Evasion with Terminal Constraints from a Group of Pursuers using a Matrix Game Formulation

Jhanani Selvakumar

Efstathios Bakolas

Abstract— We consider a class of planar pursuit-evasion games with multiple pursuers and a single evader. The evader must reach a target set while avoiding the pursuers which relay the pursuit among themselves. We model this multi-player dynamic game as a two-player multi-stage game. In particular, all the pursuers are modeled as one entity, which we refer to as the super-pursuer, which can deploy only one pursuer of choice at each instant of time. We discretize the decision space of the players and formulate a zero-sum matrix game between the super-pursuer and the evader. In the construction of the matrix game, we explore a myopic solution approach to the game against one that is more far-sighted. In particular, the stage payoffs are constructed in two different ways, namely, with a planning horizon of one stage, and with a planning horizon which is taken to be the remaining stages of the game. We compare the performance of the evasion policies that are obtained in these two cases, against pursuers who engage in relay pursuit. Finally, we compare the pursuit policies obtained from the multi-stage matrix games using extensive numerical simulations.

I. INTRODUCTION

Multi-player game theory is applicable to many real-world problems, for instance, autonomous collision avoidance, modeling biological behaviour and trading in markets. Multi-player pursuit-evasion games (PEGs) are rich in the number of parameters that govern the progress of the game. In this paper, we address an evasion problem in which a single evader tries to reach a specified target (goal) set, while avoiding a group of pursuers. The pursuers engage in a semi-cooperative pursuit strategy called relay pursuit. We formulate the continuous dynamic game as a multi-act matrix game. At each stage, we consider the pursuers as being one entity, which is engaged in a zero-sum game with the single evader. At any stage, all players have perfect information while the fixed target set is known only to the evader. To illustrate our modeling, we present a specific scenario where all the players have simple dynamics. We compare the pursuit strategy derived by solving the matrix game to the strategy of relay pursuit based on the minimum time of capture. Subsequently, we analyze the effectiveness of using the metric of minimum time of capture to decide the active pursuer.

Literature survey: Game theory with multiple players has received a lot of attention in the past and continues to interest

J. Selvakumar is a graduate student at the Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712-1221, USA, Email: jhanani@utexas.edu

E. Bakolas is an Assistant Professor at the Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712-1221, USA, Email: bakolas@austin.utexas.edu

many researchers. The framework of multi-player games can simulate many scenarios in economics, competitive environments, and defense [1]–[3]. Multi-agent problems have many aspects to them (static or dynamic, information and cooperation patterns, etc.) which dictate the solution method for the game. A team of players with a single objective could act cooperatively [3], [4] or non-cooperatively. The Nash equilibrium solutions for a non-cooperative group of players, in particular, matrix games, are discussed in detail by Basar and Olsder in [5] and Zaccour et. al in [6]. Static games [5] are played once and have fixed payoffs for discrete actions of the players, while dynamic games are repeated static games or governed by continuous time equations [7]. The folk theorem is a popular result regarding the individual game equilibria of infinitely repeated static games [8].

Multi-player pursuit evasion games with a single evader and multiple pursuers have been dealt with by several approaches such as using Voronoi partitions, switching pursuit strategies and sequential pursuit [9]–[12]. A roadmap derived from a generalized Voronoi partition of the given domain to guide the evader was first proposed in [13] and further developed in our previous work [14]. Discrete multi-player games played on a grid have also received considerable attention. A greedy pursuit policy based on the probability of finding the evader at a specific location in the domain is presented in [15]. A pursuit evasion game between a team of evaders and a team of heterogeneous pursuers using greedy policies is presented in [16]. A multi-agent pursuit evasion game in an uncertain environment is formulated as a Markov game in [17], where a receding horizon approach is used with a matrix game to obtain optimal policies for the players.

Contributions: In this paper, we have multiple pursuers engaging in relay pursuit with a single evader who has a target point to reach in the state space while avoiding capture. Hence, this is a multi-objective game for the evader. The main contributions of this paper are as follows:
1) Novel formulation of a dynamic multi-player non-zero sum PEG as a multi-act two-person zero sum game,
2) Development of a payoff function which reflects the two-fold goal of the evader,
3) Comparison and analysis of the effectiveness of different pursuit and evasion strategies.

Structure of the paper: Section II presents the formulation of the target-seeking evasion problem. This problem is framed as a multi-act two-person game and subsequently solved in Section III. In Section IV, we describe a game scenario where

the players have simple dynamics with equal or unequal speeds, along with results from numerical simulations. In Section V, we present concluding remarks.

II. FORMULATION OF TARGET-SEEKING EVASION PROBLEM

We consider a pursuit evasion game in an unconstrained domain in $\mathbb{R}^4$, with N pursuers and one evader. The upper bound for the duration of the game is known apriori and denoted by $\bar{T}_f > 0$. At any given time $t \in [0, \bar{T}_f]$, the state of the ith pursuer, where $i \in \mathcal{I} := \{1, 2, ..., N\}$, is denoted as $\boldsymbol{\xi}_i := [\boldsymbol{x}_i \ \boldsymbol{v}_i]^{\mathrm{T}} \in \mathbb{R}^4$, where $\boldsymbol{x}_i \in \mathbb{R}^2$ and $\boldsymbol{v}_i \in \mathbb{R}^2$ are its position and velocity, respectively. In general, let the pursuers have the following equations of motion:

$$\dot{\boldsymbol{\xi}}_i = \boldsymbol{f}_P(\boldsymbol{\xi}_i, \boldsymbol{u}_i), \qquad \boldsymbol{\xi}_i(0) = \bar{\boldsymbol{\xi}}_i, \qquad (1)$$

where $\boldsymbol{f}_P$ is a known function that satisfies regularity assumptions for existence of a solution to (1) and $\boldsymbol{u}_i$ denotes the input. We impose the following constraint on the magnitude of the control input: $\|\boldsymbol{u}_i(t)\| \in \{0, 1\}, \forall i \in \mathcal{I}, \forall t \in [0, \bar{T}_f]$. The position and velocity of the single evader at time t is denoted by $\boldsymbol{x}_e \in \mathbb{R}^2$ and $\boldsymbol{v}_e \in \mathbb{R}^2$ respectively, and its state is denoted by $\boldsymbol{\eta} := [\boldsymbol{x}_e \ \boldsymbol{v}_e]^{\mathrm{T}}$. The evader's dynamics is in general described by

$$\dot{\boldsymbol{\eta}} = \boldsymbol{f}_E(\boldsymbol{\eta}, \boldsymbol{u}_e), \qquad \boldsymbol{\eta}(0) = \bar{\boldsymbol{\eta}}, \qquad (2)$$

where $\boldsymbol{f}_E$ is known and satisfies similar regularity assumptions as $\boldsymbol{f}_P$, and $\boldsymbol{u}_e$ is the evader's input vector, with $\|\boldsymbol{u}_e(t)\| \in \{0, 1\}, \forall t \in [0, \bar{T}_f]$. Capture is defined as positional proximity of atleast one of the pursuers with the evader within a pre-specified tolerance $l > 0$. More precisely, the evader will be considered captured, if $\exists i \in \mathcal{I} : \|\boldsymbol{x}_i(t) - \boldsymbol{x}_e(t)\| \leq l$ for some $t \in [0, \bar{T}_f]$. The target set, which is a single point, is denoted by $\boldsymbol{x}_G \in \mathbb{R}^2$. The tolerance criterion for goal-reaching is represented by ϵ, which taken to be a positive number.

A. The Problem

The target-seeking evasion problem is a dynamic multi-player non-zero sum game. It is stated as follows: Given a set of initial conditions for all the players in the plane, find a *time-history* of input vectors for the evader to reach the target location within a desired tolerance, while the evader avoids capture by any pursuer. Formally,
Given $\bar{\boldsymbol{\xi}}_i \ \forall i \in \mathcal{I}, \bar{\boldsymbol{\eta}}, \boldsymbol{x}_G, l, \epsilon$, and $\bar{T}_f$,
Find $T_f \in [0, \bar{T}_f]$ and $\boldsymbol{u}_e(t), \ \forall t \in [0, T_f]$
such that $\|\boldsymbol{x}_e(T_f) - \boldsymbol{x}_G\| \leq \epsilon$ and $\|\boldsymbol{x}_e(t) - \boldsymbol{x}_i(t)\| > l, \ \forall i \in \mathcal{I}, \forall t \in [0, T_f]$.
Each player has perfect information about the states of all players of the game at all times. The pursuers relay the pursuit amongst themselves, such that at each instant of time, the pursuer who can capture the evader in the least amount of time is the active pursuer. In this case, we say that the minimum time of capture is the *relay metric*. The relay metric could also be a different parameter of the game. The location

of $\boldsymbol{x}_G$ is known only to the evader. In the next section, we describe the conversion of our problem into a multi-act two person zero-sum game.

III. CONVERSION TO A MULTI-ACT TWO-PERSON GAME

Let the game be played in K finite stages, with a constant time step $\Delta t > 0$. We perform a zero-order hold discretization of the dynamics in equations (1) and (2) with Δt as the sampling time. Let $k \in \{0, 1, .., K\}$ denote the current stage of the game, $\boldsymbol{\xi}_i(k)$ be the state vector of the ith pursuer at that stage and $\boldsymbol{\eta}(k)$ be the state vector of the evader. Then the new equations of motion in discrete-time are

$$\boldsymbol{\xi}_i(k+1) = \boldsymbol{f}_{Pd}(\boldsymbol{\xi}_i(k), \boldsymbol{u}_i(k)), \qquad \boldsymbol{\xi}_i(0) = \bar{\boldsymbol{\xi}}_i,$$
$$\boldsymbol{\eta}(k+1) = \boldsymbol{f}_{Ed}(\boldsymbol{\eta}(k), \boldsymbol{u}_e(k)), \qquad \boldsymbol{\eta}(0) = \bar{\boldsymbol{\eta}}, \quad (3)$$

where $\boldsymbol{u}_i(k)$ and $\boldsymbol{u}_e(k)$ denote the inputs of the ith pursuer and the evader at stage k respectively. The time-discretization of the functions $\boldsymbol{f}_P(\cdot)$ and $\boldsymbol{f}_E(\cdot)$ yields the new functions $\boldsymbol{f}_{Pd}(\cdot)$ and $\boldsymbol{f}_{Ed}(\cdot)$ respectively. Since we assume that the pursuers employ relay-pursuit, we can approximate the actions of the group of pursuers as the actions of a single entity (the *super-pursuer*) which deploys one pursuer at a time. This means that the multi-player game is essentially reduced to a two-player game between the evader and the super-pursuer. In addition, we consider that the game between the evader and the super-pursuer is zero-sum at each stage, and consequently, the entries in the payoff matrix for each game represent the reward obtained by the evader or the cost incurred by the super-pursuer (the team of pursuers).

A. Description of the payoff matrix of the game

At each stage k of the game, consider a matrix $M_k \in \mathbb{R}^{N \times (N+1)}$, whose entries are the payoffs to E at that stage. Each row of M_k represents a pure strategy played by P and each column, a pure strategy played by E. The decision space available to the players (the choice of control inputs for P and E) is infinite. We consider a restricted decision space for P, including only the actions that appear "integral" to the pursuers' goal of capturing the evader. In particular, P has exactly N pure strategies, where the ith pure strategy corresponds to the case where only the ith pursuer goes after the evader. Similarly, E's restricted decision space consists of $N + 1$ actions, where the first N correspond to evasion from each pursuer in turn (the jth action is to avoid only the jth pursuer), and the $(N + 1)^{th}$ action is the target-seeking behavior, which means that the evader directly heads towards the target.

Let i be the row index of M_k and j be the column index, where $i \in \mathcal{I}$ and $j \in \mathcal{J} := \{1, 2, .., N + 1\}$. If we consider the first N columns of M_k, each entry $M_k(i, i)$ is the payoff for the two-player zero sum game between only the ith pursuer and the evader. Every other entry $M_k(i, j), i \neq j$, represents a case when E tries to evade from the jth pursuer, when actually the ith pursuer is active. This situation can happen because while the E knows the states of all the

TABLE I
ENTRIES IN THE PAYOFF MATRIX M_k, FOR A CASE OF $N = 2$.

(1,1) $P1$ in pursuit E evading $P1$	(1,2) $P1$ in pursuit E evading $P2$	(1,3) $P1$ in pursuit E seeks $\boldsymbol{x}_G$
(2,1) $P2$ in pursuit E evading $P1$	(2,2) $P2$ in pursuit E evading $P2$	(2,3) $P2$ in pursuit E seeks $\boldsymbol{x}_G$

pursuers, it does not know the action chosen by P at the same stage. Finally, the last column of the matrix M_k represents cases where the evader is directly headed towards the target $\boldsymbol{x}_G$, and only one pursuer is active per row. A schematic construction of the matrix M_k is shown in Table I.

B. Time of capture function

Let us consider the construction of each entry of M_k. The min-max time of capture of the evader by a single pursuer plays an important role in our formulation of the matrix game payoffs. At any time, let $\phi(\boldsymbol{\eta}, \boldsymbol{\xi}_i)$ denote the min-max time of capture of the evader (whose current state is $\boldsymbol{\eta}$) by the ith pursuer (whose current state is $\boldsymbol{\xi}_i$). Depending on the dynamics in equations (1) and (2), we may be able to calculate $\phi(\boldsymbol{\eta}, \boldsymbol{\xi}_i)$ even in closed form, though more often numerical techniques must be employed. For instance, we can obtain $\phi(\boldsymbol{\eta}, \boldsymbol{\xi}_i)$, by solving a simple quadratic (when all players are single integrators) or quartic equation (pursuers with finite acceleration), or by numerical root-solving techniques. The min-max time of capture is our chosen metric to represent the risk of capture for E.

C. Elements of the payoff matrix

Each element of the payoff matrix is a numerical value that reflects the two-fold objective of the evader: (1) to avoid capture and (2) to reach the target location $\boldsymbol{x}_G$. The two components of each entry are the time that P would take to capture E, and the extent to which E's heading is towards $\boldsymbol{x}_G$ from its current location. The target-seeking component of E's velocity is measured by $\cos\psi$, where ψ is the angle between the vectors $\boldsymbol{v}_e$ and $\boldsymbol{x}_G - \boldsymbol{x}_e$.

We have to re-construct the payoff matrix at every stage since at least two players move. Even with a discounting factor $\gamma = 1$ (which means that the future is as important as the present for consideration), it is difficult to estimate the payoff that the evader will receive at the end of K stages, since the payoffs at each stage are dependent on the players' states in the current stage. Thus, the history of moves in previous play is reflected in the changing payoff values, although this information is not available directly to the players as a strategy recall.

Alternatively, instead of considering discrete states on a continuous space for each player, we could characterize the states in a different classification based on safety or proximity to the goal. Then, a choice of different set of pure strategies (actions) would yield a game with a fixed payoff matrix, and we can solve for the subgame perfect equilibria by starting from the last stage of play. This approach,

however, has the disadvantage that some information is lost when translating the Cartesian state space into a different representation, since we need to heuristically classify the states.

We have evaluated the performance of the evader when $\gamma = 1$ as well as when $\gamma = 0$. The exact steps in computing the payoff matrix M_k at stage k are detailed in the next section, for $\gamma = 0$ (the present stage is all that is taken into account).

D. Planning horizon: one stage

Let us first consider the case with a planning horizon of one stage ($\gamma = 0$). Each entry of the matrix M_k is associated with two components: one representing evasion and the other representing the target-seeking behavior, and is constructed as follows. For each pair of pure strategies, we calculate the new positions of the players after playing those strategies for one stage (execution horizon is a single stage). Then, if the time of capture for the evader using the new positions is smaller than the old positions, that component of the payoff will be set to -1, since it is favorable to the pursuer. If the new positions are favorable to the evader, the payoff will be set to $+1$. If there is no change, the payoff will be zero. Similarly, if the evader's new position is closer to the goal than previously, the goal component of the payoff will be set to $+1$, and if the evader has moved away from the goal, the payoff will be set to -1. Maintaining the same distance from the goal merits zero payoff. The sum of these two quantities yields a single entry of the matrix M_k.

All entries of M_k belong to the set $\{-2, -1, 0, 1, 2\}$. Let the input corresponding to the realization of the ith pure strategy of P be the n-tuplet $\boldsymbol{q}_i$, whose only non-zero entry is equal to one and is at the ith position (that is, the ith pursuer is active and all other pursuers have zero input). Similarly, let $\boldsymbol{p}_j$ be the input corresponding to the realization of the jth pure strategy of E. The Algorithm (1) shows the main steps for the assignment of payoffs to M_k:

---

input : $\boldsymbol{\eta}$, $\boldsymbol{x}_G$, $\boldsymbol{\xi}_i$ $\forall i \in \mathcal{I}$, k
output: M_k

for $i \leftarrow 1$ **to** N **do**
    for $j \leftarrow 1$ **to** $N + 1$ **do**
        $\boldsymbol{\xi}_i' = \boldsymbol{f}_{Pd}(\boldsymbol{\xi}_i, \boldsymbol{q}_i)$
        $\boldsymbol{\eta}' = \boldsymbol{f}_{Ed}(\boldsymbol{\eta}_i, \boldsymbol{p}_j)$
        $T_c = \text{sgn}(\phi(\boldsymbol{\eta}', \boldsymbol{\xi}_i') - \phi(\boldsymbol{\eta}, \boldsymbol{\xi}_i))$
        $G_c = \text{sgn}(\|\boldsymbol{x}_e - \boldsymbol{x}_G\| - \|\boldsymbol{x}_e' - \boldsymbol{x}_G\|)$
        $M_k(i, j) = T_c + G_c$
    end
end

Algorithm 1: Payoff Assignment to M_k

E. Planning horizon: Remaining stages of the game

When $\gamma = 1$, the payoffs are designed to reflect the long-term effects of each action. In this case, since we have an

upper bound $\bar{T}_f$ on the duration of the game, the time-of-capture component is bounded. For each pair of pure strategies (i, j), we calculate the minimum time-of-capture of E by the pursuing agent i. Note that E will play the strategy corresponding to evasion from the pursuer j. If capture is not possible, we set the value to $\bar{T}_f$. The target-seeking component is given by $\cos(\psi)$, as described in Section III-C. The Algorithm (2) shows the main steps for the assignment of payoffs to M_k:

---

input : $\boldsymbol{\eta}$, $\boldsymbol{x}_G$, $\boldsymbol{\xi}_i$ $\forall i \in \mathcal{I}$, k, $\bar{T}_f$
output: M_k

for $i \leftarrow 1$ **to** N **do**
    for $j \leftarrow 1$ **to** $N+1$ **do**
        $T_c = \min(\phi(\boldsymbol{\eta}, \boldsymbol{\xi}_i), \bar{T}_f)$
        $G_c = \frac{\langle \boldsymbol{v}_e, \boldsymbol{x}_G - \boldsymbol{x}_e \rangle}{\|\boldsymbol{v}_e\| \|\boldsymbol{x}_G - \boldsymbol{x}_e\|}$
        $M_{k1}(i, j) = T_c$
        $M_{k2}(i, j) = G_c$
    end
end
$\hat{M}_{k1} = \frac{M_{k1}}{\max_{i,j} M_{k1}}$
$M_k = \hat{M}_{k1} + M_{k2}$

Algorithm 2: Payoff Assignment to M_k

Note that we normalize the evasion component which is given by the matrix M_{k1} using the maximum entry of the matrix. This ensures that all evasion components have values between zero and unity, similar to the target-seeking component. The summation of the two components in this manner is a standard practice in multi-objective optimization where the objectives are combined into one global criterion [18].

F. Solution to the matrix game

Now that we have formulated the matrix game, we can solve for the equilibrium strategies using standard techniques. An equivalent non-zero sum formulation for our problem would consider the whole $N + 1$ player game, with cost assignments that are functions of the states of all players. Subsequently, verifying the existence of an equilibrium set of pure strategies is a hard problem, in the sense that it would require an exhaustive search among all possibilities. However, we know that a two-player zero sum multi-act game which is finite admits a saddle point solution in mixed strategies [5].

The mixed strategy for each player is a vector of probabilities. At any stage k, for P, we have a mixed strategy described by a vector $\boldsymbol{y} \in \mathbb{R}^N$, with $\sum_i y_i = 1$. Similarly for E, we have a strategy described by $\boldsymbol{z} \in \mathbb{R}^{N+1}$, $\sum_j z_j = 1$. The ith entry of the vector $\boldsymbol{y}$ (or $\boldsymbol{z}$) represents the probability of the ith pure strategy being employed by P (or E). The computation of the vectors $\boldsymbol{y}_*$ and $\boldsymbol{z}_*$ which solve for the saddle point (the min-max problem) of the game is formulated as a Linear Programming (LP) problem [6],

which can be solved using readily available solvers. The package cvx [19] was used for the simulations that will be presented in Section IV. We recompute M_k for every stage of the game as the players move in the state space. The actions (pure strategies) for a particular stage of the game are obtained as random samples from the discrete distributions given by $\boldsymbol{y}_*$ and $\boldsymbol{z}_*$ for that stage of the game.

IV. SPECIFIC EXAMPLE

In this section, we apply our proposed method of game construction and evasion solution to a specific scenario. Let us consider a game where all players have simple dynamics and all the pursuers have the same speed. In particular, the equations of motion of the players are:

$$\dot{\boldsymbol{x}}_i = v_p \boldsymbol{u}_i, \qquad \boldsymbol{x}_i(0) = \bar{\boldsymbol{x}}_i \quad \forall i \in \mathcal{I}$$
$$\dot{\boldsymbol{x}}_e = v_e \boldsymbol{u}_e, \qquad \boldsymbol{x}_e(0) = \bar{\boldsymbol{x}}_e$$

The active pursuer always follows the line of sight to the evader, that is, the pursuer engages in what is referred to as "pure pursuit" in literature. In general, the time of capture of the evader by any pursuer in this case depends on the relative velocity. If the pursuer cannot capture the evader, resulting in infinite value for the capture time $\phi(\boldsymbol{\eta}, \boldsymbol{\xi}_i)$, or if the time of capture calculated is greater than $\bar{T}_f$, we assign $\phi(\boldsymbol{\eta}, \boldsymbol{\xi}_i) = \bar{T}_f$ to ensure that the payoff values remain finite. The calculation of the off-diagonal values of the payoff matrix requires the time of capture when the evader is intercepted by a pursuer. A simple method to calculate this can be found in [20].

We consider three different cases: when the evader's speed is equal to that of the pursuers ($v_e = v_p$), when the evader is slower ($v_e < v_p$) and when the evader is faster ($v_e > v_p$). For each of these cases, we consider three different pursuit policies. The relay-pursuit strategy based on minimum time of capture will be referred to as RP, and the stage-by stage optimal policy using the matrix game for one planning horizon ($\gamma = 0$) will be referred to as SH, and the long planning horizon policy ($\gamma = 1$) will be referred to as LH. The evader's policies could be long horizon (LH) or short horizon (SH). All players use only one type of policy for the whole game. In this section, when we refer to a game, we will refer to the pursuer's strategy first and then the evader's strategy, for instance, an "RP vs LH" game indicates that the pursuers played relay pursuit based on minimum time of capture and the evader used the long planning horizon policy at each stage. For the numerical simulations, we use randomly generated target location and initial positions for the players.

In our numerical simulations, we considered cases with the number of pursuers $N \in \{2, 3, ..., 7\}$. All pursuers have unit speed ($v_p = 1$) and the other parameters are chosen to have values as follows: $l = 0.1$, $\epsilon = 0.05$, $\bar{T}_f = 14.2$ and $v_e \in \{0.9, 1, 1.1\}$. Each pair of policies was tested for $\sim 10^3$ runs of random initial conditions, where the initial distance between the evader and the goal was ensured to be always

greater than the minimum initial distance between a pursuer and the evader. In Table II, we see the percentage of cases where the evader reached the target successfully for different combinations of pursuit and evasion policies. Similarly, in Table III, we see the percentage of games that ended in the evader being captured. The rest of the games not represented in the tables above do not conclude within the fixed upper bound for duration $\bar{T}_f$.

From Table II, we observe that the LH evasion policy performs better than SH evasion against RP pursuit policy in terms of reaching the target successfully. Also, LH evasion performs better against LH pursuit than SH evasion against SH pursuit. Even if the inconclusive games are considered in favor of SH evasion, LH evasion has an advantage in that it is faster in reaching the target. Hence, as expected, the evader benefits from keeping in mind a far-reaching effect of its current actions rather than the short-sighted play of choosing the best move at each stage. This holds for all three different speeds of the evader.

The following figures (Fig.1 - Fig. 3) highlight the evader's play for a specific set of initial conditions with four pursuers. The evader's path is shown in green squares and the pursuers are shown as red triangles. The initial position of evader is solid green and those of the pursuers are solid red. The goal position is shown as a solid black circle.

Fig. 2. $N = 4$, Equal speeds for all players. LH evasion reaches the target location in both (b) and (d), while in (c), SH evasion brings the evader close to the target. In (b), the evader is close to being captured though it has reached the target.

TABLE II
OUTCOME OF GAMES - SIMPLE DYNAMICS, TARGET REACHED

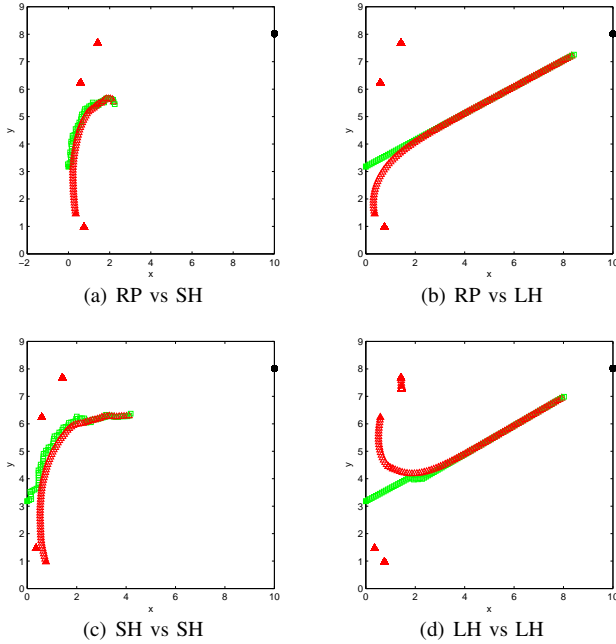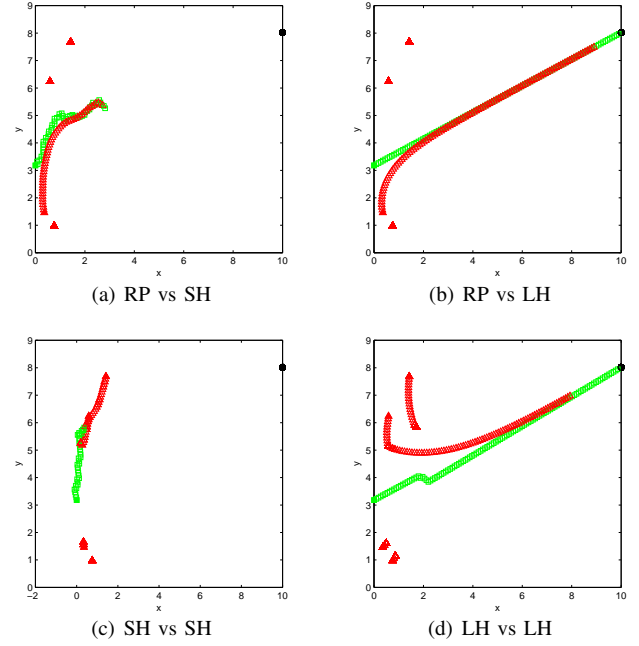| | | $v_e < v_p$ | | $v_e = v_p$ | | $v_e > v_p$ | |
P \ E		LH	SH	LH	SH	LH	SH
RP		8.67%	2.67%	54.73%	5.07%	51.60%	3.10%
LH		7.13%	-	51.67%	-	62.60%	-
SH		-	2.60%	-	5.80%	-	1.90%

Fig. 1. $N = 4$, Slower evader. LH evasion reaches the target location in (b), while in (d), LH evasion brings the evader close to the target.

In terms of pursuit policies, LH pursuit is observed to have a slight advantage over RP pursuit against an LH evader. This is expected since long-horizon planning takes into account both the time to capture and the evader's heading. Against an SH evader, however, RP pursuit has the upper hand over SH

pursuit. Hence, the pursuers would benefit from using either minimum time of capture or a long-term payoff as their relay metric.

As the evader's speed increases, we notice that more games are inconclusive in the time limit chosen for simulations. Of the concluded games, it is evident that LH evasion outperforms SH evasion against RP pursuit in terms of reaching the target and delaying capture. Considering the inconclusive games, the evader using LH evasion was closer to the goal than the nearest pursuer at the time $\bar{T}_f$ in about 3% of games compared to an evader using SH evasion policy.

TABLE III
OUTCOME OF GAMES - SIMPLE DYNAMICS, EVADER CAPTURED

| | | $v_e < v_p$ | | $v_e = v_p$ | | $v_e > v_p$ | |
P \ E		LH	SH	LH	SH	LH	SH
RP		90.97%	96.70%	29.00%	87.00%	20.20%	90.00%
LH		92.63%	-	36.67%	-	22.00%	-
SH		-	96.97%	-	26.93%	-	29.10%

In Fig. 3 the faster evader's play is shown for the same set of initial conditions. The evader is shown in green squares and the pursuers as red triangles and the goal in black.

(a) RP vs SH

(b) RP vs LH
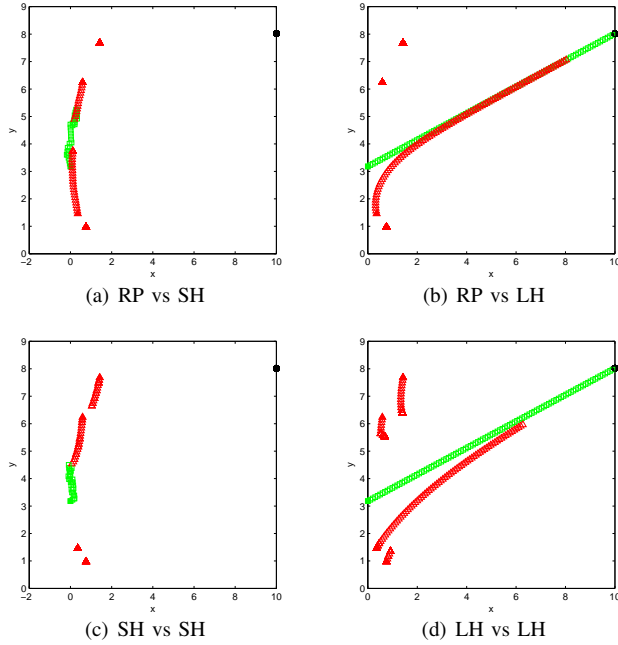
(c) SH vs SH

(d) LH vs LH

Fig. 3. $N = 4$, Faster evader. LH evasion reaches the target location in (b) and (d), while in (c), SH evasion avoids capture for a long time.

Finally, we see that in all the cases, the long planning horizon policy ($\gamma = 1$) does perform better than a single-stage planning policy against relay pursuit using minimum time of capture as the relay metric. In terms of pursuit strategy, the relay pursuit using long-horizon payoff as the metric performs better than the relay pursuit using minimum-time as the metric. Hence, the long-term payoff is an effective alternative to minimum time of capture as a relay metric.

V. CONCLUSION

In this paper, we have addressed a problem of evasion from multiple pursuers by reducing it to a multi-act, two player zero-sum game. The proposed solution approach employs the well-known framework of matrix games. In particular, at each time step, we solve a relevant matrix game to account for the dynamic nature of the game. It turns out that the method presented in this paper can be easily extended to games with more complex dynamics for the players, as long as the payoff components are computationally inexpensive to obtain. Further, based on extensive simulations, we argue that in most cases, long-term planning is more effective for evasion than a myopic strategy, in particular against the pursuers playing relay pursuit.

We would like to extend this framework to more complex games, such as games restricted to compact domains and/or domains containing regions that must be avoided (i.e., obstacles). Finally, we would like to explore the possibility of

providing guarantees on the performance of the evader, as well as analyze and quantify the sensitivity of the evader's strategy to the discounting factor.

REFERENCES

[1] R. J. Aumann, "Game theory," in *Game Theory*, pp. 1–53, Springer, 1989.
[2] W. Scott and N. Leonard, "Pursuit, herding and evasion: A three-agent model of caribou predation," in *ACC 2013*, pp. 2978–2983, June, 2013.
[3] A. Perelman, T. Shima, and I. Rusnak, "Cooperative differential games strategies for active aircraft protection from a homing missile," *J. Guid. Contr. Dynam.*, vol. 34, no. 3, pp. 761–773, 2011.
[4] Z. Fuchs, P. Khargonekar, and J. Evers, "Cooperative defense within a single-pursuer, two-evader pursuit evasion differential game," in *CDC 2010*, pp. 3091–3097, Dec., 2010.
[5] T. Basar and G. J. Olsder, *Dynamic Noncooperative Game Theory*. London: SIAM, 1995.
[6] A. Haurie, J. B. Krawczyk, G. Zaccour, *et al.*, "Games and dynamic games," *World Scientific Books*, 2012.
[7] R. Isaacs, *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization*. New York: Dover Publication, 1999.
[8] D. Fudenberg and E. Maskin, "The folk theorem in repeated games with discounting or with incomplete information," *Econometrica: Journal of the Econometric Society*, pp. 533–554, 1986.
[9] S. Pan, H. Huang, J. Ding, W. Zhang, D. Stipanovic, and C. Tomlin, "Pursuit, evasion and defense in the plane," in *ACC 2012*, pp. 4167–4173, June, 2012.
[10] H. Huang, W. Zhang, J. Ding, D. Stipanovic, and C. Tomlin, "Guaranteed decentralized pursuit-evasion in the plane with multiple pursuers," in *CDC-ECC 2011*, pp. 4835–4840, Dec., 2011.
[11] W. Sun and P. Tsiotras, "A sequential pursuer-target assignment problem under external disturbances," in *CDC 2013*, pp. 3994–3999, Dec., 2013.
[12] E. Bakolas and P. Tsiotras, "Relay pursuit of a maneuvering target using dynamic Voronoi diagrams," *Automatica*, vol. 48, no. 9, pp. 2213–2220, 2012.
[13] E. Bakolas, "Evasion from a group of pursuers with double integrator kinematics," in *CDC 2013*, pp. 1472–1477, Dec., 2013.
[14] J. Selvakumar and E. Bakolas, "Evasion from a group of pursuers with a prescribed target set for the evader," in *ACC 2016*, pp. 155–160, July 2016.
[15] J. P. Hespanha, H. J. Kim, and S. Sastry, "Multiple-agent probabilistic pursuit-evasion games," in *CDC 1999*, pp. 2432–2437, Dec 1999.
[16] R. Vidal, O. Shakernia, H. Kim, D. Shim, and S. Sastry, "Probabilistic pursuit-evasion games: theory, implementation, and experimental evaluation," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 662–669, 2002.
[17] J. P. Hespanha, M. Prandini, and S. Sastry, "Probabilistic pursuit-evasion games: A one-step Nash approach," in *CDC 2000*, pp. 2272–2277, Dec 2000.
[18] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Structural and multidisciplinary optimization*, vol. 26, no. 6, pp. 369–395, 2004.
[19] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 1.21 (2011)," *Available: cvxr. com/cvx*, 2010.
[20] E. Bakolas and P. Tsiotras, "The Zermelo-Voronoi diagram: A dynamic partition problem," *Automatica*, vol. 46, no. 12, pp. 2059–2067, 2010.