Defence Science Journal, Vol. 68, No. 5, September 2018, pp. 451-459, DOI : 10.14429/dsj.68.11896

Capturing an Evader Using Multiple Pursuers with Sensing Limitations in Convex Environment

Amit Kumar

Computer Science and Engineering, Indian Institute of Information Technology, Kota - 302 017, India
E-mail: amit@iiitkota.ac.in

ABSTRACT

A modified continuous-time pursuit-evasion game with multiple pursuers and a single evader is studied. The game has been played in an obstacle-free convex environment which consists an exit gate through which the evader may escape. The geometry of the convex is unknown to all players except pursuers know the location of the exit gate and they can communicate with each other. All players have equal maximum velocities and identical sensing range. An evader is navigating inside the environment and seeking the exit gate to win the game. A novel sweep-pursuit-capture strategy for the pursuers to search and capture the evader under some necessary and sufficient conditions is presented. We also show that three pursuers are sufficient to finish the operation successfully. Non-holonomic wheeled mobile robots of the same configurations have been used as the pursuers and the evader. Simulation studies demonstrate the performance of the proposed strategy in terms of interception time and the distance traveled by the players.

Keywords: Pursuit-evasion games; Path planning; Multi-robot systems

1. INTRODUCTION

Pursuit-evasion games involving multi-robot systems have been received tremendous attention over the past few decades. It provides a classic platform to study cooperation and coordination in multiple robots which has found extensive applications[1-3,8,9]. Apart from classical pursuit-evasion games, an important variant of the game is also under investigation. In this form of the game, multiple pursuers try to capture a single evader whereas the evader may win by exiting safely from the game region. A number of applications of this variant can be found in security and military applications[4-6]. Studied this variant of the pursuit-evasion game in the present study.

Pursuit-evasion games involving multiple pursuers and single evader have been studied into two categories (i) discrete-time, (ii) continuous time. Discrete-time formation of the game has already been received significant attention for a single as well as multiple players[1,7-10]. The continuous version of the game has been reported in the literature using model predictive control[11,12], optimisation techniques[13,14], fuzzy logic and neural network based learning approaches[13,15]. Other recently introduced methods are based on the concept of minimising the evader's Voronoi region for obtaining fast and feasible solutions in pursuit-evasion[6,16]. Sun & Tsiotras[6] have provided a pursuers' strategy based on the dynamic assignment of the pursuers. In this strategy, the working environment is partitioned into Voronoi-like regions which are called the Zermelo-Voronoi Partitions (ZVP). The generated ZVP overcomes the well-known problem with standard voronoi partitions (SVP) to select the best pursuer at any point of time to pursue an evader. The process of selecting the best pursuer using SVP relies on the Euclidean distance between the pursuers and the evader. While the presence of wind and other external disturbances can cause of taking more time from the nearest pursuer as compared to some other pursuer. Sun & Tsiotras[6], proposed strategy is based on ZVP that considers the time to intercept the evader as a relevant distance metric rather than simple Euclidean distance. The interception of single evader, as well as multiple evaders, has been achieved using the well-known Zermelo navigation law by deploying multiple pursuers and taking suitable assumptions. Pan[16], *et al.* proposed a strategy based on Voronoi-regions to capture a single evader using multiple (three) pursuers. The study was carried out in a convex bounded environment where pursuers and the evader know the position of each other at all the times. The geometric information of the environment was known to the players a priori. Pan[16], *et al.* proposed strategy partitions the environment into Voronoi-regions and assigns one pursuer as a defender at the exit gate whose role switches between defender and pursuer. The remaining two pursuers pursue the evader with the tendency to shrink the Voronoi area continuously. The authors have shown that three pursuers are needed and sufficient to catch the evader if the game begins in a configuration such that the evader's Voronoi region does not intersect the exit gate.

These techniques are based on the assumption that the pursuers know the evader's current position at all the time which is not feasible in real-time. A large number of applications can

Received : 07 October 2017, Revised : 12 July 2018
Accepted : 18 July 2018, Online published : 12 September 2018

be found where the pursuers have a finite range of sensing and so can only see the evader when it comes inside pursuer's sensing range. In the context of sensing limitations, in discrete time formulations, probabilistic and randomised strategies are presented[17,19] for multiple pursuers to locate and capture an unpredictable and arbitrary fast evader. A novel pursuit and capture strategy known as *sweep-pursuit-capture* is introduced[18] to capture an evader by a single pursuer in a bounded convex environment as well as in a boundary-less environment using multiple pursuers. The game has been played for the discrete-time domain where sensing and motion ranges are equal to all the players. The authors have introduced two variants of the game. In the first variant, a single pursuer and an evader play inside the bounded environment. The authors have provided a sufficient condition for the ratio of sensing to stepping radius for guaranteed capture. In the second variant, multiple pursuers having the equal maximum velocity as the evader try to catch the evader in open environment. The minimum probabilistic chances and upper bounds on the capture time have also been determined by the authors[18]. In continuous time formulations, control approaches are introduced[20-22] for multiple mobile robots to seek and catch an evader in an unknown and multiply connected planar environment under any shape of the curve.

In this paper, we address the case of the limited and identical sensing range of each player; the maximum velocity of the pursuers and the evader is also equal. All players navigate in an obstacle-free convex environment which has an exit gate. The geometry of the convex environment is unknown to the players except pursuers know the location of the exit gate. Pursuers can communicate with each other but do not know anything about evader's strategy. It is assumed that the evader uses well known Apollonius circle strategy to find an optimal path to escape or prolong the capture by the pursuers. The challenge of the pursuers is to search and capture the evader as soon as possible. This problem is similar to[16] but in our case, pursuers do not have prior knowledge of the environment. We present a novel *sweep-pursuit-capture* strategy for the pursuers in continuous-time domain. This strategy is inspired by the[18] which is originally introduced for the discrete-time domain. We also show that three pursuers are sufficient under some constraint to enclose and capture the evader successfully.

The present strategy works in three phases (i) sweep (ii) pursuit and (iii) capture. A novel sweep method is introduced (Section 4.1) and applied on the pursuers in order to search the evader when the game begins. The sweeping phase is terminated and the pursuit phase is activated when the evader is located by any of the pursuers. In the pursuit phase, the pursuer which initially finds the evader starts following the evader. The remaining two pursuers ensure that the evader cannot move across the exit gate and try to enclose the evader by making an equilateral triangle (Section 4.2). Finally, the capture phase is activated when all three pursuers make an equilateral triangle around the evader. In the capture phase, the area of the triangle will continuously be minimised by the pursuers and finally, they are able to capture the evader (Section 4.3).

Rest of the paper is organised as follows. The problem has been formulated in Section 2. The evader's scheme based on Apollonius circle to avoid the interception is described in Section 3. In Section 4, each phase of the proposed *sweep-pursuit-capture* strategy is explained in detail. Simulation results are presented in Section 5. Finally, conclusions and future directions of the work are highlighted in Section 6.

2. PROBLEM FORMULATION

We consider a multiple players pursuit-evasion game, consists of three pursuers and one evader, which has been played in an unknown convex environment U in R^2. All players are non-holonomic wheeled mobile robots of the same configurations including velocity and sensing range. The convex environment is assumed to be obstacle-free which has an exit gate with endpoints O_1O_2. $d(O_1,O_2) \le 6D$ where, D is the sensing range of the players (Fig. 1). Let $P_i(t) \in R^2$ for $i \in \{1,2,3\}$ and $E(t) \in R^2$ respectively be the positions of the i^{th} pursuer and the evader at the time t. The maximum velocities of all players are assumed to be equal and are defined as

$$V_p^i(t) \le V_{\max} \text{ and } V_e(t) \le V_{\max}, \ \forall t \ge 0$$

where $V_{\max}$ is the maximum velocity defined by the user. For simplicity, we will denote $P_i(t), E(t)$ as and $V_p^i(t), V_e(t)$ as respectively. Initially, at the time $\mathrm{t}=0$, the evader is located randomly inside the environment U and seeking the exit gate to survive whereas pursuers P_i for $i \in \{1,2,3\}$ are located at the exit gate as shown in Fig. 1. The pursuers do not know the position and/or moving strategy of the evader. The aim of the pursuers is to locate and capture the evader. Capture condition is defined as follows:

$$d_{\min} \le 2r + \varepsilon \tag{1}$$

where $d_{\min} = \min_i(d(E,P_i))$ for $i \in \{1,2,3\}$ and $d(E,P_i)$ denotes the Euclidean distance in the plane between the evader E and the pursuer P_i. r is the radius of the player robot and ε is a very small value i.e. pursuer and evader robots are just to collide. Equation (1) states the potential collision and the game ends when such a time (t) is achieved or the evader robot exits the environment.

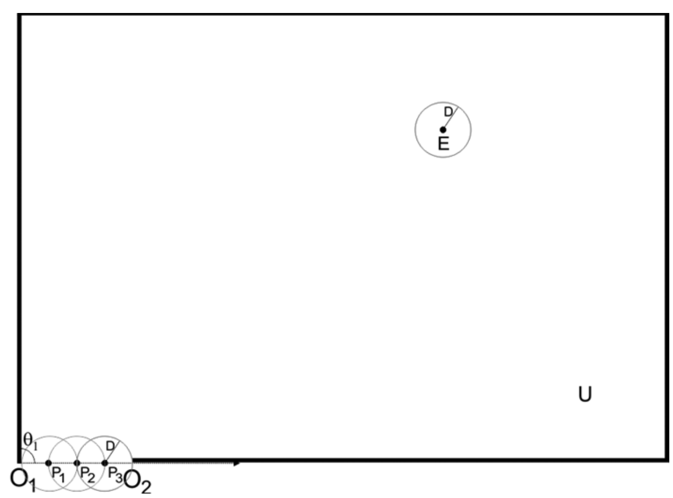


Figure 1. A sample environment *U* with the exit gate O_1 O_2.

While pursuers search and then enclose the evader in order to capture it by communicating and coordinating to each other, the evader also plans to avoid the interception. Before proceeding to the proposed pursuers' strategy, let us first discuss the evader's strategy in the following section.

3. EVADER'S MOTION STRATEGY

The evader is located randomly inside the convex environment when the game begins. The evader moves linearly in order to search the exit gate O_1O_2 . The evader uses a simple and efficient strategy to avoid the interception with the pursuer(s) and the boundary of the environment. The evader's strategy is divided into three possible cases which can occur during the game and are described as follows.

Case 1: The evader detects the boundary of the environment at its heading. In this situation, the evader starts circumnavigating the boundary of the environment in a clockwise or anti-clockwise direction until it detects the exit gate or any pursuer.

Case 2: The evader senses one or more pursuers in its sensing range D as shown in Fig. 2. The evader's strategy to avoid the pursuer(s) is described in algorithm 3.1.

Algorithm 3.1:

If ($d_{\min} \le D$ and $|A|$ equals 2) where $A = \{d(E,P_j) \mid j \in \{1,2,3\}, d(E,P_j) > D\}$ i.e. the evader senses only one pursuer then

Evader rotated by the angle $\theta_p^i - \theta_e$ and moves linearly. θ_p^i and θ_e are the moving directions of the i[th] pursuer and the evader, respectively, with respect to the reference axis as shown in Fig. 2(a). $V_e^{'}$ is the new velocity vector of the evader after rotation.

Else // Evader robot is sensing more than one pursuer as shown in Fig. 2(b)

Compute a set J as $J = \{i \mid i \in \{1,2,3\}, d(E,P_i) \le D\}$

For $k = 1$ to $|J|$-1

$\alpha_{J(k)} = \angle P_{J(k)} E P_{J(k+1)}$, $\alpha_{J(k+1)} = \angle P_{J(1)} E P_{J(|J|)}$

End for

$\max\_k = \max_{k \in J}(\alpha_k)$

Assign the value of k , for which α_k is maximum, to the variable $index$ i.e. $index = k$

$\alpha_E = \angle$Angle of Evader's moving direction with the reference P_{index}

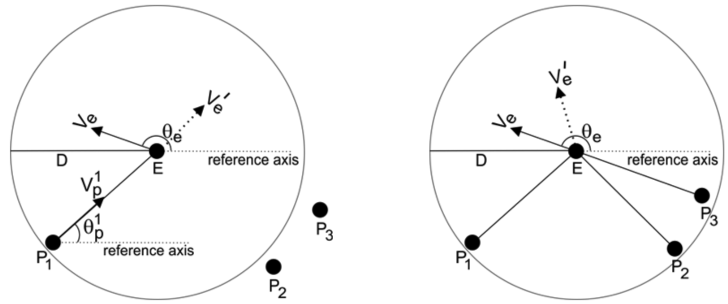
$$Turn\_\alpha = \{\pi - (\max\_k/2)\} - \alpha_E$$

Evader rotated by the angle $Turn\_\alpha$ and moves linearly

End if

Case 3: The evader finds the exit gate. It will navigate through it to win the game.

The evader uses this strategy to avoid the pursuers and the boundary obstacles of the environment while seeking the exit gate. Now the proposed strategy for the pursuers to detect, follow and capture the evader is described in the following section.



(a) (b)

Figure 2. Evader is avoiding (a) single and (b) multiple pursuers.

4. SWEEP-PURSUIT-CAPTURE STRATEGY FOR THE PURSUERS

The sweep-pursuit-capture strategy is first introduced by Bopardikar[18], *et al.* in a discrete-time sense. By inspiring of[18], we present an alternative sweep-pursuit-capture strategy for multiple pursuers in the continuous-time version of a modified pursuit-evasion game.

4.1 Sweep Phase

In this phase, the pursuers explore the unknown environment in order to search the evader. Initially, the pursuers are positioned at the exit gate to secure it. Let us suppose and (x_{end2}, y_{end2}) be the Cartesian coordinates of the endpoints O_1 and O_2 respectively, where $d(O_1,O_2) \le 6D$. Similarly (x_i, y_i) and (x_e, y_e) represent the Cartesian coordinates of the i[th] pursuer P_i and the evader E respectively at the time t . The initial position of all three pursuers is given in Eqn. (2).

$$P_i = \frac{(2i-1)\{d(O_1,O_2)/6\}x_{end2} + \{6-(2i-1)\}\{d(O_1,O_2)/6\}x_{end1}}{d(O_1,O_2)}, \frac{(2i-1)\{d(O_1,O_2)/6\}y_{end2} + \{6-(2i-1)\}\{d(O_1,O_2)/6\}y_{end1}}{d(O_1,O_2)}$$

$$\text{for } i \in \{1,2,3\} \tag{2}$$

The orientation of the pursuers is defined in Eqn. (3).

$$\text{Orientation of } P_i = \theta_1 - (2i-1)\left(\frac{\theta_1 - \theta_2}{6}\right) \text{ for } i \in \{1,2,3\} \tag{3}$$

where θ_1 and θ_2 are the angles of the boundary edges with the reference axis O_1O_2 . For example, $\theta_1 = \pi/2$ and $\theta_2 = 0$ in the sample environment (Fig. 1). In this phase, all pursuers explore the environment to search the evader using algorithm 4.1.

Algorithm 4.1

Step 1: Initialise all pursuers using Eqn. (2) and Eqn. (3).

Step 2: If during *EXPLORATION*, any one of the following conditions is satisfied, the sweeping phase will be terminated.

Condition (i): If $d_{\min} \le D$

Condition (ii): If the evader exits the environment then the game is over. Evader wins.

EXPLORATION:

All pursuers move linearly till the pursuer P_1 and P_3 sense the boundary of the convex environment.

Initialise $k = 1$

while (True)

If k equals 1 then

$$\beta_k = \pm\left[\Pi - Tan^{-1}\left(\frac{D+\lambda}{h_k}\right) - \left\{\frac{\Pi}{2} - \left(\frac{\theta_1-\theta_2}{6}\right)\right\}\right] \text{where,}$$

$$\lambda = [0, D] \text{ and } h_k = \left\{\frac{D}{Tan((\theta_1-\theta_2)/6)}\right\} Tan((\theta_1-\theta_2)/6)$$

Else

$$\beta_k = \pm\left\{\Pi - Tan^{-1}\left(\frac{D+\lambda}{h_k}\right)\right\} \text{ where, } \lambda = [0, D]$$

$$\text{and } h_k = \left\{\frac{D}{Tan((\theta_1-\theta_2)/6)} + (D+\lambda)^{k-1}\right\} Tan((\theta_1-\theta_2)/3)$$

End If

All pursuers P_i for $i \in \{1,2,3\}$ turn by the angle β_k. Sign of the angle β_k depends on which pursuer (P_1 or P_3) leaves the boundary (see Fig. 3).

All pursuers move linearly until at least one pursuer detects the boundary of the environment at its heading.

All pursuers turn by the angle $\beta_k^{'} = \mp(\Pi - |\beta_k|)$

All pursuers move linearly until at least one pursuer detects the boundary of the environment at its heading.

$k = k+1$

End while

End *EXPLORATION*

The evader may exit the environment during sweeping phase because the environment is unknown to the pursuers and may be arbitrarily large. In view of this, necessary and sufficient conditions are also given to capture the evader successfully.

Necessary Condition: $d_{\min} \le D$

Sufficient Condition:

$$\min_j[\max\{d((x',y'),(x_j,y_j)), d((x'',y''),(x_j,y_j))\}] <$$

$$\min\{d((x',y'),(x_e,y_e)), d((x'',y''),(x_e,y_e))\}$$

for $j \in \{1,2,3\}, j \ne i$

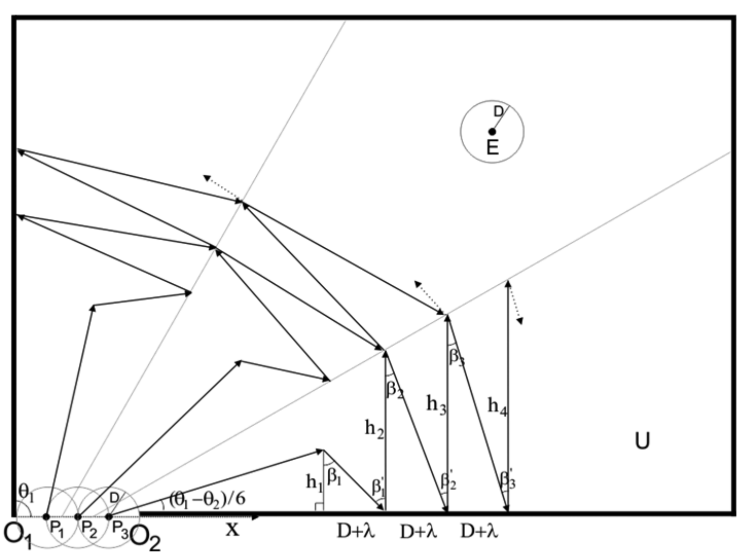


Figure 3. Pursuers are exploring the unknown environment during sweeping phase.

where $x' = \dfrac{r(x_{end2} - x_{end1}) + d(O_1,O_2)x_{end1}}{d(O_1,O_2)}$,

$$y' = \frac{r(y_{end2} - y_{end1}) + d(O_1,O_2)y_{end1}}{d(O_1,O_2)},$$

$$x'' = \frac{r(x_{end1} - x_{end2}) + d(O_1,O_2)x_{end2}}{d(O_1,O_2)} \text{ and}$$

$$y'' = \frac{r(y_{end1} - y_{end2}) + d(O_1,O_2)y_{end2}}{d(O_1,O_2)}.$$

r is the radius of the evader and the pursuers robots.

If necessary and sufficient conditions are satisfied, then it is guaranteed that the evader will be captured by the pursuers under any circumstances. The sweeping phase is terminated and the pursuit phase of the strategy is activated when the necessary condition is satisfied. A detailed description of the pursuit phase is given as follows.

4.2 Pursuit Phase

In this phase, the pursuer P_i for $\exists i \in \{1,2,3\}$, which initially detects the evader, starts following the evader by moving parallel to it. The new velocity vector of the pursuer P_i is denoted as V_p^i (see Fig. 4). To achieve this parallel motion, the next position of the pursuer P_i after one unit of time, whose Cartesian coordinates are represented as (x_i^{next}, y_i^{next}), is calculated as

$$(x_i^{next}, y_i^{next}) = (x_i, y_i) + V_e(t - t_{prev}) \quad (4)$$

where t_{prev} is the time which is defined as *current* time(t)-one unit of time.

The remaining two pursuers also enclose the evader while defending the exit gate using algorithm 4.2 and algorithm 4.3.

Algorithm 4.2

Step 1: Find the pursuer P_j which is nearer to the exit gate as $\min_j[\max\{d((x',y'),(x_j,y_j)), d((x'',y''),(x_j,y_j))\}]$ for $j \in \{1,2,3\}, j \ne i$

Step 2: Choose a point on the exit gate (x',y') or (x'',y'') whichever is nearer to the evader's current position (x_e,y_e). Draw a line segment EO from (x_e,y_e) to the chosen point (x_c,y_c).

Step 3: Compute a point S, represented by (x_s,y_s), on the line segment EO which is furthest from the chosen point (x_c,y_c) and satisfies $d((x_j,y_j),(x_s,y_s)) < d((x_e,y_e),(x_s,y_s))$

Step 4: Pursuer P_j moves toward (x_s,y_s) to create a safe circle for the exit gate with radius $d((x_c,y_c),(x_s,y_s))$ and center (x_c,y_c) (see Fig. 5).

Step 5: The remaining pursuer P_k for $k \in \{1,2,3\}, k \ne i, k \ne j$ encloses the evader using an algorithm 4.3.

Step 6: After reaching the position $(face\_x_f, face\_y_f)$, the pursuer P_k moves to the position (x_{enc1}, y_{enc1}) or (x_{enc2}, y_{enc2}) whichever is nearer

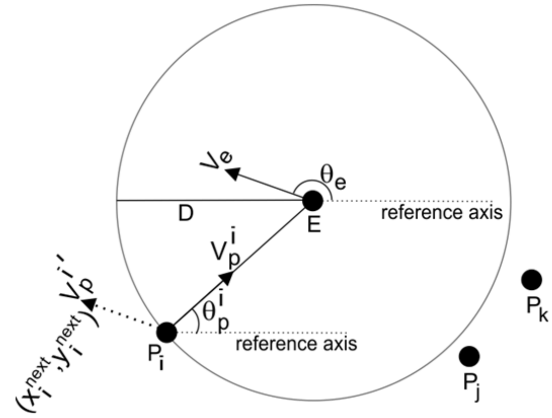


Figure 4. The pursuer P_i, which initially sensed the evader, starts moving parallel to the evader.

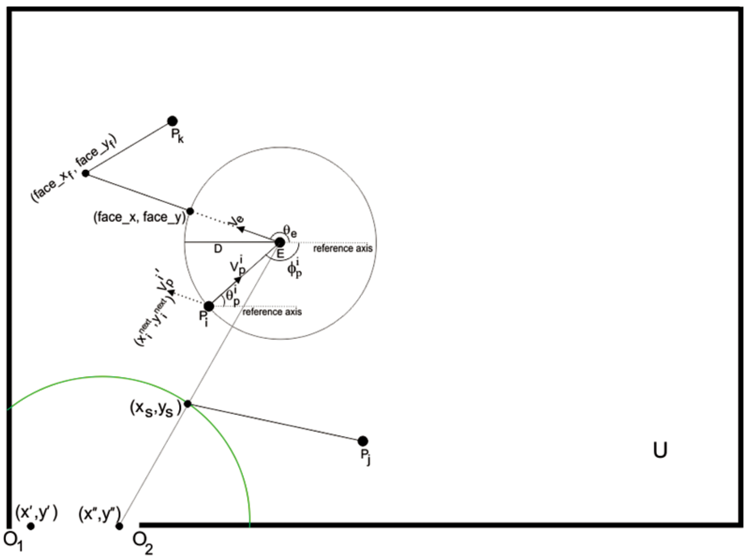


Figure 5. Pursuer P_k moves to the point ($face\_x_f$, $face\_y_f$) to enclose the evader while the pursuer P_j moves to the point (x_s , y_s) to create a safe circle around the exit gate.

$$x_{enc1} = D\cos(\varphi_p^i + 2\pi/3) + x_e, \quad y_{enc1} = D\sin(\varphi_p^i + 2\pi/3) + y_e$$

$$x_{enc2} = D\cos(\varphi_p^i + 4\pi/3) + x_e, \quad y_{enc2} = D\sin(\varphi_p^i + 4\pi/3) + y_e$$

where φ_p^i is the angle of the pursuer P_i with the reference axis of the evader (please see Fig. 5)

Step 7: Pursuer P_k also starts following the evader same as the pursuer P_i using Eqn. (4)

Step 8: The guard pursuer P_j also leaves the exit gate and encloses the evader using an algorithm 4.3 and makes an equilateral triangle using Step 6.

Algorithm 4.3

Step 1: Initialise $f = 1, flag = 0$ and $face\_x = D\cos\theta_e + x_e$, $face\_y = D\sin\theta_e + y_e$

Step 2: while $(f \le n)$ // n is a sufficiently large number

$m_2 = f$

$m_1 = d((face\_x, face\_y),(x_e, y_e)) + m_2$

$face\_x_f = \dfrac{m_1 face\_x - m_2 x_e}{m_1 - m_2}$, $face\_y_f = \dfrac{m_1 face\_y - m_2 y_e}{m_1 - m_2}$

If $d((face\_x, face\_y),(face\_x_f, face\_y_f)) > d((x_k, y_k),(face\_x_f, face\_y_f))$

flag = 1
break
End If
$f = f + 1$
End while

Step 3: If flag equals 1 Pursuer P_k moves to the position $(face\_x_f, face\_y_f)$

Else

No changes in the motion of P_k and go to Step 1.

End if

All three pursuers generate an equilateral triangle when they succeed to enclose the evader. Vertices of the triangle are the positions of the pursuers and the center is the position of the evader (see Fig. 6). At this stage, the evader can also sense all the pursuers and tries to avoid the interception using its strategy which is discussed in Section 3. On the other hand, after making an equilateral triangle, the pursuit phase of the proposed strategy is switched to the capture phase. In this phase, all pursuers move in order to capture the evader by reducing the area of the triangle. Algorithm 4.4 states the pursuers' strategy in the capture phase to finally intercept the evader.

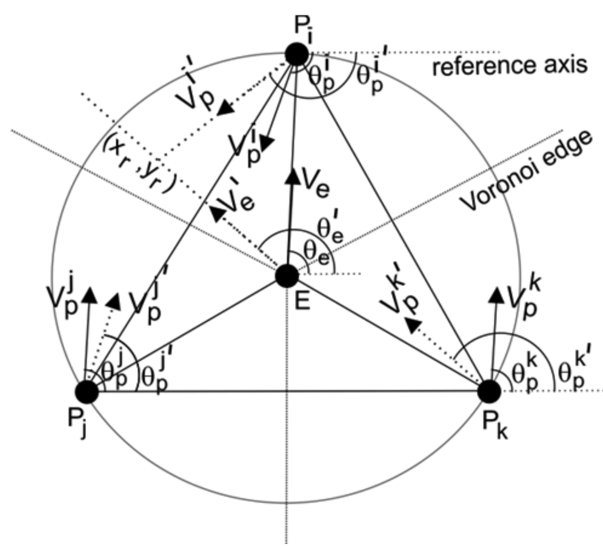


Figure 6. Pursuers enclose the evader by making an equilateral triangle and trying to capture the evader by reducing the area of the triangle.

4.3 Capture Phase

Algorithm 4.4

Step 1: Compute the new moving direction $V_e^{'}$ of the evader E.

Step 2: Pursuer P_i for $\exists! i \in \{1,2,3\}$ of the corresponding Voronoi region calculates the rendezvous point (x_r, y_r) in the moving direction of the evader such that $d((x_r, y_r),(x_i, y_i)) = d((x_r, y_r),(x_e, y_e))$ and the pursuer P_i

moves to the point (x_r, y_r) by satisfying $\left|V_p^{i'}\right| = \left|V_e^{'}\right|$.

Step 3: Out of the remaining two pursuers, select a pursuer P_j as $\min_j(d((x_j, y_j),(x_r, y_r)))$ for $j \in \{1,2,3\}, j \neq i$. The pursuer P_j also moves to the rendezvous point (x_r, y_r) with the condition $\left|V_p^{j'}\right| = \left|V_e^{'}\right|$.

Step 4: The remaining pursuer P_k for $k \in \{1,2,3\}, k \neq i, k \neq j$ moves parallel to the evader using Eqn. (4).

Step 5: If the evader changes its moving direction, repeat Step 1 to 4 again. Quit, if Eqn. (1) is satisfied at any time t.

5. SIMULATION RESULTS

In our simulations, we have taken a rectangular environment with the size of 6.5 m x 5.5 m. The size of the exit gate is kept 1 m and the sensing range of the player robots is also 1 m. The maximum velocity of each player is taken 0.5 m/s. The initial position of the evader was chosen uniformly randomly in the environment. A number of scenarios were considered for the analysis with the different initial position of the evader. Two representative scenarios are as shown in Figs. 7 and 8.

Performance of the proposed sweep-pursuit-capture strategy is as shown in Table 1 with respect to interception time and the distance travelled by the pursuers and the evader.

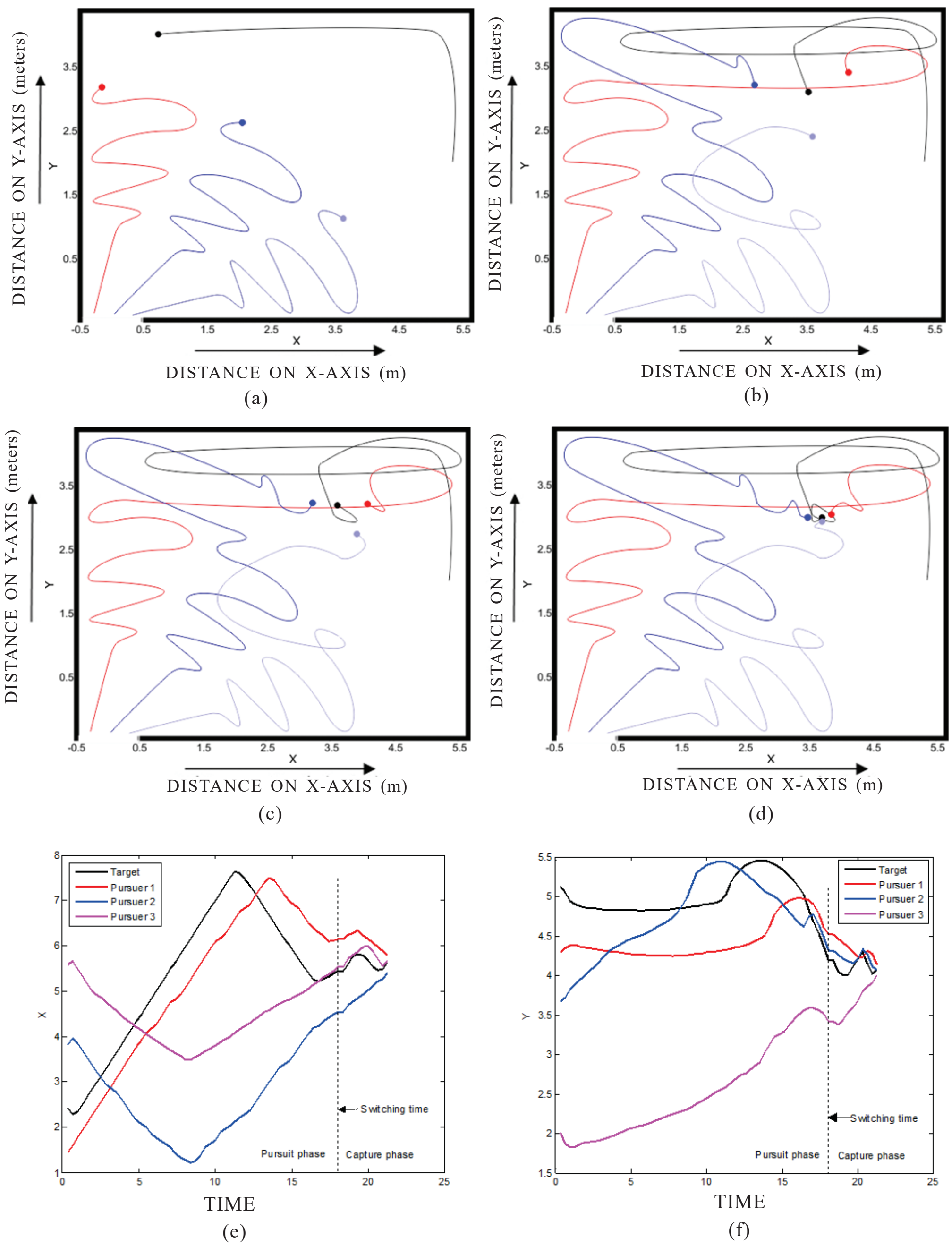


Figure 7. X-Y plot of the pursuers and the evader trajectories using (a) Sweep phase, (b) Sweep + Pursuit phase, (c), (d) Sweep + Pursuit + Capture phase. (e) and (f) X-Y position of the pursuers and the evader during pursuit and capture phase.

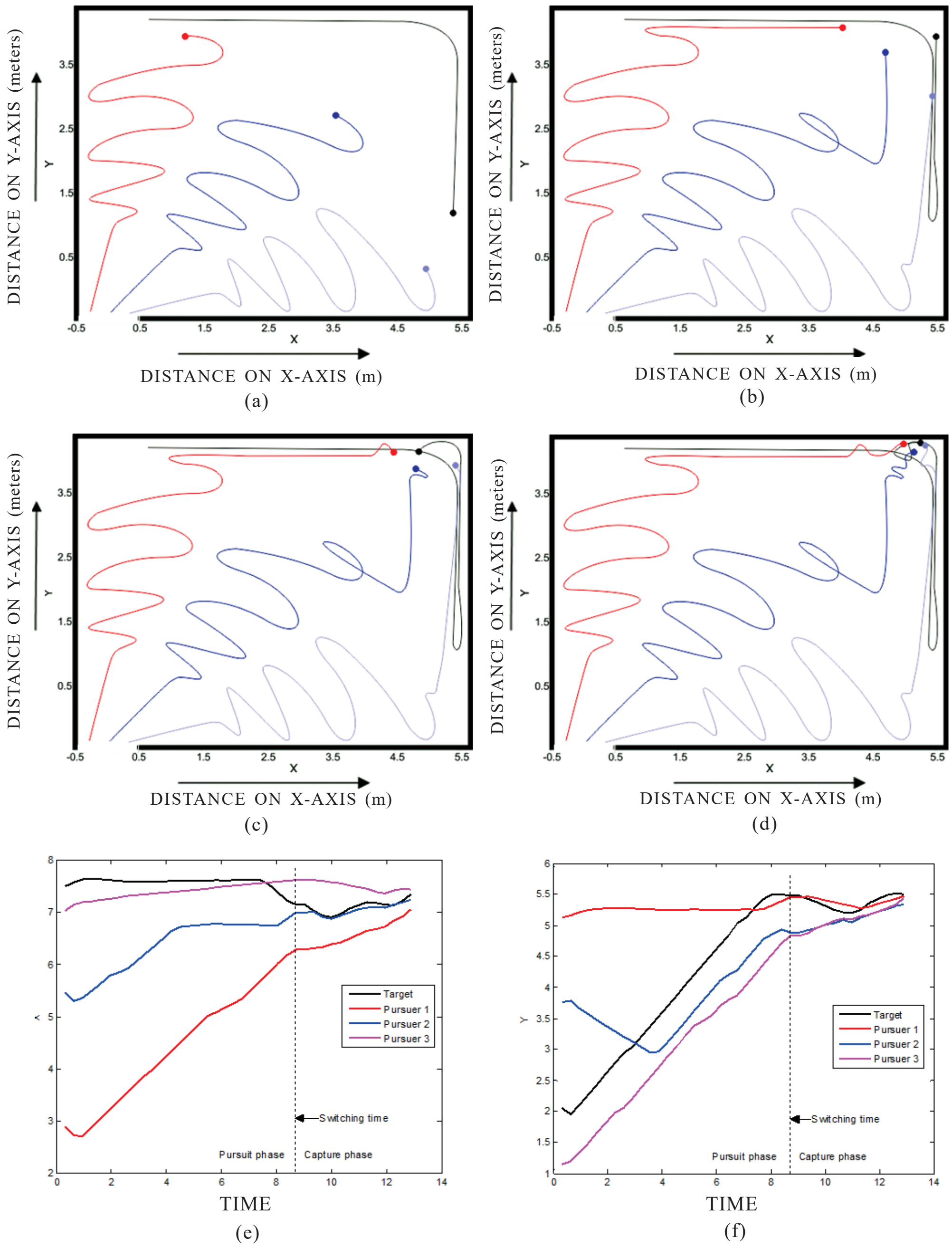


Figure 8. Another scenario, when the evader is captured in a corner. X-Y plot of the pursuers and the evader trajectories using (a) Sweep phase, (b) Sweep + Pursuit phase, (c), (d) Sweep + Pursuit + Capture phase. (e) and (f) X-Y position of the pursuers and the evader during pursuit and capture phase.

Table 1. Interception time and distance travelled by the pursuers and the evader robot in pursuit and capture phase

Scenario	Simulation result as shown in	Interception time (s)	Distance traveled by (m)			
			Evader	Pursuer 1	Pursuer 2	Pursuer 3
Scenario 1	Fig. 7	21.3	9.64	8.52	7.86	6.61
Scenario 2	Fig. 8	13.1	5.13	4.48	4.91	4.23
Scenario 3	--	16.8	7.46	6.80	7.23	6.28
Scenario 4	--	20.6	9.31	7.54	8.04	6.90
Scenario 5	--	19.9	8.85	7.77	7.29	7.53
Scenario 6	--	24.2	10.49	8.38	7.91	7.48
Scenario 7	--	22.7	9.76	8.62	8.68	9.47
Scenario 8	--	25.6	11.43	9.79	10.24	10.68

6. CONCLUSIONS AND FUTURE DIRECTIONS

We have addressed continuous-time pursuit-evasion game problem for multiple pursuers to capture a single evader in a convex environment which has an exit gate. The working environment is obstacle-free but unknown to the players. Only pursuers know the location of the exit gate and communicate and coordinate with each other. The sensing and motion capabilities of all pursuers and the evader are same. The evader is initially randomly located inside a bounded subset of the environment and seeking the exit gate. We proposed a sweep-pursuit-capture strategy for the pursuers to search and capture the evader while defending the exit gate. The proposed strategy switches from the sweep phase to the pursuit phase and finally to the capture phase to finish the game. We also stated necessary and sufficient conditions for guaranteed capture. We have also shown that three pursuers are sufficient to capture the evader if these conditions are satisfied. We presented simulation studies that addressed the performance of the proposed strategy with respect to time taken and the distance traveled to capture the evader. Based on the achieved results, it has been concluded that the evader can be captured efficiently using the proposed method. In future, we plan to design efficient strategies for multiple pursuers to capture a faster evader in a complex environment. We also plan to extend this work including more exit gates and operating in a non-convex environment which is filled with obstacles to reflect real-time situations.

REFERENCES

1. Li, W. Formulation of a cooperative-confinement-escape problem of multiple cooperative defenders against an evader escaping from a circular region. *Commun. Nonlinear Sci. Numerical Simulation*, 2016, **39**, 442–457.
   doi: 10.1016/j.cnsns.2016.02.042
2. Foderaro, G.; Swingler, A. & Ferrari, S. A model-based approach to optimizing Ms. Pac-Man game strategies in real time. *IEEE Trans. Comput. Intelligence AI Games*, 2017, **9**(2), 153-165.
   doi: 10.1109/TCIAIG.2016.2523508
3. Zhoua, Z.; Zhang, W.; Ding, J.; Huang, H.; Stipanovic, D.M. & Tomlin, C.J. Cooperative pursuit with Voronoi partitions. *Automatica*, 2016, **72**, 64-72.
   doi: 10.1016/j.automatica.2016.05.007
4. Alexopoulos, A. & Badreddin, E. Decomposition of multi-player games on the example of pursuit-evasion games with unmanned aerial vehicles. *In* American Control Conference, Boston, USA, 2016, pp. 3789-3795.
   doi: 10.1109/ACC.2016.7525503
5. Fitzpatricka, S.L.; Howell, J.; Messinger, M.E. & Pike, D.A. A deterministic version of the game of zombies and survivors on graphs. *Discrete Appl. Mathem.*, 2016, **213**, 1-12.
   doi: 10.1016/j.dam.2016.06.019
6. Sun, W. & Tsiotras, P. Sequential pursuit of multiple targets under external disturbances via Zermelo–Voronoi diagrams. *Automatica*, 2017, **81**, 253-260.
   doi: 10.1016/j.automatica.2017.03.015
7. Sgall, J. Solution of David Gale's lion and man problem. *Theoretical Comput. Sci.*, 2001, **259**(1–2), 663–670.
   doi: 10.1016/S0304-3975(00)00411-4
8. Kopparty, S. & Ravishankar, C.V. A framework for pursuit evasion games in R^n. *Information Processing Letters*, 2005, **96**(3), 114–122.
   doi: 10.1016/j.ipl.2005.04.012
9. Berger, J. & Lo, N. An innovative multi-agent search-and-rescue path planning approach. *Computers & Operations Research*, 2015, **53**, 24-31.
   doi: 10.1016/j.cor.2014.06.016
10. Johnson, C.L. & Gonzalez, A.J. Learning collaborative team behavior from observation. *Expert Sys. Appl.*, 2014, **41**(5), 2316–2328.
    doi: 10.1016/j.eswa.2013.09.029
11. Bernhard, P. Pursuit-evasion games and zero-sum two-person differential games. *Encyclopedia of Systems and Control, Springer-Verlag*, 2015, pp. 1-8.
    doi: 10.1007/978-1-4471-5102-9_270-1
12. Eklund, J.M.; Sprinkle, J. & Sastry, S.S. Switched and symmetric pursuit/evasion games using online model predictive control with application to autonomous aircraft. *IEEE Trans. Control Sys. Technol.*, 2012, **20**(3), 604–620.
    doi: 10.1109/TCST.2011.2136435
13. Yuan, L.; Dong, Y. & Yong, H. Distributed online mission planning for multi-player space pursuit and evasion. *Chinese J. Aeronautics*, 2016, **29**(6), 1709-1720.
    doi: 10.1016/j.cja.2016.10.012
14. Weng, L.; Liu, Q.; Xia, M. & Song, Y.D. Immune network-based swarm intelligence and its application to unmanned aerial vehicle (UAV) swarm coordination. *Neurocomputing*, 2014, **125**(11), 134–141.
    doi: 10.1016/j.neucom.2012.06.053
15. Hossain, M.A.; Madkour, A.A.M.; Dahal, K.P. & Zhang, L. A real-time dynamic optimal guidance scheme using a general regression neural network. *Eng. Appl. Artificial Intelligence*, 2013, **26**(4), 1230–1236.
    doi: 10.1016/j.engappai.2012.10.007
16. Pan, S. & Huang, H. *et al*. Pursuit, evasion and defense in the plane. *In* American Control Conference, Montreal, Canada, 2012, pp. 4167–4173.
    doi: 10.1109/ACC.2012.6315389
17. Isler, V.; Kannan, S. & Khanna, S. Randomized pursuit-evasion in a polygonal environment. *IEEE Transactions on Robotics*, 2005, **21**(5), 875–884.
    doi: 10.1109/TRO.2005.851373
18. Bopardikar, S.D.; Bullo, F. & Hespanha, J.P. On discrete-time pursuit-evasion games with sensing limitations. *IEEE Trans. Robotics*, 2008, **24**(6), 1429–1439.
    doi: 10.1109/TRO.2008.2006721
19. Bopardikar, S.D.; Borri, A.; Hespanha, J.P.; Prandini, M. & Benedetto, M.D. Randomized sampling for large zero-sum games. *Automatica*, 2013, **49**(5), 1184–1194.
    doi: 10.1016/j.automatica.2013.01.062
20. Durham, J.W.; Franchi, A. & Bullo, F. Distributed pursuit-evasion with limited-visibility sensors via frontier-based exploration. *In* IEEE International Conference on

Robotics and Automation, Anchorage, Alaska, 2010, pp. 3562–3568.
doi: 10.1109/ROBOT.2010.5509347

21. Stiffler, N.M. & Kane, J.M. Shortest paths for visibility-based pursuit-evasion. *In* IEEE International Conference on Robotics and Automation, Saint Paul, Minnesota, 2012, pp. 3997–4002.
doi: 10.1109/ICRA.2012.6225266
22. Liang, X. & Xiao, Y. Mobile sensor intrusion detection under any shape of curve. *Mathe. Comput. Modell.*, 2013, **57**(11-12), 2904–2917.
doi: 10.1016/j.mcm.2012.01.005

CONTRIBUTOR

Dr Amit Kumar received his BE (Computer Science and Engineering) from The Institution of Electronics and Telecommunication Engineers, New Delhi, in 2008. He received his MTech and PhD in Computer Science and Engineering from PDPM Indian Institute of Information Technology Design and Manufacturing Jabalpur, in 2010 and 2016, respectively. Presently, he is working as an Assistant Professor in Computer Science and Engineering Department in Indian Institute of Information Technology, Kota, India. His research interests include robot motion planning, deep learning, multi-robot systems.
In the current study, the whole contribution of this paper is by him starting from the concept development to implementation and results in verification.