Taylor & Francis
Taylor & Francis Group

Advanced Robotics 26 (2012) 1841–1862

ADVANCED ROBOTICS
tandfonline.com/tadr

Full paper

Mobile Robot Navigation for Moving Obstacles with Unpredictable Direction Changes, Including Humans

Lingqi Zeng and Gary M. Bone*

Department of Mechanical Engineering, McMaster University, Hamilton, Ontario, Canada, L8S 4L7

Received 7 June 2011; accepted 6 January 2012

Abstract

In many service applications, mobile robots need to share their work areas with obstacles. Avoiding moving obstacles with unpredictable direction changes, such as humans, is more challenging than avoiding moving obstacles whose motion can be predicted. Precise information on the future moving directions of humans is unobtainable for use in navigation algorithms. Furthermore, humans should be able to pursue their activities unhindered and without worrying about the robots around them. An enhanced virtual force field-based mobile robot navigation algorithm (termed EVFF) is presented for avoiding moving obstacles with unpredictable direction changes. This algorithm may be used with both holonomic and nonholonomic robots. It incorporates improved virtual force functions and an improved method for selecting the sense of the detour force to better avoid moving obstacles. For several challenging obstacle configurations, the EVFF algorithm is compared with five state-of-the-art navigation algorithms for moving obstacles. The navigation system with the new algorithm generated collision-free paths consistently. Methods for solving local minima conditions are proposed. Experimental results are also presented to further verify the avoidance performance of this algorithm.

Keywords

collision avoidance, dynamic obstacles, human-friendly robots, mobile robots, virtual force field

1. Introduction

1.1. Challenges of avoiding humans

In many service applications, mobile robots need to share their work regions with obstacles including humans. Avoiding collisions with humans and obstacles is a fundamental requirement for these robots. The objective of navigation is to plan and control the motion of the robot from its initial position to its goal position while avoiding obstacles. The navigation problem for mobile robots has

*To whom correspondence should be addressed. E-mail: gary@mcmaster.ca

 DOI: 10.1080/01691864.2012.703166

been the focus of tremendous research effort. Most of the algorithms for mobile robot navigation are only suitable for avoiding stationary obstacles or/and moving obstacles whose future position and moving direction are predictable, such as other mobile robots. However, avoiding humans is different from avoiding other kinds of obstacles, and poses the following additional challenges:

(1) Human motion is unpredictable, changing its speed or direction arbitrarily. For example, human motion is holonomic. A human can stop and move sideways without turning or directly move sideways without stopping. Predicted human positions and moving directions based upon previous human motions are not reliable. Therefore, precise predictions are unavailable for use in navigation algorithms. This makes the navigation problem more difficult.
(2) For safety, humans must always possess a higher priority in a navigation system than robots or inanimate objects. Humans should be able to pursue their activities unhindered and without worrying about the robots around them.
(3) Considering human emotional–psychological reactions, when a robot avoids a human it should not come too close to the human or block the human's path. This can be considered rude behavior and could also frighten the human, which may cause a sudden action (such as jumping away) leading to human injury.

1.2. Related Research

Currently, the majority of navigation algorithms for mobile robots can be classified into two categories. The algorithms in the first category directly provide the collision-free path(s) of the robot(s) by assuming the shapes of obstacles are known and their motion is predictable. Furthermore, they typically neglect the robot(s) dynamics. The algorithms in the second category indirectly generate the collision-free path(s) by using the current kinematic and geometric information of the robot(s) and obstacles, such as the distances between obstacles and robot(s) and the current velocities of obstacles. These algorithms consider the robot(s) dynamics. The relevant literature on mobile robot navigation algorithms will now be reviewed.

Algorithms in the first category have the advantage that optimal collision-free paths can be found with stationary obstacles [1,7,18,19], and with moving obstacles whose motions are predictable [2–5,7,8–17,20,21]. We will only review the algorithms for avoiding moving obstacles since that is the focus of this paper. In some of the algorithms, by assuming the obstacles will move with zero acceleration, their current positions and velocities are used to build forbidden regions for the robot(s) velocities. Avoiding moving obstacles is accomplished by keeping the robot(s) velocities outside of those regions. These algorithms include

"collision map" [2,13], "velocity obstacles" [6,8,12], "curvature velocity method"[4,21], and "inevitable collision states"[11]. "Dynamic window"-based algorithms [5,16] have also been proposed. They compute a dynamic window around the robot based on the velocities that the robot can reach without causing a collision in the next time interval. These two types of algorithms do not work for obstacles like humans due to the lack of accurate position/moving direction predictions. Incremental search planning algorithms are another alternative. The "Lifelong A*" [10], "Learning Real-Time A*" [20], and "D*" [3,14] algorithms resemble the well-known A* algorithm [1] except that they perform the grid-based search incrementally, based on the current positions of the robot(s) and obstacle(s). Similarly, a predicted collision-free region was combined with the A* algorithm in [22]. More recently, randomized path planning algorithms have been proposed for avoiding moving obstacles, such as the "kinodynamic" [7,9], "rapidly-exploring random tree" [15], and "RAMP" [17] algorithms. They employ randomization to explore large state spaces and efficiently plan paths.

The navigation algorithms in the second category are based on the artificial potential field (APF) (introduced by Khatib [23]) or the virtual force field (VFF) (introduced by Borenstein and Koren [24]). They are normally simple to implement and consume less computational load than those in the first category. Since virtual forces can be obtained by using the gradient descent method on APFs, they are closely related and belong to the same category. These algorithms assume the existence of a repulsive artificial potential field or a repulsive virtual force surrounding each obstacle to push robots away from the obstacle. An attractive potential field or virtual force is also assumed to be acting to pull the robot to reach the goal. VFF-based or APF-based algorithms have been extensively studied for avoiding stationary obstacles [23–25,32,33,35] and have also been proposed for avoiding moving obstacles [26–31,33,36–41]. In these algorithms, the force functions or the potential fields are related to the distance from the robot to the goal and the distance to the obstacle(s). Sometimes, they incorporate the velocities of the obstacles and robots in the force functions and potential fields to help the robots avoid dynamic obstacles [26–28,30,36,38–41]. An "active region" surrounding each obstacle is often used to define where the repulsive potential field (or repulsive virtual force) from the obstacle is active [23–30,32–34,36,38–41]. If the robot is inside an obstacle's active region, the repulsive potential field/virtual force is applied to repel the robot away from the obstacle and the attractive potential field may also be active to move the robot towards goal during navigation. If the robot is outside the region, only the attractive potential field is applied to the robot to pull it towards the goal. The APF/VFF is divided into two parts by the boundary of the active region. In [8,30,36,41], another region was introduced that is very close to the obstacle. We will refer to this as the *critical region*. In the critical region, a robot may not have sufficient space to complete avoidance, so an alternative to VFF is needed inside this region. In [28,33], a component tangential to the repulsive component

of their potential field was added. This tangential component helps the robot to detour around the obstacle. Unlike the traditional APF, the force directions were specified independently of the gradient. In [30], a virtual force that acts perpendicular to the repulsive force was added. This force helps the robot to detour around the obstacle and reduces the arrival time when the robot reaches the goal. Since this force helps the robot detour around the obstacle, we will term it the *detour force*.

The APF-based and VFF-based navigation algorithms in the second category can be suitable for avoiding moving obstacles whose moving directions are unpredictable, such as humans. This is because the active region provides the space needed to handle the unpredictability.

Most VFF-based and APF-based mobile robot navigation algorithms suffer from local minima. If a robot becomes trapped in a local minimum then it cannot reach the goal. There are four common conditions which may produce a local minimum. With the first condition, an obstacle is located between the robot and the goal, and the centers of the robot, obstacle, and goal are collinear; we will term it the *collinear condition*. The velocity directions of the robot and the obstacle are also on this line. In the second condition, the goal is within the active region of an obstacle. Since the attractive potential/virtual force will be zero, but the repulsive potential/virtual force will be nonzero, the robot cannot stay at the goal. The third condition occurs when avoiding a U-shaped obstacle. In the fourth condition, the robot intends to pass through a narrow passage between two obstacles. Some research progress has been made on solving the local minima in VFF-based/APF-based algorithms. In [26], a time-varying nonlinear virtual force was used to solve the third condition. A detour force with a predefined fixed sense was proposed in [28,34]. The detour force helped to solve the first, second, and fourth local minima conditions. However, since their detour force sense was fixed, collisions may be incurred when avoiding moving obstacles, which will be further discussed in Section 4. In [41], they utilized the detour force to solve the first and second local minima conditions. In [32], an APF enhanced with fuzzy logic, local minima detection, and escaping methods was proposed. Simulation results showed their algorithm can solve the third and fourth local minima conditions. In [33], their APF incorporated a multiplicative and additive composition of attractive and repulsive potentials for avoiding stationary obstacles. Using this composition, the goal was reached for the first, third, and fourth conditions.

1.3. Contributions and Paper Organization

The main contributions of this paper are:

(1) An enhanced VFF-based (EVFF) algorithm for holonomic and nonholonomic robots with improved force functions is designed to avoid moving obstacles with unpredictable direction changes, including humans.

(2) Solutions for the four local minima conditions.
(3) An improved method for selecting the sense of the detour force to better avoid moving obstacles.
(4) Comparisons with five state-of-the-art navigation algorithms.

The organization of the paper is as follows. In Section 2, the active and critical regions are defined for circular obstacles. The virtual force functions are designed in Section 3. The control scheme for nonholonomic robots is provided in Section 4. Next, simulations for three fundamental and one challenging obstacle configurations are presented in Section 5. First, comparative simulations with the EVFF algorithm and five navigation algorithms are performed for a walking human with an unpredictable direction change. The algorithms that succeeded with the first configuration are then compared for two more configurations. After that, the simulation results for a challenging configuration that includes a walking human and three mobile robots are presented. In Section 6, the solutions for the four types of local minima conditions are provided. The experimental setup and an experimental result for avoiding a walking human in the collinear condition are given in Section 7. Finally, conclusions are drawn in Section 8.

2. Definitions of the Active and Critical Regions

In this section, the active region and the critical region for the case when the obstacle is a human will be explained first. The robot(s) and human(s) are both modeled as a moving cylinder. These models are projected onto the floor to produce disks. The active region $\mathbb{C}_2$ is defined as the region near the human. If any mobile robot is in $\mathbb{C}_2$, the repulsive and detour virtual forces should be activated to make the robot avoid and detour around the human. The critical region $\mathbb{C}_3$ is the region in close proximity to the human, and it is very dangerous if a robot intrudes into this region. Within this region, the robot cannot complete the avoidance and should stop to prevent, or at least mitigate, the collision. Any other action made by the robot could worsen the situation. Therefore, when the robot just enters $\mathbb{C}_3$ it will be fully decelerated until it is stopped. We define $\mathbb{C}_1$ as the region which excludes $\mathbb{C}_2$ and $\mathbb{C}_3$. $\mathbb{C}_1$ contains the goal. If the robot is in this region, collision avoidance is unnecessary and the robot just moves towards its goal. Since the human's motion is arbitrary, it can change its moving velocity and direction any time subject to a limit in linear velocity magnitude. Note that for safety, we need to consider the arbitrary direction changes made by the human, so the active and critical regions for humans are represented by disk-shaped regions to consider all directions (i.e. assuming the angular velocity of the human is infinite to be conservative) and $r_2 > r_3$ to provide space for the avoidance maneuver, where r_2 and r_3 are the radii of the active and critical regions, respectively.

The sizes of the regions are related to the shapes and velocity limits of the obstacle(s) and robot and the acceleration limits of the robot. For a stationary obstacle, a slower robot only requires smaller active and critical regions. Similarly, for a moving obstacle, a slower obstacle velocity requires smaller regions. However, with a moving obstacle, a slower robot necessitates larger regions since it requires more time to avoid the obstacle. For brevity, we will not present our analytical method (described in [44]) to determine the sizes of active and critical regions here.

In this paper, the maximum human velocity is assumed to be 1 m/s (i.e. the typical walking speed [42]). The average step length of a human is about 0.8 m [43]. One half of this value will be used as the radius of the human model, i.e. $\rho_h = 0.4$ m. Our holonomic robot has a radius of: $\rho_r = 0.2$ m. The maximum speed and acceleration of our robots equals 0.7 m/s and 10 m/s^2. Using our method, if the obstacle is a stationary obstacle with a 0.1 m radius, we obtain $r_{3,st} = 0.4$ m and $r_{2,st} = 0.7$ m for the nonholonomic robot. If the obstacle is another nonholonomic robot with $\rho_r = 0.2$ m, then: $r_{3,rbt} = 0.6$ m and $r_{2,rbt} = 1.8$ m. If the obstacle is a walking human, $r_3 = 0.6$ m and $r_2 = 2.5$ m. Note that the sizes of the regions for the human were calculated for the worst case of robot motion and human reaction (see [44] for details).

3. Design of the Virtual Force Functions

This section begins by discussing the case of a robot avoiding a human as an example of a dynamic obstacle. The geometric configuration of the goal, the robot, and the human for a navigation system is presented in Fig. 1. Note that the radii of the robot and human models are not shown in this figure. We assume that the goal is stationary and outside of the active region(s) of the obstacle(s)

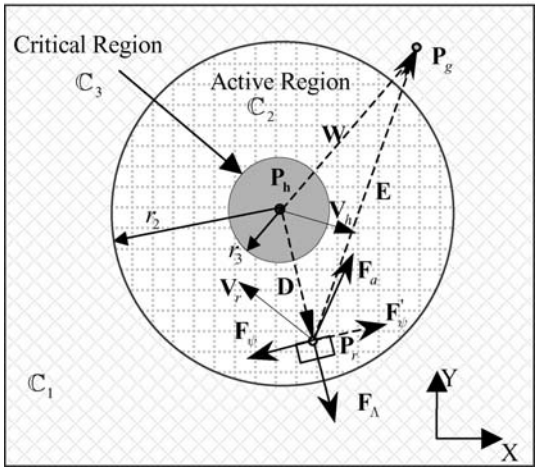

Figure 1. Configuration of the navigation problem for a mobile robot and a human obstacle.

for safety. In this figure, $\mathbf{P}_g$, $\mathbf{P}_r$ and $\mathbf{P}_h$ are the current position vectors of the goal, the robot, and the human, respectively. $\mathbf{V}_r$ and $\mathbf{V}_h$ are the current velocity vectors of the robot and the human. We have:

$$\mathbf{E} = \mathbf{P}_g - \mathbf{P}_r \tag{1}$$

$$\mathbf{D} = \mathbf{P}_r - \mathbf{P}_h \tag{2}$$

and

$$\mathbf{W} = \mathbf{P}_g - \mathbf{P}_h \tag{3}$$

where $\mathbf{E}$ is the vector from the robot to its goal, $\mathbf{D}$ is the vector from the obstacle to the robot, and $\mathbf{W}$ is the vector from the obstacle to the goal.

An attractive virtual force is used to guide the robot to the goal. It is activated when the robot is within $\mathbb{C}_1$ or $\mathbb{C}_2$. Its force function is defined as:

$$\mathbf{F}_a = \begin{cases} K_1\mathbf{E} + K_2\mathbf{E} & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \cup \mathbb{C}_1 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_3 \end{cases} \tag{4}$$

where K_1 and K_2 are the positive attractive virtual force gains.

A repulsive virtual force is used to keep the robot away from the human and is activated when the robot is within $\mathbb{C}_2$. The repulsive force is normally a function of the distance between the human and robot, $d = \|\mathbf{D}\|_2$,, and its rate of change, $\dot{d}$. Its direction is along $\mathbf{D}$. We propose the new repulsive force function:

$$\mathbf{F}_\Lambda = \begin{cases} (K_3\Lambda + K_4\dot{\Lambda}^*)\mathbf{U}_\Lambda & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \cup \mathbb{C}_3 \end{cases} \tag{5}$$

where K_3 and K_4 are the positive repulsive force gains; $\mathbf{u}_\Lambda$ is a unit vector along $\mathbf{D}$, pointing away from the human; $\Lambda = \frac{(r_2-d)^2}{d-r_3}$; and $\dot{\Lambda}^* = -\dot{d}\frac{(r_2-d)^2}{(d-r_3)^2} = -K_\Lambda\dot{d}$. Note that $\mathbf{P}_r \in \mathbb{C}_2 \Leftrightarrow r_3 < d < r_2$. The numerator term of Λ, $(r_2 - d)^2$, causes a gentle increase of the repulsive force when the robot enters the active region. Since $r_2 - d \to 0$ when $d \to r_2$, the repulsive force is continuous at the boundary between $\mathbb{C}_1$ and $\mathbb{C}_2$. This helps to reduce path oscillations. The denominators, $(d - r_3)$ and $(d - r_3)^2$, cause the force to increase greatly when the robot is near the boundary of $\mathbb{C}_3$. This is useful for avoiding multiple obstacles. When the robot is pushed by other obstacle(s) to be close to $\mathbb{C}_3$, the repulsive force from this obstacle is significantly increased. This tends to push the robot away from $\mathbb{C}_3$ of the obstacle. When the obstacle is a human, this should keep the robot away from the center of the human with a distance of least r_3. Note that the

velocities of the human and robot are included in $\dot{\Lambda}^*$ of (5). When the human and robot approach each other, $\dot{d}<0$, so $\dot{\Lambda}^*>0$ and the repulsive force will increase to push the robot away.

The detour force, $\mathbf{F}_\psi$, is a virtual force perpendicular to $\mathbf{u}_\Lambda$. Its main purpose is to push the robot to detour around the human while still moving to the goal. Its other purpose is dealing with the collinear condition. The first issue when designing this force is its sense. This issue has not been adequately addressed in the existing literature. If the sense of the force was $\mathbf{F}'_\psi$ (see Fig. 1), the robot would have to pass in front of the obstacle. This is acceptable for stationary obstacles, but is not desirable with humans for physical and emotional reasons, as was discussed in Section 1.1. The choices for the sense of the force are illustrated in Fig. 2. In Fig. 2a, the goal and robot are on different sides of the human's velocity line. In this case, the detour force should cause the robot to detour around behind the human to prevent blocking the human. In Fig. 2b, since the goal and robot are on the same side of the human's velocity line, the robot will not block the human. Then, the sense of $\mathbf{F}_\psi$ should point it towards the goal to help the robot reach its goal faster. For the collinear condition, since the robot, its goal and the obstacle are collinear and their velocity lines are also along this line, the condition for avoidance is symmetrical. When there are other obstacles, we should choose the sense to satisfy $\mathbf{u}_\psi \cdot \sum \mathbf{F}_{\psi,i}>0$, where $\sum \mathbf{F}_{\psi,i}$ is the sum of the detour forces from the other obstacles and $\mathbf{u}_\psi$ is the unit vector in the direction of the detour force. There is no velocity line for stationary obstacles. Therefore, to reach the goal early, the sense should be chosen towards the goal with stationary obstacles (see Fig. 2b). This force should be zero when the robot is between the goal and the human (on the same line), or $\alpha = \beta$ (α is the angle of the vector $\mathbf{W}$ to the positive X-axis and β is angle of $\mathbf{D}$). So, the proposed detour force function is:

$$\mathbf{F}_\psi = \begin{cases} (K_5\psi + K_6\dot{\psi}^*)\mathbf{u}_\psi & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \cup \mathbb{C}_3 \end{cases} \tag{6}$$

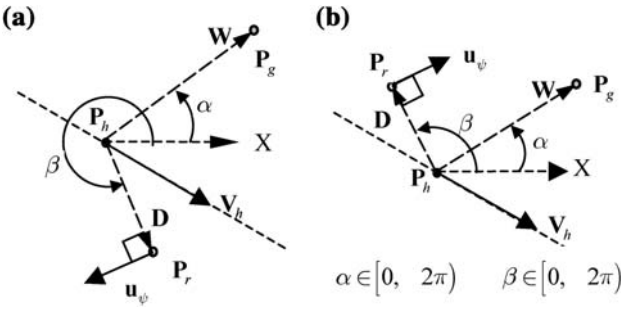

Figure 2. The sense of the detour force.

where K_5 and K_6 are the positive detour force gains; $\psi = (r_2 - d)^2 \Phi$; and $\Phi = \alpha - \beta$; $\dot{\psi}^* = (r_2 - d)^2 \dot{\Phi}$. Note that $\alpha = \pi$, $\beta = 0$, and $\mathbf{F}_\psi \neq \mathbf{0}$ in the collinear condition. For other obstacles, such as stationary obstacles, (5) and (6) are also applicable by changing r_2 and r_3 to the corresponding values for those obstacles.

The VFF is the combination of the three forces. Thus, the VFF for the robot with a single human/obstacle is:

$$\mathbf{F}_V = \begin{cases} \mathbf{F}_a & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \\ \mathbf{F}_a + \mathbf{F}_\Lambda + \mathbf{F}_\psi & \text{if } \mathbf{P}_r \in \mathbb{C}_2 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_3 \end{cases} \tag{7}$$

If the robot intrudes into $\mathbb{C}_3$, since it may not have enough space for collision avoidance the VFF is not used, and the robot is decelerated to a stop. This stopping action should prevent, or at least mitigate, the collision. Any other action made by the robot could worsen the situation. According to the above analysis, if the robot is sharing its work region with N obstacles (which could include other robots), the force field will be:

$$\mathbf{F}_V = \begin{cases} \mathbf{F}_a & \text{if } \mathbf{P}_r \in \mathbb{C}_1 \\ \text{undefined} & \text{if } \mathbf{P}_r \in \mathbb{C}_{3,i}\ i \in [1, \ldots, N] \\ \mathbf{F}_a + \sum_{i=1}^{N} c_i (\mathbf{F}_{\Lambda,i} + \mathbf{F}_{\psi,i}) & \text{otherwise} \end{cases} \tag{8}$$

where $c_i = \begin{cases} 1 & \text{if } \mathbf{P} \in \mathbb{C}_{2,i} \\ 0 & \text{otherwise} \end{cases}$; $\mathbf{F}_{\Lambda,i}$ and $\mathbf{F}_{\psi,i}$ are the repulsive and detour forces for the ith obstacle, respectively.

4. Nonholonomic robot control system

VFF-based algorithms cannot be used with nonholonomic robots without special extensions. The virtual forces are defined as vectors in two-dimensional Cartesian coordinates, i.e. $\mathbf{F}_V = [F_x \quad F_y]^{\mathrm{T}}$. For nonholonomic robots, the force component perpendicular to the robot's heading direction will be neglected since the robot cannot move sideways without turning first. As a consequence, the robot would maintain its previous moving direction rather than moving sideways to avoid the obstacles. The control system we propose for using any Cartesian VFF with a nonholonomic robot has three levels, as shown in Fig. 3. The combination of the first and second levels is similar to the *nonholonomic motion planner* presented in [45]. The first level is the path planner. In the planner, using the equations presented in Section 3, the virtual force, $\mathbf{F}_v$, is generated from the current measured position(s) of the dynamic obstacle(s), the position(s) of the stationary obstacle(s), the measured robot position, $\mathbf{P}_r$, and the velocities estimated by backwards differencing. A sensor system, typically based on computer vision, is

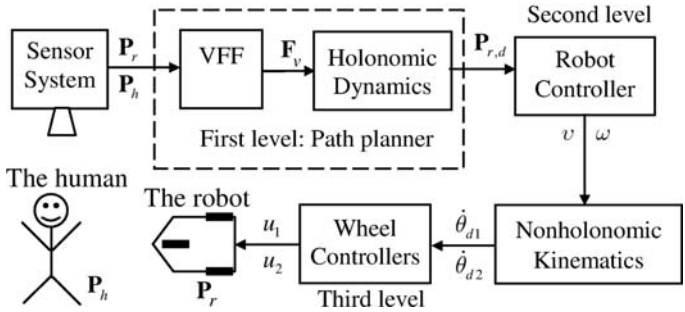

Figure 3. The nonholonomic robot control system. The case of a single human is shown as an example.

used to measure the positions. Note that Fig. 3 shows the case of one human whose measured position is $\mathbf{P}_h$, as an example. $\mathbf{F}_v$ cannot be applied directly to the nonholonomic robot dynamics since the sideways force component would be ignored. To include this force component, we apply $\mathbf{F}_v$ to the dynamic equations of a reference holonomic robot. The holonomic dynamics is defined as the dynamics of a point mass [30], allowing the acceleration vector caused by $\mathbf{F}_v$ can be easily obtained. Next, with the acceleration vector, known, the reference path $\mathbf{P}_{r,d}$ of the reference robot is derived by numerical integration. The robot controller computes the linear and angular velocities, v and ω to track $\mathbf{P}_{r,d}$ under the nonholonomic constraint. In this paper, the simple nonlinear controller from [46] is used. Next, v and ω are transformed to the desired angular velocities ($\dot{\theta}_{d1}$ and $\dot{\theta}_{d2}$) of the driven wheels of the robot. The wheel controllers generate the control signals (u_1 and u_2) used to drive the wheels.

5. Simulations for Nonholonomic Robots

5.1. Avoiding a Walking Human with an Unpredictable Direction Change

Simulations with five state-of-the-art navigation algorithms from the two categories described in Section 1 are compared with the EVFF algorithm in this subsection. A human path with an unpredictable direction change is used. The simulation paths, the distance from the robot center to the goal, e, and the center-to-center distance from the robot to the human, d, are presented in Fig. 4. In these simulations, a nonholonomic robot starts from $(4, 0)$ m and must avoid the walking human to reach its goal at $(0, 0)$ m. Note that in Fig. 4a and b (and subsequent figures), to give an indication of the speed, dots have been placed along the human's path at 0.7 s intervals. The walking human initially starts from $(0.6, 0)$ m with a $(1, 0)$ m/s velocity. After 1 s, the human slows down with a $(-1, 0)$ m/s^2 acceleration until it stops and then moves sideways with a $(0, 1)$ m/

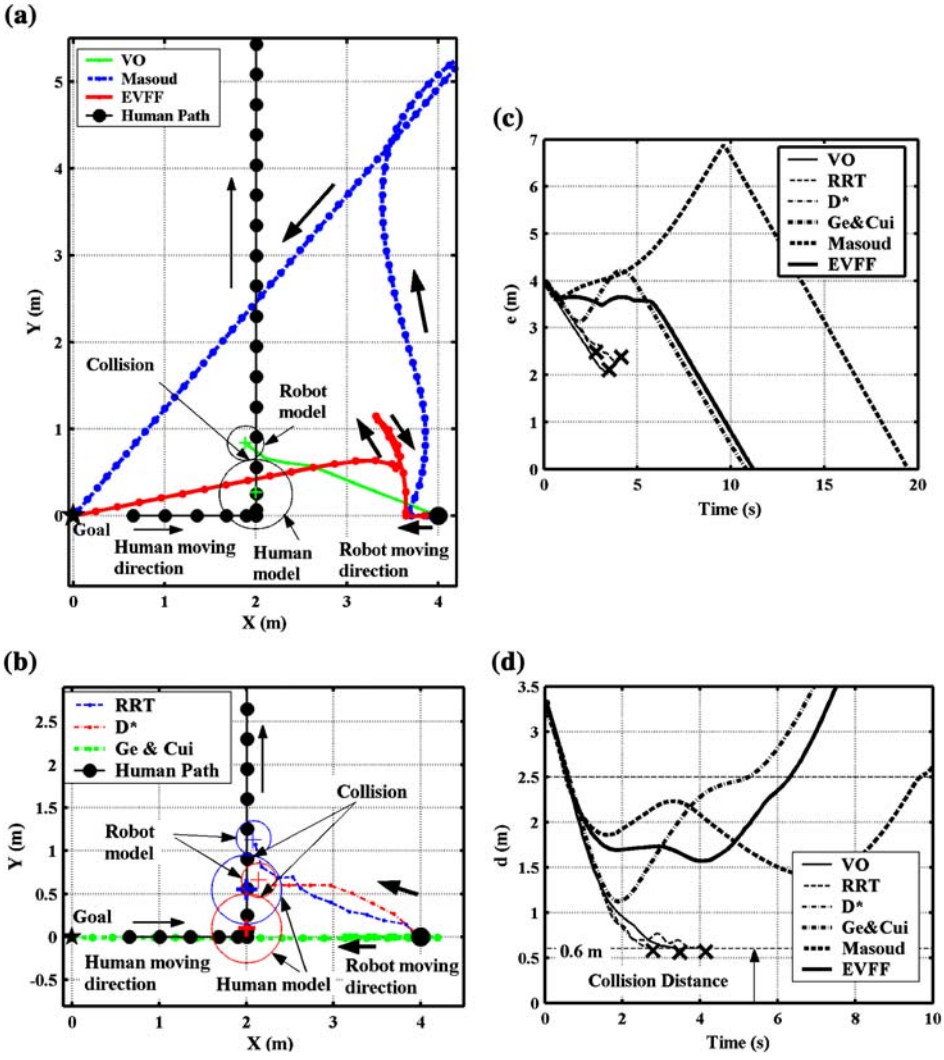

Figure 4. Comparison of the simulation results for the six algorithms with a motion-unpredictable human: (a) and (b) robot navigation paths; (c) distance from the robot to goal, e; and (d) center-to-center distance from the robot to the human, d.

s^2 acceleration. After the human's velocity reaches $(0, 1)$ m/s, the human keeps this velocity. The human's position is updated using a sampling time of 0.06 s. In Fig. 4, VO denotes the velocity obstacle navigation algorithm [12], D^* denotes the Field D^* algorithm [14], *RRT* denotes the rapidly exploring random trees algorithm [15], Masoud denotes his APF-based algorithm [34], and Ge and Cui denotes their VFF-based algorithm [30]. The gains for the latter two algorithms were optimally tuned. The detailed tuning process is omitted here for brevity and can be found in [44]. To put the EVFF algorithm at a slight

disadvantage, its gains were not optimally tuned and were given as follows: $K_1 = 3,\ K_3 = 20,\ K_5 = 40,$ and $K_2/K_1 = K_4/K_3 = K_6/K_5 = 0.25$. The sensitivity of the navigation performance to the EVFF gains will be discussed in the next subsection. The first three algorithms belong to the first category, while the remainders belong to the second. During the simulations, the future motion of the human, including the walking direction change, is unknown for all six algorithms. From Fig. 4, we can see that the three algorithms in the first category incur collisions after the human changes its direction, since $d<0.6$ m (recall that the sum of the radii of the robot and the human models equals 0.6 m). The algorithms in the second category all successfully completed the navigation. The arrival times with the EVFF and *Ge and Cui* algorithms are similar (9.9 s and 10.4 s, respectively), but the time increases to 14.0 s with the *Masoud* algorithm. Since they did not incur collisions, the remainder of the comparisons will focus on these three algorithms.

5.2. Avoiding a Walking Human

A navigation algorithm should also be able to avoid a moving human under the collinear condition. In this case (Fig. 5), the human starts from position $(0, 0)$ m, moves from left to right and then stops at $(4, 0)$ m. At the same time, the nonholonomic robot starts from $(4, 0)$ m, moves from right to left, and towards its goal at $(0, 0)$ m. The maximum human–robot approach velocity is 1.7 m/s. Since the *Ge and Cui* algorithm produces a zero detour force in the collinear condition, after initially backing away, the robot collides with the human at 3.7 s. The nonzero detour forces of the *Masoud* and EVFF algorithms push the robot sideways to avoid the human. We can see that the robot needs to change its moving direction sharply. The reason is the human's approach velocity acts to increase the magnitude of repulsive and detour forces compared with the stationary obstacle. The robot must turn quickly to avoid the human. The minimum distance between the human and the robot during the navigation is about 1 m for both algorithms, as shown in Fig. 5c.

The sensitivity of the navigation performance to the EVFF gains, K_1, K_3, and K_5 will now be investigated. Recalling Fig. 1, a large attractive force helps the robot reach the goal earlier and at the same time a large repulsive force causes the robot to avoid the human at a farther distance. Then the robot needs to travel a longer way to the goal; the time cost the robot to arrive the goal, termed T_{arrive}, increases. A detour force makes the robot avoid the human and reach the goal faster. In Fig. 5, we also compared the simulation results of the EVFF algorithm with different gains. The two most popular navigation performance indices, T_{arrive}, and the minimum distance during the navigation, termed $d_{\min}$, are also shown in Table 1 for comparison. With doubled attractive force gain K_1, the robot path has no obvious changes and T_{arrive} and $d_{\min}$ are only slightly reduced (i.e. 1%). When the repulsive force gain K_3 doubles, T_{arrive} and $d_{\min}$ increase by

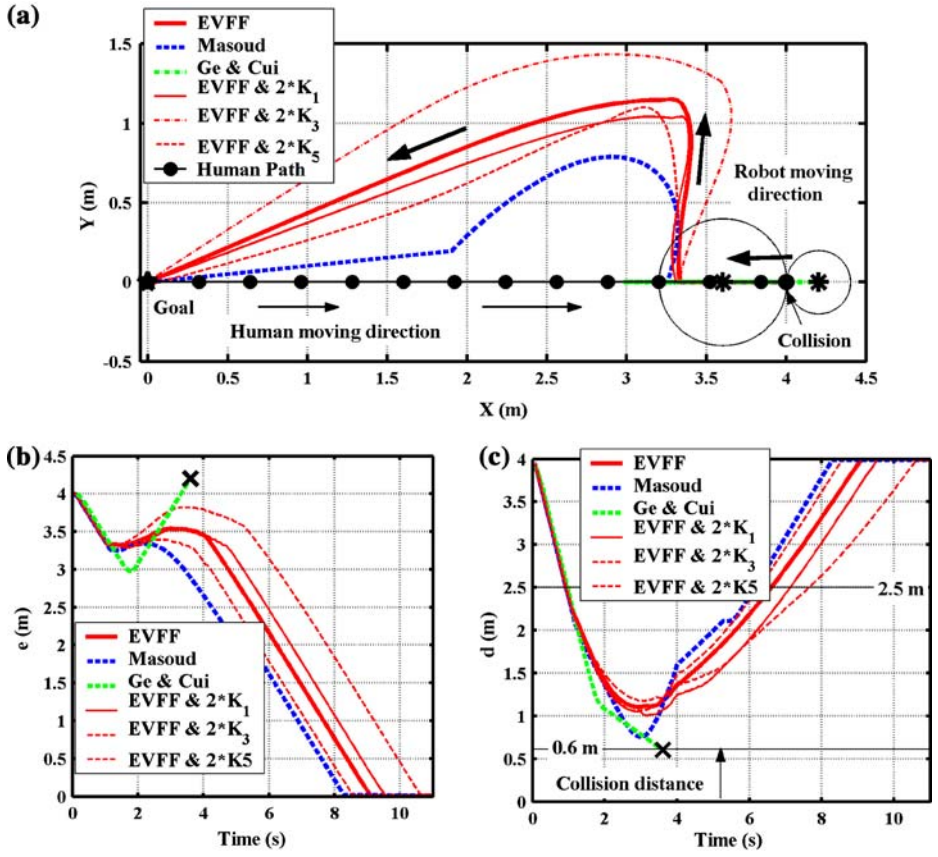

Figure 5. Simulation results for the three VFF-based algorithms for avoiding a walking human in the collinear condition. (a) Robot navigation paths, (b) *e*, and (c) *d*.

Table 1.
Comparison of the performance indices for the *EVFF* algorithm

Performance Index	$K_1 = 3$; $K_3 = 20$; $K_5 = 40$;	$K_1 = 6$; $K_3 = 20$; $K_5 = 40$;	$K_1 = 3$; $K_3 = 40$; $K_5 = 40$;	$K_1 = 3$; $K_3 = 20$; $K_5 = 80$;
T_{arrive} (s)	8.0	7.9	8.6	7.6
d_{min} (m)	1.00	0.99	1.07	0.98

7%. Doubling K_5 decreases T_{arrive} by 5% and d_{min} by 2%. From those results, when tuning EVFF, K_3 and K_5 are more important than K_1. The detailed analysis of the sensitivity of the performance indices to each of these gains was presented in [44]. From [44], the sensitivity of a total performance measure (including T_{arrive}, d_{min}, and the energy consumption) to a change of each gain is less than

1%. In general, considering both T_{arrive} and $d_{\min}$, the gains should be selected such that $K_1 < K_3 < K_5$.

In Fig. 5, the collinear condition was used since it is the worst case for the EVFF and *Ge and Cui* algorithms. It is not the worst case for the *Masoud* algorithm. In the next simulation, the case of a human moving parallel to the initial robot velocity direction is performed with only the EVFF and *Masoud* algorithms. In this configuration (Fig. 6), the human starts from position coordinates $(0.6, 0.6)$ m, moves from left to right, and then stops at $(4, 0.6)$ m. At the same time, the robot starts from $(4, 0)$ m and moves from right to left and towards its goal at $(0, 0)$ m. In Fig. 6, the navigation paths and distances e and d for the two algorithms are presented. We can see that the EVFF algorithm completes the navigation without a collision. The *Masoud* algorithm produces a collision at 3.1 s due to the incorrect fixed sense of the detour force. The detour force drags the robot to move in front of the human and that action incurs a collision.

5.3. Navigation with a Walking Human and Three Robots

A more challenging test is navigation including a walking human with an unpredictable direction change and three mobile robots. Simulation results for the EVFF algorithm are shown in Fig. 7. The walking human's path is the same as in Section 5.1. Robot A starts from $(4, 0)$ m and tries to reach its goal at $(-2, 0)$ m, while Robot B starts from $(2, -2)$ m and aims for $(2, 3)$ m, and Robot C starts from $(1, 3)$ m and aims for $(3, -1)$ m. Robot A is assigned the highest priority, followed by Robot B, and then Robot C. This is accomplished by setting Robot A to avoid only the walking human, while Robot B must avoid the human and Robot A, and Robot C must avoid the human and the other two robots. The robots safely reach their goals at 10, 10.5, and 12.9 s, respectively.

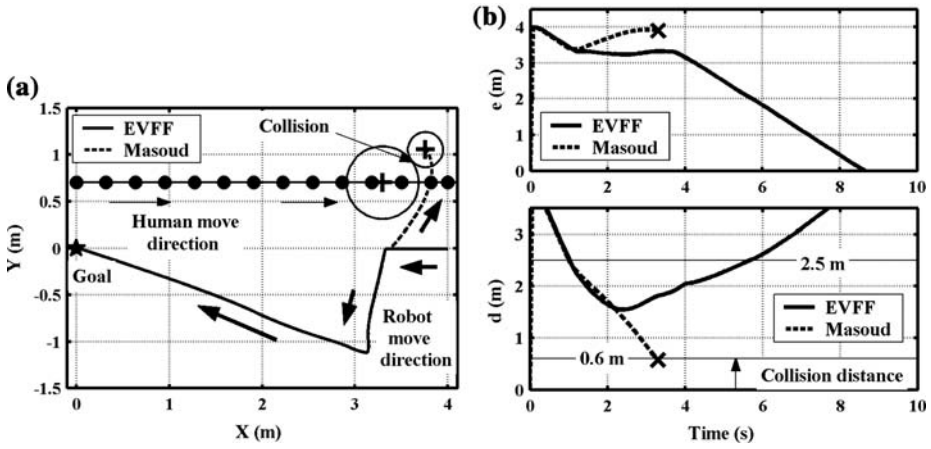

Figure 6. Simulation results with two VFF-based algorithms for avoiding a walking human in the noncollinear condition. (a) Robot navigation paths and (b) e and d.

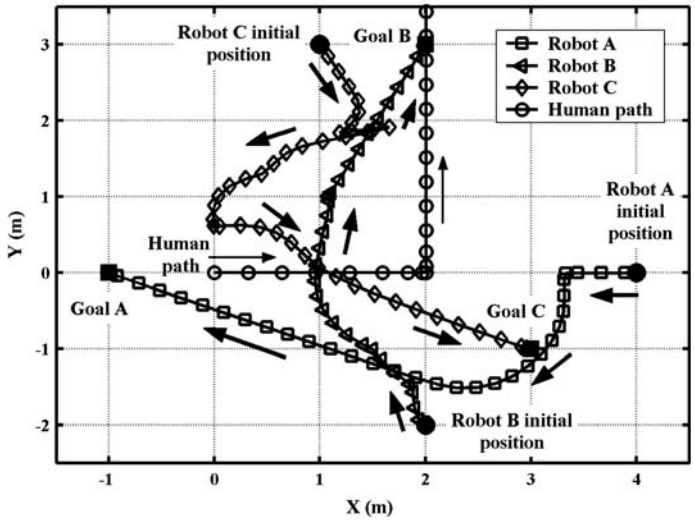

Figure 7. Simulated robot paths with the EVFF algorithm for avoiding a walking human and three nonholonomic mobile robots.

The minimum center-to-center distance to the human for the three robots is over 1.1 m. The minimum distance among the three robots is 1 m.

6. Handling Local Minima Conditions

In Fig. 5, we have shown that the EVFF algorithm can overcome the first local minima condition discussed in Section 1.2. For the second condition (i.e. when the goal is within the active region of an obstacle), if the obstacle is moving, the solution is to temporarily move the goal outside the active region until the obstacle leaves. After the original goal position is outside of the active region, the original goal position is restored. If the obstacle is stationary, the velocity limit of the robot should be reduced, allowing the size of the active region of the obstacle to be decreased. This reduction should be continued until the goal is outside of the reduced active region, and then the robot can reach the goal.

For the third local minima condition, with nonconvex polygonal obstacles (e.g. a U-shaped obstacle), the polygon must first be decomposed into convex pieces (e.g. three rectangular pieces, and their $\mathbb{C}_2$ as shown in Fig. 8 by using the dash lines). The regions and forces are then calculated separately for each piece, and the forces summed to obtain $\mathbf{F}_v$ for the given $\mathbf{P}_r$. By selecting the sense of the detour force heuristically to shorten the path to the goal, our method can solve this condition. The simulation results for avoiding a U-shaped obstacle with a nonholonomic robot are presented in Fig. 8. In this simulation, the corners of the U-shaped obstacle are at $(2.0, -1.1)$, $(2.6, -1.1)$, $(2.6, -1.0)$, $(1.9, -1.0)$,

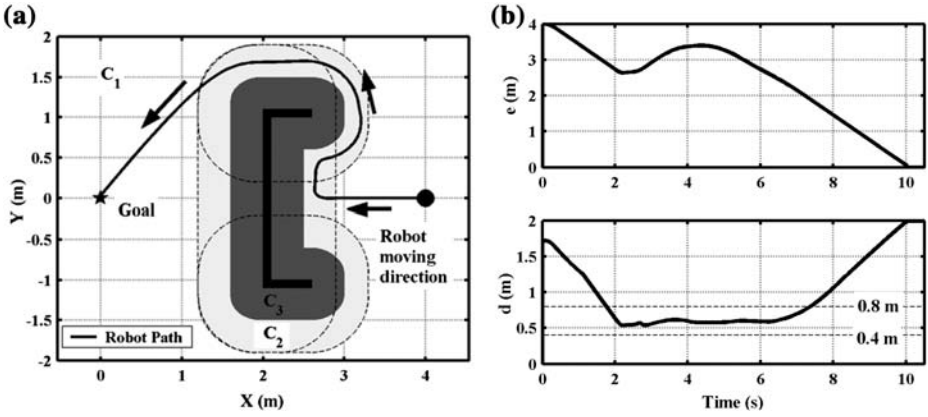

Figure 8. Simulation results for avoiding a U shaped obstacle, i.e. the third local minima condition. (a) Robot navigation path and (b) e and minimal d to the exterior contour of the obstacle.

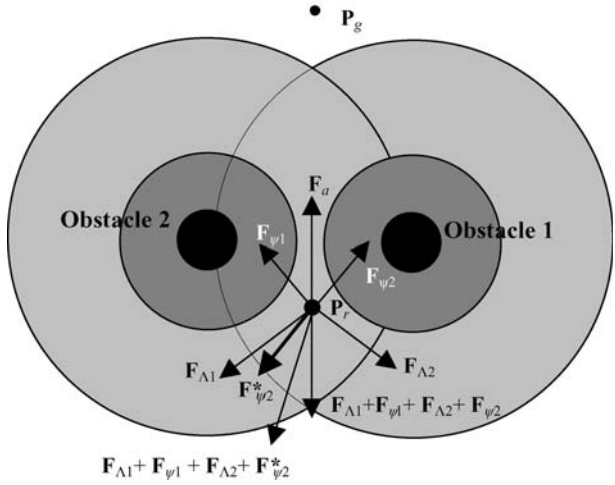

Figure 9. The fourth local minima condition and the solution with the EVFF algorithm.

(1.9, 1.0), (2.6, 1.0), (2.6, 1.1), and (2.0, 1.1) m. The robot starts from (4, 0) m and its goal is at (0, 0) m. Since analytically calculating the active region for this obstacle is intractable, we manually selected $r_{2,U} = 0.8$ m and $r_{3,U} = 0.4$ m. From the figure, we can see the robot avoids the U-shaped obstacle and reaches its goal at 10.1 s. Note that in Fig. 8b, d denotes the minimum distance from the robot to the exterior contour of the U-shaped obstacle during the navigation.

The fourth local minima condition is illustrated in Fig. 9. Note that the vectors are not shown to scale. In this figure, we can see the existence of a narrow passage between the two disk-shaped obstacles. When the robot tries to pass

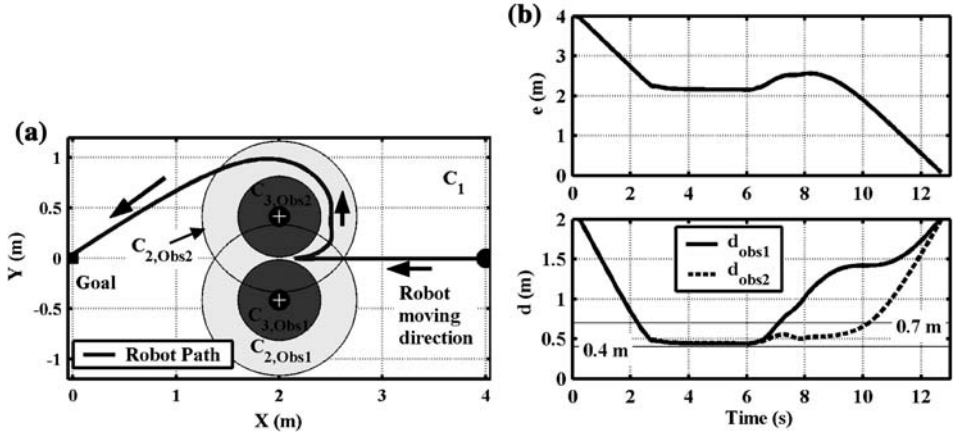

Figure 10. Simulation results with the EVFF for the fourth local minima condition. (a) Robot path and (b) e and the center-to-center distance from the robot to the obstacles, d.

through this passage, the robot will stop at the position $\mathbf{P}_{LM}$ since at this point the magnitude of the VFF is zero; i.e. $\mathbf{F}_a + \mathbf{F}_{\Lambda 1} + \mathbf{F}_{\psi 1} + \mathbf{F}_{\Lambda 2} + \mathbf{F}_{\psi 2} + \mathbf{F}_L = \mathbf{0}$. Then, the robot will remain at $\mathbf{P}_{LM}$ and will not reach its goal. To solve this local minima condition, when the robot is stopped at a point other than its goal, the sense of the detour force from one obstacle should be switched and this sense should be kept until the robot is outside of the active region of that obstacle. In Fig. 9, the switch was performed for Obstacle 2 and the altered detour force is labeled as $\mathbf{F}^*_{\psi 2}$. Simulation results with this method are presented in Fig. 10.The center positions of the obstacles are located at $(2.0, -0.42)$ and $(2.0, 0.42)$ m. The robot starts from $(4, 0)$ m and moves towards its goal at $(0, 0)$ m. The local minimum is detected and the sense of the detour force from Obstacle 2 is switched at 6 s (after the robot stops for about 1 s). Then the robot avoids the two obstacles and reaches its goal at 12.4 s.

7. Experiments with a Nonholonomic Robot

7.1. Experimental Setup

In order to further verify the avoidance performance of our algorithm, a differential-drive nonholonomic robot was built for performing experiments. The kinematic equations of the robot are excluded here for brevity, and can be found in [30]. The experimental setup includes a calibrated color video camera (PGR model DR2 HICOL) for capturing images of the human and the robot. Color patches are attached to the human's shoulder and the robot at known Z heights, as shown in Fig. 11. With a standard PC, the X-Y positions of the human and robot are reconstructed from the image centroids of the color patches using the

camera calibration matrix and the patch heights. The sampling frequency of the image processing is 16.7 Hz. The maximum error of the position measurements is 0.025 m. The position data are transferred via serial communication to a

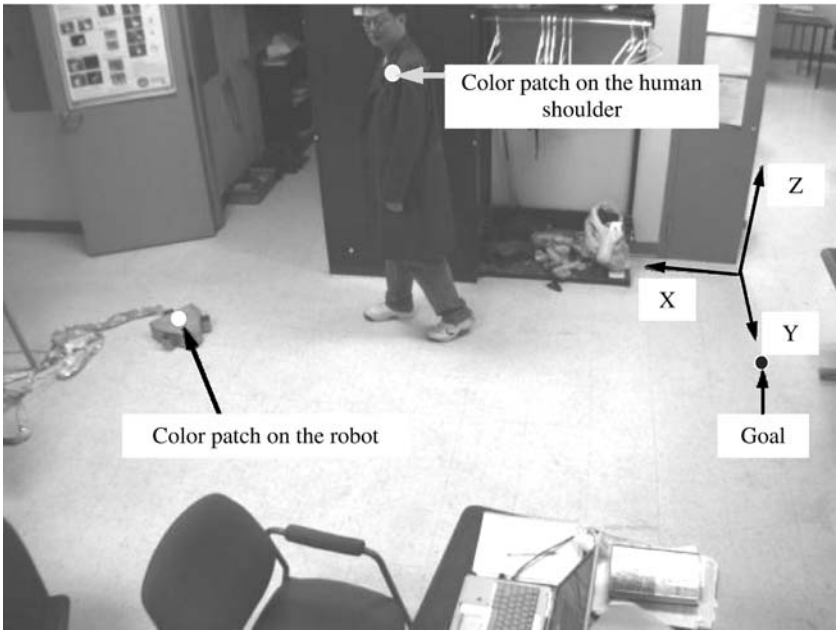

Figure 11. The video camera's view during the experiments.

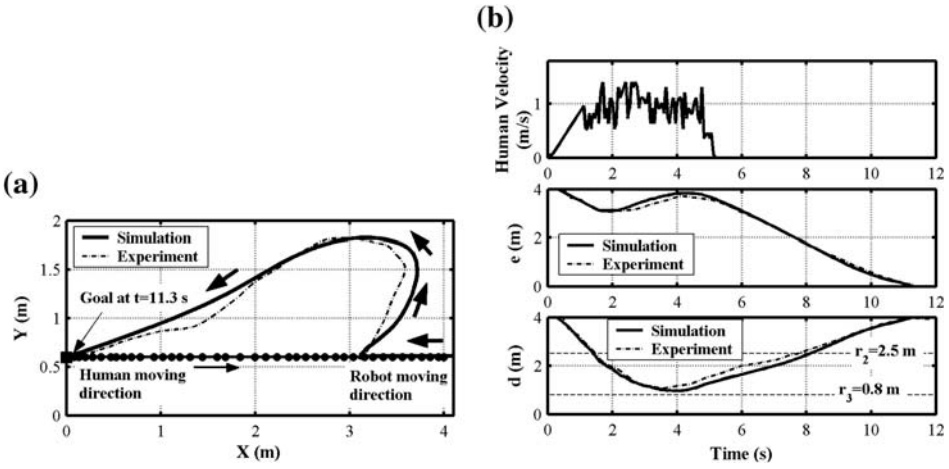

Figure 12. Experimental results when avoiding a walking human under the collinear condition with the EVFF algorithm.

second PC. The position(s) of the stationary obstacle(s) are predefined. The second PC acts as the path planner, robot controller, and wheel controller, configured as discussed in Section 4.

7.2. Avoiding a Moving Human in the Collinear Condition

Avoiding a moving human under the collinear condition will be used to demonstrate the experimental performance of the EVFF. In this case, shown in Fig. 12, the human starts from position $(0.2, 0.60)$, moves from left to right, and then stops at $(4.0, 0.60)$. At the same time, the robot starts from $(3.99, 0.60)$ and moves from right to left and towards its goal at $(0.0, 0.60)$. The maximum approach velocity was 1.7 m/s. For this case, the existing VFF algorithms either result in a collision or the robot failing to reach the goal. Fig. 12 shows that the measured human velocity (i.e. V_h) varies by ± 0.4 m/s, and has an average of around 1.0 m/s while walking (i.e. before 5.2 s). The minimum distance between the human and the robot is over 0.9 m in the simulation (around 1.0 m in the experiment), and larger than r_3 (0.8 m for humans). The robot is always outside of the critical region of the human. It reaches the goal at 11.3 s. Note that the human positions from the experimental data are used in the simulation. The discrepancy between the simulation and experimental results is mainly caused by the error of the vision system and wheel slip. The vision system error also influences the calculation of the virtual forces and causes a larger difference between the simulation and experimental results.

8. Conclusions

In this paper, a novel VFF-based navigation algorithm for holonomic and nonholonomic mobile robots has been proposed. The VFF is designed for avoiding moving obstacles with unpredictable direction changes, such as humans. It incorporates improved functions for the repulsive and detour forces. The sense of the detour force is chosen to drive the robot behind the human and to avoid tripping or frightening them. Methods for solving the four common local minima conditions are proposed. In addition to the conventional active region, a critical region is defined around each obstacle to further enhance safety.

Using a nonholonomic robot, the EVFF algorithm is compared with five state-of-the-art navigation algorithms in a simulation study that included a walking human with unpredictable direction change, a walking human with constant velocity, and multiple moving obstacles. It is shown to successfully avoid the obstacles and local minima in all cases. An experimental result for avoiding a walking human is also presented. This demonstrates that the algorithm works in the presence of sensor noise and wheel slip. In Fig. 8, we utilized a U-shaped obstacle as an example of a polygonal obstacle. The detailed explanations for computing the VFF and determining the sense of the detour force for polygonal

obstacles were excluded from this paper for brevity. These plus a detailed sensitivity analysis and an analytical method to determine the sizes of active and critical regions will be presented in a future paper. It must be noted that the EVFF algorithm is not intended for use in highly cluttered environments since it requires large active regions to smoothly avoid humans. In a highly cluttered environment this requirement will not be achievable.

Acknowledgement

The funding received from the Natural Sciences and Engineering Research Council of Canada (NSERC) is gratefully acknowledged.

References

1. P. Hart, N. Nilsson and B. Raphael, A formal basis for the heuristic determination of minimum cost paths, *IEEE Trans. Syst. Sci. Cybern.* **4**, 100–107 (1968).
2. B. Lee and C. Lee, Collision-free motion planning of two robots, IEEE Trans. Syst. Man Cybern. **17**, 21–31 (1987).
3. A. Stentz, Optimal and efficient path planning for partially-known environment, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, San Diego, CA, pp. 3310–3317 (1994).
4. R. Simmons, The curvature-velocity method for local obstacle avoidance, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Minneapolis, MN, pp. 3375–3382 (1996).
5. D. Fox, W. Burgard and S. Thrun, The dynamic window approach to collision avoidance, *IEEE Robot. Autom. Mag.* **4**, 23–33 (1997).
6. P. Fiorini and Z. Shiller, Motion planning in dynamic environments using velocity obstacles, *Int. J. Robot. Res.* **17**, 760–772 (1998).
7. S. Lavalle and J. Kuffner, Randomized kinodynamic planning, *Int. J. Robot. Res.* **20**, 378–400 (2001).
8. M. Yamamoto, M. Shimada and A. Mohri, Online navigation of mobile robot under the existence of dynamically moving multiple obstacles, in: *Proc. 4th IEEE Int. Symp. on Assembly and Task Planning*, Fukuoka, Japan, pp. 13–18 (2001).
9. D. Hsu, R. Kindel, J. Latombe and S. Rock, Randomized kinodynamic motion planning with moving obstacles, *Int. J. Robot. Res.* **21**, 233–255 (2002).
10. S. Koenig and M. Likhachev, Improve fast replanning for robot navigation in unknown terrain, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington, DC, pp. 968–975 (2002).
11. T. Fraichard and H. Asama, Inevitable Collision States: a step towards safer roots? *Adv. Robotics* **18**, 1001–1024 (2004).
12. F. Large, C. Laugier and Z. Shiller, Navigation among moving obstacles using the NLVO: principles and applications to intelligent vehicles, *Auton. Robot* **19**, 159–171 (2005).
13. S. Park and B. Lee, Analysis of robot collision characteristics using concept of the collision map, *Robotica* **24**, 295–303 (2006).
14. D. Ferguson and A. Stentz, Using interpolation to improve path planning the field D* algorithm, *J. Field Robot.* **23**, 79–101 (2006).
15. J. Bruce and M. Veloso, Safe multirobot navigation with dynamics constraints, *Proc. IEEE* **94**, 1398–1410 (2006).

16. M. Seder and I. Petrovic, Dynamic window based approach to mobile root motion control in the presence of moving obstacles, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Roma, Italy, pp. 1986–1991 (2007).
17. J. Vannoy and J. Xiao, Real-time adaptive motion planning (RAMP) of mobile manipulators in dynamic environments with unforeseen changes, *IEEE Trans. Robot.* **24**, 1199–1212 (2008).
18. K. Bendjilali and F. Belkhouche, Kinematics-based navigation functions, *Adv. Robot.* **22**, 1243–1264 (2008).
19. J. Minguez and L. Montano, Extending collision avoidance methods to consider the vehicle shape, kinematics, and dynamics of a mobile robot, *IEEE Trans. Robot.* **25**, 367–381 (2009).
20. J. Ota, Rearrangement planning of multiple movable objects by a mobile robot, *Adv. Robot.* **23**, 1–18 (2009).
21. C. Shi, Y. Wang and J. Yang, A local obstacle avoidance method for mobile robots in partially known environment, *Robot. Auton. Syst.* **58**, 425–434 (2010).
22. S. Jacobs, A. Ferrein, S. Schiffer, D. Beck and G. Lakemeyer, Robust collision avoidance in unknown domestic environments, in: *RoboCup 2009: Robot Soccer World Cup XIII*. Springer, Berlin, pp. 116–127 (2010).
23. O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, St. Louis, MO, pp. 500–505 (1985).
24. J. Borenstein and Y. Koren, Real-time obstacle avoidance for fast mobile robots, IEEE Trans. Syst., Man, Cybern. **19**, 1179–1187 (1989).
25. Y. Koren and J. Borenstein, Potential field methods and their inherent limitations for mobile robot navigation, in: *Proc. IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, pp. 1398–1404 (1991).
26. C. Choi and J. Lee, Dynamical path-planning algorithm of a mobile robot: Local minima problem and nonstationary environments, *Mechatronics* **6**, 81–100 (1996).
27. O. Khatib, K. Yokoi, O. Brock, K. Chang and A. Casal, Robots in human environments: basic autonomous capabilities, *Int. J. Robot. Res.* **18**, 684–696 (1999).
28. S. Masoud and A. Masoud, Constrained motion control using vector potential fields, IEEE Trans. Syst. Man Cybern. A Syst. Humans **30**, 251–272 (2000).
29. S. Yang and M. Meng, Neural network approaches to dynamic collision-free trajectory generation, IEEE Trans. Syst. Man Cybern. B Cybern. **31**, 302–318 (2001).
30. S. Ge and Y. Cui, Dynamic motion planning for mobile robots using potential field method, *Auton. Robots* **13**, 207–222 (2002).
31. H. Tanner, S. Loizou and K. Kyriakopoulos Nonholonomic navigation and control of cooperating mobile manipulators, IEEE Trans. Robot. Autom. **19**, 53–64 (2003).
32. D. Kim and S. Shin, Local path planning using a new artificial potential function composition and its analytical design guidelines, *Adv. Robot.* **20**, 115–135 (2006).
33. J. Velagic, B. Lacevic and B. Perunicic, A 3-level autonomous mobile robot navigation system designed by using reasoning/search approaches, *Robot. Auton. Syst.* **54**, 989–1004 (2006).
34. A. Masoud, Decentralized self-organizing potential field-based control for individually motivated mobile agents in a clustered environment: a vector-harmonic potential field approach, IEEE Trans. Syst. Man Cybern. A Syst. Humans **37**, 372–390 (2007).
35. B. Hamner, S. Singh, S. Roth and T. Takahashi, An efficient system for combined route traversal and collision avoidance, *Auton. Robots* **24**, 365–385 (2008).

36. S. Mastellone, D. Stipanović, C. Graunke, K. Intlekofer and M. Spong, Formation control and collision for multi-agent nonholonomic systems: Theory and experiments, *Int. J. Robot. Res.* **27**, 107–126 (2008).
37. S. Loizou and K. Kyriakopoulos, Navigation of multiple kinematically constrained robots, *IEEE Trans. Robot.* **24**(1), 221–231 (2008).
38. F. Fahimi, C. Nataraj and H. Ashrafiuon, Real-time obstacle avoidance for multiple mobile robots, *Robotica* **27**, 189–198 (2009).
39. A. Rosales, G. Scaglia, V. Mut and F. di Sciascio, Trajectory tracking of mobile robots in dynamic environments-a linear algebra approach, *Robotica* **27**, 981–997 (2009).
40. M. Takahashi, T. Suzuki, H. Shitamoto, T. Moriguchi and K. Yoshida, Developing a mobile robot for transport applications in hospital domain, *Robot. Auton. Syst.* **58**, 889–899 (2010).
41. M. Deng, A. Inoue, K. Sekiguchi and L. Jiang, Two-wheeled mobile robot motion control in dynamic environments, *Robot. Comp.-Int. Manuf.* **26**, 268–272 (2010).
42. R. van Emmerik and R. Wagenaar, Effects of walking velocity on relative phase dynamics in trunk in human walking, *J. Biomech.* **29**, 1175–1184 (1996).
43. P. Martin and A. Marsh, Step length and frequency effects on ground reaction forces during walking, *J. Biomech.* **25**, 1237–1239 (1992).
44. L. Zeng, *Design and control of human-friendly robots*, Ph.D. thesis, McMaster University, Canada, 2011.
45. J. Laumond, P. Jacobs, M. Taïx and M. Murray, A motion planner for nonholonomic robots, *IEEE Trans. Robot. Autom.* **10**, 577–593 (1994).
46. G. Klančar and I. Škrjanc, Tracking-error model-based predictive control for mobile robots in real time, *Robot. Auton. Syst.* **55**, 460–469 (2007).

About the Authors

Lingqi Zeng received his B.Eng. degree in Mechanical Engineering from Luoyang University of Technology in 1995, his M.Eng. degree from Xi'an University of Technology in 1998, and his PhD degree from McMaster University in 2011. He works in the System and Process Control group of Hatch Ltd. His research interests include mobile robot navigation and distributed control systems for mining.

Gary M. Bone received his BSc (Ap.Sc.) degree from the Department of Mechanical Engineering, Queen's University, Canada and his M.Eng. and PhD degrees from McMaster University in 1986, 1988, and 1993, respectively. He joined the Faculty of Engineering at McMaster University in 1994, where he is currently a professor in the Department of Mechanical Engineering. His research interests include: design and control of human-friendly robots; path planning and collision avoidance; grasp planning and robotic hands; and servo pneumatic actuators.