Robotica (2005) volume 23, pp. 709–720. © 2005 Cambridge University Press
doi:10.1017/S0263574704001523 Printed in the United Kingdom

A method for robot navigation toward a moving goal with unknown maneuvers

F. Belkhouche* and B. Belkhouche

EECS department, Tulane University, New Orleans, LA 70118 (USA)

(Received in Final Form: December 2, 2004)

SUMMARY

This paper deals with a method for robot navigation towards a moving goal. The goal maneuvers are not a priori known to the robot. Our method is based on the use of the kinematics equations of the robot and the goal combined with geometrical rules. First a kinematics model for the tracking problem is derived and two strategies are suggested for robot navigation, namely the velocity pursuit guidance law and the deviated pursuit guidance law. It turns out that in both cases, the robot's angular velocity is equal to the line of sight angle rate. Important properties of the navigation strategies are discussed and proven. In the presence of obstacles, two navigation modes are used: the tracking mode, which has a global aspect and the obstacle avoidance mode, which has a local aspect. In the obstacle avoidance mode, a polar diagram combining information about obstacles and directions corresponding to the pursuit is constructed. An extensive simulation study is carried out, where the efficiency of both strategies is illustrated for different scenarios.

KEYWORDS: Robot navigation; Target tracking; Moving goal.

I. INTRODUCTION

Robot's navigation is among the most important functions in mobile robotics. Robot's navigation has two aspects, a global aspect in order to obtain a feasible path leading to the goal and a local aspect in order to avoid obstacles. The literature on robot navigation and obstacle avoidance is substantial,[1–4] where various methodologies such as potential fields, vector field histogram, etc. were suggested. Most researchers have focused on solving the motion planning problem in the case of stationary targets. Artificial vision algorithms are among the most used techniques for this purpose. A technique from artificial vision was used by Feyrer and Zell[5,6] for the detection and the tracking of humans. Simulation of the pursuit was also considered,[7] where the robot navigates based on an artificial vision strategy. In general, algorithms based on artificial vision face the following problems

(i) The dynamics and kinematics constraints of the robot are not directly taken into account.
(ii) Because of the limitation of the camera scope, the assumption that the target stays in the camera scope is necessary. Otherwise, the navigation task cannot be accomplished successfully.
(iii) The robot must process in real time a huge amount of data coming from the camera. This requires good computational capabilities.

Fuzzy logic controllers were also used to accomplish the pursuit task. Fuzzy sliding mode control strategy is suggested for the control of the steering angle and the speed,[8] where the target is another mobile robot that moves in a specific path. Fuzzy logic controller for multiple robots[9] tracking a moving object was also considered. The advantage of fuzzy logic controllers is that no comprehensive or precise information on the environment is required. A dynamic potential field method[10] was also suggested. The approach is a generalization of the classical potential field method, and the problem of local minima still exists. Furthermore the dynamic potential field method may be computationally expensive. Adams[11] considered the problem of high speed target pursuit, where it was shown that nonlinear control analysis provides a useful tool for quantifying various path planning parameters in order that stable asymptotic convergence of a mobile robot to its target is guaranteed. Lee et al.[12] suggested to use Lyapunov theory for this purpose, where a global asymptotic stable controller is designed for tracking a moving target.

This paper deals with robot navigation towards a moving goal. The path traveled by the goal is not a priori known to the robot. Thus the control strategy must be elaborated in real-time by taking into account the instantaneous target's motion. Our technique is based on the use of geometric rules combined with the kinematics models of the robot and the moving goal. The tracking problem is considered in both obstacle–free workspace and in the presence of obstacles. Two strategies are used for both cases, and important properties of the method are discussed and proven. In the presence of obstacles a combination of local path planning and global path planning is used, where two navigation modes are incorporated. A polar diagram which provides information about the environment and directions corresponding to the pursuit is constructed. This allows to compute the appropriate steering angle for the robot. This paper is organized as follows: In section II, we formulate the problem. In section III, we describe the model of the robot and the goal. In section IV, we develop the tracking geometry and the kinematics equations. In section V, we derive a tracking model using polar coordinates. In section VI, we

* Corresponding author. E-mail: belkhouf@eecs.tulane.edu

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

discuss the navigation strategies in obstacle-free workspace. In section VII, we address the problem in the presence of obstacles. Finally, in section VIII, we present the results of an extensive simulation.

II. FORMULATION OF THE PROBLEM

The wheeled mobile robot and the moving goal (target) move in an open rectangular region in the Euclidean subspace which corresponds to the horizontal plane. Let $P_G(t)$ denote the position of the goal in the workspace at time $t \geq t_0 \geq 0$, and $P_R(t)$ denotes the position of the robot at time $t \geq t_0 \geq 0$, t_0 is the initial time. $P_R(t)$ and $P_G(t)$ are measured in an inertial frame of coordinates. It is assumed that $P_G(t)$ is a continuous function. The target is moving in the workspace with a given velocity vector and orientation angle. The target's maneuvers are unknown to the robot, which means that off-line strategies are not possible. It is assumed that the robot has a sensory system which allows the detection of the target and provides other information such as the target's position in a reference frame of coordinates. The aim in this paper is to design a navigation law for the robot in order to reach a target which is moving with unknown maneuvers while avoiding obstacles. Reaching the target by the robot means approximately matching the positions of the robot and the moving target at a given time t_f. Mathematically, this can be expressed as follows: $\|P_G(t_f) - P_R(t_f)\| \leq \varepsilon$, ε being a small real number.

III. ROBOT AND GOAL MODEL

The robot is a simple wheeled mobile robot of the unicycle type. The position of the wheeled mobile robot in the Cartesian frame of coordinates is completely specified by the triple $q_R = (x_R, y_R, \theta_R)^T$, where x_R and y_R represent the coordinates of the center of mass of the robot in the Cartesian frame of coordinates and θ_R is the robot orientation angle with respect to the positive x-axis. We have $P_R(t) = (x_R(t), y_R(t))$. The kinematics equations of motion for this type of robot are given by

$$\begin{aligned} \dot{x}_R &= v_R \cos\theta_R \\ \dot{y}_R &= v_R \sin\theta_R \\ \dot{\theta}_R &= w_R \end{aligned} \tag{1}$$

where v_R is the linear velocity and w_R is the angular velocity, v_R and w_R are also the control inputs.

In this paper we consider polar coordinates representation of the kinematics equations of the robot. This allows us to obtain a polar model for the tracking problem. The polar representation of the robot is obtained by considering the following change of variable

$$\begin{aligned} x &= r\cos\lambda \\ y &= r\sin\lambda \end{aligned} \tag{2}$$

where r is the radial coordinate and λ is the angular coordinate as shown in Figure 1 (for the robot and the target).

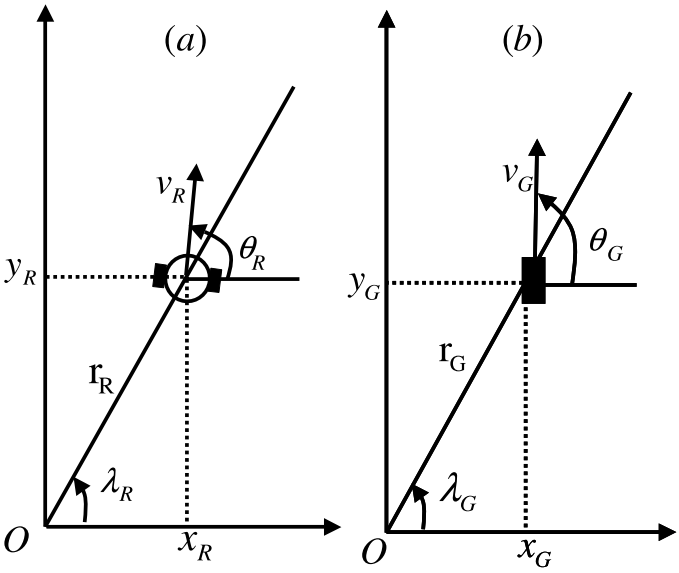


Fig. 1. Cartesian and polar representation of the robot and the target, (a) robot representation, (b) target representation.

By taking the time derivative for r and λ we get

$$\dot{r} = \frac{\dot{x}x + \dot{y}y}{r} \tag{3}$$

and

$$\dot{\lambda} = \frac{\dot{y}x - \dot{x}y}{r^2} \tag{4}$$

Let v_{Rr} and $v_{R\lambda}$ be the components of the robot velocity vector along and across the line OR (v_{Rr} and $v_{R\lambda}$ are the radial and tangential velocities, respectively). The expression for v_{Rr} and $v_{R\lambda}$ can be obtained by considering the kinematics equations (1) and the change of variable (2), which results in the following system

$$\begin{aligned} v_{Rr} &= \dot{r}_R = v_R\cos(\theta_R - \lambda_R) \\ v_{R\lambda} &= r_R\,\dot{\lambda}_R = v_R\sin(\theta_R - \lambda_R) \end{aligned} \tag{5}$$

Equation (5) describes the time variation of r_R and λ_R as a function of the robot linear and angular velocities.

The target is also shown in Figure 1, where x_G and y_G are the coordinates of the target in the Cartesian frame of coordinates. The linear velocity of the target is denoted by v_G. Similarly to the robot, the target velocity vector can be decomposed into radial and tangential components denoted by v_{Gr} and $v_{G\lambda}$, respectively. Their values are given by

$$\begin{aligned} v_{Gr} &= \dot{r}_G = v_G\cos(\theta_G - \lambda_G) \\ v_{G\lambda} &= r_G\,\dot{\lambda}_G = v_G\sin(\theta_G - \lambda_G) \end{aligned} \tag{6}$$

where θ_G is the orientation angle of the moving target, and $P_G(t) = (x_G(t), y_G(t))$ is its position in the Cartesian frame of reference.

The target can perform two types of motions, namely accelerating and non-accelerating motions. For the non-accelerating case, the moving target moves with constant linear velocity and constant orientation angle, i.e., $v_G =$ constant, $\dot{\theta}_G = 0$. In the case of an accelerating target, at

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

least the linear velocity or the orientation angle varies as a function of time.

It is worth noting that polar representation has been used by many authors for controller design for wheeled mobile robots of the unicycle type.[12,13]

IV. GEOMETRY OF THE INTERCEPTION AND KINEMATICS EQUATIONS

Our strategy is based on geometrical rules combined with the kinematics equations of the robot and the moving goal.

The geometry of the tracking problem in the horizontal plane is shown in Figure 2. The wheeled mobile robot is denoted by R and the target by G. The positions of the robot and the goal in an inertial reference frame of coordinates are given by the vectors $\mathbf{r}_R = \mathbf{OR}$ and $\mathbf{r}_G = \mathbf{OG}$, respectively, where O is the origin of the reference frame. Clearly the velocity vectors for the robot and the moving goal can be written as follows

$$\begin{aligned} \mathbf{v}_R &= \dot{\mathbf{r}}_R = v_{Rr}\mathbf{e}_r + v_{R\lambda}\mathbf{e}_\lambda = \dot{x}_R\mathbf{e}_x + \dot{y}_R\mathbf{e}_y \\ \mathbf{v}_G &= \dot{\mathbf{r}}_G = v_{Gr}\mathbf{e}_r + v_{G\lambda}\mathbf{e}_\lambda = \dot{x}_G\mathbf{e}_x + \dot{y}_G\mathbf{e}_y \end{aligned} \tag{7}$$

where $\mathbf{e}_r$ and $\mathbf{e}_\lambda$ are the unit vectors along and across the lines **OR** and **OG** and $\mathbf{e}_x$ and $\mathbf{e}_y$ are the unit vectors in the Cartesian frame of reference. From Figure 2, we define the following geometric quantities

(i) The line of sight $\mathbf{L}_G$ is the straight line that starts at the reference point of the robot and is directed at the target.

(ii) The Euclidean distance robot-goal is denoted by r_{GR}. The projection of the relative distance on the x- and y-axes gives

$$\begin{aligned} x_d &= x_G - x_R \\ y_d &= y_G - y_R \end{aligned} \tag{8}$$

Clearly $r_{GR} = \sqrt{(x_G - x_R)^2 + (y_G - y_R)^2}$.

(iii) The line of sight angle robot-target is the angle from the positive x-axis to the line of sight. This angle is denoted by λ_{GR}, and is given by

$$\tan\lambda_{GR} = \frac{y_d}{x_d} \tag{9}$$

(iv) The angles α_R and α_G are the angles from the line of sight to the velocity vectors of the robot and the goal, respectively. They are called lead angles.

The line of sight angle is not defined for $r_{GR} = 0$, since the positions of the robot and the target coincide when $r_{GR} = 0$.

The aim of the wheeled mobile robot is to reach the moving target. The robot reaches its target when the positions of the target and the robot coincide with a small error, which means that $\|P_G(t_f) - P_R(t_f)\| \leq \varepsilon$, with $t_f < +\infty$. It is important to note that the target maneuvers are not *a priori* known to the robot, which means that the navigation requires a real time strategy. Navigation towards the target is a global navigation problem. In general, global navigation algorithms work off-line; however, the problem of reaching a moving target is a real-time problem. The problem is to find a closed loop control law for the robot that achieves $r_{GR}(t_f) \approx 0$. We assume that

(i) The robot is faster than the target, $v_R > v_G$. In some situations, the robot can reach the moving target even when the target is faster. However, this is not the case in general.

(ii) The robot's minimum turning radius is smaller than the minimum turning radius of the target.

(iii) The robot has a sensory system which allows the detection of the target and the obstacles and provide the information required by the control loop.

Before we discuss the navigation method, we derive a simple kinematics model for the tracking problem. This is discussed in the next section.

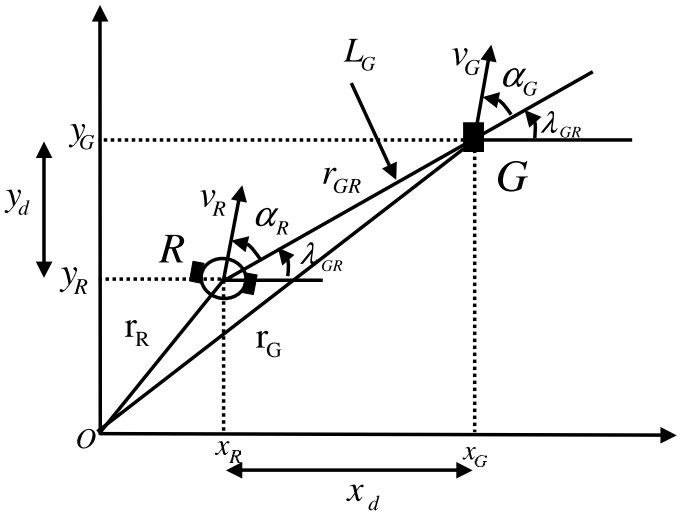


Fig. 2. Geometry of the tracking problem in the horizontal plane.

V. KINEMATICS MODEL OF THE MOTION

Our aim in this section is to derive a kinematics model of the robot and target motion from which the navigation law is derived. Consider the following relative velocity

$$\dot{\mathbf{r}}_{GR} = \dot{\mathbf{r}}_G - \dot{\mathbf{r}}_R \tag{10}$$

This relative velocity can be decomposed into radial and tangential components. By considering the kinematics model for the robot and the target we obtain

$$\begin{aligned} v_r &= \dot{r}_{GR} = v_G\cos\alpha_G - v_R\cos\alpha_R \\ v_\lambda &= r_{GR}\dot{\lambda}_{GR} = v_G\sin\alpha_G - v_R\sin\alpha_R \end{aligned} \tag{11}$$

where α_G and α_R are the lead angles for the goal and the robot, respectively, with

$$\begin{aligned} \alpha_G &= \theta_G - \lambda_{GR} \\ \alpha_R &= \theta_R - \lambda_{GR} \end{aligned} \tag{12}$$

v_r and v_λ are the velocities of the target (goal) seen by the robot, along and across the line of sight $\mathbf{L}_G$. The relative

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

kinematics model in polar coordinates will be used through this paper to model the navigation problem. The kinematics model in equation (11) takes into account the linear and angular velocities of the target and the robot and also the line of sight angle, which represents an important geometric quantity. This model facilitates the derivation and the analysis of the navigation strategy.

VI. NAVIGATION TOWARDS A MOVING GOAL IN THE ABSENCE OF OBSTACLES

The problem of navigation in the absence of obstacles is considered in this section. Two strategies are suggested here, namely the velocity pursuit and the deviated pursuit guidance laws. Both strategies consist of a closed loop system, that is, the navigation law is a function of the states of the target and the robot. Closed loop systems are preferred over open loop systems, since the strategy of the robot must change if the target changes its strategy.

VI.1. Velocity pursuit guidance law

The velocity pursuit guidance law (VPGL) is a guidance law which can be used to reach moving objects. Different variant of the pursuit laws are among the most important guidance laws discussed in the aerospace community,[14,15] and some fields of mathematics.[16–19] The velocity pursuit guidance law is a special case of the proportional navigation, which is the most discussed guidance strategy in the aerospace community. Many predators use similar strategies in order to catch their prey. Other animals use the opposite technique in order to escape to their predators. It is very important to note that the VPGL is different from the algorithm called the pure pursuit by some authors in robotics, which is a tracking algorithm that works by calculating the curvature that will move the robot to its target position. This technique is used to reach stationary points.[20,21] However, the VPGL allows tracking and reaching an unpredictably moving target. To the best of our knowledge, the VPGL has never been used for robotics navigation. The principle of the VPGL is simple; the velocity vector of the robot lies on the line of sight $\mathbf{L}_G$, and as a result, the robot's tangential velocity is zero, i.e., $v_{R\lambda} = 0$. Mathematically, the VPGL is expressed by the following vectorial relation

$$\mathbf{v}_R \times \mathbf{r}_{GR} = \mathbf{0} \tag{13}$$

with the following constraint

$$\mathbf{v}_R \cdot \mathbf{r}_{GR} > 0 \tag{14}$$

Equation (14) states that the velocity of the robot along the line of sight is positive; thus, the robot moves towards the goal and not in the opposite direction in the line of sight. The robot's orientation angle can be deduced easily from equations (13) and (14) as follows

$$\theta_R = \lambda_{GR} \tag{15}$$

The nonlinear function $\lambda_{GR}(t)$ is continuous, since it is assumed that $P_G(t)$ is continuous. By taking the derivative of equation (15) with respect to time, we get the expression of the control strategy in terms of the robot's angular velocity

$$w_R = \dot{\lambda}_{GR} \tag{16}$$

Equation (16) describes the control input for the robot's angular velocity. It also summarizes the principle of the velocity pursuit guidance law applied to robot navigation, where according to the velocity pursuit guidance law, the angular velocity of the robot is equal to the rate of turn of the line of sight angle.

The line of sight angle rate can also be written as a function of the goal's angular velocity as follows

$$\dot{\lambda}_{GR} = \dot{\theta}_G - \dot{\alpha}_G \tag{17}$$

In the case of the velocity pursuit guidance law, the angle α_G is the angle between the velocity vectors of the robot and the goal. The relative velocities of the goal with respect to the robot, along and across the line of sight under the velocity pursuit guidance law are given as follows

$$\begin{aligned} \dot{r}_{GR} &= v_G \cos(\theta_G - \lambda_{GR}) - v_R \\ r_{GR}\,\dot{\lambda}_{GR} &= v_G \sin(\theta_G - \lambda_{GR}) \end{aligned} \tag{18}$$

This system is obtained by considering the kinematics equations model (11) subject to the VPGL strategy. The kinematics equations of the wheeled mobile robot in the Cartesian frame of reference under the velocity pursuit guidance law are as follows

$$\begin{aligned} \dot{x}_R &= v_R \cos\lambda_{GR} \\ \dot{y}_R &= v_R \sin\lambda_{GR} \\ \dot{\phi}_R &= w_R = \dot{\lambda}_{GR} \end{aligned} \tag{19}$$

By replacing $\cos\lambda_{GR}$ and $\sin\lambda_{GR}$ by their values, we get

$$\begin{aligned} \dot{x}_R &= \frac{v_R(x_T - x_R)}{r_{GR}} \\ \dot{y}_R &= \frac{v_R(y_T - y_R)}{r_{GR}} \\ \dot{\phi}_R &= w_R = \dot{\lambda}_{GR} \end{aligned} \tag{20}$$

The kinematics equations of the robot under the VPGL given by (19) and (20) show the simplicity of our approach. Even though the principle of the approach is quite simple, the closed form solution of the kinematics equations is not possible in general, which renders the analysis of the navigation law more difficult. However, rigorous results can be obtained in some particular cases, for example, when the target is moving in a straight line with constant velocity.

In the velocity pursuit guidance law, the mobile robot aims to reach its moving target by nulling the angle α_G between the velocity vectors of the target and the robot. This means that the robot's orientation angle tends to the target's orientation angle. This result is stated as follows

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

Proposition 1: Under the velocity pursuit guidance law, the robot orientation angle tends to the target's orientation angle.

Proof. The proof is simple if we consider the kinematics model derived previously, from which we have the equation for the line of sight angle given by

$$\dot{\lambda}_{GR} = \frac{v_G \sin(\theta_G - \lambda_{GR})}{r_{GR}} \tag{21}$$

This is a scalar nonlinear differential equation with two equilibrium solutions, namely: $\lambda_{GR1}(t) = \theta_G(t)$ and $\lambda_{GR2}(t) = \theta_G(t) + \pi$. The classical linearization[22,23] near these equilibria allows us to obtain
Near $\lambda_{GR1}(t)$

$$a_1 = -\frac{v_G}{r_{GR}} \tag{22}$$

Near $\lambda_{GR2}(t)$

$$a_2 = \frac{v_G}{r_{GR}} \tag{23}$$

According to Hartman and Grobman theorem[22,23] there exists a topological equivalence between the nonlinear system and its linearized systems since a_1 and a_2 are both different from zero; therefore, the linear system and the nonlinear system have locally the same behavior. Since v_G and r_{GR} are positive functions, λ_{GR1} is a stable solution (since a_1 is negative for all values of t) and λ_{GR2} is an unstable solution (since a_2 is positive for all values of t), which means that $\lambda_{GR}(t)$ tracks its stable solution, i.e., $\lambda_{GR}(t) \to \theta_G(t)$, and since the VPGL states that $\lambda_{GR}(t) = \theta_R(t)$, we get $\theta_R(t) \to \theta_G(t)$; thus, in the velocity pursuit guidance law, the robot aims to match the orientation angle of the moving goal. □

The next result concerns the conditions under which the robot reaches its moving goal when navigating under the velocity pursuit guidance law.

Proposition 2: The mobile robot navigating under the velocity pursuit guidance law reaches its moving target $\forall v_G, \forall \theta_G$ when $v_R > v_G$.

Proof. The proof is simple when considering the equation for the relative distance given by the first equation in (18). For $\dot{r}_{GR} < 0$, the distance is decreasing and the robot reaches its target. Clearly, when $v_R > v_G$, we have $v_G \cos(\theta_G - \lambda_{GR}) < v_R$, $\forall \theta_G, \forall \lambda_{GR}, \forall v_G$. Thus, $\dot{r}_{GR} < 0$ for all the possible strategies of the target. □

The interception point, where the robot reaches its target is not known a priori. It depends on various factors such as the initial positions, the target's linear and angular velocities and the robot linear velocity.

Unlike many other navigation methods, the velocity pursuit guidance law allows to determine the time which takes to the robot to reach its target. This can be accomplished using the equation for the relative radial velocity, from which we get

$$r(t_f) = 0 \Rightarrow \int_{t_0}^{t_f} (v_G \cos\alpha_G - v_R)\, dt = r_0 \tag{24}$$

The interception time can be obtained from the integration of (24). In the case where v_G, α_G and v_R are constant, we have

$$t_f = \frac{r_0}{v_R - v_G \cos\alpha_G} \tag{25}$$

Clearly, it is necessary that $v_R > v_G \cos\alpha_G$ in order for t_f to be positive. This is always satisfied when $v_R > v_G$.

VI.2. Navigation using the deviated pursuit guidance law
Our aim in this section is to generalize the method based on the velocity pursuit guidance law. Here, there exists a non-zero angle between the line of sight and the robot's velocity vector. We define the angle α_{R0} as follows

$$\alpha_{R0} = \text{angle}(\mathbf{L}_G, \mathbf{v_R}) \tag{26}$$

where α_{R0} is a constant deviation angle. This means that the robot's velocity vector is deviated from the line of sight with a constant angle. This strategy will be called the deviated pursuit guidance law (DPGL). In the deviated pursuit guidance law, the radial and the tangential (along and across $\mathbf{L}_G$) components of the robot's velocity are given by

$$\begin{aligned} v_{Rr} &= v_R \cos\alpha_{R0} \\ v_{R\lambda} &= v_R \sin\alpha_{R0} \end{aligned} \tag{27}$$

It is easy to see that the VPGL is a special case of the DPGL with $\alpha_{R0} = 0$. The path traveled by the robot under the DPGL is different from the path under the VPGL. Also, different paths are obtained for different deviation angles. This offers a large number of possibilities for the navigation under the DPGL. In the case of the DPGL, the angle between the velocity vectors of the target and the robot is $\alpha_{R0} + \alpha_G$, and thus it is time varying in most cases, since α_G is time varying in general. One advantage of using the DPGL resides in obstacle avoidance, where the deviation angle can be adjusted to choose the robot's path and avoid local obstacles. It is worth noting that there exists a constraint on the deviation angle, where $-\frac{\pi}{2} < \alpha_{R0} < \frac{\pi}{2}$. Otherwise, the problem becomes a deviated escape. The mobile robot orientation angle under the deviated pursuit guidance law varies as follows

$$\theta_R = \lambda_{GR} + \alpha_{R0} \tag{28}$$

The control input for the robot angular velocity is the same as the velocity pursuit guidance law, where

$$w_R = \dot{\theta}_R = \dot{\lambda}_{GR} \tag{29}$$

Thus, the robot angular velocity is equal to the rate of turn of the line of sight angle between the robot and the target.

In a similar way to the case of the VPGL, the velocities of the target point with respect to the robot in the case of the

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

DPGL are described by

$$\begin{aligned} \dot{r}_{GR} &= v_G \cos(\theta_G - \lambda_{GR}) - v_R \cos\alpha_{R0} \\ r_{GR}\,\dot{\lambda}_{GR} &= v_G \sin(\theta_G - \lambda_{GR}) - v_R \sin\alpha_{R0} \end{aligned} \tag{30}$$

The wheeled mobile robot kinematics equations are given by

$$\begin{aligned} \dot{x}_R &= v_R \cos(\alpha_{R0} + \lambda_{GR}) \\ \dot{y}_R &= v_R \sin(\alpha_{R0} + \lambda_{GR}) \\ \dot{\phi}_R &= w_R = \dot{\lambda}_{GR} \end{aligned} \tag{31}$$

The following proposition concerns the conditions on the robot control inputs for which the robot reaches its moving goal.

Proposition 3: The robot navigating under the deviated pursuit guidance law reaches the moving target $\forall v_G$, $\forall \theta_G$, when v_R and α_{R0} satisfy

$$v_R > v_G \tag{32}$$

and

$$\alpha_{R0} \in \left(-\frac{\pi}{2}, +\frac{\pi}{2}\right) \tag{33}$$

Proof. Similarly to the case of the velocity pursuit guidance law, the proof is simple when considering the kinematics equations for the deviated pursuit guidance law given by (30). In this case, the relative distance robot-target is decreasing ($\dot{r}_{GR} < 0$) when

$$v_R \cos\alpha_{R0} > v_G \cos(\theta_G - \lambda_{GR}) \tag{34}$$

This inequality is always satisfied when (32) and (33) are satisfied. □

In the case where the robot and the target are moving with constant velocities, the deviation angle α_{R0} plays a crucial role. As we mentioned previously, for the DPGL, different paths are obtained for different deviation angles. The choice of α_G can increase or reduce the distance and the travel time of the robot before it reaches its target. This will be illustrated in our simulation examples.

VI.3. Navigation using the DPGL towards a fixed point

In the simplest case when the target is stationary, the deviated pursuit allows the robot to reach the target point when the deviation angle is chosen properly. This result is stated as follows:

Proposition 4: When $\alpha_{R0} \in (-\frac{\pi}{2}, +\frac{\pi}{2})$, the robot navigating under the DPGL reaches its fixed target. If v_R is constant then the robot reaches its target point at time $t_f = \frac{r_{GR}(t_0)}{v_R \cos\alpha_{R0}}$, $r_{GR}(t_0)$ is the initial distance robot-target.

Proof. The proof is similar to the general case, the kinematics equations can be written as follows

$$\begin{aligned} \dot{r}_{GR} &= -v_R \cos\alpha_{R0} \\ r_{GR}\,\dot{\lambda}_{GR} &= -v_R \sin\alpha_{R0} \end{aligned} \tag{35}$$

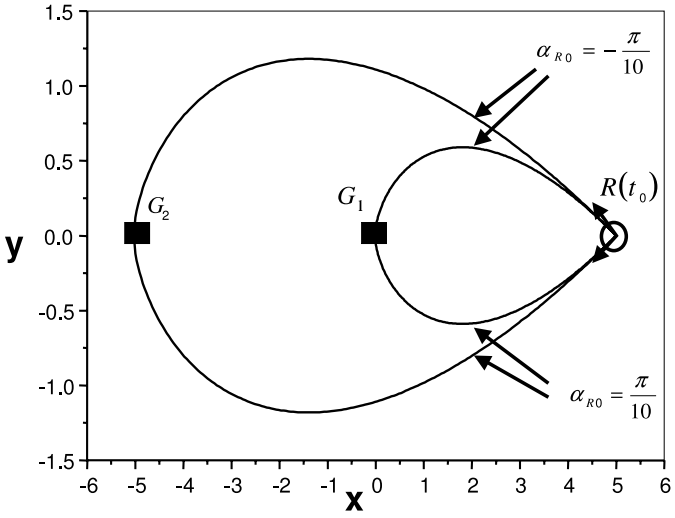


Fig. 3. An illustration of the navigation using the DPGL towards fixed goals G_1 and G_2, with two deviation angles, $\alpha_{R0} = \pm\frac{\pi}{10}$.

The robot reaches the target when $\dot{r}_{GR} < 0$, which corresponds to $\alpha_{R0} \in (-\frac{\pi}{2}, +\frac{\pi}{2})$. When v_R is constant, the solution for the relative range is given by

$$r_{GR}(t) - r_{GR}(t_0) = -(v_R \cos\alpha_{R0})t \tag{36}$$

which gives $t_f = \frac{r_{GR}(t_0)}{v_R \cos\alpha_{R0}}$. □

Unlike the VPGL, the line of sight is not a constant in the case of the DPGL. Furthermore, the time which takes the robot to reach its target in the case of the VPGL is smaller than the DPGL. For two deviation angles α_{R0} and $-\alpha_{R0}$, the paths obtained when the robot navigates towards a stationary point are symmetric with respect to the line of sight. An illustration of the robot's navigation using the DPGL towards a stationary point is shown in Figure 3. The deviation angles are $\alpha_{R0} = \pm\frac{\pi}{10}$. The robot initial position is $P_R(t_0) = (5, 0)$, and the goals coordinates are $G_2(0, 0)$ and $G_1(-5, 0)$. The paths traveled by the robot are shown in Figure 3. As proven previously, the robot reaches its goal in both cases.

VII. NAVIGATION TOWARDS A MOVING OBJECT IN THE PRESENCE OF OBSTACLES

Here, the task of navigation is to plan a path to reach the moving target and modify it as necessary to avoid unexpected obstacles. The problem has two levels of difficulty: first establish a navigation law which allows to reach the moving goal, and second establish an algorithm which allows to avoid unexpected obstacles. The robot moves in two modes: the pursuit mode in order to reach the moving goal and the obstacle avoidance mode.

Before we discuss the problem of navigation towards a moving target and obstacle avoidance using the deviated pursuit, we consider a simple example, where the target is stationary. Consider the configuration given in Figure 4, where the obstacle appears in the line of sight robot-goal.

Let $\mathbf{L_G}$ be the line of sight between the robot and the target, and $\mathbf{L_B}$ the line of sight between the robot to the obstacle. Initially at time t_0, $\mathbf{L_G}$ and $\mathbf{L_B}$ coincide. Clearly because of

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

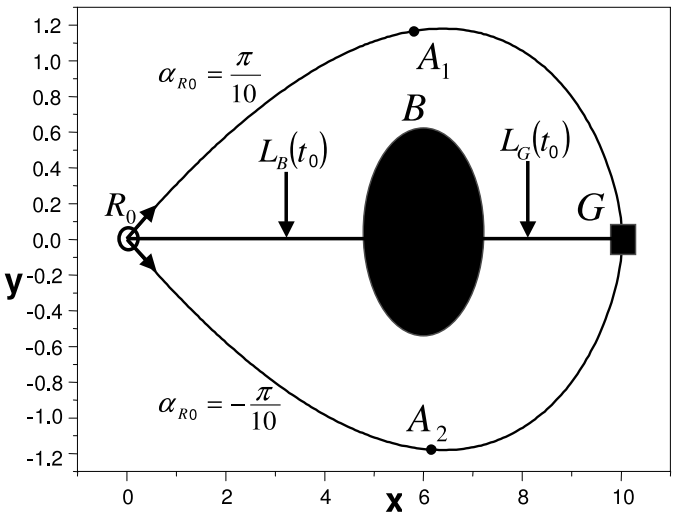


Fig. 4. An illustration of the navigation using the DPGL towards a fixed goal, while avoiding an obstacle (B), which appears in the line of sight.

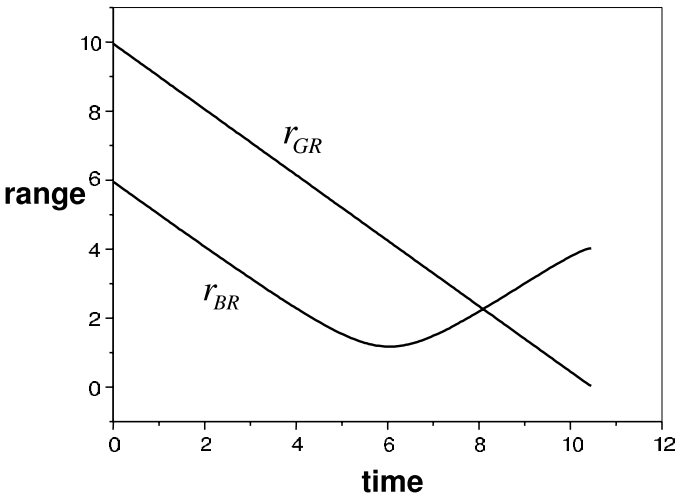


Fig. 5. Evolution of the ranges robot-goal and robot-obstacle for the scenario represented in Figure 4.

the position of the obstacle, the application of the VPGL is not possible. However, the application of the DPGL allows the robot to avoid the obstacle and reach the target. The kinematics equations are as follows

Robot-target

$$\begin{aligned} \dot{r}_{GR} &= -v_R \cos\alpha_{R0} \\ r_{GR}\,\dot{\lambda}_{GR} &= -v_R \sin\alpha_{R0} \end{aligned} \tag{37}$$

Robot-obstacle

$$\begin{aligned} \dot{r}_{BR} &= -v_R \cos(\theta_R - \lambda_{BR}) \\ r_{BR}\,\dot{\lambda}_{BR} &= -v_R \sin(\theta_R - \lambda_{BR}) \end{aligned} \tag{38}$$

Recall that α_{R0} is a constant angle. The robot reaches its target when $\alpha_{R0} \in (-\frac{\pi}{2}, \frac{\pi}{2})$ as proven previously. The lead angle $\alpha_B = \theta_R - \lambda_{BR}$ is measured in terms of the angle of the line of sight robot-obstacle and it is not constant when the robot navigates. Let R_0 be the initial position of the robot, the points A_1 and A_2 are chosen as shown in Figure 4. When the robot navigates between points R_0 and A_1 or R_0 and A_2, the lead angle satisfies $\alpha_B \in (-\frac{\pi}{2}, \frac{\pi}{2})$, thus the robot is approaching the obstacle, after the robot reaches point A_1, (or A_2) we have $\alpha_B \in (\frac{\pi}{2}, \frac{3\pi}{2})$ and the robot is moving away from the obstacle towards the target. This fact is shown in Figure 5, which represents the variation of the relative distances r_{GR} and r_{BR}. Note that points A_1 and A_2 are in the path described by (37) when α_{R0} satisfies $\alpha_{R0} \in (-\frac{\pi}{2}, \frac{\pi}{2})$.

Points A_1 and A_2 move to other positions when α_{R0} varies. In the simulation shown in Figure 4, the robot navigates with $\alpha_{R0} = \pm\frac{\pi}{10}$, with $P_R(t_0) = (0, 0)$, and $P_G = (10, 0)$.

This example shows the possibility of using the deviated pursuit for the purpose of obstacle avoidance, when the obstacle is in the line of sight robot-target. The approach is easily generalized when the target is moving.

The robot moves in two modes: pursuit mode, where the robot navigates using the VPGL or the DPGL and obstacle avoidance mode, where a different deviation angle satisfying the deviated pursuit is used. The obstacle avoidance problem

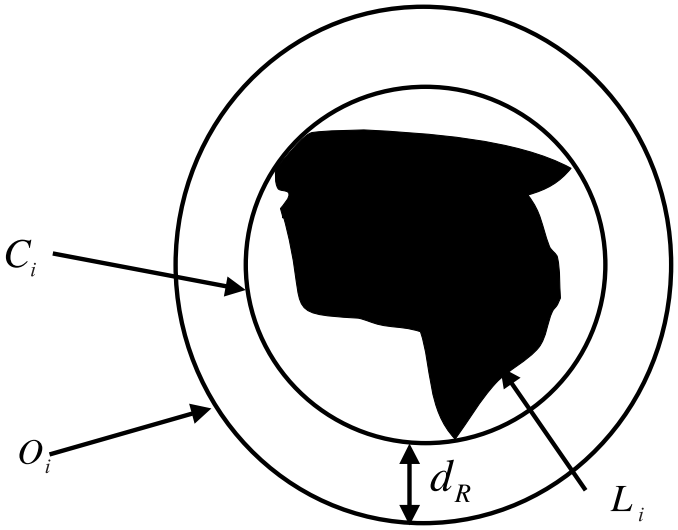


Fig. 6. A representation of the obstacle.

has a local aspect. For this reason only obstacles within a given active region are considered. The active region is defined in terms of some view-range from the robot.

In general dealing with arbitrary shaped obstacles is a difficult task. To address the problem, we approximate obstacles by circles. This techniques is used by many authors.[1,3,24,25] Consider an arbitrary shaped obstacle L_i detected by the robot, a circle C_i enclosing the obstacle is constructed such that $L_i \subset C_i$ as shown in Figure 6. This representation of the obstacles can be useful in avoiding problems posed by some obstacles such as U-shape obstacles. In some situations it is more convenient to approximate the obstacle by an ellipsis instead of a circle. This is the case of obstacles of rectangular shape when the obstacle's length is much greater than its width. In this case it is more convenient to approximate the obstacle by an ellipsis, since the circle will cover more of the free space. To compensate for the width of the robot, obstacle cells given by C_i are enlarged by the robot radius d_R. When the robot is not circular, d_R is defined as the distance from the robot center to its further perimeter point.

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

After this enlargement, the obstacle denoted by O_i has d_{O_i} as radius, with $L_i \subset C_i \subset O_i$. The obstacle O_i is characterized as follows $O_i(x_{Oi}, y_{Oi}, d_{Oi})$, and the robot can be seen as a point-like robot.

Two objects, the robot R and the target G are moving in the two-dimensional Cartesian plane. Consider a number of obstacles denoted by $O_i, i = 1, 2, \ldots, n$. The aim is to design a control strategy for the robot which allows $\|P_R(t_f) - P_G(t_f)\| \leq \varepsilon$, with $P_R(t) \cap P_{O_i} = \emptyset, \forall t \in [t_0, t_f]$, where P_{O_i} is the position of the obstacle O_i, i.e., $P_{O_i} = (x_{O_i}, y_{O_i})$. The approach combines a global navigation method to obtain the path leading to the moving target, and a local navigation method to avoid obstacles. Thus, only obstacles that appear in the robot path during the pursuit mode are considered.

Initially, the robot is controlled to reach its goal using the VPGL or the DPGL; this is the pursuit mode. When an obstacle is encountered, the robot deviates from its initial path in order to avoid the obstacles. In this paper, the obstacle avoidance algorithm consists of three stages. First a simple polar diagram is constructed, which gives a local representation of the robot's environment. Another polar diagram is built to determine the directions for the pursuit and the escape based on geometric rules (position of the target and the line of sight angle). In the third stage an appropriate deviation angle which allows to move around the obstacle is calculated. After the obstacle is passed, the robot returns to the pursuit mode.

a. First stage. In the first stage, a polar representation of the robot's surrounding environment is considered, where a one-dimensional polar diagram is constructed based on the position of the robot and the obstacles in the active region. Consider Figure 7, which shows the robot with two obstacles in the active region. For obstacle O_i, the sensors aboard the robot return the distances d_{1i} and d_{2i}. The angles ρ_{1i} and ρ_{2i} are the limit angles from the positive x-axis to the lines d_{1i} and d_{2i}, respectively. These angles are given by

$$\begin{aligned} \rho_{1i} &= \lambda_{O_i R} - \beta_i \\ \rho_{2i} &= \lambda_{O_i R} + \beta_i \end{aligned} \tag{39}$$

where $\lambda_{O_i R}$ is the line of sight angle between the robot and obstacle O_i, and the angle β_i is given by

$$\tan \beta_i = \frac{d_{O_i}}{r_{O_i R}} \tag{40}$$

where $r_{O_i R}$ is the distance from the robot to the center of the obstacle O_i. The distance that the point robot would travel before hitting the obstacle O_i is denoted by l_{Oi} and is given by

$$l_{Oi} = r_{O_i R} - d_{Oi} \tag{41}$$

Both l_{Oi} and $r_{O_i R}$ lie on the line of sight joining the robot and obstacle O_i. For each obstacle $O_i(x_{O_i}, y_{O_i}, d_{O_i})$, we assume a beam $B_i(\rho_{1i}, \rho_{2i}, l_{Oi})$ where ρ_{1i} and ρ_{2i} are the beam's angle. The aim is to build a polar diagram which stores the beam's angle for all obstacles in the active region. The polar diagram denoted by H_1 is constructed as follows

$$H_1 = \sum_{i=1}^{k} h_i \tag{42}$$

where k is the number of obstacles in the active region, and h_i is given by

$$\begin{aligned} h_i &= 1, \quad \text{if: } \gamma \in [\lambda_{O_i R} - \beta, \lambda_{O_i R} + \beta] \\ h_i &= 0, \quad \text{otherwise} \end{aligned} \tag{43}$$

where γ is the angular variable of the polar diagram. Clearly, from equation (42) the diagram is built by taking into account all obstacles in the active region, which is represented as a rectangle in Figure 7. Free and occupied directions around the momentary position of the robot are stored using the polar diagram H_1. Based on the polar diagram it is easy to find possible-ways-out when obstacles are encountered and navigate the robot safely around the obstacles.

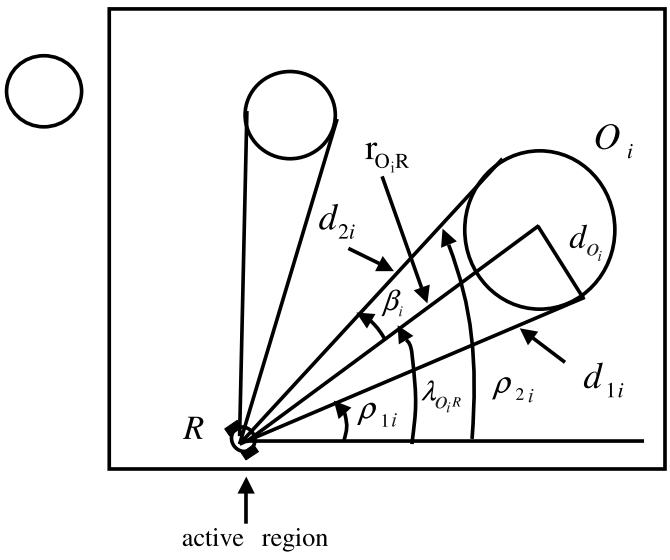


Fig. 7. Representation of the robot-obstacles environment and geometrical quantities.

b. Second stage. Here, the moving target is also considered as shown in Figure 8. Another polar diagram which will be integrated with the previous diagram is constructed. This diagram gives the directions corresponding to the deviated pursuit and the deviated escape. As mentioned previously, values of α_R in the interval $(-\frac{\pi}{2}, +\frac{\pi}{2})$ allows to perform a deviated pursuit. The second diagram is denoted by H_2 and constructed as follows

$$\begin{aligned} H_2 &= 0, \quad \text{if: } \gamma \in \left(\lambda_{GR} - \frac{\pi}{2}, \lambda_{GR} + \frac{\pi}{2}\right) \\ H_2 &= 1, \quad \text{otherwise} \end{aligned} \tag{44}$$

The value 0 is attributed to the case of the pursuit, and 1 to the case of the escape. As it can be seen, the line of sight angle robot-target plays a crucial role in the construction of the second polar diagram.

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

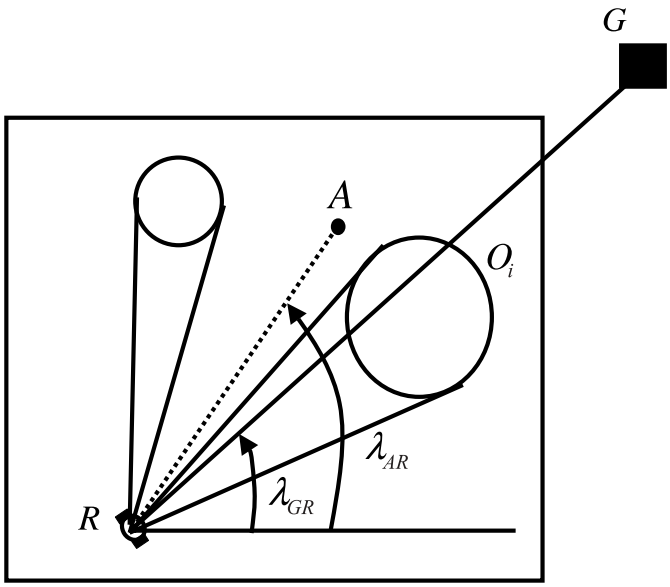


Fig. 8. Representation of the robot-target-obstacles in the horizontal plane.

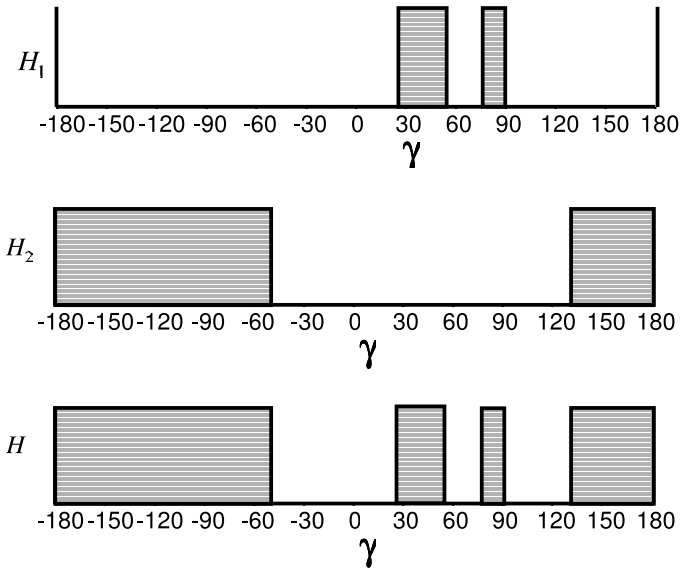


Fig. 9. Polar diagram for the configuration of Figure 8.

c. Third stage. The combination of the two diagrams is used to determine the directions in the unit circle corresponding to: (1) collision-free directions and direction occupied by the obstacles and (2) directions corresponding to the pursuit (velocity or deviated) or escape (velocity or deviated). In the third stage, the orientation angle allowing to put the robot again in a state where the pursuit mode can be applied is calculated. Consider the configuration of Figure 8. Clearly the velocity pursuit cannot be applied, and a deviated pursuit must be used to avoid the obstacle. After finding an obstacle, the robot must deviate from its initial path in order to avoid the obstacle; this can be elaborated as follows:

(i) Decide the detour direction
(ii) Turn around the obstacle
(iii) Once the robot is on the other side of the obstacle, the robot returns to the pursuit mode (VPGL or DPGL).

There are two detour modes: the right detour and the left detour. For the selection of the detour mode, the goal momentary location must be considered, since local information does not in general allow to establish detour criteria. A right detour corresponds to $\theta_R < \rho_{1i}$ and a left detour corresponds to $\theta_R > \rho_{2i}$. In order to decide the detour mode, we consider the following errors in the angles

$$\begin{aligned} \varepsilon_1 &= |\lambda_{GR} - \rho_{1i}| \\ \varepsilon_2 &= |\lambda_{GR} - \rho_{2i}| \end{aligned} \tag{45}$$

ε_1 and ε_2 are the errors between the beam's lower and upper angles and the angle of the line of sight robot-target. There exist several possibilities to chose the detour mode. We suggest that the robot makes

(i) A left detour if $\varepsilon_2 < \varepsilon_1$
(ii) A right detour if $\varepsilon_2 \geq \varepsilon_1$

For the configuration of Figure 8, the detour decision is pending. After the detour decision is made, a point in the free space is chosen. This point is denoted by A in Figure 8. At the initial time when the obstacle avoidance mode is activated, point A is in the line of sight view of the robot.

A final diagram denoted by H is constructed from H_1 and H_2 as follows

$$H = H_1 \cup H_2 \tag{46}$$

In addition to the two diagrams, the line of sight angle robot-target can also be represented as a straight line in H. The diagrams H_1, H_2, and H for the configuration of Figure 8 are shown in Figure 9. Point A must satisfy two conditions

1. It corresponds to a free angle in the diagram H.
2. It satisfies the detour decision.

Now, the aim is calculate the robot's orientation angle by choosing the appropriate value for θ_R to drive the robot to point A. The coordinates of point A are (x_A, y_A), and they are known. The initial distance between the robot and point A is given by

$$r_{AR}(t_0) = \sqrt{(y_A - y_R(t_0))^2 + (x_A - x_R(t_0))^2} \tag{47}$$

Both the VPGL and the DPGL navigation towards a fixed point can be used during this mode. For the VPGL, the deviation angle is $\alpha_{R0} = 0$, or, $\theta_R = \lambda_{AR}$, where $\lambda_{AR} = \lambda_{AR}(t_0)$ is constant when navigating under the VPGL. We have

$$\tan \lambda_{AR}(t_0) = \frac{y_A - y_R(t_0)}{x_A - x_R(t_0)} \tag{48}$$

For the DPGL, the appropriate deviation angle must be calculated in advance. However, we choose a value of θ_R which is close to $\lambda_{AR}(t_0)$, and thus α_{R0} is close to zero.

VIII. SIMULATION

Simulation of the robot navigation towards a moving goal using the VPGL and DPGL is considered in this section

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

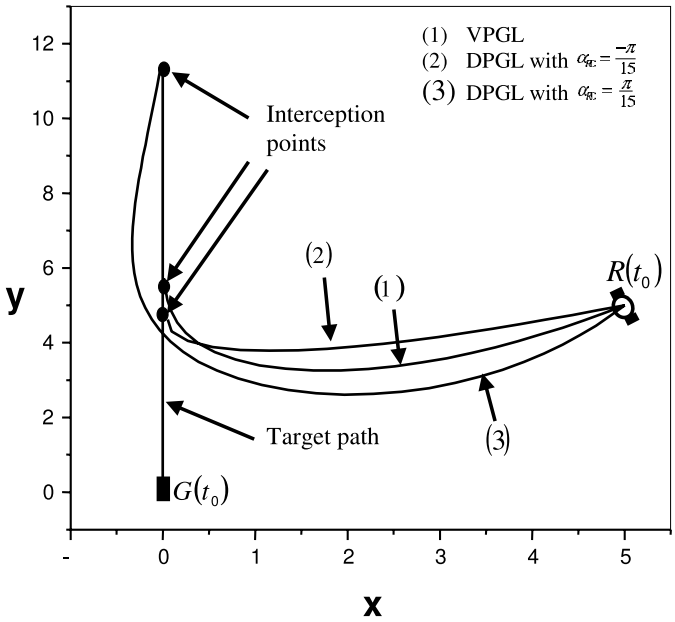


Fig. 10. Robot navigation using the VPGL and the DPGL, the goal moves in a straight line.

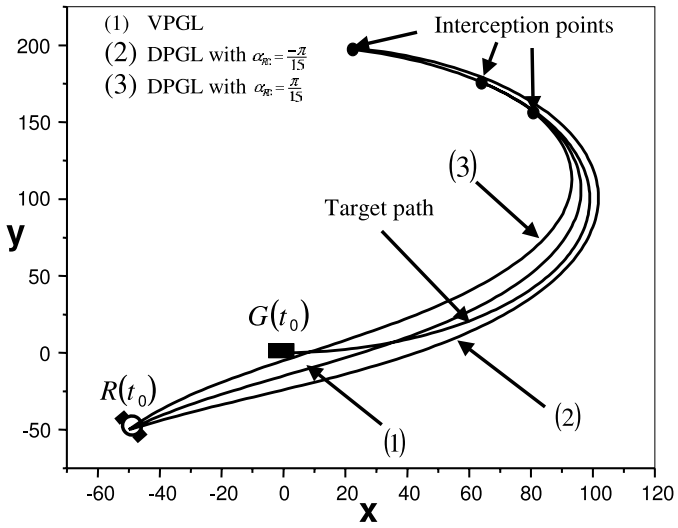


Fig. 11. Robot navigation using the VPGL and the DPGL, with $P_R(t_0) = (-50, -50)$.

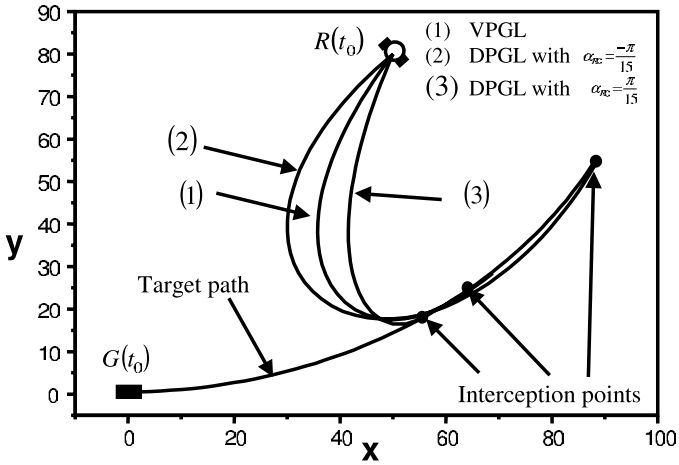


Fig. 12. Robot navigation using the VPGL and the DPGL, with $P_R(t_0) = (50, 50)$.

through four examples. Two examples in the case of obstacle-free workspace are considered, and a comparison between the VPGL and the DPGL with different deviation angles is carried out. In the two other examples, similar problems are considered but in the presence of obstacles. Here, the obstacle avoidance mode is activated.

VIII.1. Scenario 1

In this scenario, the goal is moving in a straight line with a constant linear velocity, $v_G = 1$ (it is assumed that the speed, the distance and the time are without units). This scenario is shown in Figure 10, where the initial position of the target is $P_G(t_0) = (0, 0)$. The initial position of the robot is $P_R(t_0) = (5, 5)$. The robot moves with constant linear velocity $v_R = 1.25$. In the simulation of this scenario, we consider three cases, namely the robot navigating under the VPGL, and the robot navigating under the DPGL for two different deviation angles, $\alpha_{R0} = -\frac{\pi}{15}$ and $\alpha_{R0} = \frac{\pi}{15}$. From simulation, the robot reaches its moving target for all cases. The difference in the path between the DPGL for $\alpha_{R0} = -\frac{\pi}{15}$ and $\alpha_{R0} = \frac{\pi}{15}$ is clear. As shown in Figure 10, the robot navigating under the DPGL with $\alpha_{R0} = -\frac{\pi}{15}$ reaches the moving target much faster than $\alpha_{R0} = \frac{\pi}{15}$. This is due to the fact that when $\alpha_{R0} = -\frac{\pi}{15}$, the robot leads the target. Note that $R(t_0)$ and $G(t_0)$ stand for the initial positions for the robot and the target, respectively.

VIII.2. Scenario 2

We consider navigation using the VPGL and the DPGL, where the target performs the same motion with $v_G = 1$, $P_G(t_0) = (0, 0)$. The robot moves with $v_R = 1.25$. Two initial positions for the robot are considered, namely $P_R(t_0) = (-50, -50)$ and $(50, 50)$ as shown in Figures 11 and 12, respectively. For the DPGL, we choose $\alpha_{R0} = \pm\frac{\pi}{15}$. In Figure 11, the scenario is represented for $P_R(t_0) = (-50, -50)$. The time that took to the robot to reach the moving goal is: for the VPGL, $t_f = 242$, for the DPGL with $\alpha_{R0} = -\frac{\pi}{15}$, $t_f = 288$ and for the DPGL with $\alpha_{R0} = \frac{\pi}{15}$, $t_f = 214$. Unlike the previous case, the robot reaches its target faster when navigating with $\alpha_{R0} = \frac{\pi}{15}$. In this case the robot leads the target for $\alpha_{R0} = \frac{\pi}{15}$. Figure 12 represents a similar scenario with a different initial position for the robot. Clearly, the DPGL and the VPGL allow the robot to reach its moving target from any position when the interception conditions established previously are satisfied.

VIII.3. Scenario 3

The goal is moving in a straight line parallel to the y-axis, and the workspace has two obstacles. Figure 13 shows the path traveled by the robot under the pursuit and the obstacle-avoidance modes. The VPGL is applied from the initial position of the robot to point A. After this point, the obstacle–avoidance mode is activated, and different deviation angles are used. After point C, the VPGL is used again. Figure 14 shows the final diagram H constructed at positions A, B, C and D. Free directions are directions which do not correspond to escape or obstacle directions. For a comparison, the path

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

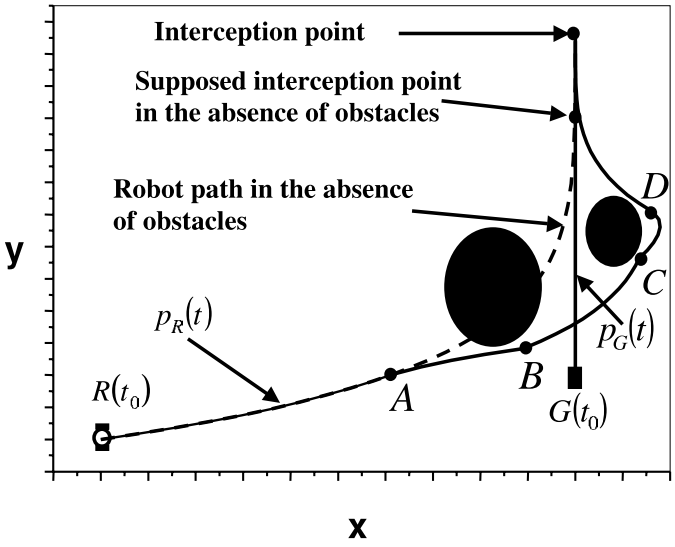


Fig. 13. An illustration of robot navigation under the VPGL and DPGL in order to reach a moving target in the presence of obstacles.

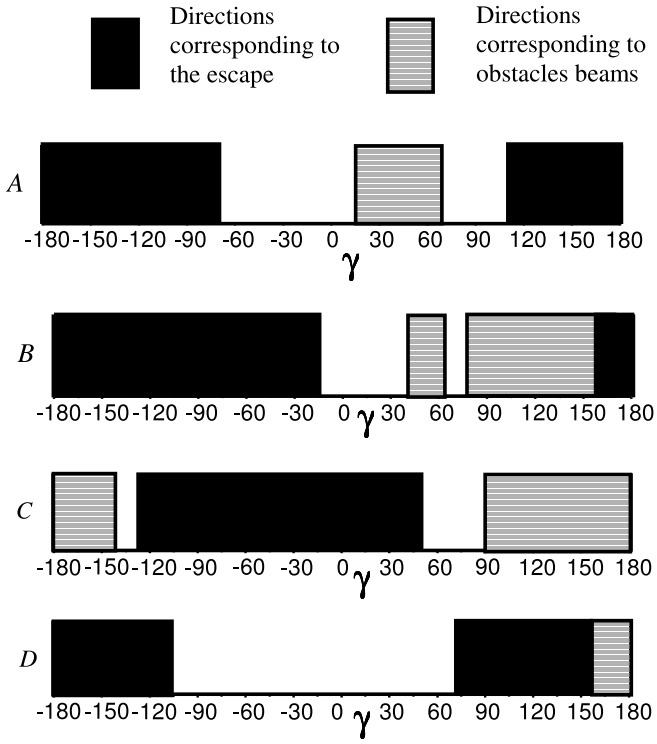


Fig. 14. Polar diagram constructed for the scenario of Figure 13 at different points.

of the robot under the VPGL in the absence of obstacles is also depicted in Figure 13 with a dashed line.

VIII.4. Scenario 4

The last scenario is illustrated in Figure 15, where the paths traveled by the robot and the goal are depicted. Similarly to the previous scenario, pursuit and obstacle–avoidance modes are combined. When the robot navigates, two obstacles appear in its path. A left detour is performed in both cases, and the robot reaches its moving goal successfully.

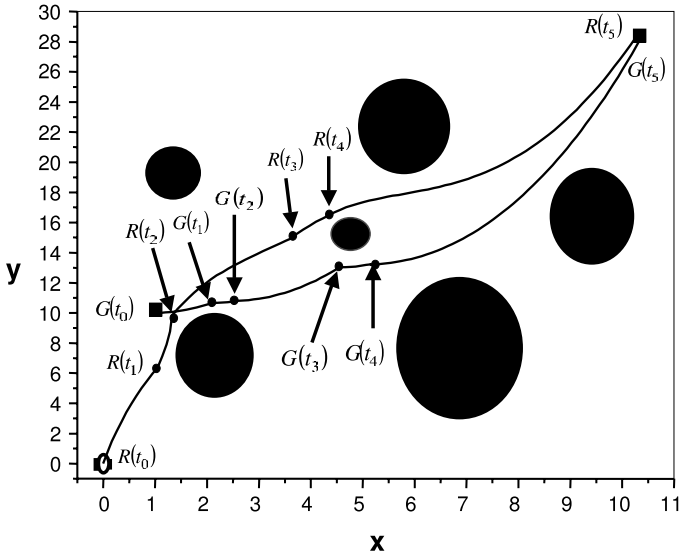


Fig. 15. An illustration of robot navigation under the VPGL and DPGL in order to reach a moving target in the of obstacles.

IX. CONCLUSION

In this paper, we presented a strategy for robot navigation towards a goal moving with unknown maneuvers. Our navigation strategy consists of a closed loop system, where the kinematics models of the robot and the goal combined with gemetrical rules are used. First, a model for the tracking problem based on polar representation of the kinematics equations is derived. Two variants of the control strategy are used, namely the velocity pursuit guidance law and the deviated pursuit guidance law. In both cases, the robot angular velocity is equal to the rate of turn of the line of sight angle between the robot and the goal. The robot kinematics equations under the navigation laws are easily deduced. Important properties of the method are discussed and proven based on the kinematics models. In the presence of obstacles, two navigation modes are combined together, namely, the pursuit mode and the obstacle–avoidance mode. The obstacle–avoidance mode uses polar diagrams to determine the appropriate orientation angle. Various simulation examples are considered to demonstrate the efficiency of the method.

References

1. J. L. Fernandez, R. Sanz, J. A. Benayas and A. R. Dieguez, "Improving collision avoidance for mobile robots in partially known environments: the beam curvature method," *Robotics and Autonomous Systems* **46**, 205–219 (2004).
2. F. Xu, H. Van Brussel, M. Nuttin and R. Moreas, "Concepts for dynamic obstacle avoidance and their extended application in underground navigation," *Robotics and Autonomous Systems* **42**, 1–15 (2003).
3. A. Tsoularis and C. Kambhampati, "On-line planning for collision avoidance on the nominal path," *Journal of Intelligent and Robotic Systems* **21**, 327–371 (1998).
4. J. C. Latombe, *Robot Motion and Planning* (Kluwer academic publishers, Massachusetts, 1991).
5. S. Feyrer and A. Zell, "Detection, tracking and pursuit of humans with an autonomous mobile robot," *Proceedings of IEEE/RSJ international conference on intelligent robots and systems* (1999) pp. 864–869.

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press

6. S. Feyrer and A. Zell, "Tracking and pursuing persons with a mobile robot," *Proceedings of the international workshop on recognition, analyses and tracking faces and gestures in real-time systems* (1999) pp. 83–87.
7. J. Dias, C. Paredes, I. Fonseca, J. Batista and A. Almeida, "Simulating pursuit with machines experiment with robots and artificial vision," *IEEE Transactions on Robotics and Automation* **14**, 1–17 (1998).
8. W. Tong and T. H. S. Li, "Realization of two-dimensional target tracking problem via autonomous mobile robots using fuzzy sliding mode control," *Proceedings of IEEE international conference on industrial electronics* (1998) pp. 1158–1163.
9. I. K. Jeong and J. J. Lee, "Evolving fuzzy logic controllers for multiple mobile robots solving a continuous pursuit problem," *Proceedings of IEEE international conference on fuzzy systems* (1999) pp. 685–690.
10. S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots* **13**, 207–222 (2002).
11. M. D. Adams, "High speed target pursuit and asymptotic stability in mobile robotics," *IEEE Transactions on Robotics and Automation* **15**, 230–236 (1999).
12. S. Lee, Y. Cho, M. Hwang-Bo, B. You and S. Oh, "A stable target-tracking control for unicycle mobile robots," *Proceedings of IEEE/RSJ international conference on intelligent robots and systems* (2000) pp. 1822–1887.
13. M. Aicardi, G. Bicchi and A. Balestrino, "Closed loop steering of unicycle-like vehicles via Lyapunov techniques," *IEEE Robotics and Automation Magazine* 27–35 (1995).
14. M. Guelman, "A qualitative study of the proportional navigation," *IEEE Transactions on Aerospace and Electronic Systems* **7**, 637–643 (1971).
15. C. D. Yang and C. Yang, "A unified approach to proportional navigation," *IEEE Transactions on Aerospace and Electronic Systems* **33**, 557–567 (1997).
16. A. Bruckstein, "Why the ant trails looks so straight and nice," *The Mathematical Intelligencer* **15**, 59–62 (1993).
17. O. Ibidapo-obe, O. S. Asaolu and A. B. Badiru, "Generalized solutions of the pursuit problem in three-dimensional Euclidean space," *Applied Mathematics and Computation* **119**, 35–45 (2001).
18. B. N. Pshenichnyi, "Special pursuit-and-evasion problem with incomplete information," *Cybernetics and Systems Analysis* **31**, 246–251 (1995).
19. A. M. Bruckstein, C. L. Mallows and I. A. Wagner, "Probabilistic pursuits on the grid," (to appear in *The American Mathematical Monthly*).
20. O. Amidi, "Integrated mobile robot control," Master thesis, (Dept. of Electrical and Computer Engineering, Carnegie Mellon 1990).
21. D. H. Shin, "High performance tracking of explicit paths by roadworthy mobile robots," *Doctoral thesis* (Dept. of Civil Engineering, Carnegie Mellon, 1990).
22. S. Wiggins, *Introduction to Applied nonlinear Dynamical Systems and Chaos* (Springer-Verlag, New York, 1990).
23. J. K. Hale and H. Kocak, *Dynamics and Bifurcation* (Springer-Verlag, New York, 1991).
24. R. Simmons, "The curvature-velocity method for local obstacle avoidance," *Proceedings of IEEE international conference on robotics and automation* (1996) pp. 3375–3382.
25. I. Ulrich and J. Borenstein, "VFH +: Reliable obstacle avoidance for fast mobile robots," *Proceedings of IEEE international conference on robotics and automation* (1998) pp. 1572–1577.

https://doi.org/10.1017/S0263574704001523 Published online by Cambridge University Press