Robotica (2007) volume 25, pp. 63–74. © 2006 Cambridge University Press
doi:10.1017/S0263574706002992 Printed in the United Kingdom

Parallel navigation for reaching a moving goal by a mobile robot

F. Belkhouche*, B. Belkhouche and P. Rastgoufard

Department of Electrical Engineering and Computer Science, Tulane University, New Orleans, LA 70118, USA

(Received in Final Form: June 21, 2006, First published online: September 6, 2006)

SUMMARY

In this paper, we present a method for robot navigation toward a moving object with unknown maneuvers. Our strategy is based on the integration of the robot and the target kinematics equations with geometric rules. The tracking problem is modeled in polar coordinates using a two-dimensional system of differential equations. The control law is then derived based on this model. Our approach consists of a rendezvous course, which means that the robot reaches the moving goal without following its path. In the presence of obstacles, two navigation modes are integrated, namely the tracking and the obstacle-avoidance modes. To confirm our theoretical results, the navigation strategy is illustrated using an extensive simulation for different scenarios.

KEYWORDS: Robot navigation; Parallel navigation; Dynamic environment.

1. Introduction

The use of autonomous robots in surveillance and security applications[1–3] has undergone important developments in the last decade. One particular application is tracking a moving object in a given workspace. In most cases, the moving object moves with unknown maneuvers. This application combines different aspects such as tracking, data processing, and navigation toward the target. Methods ranging over visual servoing to Lyapunov theory are used for this purpose. Many authors combine the problem of navigation toward a target with a tracking algorithm. This is the case for most visual servoing methods. There exist mainly two families of methods used for navigation toward a moving object: feature-based and model-based. Feature-based methods track features such as geometric shapes, region of interest, etc.[4] Model-based methods use a model of the moving object. Visual servoing is among the most important feature-based methods.[4] Tracking humans using mobile robots based on vision is discussed by many authors.[5–11] The work of Feyrer and Zell[6,7] is based on vision, where they suggest the use of a multimodal approach combining color, motion, and contour information to accomplish the task. The work by Davis *et al*. is also based on computer vision, where they use a deformable shape model to track humans from a moving platform. Another method for humans tracking using mobile robots is suggested by Shulz *et al*.,[12] where the authors used sample-based joint probabilistic data association filters. The problem of tracking humans with robots is a particular case of the general target tracking problem. This problem is also widely discussed in the literature. Asada and Nakamura[13] suggest a learning method for target reaching while detecting and avoiding collision. Another learning algorithm for target pursuit is suggested by Gaskett *et al*.,[14] where the system learns to perform visual servoing based on rewards relative to tracking performance. Some authors consider the problem of tracking multiple moving objects instead of a single target.[8,15]

Positioning and localization of a robot with respect to the moving object is also discussed in the literature.[16,17] Localization of a robot with respect to the target allows one to design control laws for tracking and navigation toward the target. Simulation of the pursuit of moving objects using a mobile robot is considered by Dias *et al*.,[18] where the solution deals with the interaction of different control systems using visual feedback. Chung and Yang[8] consider the problem of multiple targets tracking using a mobile robot, where a real-time visual feedback law using on-board processors is discussed. Even though vision-based control is widely used, algorithms based on visual servoing may suffer from the following drawbacks:

1. Visual servoing requires high computational capabilities. Most visual perception systems are potentially slow for real-time implementation, especially for fast-moving targets.
2. The dynamics and the kinematics constraints of the robot are not directly taken into account.
3. Camera calibration is necessary, since moving targets must stay within the camera scope.
4. Unstable motion may occur when the goal is highly maneuvering.

Different solutions were suggested to solve these problems. A real-time implementation is addressed by many authors[4,8,19] using different approaches. Reduction of visual data for the real-time implementation presents another potential solution.[9]

Vision sensors are not the only sensors used for tracking. Many authors use different types of sensors such as accoustic sensors,[20] directional sensors,[21] ladar-based sensors,[22] and ultrasonic sensors.[23] These sensors may simplify the equipment to accomplish the tracking task; however, vision sensors allow us to obtain richer data about the target. A comparison between different approaches for human tracking such as GPS tracking and laser tracking is discussed in ref. [11], where the drawbacks and the advantages of the method are discussed in some detail.

* Corresponding author. E-mail: fbelkhou@tulane.edu

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

Maintaining the target in the field of view of the robot's sensory system is another widely discussed topic,[16,25–30] especially in surveillance applications. Different types of sensors are used by the robot, mainly the vision sensors.[26,27] This problem combines motion planning for the robot, i.e., sensor placement, and camera calibration for the vision sensors.

Tracking and navigation toward a moving goal is a difficult problem compared to navigation toward a stationary point. In the case of a moving goal, the navigation problem is a real-time problem and offline strategies are not effective.

The potential field method is also used for robot navigation toward a moving goal,[31] where a new potential function is defined. The problem of local minima is also discussed. An integration of the artificial potential field method with the Lyapunov theory for high-speed target pursuit is considered by Adams.[32] A strategy based on the Lyapunov theory[33] in order to design a stable target tracking law for a unicycle mobile robot is suggested by Lee *et al*. Potential field methods may suffer from the local minima problem. This problem may appear more frequently in the case of a moving goal.

Different fuzzy logic approaches are combined with various control strategies such as visual control. Various strategies such as tracking by the Grey prediction theory integrated with look-ahead fuzzy logic controller,[34] tracking using hierarchical Grey fuzzy motion decision-making method,[35] and fuzzy sliding mode control,[36] were suggested. The fuzzy logic controller is also used for tracking multiple targets.[37] Fuzzy logic may simplify the sensory system because it does not require precise information on the target.

In many situations, the target performs evasive maneuvers. In this case, most authors[38–41] suggest the use of methods from the game theory. Pursuit–evasion problems are considered in different environments such as planar,[38] polygonal,[39] and curved.[40] This is due to the fact that the action of the robot depends on the environment. To simplify the problem, many authors consider it in the absence of obstacles. There exist two approaches for the representation of the pursuit–evasion problem: continuous representation and discrete representation. In the discrete representation, the problem is represented in a grid. Both probabilistic and deterministic methods are used. Hunting behavior of a moving target using a mobile robot or a group of mobile robots is a related problem. This problem is considered by Yamaguchi,[42,43] where a smooth time-varying control law is used. Most tracking algorithms are designed for wheeled mobile robots. However, target tracking by underwater robots[44] or air vehicles[45] is also discussed.

In this paper, we address the problem of robot navigation toward a goal moving with unknown maneuvers, and suggest a solution to it. The navigation problem is considered in both the absence and the presence of obstacles. Clearly, the problem is more difficult in the presence of obstacles. Two navigation modes are integrated as follows:

1. *Navigation toward the target*: The aim is to design a control strategy for the robot in order to reach the moving target. In this mode, path planning has a global aspect.
2. *Obstacle-avoidance mode*: The objective is to avoid local obstacles and put the robot in a position where the tracking mode can be activated again. Here, path planning has a local aspect where different techniques can be used.

In the navigation toward the goal mode, we use a strategy based on the integration of the robot and the target kinematics equations with geometric rules. The aim of the guidance strategy is to put the robot in a rendezvous course with the target. This paper is organized as follows: In Section 2, we formulate the problem. In Section 3, we give the geometric representation of the navigation problem. In Section 4, we derive the kinematics models of the robot and the goal in polar coordinates. In Section 5, we derive a relative kinematics model, which models the navigation problem in polar coordinates. In Section 6, we introduce our control strategy, and prove the main result. In Section 7, we generalize the navigation problem to the case where obstacles are present. Finally in Section 9, we give an extensive simulation study with various scenarios.

2. Problem Formulation

The workspace W consists of a subset of $\mathbb{R}^2$. Let point O be the origin of the world coordinates system. The moving goal (or target) moves in the workspace with maneuvers unknown to the robot. The path of the goal is denoted by $P_{\mathrm{T}}(t) = [x_{\mathrm{T}}(t), y_{\mathrm{T}}(t)]$, where $(x_{\mathrm{T}}, y_{\mathrm{T}})$ are the goal Cartesian coordinates in the world coordinates system. In a similar way, the path of the robot is denoted by $P_{\mathrm{R}}(t) = [x_{\mathrm{R}}(t), y_{\mathrm{R}}(t)]$, where $(x_{\mathrm{R}}, y_{\mathrm{R}})$ are the robot Cartesian coordinates in the world coordinates system. The aim is to design a closed-loop control law for the robot steering angle that would guarantee reaching the moving goal. This can be expressed as $P_{\mathrm{R}}(t_f) \simeq P_{\mathrm{T}}(t_f)$, where t_f is the interception time. We assume that the following conditions are satisfied.

- *H1*: The path of the moving object is smooth, and thus, does not present sharp jumps.
- *H2*: The robot is faster than the moving goal. Here, it is assumed that the robot is a unicycle one.
- *H3*: The minimum turning radius of the robot is smaller than the minimum turning radius of the moving object.
- *H4*: The robot has a sensory system that allows the detection of the obstacles and provides the necessary information to the robot about the moving goal and the obstacles. The influence of the sensory system on the tracking problem is beyond the scope of this paper.

In this paper, we model the wheeled mobile robot by the kinematics equations of a unicycle robot. It is important to note that the method is not restricted to unicycle robots, but it works for other types of mobile robots as well. We chose the unicycle model for its simplicity. Since the target's maneuvers are not *a priori* known to the robot, the path-planning strategy must be elaborated in real-time.

There exist two types of motions that can be accomplished by the moving goal: accelerating and non-accelerating. An accelerating goal moves with a time-varying speed or a time-varying orientation angle. A non-accelerating goal moves with a constant speed and a constant orientation angle. Clearly, navigation toward an accelerating goal is a more difficult and challenging problem.

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

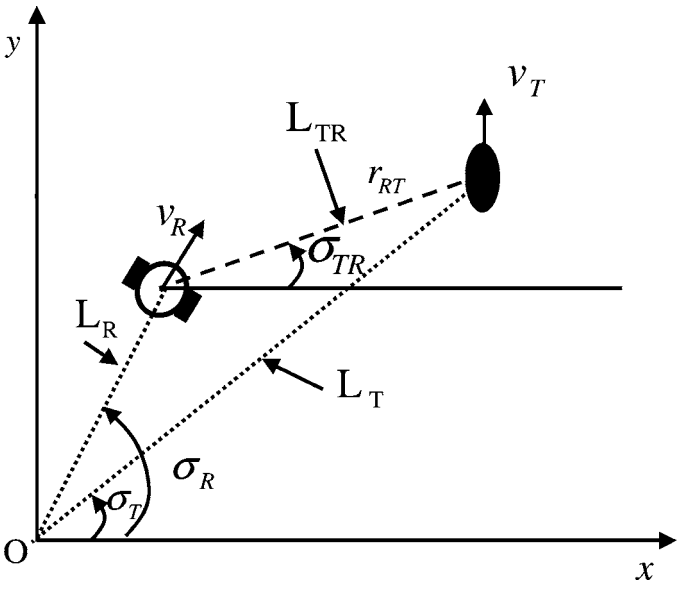


Fig. 1. An illustration of the geometry of the navigation problem.

3. Geometric Representation of the Navigation Problem

Our aim in this section is to introduce the geometric representation of the navigation problem. The robot and its moving target are shown in Fig. 1 (the robot is denoted by R and the moving goal or target by T). We define the following quantities:

1. The robot's line of sight $\mathbf{L}_{\mathrm{R}}$ is the imaginary straight line starting from the origin of the reference frame and directed toward the robot's reference point. The target's line of sight $\mathbf{L}_{\mathrm{T}}$ is the imaginary straight line starting from the origin of the reference frame and directed toward the target.
2. σ_{R} and σ_{T} are the line of sight angles of the robot and the target, respectively. They represent the angles from the reference line (parallel to the x-axis) to the lines of sight of $\mathbf{L}_{\mathrm{R}}$ and $\mathbf{L}_{\mathrm{T}}$, respectively.
3. The line of sight of robot–target is the imaginary straight line that starts at the robot's reference point and is directed toward the target. This line is denoted by $\mathbf{L}_{\mathrm{TR}}$.
4. The line-of-sight angle is the angle between the reference line (parallel to the x-axis) and the line of sight $\mathbf{L}_{\mathrm{TR}}$. This angle is denoted by σ_{TR}.
5. The relative distance of robot–target is denoted by r_{TR}, and is given by

$$r_{\mathrm{TR}} = \sqrt{(x_{\mathrm{T}} - x_{\mathrm{R}})^2 + (y_{\mathrm{T}} - y_{\mathrm{R}})^2}.$$

The robot reaches its moving target when $r_{\mathrm{TR}}(t_f) \approx 0$, with $t_f < +\infty$, where t_f is the interception time. The line-of-sight angle σ_{TR} is expressed in terms of the robot and the target coordinates as follows:

$$\tan \sigma_{\mathrm{TR}} = \frac{y_{\mathrm{T}} - y_{\mathrm{R}}}{x_{\mathrm{T}} - x_{\mathrm{R}}}.$$

Note that σ_{TR} and $\dot{\sigma}_{\mathrm{TR}}$ are not defined when the positions of the robot and the target match. In the next section, we discuss the kinematics models for the robot and the moving goal.

4. Modeling the Robot and the Goal

The robot is a simple wheeled mobile robot of the unicycle type. The kinematics equations of this type of robots is given by

$$\begin{aligned} \dot{x}_{\mathrm{R}} &= v_{\mathrm{R}} \cos \theta_{\mathrm{R}} \\ \dot{y}_{\mathrm{R}} &= v_{\mathrm{R}} \sin \theta_{\mathrm{R}} \\ \dot{\theta}_{\mathrm{R}} &= \omega_{\mathrm{R}} \end{aligned} \tag{1}$$

where $(x_{\mathrm{R}}, y_{\mathrm{R}})$ denote the robot coordinates in the Cartesian frame of reference, θ_{R} is the robot orientation angle with respect to the reference line that is parallel to the x-axis, v_{R} and ω_{R} are the robot control inputs representing the linear and angular velocities of the robot. The target is moving in the Cartesian frame of reference according to the following kinematics equations:

$$\begin{aligned} \dot{x}_{\mathrm{T}} &= v_{\mathrm{T}} \cos \theta_{\mathrm{T}} \\ \dot{y}_{\mathrm{T}} &= v_{\mathrm{T}} \sin \theta_{\mathrm{T}} \end{aligned} \tag{2}$$

where $(x_{\mathrm{T}}, y_{\mathrm{T}})$ denote the target's coordinates in the Cartesian frame of reference, θ_{T} is the orientation angle of the target with respect to the positive x-axis, and v_{T} is the target linear velocity. In this paper, we use polar coordinates representation of the kinematics equations to model the tracking problem. Polar coordinates were used by many authors for controlling mobile robots of the unicycle type.[24,33] In order to derive the equivalent kinematics models in polar coordinates, we use the following change of variable:

$$\begin{aligned} x &= r \cos \sigma \\ y &= r \sin \sigma \end{aligned} \tag{3}$$

where r is the radial variable and σ is the angular variable. By taking the time derivative of r and σ, we get, respectively

$$\dot{r} = \frac{\dot{x}x + \dot{y}y}{r} \tag{4}$$

and

$$\dot{\sigma} = \frac{\dot{y}x - \dot{x}y}{r^2}. \tag{5}$$

The derivation of these equations can be done as follows. We already know that

$$r^2 = x^2 + y^2 \tag{6}$$

and

$$\tan \sigma = \frac{y}{x}. \tag{7}$$

By taking the derivative with respect to time in Eq. (6), we obtain

$$2r\dot{r} = 2x\dot{x} + 2y\dot{y}. \tag{8}$$

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

Similarly by taking the derivative of Eq. (7) with respect to time, we get

$$\dot{\sigma}(1+\tan^2\sigma_{\mathrm{TR}}) = \frac{x\dot{y}-\dot{x}y}{x^2} \tag{9}$$

$$\dot{\sigma}\left[1+\left(\frac{y}{x}\right)^2\right] = \frac{x\dot{y}-\dot{x}y}{x^2} \tag{10}$$

$$\dot{\sigma}r^2 = x\dot{y}-\dot{x}y \tag{11}$$

which gives

$$\dot{r} = \frac{x\dot{x}+y\dot{y}}{r} \tag{12}$$

and

$$\dot{\sigma} = \frac{x\dot{y}-\dot{x}y}{r^2}. \tag{13}$$

Using the robot kinematics equations given in Eq. (1) and the change of variable given in Eq. (3), we obtain the robot kinematics equations in polar coordinates

$$\begin{aligned} v_{\mathrm{R}\parallel} &= \dot{r}_{\mathrm{R}} = v_{\mathrm{R}}\cos(\theta_{\mathrm{R}}-\sigma_{\mathrm{R}}) \\ v_{\mathrm{R}\perp} &= r_{\mathrm{R}}\dot{\sigma}_{\mathrm{R}} = v_{\mathrm{R}}\sin(\theta_{\mathrm{R}}-\sigma_{\mathrm{R}}) \end{aligned} \tag{14}$$

where $v_{\mathrm{R}\parallel}$ and $v_{\mathrm{R}\perp}$ are the radial and tangential velocities of the robot, respectively. They represent the robot velocity components along and across the line of sight $\mathbf{L}_{\mathrm{R}}$. Similarly, we get the following for the moving goal:

$$\begin{aligned} v_{\mathrm{T}\parallel} &= \dot{r}_{\mathrm{T}} = v_{\mathrm{T}}\cos(\theta_{\mathrm{T}}-\sigma_{\mathrm{T}}) \\ v_{\mathrm{T}\perp} &= r_{\mathrm{T}}\dot{\sigma}_{\mathrm{T}} = v_{\mathrm{T}}\sin(\theta_{\mathrm{T}}-\sigma_{\mathrm{T}}) \end{aligned} \tag{15}$$

where $v_{\mathrm{T}\parallel}$ and $v_{\mathrm{T}\perp}$ are the radial and tangential velocities of the target, respectively. They represent the target velocity components along and across the line of sight $\mathbf{L}_{\mathrm{T}}$. By introducing the change of variable given in Eq. (3), the control input becomes $(v_{\mathrm{R}}, \theta_{\mathrm{R}})$ instead of $(v_{\mathrm{R}}, \omega_{\mathrm{R}})$. Our control strategy is based on the use of the kinematics equations in polar coordinates. In the next section, we derive a relative kinematics model that integrates the motion of the robot and the target, and allows to model the tracking problem.

5. Relative Kinematics Model for the Navigation Problem

Let us consider the following relative velocity:

$$\dot{\mathbf{r}}_{\mathrm{TR}} = \dot{\mathbf{r}}_{\mathrm{T}} - \dot{\mathbf{r}}_{\mathrm{R}} \tag{16}$$

which represents the velocity of the moving target as seen by the robot. This relative velocity can be broken down into two components, along and across the line of sight $\mathbf{L}_{\mathrm{TR}}$. By replacing $\dot{\mathbf{r}}_{\mathrm{R}}$ and $\dot{\mathbf{r}}_{\mathrm{T}}$ by their values along and across the line of sight $\mathbf{L}_{\mathrm{TR}}$, we get

$$\begin{aligned} v_{\parallel} &= \dot{r}_{\mathrm{TR}} = v_{\mathrm{T}}\cos(\theta_{\mathrm{T}}-\sigma_{\mathrm{TR}}) - v_{\mathrm{R}}\cos(\theta_{\mathrm{R}}-\sigma_{\mathrm{TR}}) \\ v_{\perp} &= r_{\mathrm{TR}}\dot{\sigma}_{\mathrm{TR}} = v_{\mathrm{T}}\sin(\theta_{\mathrm{T}}-\sigma_{\mathrm{TR}}) - v_{\mathrm{R}}\sin(\theta_{\mathrm{R}}-\sigma_{\mathrm{TR}}). \end{aligned} \tag{17}$$

This system gives the velocity of the goal seen by the robot, along and across the line of sight $\mathbf{L}_{\mathrm{TR}}$. $v_{\parallel}$ gives the rate of change of the relative range between the robot and the goal, and $v_{\perp}$ gives the rate of turn of the goal with respect to the robot. The kinematics model given by Eq. (17) takes into account the velocities and the orientation angles of the robot and the target, and also the line of sight angle, which is a geometric quantity. An equivalent model in the Cartesian coordinates can be written as follows:

$$\begin{aligned} \dot{x}_d &= v_{\mathrm{T}}\cos\theta_{\mathrm{T}} - v_{\mathrm{R}}\cos\theta_{\mathrm{R}} \\ \dot{y}_d &= v_{\mathrm{T}}\sin\theta_{\mathrm{T}} - v_{\mathrm{R}}\sin\theta_{\mathrm{R}} \end{aligned} \tag{18}$$

where $x_d = x_{\mathrm{T}} - x_{\mathrm{R}}$ and $y_d = y_{\mathrm{T}} - y_{\mathrm{R}}$. This system gives the relative velocity of the goal seen by the robot in the Cartesian coordinates.

6. Navigation in the Absence of Obstacles: Parallel Navigation Guidance Strategy

Navigation toward a moving goal can be established in the following two different ways:

1. *Pursuit:* In this case, the robot follows the path of the target, where the velocity vector of the robot is always directed toward the goal.
2. *Rendezvous course:* In this case, the robot does not track the path of the goal, but it computes a point ahead of the goal, where both the robot and the target will arrive at the same time.

Our strategy is based on the parallel navigation guidance law, which uses a rendezvous course. This guidance strategy integrates the kinematics equations of the robot and the target with geometric rules.[49,50] The aim of the parallel navigation is to put the robot in a rendezvous course with the goal. Thus, the robot reaches the goal without following the path traveled by the goal. This can be achieved by controlling the motion of the robot such that the angle of the line of sight of robot–target is constant, i.e.,

$$\sigma_{\mathrm{TR}} = \text{constant}. \tag{19}$$

Thus, the robot moves in lines that are parallel to the initial line of sight. Since the line-of-sight angle is constant, we have the following for the line-of-sight angle rate

$$\dot{\sigma}_{\mathrm{TR}} = 0. \tag{20}$$

From the second expression in Eq. (17), we get

$$v_{\mathrm{R}}\sin(\theta_{\mathrm{R}}-\sigma_{\mathrm{TR}}) = v_{\mathrm{T}}\sin(\theta_{\mathrm{T}}-\sigma_{\mathrm{TR}}). \tag{21}$$

This equation gives the relationship between $(v_{\mathrm{R}}, \theta_{\mathrm{R}})$ and $(v_{\mathrm{T}}, \theta_{\mathrm{T}})$ so that the robot is in a rendezvous course with its moving goal. Let us put $k = v_{\mathrm{R}}/v_{\mathrm{T}}$, where k is the velocity ratio. Under assumption H2, we have $k > 1$. From Eq. (21), we obtain the following for the robot steering angle

$$\theta_{\mathrm{R}} = \sigma_{\mathrm{TR}} + \sin^{-1}\left[\frac{1}{k}\sin(\theta_{\mathrm{T}}-\sigma_{\mathrm{TR}})\right]. \tag{22}$$

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

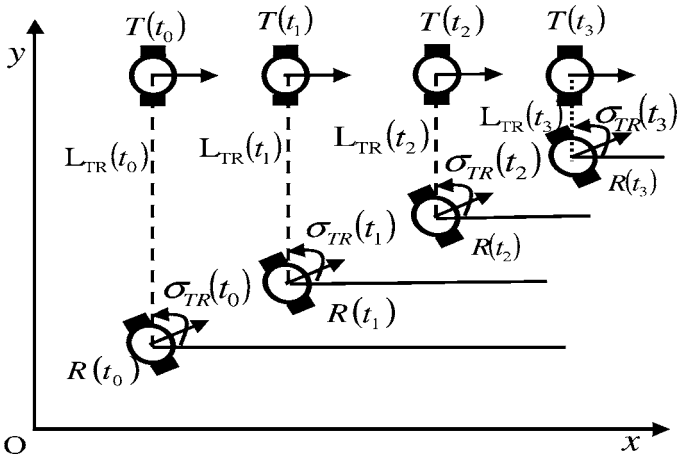


Fig. 2. An illustration of parallel navigation.

From Eq. (22), the robot steering angle is a function of the goal's maneuvers (v_T, θ_T), the robot linear velocity, and the line-of-sight angle between the robot and the goal. An illustration of the parallel navigation is shown in Fig. 2. The initial line-of-sight is $\mathbf{L}_{TR}(t_0)$, and the initial line-of-sight angle is $\sigma_{TR}(t_0) = 90^\circ$. The robot moves in lines $\mathbf{L}_{TR}(t_1)$, $\mathbf{L}_{TR}(t_2)$, etc. that are parallel to $\mathbf{L}_{TR}(t_0)$ by keeping σ_{TR} constant and equal to $\sigma_{TR}(t_0)$, as illustrated in Fig. 2. The relative kinematics equations under parallel navigation are given by

$$\begin{aligned}
\dot{r}_{TR} &= v_{T\|} - v_{R\|} \\
\dot{r}_{TR} &= v_T \cos(\theta_T - \sigma_{TR}) - v_R \cos\left[\sin^{-1}\left(\tfrac{1}{k}\sin(\theta_T - \sigma_{TR})\right)\right] \\
\dot{\sigma}_{TR} &= 0.
\end{aligned} \tag{23}$$

The kinematics equations of the robot under parallel navigation are given by

$$\begin{aligned}
\dot{x}_R &= v_R \cos\left[\sigma_{TR} + \sin^{-1}\left(\tfrac{1}{k}\sin(\theta_T - \sigma_{TR})\right)\right] \\
\dot{y}_R &= v_R \sin\theta_R\left[\sigma_{TR} + \sin^{-1}\left(\tfrac{1}{k}\sin(\theta_T - \sigma_{TR})\right)\right] \\
\dot{\theta}_R &= \omega_R.
\end{aligned} \tag{24}$$

The parallel navigation can be also expressed in the Cartesian coordinates. Noting that the line-of-sight angle is constant under parallel navigation, it is possible to write

$$\frac{y_d}{x_d} = \tan\sigma_{TR} = \text{constant}. \tag{25}$$

If we put $N = \tan\sigma_{TR} = \tan\sigma_{TR}(t_0)$, we can write

$$y_d = N x_d. \tag{26}$$

This means that the projection of the relative distance r_{TR} on the y-axis is proportional to its projection on the x-axis. The proportionality factor is simply the tangent function of the initial value of the robot–target line-of-sight angle. The relative range of robot–target can be written as

$$r_{TR} = x_d\sqrt{1+N^2} = y_d\sqrt{1+\frac{1}{N^2}} \tag{27}$$

and the relative range rate varies as follows:

$$\dot{r}_{TR} = \dot{x}_d\sqrt{1+N^2} \tag{28}$$

$$= \dot{y}_d\sqrt{1+\frac{1}{N^2}}. \tag{29}$$

The following result states that the robot navigating under the parallel navigation law reaches the goal when the previous assumptions are satisfied.

Proposition *Under the control law* (22) *for the robot steering angle, and the assumptions stated earlier, the robot reaches the moving goal successfully.*

Proof. In order to prove that the robot reaches the goal, we proceed by proving that $\dot{r}_{TR} < 0$, and thus, the relative range is a decreasing function of time. The proof is based on the following remarks:

1. Under assumption H2, we have $k > 1$. The inverse function of the sine function maps the domain $[-1, 1]$ into $[-\frac{\pi}{2}, \frac{\pi}{2}]$, and since $k > 1$, we have

$$\sin^{-1}\left(\frac{1}{k}\sin(\theta_T - \sigma_d)\right) \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right). \tag{30}$$

Note that under the parallel navigation, the robot's radial velocity along the line of sight of robot–target is given by

$$v_{R\|} = v_R \cos\left[\sin^{-1}\left(\frac{1}{k}\sin(\theta_T - \sigma_{TR})\right)\right]. \tag{31}$$

2. The cosine function of x when $x \in (-\frac{\pi}{2}, \frac{\pi}{2})$ is always positive.

From the above two remarks, we have

$$\cos\left[\sin^{-1}\left(\frac{1}{k}\sin(\theta_T - \sigma_d)\right)\right] > 0. \tag{32}$$

Thus, under parallel navigation, the robot's radial velocity along the line of sight of robot–target satisfies $v_{R\|} > 0$, and it is possible to write

$$\begin{aligned}
&v_R \cos\left[\sin^{-1}\left(\frac{1}{k}\sin(\theta_T - \sigma_d)\right)\right] \\
&\quad = v_R\sqrt{1 - \frac{1}{k^2}\sin^2(\theta_T - \sigma_{TR})}.
\end{aligned} \tag{33}$$

Now, we consider the motion of the moving goal.

Case 1: $v_{T\|} < 0$

The relative range $\dot{r}_{TR}$ varies as follows

$$\begin{aligned}
\dot{r}_{TR} &= v_{T\|} - v_{R\|} \\
\dot{r}_{TR} &= -v_T\sqrt{1-\sin^2(\theta_T - \sigma_{TR})} - v_R\sqrt{1 - \tfrac{1}{k^2}\sin^2(\theta_T - \sigma_{TR})}
\end{aligned} \tag{34}$$

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

Since $v_{T\|}$ is negative and $v_{R\|}$ is positive, we have $\dot{r}_{TR} < 0$; thus, the relative range is decreasing and the robot reaches its moving goal.

Case 2: $v_{T\|} > 0$

The relative range $\dot{r}_{TR}$ varies as follows

$$\dot{r}_{TR} = v_T\sqrt{1 - \sin^2(\theta_T - \sigma_{TR})} - v_R\sqrt{1 - \frac{1}{k^2}\sin^2(\theta_T - \sigma_{TR})} \tag{35}$$

Both $v_{T\|}$ and $v_{R\|}$ are positive. Note that, since $k > 1$, we have

$$\sqrt{1 - \sin^2(\theta_T - \sigma_{TR})} < \sqrt{1 - \frac{1}{k^2}\sin^2(\theta_T - \sigma_{TR})} \tag{36}$$

and since assumption (H2) states that $v_R > v_T$, we have

$$v_{T\|} < v_{R\|} \tag{37}$$

from which we get $\dot{r}_{TR} < 0$, which means that the relative range between the robot and the goal is decreasing, and the robot reaches its moving goal. □

The following result deals with the particular problem when the target moves in a straight line.

Proposition *If the moving goal is not accelerating, then the robot navigating under parallel navigation is also not accelerating.*

Proof. When the goal moves in a straight line with constant linear velocity, the robot under parallel navigation also moves in a straight line. The proof is simple, when the target moves in a straight line, we have $\theta_T = \text{constant}$. Recall that according to parallel navigation the line-of-sight angle is constant, i.e., $\sigma_{TR} = \text{constant}$. By considering the control input for the robot steering angle [Eq. (22)], we obtain $\theta_R = \text{constant}$. Thus, the robot also moves in a straight line. □

It is worth noting that the radial velocities of the robot and the target along the line of sight of robot–target ($v_{R\|}$ and $v_{T\|}$, respectively) are constant when the target moves in a straight line with $k = \text{constant}$. This can be seen easily from Eq. (23), where a constant value of θ_T results in constant values for $v_{R\|}$ and $v_{T\|}$ (recall that σ_{TR} is constant under the navigation law). This result allows us to derive the following result on the interception time.

Proposition *When the goal moves in a straight line with constant linear velocity, the robot reaches the moving goal at time*

$$t_f = \frac{r_{TR}(t_0)}{v_{R\|} - v_{T\|}} \tag{38}$$

with

$$\begin{aligned} v_{R\|} &= v_R\cos(\theta_R - \sigma_{TR}) \\ v_{T\|} &= v_T\cos(\theta_T - \sigma_{TR}) \end{aligned} \tag{39}$$

where $r_{TR}(t_0)$ is the initial value of the relative distance of robot–target.

Proof. The first expression in Eq. (17) can be rewritten as

$$\dot{r}_{TR}(t) = v_{T\|} - v_{R\|}. \tag{40}$$

Since $v_{R\|}$ and $v_{T\|}$ are constant, it is possible to write the solution of the relative range of robot–target as follows:

$$r_{TR}(t) = [v_{T\|} - v_{R\|}]t + r_{TR}(t_0) \tag{41}$$

and the interception time corresponds to

$$r_{TR}(t_f) = 0. \tag{42}$$

From Eqs. (41) and (42), we get

$$t_f = \frac{r_{TR}(t_0)}{v_{R\|} - v_{T\|}}. \tag{43}$$

□

6.1. Heading regulation

In most cases, a heading regulation is necessary in order to put the robot in a configuration where the application of parallel navigation is possible. The initial value of the robot orientation is $\theta_R(t_0)$, which may be different from the orientation angle required by the control law. The heading regulation is accomplsihed by putting

$$\dot{\theta}_R = -K\left(\theta_R - \theta_R^{des}\right) \tag{44}$$

where θ_R^{des} is given by the control law. K is a real positive number. This approach allows to derive $\theta_R(t)$ to its desired value from its initial value given by $\theta_R(t_0)$. An example is shown in Fig. 3.

6.2. A comparison with the pursuit

The pursuit is the most classical navigation law used to reach a moving goal. It is implemented using different types of sensors. In pursuit, the robot localizes the target and moves toward it. Here, we present a simple comparison between pursuit and parallel navigation, where the target moves in a straight line. The comparison is shown in Fig. 4, from which we have the following remarks: Parallel navigation leads to

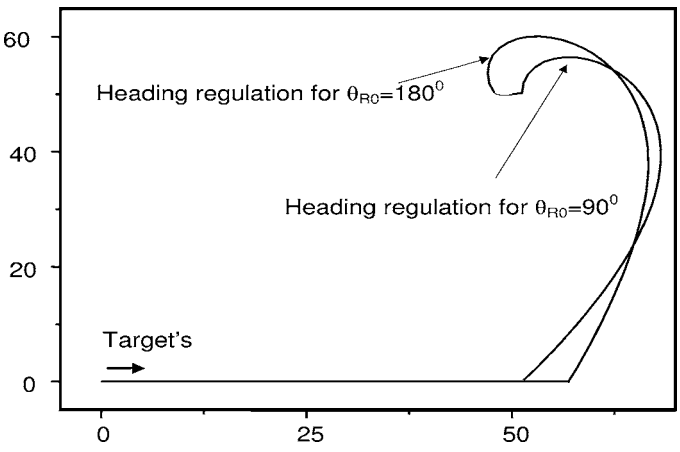


Fig. 3. Parallel navigation after heading regulation.

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

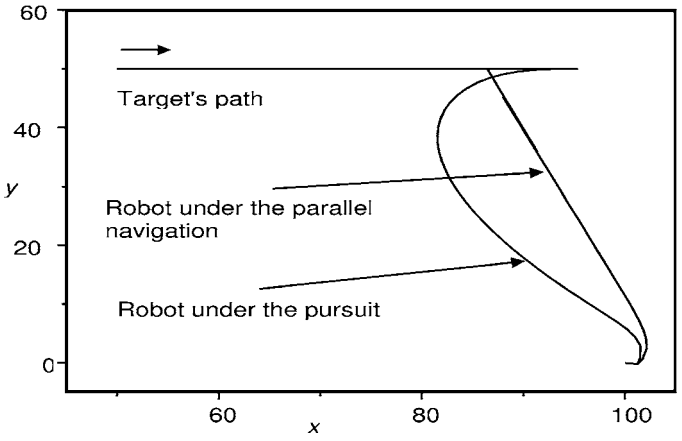


Fig. 4. A comparison with pure pursuit.

reaching the goal faster than pursuit. The path of pursuit is more curved than the path of parallel navigation. This means that pursuit results in a higher acceleration.

7. In the Presence of Obstacles

The problem of navigation toward a moving goal becomes more difficult in the presence of obstacles. In this case, the integration of the local and the global path-planning algorithms is necessary. The sensory system provides the robot with the necessary information on the obstacles and the goal. We suggest to use the parallel navigation guidance law in combination with an obstacle-avoidance algorithm. Thus, the robot moves in two modes, the navigation mode and the obstacle-avoidance mode. We choose an approximate cell decomposition approach. This approach is simple, and can be easily integrated with the parallel navigation law. Initially, the robot navigates using the parallel navigation, and when an obstacle is detected, the obstacle-avoidance mode is activated. After the obstacle is avoided, the robot navigates using the parallel navigation law.

7.1. Obstacle-avoidance mode

The robot and the moving goal move in the workspace $W \subset I\!R^2$, where W is cluttered with K obstacles B_j, $j = 1, \ldots, K$. The region in the workspace formed by the sum of the obstacles is denoted by O_r ($O_r = \cup_{j=1}^{K} B_j$), and $C_{free} = W - O_r$ is the free space. The robot path must lie in C_{free}. In a cell decomposition algorithm, the workspace W is broken down into nonoverlapping cells. The size of the cells can be locally adapted to the geometry of the obstacles. A cell decomposition of W is defined as a finite collection of cells $\varepsilon_i, i = 1, \ldots, M$, such that

1. $W = \cup_{i=1}^{M} \varepsilon_i$
2. $\forall i, \forall j,\ i = 1, \ldots, M,\ j = 1, \ldots M$, for $i \neq j$, $\text{int}(\varepsilon_i) \cap \text{int}(\varepsilon_j) = \phi$.

This means that first, the sum of all cells is the workspace, and second, the cells do not overlap. A cell ε_i can be classified as follows:

1. Empty: if the interior of the cell does not intersect with O_r, i.e., $\text{int}(\varepsilon_i) \cap O_r = \phi$.
2. Full: if ε_i is entirely contained in O_r, i.e. $\varepsilon_i \subseteq O_r$.
3. Mixed: if it is neither empty nor full.

In order to adapt the parallel navigation to the obstacle-avoidance mode, we discretize the robot kinematics equations. Using Euler algorithm, we get

$$\begin{aligned} x_R(n+1) &= h v_R \cos[\theta_R(n)] + x_R(n) \\ y_R(n+1) &= h v_R \sin[\theta_R(n)] + y_R(n) \\ \theta_R(n) &= \sigma_{TR} + \sin^{-1}\left[\tfrac{1}{k}\sin(\theta_T(n) - \sigma_{TR})\right] \end{aligned} \tag{45}$$

where h is the step size. Recall that σ_{TR} is constant and equal to its initial value in the case of our control law. The integrated algorithm, which is activated after an obstacle is detected in the robot active region is the following:

1. Compute $[x(n+1), y(n+1)]$ using Eq. (45) .
2. Does $[x(n+1), y(n+1)]$ fall in an empty cell?
   yes: move to $[x(n+1), y(n+1)]$, put $n \leftarrow n+1$ and go to 1.
   no: move to the nearest empty cell to $[x(n+1), y(n+1)]$, put $n \leftarrow n+1$ and go to 1.
3. Stop when goal is reached.

8. In the Presence of Uncertainties

Our goal here is to present a brief study on the influence of uncertainties. The study of the navigation problem in the presence of uncertainty is another difficult and complex problem that will be considered in our future research.

8.1. Uncertainty in the target position and orientation angle

Uncertainty in the goal's position and orientation angle is the most important part of uncertainty in the navigation problem. An approach based on odometry is used here. Odometry is widely used to provide real-time position estimation of the target. The pose of the target is defined in the form of estimated values for position and orientation as follows:

$$X_T = [x_T, y_T, \theta_T]^T . \tag{46}$$

The uncertainty in pose is represented by a covariance matrix as follows:

$$C_T = \begin{bmatrix} \sigma_{x_T}^2 & \sigma_{x_T y_T} & \sigma_{x_T \theta_T} \\ \sigma_{y_T x_T} & \sigma_{y_T}^2 & \sigma_{y_T \theta_T} \\ \sigma_{\theta_T x_T} & \sigma_{\theta_T y_T} & \sigma_{\theta_T}^2 \end{bmatrix} . \tag{47}$$

The odometric position estimation process is activated at a regular interval Δt. Let Δs_T and $\Delta \alpha_T$ be the change in the translation and rotation, respectively. We can write

$$\Delta y_T = \begin{bmatrix} \Delta s_T \\ \Delta \alpha_T \end{bmatrix} . \tag{48}$$

The accumulated translation and rotation are calculated as

$$s_T^{new} = s_T^{old} + \Delta s_T \tag{49}$$

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

$$\theta_{\mathrm{T}}^{\mathrm{new}} = \theta_{\mathrm{T}}^{\mathrm{old}} + \Delta\alpha_{\mathrm{T}}. \tag{50}$$

The change in pose of the target is given by

$$\Delta X_{\mathrm{T}} = \begin{bmatrix} \Delta x_{\mathrm{T}} \\ \Delta y_{\mathrm{T}} \\ \Delta\theta_{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} \Delta s \cos\left(\theta_{\mathrm{T}} + \frac{\Delta\alpha_{\mathrm{T}}}{2}\right) \\ \Delta s \sin\left(\theta_{\mathrm{T}} + \frac{\Delta\alpha_{\mathrm{T}}}{2}\right) \\ \Delta\alpha_{\mathrm{T}} \end{bmatrix}. \tag{51}$$

The target position is estimated as follows and then updated

$$X_{\mathrm{T}}^{\mathrm{new}} = X_{\mathrm{T}}^{\mathrm{old}} + \Delta X_{\mathrm{T}}. \tag{52}$$

The estimated position of the target is accompanied by an estimate of the uncertainty expressed by the covariance matrix C_{T}. The sensitivity of X_{T} to the observed value is characterized by the Jacobian matrix J, which is given by

$$J = \frac{\partial X_{\mathrm{T}}}{\partial \Delta Y_{\mathrm{T}}} \tag{53}$$

from which we get

$$J = \frac{\partial \begin{bmatrix} x_{\mathrm{T}} \\ y_{\mathrm{T}} \\ \theta_{\mathrm{T}} \end{bmatrix}}{\partial \Delta y_{\mathrm{T}}} = \frac{\partial \begin{bmatrix} \Delta s \cos(\theta_{\mathrm{T}}) \\ \Delta s \sin(\theta_{\mathrm{T}}) \\ \Delta\alpha_{\mathrm{T}} \end{bmatrix}}{\partial \Delta y_{\mathrm{T}}} \tag{54}$$

which gives

$$J = \begin{bmatrix} \cos(\theta_{\mathrm{T}}) & 0 \\ \sin(\theta_{\mathrm{T}}) & 0 \\ 0 & 1 \end{bmatrix}. \tag{55}$$

The Jacobian matrix allows us to write the estimate of the position as follows:

$$X_{\mathrm{T}}^{\mathrm{new}} = X_{\mathrm{T}}^{\mathrm{old}} + J\Delta Y.$$

Thus, the covariance matrix is updated as follows:

$$C_{\mathrm{T}}^{\mathrm{new}} = C_{\mathrm{T}}^{\mathrm{old}} + J^{\mathrm{T}} C_{\mathrm{T}}^{\mathrm{old}} J.$$

It is important to note that uncertainty in orientation strongly contributes to the Cartesian position.

8.2. Uncertainty in the line-of-sight angle

Recall that the line-of-sight angle is given by

$$\tan\sigma_{\mathrm{TR}} = \frac{y_{\mathrm{T}} - y_{\mathrm{R}}}{x_{\mathrm{T}} - x_{\mathrm{R}}}. \tag{56}$$

We put

$$z = \tan\sigma_{\mathrm{TR}} = f(x_{\mathrm{T}}, y_{\mathrm{T}}, x_{\mathrm{R}}, y_{\mathrm{R}}). \tag{57}$$

The uncertainty in the input is represented by a covariance matrix as follows:

$$C_{\mathrm{I}} = \begin{bmatrix} \sigma_{x_{\mathrm{T}}}^2 & \sigma_{x_{\mathrm{T}}y_{\mathrm{T}}} & \sigma_{x_{\mathrm{T}}x_{\mathrm{R}}} & \sigma_{x_{\mathrm{T}}y_{\mathrm{R}}} \\ \sigma_{y_{\mathrm{T}}x_{\mathrm{T}}} & \sigma_{y_{\mathrm{T}}}^2 & \sigma_{y_{\mathrm{T}}x_{\mathrm{R}}} & \sigma_{y_{\mathrm{T}}y_{\mathrm{R}}} \\ \sigma_{x_{\mathrm{R}}x_{\mathrm{T}}} & \sigma_{x_{\mathrm{R}}y_{\mathrm{T}}} & \sigma_{x_{\mathrm{R}}}^2 & \sigma_{x_{\mathrm{R}}y_{\mathrm{T}}} \\ \sigma_{y_{\mathrm{R}}x_{\mathrm{T}}} & \sigma_{y_{\mathrm{R}}y_{\mathrm{T}}} & \sigma_{y_{\mathrm{T}}x_{\mathrm{R}}} & \sigma_{y_{\mathrm{R}}}^2 \end{bmatrix}. \tag{58}$$

The uncertainty in z is then given by

$$C_z = \nabla f C_{\mathrm{I}} [\nabla f]^T \tag{59}$$

with

$$\nabla f = \left[\frac{\partial f}{\partial x_{\mathrm{T}}}, \frac{\partial f}{\partial y_{\mathrm{T}}}, \frac{\partial f}{\partial x_{\mathrm{R}}}, \frac{\partial f}{\partial y_{\mathrm{R}}}\right] \tag{60}$$

and $[\cdot]^T$ stands for the transpose matrix. Simulation examples under uncertainties in the line-of-sight angle are shown in Section 9.

9. Simulation

In this section, we use extensive simulation to illustrate our approach, where different scenarios are considered. Obstacle-free workspace is considered first. Both accelerating and nonaccelerating targets are considered.

9.1. The case of a nonaccelerating goal

In this case, the moving goal moves with a constant speed and a constant orientation angle. For this scenario, we consider a goal moving in a straight line parallel to the y-axis. We consider two different cases. In the first case, the goal is approaching the robot ($v_{\mathrm{T}\|} < 0$), and in the second case, the goal is moving away from the robot ($v_{\mathrm{T}\|} > 0$). We have for the robot $v_{\mathrm{R}} = 3\,\mathrm{m/s}$, and for the goal, $v_{\mathrm{T}} = 2\,\mathrm{m/s}$. The robot and the goal paths for these scenarios are shown in Figs. 5 and 6. In both cases, the robot navigates in a straight line and reaches the moving goal successfully.

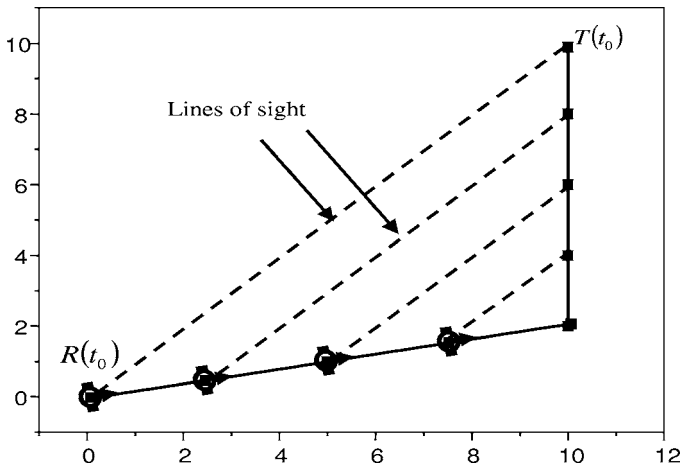


Fig. 5. Robot's navigation toward a goal moving in a straight line (the goal is approaching).

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

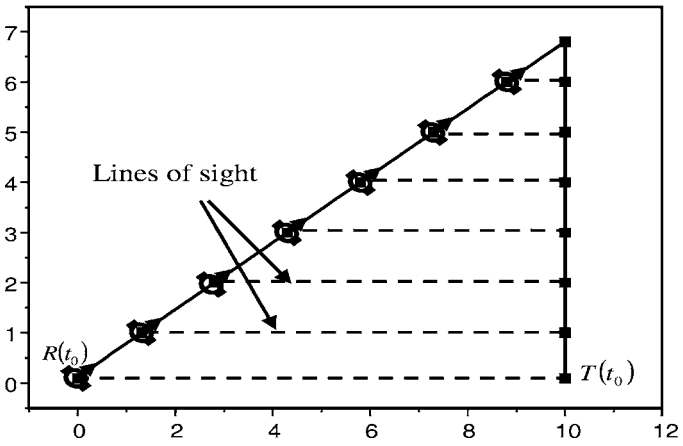


Fig. 6. Robot's navigation toward a goal moving in a straight line (goal moving away).

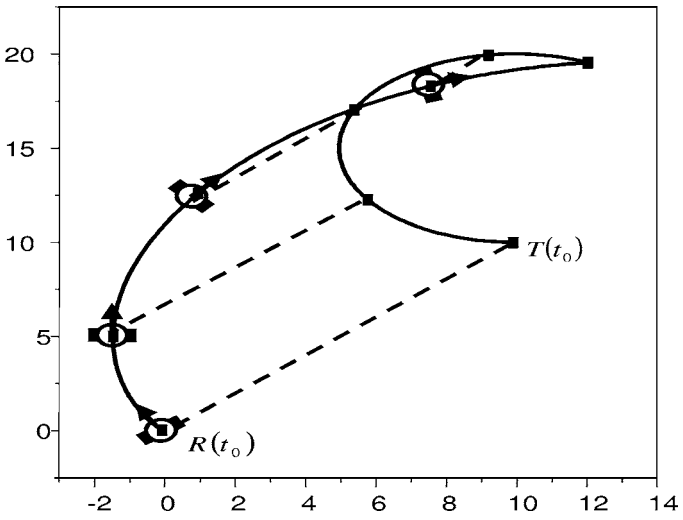


Fig. 7. Robot's navigation toward a target moving in a circle (clockwise motion).

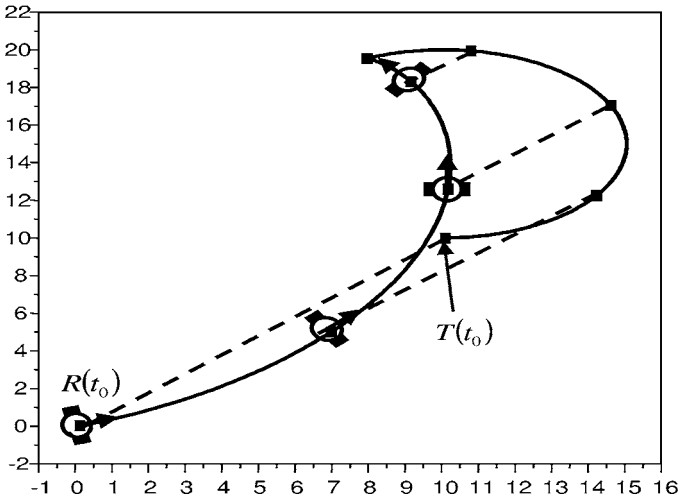


Fig. 8. Robot's navigation toward a goal moving in a circle (counterclockwise motion).

9.2. Goal moving in a circle

In this example, we illustrate the robot navigation toward a goal moving in a circle. The paths of the robot and the target are shown in Figs. 7 and 8. In Figs. 7, the robot moves clockwise, and in Fig. 6, it moves counterclockwise. In both cases, the robot reaches the moving goal successfully.

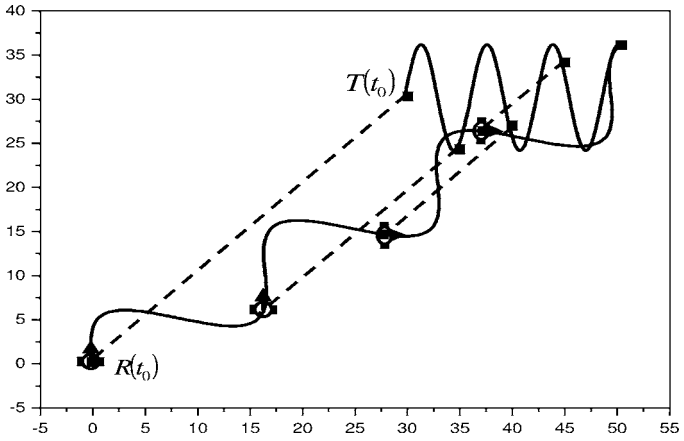


Fig. 9. Robot's navigation toward a goal moving in a sinusoidal motion.

9.3. Goal moving in a sinusoidal motion

Goal moving in a sinusoidal motion is among the most difficult maneuvers. This case is considered for simulation, and is illustrated in Fig. 9. It turns out that the robot path under our navigation law is also sinusoidal, as shown in Fig. 9. The robot reaches the moving goal successfully even for this difficult type of motion.

9.4. Navigation for different velocity ratios

As shown in Eq. (22), the robot steering angle is a function of the velocity ratio. Our aim is to compare different velocity ratios, namely $k = 1.25$, 1.5, and 2. The navigation toward a moving goal using these values of k is illustrated in Fig. 10. Curve 1 represents the path for $k = 2$, curve 2 for $k = 1.5$, and curve 3 for $k = 1.25$. Points P_1, P_2, and P_3 correspond to the interception points for $k = 2$, 1.5, and 1.25, respectively. The path traveled by the robot is different for different values of the velocity ratio. The robot reaches its moving goal faster for higher values of k.

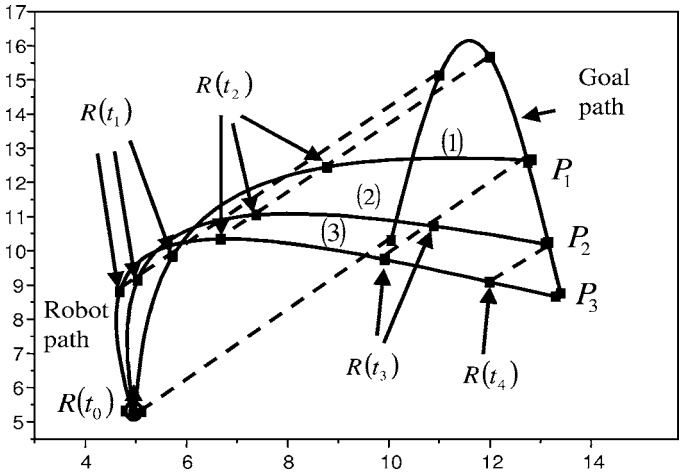


Fig. 10. Comparison of the robot's path for different velocity ratios, Curve 1: $k = 2$; Curve 2: $k = 1.5$; Curve 3: $k = 1.25$. P_1, P_2, and P_3 are the interception points for $k = 2$, 1.5, and 1.25, respectively.

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

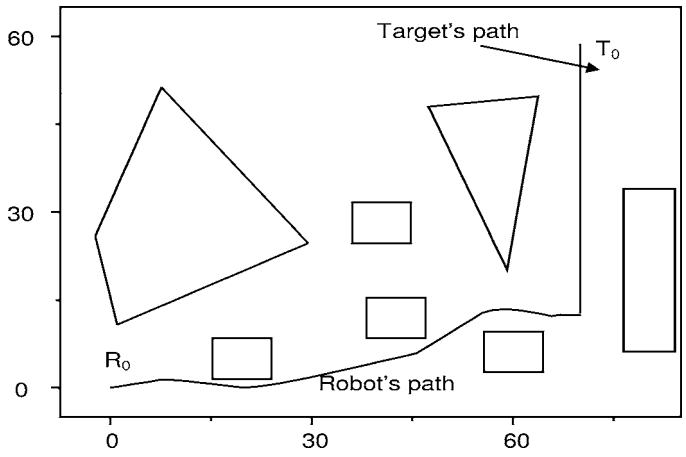


Fig. 11. Robot's is navigation toward a goal moving in a straight line in the presence of obstacles (first scenario).

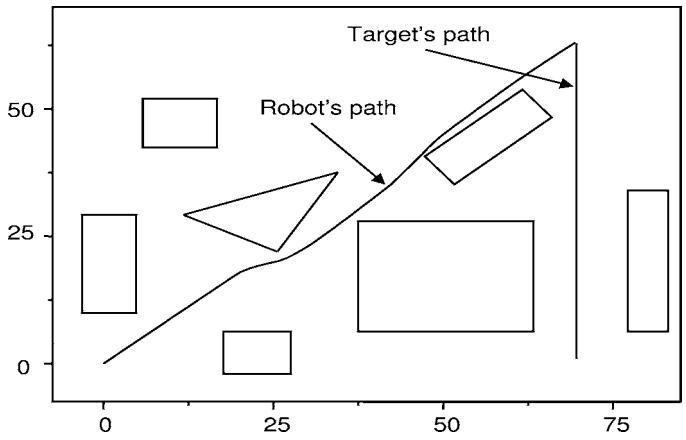


Fig. 12. Robot's navigation toward a goal moving in a straight line in the presence of obstacles (second scenario).

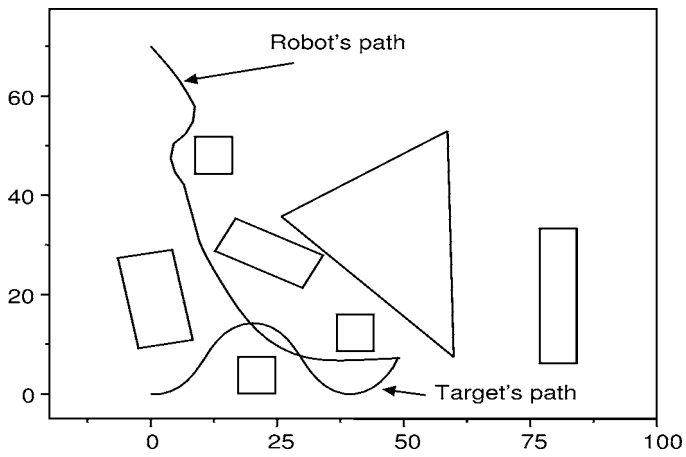


Fig. 13. Robot's navigation toward a goal moving in a sinusoidal motion in the presence of obstacles (first scenario).

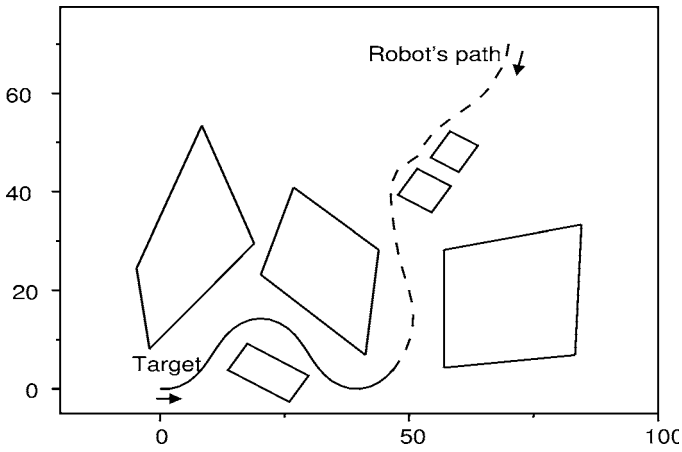


Fig. 14. Robot's navigation toward a goal moving in a sinusoidal motion in the presence of obstacles (second scenario).

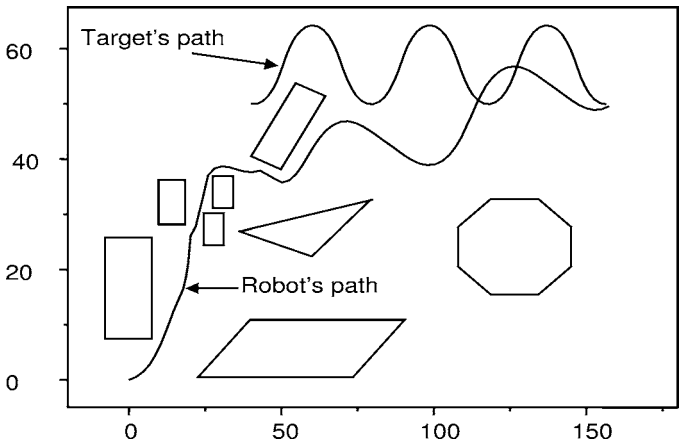


Fig. 15. Robot's navigation toward a goal moving in a sinusoidal motion in the presence of obstacles (third scenario).

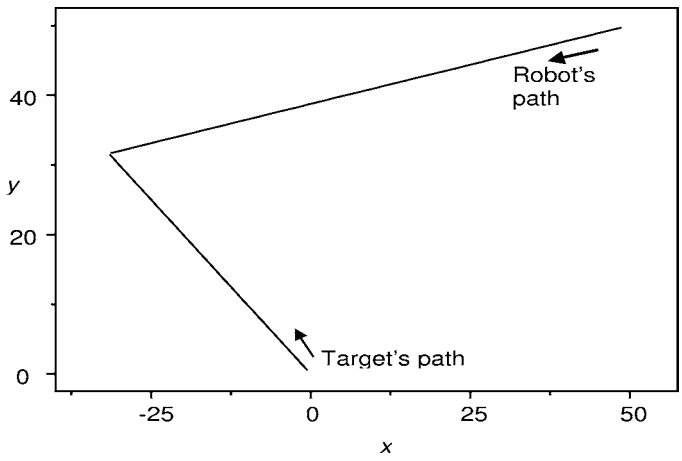


Fig. 16. Robot's navigation toward a goal moving in a straight line in the absence of uncertainties.

9.5. In the presence of obstacles

The robot aims to reach the moving goal with the constraint that $P_R(t)$ lies in C_{free}. Both navigation and obstacle-avoidance modes are used. Five scenarios are considered here. In the scenario of Figs. 11 and 12, the goal moves in a straight line. More difficult scenarios are shown in Figs. 13–15, where the target performs a sinusoidal motion. The robot performs both tasks, navigation toward the goal and obstacle-avoidance, successfully.

9.6. In the presence of uncertainties

We also consider one simulation example where the robot moves in a straight line. We also consider uncertainties in velocity ratio and in the line-of-sight angle. The scenario in the absence of uncertainties is shown in Fig. 16. Figure 17

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

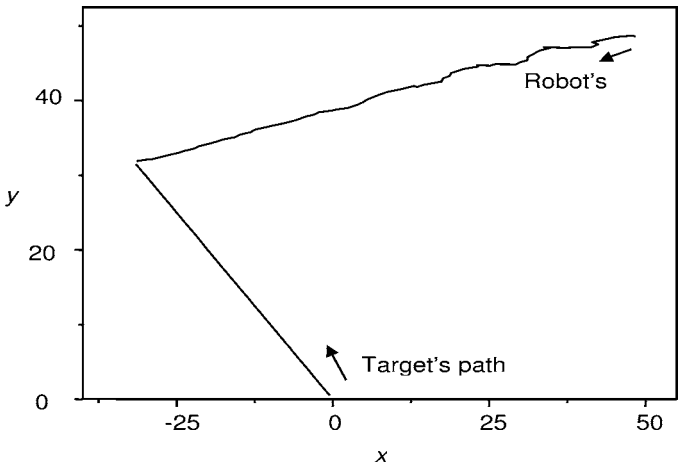


Fig. 17. Robot's navigation toward a goal moving in a straight line in the presence of uncertainties in the line-of-sight angle.

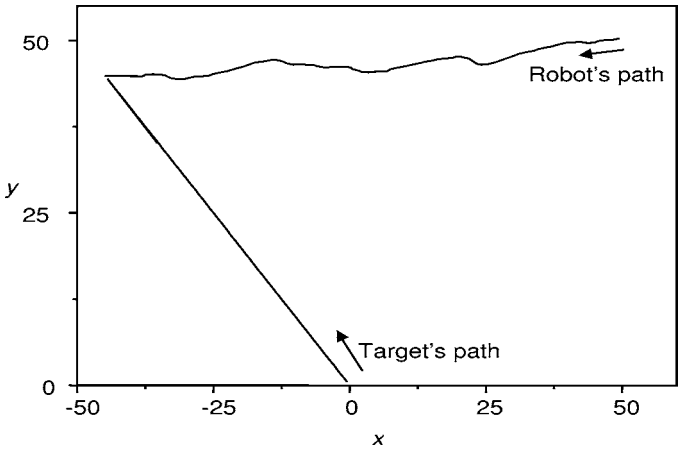


Fig. 18. Robot's navigation toward a goal moving in a straight line in the presence of uncertainties in the velocity ratio.

shows the robot's path in the presence of uncertainty in the line-of-sight angle. Note that the path becomes a straight line after a certain time. This is due to improvement in the estimation of the line of sight when the robot gets closer to the target. Figure 18 shows the robot's path in the presence of uncertainty in the velocity ratio. Clearly, even under uncertainties, the robot reaches the goal. Also filtering techniques can be used to significantly improve the navigation in the presence of uncertainties.

10. Conclusion

In this paper, a method for robot navigation toward moving objects with unknown maneuvers is presented. This is a real-time problem, since the goal maneuvers are not *a priori* known to the robot. Our strategy is based on parallel navigation. We first derive a navigation model representing the motion of the target as seen by the robot. The aim of the control strategy is to keep the line-of-sight angle constant between the robot and the target. Thus, the robot moves in lines that are parallel to the initial line of sight. We considered uncertainties in the target's position and the line-of-sight angle. It is shown that uncertainty affects the path, but it does not affect interception. This problem will investigated in more detail in our future research. In the presence of obstacles, two navigation modes are used, namely, parallel navigation and obstacle-avoidance mode. It is proven that the robot navigating using parallel navigation reaches the moving target successfully under some conditions. The navigation strategy is illustrated using an extensive simulation, where various scenarios are considered.

References

1. Z. Tang and U. Ozguner, "Motion planning for multitarget surveillance with mobile sensor agents," *IEEE Trans. Robot.* **21**, 1–11 (2005).
2. P. Rybski, S. Stoeter, M. Erickson, M. Gini, D. Hougen and N. Papanikolopoulos, "A team of robotic agents for surveillance," *Proceedings of the 4th International Conference on Autonomous Agents*, Spain (2000) pp. 9–16.
3. M. Saptharishi, C. Oliver, C. Diehl, K. Bhat, J. Dolan, A. Trebi–Ollennu and P. Kholsa, "Distributed surveillance and reconnaissance using multiple autonomous ATV's cyberscout," *IEEE Trans. Robot. Autom.* **18**, 826–836 (2002).
4. E. Marchand, P. Bouthemy, F. Chaumette and V. Moreau, "Robust real-time visual tracking using a 2D–3D model-based approach," *Proceedings of the IEEE International Conference on Robotics and Automation*, Corfu, Greece (1999) pp. 262–268.
5. L. Davis, V. Philomin and R. Duraiswami, "Tracking humans from a moving platform," *Proceedings of the IEEE International Conference on Pattern Recognition*, Barcelona, Spain (2000) pp. 171–178.
6. S. Feyrer and A. Zell, "Detection, tracking, and pursuit of humans with an autonomous mobile robot," *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, Korea (1999) pp. 864–869.
7. S. Feyrer and A. Zell, "Tracking and pursuing persons with a mobile robot," *Proceedings of the International Workshop on Recognition, Analysis and Tracking Faces and Gestures in Real time Systems*, Corfu, Greece (1999) pp. 83–88.
8. J. Chung and H. S. Yang, "Fast and effective multiple moving targets tracking method for mobile robots," *Proceedings of the IEEE International Conference on Robotics and Automation*, Nagoya, Japan (1995) pp. 2645–2650.
9. C. Coue and P. Bessiere, "Chasing an illusive target with a mobile robot," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Hawaii (2001) pp. 1370–1375.
10. C. Y. Lee, H. Gonzalez-Banos and J. C. Latombe, "Real-time tracking of an unpredictable target amidst unknown obstacles," *Proceedings of the 7th International Conference on Control, Automation, Robotics and Vision*, Singapore (2002) pp. 596–601.
11. J. Graham and K. Shillcutt, "Robot tracking of human subjects in field environments," *Proceedings of the 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space* (2003) pp. 1–7.
12. D. Schulz, W. Burgard, D. Fox and A. B. Cremers, "People tracking with a mobile robot using sample-based joint probabilistic data association filters", *Int. J. Robot. Res.* **22**, 99–116 (2003).
13. M. Asada and T. Nakamura,"Target reaching behavior learning with occlusion detection and avoidance for a stereo vision-based mobile robot," *Proceedings of ROBOLEARN96: An International Workshop on Learning for Autonomous Robots* (1996) pp. 1–10 .
14. C. Gaskett, P. Brown, G. Cheng and A. Zelinsky, "Learning implicit models during target pursuit," *Proceedings of the IEEE International Conference on Robotics and Automation*, Taipei (2003) pp. 4122–4129.
15. D. Schulz, W. Burgard, D. Fox and A. B. Cremers, "Tracking multiple moving targets with a mobile robot using particle

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press

filters and statistical data association," *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, South Korea (2001) pp. 1665–1670.
16. F. Chaumette, P. Rives and B. Espiau, "Positioning of a robot with respect to an object, tracking it and estimating its velocity by visual servoing," *Proceedings of the IEEE International Conference on Robotics and Automation*, California (1991) pp. 2248–2253.
17. B. H. Kim, D. K. Roh, J. M. Lee, M. H. Lee, K. Son, M. C. Lee, J. W. Choi and S. H. Han, "Localization of a mobile robot using images of a moving target," *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, South Korea (2001) pp. 253–258.
18. J. Dias, C. Peredes, I. Fonseca, H. Araujo, J. Batista and A. T. Almeida, "Simulating pursuit with machine experiments with robots and artificial vision," *IEEE Trans. Robot. Autom.* **3**, 1–18 (1998).
19. M. C. Tsai, K. Y. Chen, M. Y. Cheng and K. C. Lin, "Implementation of a real-time moving object tracking system using visual servoing," *Robotica* **21**, 615–625 (2003).
20. L. Parker, B. Birch and C. Reardon, "Indoor target intercept using an acoustic sensor network and dual wavefront path planning," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, NV (2003) pp. 278–283.
21. M. Mazo, A. Speranzon, K. H. Johansson and X. Hu, "Multi-robot tracking of a moving object using directional sensors," *Proceedings of the IEEE International Conference on Robotics and Automation*, New Orleans (2004) pp. 1103–1108.
22. C. Wang, C. Thorpe and A. Suppe, "Speeds," *Proceedings of the IEEE Intelligent Vehicles Symposium*, Brooklyn, NY (2003) pp. 416–421.
23. W. Kuang and A. Morris, "Ultrasound speed compensation in an ultrasonic robot tracking system," *Robotica* **18**, 633–637 (2000).
24. M. Aircardi, G. Casalino, A. Bichi and A. Balestrio, "Closed loop steering of unicycle-like vehicles via Lyapunov techniques," *IEEE Robot. Autom. Mag.* 27–35 (1995).
25. H. Yang and B. Sikdar, "A protocol for tracking mobile targets using sensor networks," *Proceedings of the IEEE International Workshop on Sensor Network Protocols and Applications* (2003) pp. 71–81.
26. R. Murrieta, A. Sarmiento and S. Hutchinson, "On the existence of a strategy to maintain a moving target within the sensing range of an observer reacting with delay," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, NV (2003) pp. 1184–1191.
27. B. Thuilot, P. Martinet, L. Cordesses and J. Gallice, "Position based visual servoing: Keeping the object in the field of vision," *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, DC, 2002 pp. 1624–1629.
28. R. Murrieta-Cid, A. Sarmiento, S. Kloder, S. Hutchinson, F. Lamiraux and J. P. Laumond, "Maintaining visibility of a moving holonomic target at a fixed distance with a non-holonomic robot," *LAAS report N 04575*, 2004.
29. T. Muppirala, R. Murriera-Cid and S. Hutchinson, "Optimal motion strategies based on critical events to maintain visibility of a moving target," *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain (2005) pp. 3826–3831.
30. J. Spletzer and C. taylor, "dynamic sensor planning and control for optimally tracking targets," *Int. J. Robot. Res.* **22**, 7–20 (2003).
31. S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonom. Robots* **13**, 207–222 (2002).
32. M. D. Adams, "High speed target pursuit and asymptotic stability in mobile robotics," *IEEE Trans. Robot. Autom.* **15**, 230–236 (1999).
33. S. O. Lee, Y. J. Cho, M. Hwang-Bo, B. J. You and S. R. Oh, "A stable target-tracking control for unicycle mobile robots," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2000) pp. 1822–1827.
34. R. C. Luo and T. M. Chen, "Traget tracking by Grey prediction theory and look-ahead fuzzy logic control," *Proceedings of the IEEE International Conference on Robotics and Automation*, Michigan (1999) pp. 1176–1182.
35. T. M. Chen and R. C. Luo, "Mobile target tracking using hierarchical Grey-fuzzy motion decision-making method," *Proceedings of the IEEE International Conference on Robotics and Automation*, California (2000) pp. 2118–2123.
36. W. Tong and T. H. S. Li, "Realization of two-dimensional target tracking problem via autonomous mobile robot using fuzzy sliding mode control," *Proceedings of the IEEE International Conference on Industrial Electronics*, Aachen (1998) pp. 1158–1164.
37. I. K. Jeong and J. J. Lee, "Evolving fuzzy logic controllers for multiple mobile robots solving a continuous pursuit problem," *Proceedings of the IEEE International Conference on Fuzzy Systems*, Seoul South Korea (1999) pp. 685–690.
38. S. Sachs, S. LaValle and S. Rajko, "Visibility-based pursuit-evasion in an unknown planar environment," *Int. J. Robot. Res.* **23**, 3–23 (2004).
39. V. Isler, S. Sampath and S. Khanna, "Randomized pursuit-evasion in a polygonal environment," *IEEE Trans. Robot.* **21**, 875–884 (2005).
40. S. LaValle and J. Hutchinson, "Visibility-based pursuit-evasion: The case of curved environment," *IEEE Trans. Robot. Autom.* **17**, 196–203 (2001).
41. P. Kachroo, S. Shedied and H. Vanlandingham, "Pursuit-evasion: The herding noncooperative dynamic game—The stochastic model," *IEEE Trans. Robot. Autom.* **32**, 37–42 (2001).
42. H. Yamaguchi, "A distributed motion coordination strategy for multiple nonholonomic mobile robots in cooperative hunting operations," *Robot. Auton. Syst.* **43**, 257–282 (2003).
43. H. Yamaguchi, "A cooperative hunting behavior by mobile-robot troops," *Robot. Auton. Syst.* **18**, 931–940 (1999).
44. Y. Fan and A. Balasuria, "Target tracking by underwater robots," *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, Tucson, AZ (2001) pp. 9–15.
45. R. Vidal, O. Shakernia, H. Kim, D. Shim and S. Sastry, "Probabilistic pursuit–evasion games: Theory, implementation, and experimental evaluation," *IEEE Trans. Robot. Autom.* **18**, 262–669 (2002).
46. A. Creual, F. Chaumette and P. Bouthemy, "Complex object tracking by visual servoing based on 2D image motion," *Proceedings of the IEEE International Conference on Pattern Recognition*, Brisbane, Australia (1998) pp. 1251–1254.
47. F. Bensalah and F. Chaumette, "Compensation of abrupt motion changes in target tracking by visual servoing," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pittsburgh, PA (1995) pp. 181–187.
48. L. Hsu and F. Lizarralde, "Robust adaptive visual tracking control: Analysis and experiments," *Proceedings of the IEEE International Conference on Control Applications*, Anchorage, AK (2000) pp. 874–879.
49. N. A. Sheyndor, *Missile Guidance and Pursuit, Kinematics, Dynamics and Control* (Hartwood, Chichester, UK, 1998).
50. F. Belkhouche and B. Belkhouche, "Ball interception by a mobile robot goalkeeper using parallel navigation," *Proceedings of the 35th International Symposium on Robotics*, Paris (2004) paper WE14-145.
51. E. Huber and D. Kortenkamp, "Using stereo vision to pursue moving agents with a mobile robot," *Proceedings of the IEEE International Conference on Robotics and Automation*, Nagoya, Japan (1995) pp. 2340–2346.
52. D. Schulz, W. Burgard, D. Fox and A. B. Cremers, "Tracking multiple moving objects with a mobile robot," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Hawaii (2001) pp. 371–377.

https://doi.org/10.1017/S0263574706002992 Published online by Cambridge University Press