioritized scheme is introduced to realize cooperative path planning. In the scheme, each robot calculates a priority value according to its situation at each instant in time, which will determine the robot's priority. Higher-priority robots can ignore lower-priority robots, whereas lower-priority robots should avoid collisions with higher-priority robots. To minimize the path length for MMRS, a least path length constraint is added. The priority value is also calculated by a path cost function that takes the path length into consideration. Unlike other priority methods, the algorithm proposed is not time consuming; therefore, it is suitable for dynamic environments. Simulation results are presented to verify the effectiveness of the proposed algorithm.

Keywords Analytical Path Planning, Multiple Mobile Robot System, Kinematic Constraints, Distributed Prioritized Scheme

1. Introduction

As autonomous mobile robots are being deployed in increasingly complex scenarios, it is essential for a team of robots to be able to work in parallel and perform more complex tasks than can be accomplished by a single robot [1]. A well-known and essential requirement for any MMRS is that the robots must be able to navigate safely through the environment in which the tasks are to be performed [2]; thus, proper path planning must be developed.

The goal of path planning is to find a continuous trajectory for a robot from the initial state to the goal state without colliding with obstacles, while maintaining robot-specific constraints [3]. As a result of its importance, path planning has attracted much attention in the field of robotics.

The potential field method is a classic and useful method for path planning. In Reference [4], a potential field is utilized to represent the influence of complex-shaped obstacles. The influence of all obstacles is calculated by a weighted average of the circular obstacle potential fields. The potential field method is extremely fast and also striking, because of its mathematical elegance and simplicity. Nevertheless, a shortcoming of this method is the existence of local minima in the potential function, which cause the robot to become trapped. In addition, the optimality of this method is not assured [5].

The grid-based method combined with the A* algorithm is also considered a good choice for path planning [6]. This method uses a grid-based representation of the environment. The A* algorithm can then be used to solve grid-based path planning problems [7]. The algorithm uses a heuristic concept based on the Euclidean distance to search for the shortest paths. Alternatively, the breadth-first search is a graph search algorithm that is also suitable for grid-based path planning. This algorithm begins at the root node and explores all neighbouring nodes, then explores all neighbouring nodes of each nearest node. Grid-based methods can effectively represent the environment; however, a shortcoming of grid-based path planning is that higher precision requires more computation, making this method unsuitable for complex environments [8].

The probabilistic roadmap method is also a popular path planning scheme for mobile robots. This method uses a sampling technique to discover a sparse representation of obstacles in a configuration space. Reference [9] makes use of a probabilistic roadmap to avoid occlusions of the target and any obstacles. The probabilistic roadmap method is easily implemented and can be applied to complex environments. However, this method may fail when the environment does not have a sufficient number of free points with which to construct a probabilistic map.

There are also many heuristic methods that can be used for path planning, such as rapidly exploring random trees [10], neural networks [11], genetic algorithms [12], simulated annealing [13], ant colony optimization [14], particle swarm optimizer and fuzzy logic [15, 16]. These methods can yield feasible schemes; however, their optimality cannot be assured with any of the above-mentioned methods.

Mobile robots are usually non-holonomic systems; on the other hand, most path planning methods are merely developed to address geometrical constraints, while neglecting kinematic ones. However, Reference [17] presents a feasible algorithm for addressing kinematic constraints.

In Reference [17], a new analytical solution to mobile robot path planning is proposed. This method can provide

a real-time collision-free path for a car-like mobile robot moving in a dynamically changing environment. Based on the explicit consideration of the robot's kinematic model and the collision avoidance condition, a family of feasible trajectories and their corresponding steering controls are derived in terms of one adjustable parameter. This method exhibits excellent performance with respect to precision and real-time ability for mobile robot path planning.

All of the abovementioned methods concern path planning for single robots, which in turn form the basis of path planning for MMRS. The MMRS path planning problem can be defined as follows: given a set of m robots in a k -dimensional workspace, each robot has an initial starting configuration (e.g., position and orientation) and a desired goal configuration; thus, the path should be planned for each robot to reach its goal and avoid collisions with obstacles, as well as other robots in a given workspace [18].

The difference between single robot path planning and MMRS path planning is that various planning approaches are incorporated in the latter to handle conflict between autonomous robots. The path planning approaches for MMRS can mainly be categorized into two classes [19].

The first class is referred to as coupled centralized approaches [20], which apply global information and plans directly. These approaches treat an entire team of robots as a single composite system, to which the classical single-robot path planning algorithms are applied. These approaches pursue both optimality and completeness; thus, they are too computationally intensive to use in practice.

The second class is referred to as decoupled approaches [21]. These approaches can find good solutions quickly, but at the cost of losing optimality and completeness. Decoupled approaches are typically further sub-divided into two broad categories: path coordination and prioritized planning.

Path coordination methods first plan independent paths for robots separately; then, they seek to plan the robots' velocities to avoid collisions along these paths [22]. In path coordination, the area the robots travel across is treated as a shared resource. The decomposition of path planning and velocity planning incorporates an additional time dimension into calculations.

In prioritized planning approaches, different priorities are assigned to each robot [23], either randomly or determined by motion constraints; for example, more highly constrained robots are given higher priority. The sequence of path planning in MMRS is then determined by the priority. This

method can prevent collisions between robots. The advantage of prioritized planning approaches is that they reduce a single planning problem in a very high-dimensional space into a sequence of planning problems in a much lower-dimensional space. In this approach the paths must be computed in a sequence and the sequence is determined by the priority. The computation will last until the robot with the lowest priority plans its path. Thus, it can be observed that the disadvantage of the prioritized planning approach is time consumption.

In this paper, we reconstruct an analytical path planning method to accommodate multiple mobile robot systems with the assistance of the prioritized planning scheme. A practical and cooperative analytical path planning algorithm is presented for MMRS.

This algorithm is completely distributed and each robot plans its path by utilizing local information. To realize cooperative path planning, a distributed prioritized scheme is introduced. First, each robot calculates a priority value according to its situation at each instant and then the priority value is sent to the other robots. The robot's priority is then determined to realize cooperative path planning. Higher-priority robots can ignore lower-priority robots, whereas lower-priority robots should regard higher-priority robots as obstacles. After a confirmation of each robot's priority, the robots plan their paths using the basic analytical path planning method, while taking into consideration the path length constraint. To minimize the path length for MMRS, the priority value is also calculated by a path cost function that takes the path length into consideration.

A traditional priority scheme is time consuming; computation will last until the robot with the lowest priority has planned its path. The priority scheme used in this paper overcomes the time-consumption issue. That is because once the priority has been ensured, the path will be planned in parallel, thereby accelerating the algorithm's execution.

Although only a car-like mobile robot is used in this paper, the cooperative analytical path planning method is also suitable for mobile robots with other kinematic models.

The rest of the paper is organized as follows. Section 2 describes the kinematic model of the car-like non-holonomic mobile robot. Section 3 describes the analytical path planning algorithm for one non-holonomic mobile robot. The formulation of path planning for multiple mobile robot systems is described in section 4. Then, the prioritized path planning method for MMRS is presented in section 5. The experimental results are presented in section 6. Section 7 concludes the paper and outlines future research.

2. Kinematic Model of the Car-like Non-holonomic Mobile Robot

In this paper, the kinematic model of a car-like mobile robot is used. The main feature of the kinematic model of the car-like mobile robot is its non-holonomic constraints, because a rolling-without-slipping condition exists between the wheels and the ground.

The generalized coordinates of the car-like robot model is shown in Figure 1, where the rear wheels are driving wheels and the front wheels are steering wheels. The radius of the wheels is ρ . The car-like robot is round; O is the centre, where the Cartesian coordinate of O is (x, y) and the radius of the car-like robot is R_o . Point W is the midpoint of the rear wheel axis and point V is the midpoint of the front wheel axis. l is the distance between M and V. θ measures the orientation of the car body with respect to the x -axis. φ is the steering angle. Let u_1 be the angular velocity of the driving wheels and u_2 be the steering rate of the front wheels; thus, the kinematic model for the car-like robot can be obtained by equation (1).

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \rho(\cos(\theta) - \tan(\varphi)\sin(\theta)/2) \\ \rho(\sin(\theta) + \tan(\varphi)\cos(\theta)/2) \\ \rho \tan(\varphi)/l \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u_2 \quad (1)$$

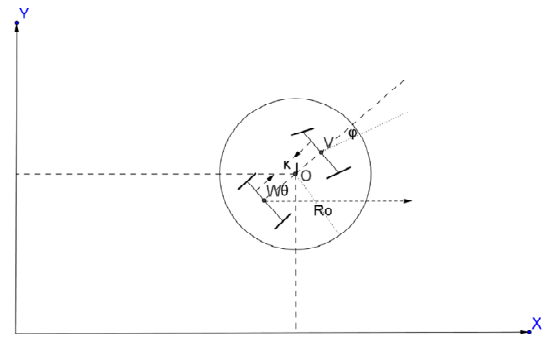


Figure 1. Generalized coordinates of the car-like robot

There is a model singularity at $\varphi = \pm\pi/2$, where the car may become jammed. It is supposed that $\varphi \in (-\pi/2, \pi/2)$. The importance of this singularity is limited due to the restricted range of the steering angle of φ in most practical cases.

To reveal the potential linear characteristic of the kinematic model, the model can be translated into the chained form as follows:

$$\begin{aligned}
z_1 &= x - l \cos(\theta) / 2 \\
z_2 &= \tan(\varphi) / (l \cos^3(\theta)) \\
z_3 &= \tan(\theta) \\
z_4 &= y - l \sin(\theta) / 2 \\
u_1 &= v_{c1} / (\rho \cos(\theta)) \\
u_2 &= l \cos^3(\theta) \cos^2(\varphi) v_{c2} - 3 \sin(\theta) \sin^2(\varphi) v_{c1} / (l \cos^2(\theta))
\end{aligned} \tag{2}$$

where

$$\begin{cases}
\dot{z}_1 = v_{c1} \\
\dot{z}_2 = v_{c2} \\
\dot{z}_3 = z_2 v_{c1} \\
\dot{z}_4 = z_3 v_{c1}
\end{cases} \tag{3}$$

From the chained form shown in (2), it can be observed that the chained form has a strong underlying linear structure, which is nonlinear.

3. Analytical Path Planning for One Non-holonomic Mobile Robot

Analytical path planning takes into consideration both the boundaries and the obstacle-avoidance criterion and then describes the feasible paths by sixth-order piecewise-constant polynomials. Thus, the path planning problem is transformed into a problem of calculating the parameters of the polynomials.

3.1 Feasible Path Planning without Obstacles

If there is no obstacle, the feasible path can be described as follows:

$$z_4 = F(z_1) \tag{4}$$

Without taking the obstacles into consideration, the feasible path only has to satisfy the initial boundary condition $(x_0, y_0, \theta_0, \varphi_0)$ at time t_0 and the goal boundary condition $(x_f, y_f, \theta_f, \varphi_f)$ at the end time t_f . According to (2) and the boundary conditions, the boundary states $z(t_0) = z^0 = (z_1^0, z_2^0, z_3^0, z_4^0)$ and $z(t_f) = z^f = (z_1^f, z_2^f, z_3^f, z_4^f)$ can be obtained.

3.2 Obstacle-Avoidance Criterion

The environment in which the mobile robots move contains obstacles; thus, the collision avoidance criterion should be considered during path planning.

The obstacles are assumed to be dynamic and the i^{th} obstacle's coordinates are $O_i = (x_i(t), y_i(t))$ for the period $t \in [t_0 + kT_s, t_0 + (k+1)T_s]$, where T_s is the sampling period.

During this period, the i^{th} obstacle is moving at a known velocity $v_i^k \triangleq [v_{i,x}^k, v_{i,y}^k]^T$.

Thus, the obstacle-avoidance criterion is expressed as follows:

$$(z_{4,i}' - y_i^k)^2 + (z_{1,i}' - x_i^k)^2 \geq (r_i + R + l/2)^2 \tag{5}$$

where

$$\begin{aligned}
z_{1,i}' &= z_1 - v_{i,x}^k \tau \\
z_{4,i}' &= z_4 - v_{i,y}^k \tau \\
x_i^k &\in [z_{1,i}' - r_i - R, z_{1,i}' + r_i + R + l/2]
\end{aligned}$$

The velocity of the mobile robot along the x-axis is assumed to be a constant C , which can be calculated as follows:

$$C = \frac{x_f - x_0}{T_f - T_0} \tag{6}$$

3.3 Procedure for Feasible Path Planning in a Dynamic Environment

The function $z_4 = F(z_1)$ is chosen to be a six-order polynomial. Thus, the function $z_4 = F(z_1)$ can be expressed as follows:

where a^k is a constant vector to be determined. Therefore, the path can be calculated analytically by adopting the three following procedures and explicit proof of the procedure can be found in [17].

$$z_4 = F(z_1) = a^k f(z_1) = [a_0^k, a_1^k, \dots, a_6^k] \begin{bmatrix} 1 \\ z_1(t) \\ \dots \\ (z_1(t))^6 \end{bmatrix} \tag{7}$$

Step 1: utilize the initial boundary condition and the goal boundary condition to represent the feasible paths in terms of the adjustable parameter a_6^k .

First, the boundary conditions z^0 and z^f should be ensured and the chained form should be obtained.

Then, during time interval $t \in [t_0 + kT_s, t_0 + (k+1)T_s]$, the boundary conditions are z^k and z^f , where z^k can be calculated by integrating (3). By applying the boundary conditions we can obtain (8), which shows that $[a_0^k, a_1^k, a_3^k, a_4^k, a_5^k]$ can be determined by a_6^k .

$$B^k [a_0^k, a_1^k, a_3^k, a_4^k, a_5^k] = Y^k - A^k a_6^k \tag{8}$$

where

$$Y^k = \begin{bmatrix} z_4^k \\ z_3^k \\ z_2^k \\ y_f - l \sin(\theta_f) / 2 \\ \tan(\theta_f) \\ 0 \end{bmatrix}$$

$$A^k = \begin{bmatrix} (z_1^k)^6 \\ 6(z_1^k)^5 \\ 30(z_1^k)^4 \\ (z_1^f)^6 \\ 6(z_1^f)^5 \\ 30(z_1^f)^4 \end{bmatrix}$$

$$B^k = \begin{bmatrix} 1 & z_1^k & (z_1^k)^2 & (z_1^k)^3 & (z_1^k)^4 & (z_1^k)^5 \\ 0 & 1 & 2z_1^k & 3(z_1^k)^2 & 4(z_1^k)^3 & 5(z_1^k)^4 \\ 0 & 0 & 2 & 6z_1^k & 12(z_1^k)^2 & 20(z_1^k)^3 \\ 1 & z_1^f & (z_1^f)^2 & (z_1^f)^3 & (z_1^f)^4 & (z_1^f)^5 \\ 0 & 1 & 2z_1^f & 3(z_1^f)^2 & 4(z_1^f)^3 & 5(z_1^f)^4 \\ 0 & 0 & 2 & 6z_1^f & 12(z_1^f)^2 & 20(z_1^f)^3 \end{bmatrix}$$

From (8), we know that if a_6^k is acquired, the function $z_4 = F(z_1)$ is guaranteed.

Step 2: applying the obstacle-avoidance criterion to determine the constant a_6^k .

The criterion is expressed as follows:

$$\min_{t \in [t_i^-, t_i^+]} g_2(z_1(t), k)(a_6^k)^2 + g_{1,i}(z_1(t), k, \tau)a_6^k + g_{0,i}(z_1(t), k, \tau) \geq 0 \quad (9)$$

where $\tau = t - t_0 - kT_s$.

$[t_i^-, t_i^+] \subset [t_0 + kT_s, t_f]$ is the time interval when it exists

$$x_i^k \in [z_1(t) - v_{i,x}^k \tau - r_i - R, z_1(t) - v_{i,x}^k \tau + 0.5l + r_i + R].$$

$$F^k = [1 \quad z_1(t) \quad (z_1(t))^2 \quad (z_1(t))^3 \quad (z_1(t))^4 \quad (z_1(t))^5] \quad (10)$$

$$g_2(z_1(t), k) = [(z_1(t))^6 - F^k(B^k)^{-1}A^k]^2 \quad (11)$$

$$g_{1,i}(z_1(t), k, \tau) = 2[(z_1(t))^6 - F^k(B^k)^{-1}A^k][F^k(B^k)^{-1}A^k - y_i^k - v_{i,y}^k \tau] \quad (12)$$

$$g_{0,i}(z_1(t), k, \tau) = [F^k(B^k)^{-1}A^k - y_i^k - v_{i,y}^k \tau]^2 + (z_1(t) - x_i^k - v_{i,x}^k \tau)^2 - (r_i + R + 0.5l)^2 \quad (13)$$

$$x_i^k = x_i^0 + T_s \sum_{j=0}^{k-1} v_{i,x}^j \quad (14)$$

$$y_i^k = y_i^0 + T_s \sum_{j=0}^{k-1} v_{i,y}^j \quad (15)$$

Step 3: determining the feasible path and calculating the steering inputs.

For $t \in (t_0 + kT_s, t_0 + (k+1)T_s]$, $v_{c1}(t) = v_{c1}^k$, $v_{c2}(t) = v_{c2}^k$, calculate v_{c1}^k and v_{c2}^k .

$$v_{c1}^k = \frac{z_1^f - z_1^0}{T} \quad (16)$$

$$v_{c2}^k = 6[a_3^k + 4a_4^k z_1^k + 10a_5^k (z_1^k)^2 + 20a_6^k (z_1^k)^3]v_{c1}^0 + 24[a_4^k + 5a_5^k z_1^k + 15a_6^k (z_1^k)^2](t - t_0 - kT_s)(v_{c1}^0)^2 + 60[a_5^k + 6a_6^k z_1^k](t - t_0 - kT_s)^2(v_{c1}^0)^3 + 120a_6^k(t - t_0 - kT_s)^3(v_{c1}^0)^4 \quad (17)$$

$$z_4 = F(z_1) \Rightarrow y - 0.5l \sin(\theta) = F(x - 0.5l \sin(\theta)) \Rightarrow y = F(x - 0.5l \sin(\theta) + 0.5l \sin(\theta)) \quad (18)$$

where θ can be calculated by (2).

4. Formulation of Path Planning for MMRS

4.1 The World Model for the Mobile Robot in MMRS

It is assumed that in MMRS, there are many robots moving in a two-dimensional environment. All of the robots have to avoid colliding with both moving obstacles and other robots. Each robot has limited sensing and communication abilities; thus, not all of the information about the environment can be obtained. Usually, information about obstacles and other robots is essential for any robot to plan its path; therefore, the world model of the mobile robots mainly maintains information about the obstacles in the environment and the robots.

Suppose that there is a team of moving robots $R = \{R_1, \dots, R_n\}$, where $i \in \{1, \dots, n\}$. L_s^i is the limited sensing range of robot R_i ; L_c^i is the limited communication range of robot R_i . B_s^i represents the set of obstacles that can be sensed by R_i . All obstacles in set B_s^i should satisfy equation (19).

$$q_o^j - q_i \leq L_s^i \quad (19)$$

where q_o^j is the position of the j^{th} obstacle in the set B_s^i and q_i is the position of R_i .

B_c^i is the set of perceptible robots of R_i . Information regarding B_c^i can be acquired through the robot's sensing or communication abilities; therefore, all of the robots in the set B_c^i should satisfy equation (20) and equation (21).

$$q_l - q_i \leq L_c^i \quad (20)$$

$$q_l - q_i \leq L_s^i \quad (21)$$

where q_l is the position of R_l , $l \in \{1, \dots, n\}$ and $l \neq i$. Usually, the communication range of one robot is greater than its sensing range.

$$B_i = B_s^i \cup B_c^i \quad (22)$$

where B_i is the set of all obstacles and robots in the i^{th} robot world model that need to be considered during path planning.

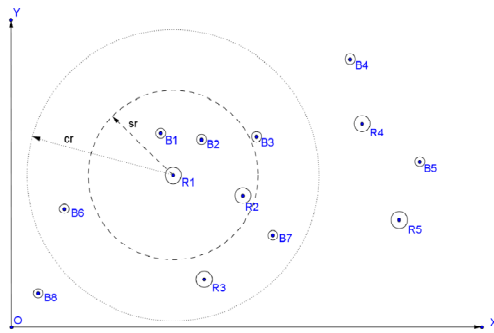


Figure 2. All of the mobile robots and the dynamic obstacles surrounding R_0

As shown in Figure 2, many mobile robots and moving obstacles surround R_1 . Because only obstacles B_1 and B_2 satisfy equation (19) and robots R_2 and R_3 satisfy equation (20) and (21), the world model of R_1 only maintains information about the objects as shown in Figure 3.

Assumption 1: in order to facilitate the avoidance criterion, the robots and obstacles are represented by circles.

Assumption 2: all of the robots exhibit accurate self-localization and can exhibit accurate obstacle localization.

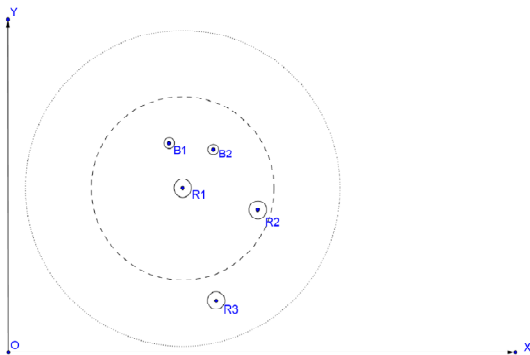


Figure 3. Dynamic obstacle model

4.2 The Description of Path Planning for MMRS

Suppose there is a team of n robots $R = \{r_1, \dots, r_n\}$, traversing binary map M . To navigate freely, the robots should plan their paths to avoid collisions with obstacles and other robots. Moreover, the planned paths should be as short as possible. The MMRS path planning problem therefore has five requirements:

- Satisfying the initial starting configuration for each robot, such as the position and orientation at the starting point. C_{init}^i is the initial configuration for robot r_i , $i \in \{1, \dots, n\}$.
- Satisfying the goal configuration for each robot, such as the position and orientation at the goal point. C_{goal}^i is the goal configuration for robot r_i .
- Avoiding collisions with obstacles.
- Avoiding collisions with other moving robots.
- Minimizing the total path length of the MMRS.

5. Prioritized Path Planning Method for Multiple Mobile Robots

The cooperative analytical path planning algorithm is a decoupled strategy. Given a team of robots, each robot can be assigned a priority value that can determine its priority. When planning the path, a robot with high priority can neglect robots with lower priority, whereas a robot with low priority should avoid robots with higher priority. This scheme can coordinate the paths of robots and prevent collisions between robots.

For the cooperative analytical path planning algorithm for MMRS, the priority has a serious effect on the quality of the planning path; thus, the priority should be determined appropriately. In this paper, a priority function is proposed to calculate the priority value. This priority value is transmitted to all robots and priority is then determined through the distributed, prioritized planning scheme.

5.1 The Calculation of Priority

A priority function is proposed to calculate the priority value of the mobile robots. When robot R_i is located at point p , the priority value can be calculated by equation (23):

$$f(p, i) = h(p, i) + g(p, i) \quad (23)$$

where $f(p, i)$ is the path cost of the i^{th} robot when it is located at point p ; $h(p, i)$ is the cost of moving from the initial point (x_0, y_0) to point p ; $g(p, i)$ is the cost of moving from point p to the goal point (x_f, y_f) . In this function, the effect of orientation is neglected. The path cost can be measured by the path length.

$$h(p, i) = L_i^{t-1} + \int_{t-1}^t \sqrt{1 + \left(\frac{dy}{dx}\right)^2} dx \quad (24)$$

$$g(p, i) = \sqrt{(x_f - p_x)^2 + (y_f - p_y)^2} \quad (25)$$

5.2 Distributed Prioritized Planning Scheme for MMRS

Each robot utilizes local information to calculate a priority value by (23). Then, each robot has to verify the order of the priority. The order will determine the way each robot coordinates path planning with other robots.

Assume that the mobile robots within the world model have complete communication. The robots exchange messages $m_i^t = \langle i, t, p_i^t, B_i^t \rangle$, where p_i^t is the priority value of the i^{th} robot at time t . The principle for determining the order of the robot's priority D_i^t can be described as follows:

$$\begin{aligned} &\text{if } p_i^t > p_j^t \\ &\quad D_i^t > D_j^t \\ &\text{else if } p_i^t < p_j^t \\ &\quad D_i^t < D_j^t \\ &\text{else if } p_i^t = p_j^t \\ &\quad \text{if } B_i^t > B_j^t \\ &\quad \quad D_i^t > D_j^t \\ &\quad \text{else if } B_i^t < B_j^t \\ &\quad \quad D_i^t < D_j^t \\ &\quad \text{else if } B_i^t = B_j^t \\ &\quad \quad \text{if } i > j \\ &\quad \quad \quad D_i^t > D_j^t \\ &\quad \quad \text{else} \\ &\quad \quad \quad D_i^t < D_j^t \end{aligned}$$

Where B_i^t means the number of obstacles in B_i at t .

5.3 The Constraint for Minimizing the Path Length of Multiple Mobile Robot Systems

Suppose a path for robot i needs to be planned from time t to time t_f . The planned path is described by (26).

$$\begin{cases} x = x_i & x_i^t \leq x_i \leq x_i^f \\ y = F(x_i) \end{cases} \quad (26)$$

The function F has a continuous derivative and the element of the arc length can be expressed as follows:

$$dL_i^t = \sqrt{(dx)^2 + (dy)^2} = \sqrt{1 + \left(\frac{dy}{dx}\right)^2} dx \quad (27)$$

where L_i^t is the path length of the i^{th} robot planned at time t .

Thus, the length of the path can be calculated by the definite integral of the function F :

$$L_i^t = \int_{x_i^t}^{x_i^f} \sqrt{1 + \left(\frac{dy}{dx}\right)^2} dx \quad (28)$$

Hence, let m be a function that defines all path lengths of the multiple mobile robot system:

$$m(t) = \sum_{i=1}^n L_i^t \quad (29)$$

Therefore, one objective of path planning for multiple mobile robot systems is minimizing the function m .

Based on equation (29), we can observe that only centralized planning can resolve this optimization problem. Because centralized planning can create heavy communication and computation loads, the constraint for minimizing the path length of multiple mobile robots can be divided into n sub-constraints:

$$m(i, t) = L_i^t = \int_{x_i^t}^{x_i^f} \sqrt{1 + \left(\frac{dy}{dx}\right)^2} dx \quad (30)$$

Therefore, during the path planning of each robot, both the path length constraint and the criterion for avoiding dynamic obstacles and other robots should be taken into consideration.

5.4 Cooperative Analytical Path Planning Algorithm for MMRS

In the cooperative analytical path planning algorithm, the analytical path planning method is combined with the distributed priority scheme to obtain a new multiple mobile robot path planning method that can plan sub-optimal obstacle-avoidance paths in real time.

In the cooperative analytical path planning algorithm, each robot uses the same planning scheme at each instant. First, the initial and goal conditions should be verified; then, the path from the current time position to the goal position is determined and each robot will move along its planned path at this time. At the next time interval, the boundary condition is modified and a new trajectory that can also satisfy the obstacle-avoidance criterion is planned. The flow of the cooperative analytical path planning algorithm for MMRS can be illustrated by the i^{th} robot path planning at time $(t_0 + kT_s, t_0 + (k+1)T_s]$.

- 1 Verify that the boundary condition contains z_i^k and z_i^f . Utilize the initial boundary condition and the goal boundary condition to represent the feasible

paths in terms of one adjustable parameter a_6^k of the i^{th} robot.

- 2 Calculate the priority value of the i^{th} robot according to equation (23) and then determine the order of the priority value through the communication between the robots within their limited communication range.
- 3 Determine which robots can be regarded as obstacles and which robots can be ignored. Then, apply the obstacle-avoidance criterion to determine the range of the constant a_6^k of the i^{th} robot.
- 4 Choose the best a_6^k that can satisfy the constraint for minimizing the path length.
- 5 Determine the feasible path and calculate the steering inputs.

5.5 Proof for the Effectiveness of the Cooperative Analytical Path Planning Algorithm

Theorem 1: if a multiple mobile robot system has complete communication, the cooperative analytical path planning algorithm can produce paths that can prevent collisions with moving obstacles and other robots according to the path length constraints.

Proof: the proof follows the rules of logical induction.

For a team consisting of only one robot, it is clear that the robot can produce the shortest path and avoid collisions with any moving obstacles.

Consider a team of mobile robots $R = \{R_1, \dots, R_n\}$ composed of n autonomous mobile robots. Each robot can avoid collisions with obstacles using the analytical path planning method.

At time t , the i^{th} robot calculates a priority value p_i^t ; suppose the order of the priorities satisfies equation (31) and that the priority can be determined according to this equation.

$$p_1^t > p_2^t > \dots > p_n^t \quad (31)$$

Robot R_n has the lowest priority; therefore, the path generated by R_n will prevent collisions with all robots $\{R_1, \dots, R_{n-1}\}$ subject to the path length constraint.

The priority of robot R_{n-1} is higher than that of R_n ; therefore, the path generated by R_{n-1} will ignore the motion of R_n and avoid collisions with robots $\{R_1, \dots, R_{n-2}\}$ that satisfy the path length constraint.

The rest of the paths can be deduced by analogy up to robot R_1 .

Robot R_1 has the highest priority; therefore, it can ignore all other robots and plan its path according to the path length constraints. R_1 does not have to account for collisions with other robots because all other robots will actively avoid it.

Therefore, it can be observed that if priority can be transmitted through the entire multiple mobile robot system, each robot will utilize the distributed prioritized planning scheme to produce a feasible path that can prevent collisions with moving obstacles and other robots.

It is worth mentioning that although each robot plans its path according to the path length constraints, the path length of the MMRS is not guaranteed to be the shortest. The path length constraint can only ensure that each robot can produce the shortest path when it treats other robots with higher priority as obstacles. Thus, the algorithm presented in this paper is a sub-optimal algorithm.

6. Simulation Results

To verify the effectiveness of the proposed algorithm, we performed a number of simulations. The simulations were carried out on an HP computer with 4G EMS memory and an Intel Core i2-2120 CPU.

Suppose that there are two robots moving within the same two-dimensional environment where three mobile obstacles exist.

In the simulations, the parameters of the robots were set as follows:

For Robot 1,

$$R_{01}=1, l_1=0.8, \rho_1=0.1.$$

The initial condition is

$$x_1^0 = 0, y_1^0 = 2, \theta_1^0 = \pi / 4, \varphi_1^0 = 0.$$

The final condition is

$$x_1^f = 18, y_1^f = 12, \theta_1^f = -\pi / 4, \varphi_1^f = 0.$$

For Robot 2,

$$R_{02}=1, l_2=0.8, \rho_2=0.1.$$

The initial condition is

$$x_2^0 = 0, y_2^0 = -1, \theta_2^0 = \pi / 4, \varphi_2^0 = 0.$$

The final condition is

$$x_2^f = 24, y_2^f = 9, \theta_2^f = -\pi / 4, \varphi_2^f = 0.$$

Assume that there are three obstacles moving within the environment. All of the obstacles are assumed to be circular and their radii are all set to 0.5, and each obstacle moves at different velocities that can change dynamically. The initial position of obstacle 1 is (5, 0). The initial position of obstacle 2 is (9, 4). The initial position of obstacle 3 is (19, 10). The velocity of each obstacle in the simulation is shown in Table 1.

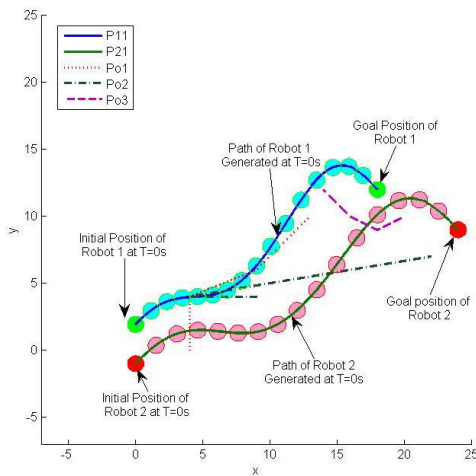
At each instant, a new path will be planned for the robot according to the initial position, the goal position and the current state of the environment. It can be seen in Figure 4 that the initial position at each time instant changes, so a new path will be planned for each robot according to the current situation.

To illustrate this characteristic, the planned paths of two robots at six instants are shown in Figure 4. At the first instant, Robot 1 and Robot 2 plan their paths according to the information obtained at time 0. The planned paths can satisfy both the current and goal conditions without colliding with the three obstacles or the other mobile robots.

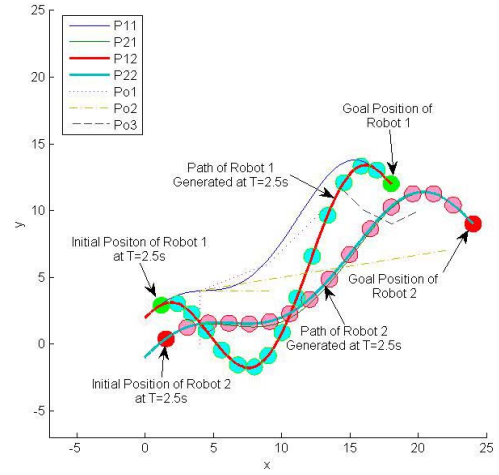
At the second instant, the obstacles' states change and thus, Robot 1 plans its path again to avoid colliding with obstacle 1.

Time		0s	2.5s	10s	20s	25s	30s
Obstacle1	v_x	0	0	0.5	0.2	0.2	0.2
	v_y	0.4	0.4	0.2	0.2	0.2	0.2
Obstacle2	v_x	-0.5	-0.5	0.6	0.6	0.6	0.6
	v_y	0	0	0.1	0.1	0.1	0.1
Obstacle3	v_x	-0.2	-0.2	-0.2	-0.1	-0.1	-0.1
	v_y	-0.1	-0.1	0.1	0.1	0.1	0.1

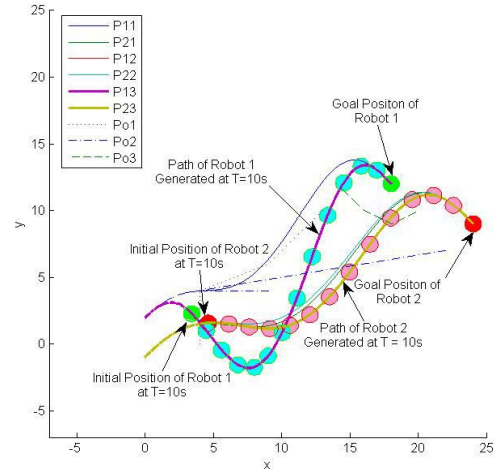
Table 1. The velocity parameters of three obstacles



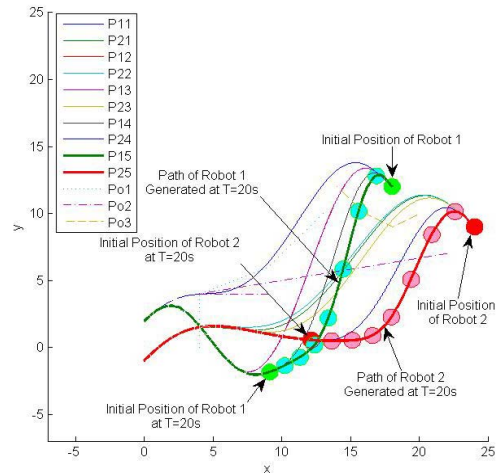
a) T=0 s. P11 represents the path of Robot 1 generated at T=0 s; P21 represents the path of Robot 2 generated at T=0 s



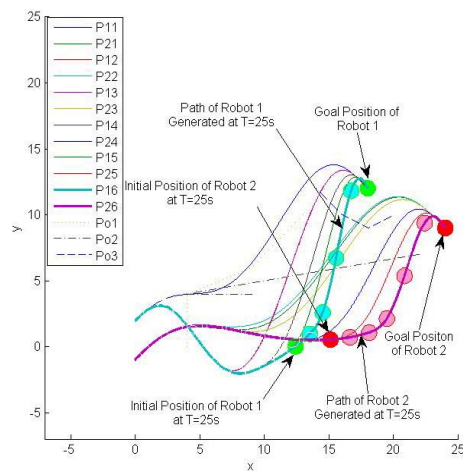
b) T=2.5 s. P12 represents the path of Robot 1 generated at T=2.5 s; P22 represents the path of Robot 2 generated at T=2.5 s



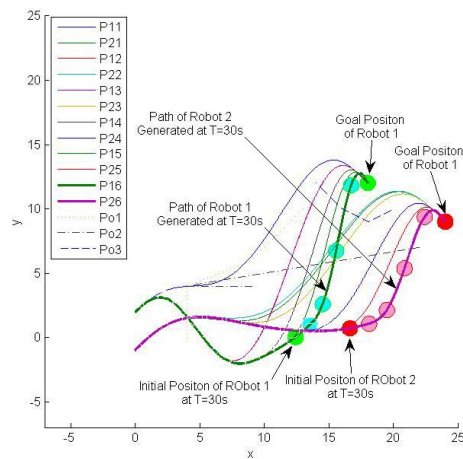
c) T=10 s. P13 represents the path of Robot 1 generated at T=10 s; P23 represents the path of Robot 2 generated at T=10 s



d) T=20 s. P14 represents the path of Robot 1 generated at T=20 s; P24 represents the path of Robot 2 generated at T=20 s



e) $T=25$ s. P15 represents the path of Robot 1 generated at $T=25$ s ;
P25 represents the path of Robot 2 generated at $T=25$ s



f) $T=30$ s. P16 represents the path of Robot 1 generated at $T=30$ s ;
P26 represents the path of Robot 2 generated at $T=30$ s

Figure 4. The paths of Robot 1 and Robot 2 at six instants

At the third instant, there may be a conflict between Robot 1 and Robot 2. According to equation (23), the priority value of Robot 1 is greater than that of Robot 2; therefore, during path planning, Robot 1 can ignore Robot 2, but Robot 2 needs to change its path to avoid colliding with Robot 1.

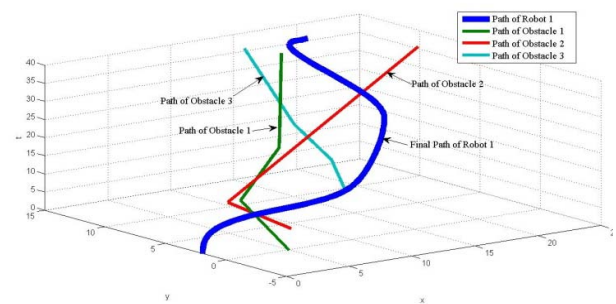
At the last three instants, Robot 1 and Robot 2 alter their paths to accommodate the movements of the three obstacles.

Briefly, at each time instant, the algorithm proposed in this paper is utilized to plan a new path from the current position to the goal position, and the path planned at the last instant is the true path along which the robots move. The path planned at the last instant can be called the final path of the robot.

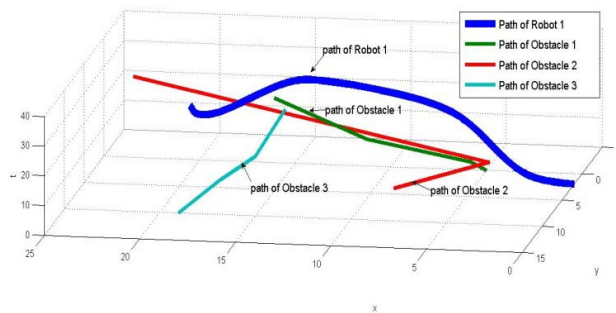
As shown in Figure 4, it is not immediately clear that there are no conflicts between robots and obstacles. To

demonstrate the effectiveness of the algorithm, the final paths of the robots and obstacles are depicted as three-dimensional figures.

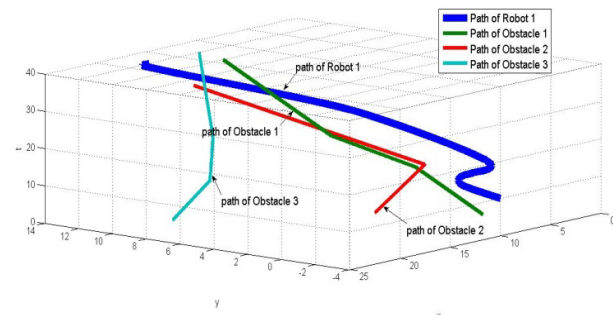
Figure 5(a) is a normal three-dimensional figure that shows the paths of Robot 1 and the three obstacles from time 0 to time 40. Figure 5(a) is rotated to obtain Figure 5(b) and Figure 5(c). Figure 5(b) shows that there are no conflicts between Robot 1 and obstacle 1 or between Robot 1 and obstacle 3. Figure 5(c) demonstrates that there are no conflicts between Robot 1 and obstacle 2.



a) Normal three-dimensional figure



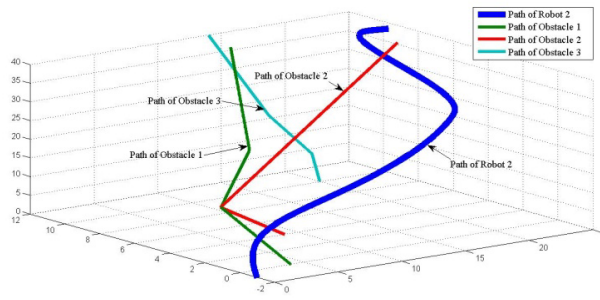
b) Rotated three-dimensional figure



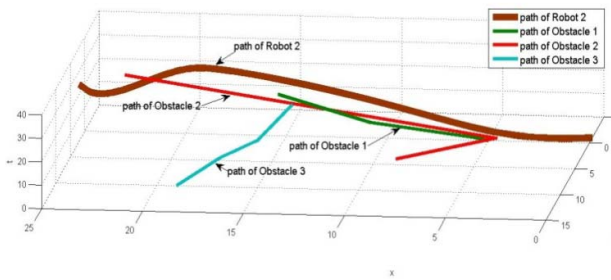
c) Rotated three-dimensional figure

Figure 5. The final paths of Robot 1 and three obstacles

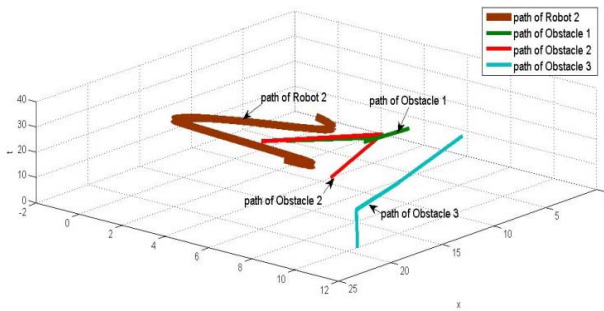
Figure 6(a) is a normal three-dimensional figure that shows the paths of Robot 2 and the three obstacles from time 0 to time 40. Figure 6(a) is rotated to obtain Figure 6(b) and Figure 6(c). Figure 6(b) shows that there are no conflicts between Robot 1 and obstacle 1 or between Robot 1 and obstacle 3. Figure 6(c) demonstrates that there are no conflicts between Robot 1 and obstacle 2.



a) Normal three-dimensional figure



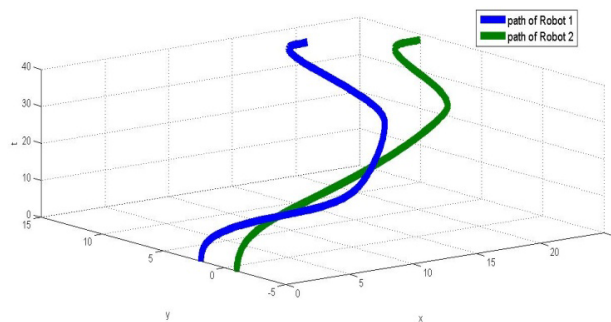
b) Rotated three-dimensional figure



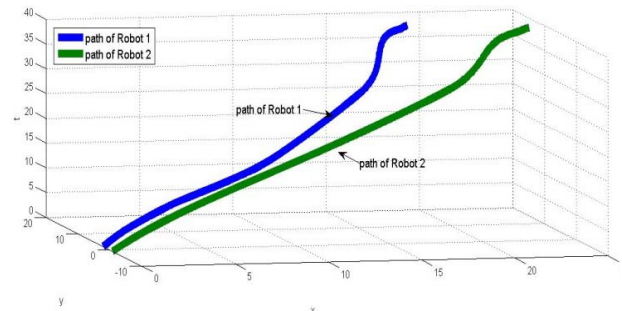
c) Rotated three-dimensional figure

Figure 6. The final paths of Robot 2 and three obstacles

Figure 7(a) is a normal three-dimensional figure that shows the paths of Robot 1 and Robot 2 from time 0 to time 40. Figure 7(a) is rotated to obtain Figure 7(b). Figure 7(b) clearly shows that there are no conflicts between Robot 1 and Robot 2.



a) Normal three-dimensional figure



b) Rotated three-dimensional figure

Figure 7. The final paths of Robot 1 and Robot 2

7. Conclusions

In this paper, a practical and cooperative path planning method for MMRS that can produce sub-optimal or optimal obstacle-avoidance paths in real time within a dynamic environment is presented. This algorithm appropriately combines the analytical path planning method with a priority scheme. The priority scheme used in this paper does not require significant computation; thus, it can accommodate the dynamic environment.

The cooperative path planning algorithm requires accurate robot self-localization and target localization. The paths generated by inaccurate localization information may create conflicts. However, for most real robots, whose navigation systems usually rely on vision systems or laser sensors, accurate self-localization and target localization are hard to accomplish. Therefore, in future studies, we will explore how to modify the algorithm proposed in this paper to tolerate certain localization errors in a bid to expand application of the algorithm.

8. Acknowledgements

This paper was supported by the National Natural Science Foundation of China (No. 61379111 and 61202342).

9. References

- [1] Jaillet L, Cortes J, Simeon T (2010) Sampling-based Path Planning on Configuration-space Costmaps. *IEEE t. robot.*, 26: 635-646.
- [2] Li C, Wang F, Zhao L, Li Y, Song Y (2013) An Improved Chaotic Motion Path Planner for Autonomous Mobile Robots Based on A Logistic Map. *Int. j. adv. robot. syst.* Available from: http://www.intechopen.com/journals/international_journal_of_advanced_robotic_systems/an-improved-chaotic-motion-path-planner-for-autonomous-mobile-robots-based-on-a-logistic-map. Accessed 18 Jun 2013.
- [3] Kala R, Shukla A, Tiwari R (2010) Dynamic Environment Robot Path Planning Using Hierarchical Evolutionary Algorithms. *cybernet. syst.*, 41: 435-454.

- [4] Daily R, Bevly D (2010) Harmonic Potential Field Path Planning for High Speed Vehicles. In: American Control Conference, 2008 Jun 11-13; Washington, USA. pp. 4609-4614.
- [5] Koren Y., Borenstein J (1991) Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation. In: IEEE Conference on Robotics and Automation, 1991 Apr 15-17; California, USA. pp. 1398-1404.
- [6] Kanehara M., Kagami S., Kuffner J., Thompson S. (2007) Path Shortening And Smoothing of Grid-based Path Planning with Consideration of Obstacles. In: IEEE International Conference on Systems, Man and Cybernetics, 2007 Oct 7-17; Montreal, Canada. pp. 991-996.
- [7] Kim M, Lee H, Wei Y, Lee M (2012) A Study of New Path Planning Algorithm Using Extended A* Algorithm with Survivability. *Lecture Notes in Computer Science*, 7508: 835-846.
- [8] Gorerzen C, Kong Z, Mettler B (2010) A Survey of Motion Planning Algorithms from the Perspective of Autonomous UAV Guidance. *j. intell. robot. syst.* 87: 65-100.
- [9] Park J, Kim J, Song J (2007) Path Planning for A Robot Manipulator Based on Probabilistic Roadmap and Reinforcement Learning. *int. j. control autom*, 5: 674-680.
- [10] Bruce J., Veloso M. (2002) Real-time Randomized Path Planning for Robot Navigation. In: IEEE International Conference on Intelligent Robots and Systems, 2002 Oct 1-4; Lausanne, Switzerland. pp. 2383-2388..
- [11] Hong Q, Yang S, Willms A, Yi Z (2009) Real-time Robot Path Planning Based on a Modified Pulse-coupled Neural Network Model. *IEEE t. neural networ*, 20: 1724-1739.
- [12] Liu C, Liu H, Yang J (2011) A Path Planning Method Based on Adaptive Genetic Algorithm for Mobile Robot. *Journal of Information & Computational Science*, 8: 808-814.
- [13] Kindl M, Rowe N (2012) Evaluating Simulated Annealing for the Weighted-region Path Planning Problem. In: International Conference on Advanced Information Networking and Application Workshops, March 9-12; Fukuoka, Japan. pp. 926-931.
- [14] Garcia M, Montiel O, Castillo O, Sepulveda R, Melin P (2009) Path Planning for Autonomous Mobile Robot Navigation with Ant Colony Optimization and Fuzzy Cost Function Evaluation. *appl. soft comput.* 9: 1102-1110.
- [15] Zhang Y, Gong D, Zhang J (2013) Robot Path Planning in Uncertain Environment Using Multi-objective Particle Swarm Optimization. *Neurocomputing*, 103: 172-185.
- [16] Zoumponos G, Aspragathos N (2006) Fuzzy Logic Path Planning for the Robotic Placement of Fabrics on a Work Table. *robot. cim-int. manuf*, 24,: 174-186.
- [17] Qu Z, Wang J, Plaisted C (2004) A New Analytical Solution to Mobile Robot Trajectory Generation in the Presence of Moving Obstacles. *IEEE t. robot.* 20: 978-993.
- [18] Chakraborty J, Konar A, Jain L, Chakraborty K (2009) Cooperative Multi-robot Path Planning Using Differential Evolution. *j. intell. fuzzy syst.* 20: 13-27.
- [19] Desaraju V (2012) Decentralized Path Planning for Multi-agent Teams with Complex Constraints. *auton robot*, 2012: 385-403.
- [20] Luna R, Bekris K (2011) Efficient and Complete Centralized Multi-robot Path Planning. In: International Conference on Intelligent Robots and Systems, 2011 Sep 22-25; San Francisco, USA . pp. 3268-3265.
- [21] Bennewitz M, Burgard W, Thrun S (2002) Finding and Optimizing Solvable Priority Schemes for Decoupled Path Planning Techniques for Teams of Mobile Robots. *robot auton syst*, 41: 89-99.
- [22] Ding X, Rahmani A, Egerstedt M (2010) Multi-UAV Convoy Protection: An Optimal Approach to Path Planning and Coordination. *IEEE t. robot.*, 26: 256-268.
- [23] Velagapudi P, Sycara K, Scerri P (2010) Decentralized Prioritized Planning in Large Multirobot Teams. In: International Conference on Intelligent Robots and Systems, 2010 Oct 8 12-15; Taipei, Taiwan. pp. 4603-4609.