Towards Real-Time Autonomous Target Area Protection: Theory and Implementation

Jitesh Mohanan[1], S.R. Manikandasriram[2], R. Harini Venkatesan.[1], and B. Bhikkaji[1], *Member, IEEE*

***Abstract*—This paper considers the Target Guarding Problem (TGP) with a single pursuer P, a single evader E and a stationary target T. The goal of P is to prevent E from capturing T, by intercepting E as far away from T as possible. An optimal solution to this problem, referred to as Command to Optimal Interception Point (COIP), was proposed recently. This guidance law in its current form is not amenable for a real time implementation, *i.e.,* cannot be easily implemented in most commonly available robotic hardware. Here the TGP is revisited and the optimal solution is reformulated to expressions that can be computed faster. These expressions allow for seamless real-time implementation in robotic hardware. Moreover, the reformulation enables the optimal solution to be coded as a lookup table to further increase the speed of computations. An experimental set up with mobile robots and a motion capture system is then used to validate the claims. The case of T lying in E's dominance region is considered a lost game for P. However, this is true only if E plays optimally. If E plays sub-optimally P stands a chance to win the game. This case, which has not been analyzed earlier, is also discussed in this paper, and an optimal strategy for P is presented.**

***Index Terms*—Target protection games; Pursuit-evasion games; Region of Dominance; Geometric approaches; Multi-agent systems; Autonomous mobile robots;**

I. Introduction

THIS paper studies optimal strategies for agents in the Target Guarding Problem (TGP). The problem chosen involves two mobile agents: the *pursuer* and the *evader*. Both the pursuer P and the evader E move on a two dimensional planar surface, as shown in Figure 1. The game also features a target zone, T, which is guarded by the pursuer P. The goal of the evader is to capture the target. The goal of the pursuer is to guard the target against an attack by the evader. Thus, the pursuer aims to capture the evader before the target is attacked. It is usually desired that the pursuer intercepts the evader as far away from the target as possible. The aim is to derive an optimal strategy (guidance law) for the pursuer to intercept the evader in *real-time*.

Here, a geometrical approach is taken for determining an optimal guidance law for the pursuer. The dominance region of the pursuer constitutes the set of points in the play area that can be reached by the pursuer before the evader. Alternately, the dominance region of the evader is the set of points in the play area that can be reached by the evader before the pursuer. If the target is present in the dominance region of the pursuer, the game is won by the pursuer using the optimal strategy.

An optimal strategy for the pursuer was presented in [1]. It was referred to as Command to Optimal Interception Point (COIP) guidance law. Though this optimal strategy ensures the victory of the pursuer P when the target is in P's dominance region, it cannot be easily implemented in real time. The guidance law has to be reworked to enable a real-time implementation. In situations where the target T lies in the evader E's dominance region, the game is considered lost, assuming E plays optimally. If the evader plays sub-optimally the pursuer must take advantage of this and maximize his chances of winning.

The key focus of this paper is to derive an optimal strategy for the pursuer that is implementable in *real-time*. The optimal guidance law is reformulated to get to a form that can be coded as a lookup table, enabling an effective real-time implementation. The lookup table based optimal guidance law is then experimentally validated using mobile robots. The case of the target lying in the evader's dominance region is also investigated and the optimal strategy for the pursuer is derived in such a case.

This paper is organized as follows. In Section II problems that are closely related to TGP and the solutions presented in existing literature are discussed. In Section III existing solutions to the TGP are presented. In Section IV, the expressions for the Command to Optimal Interception Point (COIP) guidance law derived in [1] are presented. In Subsections V-A and V-B the problem statement and the contributions of this paper are made clear. The COIP law is revisited and expressions that lend themselves useful for a real time implementation are presented in Section VI. Simulations comparing the original COIP, the reformulated COIP and its lookup table implementation are presented in Section VII. Section VIII presents experiments that validate the reformulated COIP and the corresponding lookup table implementation. In Section IX the case where the target T lies in the evader E's dominance region is analyzed and an optimal guidance law for the pursuer P is derived. The paper concludes with Section X discussing the results and possible future extensions.

II. Motivation and Related work

The problem of protecting a target from an attacker using an autonomous defender, is of interest in many areas of defense and robotics. In particular, this problem finds applications in border patrolling and military combat operations [2], missile-aircraft engagements [3] and vaccine design [4].

[1] Department of Electrical Engineering, Indian Institute of Technology Madras, India, 600036 e-mail: jiteshnov1@gmail.com

[2] Robotics Institute, University of Michigan, Ann Arbor, USA 48105 e-mail: srmanikandasriram@gmail.com

This work was partly sponsored by Indian Institute of Technology Madras.

Manuscript received April xx, 2017; revised September xx, 2017.

The TGP was first studied by Isaacs [5] in his pioneering work on pursuit evasion games. Thereafter, research leading to real-time solutions for such games has been minimal. The primary reason is that most of these games have been modeled as differential games, the solution for which are the saddle point equilibria of the Hamilton-Jacobi-Isaacs equation [6]. However, determining the equilibrium points, especially if there are more than one, in a real time scenario would be computationally infeasible. More importantly, the solution of the differential game is not guaranteed to exist or be smooth everywhere on the state-space of the game [5].

Games similar to the TGP have also been of interest in several other studies. In [7], a variant of the TGP is analysed, where the evader does not have information about the position or velocity of the pursuer, but has a better maneuverability than the pursuer. However, the pursuer has complete information about the evader and the environment. In [8], [9] and [10], the TGP is analyzed with the special case that the pursuer and the target co-operate and form joint strategies for protection from the evader. In [11] and [12], the authors analyze line-of-sight algorithms for defending the target against the evader. A pursuer following this algorithm will always keep his heading-direction towards the evader to ensure interception. This, being a greedy algorithm will not guarantee capture of the evader either in minimal time or at maximum distance from the target. The authors in [3] analyze the case with a moving target, which is also an active player in the game. In a limiting case, it is shown that this game is similar to a game played only by the target and the evader. All of the studies mentioned above discuss variants of the TGP from a theoretical perspective. The guidance laws presented do not lend themselves to a seamless real-time implementation. Moreover, most of these studies were done for defense scenarios involving missile and aircraft engagements, the dynamics of which are different from autonomous ground robots protecting a target of interest.

A real-time strategy for the pursuer in the TGP has been presented in [13], which uses the Rapidly Exploring Random Trees (RRT) algorithm. The optimality of the solution obtained from this algorithm depends on the number of nodes generated for the random tree. Thus, deriving an optimal path for the pursuer using RRT increases computational complexity. Simulation results therein suggest that the time taken to generate the near-optimal path for the pursuer (which is of the order of a few seconds) restricts the use of the algorithm to slow agents.

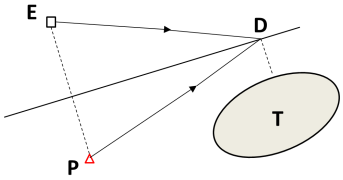
In addition, most of the algorithms reported so far requires the knowledge of the instantaneous positions of the pursuer and the evader. As will be seen later, the algorithm proposed in this paper generates an optimal path, that does not require the instantaneous positions of the agents, but only on the ratio of their relative distances from each other and the target.

In summary it appears that

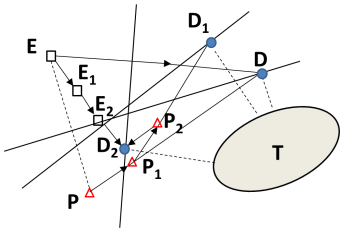
- there has been very little reporting on optimal strategies (for the pursuer in the TGP), that are implementable in real-time, and
- the pursuer's strategy in the case of the target lying in the evader's dominance region has not been dealt with in the literature so far.

The work in this paper attempts to fill the above mentioned gaps in the literature.

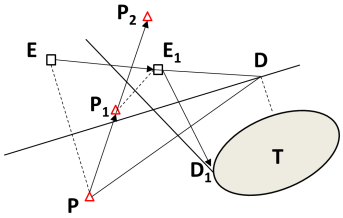
III. The Target Guarding Problem: Existing Solutions



(a) P and E play optimally



(b) P plays optimally, but E does not



(c) E plays optimally, but P does not

Fig. 1. Target Guarding Problem from [5].

The TGP features the pursuer P, the evader E, and the target area to be protected T, shown in Figure 1. The objective of E is to capture T, and that of P is to capture E, before E captures T. Isaacs [5] considered the special case of the velocity of E being equal to that of P, *i.e.*, $v_e = v_p$. For this case, a comparison of the optimal strategies for P and E are shown in Figure 1. As $v_e = v_p$, the perpendicular bisector of the line joining the initial positions of P and E divides the space into regions where P or E can reach *before* the other. In Figure 1, T lies in the region where P can reach *before* E. In other words, T lies in P's dominance region. Therefore, P will be able to protect T. But what would be P's *optimal strategy* to protect T, and what would be E's, *knowing* that he would be captured by P ? The optimal strategy proposed by Isaacs, for either players, is to head straight to the point D, which is the closest point to the target area T on the perpendicular bisector. If either of them deviates from this optimal strategy, it benefits their opponent. In Figure 1(b), E takes a different path (E - E_1 - E_2 - D_2) than the optimal one (E - D). P always takes the optimal path, *i.e.*, towards the point on the perpendicular bisector of PE, closest to the target. Thus, when E moves to E_1, P moves to P_1. The point closest to the target on the perpendicular bisector of P_1E_1 is now D_1 and P heads towards D_1. E continues on its initial path and gets to E_2,

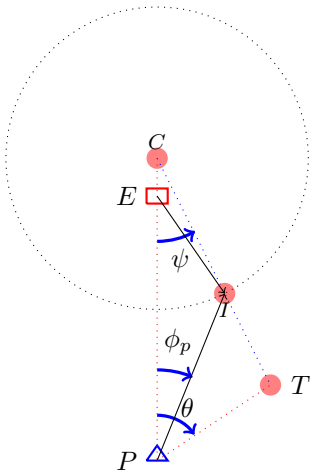


Fig. 2. Optimal strategies for P and E, for a static point target at T ($v_p > v_e$) [14]

while P gets to P_2. This continues till E gets intercepted by P at D_2. The interception point in this case is thus at a longer distance from the target than in Figure 1(a). Thus, the more E plays sub-optimally, the farther he will be from the target during interception. In Figure 1(c), P takes a different path, eventually resulting in E being able to capture T. It is to be noted that the strategies mentioned so far are only true in the case of the target lying in the pursuer's dominance region. If the target lies in the evader's dominance region, the optimal path for the evader will be to head straight to the target. The optimal path for the pursuer in such a case is analyzed in a later section.

The authors in [14] extended the analysis of the TGP studied by Isaacs to the case when $v_e \neq v_p$. It was shown that the target cannot be protected when $v_p < v_e$, and for certain initial conditions. When $v_p > v_e$, the capture of E by P would happen on the Apollonian circle corresponding to P's and E's initial positions and the ratio of velocities $\frac{v_e}{v_p}$. The circle is shown using a dotted line in Figure 2, and separates the space into regions where one player can reach before the other, when they head straight to the same point. E can reach all points inside the circle faster than P, and therefore, it is E's *dominance region*. Similarly, P's dominance region lies outside the circle. Since $v_p > v_e$, E's dominance region is smaller than P's. As the target, denoted by T lies in P's dominance region, the target can be protected, and the optimal strategies for P and E in this case is to head straight to the point I, which is the point closest to T on the Apollonian circle. Based on this, the authors also proposed a new guidance law for the pursuer called the Command to Optimal Interception Point (COIP) guidance law [1], which will direct the pursuer to always pursue the point I.

IV. The Command to Optimal Interception Point (COIP) Guidance Law for a Two-Dimensional Scenario

In Figure 2, let P, E and T denote the instantaneous locations of the players, and O be the origin of the coordinate system describing the space in which the game is being played. Therefore, $\overrightarrow{OP}$, $\overrightarrow{OE}$ and $\overrightarrow{OT}$ are their position vectors respectively. Also, let $\frac{v_e}{v_p} = \frac{e}{p}$. Through the guidance law, the aim is to calculate $\angle EPI = \phi_P$, the heading angle off $\overrightarrow{PE}$, as shown in the figure.

For the TGP in two-dimensions, the following steps constitute the COIP guidance law [1]:

$$\overrightarrow{PE} = \overrightarrow{OE} - \overrightarrow{OP} \quad (1)$$

$$\overrightarrow{PT} = \overrightarrow{OT} - \overrightarrow{OP} \quad (2)$$

$$\overrightarrow{PC} = \frac{1}{1-\frac{e^2}{p^2}}\overrightarrow{PE} \quad (3)$$

$$R = \frac{e}{p}|\overrightarrow{PC}| \quad (4)$$

$$\overrightarrow{CT} = \overrightarrow{PT} - \overrightarrow{PC} \quad (5)$$

$$\overrightarrow{CI} = R\frac{\overrightarrow{CT}}{|\overrightarrow{CT}|} \quad (6)$$

$$\overrightarrow{PI} = \overrightarrow{PC} + \overrightarrow{CI} \quad (7)$$

Once the vectors $\overrightarrow{PI}$ and $\overrightarrow{PC}$ are determined, the angle between the two vectors (the required heading angle for the pursuer) is given by:

$$\phi_P = \tan^{-1}\left(|\overrightarrow{PI} \times \overrightarrow{PC}|, \overrightarrow{PI}.\overrightarrow{PC}\right) \quad (8)$$

Equation (8) involves four operations–a cross product, a norm, a dot product, and a $\tan^{-1}$, and hence, takes the most time to compute.

The speed of calculating the COIP guidance law can thus be greatly improved by computing (1)-(8) by a single lookup table and eliminating the operations of the cross product and the dot product. Moreover, it would be difficult to construct a lookup table for the COIP guidance law in its current form, due to the fact that the distances $|\overrightarrow{PE}|$ and $|\overrightarrow{PT}|$ can take on arbitrarily large values. It can be inferred from Figure 2 that, if both $|\overrightarrow{PE}|$ and $|\overrightarrow{PT}|$ are scaled by the same constant, ϕ_P would remain the same. Thus, instead of dealing with $|\overrightarrow{PE}|$ and $|\overrightarrow{PT}|$ separately, if (8) can be calculated with a single variable given by the ratio $\frac{|\overrightarrow{PT}|}{|\overrightarrow{PE}|}$, tabulating the optimal heading direction would result in a smaller lookup table that greatly reduces the redundancy. Fortunately, this is permitted by the dynamics of the target-guarding game, the expressions for which are derived in the Section VI.

V. The Problem Statement and Contributions

A. The Problem Statement

This paper addresses two problems that form part of the *real-time* solution for the TGP:

- Find the optimal heading direction or the optimal guidance law (COIP) for the pursuer to intercept the evader before the evader captures the target. The COIP should be found out in real time.
- Find out the optimal strategy for the pursuer when the target is located in the evader's dominance region.

For all the cases, it is assumed that the initial positions of the pursuer and the evader are known, and both of them move with constant but unequal velocities. Both the pursuer and evader have all information pertaining to the environment and each other.

B. Original Contributions

The original contributions of the paper are as follows:

- A new algorithm for computing the optimized COIP law which is amenable for implementation in real-time.
- A lookup table strategy for the optimized COIP law, enabling a faster real-time implementation.
- Experimental verification of the optimized COIP law on a test bed consisting of mobile robots and a motion capture system.
- The optimal strategy of the pursuer when the target lies in the evader's dominance region.

Each of the contributions of this paper solves a specific part of the problem statement of interest, and thus are enablers for a real-time implementation of the solution to the TGP.

VI. Reformulating the COIP Guidance Law

As described in Section IV, the objective of this exercise is to find an expression for ϕ_P which would be computationally faster and allow for easier fetching of data from the lookup table. A unique value for ϕ_P can be determined, given the following three variables:

$$k_0 = \frac{|\overrightarrow{PT}|}{|\overrightarrow{PE}|} \quad (9)$$

$$k_1 = \frac{e}{p} \quad (10)$$

$$\theta = \angle EPT \quad (11)$$

Equation (9) determines how far the target is, from the pursuer as compared to the evader, (10) determines the size of the dominance region of the evader, and (11) is the angle between the lines PE and PT, as shown in Figure 2.

The expression of $\phi_P = f(k_0, k_1, \theta)$ is derived as follows:

Let the magnitude of vectors be represented without the arrow on the top, for example, $|\overrightarrow{PC}| = PC$, $|\overrightarrow{PE}| = PE$ and so on. Using (3), (4) and (10), PC can be written as

$$PC = \frac{PE}{1-k_1^2} \quad (12)$$

$$R = k_1 PC = \frac{k_1}{1-k_1^2} PE \quad (13)$$

In $\triangle CPI$, using the sine rule,

$$\begin{aligned} \frac{PI}{\sin\psi} = \frac{CI}{\sin\phi_P} &= \frac{R}{\sin\phi_P} \\ \implies \sin^2\phi_P &= \frac{R^2}{PI^2}\sin^2\psi \\ \implies \sin^2\phi_P &= \frac{\sin^2\psi}{\left(\frac{PI}{R}\right)^2} \end{aligned} \quad (14)$$

In $\triangle CPI$, using the cosine rule,

$$\begin{aligned} PI^2 &= CI^2 + PC^2 - 2.CI.PC.\cos\psi \\ &= R^2 + PC^2 - 2.R.PC.\cos\psi \end{aligned} \quad (15)$$

Dividing (15) by R^2, it is seen that

$$\left(\frac{PI}{R}\right)^2 = 1 + \left(\frac{PC}{R}\right)^2 - 2\left(\frac{PC}{R}\right)\cos\psi \quad (16)$$

Using (12) and (13),

$$\frac{PC}{R} = \frac{1}{k_1} \quad (17)$$

and therefore, (16) becomes

$$\left(\frac{PI}{R}\right)^2 = 1 + \frac{1}{k_1^2} - \frac{2}{k_1}\cos\psi \quad (18)$$

Substituting (18) in (14),

$$\begin{aligned} \sin^2\phi_P &= \frac{\sin^2\psi}{1+\frac{1}{k_1^2}-\frac{2}{k_1}\cos\psi} \\ &= \frac{k_1^2\sin^2\psi}{k_1^2 - 2k_1\cos\psi + 1} \\ &= \frac{k_1^2\sin^2\psi}{k_1^2 - 2k_1\sqrt{1-\sin^2\psi} + 1} \end{aligned} \quad (19)$$

In $\triangle CPT$, using the sine rule,

$$\begin{aligned} \frac{CT}{\sin\theta} &= \frac{PT}{\sin\psi} \\ \implies \sin^2\psi &= \left(\frac{PT}{CT}\right)^2\sin^2\theta \\ \implies \sin^2\psi &= \frac{\sin^2\theta}{\left(\frac{CT}{PT}\right)^2} \end{aligned} \quad (20)$$

In $\triangle CPT$, using the cosine rule,

$$\begin{aligned} CT^2 &= PT^2 + PC^2 - 2.PT.PC.\cos\theta \\ \implies \left(\frac{CT}{PT}\right)^2 &= 1 + \left(\frac{PC}{PT}\right)^2 - 2\left(\frac{PC}{PT}\right)\cos\theta \end{aligned} \quad (21)$$

From (12),

$$\begin{aligned} \frac{PC}{PT} &= \frac{1}{1-k_1^2}\left(\frac{PE}{PT}\right) \\ &= \frac{1}{k_0(1-k_1^2)} \end{aligned} \quad (22)$$

by using (9). Let

$$k_2 = k_0(1-k_1^2). \quad (23)$$

Therefore, (22) becomes

$$\frac{PC}{PT} = \frac{1}{k_2} \quad (24)$$

and substituting (23) in (21),

$$\left(\frac{CT}{PT}\right)^2 = 1 + \frac{1}{k_2^2} - \frac{2}{k_2}\cos\theta \quad (25)$$

Substituting (25) in (20),

$$\begin{aligned}\sin^2\psi &= \frac{\sin^2\theta}{1+\frac{1}{k_2^2}-\frac{2}{k_2}\cos\theta} \\ &= \frac{k_2^2\sin^2\theta}{k_2^2-2k_2\cos\theta+1} \end{aligned} \quad (26)$$

Let

$$k_3 = k_2^2 - 2k_2\cos\theta + 1. \quad (27)$$

Therefore, (26) becomes

$$\sin^2\psi = \frac{k_2^2\sin^2\theta}{k_3} \quad (28)$$

Substituting (28) in (19),

$$\begin{aligned}\sin^2\phi_P &= \frac{k_1^2k_2^2\sin^2\theta}{k_3\left[k_1^2-2k_1\sqrt{1-\left(\frac{k_2^2\sin^2\theta}{k_2^2-2k_2\cos\theta+1}\right)}+1\right]} \\ &= \frac{k_1^2k_2^2\sin^2\theta}{k_3\left[k_1^2-\frac{2k_1|k_2\cos\theta-1|}{\sqrt{k_3}}+1\right]} \\ \implies \sin\phi_P &= \sqrt{\frac{k_1^2k_2^2\sin^2\theta}{k_3\left[k_1^2-\frac{2k_1|k_2\cos\theta-1|}{\sqrt{k_3}}+1\right]}} \\ \implies \phi_P &= \sin^{-1}\left(\sqrt{\frac{k_1^2k_2^2\sin^2\theta}{k_3\left[k_1^2-\frac{2k_1|k_2\cos\theta-1|}{\sqrt{k_3}}+1\right]}}\right) \end{aligned} \quad (29)$$

where both k_2 and k_3 are functions of (k_0, k_1, θ) given by (23) and (27) respectively. Although the expression of ϕ_P in (29) appears complex, it is of the form $\phi_P = f(k_0, k_1, \theta)$, using which values of ϕ_P were tabulated for different values of k_0, k_1 and θ. A sample plot from this table is shown in Figure 3.

Summarizing, the equations used for computing the optimal pursuer heading angle are given as:

$$\overrightarrow{PE} = \overrightarrow{OE} - \overrightarrow{OP} \equiv PE_x\hat{i} + PE_y\hat{j} \quad (30)$$

$$\overrightarrow{PT} = \overrightarrow{OT} - \overrightarrow{OP} \equiv PT_x\hat{i} + PT_y\hat{j} \quad (31)$$

$$k_0 = \frac{|\overrightarrow{PT}|}{|\overrightarrow{PE}|} \quad (32)$$

$$k_1 = \frac{e}{p} \quad (33)$$

$$k_2 = k_0(1-k_1^2) \quad (34)$$

$$\theta = \tan^{-1}(PE_y, PE_x) - \tan^{-1}(PT_y, PT_x) \quad (35)$$

$$k_3 = k_2^2 - 2k_2\cos\theta + 1 \quad (36)$$

$$\phi_P = \sin^{-1}\left(\sqrt{\frac{k_1^2k_2^2\sin^2\theta}{k3\left[k_1^2-\frac{2k_1|k_2\cos\theta-1|}{\sqrt{k_3}}+1\right]}}\right) \quad (37)$$

VII. Performance Comparison and Discussion

To test the computational speed, simulations with all the three implementations of the COIP guidance law were carried out in MATLAB. The first method involved calculating ϕ_P from the algorithm given by expressions (1) - (8) at each sampling instance. The second method involved calculating ϕ_P from the algorithm given by Equations (30) - (37). The third implementation involved looking up ϕ_P from the table of $\phi_P = f(k_0, k_1, \theta)$ described in the previous section. For the experiments, the table was constructed using 100 samples each for k_0, k_1 and θ for satisfactory precision, amounting to a total of 1000000 points for ϕ_P. The memory required to store this table was around 7 MB.

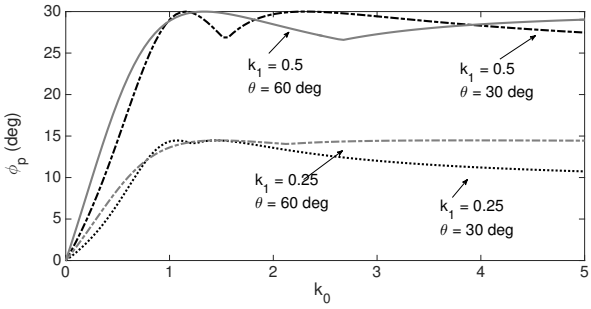


Fig. 3. Sample plot of ϕ_p for different values of k_0, k_1 and θ

The computational experiments involved 100 trials of recording the computational time for all the three different implementations of the COIP guidance law.

The experiments were conducted on a workstation employing an Intel Core i7 processor running at a speed of 3.4 GHz. The results are shown in Figure 4. It can be seen that the new implementation model has consistently lesser computation time compared to the original COIP implementation, and the lookup table implementation has an even lesser computation time. The average performance gain in terms of computation time for the two new implementations are shown in Figure 5. As it can be seen, the new algorithm is about 2.3 times faster than the original algorithm, and the lookup table method about 6.7 times faster. On an average, the lookup table method requires less than 0.05 ms to generate the optimal heading angle for the pursuer. The computational times were calculated using the 'tic' and 'tac' commands in Matlab.

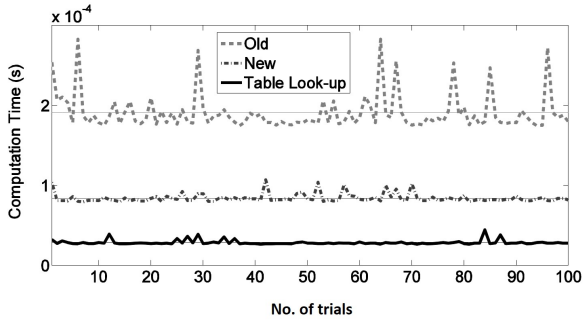


Fig. 4. Comparison of the computational times for the three different implementations of COIP guidance law over 100 trials

These improvements in computation time are significant, and therefore favor the use of the new algorithm for real-time applications. Also, the memory demands of storing a table, as suggested, can be met by most robotic systems hardware. Another advantage with the proposed new implementation is that, it only requires the ratio of the distances $k_0 = \frac{PT}{PE}$, and not PT and PE like the original algorithm. If the pursuer

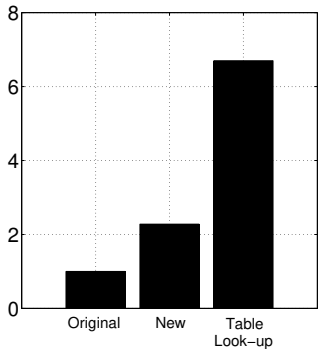


Fig. 5. Average performance gain in terms of computation time for the two new implementations of COIP guidance law over 100 trials

robot were to use a ranging system to detect the target and evader, the ratio k_0 can be calculated as the inverse of the times-of-flight of its ranging signal between the attacker and the target, *i.e.,*

$$k_0 = \frac{t_{PE_f}}{t_{PT_f}} = \frac{PT}{PE} \tag{38}$$

The same ranging system could also be used to calculate the value of θ. This means that it would be possible to calculate ϕ_P even in cases where a localization system is not available to provide the exact coordinates of E and T. This is in contrast to the implementations which require the coordinates of E and P to calculate ϕ_P.

It can also be seen that the maximum value of k_0 is finite. This is because, when using actual robots, the minimum value of PE is not zero, but is given by the distance between the centroids of the robots when interception takes place.

VIII. Experimental Validation of Modified COIP Law

Since simulations suggest that the computations of the alternate formulations of the COIP guidance law are faster, they were put to test in an experimental set-up. Two differential drive robots built using LEGO EV3 Mindstorms kits (Figure 8) were used as the pursuer and evader robots. The evader robot was manually controlled using a joystick while the proposed guidance law was used to autonomously control the pursuer robot. A stationary target with a predetermined location was used throughout the game.

The coordinates of the pursuer and evader robots were determined using a ceiling mounted motion capture system (Optitrack Flex3), with a total of six cameras providing real-time instantaneous information to a central server (Figure 9). The same motion capture system was used to determine the orientation of the individual robots. The algorithm for the guidance law was implemented in SIMULINK.

Recall that, the optimal strategies used in this work have been derived assuming a robot that can instantaneously move along any direction *i.e.,* a 2D holonomic robot. But most robots in real world are non-holonomic such as the differential drive configuration of the LEGO EV3 robots used in this study. These robots require additional time to orient themselves along the desired direction of movement. Here, a simple proportional controller is used for each robot as shown in Figure 6 to maintain the desired orientation of the robots.

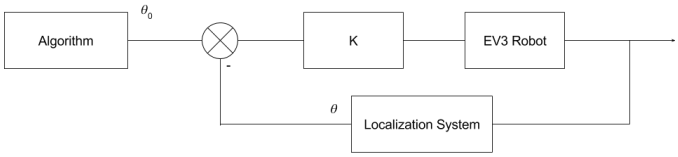


Fig. 6. A proportional controller for the robot to maintain desired orientation

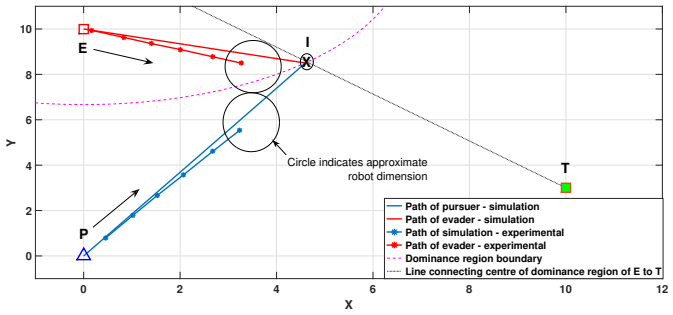


Fig. 7. *P* successfully intercepts *E* before *E* captures *T* - Experimental results

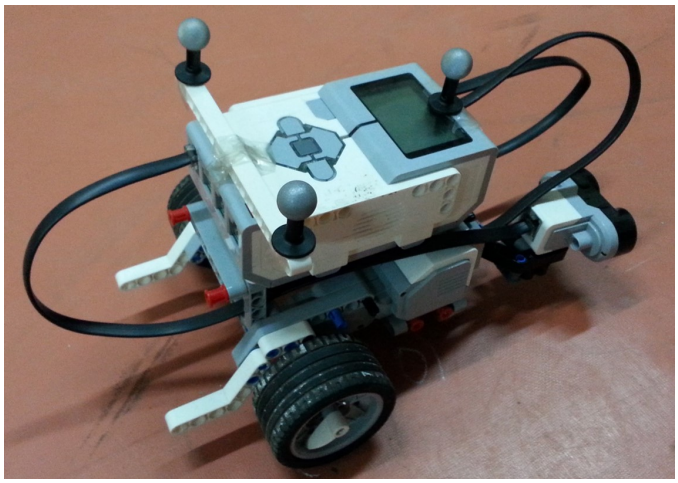


Fig. 8. The LEGO EV3 Robot

In order to eliminate the effect of the differential drive constraint on the performance of the algorithm, both the robots were set to face along their desired orientations, P along PI and E along EI. The velocity of the pursuer was set to be double that of the velocity of the evader. The result of the experiment is shown in Figure 7. The plot shows the simulated as well as the experimental trajectories of both the robots. The close match between both the sets of trajectories verifies that the proposed new algorithm successfully controls the pursuer agent autonomously to intercept the evader before the target is captured. In Figure 7, the experimental paths of both the robots stop short of meeting at point I. The reason for this is that the robots used for simulation studies are point robots, but the ones used for experiments have non-zero dimensions. Hence,

Fig. 9. The ceiling mounted motion capture system

Fig. 10. One out of six Flex3 cameras used in the motion capture system

interception occurs with the edges of the robot meeting, rather than their centroids meeting. The approximate dimensions of the robots are shown as circles in the figure.

A video demonstrating the above experiment can be seen at [15]

IX. Guarding a Target Located in the Evader's Dominance Zone

Most of the literature on TGP, starting with the pioneering work in [5], only consider the case of T being in P's dominance region. This is because if T were to be in E's dominance region (Figure 12), E can reach T before P could capture E, and thus, ending the game.

However, this is true only if E plays optimally throughout the game. If T were to be located in E's dominance region (within the circle), and if E wrongly estimates T to be outside it, P's strategy should capitalize on this mistake. This scenario is shown in Figure 11. If E operates based on this misinterpretation for sufficiently long, it might be possible for P to protect T, despite starting at a disadvantage. It is in view of this that the following section analyze what should P's strategy be, in order to take maximal advantage of E's mistakes, when T is in E's dominance region.

The analysis proceeds as follows: At any instant, let the positions of the players P, E and T be $(x_p, y_p), (x_e, y_e)$ and (x_t, y_t) respectively. Ideally, P would want to move in a

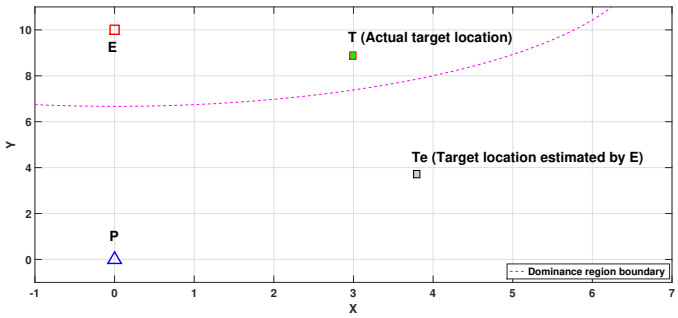


Fig. 11. E misinterprets the target to lie outside his dominance region

direction that makes the target T lie inside his dominance region. In other words, the objective of P is to minimize

$$\alpha = R - |\overrightarrow{CT}| \tag{39}$$

Let

$$\overrightarrow{\Delta P} = \Delta r \left(\cos\beta\hat{i} + \sin\beta\hat{j}\right) \tag{40}$$

be the vector that results out of P's instantaneous strategy.

Let $\frac{d}{d\beta}$ be denoted by '$'$'. The objective for P is to choose β to minimize (39), which means β should be chosen such that:

$$\alpha' = R' - \frac{\overrightarrow{CT}.\overrightarrow{CT}'}{|\overrightarrow{CT}|} = 0 \tag{41}$$

This results in P choosing a heading angle β, where

$$\tan\beta = \frac{y_p - y_i}{x_p - x_i} \tag{42}$$

Derivation of (42) involves a series of algebraic manipulations, and hence is presented in the Appendix.

Equation (42) implies that P should move towards the point I, which is the same strategy as the case when T is in P's dominance region. Note that, instead of pursuing T, E is now effectively pursuing I, because of the miscalculation of T's position on E's part. The value of β given by (42) also ensures minimization of α given in (39). This is because, geometrically, minimizing α is equivalent to ensuring that I is as close to T as possible. This is indeed the case, since C, T and I are collinear (Figure 12), and CI is the diameter of a circle (the actual dominance region of E). So an interesting result is obtained that the optimal strategy for the pursuer remains the same, irrespective of whether T is located inside or outside P's dominance region.

To test the effectiveness of the strategy, the target guarding game was simulated with the target now lying inside the evader's dominance region. In this case, under normal circumstances, the evader will be able to capture the target before being intercepted by the pursuer. This is shown in Figure 13.

If the evader, by any chance, perceives the target to lie outside his dominance region, the pursuer will be able to capture him by following the derived optimal strategy. This is shown in Figure 14. The dotted line paths in the figure corresponds to the one shown in Figure 13. Now, E miscalculates the position of the target T, to be outside his dominance region. Thus, E

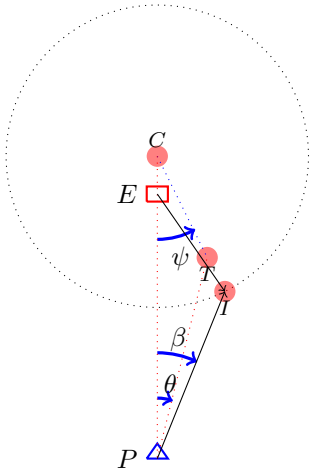


Fig. 12. Target in evader's dominance region

will pursue I instead of T. This results in P intercepting E before E reaches the target. Thus, the target is protected.

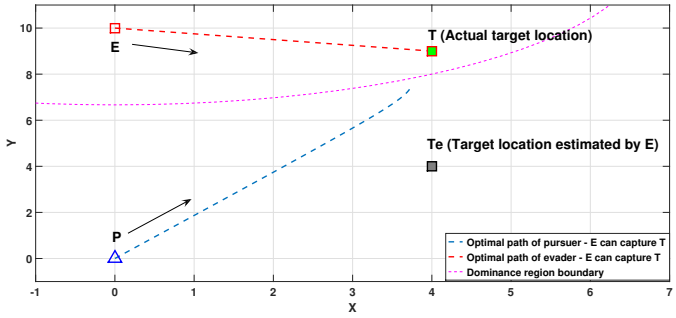


Fig. 13. E can capture T before P intercepts E, if E does not make any estimation error

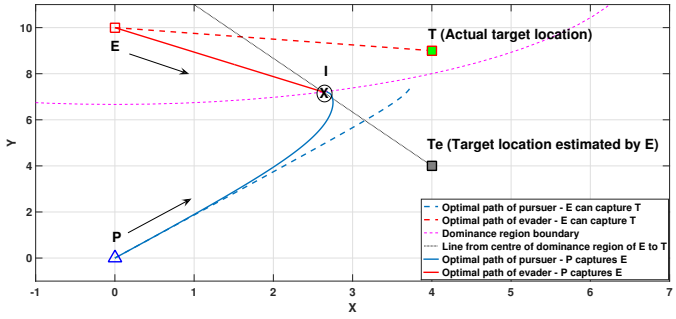


Fig. 14. P intercepts E before E captures T, if E makes an estimation error for a long time

X. Conclusions and Future Work

A. Conclusions

This paper studies the TGP with one pursuer, one evader and a stationary target. A new real time algorithm for deriving the instantaneous optimal path for the pursuer has been presented in this paper. This algorithm is an improvement to the previously proposed COIP guidance law, because it is at least two times faster, and requires only the ratio of the distance between the pursuer and the target to the distance between the pursuer and the evader. It's worth noting that the guidance laws reported in prior literature required the actual co-ordinates of the agents with respect to a static reference frame. Thus the proposed algorithm is useful in scenarios where a localization system is not available to provide the coordinates of the individual players. Also, it facilitates the tabulation of optimal heading values for different locations of the target and the evader, which when used for computing optimal heading in real-time is about seven times faster than the original COIP algorithm.

The modified COIP law has been further tested and validated on an experimental set-up consisting of LEGO robots and a motion capture system.

The case of the target lying in the evader's dominance region was also discussed. The analysis suggests that the optimal heading angle for the pursuer remains the same as that of the case with the target lying inside the pursuer's dominance region.

B. Future work

The proposed algorithm has been derived assuming a two-player single target game. A logical next step would be to modify this algorithm for application to a multi-player multi-target game. In [16], a pursuit-evasion game with obstacles was analysed and dominance regions derived for the same. However, the dynamics of the game was different from TGP. A natural extension would be to analyse the TGP with obstacles and derive real-time implementable algorithms. Other directions for research include handling incomplete or intermittent information on the part of the players playing the game.

Appendix
The heading angle of P when T lies in E's dominance region

Using (1) in (3), it is seen that

$$\overrightarrow{PC} = \frac{1}{1-\frac{e^2}{p^2}}(\overrightarrow{OE}-\overrightarrow{OP}) \tag{43}$$

Therefore,

$$\begin{aligned}\overrightarrow{OC} &= \overrightarrow{OP}+\overrightarrow{PC} \\ &= \frac{1}{1-\frac{e^2}{p^2}}\overrightarrow{OE}-\left(\frac{1}{1-\frac{e^2}{p^2}}-1\right)\overrightarrow{OP}\end{aligned} \tag{44}$$

Let $a=\frac{1}{1-\frac{e^2}{p^2}}$ and $b=\frac{1}{1-\frac{e^2}{p^2}}-1$. Therefore, (44) becomes

$$\overrightarrow{OC} = a\overrightarrow{OE}-b\overrightarrow{OP} \tag{45}$$

and

$$a = b+1 \tag{46}$$

$$\frac{e^2}{p^2} = \frac{b}{a} \tag{47}$$

From (4),

$$\begin{aligned} R^2 &= \frac{b}{a}\overrightarrow{PC}.\overrightarrow{PC} \\ &= \frac{b}{a}\left(\overrightarrow{OC}-\overrightarrow{OP}\right).\left(\overrightarrow{OC}-\overrightarrow{OP}\right) \\ &= \frac{b}{a}\left(|\overrightarrow{OC}|^2+|\overrightarrow{OP}|^2-2\overrightarrow{OP}.\overrightarrow{OC}\right) \\ &= |\overrightarrow{OC}|^2+\frac{b}{a}\left(|\overrightarrow{OC}|^2+|\overrightarrow{OP}|^2-2\overrightarrow{OP}.\overrightarrow{OC}\right) \\ &\quad -|\overrightarrow{OC}|^2 \end{aligned} \tag{48}$$

Substituting (45) for $\overrightarrow{OC}$ in the second and third terms on the R.H.S in (48),

$$R^2 = |\overrightarrow{OC}|^2+b|\overrightarrow{OP}|^2-a|\overrightarrow{OE}|^2 \tag{49}$$

Now,
$$\overrightarrow{\Delta P} = \Delta r\left(\cos\beta\hat{i}+\sin\beta\hat{j}\right) \tag{50}$$

is the vector that results out of P's instantaneous strategy. This implies

$$\overrightarrow{OP}_{\text{new}} = \overrightarrow{OP}+\overrightarrow{\Delta P} \tag{51}$$

Let $\frac{d}{d\beta}$ be denoted by '$'$'. From (50), the objective for P is to choose β to minimize (39), which means β should be chosen such that:

$$\begin{aligned} \alpha' &= R'-\frac{\overrightarrow{CT}.\overrightarrow{CT}'}{|\overrightarrow{CT}|}=0 \qquad (52)\\ &\implies R'-\frac{\overrightarrow{CT}.\left(\overrightarrow{OT}'-\overrightarrow{OC}'\right)}{|\overrightarrow{CT}|}=0 \\ &\implies R'+\frac{\overrightarrow{CT}.\overrightarrow{OC}'}{|\overrightarrow{CT}|}=0 \\ &\implies R'=-\frac{\overrightarrow{CT}.\overrightarrow{OC}'}{|\overrightarrow{CT}|} \qquad (53) \end{aligned}$$

because $\overrightarrow{OT}'=\overrightarrow{0}$ as the target is not moving.

Differentiating (49) w.r.t β,

$$RR' = \overrightarrow{OC}.\overrightarrow{OC}'+b\overrightarrow{OP}.\overrightarrow{OP}'-a\overrightarrow{OE}.\overrightarrow{OE}' \tag{54}$$

Substituting (53) in (54) and rearranging,

$$\left(\overrightarrow{OC}+R\frac{\overrightarrow{CT}}{|\overrightarrow{CT}|}\right).\overrightarrow{OC}' = a\overrightarrow{OE}.\overrightarrow{OE}'-b\overrightarrow{OP}.\overrightarrow{OP}' \tag{55}$$

But from (6), $R\frac{\overrightarrow{CT}}{|\overrightarrow{CT}|}=\overrightarrow{CI}$, and therefore, (55) becomes

$$\overrightarrow{OI}.\overrightarrow{OC}' = a\overrightarrow{OE}.\overrightarrow{OE}'-b\overrightarrow{OP}.\overrightarrow{OP}' \tag{56}$$

Substituting (45) in (56),

$$\overrightarrow{OI}.\left(a\overrightarrow{OE}'-b\overrightarrow{OP}'\right) = a\overrightarrow{OE}.\overrightarrow{OE}'-b\overrightarrow{OP}.\overrightarrow{OP}' \tag{57}$$

But $\overrightarrow{OE}'=\frac{d\overrightarrow{OE}'}{d\beta}=0$. Therefore, (57) becomes

$$b\left(\overrightarrow{OP}-\overrightarrow{OI}\right).\overrightarrow{OP}'=0 \tag{58}$$

Since $b\neq 0$,

$$\begin{aligned} \left(\overrightarrow{OP}-\overrightarrow{OI}\right).\overrightarrow{OP}' &= 0 \qquad (59)\\ \implies \left(\overrightarrow{OP}_{\text{new}}-\overrightarrow{OI}_{\text{new}}\right).\overrightarrow{OP}'_{\text{new}} &= 0 \qquad (60) \end{aligned}$$

From (51), $\overrightarrow{OP}'_{\text{new}}=\frac{d}{d\beta}\overrightarrow{\Delta P}$, as $\frac{d}{d\beta}\overrightarrow{OP}=0$.

Therefore, using (50)

$$\overrightarrow{OP}'_{\text{new}} = \frac{d}{d\beta}\overrightarrow{\Delta P} = \Delta r\left(-\sin\beta\hat{i}+\cos\beta\hat{j}\right) \tag{61}$$

In the limiting case of $\Delta r\to 0$, $\overrightarrow{OP}_{\text{new}}\to\overrightarrow{OP}$ and $\overrightarrow{OI}_{\text{new}}\to\overrightarrow{OI}$. Using these and (61) in (60),

$$\left(\overrightarrow{OP}-\overrightarrow{OI}\right).\Delta r\left(-\sin\beta\hat{i}+\cos\beta\hat{j}\right) = 0 \tag{62}$$

Now, since $\Delta r\to 0$ ($\Delta r\neq 0$), (62) becomes

$$\left[(x_p-x_i)\hat{i}+(y_p-y_i)\hat{j}\right].\left(-\sin\beta\hat{i}+\cos\beta\hat{j}\right)=0 \tag{63}$$

Therefore,

$$\begin{aligned} -(x_p-x_i)\sin\beta+(y_p-y_i)\cos\beta &= 0 \qquad (64)\\ \implies \tan\beta &= \frac{y_p-y_i}{x_p-x_i} \qquad (65) \end{aligned}$$

REFERENCES

[1] R. Harini Venkatesan and N. K. Sinha, "A new guidance law for the defense missile of non-maneuverable aircraft," *Control System Technology, IEEE Transactions on*, vol. 23, no. 6, pp. 2424–2431, 2015.

[2] N. G. Francesco Amigoni and A. Ippedico, "A game-theoretic approach to determining efficient patrolling strategies for mobile robots," *Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT '08. IEEE/WIC/ACM International Conference on*, pp. 500–503, 2008.

[3] I. Rusnak, H. Weiss, and G. Hexner, "Guidance laws in target-missile-defender scenario with an aggressive defender," in *Proceedings of the 18th IFAC World Congress*, vol. 18. Elsevier Milan, 2011, pp. 9349–9354.

[4] S. Panda and Y. Vorobeychik, "Stackelberg games for vaccine design," *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*, 2015.

[5] R. Isaacs, *Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization*. New York: John Wiley and Sons, Inc., 1965.

[6] G. J. O. Tamer Basar, *Dynamic Noncooperative Game Theory*. Philadelphia: SIAM., 1999.

[7] J. Shinar and G. Silberman, "A discrete dynamic game modelling anti-missile defense scenarios," *Dynamics and Control*, vol. 5, no. 1, pp. 55–67, 1995.

[8] A. Perelman, T. Shima, and I. Rusnak, "Cooperative differential games strategies for active aircraft protection from a homing missile," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 3, pp. 761–773, 2011.

[9] V. Shaferman and T. Shima, "Cooperative multiple-model adaptive guidance for an aircraft defending missile," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 6, pp. 1801–1813, 2010.

[10] T. Shima, "Optimal cooperative pursuit and evasion strategies against a homing missile," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 2, pp. 414–425, 2011.

[11] A. Ratnoo and T. Shima, "Line-of-sight interceptor guidance for defending an aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 2, pp. 522–532, 2011.

[12] T. Yamasaki, S. Balakrishnan, and H. Takano, "Modified command to line-of-sight intercept guidance for aircraft defense," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 3, pp. 898–902, 2013.

[13] G. Lau and H. Liu, "Real-time path planning algorithm for autonomous border patrol: Design, simulation and experimentation," *Journal of Intelligent and Robotic Systems*, pp. 517–539, 2014.

[14] R. Harini Venkatesan and N. K. Sinha, "The target guarding problem revisited: Some interesting revelations," in *Proceedings of the 19th World IFAC Congress 2014, Cape Town, South Africa*. IFAC, 24-29 August 2014, pp. 1556–1561.

[15] J. Mohanan, "Real Time Autonomous Target Area Protection," https://youtu.be/ys9uJykYL6k, 2017.
[16] D. W. Oyler, P. T. Kabamba and A. R. Girard, "Pursuit-evasion games in the presence of obstacles," *Automatica*, pp. 1–11, 2016.