Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Tracking under the nonholonomic constraint using cubic navigation laws

J. Vargas, S. Mendez and F. Belkhouche

***Abstract*—This paper deals with the problem of mobile robot tracking an unpredictably moving target in the presence of obstacles. We suggest a new family of navigation laws that consists of cubic navigation functions. These laws depend on various external parameters that can be used to adjust the robot's path, especially to avoid collision with obstacles. The suggested method allows an easy combination between the global aspects of target tracking and the local aspects of obstacle avoidance. A method is suggested to tune and adjust the parameters of the navigation laws and derive a locally optimal path. The method is illustrated using an extensive simulation.**

***Index Terms*—tracking, path planning.**

I. Introduction

Wheeled mobile robots navigation in the presence of obstacles is among the most important topics in the field of robotics. The research on the problem of navigation to reach a stationary object is substantial. However, less attention is given to the problem of navigation to reach or track a moving target. Tracking and reaching a moving target by using a wheeled mobile robot is important in various applications, such as soccer robotics, automated surveillance, and various military applications. The problem of tracking-interception or pursuit is studied in different environment contests: plane, grid, or graph [22]. In a grid representation, the controls and the space are discrete. In this paper, the problem is studied in the plane, which represents a more realistic approach, since it uses a continuous representation for the space and the control inputs. The families of methods used for tracking-interception of a moving target in the plane can be divided mainly into two groups:

- Methods based on pursuit-like approach: In this approach, the robot always tracks the path of the target. Thus the velocity vector of the robot is always directed towards the target's position, and the robot is always heading towards the target. In certain situations, there exist some deviation angle, thus the robot follows the target's path with a certain deviation. This case corresponds to either lead or lag pursuit. Many sensor-based methods use this strategy. Most animals predators use the pursuit to catch the prey.
- Methods based on rendezvous-like approach: In this approach the robot calculates a rendezvous point where both the robot and the target will arrive at the same time, and moves towards this point. This approach aims to keep the visibility line robot-target parallel to the initial visibility line. This strategy is used by players in certain games [1].

The authors are with TAMIU. This project is partially supported by an Award from the U.S. Department of Education, Award # P031C080083 .

The suggested approach combines the various aspects of the pursuit and a rendezvous in a single law. In this paper, we consider the tracking-interception problem for accelerating targets in the presence of obstacles. It is clear that the problem in this case is more difficult and requires the integration of local path planning for obstacle avoidance and global path planning to track the target. When the target motion is known in advance with a certain precision, the problem becomes easier, and the motion of the robot can be pre-planned. The problem is more interesting when the target's motion is not a-priori known. This kind of problems is discussed not only for wheeled mobile robots, but also for robotic manipulators to achieve smooth grasping of stationary and moving objects ([2], [3]). In ([2], [3]), the authors used the proportion navigation and its variants. The proportional navigation is widely studied in the aerospace community [4]. Vision-based methods are widely used to solve tracking-interception problems ([8], [7]). These methods are among the most important feature-based methods, where modeling the target's motion is not necessary. In [6] the authors suggest a method for tracking humans in real time based on feature detection. In [12] the authors suggest an algorithm for tracking humans from a moving platform by using a camera. The problem of using multiple robots for cooperative hunting behavior is studied in [9] and [10], where model-based methods are used. In [10], the authors suggest to use a rendezvous-like approach. The problem of pursuit-evasion is widely considered in the artificial intelligence community. In the pursuit-evasion problems, the goal is to plan the path of one or more searchers (robots) moving in a given working space so as to guarantee the detection of the a moving target. In many situations, the target is also intelligent, where it applies evasive control laws. Clearly, this renders the problem more difficult. Pursuit-evasion problems are studied using two different families of methods: (a) Continuous approaches that are based on differential games and optimal control theory and (b) discrete approaches that are based on graph theory. Many authors consider the problem of pursuit evasion under different constraints such as limited field of view [20], imprecise target location [18], and unknown environment [13]. In [19] a hybrid control strategy that combines the continuous and discrete approaches is used. In [14], the authors suggest human navigational principles for robotic interception, where the perceptual principles allow the robot to intercept the target without the need for time consuming calculations to determine the world coordinates of the robot and the target. Note that the problem of target tracking and interception is also studied in 3D environments for air vehicles ([16], [17]) and underwater

978-1-4244-2794-9/09/$25.00

vehicles [15].

In this paper we consider a surveillance problem of intercepting and maintaining the target at constant distance from the moving observer. We suggest to solve this problems by using a family of nonlinear kinematics functions [23]. These navigation functions are based on the relative kinematics equations combines with geometric rules. They are adequate for tracking maneuvering targets. The navigation functions suggested here are functions of one variable only, which can be derived from the relative position. This paper is organized as follows. In the next section, modeling and geometry are discussed. In section III, the control law is introduced. Results concerning interception are introduced and proven. The collision avoidance problem is discussed in section IV, and a simulation study is shown in section V.

II. POSITION OF THE PROBLEM, MODELING AND GEOMETRY

In this section we recall the geometry of the tracking problem, and we write the relative kinematics equations of motion. For more details about the derivation, the reader is referred to [21]. The working space is attached to a global reference frame of coordinates {W}. The goal of the robot is to reach the target with a desired final orientation angle. This is equivalent to driving the robot from an initial configuration q_i to a desired final configuration q_f. The difficulty of the problem comes from the fact that the desired final orientation is not known and depends on the target motion. The target is modeled as a circular object that is moving in an unpredictable fashion in a two dimensional working space. The working space is cluttered with convex polygonal obstacles. The robot is a simple car-like robot with the nonholonomic constraint. With reference to figure 1, the visibility line is the imaginary straight line constructed from the robot to the target. The visibility angle is the angle from the positive x-axis and the visibility line. The visibility angle is denoted by φ. The range between the robot and the target is denoted by r. The velocity ratio is defined as $k = v_R/v_T$, where v_R is the robots linear velocity and v_T is the targets linear velocity. In this paper, it is assumed that $k > 1$. The closing velocity is denoted by $\vec{v}_C$ and given by

$$\vec{v}_C = \vec{v}_T - \vec{v}_R \tag{1}$$

The closing velocity is a vector that can be resolved in different ways. One particularly interesting way is to use polar coordinates and resolve the closing velocity into components along and a cross the visibility line, where $\vec{v}_C^{\parallel}$ and $\vec{v}_C^{\perp}$ represent the tangential and normal components of $\vec{v}_C$, respectively. It is possible to prove that [21]

$$\begin{aligned} \vec{v}_C^{\parallel} &= \dot{r}\vec{U}^{\parallel} \\ \vec{v}_C^{\perp} &= r\dot{\varphi}\vec{U}^{\perp} \end{aligned} \tag{2}$$

where $\vec{U}^{\parallel}$ and $\vec{U}^{\perp}$ represent the unit vectors along and across the visibility line, respectively. The robots equations of motion are given by

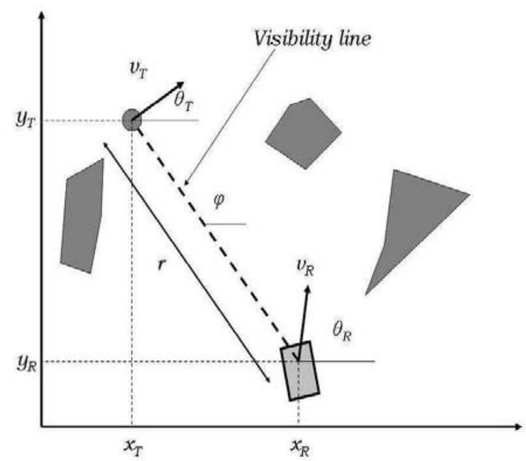

Fig. 1. Geometry of the tracking problem

$$\begin{aligned} \dot{x}_R &= v_R\cos(\theta_R)\cos(\phi_R) \\ \dot{y}_R &= v_R\sin(\theta_R)\cos(\phi_R) \\ L\dot{\theta}_R &= v_R\sin(\phi_R) \\ \dot{\phi}_R &= \omega_R \end{aligned} \tag{3}$$

where (x_R, y_R) represent the position of the robot's reference point in the inertial frame of reference, θ_R is the robot's orientation angle with respect to the x-axis and ϕ_R is the robot steering angle. ω_R represents the angular velocity of the robot. The configuration of the robot is given by $q_R = (x_R, y_R, \theta_R, \phi_R)$. The target moves according to the following kinematics

$$\begin{aligned} \dot{x}_T &= v_T\cos(\theta_T) \\ \dot{y}_T &= v_T\sin(\theta_T) \end{aligned} \tag{4}$$

where (x_T, y_T) represent the position of the target reference point in the inertial frame of reference $\{W\}$. Note that the visibility angle is given in terms of the robot and target positions as follows

$$\tan\varphi = \frac{y_T - y_R}{x_T - x_R} \tag{5}$$

The closing velocity $\vec{v}_C$ allows to write the relative kinematics equations in polar coordinates. In polar formulation, the decomposition of the relative velocity gives the relative kinematics equations between the robot and the target as follows

$$\begin{aligned} \dot{r} &= v_T\cos(\theta_T - \varphi) - v_R\cos(\theta_R - \varphi) \\ r\dot{\varphi} &= v_T\sin(\theta_T - \varphi) - v_R\sin(\theta_R - \varphi) \end{aligned} \tag{6}$$

This model represents a two-dimensional nonlinear system of differential equations. The analytical solution is not possible in general. However it is possible to solve for (r, φ) numerically. The first equation in (6) gives the relative range. A negative value for r means that the robot and the target are approaching from each other. The second equation gives the rate of turn of the robot with respect to the target. Constant value for the visibility angle means that the motion of the target seen by the robot is a straight line. The boundary conditions associated with system (6) are as follows

$$\begin{array}{ll} r(t_0) = r_0; & \varphi_(t_0) = \varphi_0 \\ r(t_f) = c; & \theta_R(t_f) \texttt{ is free} \\ \theta_R(t_0) = \theta_{R_0}; & v_R \texttt{ and } v_T \texttt{ are free} \\ \varphi(t_f) \texttt{ is free;} & \theta_T \texttt{ is free} \end{array} \quad (7)$$

where c is a constant number ($c \simeq 0$ for interception), t_0 is the initial time and t_f is the final time. Note that, the pure rendezvous is characterized by a constant value of the visibility angle. The control law is discussed in the next section. We assume the following

- The robot is faster than the moving target.
- The minimum turning radii for the robot and the target are equal.

III. The control law

A. *Reaching the target*

The goal here is to design a control law for the robots orientation angle that allows to track and reach the target with a desired final configuration. This is a real time problem, and thus the control input must be calculated online. This problem is substantially different from classical navigation problems. The most classical approach used to track a moving target is based on the pure pursuit, where the robot follows the path of the target, thus $\theta_R = \varphi$. Many animals predators use the pure pursuit to catch their prey. This approach is quite simple and works well; however, in most cases it is not optimal and results in longer interception times. We will see that the pure pursuit is just a particular case of the suggested method. The goal of the suggested law is to drive the visibility angle to a value for which the range is a decreasing function. This principle is quite similar to the pure pursuit. However it will result in various optimal paths. The navigation law must satisfy certain conditions. The most important condition is that the equation for the visibility angle rate has an asymptotically stable equilibrium that results in a decreasing range. We suggest to use the following form for the control law

$$\theta_R = \varphi + A\varphi^3 + C_i exp(-bt) + C_f \quad (8)$$

where $A > 0$ is a real number called the navigation factor, C_i and C_f are navigation parameters satisfying certain conditions. C_f is a deviation term. b is a positive gain. The term $C_i exp(-bt)$ is a heading regulation term that goes to zero with time as it can be seen from equation (8). The path of the robot can be controlled by acting on the values of $A; C_i; C_f$. Note the existence of the following particular cases:

- The navigation function given in (8) becomes equivalent to the pure pursuit when $A = 0; C_f = 0$. This case corresponds to $\varphi \to 0$.
- In the case when $A = 0; C_f \neq 0$, we have a deviated pursuit.

The equations for the pure pursuit are give by

$$\begin{array}{l} \dot{r} = v_T \cos(\theta_T - \varphi) - v_R \\ r\dot{\varphi} = v_T \sin(\theta_T - \varphi) \end{array} \quad (9)$$

In this particular case, it is obvious that $\dot{r} < 0$ and therefore the range is a decreasing function since $k > 1$. The following result deals with the general case.

Proposition

Under the control law given by

$$\theta_R = \varphi + A\varphi^3 \quad (10)$$

The robot reaches the moving target successfully.

Proof

$$\begin{array}{l} \dot{r} = v_T \cos(\theta_T - \varphi) - v_R \cos(A\varphi^3) \\ r\dot{\varphi} = v_T \sin(\theta_T - \varphi) - v_R \sin(A\varphi^3) \end{array} \quad (11)$$

The asymptotically stable equilibrium solutions for the line of sight angle is given by

$$\varphi^3 = \frac{1}{A} \arcsin(\frac{v_T}{v_R} \sin(\theta_T - \varphi)) \quad (12)$$

By replacing the equilibrium position in the range equation, we get

$$\dot{r} = v_T \cos(\theta_T - \varphi) - v_R \cos(\arcsin(\frac{v_T}{v_R} \sin(\theta_T - \varphi))) \quad (13)$$

which implies that the range rate is negative, and therefore the distance is decreasing until the robot reaches the target.

B. *Tracking with constant distance*

In many classic surveillance problems, the agent (robot) tracks a moving target without catching it. In this case, the robot keeps a constant distance from the target. The goal here is to derive a control law for the robot speed to accomplish this task. Clearly, a constant speed corresponds to $\dot{r} = 0$. Under the cubic navigation law, this corresponds to

$$v_T \cos(\theta_T - \varphi) = v_R \cos(A\varphi^3 + C_f) \quad (14)$$

which gives for the robot speed

$$v_R = \frac{v_T cos(\theta_T - \varphi)}{\cos(A\varphi^3 + C_f)} \quad (15)$$

IV. Collision avoidance

In the presence of obstacles, the problem of interception tracking becomes a difficult problem. In this case, the control law must combine global path planning to reach/track the moving goal, and local path planning that consists of collision avoidance. Thus the robot moves in two modes: The collision avoidance mode and the target tracking mode. Clearly, the collision avoidance mode has the priority over the target tracking mode. The collision cone approach is used to avoid collision with obstacles. The kinematics equations between the robot and obstacle B_i are given by

$$\begin{array}{l} r_i = -v_R \cos(\theta_R - \varphi_i) \\ r_i\varphi_i = -v_R \sin(\theta_R - \varphi_i) \end{array} \quad (16)$$

The first equation allows to determine whether the robot is approaching from the obstacle. In order to avoid collision, the robot first constructs the collision diagram based on its sensory information. The collision diagram shows free directions and directions corresponding to obstacles.

The robot changes its path when a collision is detected. This is achieved by changing the values of A and C_f . The smoothness of the path is an important aspect that must be taken into consideration. Let t_1 be the time when the robot begins changing its orientation angle. At this time, just before deviation, the robots orientation angle is given by

$$\theta_{R1}(t_1) = \varphi(t_1) + A\varphi^3(t_1) + C_{i1}exp(-b(t_1 - t_0)) + C_{f1} \quad (17)$$

The robot begins deviating by changing the values of its parameters (or one of them). Thus the new value of θ_R at time t_1 is given by

$$\theta_{R2}(t_1) = \varphi(t_1) + A\varphi^3(t_1) + C_{i2}exp(-b(t_1 - t_0)) + C_{f2} \quad (18)$$

Clearly, the smoothness of the path requires that

$$\theta_{R1}(t_1) = \theta_{R2}(t_1) \quad (19)$$

The values of the navigation and deviation parameters are chosen so that (19) is satisfied. The new deviation parameters are used to reach the deviation point. After the robot reaches the deviation point, it goes back to the tracking-interception mode. The problem that may be posed here is to find the optimal deviation. Since we are dealing with a tracking interception problem. It is necessary to consider the path of the target.

V. Simulation

A. Illustration of the deviated pursuit

In the case of the pure pursuit, the robot orientation angle is equal to the visibility angle, that is after the heading regulation phase, we have

$$\theta_R = \varphi \quad (20)$$

It can be easily proven that $\varphi \to \theta_T$, and therefore $\theta_R \to \theta_T$. The deviated pursuit is characterized by

$$\theta_R = \varphi + C_f \quad (21)$$

where C_f indicates the direction of the deviation (lead or lag). Two examples illustrating the deviated pursuit are shown in our simulation. The target performs a sinusoidal motion with a speed $v_T = 2m/s$. The robots speed is $v_R = 3m/s$. The robots and targets initial positions are (1,1) and (12,12), respectively. The initial orientation of the robot is 90^o. Figure 2 shows a lag pursuit with $C_f = 50^o$, while figure 3 shows a lead pursuit with $C_f = -50^o$.

B. Illustration of the cubic navigation function

Different values of A and C_f are used by the robot to track the target. Again the robots initial position is (1,1), and the targets initial position is (12,12). The initial orientation of the robot is 90^o. Four scenarios are shown in figures 4-7. In all cases the robot reaches the goal successfully.

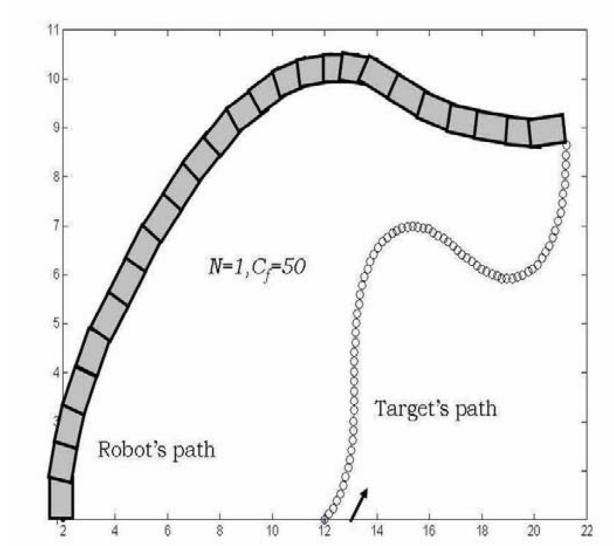

Fig. 2. Tracking using the deviated pursuit law with $C_f = 50^o$

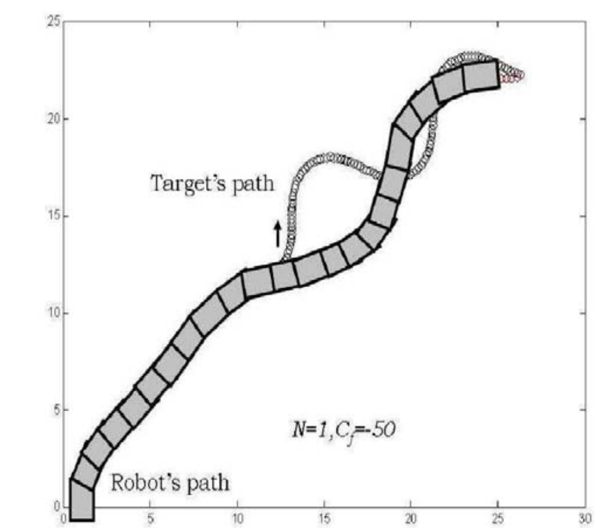

Fig. 3. Tracking using the deviated pursuit law with $C_f = -50^o$

VI. Conclusion

In this paper, we presented a method for robot tracking interception of a target moving in a two-dimensional working space with unknown maneuvers. An algorithm that allows the robot to tune the navigation parameters is suggested. The method uses cubic navigation functions, which belong to the family of nonlinear navigation functions. These functions are based on the combination of the kinematics and geometric rules. The problem of trackinginterception is considered for maneuvering targets, where the target frequently changes its orientation angle or speed. In the presence of obstacles, the

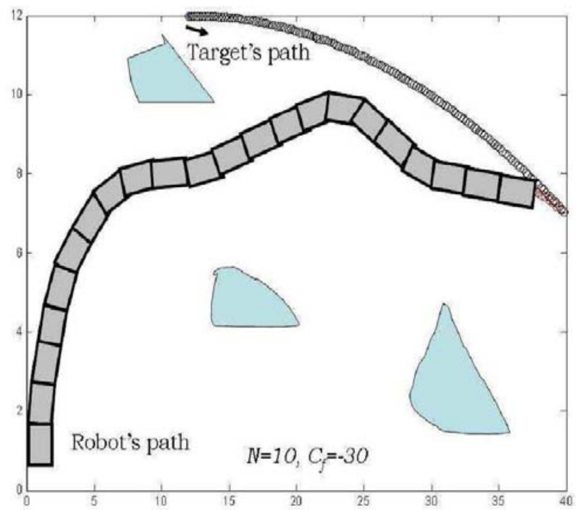

Fig. 4. Tracking using cubic navigation law with $A = 10, C_f = -30^o$

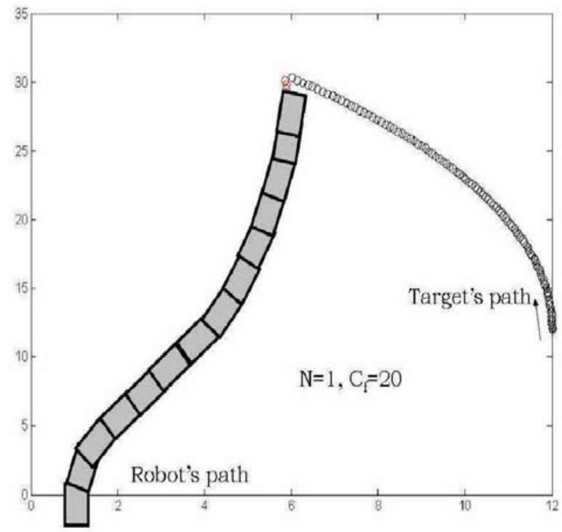

Fig. 6. racking using cubic navigation law with $A = 2; C_f = 20^o$

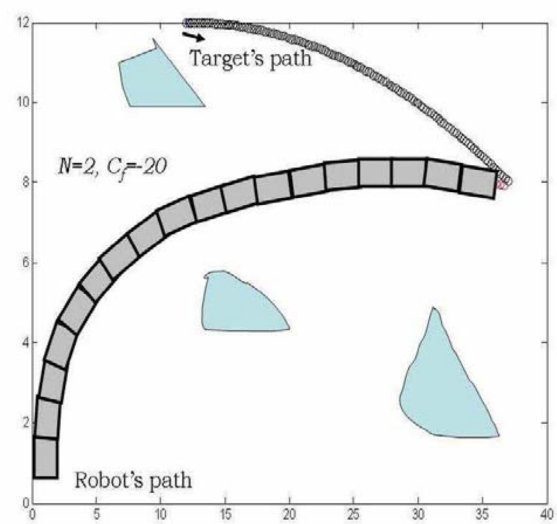

Fig. 5. Tracking using cubic navigation law with $A = 2; C_f = -20^o$

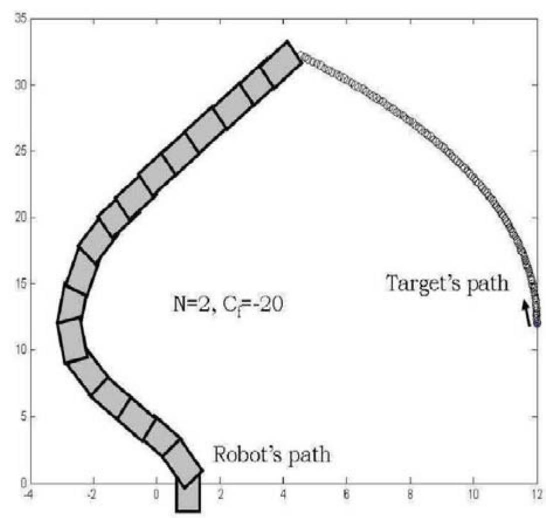

Fig. 7. Tracking using cubic navigation law with $A = 2, C_f = 20^o$

method is combined with a collision avoidance algorithm. These navigation functions depend on various external parameters. This property allows to change the robots path in realtime. The richness of the method is illustrated using an extensive simulation.

REFERENCES

[1]

[2] M. K. McBeath, D. M. Shaffer, and M. K. Kaiser, How baseball outfielders determine where to run to catch fly balls, Science 28 April 1995: Vol. 268. no. 5210, pp. 569 - 573

[3] H. Piccardo and G. Honderd, A new real-time algorithm for robotic manipulator path planning to enable the grasping of moving targets: some design aspects, Proceedings International Conference on Control, Edinburgh, March, 795-800 (1991).

[4] E. A. Croft, R. G. Fenton and B. Benhabib. An on-line robot planning strategy for target interception Journal of Robotic Systems, 13:97-114, 1997.

[5] M. Mehrandezh, M. Sela, R. Fenton, and B. Benhabib, Proportional navigation guidance for robotics interception of moving objects, Journal Of Robotics Systems, vol. 17, no. 6, pp. 321340, 2000.

[6] J. Kovecses, W.L. Cleghorn and R.G. Fenton. Dynamic Modeling and Analysis of a Robot Manipulator Intercepting and Capturing a Moving Object, with the Consideration of Structural Flexibility. Multibody System Dynamics, 3:137-162, 1999.

[7] 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, 864-869, 1999.

[8] Armel Cretual and Francois Chaumette, Application of Motion-Based Visual Servoing to Target Tracking, The International Journal of Robotics Research, 20:878-890, (2001).
[9] L. Freda and G. Oriolo, Vision-based interception of a moving target with nonholonomic mobile robot, Robotics and Autonomous Systems, 55:419-432 (2007).
[10] A. Yamaguchi, A Cooperative Hunting Behavior by Mobile-Robot Troops The International Journal of Robotics. 18: 931-940, (1999).
[11] Fethi Belkhouche, B. Belkhouche and P. Rastgoufard, Multi-robot hunting behavior, IEEE International Conferences on Systems, Man and Cybernetics, Hawaii, October 2005 2299-2304, (2005).
[12] 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, 83-88.
[13] L. Davis, V. Philomin and R. Duraiswami. Tracking humans from a moving platform. Proceedings of IEEE International Conference on Pattern recognition, 171-178.
[14] S. Sachs, S.M. LaValle and S. Rajko, Visibility-Based Pursuit-Evasion in an Unknown Planar Environment, The International Journal of Robotics Research, Vol. 23, No. 1, 3-26 (2004).
[15] T. Sugar, M. McBeath, A. Suluh and K. mundhra, Mobile robot interception using human navigational principles: Comparison of active versus passive tracking algorithms, Autonomous Robots, 43:43-54, (2006).
[16] Jongho Lee, J. Mikael Eklund, and Shankar Sastry, Group Pursuit Evasion Games of Unmanned Underwater Vehicles, BEARS 2006
[17] J. Mikael Eklund, Jonathan Sprinkle, S. Shankar Sastry, Implementing and testing a nonlinear model predictive tracking controller for aerial pursuit evasion games on a fixed wing aircraft Proceedings of American Control Conference, June, 1509-1514 (2005).
[18] Rene Vidal, Omid Shakernia, H. Jin Kim, David Hyunchul Shim, Shankar Sastry, Probabilistic pursuit-evasion games: theory, implementation and eperimental evaluation, IEEE Transactions on Robotics and Automation,18: 662-669, (2002).
[19] G. Rote, Pursuit-evasion with imprecise target location, Proceedings of the fourteenth annual ACM-SIAM, Baltimore, 747-753.
[20] V. Isler, C. Belta, K. Daniilidis and G. Pappas, Hybrid control for visibility-based pursuit-evasion games. Proceedings of IEEE/RSJ International Conference on Intelligent Robotics Systems, September, Sendai, 1432-1438, (2004).
[21] Brian P. Gerkey Sebastian Thrun Geoff Gordon, Visibility-based pursuit-evasion with limited field of view, Proceedings of the National Conference on Artificial Intelligence, San Jose, 20-27, (2004).
[22] Fethi Belkhouche, B. Belkhouche and P. Rastgoufard, Line of sight robot navigation towards a moving goal, IEEE Transactions on Systems, Man and Cybernetics, Part B. 36:255-267, (2006).
[23] V. Isler, S. Kannan, and S. Khanna, Randomized pursuit-evasion with local visibility, Randomized pursuit-evasion with local visibility, 20:26-41, (2006).
[24] K. Bendjilali and Fethi Belkhouche, Kinematics-based navigation functions, Advanced Robotics, Volume 22, Number 11, pp. 1243-1264, 2008.