INTECH
open science | open minds

OPEN ACCESS
ARTICLE

International Journal of Advanced Robotic Systems

Real-Time Navigation of Nonholonomic Mobile Robots under Velocity Vector Control

Regular Paper

Zi-hui Zhang[1,*], Yue-shan Xiong[1], Pei Fan[1], Yue Liu[1] and Matis Cheng[2]

1 School of Computer, National University of Defence and Technology, P.R.China
2 Department of Computer Science, University of Victoria, Canada
* Corresponding author E-mail: peter1982.10@163.com

Received 04 Jan 2012; Accepted 24 Mar 2012

DOI: 10.5772/51855

Abstract In this paper, linear navigation law is studied in depth and we suggest an efficient, practical and simple approach for nonholonomic mobile robot navigation under velocity vector control based on the linear navigation law. First of all, an obstacle is equivalent to a velocity vector when detected by a robot's sensory system according to the relative distant and relative direction between the robot and the obstacle. Then the vector sum of all obstacles' equivalent velocity vectors (OEVVs) and the linear navigation velocity vector (LNVV) derived from the linear navigation law drives the robot to reach the desired goal position without colliding with any obstacle in the robot's workspace. Furthermore, during the process of driving the mobile robot under the resultant velocity vector, a set of strategies for velocity and acceleration constraints (VAC) is devised to make kinematic behaviours of the mobile robot more practical. Finally, to validate the effectiveness and superiority, extensive simulation results with no obstacles, a single obstacle and multiple obstacles are provided.

Keywords Velocity vector, nonholonomic mobile robots, real-time navigation, velocity and acceleration constraints, obstacle's equivalent velocity vector.

1. Introduction

Navigation is among the most important and fundamental problems in mobile robotics. This problem is not only of interest to mobile robotics, but has also been widely studied from different points of view (control theory [1], artificial intelligence [2, 3], hydromechanics [4] and so on). Various methods are discussed in the literature, such as the roadmap method [5], the well-known Khatib artificial potential field [6], and most recently, the linear navigation law [7]. Due to the broad and promising applications of mobile robots in the industrial and service fields, researchers have been looking for a more efficient, practical and simpler navigation method.

The idea of the artificial potential field was originally suggested for manipulator collision avoidance [6] and is also widely used for mobile robots. The main drawback of this method is the local minima [8, 9]. One challenge problem [10] of the potential field method is to find a suitable potential energy function which ensures obstacle avoidance and goal attainability at the same time. Another challenging problem is to calculate the potential energy for an arbitrarily shaped obstacle. Once the potential field is determined, the distribution of the velocity field can be specified by computing the derivative to each point of the robot's workspace. Intelligent generation methods of the angle and speed components of the velocity vector field are discussed by Claudia [11, 12] respectively based on hydrodynamics and the fuzzy-logic technique.

Most recently, a new navigation method called the linear navigation law [7, 13] has been suggested for nonholonomic mobile robots. This method is based upon the proportional navigation guidance law [15] that is a navigation method well known and widely discussed in the aerospace community. In the linear navigation law a mobile robot's orientation angle is a linear function of the visibility angle, i.e., line of sight, and collision avoidance can be achieved by modifying the linear proportional parameters in an online or offline manner, details can be found in [14]. Since the standard proportional navigation can be used to track moving objects, through some improvements the linear navigation law could be capable of real-time navigation and trajectory tracking for moving goals. In addition, the linear navigation law is a simple and efficient method that can be applied to almost all types of mobile robots without any restriction to the robot's dimension and the obstacles' geometric shapes.

Owing to these advantages, many attempts to use this method for mobile robot navigation have been made. In [7], the linear navigation function, based on kinematic equations with exponential and deviation terms, is used for nonholonomic wheeled mobile robots. In [13], a detailed deduction from the proportional navigation guidance law to the linear navigation law is studied, various simulation examples are presented to confirm their theoretical results and the results are proven rigorously in mathematics. Although some promising results have been achieved in previous work, it does not mean that the linear navigation law has been studied in depth and several issues deserve to be investigated. For example, when an obstacle is detected by the sensory system, the linear navigation parameters need to be tuned online and a local goal position needs to be selected for collision avoidance; this strategy is inefficient and cannot be extended to multiple robots. Besides, the linear navigation law does not take the kinematic constraints of a real mobile robot into account.

In this paper, the real-time navigation problem of a nonholonomic mobile robot in static environments is discussed and we suggest an efficient and practical real-time navigation approach for the nonholonomic mobile robot under velocity vector control based on the linear navigation law. For each navigation period, an obstacle is equivalent to a velocity vector when detected by the robot's sensory system according to the relative distant and relative direction between the robot and the obstacle. Then the vector sum of all obstacles' equivalent velocity vectors (OEVVs) and the linear navigation velocity vector (LNVV) derived from the linear navigation law makes the robot move to the desired goal position without any collision. Actually, the OEVV effects locally within a certain distance of an obstacle to move the robot away from the obstacle, whereas the LNVV effects globally in the whole workspace, moving a robot to the desired goal. Furthermore, in order to make the kinematic behaviour of a mobile robot more practical, a set of strategies for velocity and acceleration constraints (VAC) is devised. The advantages of our approach can be summarized as follows.

1) Compared with the potential field method, our approach does not have oscillations in the presence of obstacles.
2) As a result of the strategies for VAC, the path resulting from our approach becomes shorter and more practical when compared with the linear navigation law.
3) The relative distance between a robot and an obstacle can be changed by tuning the weighting factor of our approach.
4) In the multi-obstacle environment, these processes in the linear navigation law of tuning navigation parameters online and selecting a local goal position for collision avoidance are not needed in our approach, and the navigation path obtained is smoother.
5) Our approach is quite simple and highly efficient. More importantly, it could be adapted to the navigation problem of multiple mobile robots in dynamic environments with some improvements.

This paper is organized as follows: in Section 2, we introduce the problem. In Section 3, we discuss the technique of OEVV and the strategies for VAC. An extensive simulation is carried out to illustrate our approach in Section 4. Conclusions are drawn in Section 5.

2. Problem Description

The real-time navigation problem of a mobile robot is shown in Fig.1. Our aim is to develop an approach that ensures obstacle avoidance and goal attainability at the same time. For simplicity, the following definitions/assumptions are made.

1) The workspace is attached to a global fixed reference frame of coordinates {W}. Its origin point is O.
2) The mobile robot with a circular shape is denoted by R, its radius is d_r. The robot's reference point located at the geometric centre is denoted by R(t) and has coordinates (x_r, y_r) in {W}. The start position of the robot is $R(t_0) = (x_{r0}, y_{r0})$. The linear velocity and orientation angle of the robot is denoted respectively by $v_r(t)$ and $\theta_r(t)$. The initial value of linear velocity is $v_r(t_0) = 0$.
3) The obstacle denoted by O_i is a circular object with radius d_i. The obstacle's reference point located at the geometric centre is denoted by $O_i(t)$ and has coordinates (x_{oi}, y_{oi}) in {W}.
4) The mobile robot has a semi-omnidirectional sensory system that allows it to measure in real-time the position and the size of obstacles in the coverage area (CA) of the robot's sensory system. The CA is characterized by a semicircle in front of the robot with a radius D_0, where D_0 is the detection range of the sensory system. For each navigation period, only obstacles in the CA are taken into account.
5) The goal position of the robot is denoted by G with coordinates (x_G, y_G). Variable θ_{rg} denotes the line of sight angle from the robot to the goal position.
6) The robot can be treated as a point by extending the size of obstacles by at least half of the size of the robot.

In addition, it is assumed that the wheels do not slide, that is to say, a nonholonomic constraint exists on the motion of the mobile robot in the form

$$\dot{x}_r \sin\theta_r - \dot{y}_r \cos\theta_r = 0 \tag{1}$$

Therefore, the kinematic model of the mobile robot can be described using the following equations. Where ω_r is the angular velocity of the mobile robot with $\omega_r(t_0) = 0$.

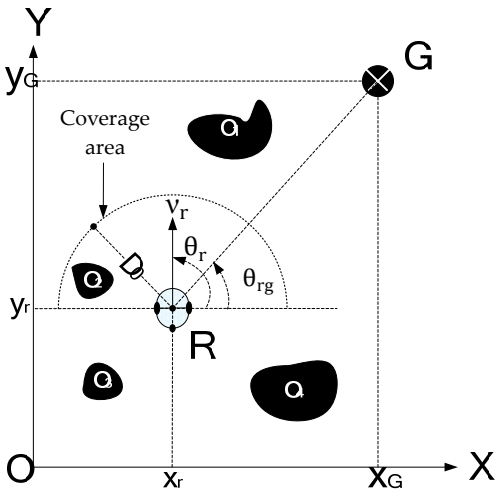

Figure 1. The real-time navigation problem of a mobile robot

$$\begin{cases} \dot{x}_r(t) = v_r(t)\cos\theta_r(t) \\ \dot{y}_r(t) = v_r(t)\sin\theta_r(t) \\ \dot{\theta}_r(t) = \omega_r(t) \end{cases} \tag{2}$$

A discrete-time version of the dynamic model of the mobile robot is:

$$\begin{cases} x_r(t_k) = x_r(t_{k-1}) + Tv_r(t_{k-1})\cos(\theta_r(t_{k-1}) + T\omega_r(t_{k-1})/2) \\ y_r(t_k) = y_r(t_{k-1}) + Tv_r(t_{k-1})\sin(\theta_r(t_{k-1}) + T\omega_r(t_{k-1})/2) \\ \theta_r(t_k) = \theta_r(t_{k-1}) + T\omega_r(t_{k-1}) \end{cases} \tag{3}$$

Where T is the navigation period with $t_k = kT$ $(k \in Z)$ and $k \geq 1$.

Without loss of practicability, it is assumed that the mobile robot is subject to the following VAC.

$$\begin{matrix} |v_r(t)| \leq v_{max} & |\dot{v}_r(t)| \leq \dot{v}_{max} \\ |\omega_r(t)| \leq \omega_{max} & |\dot{\omega}_r(t)| \leq \dot{\omega}_{max} \end{matrix} \tag{4}$$

3. Real-Time Navigation under Velocity Vector Control

For the real-time navigation problem of a mobile robot, some variables have to be defined to represent the instant status messages of the robot, the goal and the obstacles. In the potential field method [16], a goal is characterized by an attractive force and an obstacle is characterized by a repulsive force, afterwards, a robot moves in the resultant potential field. In the linear navigation law method [7], the orientation angle of a mobile robot is represented in real-time by the relative distance and direction between the robot and the goal. While an obstacle is perceived by the sensory system, a polar histogram providing free directions is constructed, then the linear navigation parameters are tuned online and a local goal position is selected for collision avoidance.

In our approach an obstacle in the CA of the sensory system is equivalent to a velocity vector, the resultant vector of OEVV and LNVV drives the robot to reach the desired goal position without colliding with any obstacle in the robot's workspace. See Fig. 2, the obstacles of O_2 and O_3 are located in the CA, so they need to be taken into account in this navigation period. The OEVV of O_2 and O_3 are represented respectively by $\mathbf{v_{o2}}$ and $\mathbf{v_{o3}}$. The function $\sum \mathbf{v_{oi}}$ is the sum of the OEVVs of obstacles located in the CA. Variables of θ_l and v_l are the orientation angle and linear velocity derived from the linear navigation law. Hence, the OEVV effects locally in the CA to move the robot away from the obstacle, whereas the LNVV effects globally in the whole workspace to move the robot to reach the desired goal. The formulation of our approach can be seen in Eq. (5), the parameters of α and β define the weighting factors of LNVV and OEVV respectively, γ_i defines the weighting factor of the obstacle O_i and n is the number of obstacles in the CA.

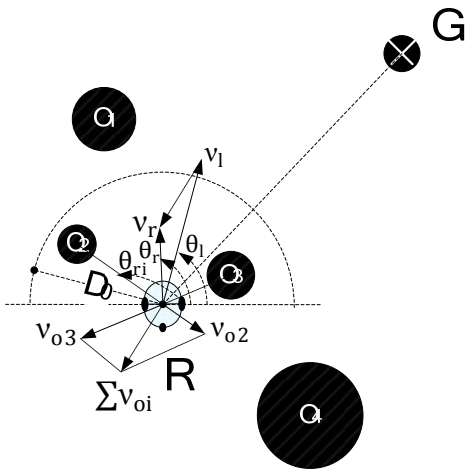

Figure 2. Real-time navigation under velocity vector control

$$\mathbf{v_r} = \alpha\mathbf{v_l} + \beta\sum_{i=1}^{n}\gamma_i\mathbf{v_{oi}} \tag{5}$$

In general conditions $\alpha = \beta = \gamma_i$, however, these parameters can be changed for different requirements. For example, while considering security the robot is required to be further away from all obstacles, the parameter of β can be increased. When the robot is required to be further away from one or several obstacles, the parameters of γ_i corresponding to these obstacles can be increased. Note that the control inputs of the mobile robot are the angle and speed components of the resultant velocity vector, i.e., the orientation angle and linear velocity.

Compared with the potential field method, our approach calculates directly the velocity vector for controlling a mobile robot, without the processes of computing the attractive force/repulsive force and transforming them to the linear velocity/orientation angle. Moreover, the local minima problem does not exist in our approach. Compared with the linear navigation law, when an obstacle is detected, our approach does not need to construct a polar histogram for free directions and select a local goal position for collision avoidance. It can avoid the obstacle according to the resultant velocity vector from the OEVV and LNVV, therefore, the path is shorter. Furthermore, multiple mobile robot navigation and collision avoidance in dynamic environments are related problems that can be solved using our approach through certain improvements, however, this is beyond the scope of this paper.

3.1 Technique of the OEVV

As long as an obstacle is located in the CA, it has to be dealt with; in other situations it cannot be considered, see Fig. 2. This is reasonable, since in practical terms, when an obstacle is located outside of the CA, it cannot be perceived by the robot's sensory system; so it would naturally not be taken into account in the navigation decision-making process.

The variable d_{ri} represents the minimum distance between the robot and the outside edge of the obstacle O_i; θ_{ri} is the angle from the robot to the obstacle O_i.

$$d_{ri} = \sqrt{(x_r - x_{oi})^2 + (y_r - y_{oi})^2} - d_i \tag{6}$$

$$\theta_{ri} = \tan^{-1}\left(\frac{y_{oi} - y_r}{x_{oi} - x_r}\right) \tag{7}$$

The technique of the OEVV for the obstacle O_i is formulated in Eq. (8), the direction of the OEVV for O_i is from the obstacle O_i to the robot. Variable v_o is a generic basic speed for all obstacles, see Eq. (9); ρ is a ratio of the generic basic speed to the magnitude of LNVV. To prevent the oscillations in the presence of obstacles [17], a smaller value of ρ is assigned typically, such as 0.2. The basic speed of the obstacle O_i is denoted by $\lambda_i v_o$, where λ_i is a scale factor.

$$v_{oi} = \begin{cases} \lambda_i v_o \tan^{-1}\left(\frac{1}{d_{ri}} - \frac{1}{D_0}\right)^2 & \frac{D_0}{2} < d_{ri} \le D_0 \\ & \cap\ |\theta_{ri} - \theta_r| \le \frac{\pi}{2} \\ \lambda_i v_o \left[\tan^{-1}\left(\frac{1}{D_0}\right)^2 + \left(\frac{1}{2d_{ri}} - \frac{1}{D_0}\right)\right] & 0 < d_{ri} \le \frac{D_0}{2} \\ & \cap\ |\theta_{ri} - \theta_r| \le \frac{\pi}{2} \\ +\infty & d_{ri} = 0 \\ 0 & \text{else} \end{cases} \tag{8}$$

$$v_o = \rho v_l \tag{9}$$

The curve of the magnitude of OEVV can be seen in Fig. 3, which illustrates the variation of the speed v_{oi} with the distance d_{ri}. It can be concluded from this figure that the continuous and smooth changes of the magnitude of OEVV guarantee no oscillations. When an obstacle is just entering the CA, $v_{oi} = 0$. With the decreasing of d_{ri}, the risk of collision between the robot and the obstacle increases, along with the magnitude of OEVV. When $d_{ri} \le D_0/2$, the magnitude of OEVV increases more rapidly.

Fig. 4 illustrates that when using our approach there is no oscillation in the presence of an obstacle, an inherent limitation in the potential field method. Next, we give brief proof for this proposition. As soon as an obstacle O_i enters into the CA, in the worst case, the minimum distance between the robot and the obstacle is $\min(d_{ri}) = D_0 - v_l T$. According to Eq. (8) the OEVV for the obstacle O_i is $\lambda_i v_o \tan^{-1}\left(\frac{1}{D_0 - v_l T} - \frac{1}{D_0}\right)^2$. Combining with Eq. (9), we have $v_{oi} = \lambda_i \rho v_l \tan^{-1}\left(\frac{1}{D_0 - v_l T} - \frac{1}{D_0}\right)^2$. See Fig. 5, when the speed v_{oi} is constant and the vector $\mathbf{v_r}$ is tangential to

the circle with a radius ν_{oi}, the shift of orientation angle of the robot ($\Delta\theta$) becomes maximum and $\max(\Delta\theta) = \sin^{-1}(\frac{\nu_{oi}}{\nu_l}) = \sin^{-1}\left[\lambda_i \rho \tan^{-1}\left(\frac{1}{D_0 - \nu_l T} - \frac{1}{D_0}\right)^2\right]$. Generally, the parameters are set as follows: $\nu_l < 1$, $D_0 > 2$, $T = 1$, $\lambda_i = 1$ and $0 < \rho < 1$. Hence, we can conclude that $\left(\frac{1}{D_0 - \nu_l T} - \frac{1}{D_0}\right)^2 \approx 0$ and $\tan^{-1}\left(\frac{1}{D_0 - \nu_l T} - \frac{1}{D_0}\right)^2 \approx 0$. Substituting them into the function $\max(\Delta\theta)$ results in $\max(\Delta\theta) \approx 0$. In a word, a very small shift of the robot's orientation angle is affected in the presence of an obstacle in our approach and no oscillations happen.

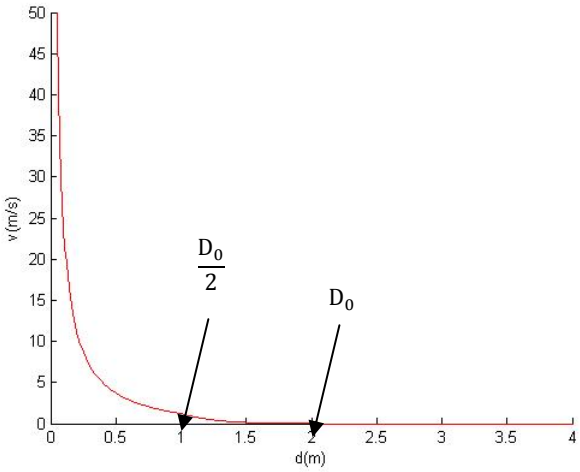

Figure 3. The curve of the magnitude of OEVV

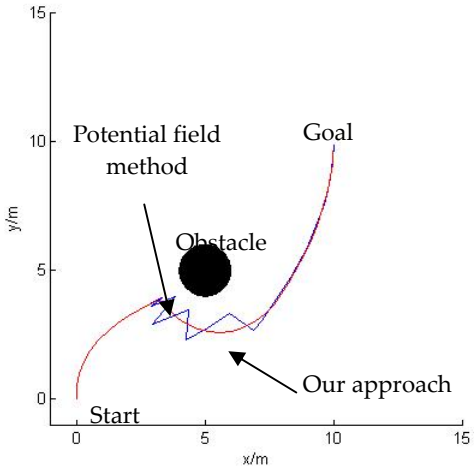

Figure 4. Oscillations in the presence of an obstacle

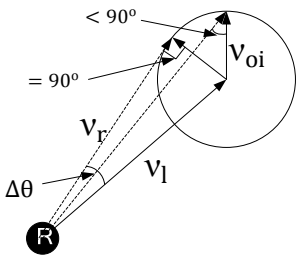

Figure 5. The shift of a robot's orientation angle caused by the OEVV

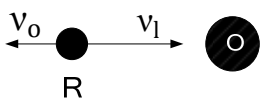

Figure 6. The opposite directions of the OEVV and LNVV

Next the security of collision avoidance and goal attainability are discussed.

SECURITY ANALYSIS: For any known linear navigation speed, we may assume that $\nu_l = \nu_{con} = \text{constant}$. Based on Eq. (8) it is easy to see that there must exist a variable $\delta \neq 0$, when $d_{ri} < \delta$ the inequation of $\nu_{oi} > \nu_{con}$ is satisfied. In other words, when the distance between a robot and an obstacle is adequately close, the OEVV would become large enough to move the robot away from the obstacle for security.

ATTAINABILITY ANALYSIS: Provided that the predefined goal is affected by no obstacles. In the case that a robot does not encounter any obstacle in the whole navigation course, it is possible to prove based on the linear navigation law that the goal can be reachable. In other cases, a robot is affected only if one or several obstacles are located in the CA of the robot's sensory system. Therefore, when the robot moves outside of the influence regions of these obstacles, it would be driven to the predefined goal with the control of LNVV. In addition, Fig. 6 shows a particular case where these directions of the OEVV and LNVV are opposite. In this case, the linear navigation parameters can be tuned online to shift the robot's orientation angle for collision avoidance and drive the robot to the predefined goal, the details can be found in Ref. [7].

3.2 Strategies for the VAC

It can be recognized that in absence of kinematic limits, that is the VAC, these control inputs of ν_r and θ_r derived from Eq. (5) guarantee fulfilment of the real-time robot navigation. However, these values and/or the corresponding time derivatives may violate the VAC in practice. For example, if $\nu_r = 3\text{m/s}$ but $\nu_{max} = 1\text{m/s}$, it is clear that this control input cannot be implemented for a real robot. Therefore, the VAC must be considered.

These control inputs of linear velocity and orientation angle for a real mobile robot are ν_{ref} and θ_{ref}. In the very beginning, without the VAC that is formulated in Eq. (4), we have:

$$\nu_{ref}(t_k) = \nu_r(t_k) \qquad \theta_{ref}(t_k) = \theta_r(t_k) \qquad (10)$$

Considering the velocity constraints, the following strategies are adopted to limit the linear velocity and angular velocity.

$$v_a(t_k) = \begin{cases} v_{max} & \text{if } v_r(t_k) > v_{max} \\ v_r(t_k) & \text{if } v_r(t_k) \le v_{max} \end{cases} \quad (11)$$

$$\dot{v}_a(t_k) = \frac{[v_a(t_k) - v(t_{k-1})]}{T} \quad (12)$$

$$\omega_r(t_k) = \frac{[\theta_r(t_k) - \theta_r(t_{k-1})]}{T} \quad (13)$$

$$\omega_a(t_k) = \begin{cases} \text{sign}[\omega_r(t_k)]\omega_{max} & \text{if } |\omega_r(t_k)| > \omega_{max} \\ \omega_r(t_k) & \text{if } |\omega_r(t_k)| \le \omega_{max} \end{cases} \quad (14)$$

$$\dot{\omega}_a(t_k) = \frac{[\omega_a(t_k) - \omega(t_{k-1})]}{T} \quad (15)$$

Considering the acceleration constraints, the following strategies are adopted to limit the linear acceleration and angular acceleration.

$$\dot{v}_b(t_k) = \begin{cases} \text{sign}[\dot{v}_a(t_k)]\dot{v}_{max} & \text{if } |\dot{v}_a(t_k)| > \dot{v}_{max} \\ \dot{v}_a(t_k) & \text{if } |\dot{v}_a(t_k)| \le \dot{v}_{max} \end{cases} \quad (16)$$

$$v_c(t_k) = v(t_{k-1}) + \dot{v}_b(t_k)T \quad (17)$$

$$\dot{\omega}_b(t_k) = \begin{cases} \text{sign}[\dot{\omega}_a(t_k)]\dot{\omega}_{max} & \text{if } |\dot{\omega}_a(t_k)| > \dot{\omega}_{max} \\ \dot{\omega}_a(t_k) & \text{if } |\dot{\omega}_a(t_k)| \le \dot{\omega}_{max} \end{cases} \quad (18)$$

$$\omega_c(t_k) = \omega(t_{k-1}) + \dot{\omega}_b(t_k)T \quad (19)$$

$$\theta_c(t_k) = \theta(t_{k-1}) + \omega(t_{k-1})T + \frac{1}{2}\dot{\omega}_b(t_k)T^2 \quad (20)$$

Remarkably, by construction, it is

$$\begin{matrix} |v_c(t_k)| \le v_{max} & |\dot{v}_b(t_k)| \le \dot{v}_{max} \\ |\omega_c(t_k)| \le \omega_{max} & |\dot{\omega}_b(t_k)| \le \dot{\omega}_{max} \end{matrix} \quad (21)$$

Hence, the control inputs of linear velocity and orientation angle for a real mobile robot can be finally set:

$$v_{ref}(t_k) = v_c(t_k) \qquad \theta_{ref}(t_k) = \theta_c(t_k) \quad (22)$$

4. Simulations

In order to validate the effectiveness and superiority of our approach, extensive simulation results are presented in this section. The parameters are set as follows: $\alpha = 1$, $T = 1(s)$, $\rho = 0.2$, $\lambda_i = 1$, $v_l = 0.3(m/s)$, $v_{max} = 0.5(m/s)$, $\dot{v}_{max} = 0.5(m/s)$, $\omega_{max} = \pi/2\,(rad/s)$, $\dot{\omega}_{max} = \pi/2\,(rad^2/s)$.

The experiment results of navigation without obstacles illustrate the effectiveness. The sharp points on the navigation path in the linear navigation law do not occur in our approach and the path obtained is shorter and more practical. The experiment results of navigation with a single robot illustrate that an obstacle can be avoided effectively using our approach; in addition, the distance between the robot and the obstacle can be changed online by tuning the weight factor β. The experiment results of navigation with multiple obstacles illustrate that our approach is still effective in a complex environment and the processes in the linear navigation law of tuning the navigation parameters online and selecting a local goal position for collision avoidance are not needed in our approach. Note that simulation results with regard to the lack of oscillations in our approach are no longer given in this section, however, you can find them in Fig. 4.

4.1 Navigation without Obstacles

The navigation results of our approach and the linear navigation law without obstacles are shown in Fig. 7. According to Eq. (5), when no obstacles are perceived by a robot, it is clear that the formulation difference between our approach and the linear navigation law is nothing more than a proportional factor. However, due to the VAC, the navigation path derived from our approach does not have a sharp point that occurs in the linear navigation law; moreover, the path is shorter and seems more practical. Fig. 8 shows the curves of the robot's orientation angle in the navigation course, which illustrate once again that our approach can generate a continuous and smooth orientation angle while the linear navigation law cannot. Fig. 9 shows the curves of the robot's linear velocity, from which it is evident that these two methods have the same trendline.

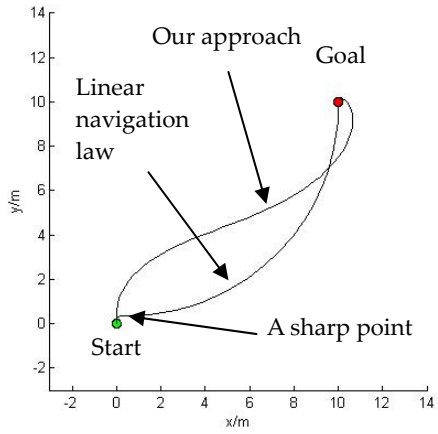

Figure 7. Experimental results without obstacles when compared with the linear navigation law

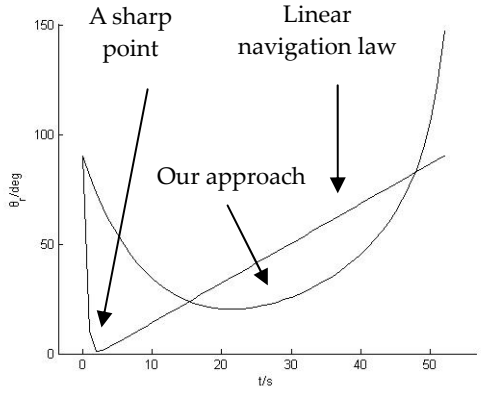

Figure 8. Curves of the orientation angle without obstacles

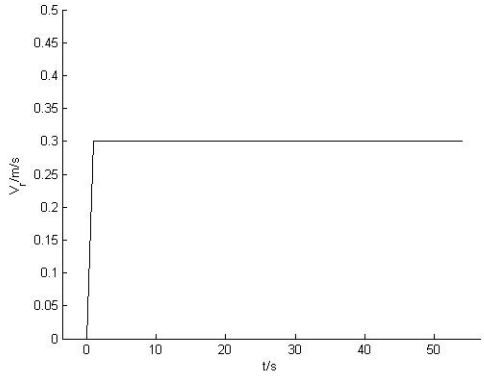

Figure 9. Curves of the linear velocity without obstacle

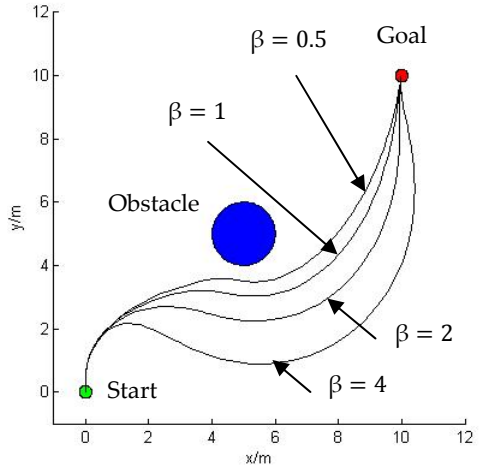

Figure 10. Experimental results with a single obstacle

4.2 Navigation with a Single Obstacle

The navigation results with a single obstacle with different values of β are shown in Fig. 10. The start and goal positions are (0,0)m and (10,10)m. The initial value of the robot's orientation angle is $\theta_r(t_0) = 90°$, $D_0 = 3$. As seen from the figure, a small weight factor β produces a short path close to the obstacle, whereas a larger β produces a less economical path farther away from the obstacle. The curves of orientation angle and linear velocities are shown in Fig. 11 and Fig. 12. The value of β becomes larger/smaller, the effect of the obstacle becomes more evident and the range of the robot's orientation angle becomes wider. Similarly, the robot's linear velocity has an analogous profile.

Fig. 13 depicts curves of the OEVV (v_o) with different values of β, t_{detect} denotes the time point when the obstacle is detected by the sensory system. Two conclusions can be drawn as follows:

1) The maximum of OEVV, that is max(v_o), becomes larger when β is smaller. The reason is that a smaller β produces a path closer to an obstacle so that the OEVV becomes larger and so does the max(v_o).

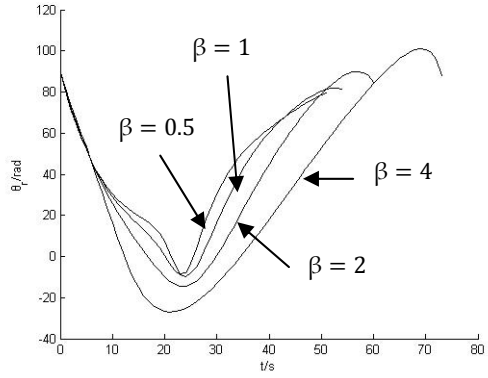

Figure 11. Curves of the orientation angle with a single obstacle

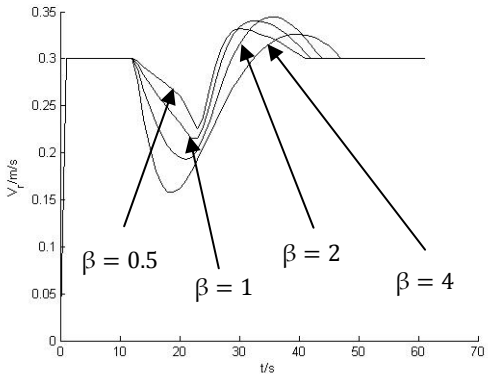

Figure 12. Curves of the linear velocity with a single obstacle

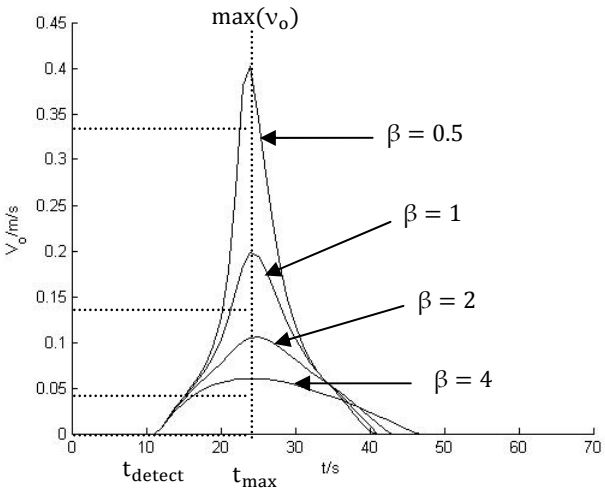

Figure 13. Curves of the OEVV with different values of β

	β = 0.5	β = 1	β = 2	β = 4
max(v_o)(m/s)	0.4026	0.1975	0.1054	0.0603
t_{max}(s)	24	24	25	24

Table 1. Values of max(v_o) and t_{max} that correspond to different values of β

2) An approximate proportion function lies between the $\max(v_o)$ and β, i.e., $\max(v_o) \propto \beta$ (see Table 1). Furthermore, the values of t_{max} corresponding to $\max(v_o)$ with different values of β are almost the same, see Fig. 13 and Table 1. Thus, the time interval $\Delta t = t_{max} - t_{detect}$ is almost invariable with different β. In other words, the time interval between the time when an obstacle is detected and the time when the obstacle produces a maximal OEVV is nearly the same. This phenomenon validates the reasonability and correctness of our approach.

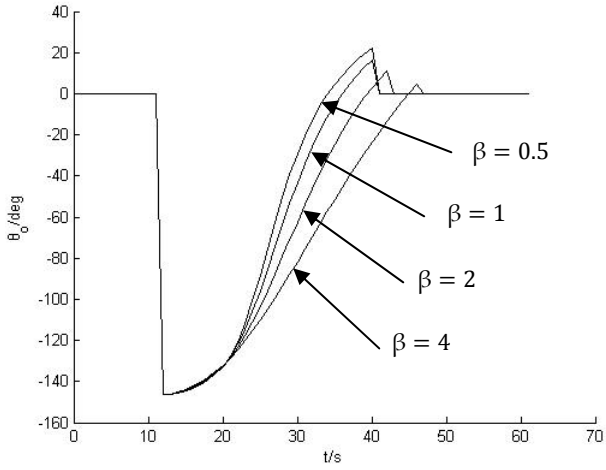

Figure 14. Curves of the directions of OEVV

Fig. 14 depicts the curves of the directions of the OEVV. In the beginning and the final phases, the direction angle is $\theta_o = 0$ because the robot isn't affected by any obstacle. At the moment when a robot is entering into and leaving the influence region of an obstacle, the value of θ_o increases abruptly from zero to some very large value or decreases abruptly from a large value to zero. This action is admissible, since the direction of OEVV is a virtual direction and it therefore does not need to accord with the requirements of a real robot's orientation angle. In the same way, the directions of OEVV change more significantly with a smaller β.

4.3 Navigation with Multiple Obstacles

The navigation results with multiple obstacles are given in Fig. 15. The start and goal positions are (0,0)m and (12,12)m, $\theta_r(t_0) = 45°$, $\beta = 1$, $D_0 = 3$, the weight factors for all obstacles are set as $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = 1$. It can be seen that our approach holds good in a multiple obstacle environment. Compared with the linear navigation law, when an obstacle is detected, our approach does not need to construct a polar histogram for free directions and select a local goal position (such as the G1, G2 and so on in Fig. 15) for collision avoidance, and the navigation path is shorter and smoother. In terms of the obstacles O_2 and O_3, the navigation paths are given in Fig. 16 with different weight factors. A larger γ_i implies more danger of the obstacle O_i, thereby a greater repulsive vector is produced to put the robot farther away from obstacle O_i. Fig. 17 and Fig. 18 depict the curves of orientation angle and linear velocity with $\gamma_2 = 1, \gamma_3 = 1$, clearly, they are among the bounds of the VAC.

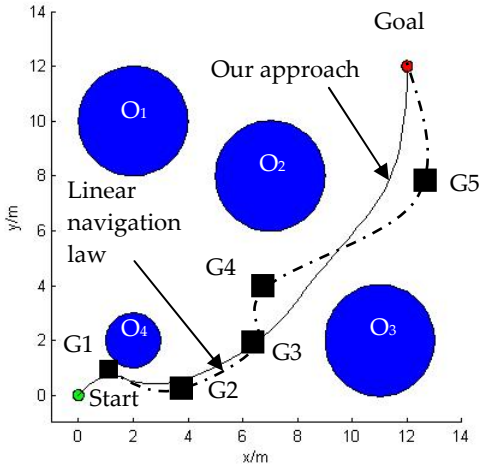

Figure 15. Experimental results with multiple obstacles when compared with the linear navigation law

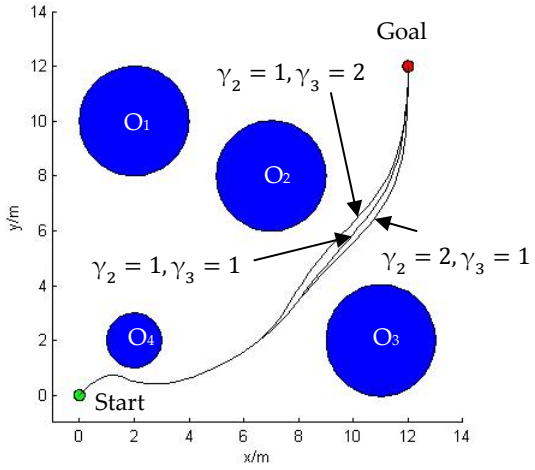

Figure 16. The effect of the weight factor γ_i on the navigation path

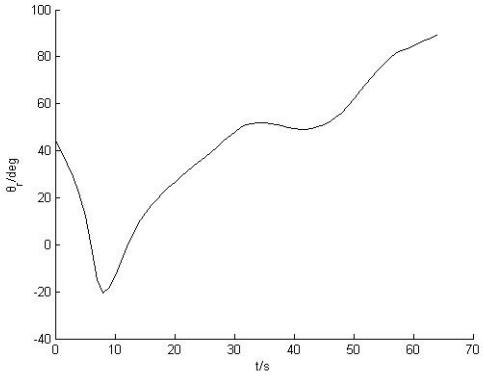

Figure 17. Curves of the orientation angle with multiple obstacles

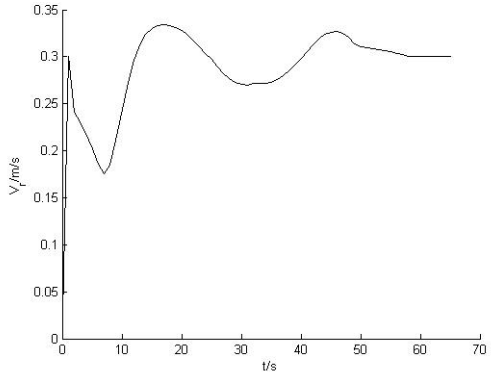

Figure 18. Curves of the linear velocity with multiple obstacles

5. Conclusions

In this paper we suggest an efficient, practical real-time navigation approach for nonholonomic mobile robots under velocity vector control based on the linear navigation law. When an obstacle is detected by a robot's sensory system, the obstacle is equivalent to a velocity vector according to the relative distant and relative direction between the robot and the obstacle. Then the velocity vector sum of the OEVV and LNVV makes the robot move to the desired goal position without any collision. In order to make the kinematic behaviour of the mobile robot more practical, a set of strategies for VAC is devised. The effectiveness and superiority of our approach is validated by extensive experimental results for no obstacles, a single obstacle and multiple obstacles. The suggested approach could be adapted to the real-time navigation problem of multiple mobile robots in dynamic environments with some improvements.

6. Acknowledgements

The work was partially supported by the National High-Tech Research and Development Plan of China (No.2007AA01Z313), the National Natural Science Foundation of China (No.60773022) and the Research Fund for the Doctoral Programme of Higher Education of China (No.20104307110003).

7. References

[1] H. Ehsan, J. M. Ghaffari, J. N. Ghaffari. Model-based PI-fuzzy control of four-wheeled omni-directional mobile robots. *Robotics and Autonomous Systems*, 2011, 59(11), pp.930-942.

[2] Q. B. Zhu, J. Hu, W. B. Cai. A new robot navigation algorithm for dynamic unknown environments based on dynamic path re-computation and an improved scout ant algorithm. *Applied Soft Computing*, 2011, 11(8), pp.4667-4676.

[3] J. Chakraborty, A. Konar, L. C. Jain, U. K. Chakraborty. Cooperative multi-robot path planning using differential evolution. *Journal of Intelligent and Fuzzy Systems*, 2009, 20(1), pp.13-27.

[4] F. Fahimi, C. Nataraj, H. Ashrafiuon. Real-time obstacle avoidance for multiple mobile robots. *Robotica*, 2009, 27, pp.189-198.

[5] Z. W. Yao, G. Kamal. Distributed Roadmaps for Robot Navigation in Sensor Networks. *IEEE Transactions on Robotics*, 2011, 27(5), pp.997-1004.

[6] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robot. *International Journal of Robotics Research*, 1986, 5(1), pp.90-98.

[7] F. Belkhouche, B. Belkhouche. Wheel mobile robot navigation using proportional navigation. *Advanced Robotics*, 2007, 4(1), pp.1-26.

[8] M. H. Mabrouk, C. R. McInnes. Solving the potential field local minimum problem using internal agent states. *Robotics and Autonomous Systems*, 2008, 56(12), pp.1050-1060.

[9] M. H. Mabrouk, C. R. McInnes. An emergent wall following behaviour to escape local minima for swarms of agents. *International Journal of Computer*, 2008, 35(4), pp.463-476.

[10] I. Mantegh, M. Jenkin, A. Goldenberg. Path planning for autonomous mobile robots using the boundary integral equation method. *Journal of Intelligent and Robotic Systems*, 2010, 59(2), pp.191-220.

[11] P. D. Claudia, M. M. Wilfredis, F Leonardo, G. Jose, F. L. Gerardo, C. G. Juan. Dynamic velocity field angle generation for obstacle avoidance in mobile robots using hydrodynamics. *Advances in Artificial Intelligence, 2008*, 5290, pp.372-381.

[12] P. D. Claudia, M. M. Wilfredis, F Leonardo, G. Jose, F. L. Gerardo. Fuzzy logic based speed planning for autonomous navigation under velocity field control. *Proceedings of the 2009 IEEE International Conference on Mechatronics*, 2009, April, Malaga, Spain.

[13] F. Belkhouche. Nonholonomic robots navigation using linear navigation functions. *Proceedings of the 2007 American Control Conference*, 2007, July, New York, USA.

[14] F. Belkhouche, P. Rastgoufard. Line of sight robot navigation toward a moving goal. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 2006, 36(2), pp.255-267.

[15] J H Oh. Solving a nonlinear output regulation problem: zero miss distance of pure PNG, *IEEE Transaction on Aerospace and Electron Systems*, 2002, 47, pp.169-173.

[16] J. Barraquand, B. Langlois, J. C. Latombe. Numerical potential field techniques for robot path planning. *IEEE Transactions on Systems, Man and Cybernetics*, 1992, 22(2), pp.224-241.

[17] Y. Koren, J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, April, Sacramento, USA, pp.1298-1402.