JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 4, March 2007

Multi-Pursuer Multi-Evader Pursuit-Evasion Games with Jamming Confrontation

Mo Wei,* Genshe Chen,† Jose B. Cruz, Jr.,‡ Leonard S. Haynes,§ Khanh Pham,¶ and Erik Blasch#

Existing pursuer-evader (PE) game algorithms do not provide good real-time solutions for situations with the following complexities: (1) multi-pursuer multi-evader, (2) multiple evaders with superior control resources such as higher speeds, and (3) jamming confrontation between pursuers and evaders. This paper introduces a real-time decentralized approach, in which decentralization strategy reduces computational complexity in multi-pursuer multi-evader situations, cooperative chasing strategy guarantees capture of some superior evaders, and min-max double-sided jamming confrontation provides optimal jamming-estimation strategies under adversarial noisy environments. Extensive simulations confirm the efficiency of this approach.

I. Introduction

MUCH efforts has been devoted to pursuer-evader games in noisy environments for a wide number of applications such as military battlefield engagements, criminal pursuit, collision avoidance designs in intelligent transportation systems, space debris tracking and collection, and other related areas. In such games, pursuer(s) wish to capture evader(s), while evader(s) try to avoid capture. For a pursuer-evader pair, if at some point in time, the distance between the pursuer and evader is less than some predefined unit distance, the pursuer captures the evader. If the evader can stay away from this range forever, the evader wins.

The starting point of pursuit-evasion (PE) games is the one-pursuer one-evader case, which is a two-player, zero-sum game that could be solved via Hamilton-Jacobi-Isaacs (HJI) equations.[1] Different varieties of this problem such as (a) the lion-man problem[2] in which a pursuer tries to catch a same-speed evader in a bounded space, (b) pursuit-evasion in polygonal environments, and (c) pursuit-evasion in polygonal environments with a door[3] have been studied. In real-world applications, algorithms such as randomized search[4] and level sets[5] have been used to calculate applicable numerical control strategies for one or both players. In addition, techniques such as statistical reasoning strategies have been developed so that learning about the patterns of moving player is possible.[7]

During the last decade, such PE game strategies have been proved to be efficient when dealing with situations mentioned above. However, modern PE cases, especially battlefield situations, are often much more complex and pose great challenges to them.

Challenge 1: How can the one-pursuer one-evader game be extended to the multi-pursuer multi-evader game

Since the practical applications of one-pursuer one-evader game are somewhat limited, there has been increasing interest in multi-player pursuit-evasion situations. However, there are fundamental problems with extending the solutions of one-pursuer one-evader games to multi-player situations. First, the final state of the game cannot be

Received 20 May 2006; revision received 07 February 2007; accepted for publication 07 February 2007.
* Ph. D. Student, Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210.
† Senior Research Scientist, Intelligent Automation, Inc., AIAA Senior Member. gchen@i-a-i.com.
‡ Distinguished Professor of Engineering, The Ohio State University, Columbus, OH 43210.
§ President, Intelligent Automation, Inc., 15400 Calhoun Dr., Suite 400, Rockville, MD 20855.
¶ AFRL/VSSV, Kirtland AFB, NM 87117.
# AFRL/SNAA, WPAFB, OH 45433.

easily specified. Second, the value function of the game is extremely difficult to calculate and the reasonability of existing calculation approaches needs further investigation. Unfortunately, calculating value functions is the basis of solving HJI-like equations. Third, if there are evaders with superior control resources such as higher speed, the problem will be even more complex because sufficient conditions for capturing superior evaders are difficult to identify. Finally, for possible centralized algorithms, the computation required will be prohibitive even for a very small-scale game such as a two-pursuer three-evader case.[8]

To mitigate such difficulties, researchers have studied two kinds of sub-problems: one-pursuer multi-evader game and multi-pursuer one-evader game. Zemskov and Pashkov[9] studied the problem with one pursuer and two evaders, assuming one of the evaders must disappear at some time. However, which evader will disappear and when it will disappear are not known at the beginning of the game. Pashkov and Sinitsyn[10] studied a three-pursuer one-evader game with fixed game time. The above researches have greatly extended PE studies, and might lead to a future optimal algorithm for multi-pursuer multi-evader situations. However, a practical algorithm is required so that fast solutions with acceptable sub-optimality can be obtained for such complex situations. Some theoretical aspects of multi-player PE games have been addressed.[25,26]

Challenge 2: How to capture a superior evader

The existence of superior evader(s) cause extra problems for traditional PE game strategies. In this paper, an evader whose speed is similar to or higher than that of the corresponding pursuer is said to be "superior to that pursuer". If an evader is superior to all pursuers in the game, it is said to be a "superior evader". To capture a superior evader is a long standing problem and there is no applicable algorithm can guarantee the capture of high speed evader especially if the evade can move without mobility limits such as space limits, inertia, fuel limits, etc. This is because: 1) It is common knowledge in a pursuit evasion game, if the evader and the pursuer have the same maneuverability, to escape from a pursuer is usually much easier than to capture an evader; 2) Better maneuverability (such as higher speed) often gives the vehicle an overwhelming advantage.

For these reasons, to intercept a high speed enemy missile and guard a safe zone in an open space, it is anticipated that at least the following conditions must be satisfied: 1) There are multiple pursuers; 2) The initial positions of these pursuers should be "around" the evader, not just on one side of it; 3) The speeds and the initial positions of pursuers should satisfy some requirements. All these cause great difficulties for traditional PE theories.

Challenge 3: How to analyze possible jamming influences in PE games

Jamming[17,19] and estimation[22–24] have recently received more and more attention in adversarial PE situations due to the increasing availability of portable signal processing instruments. Here, jamming is defined as follows:

Definition 1.1 Jamming is a soft kill action that attempts to dilute the effectiveness of an enemy weapon system through confusion, distraction, or deception.

The most common jamming strategy is to send interfering signals, which is the case discussed in Section IV. The reason why jamming is important in PE games is that both sides might use jamming to confuse the enemy. Since World War II, interest has been growing steadily in techniques that exploit the electro-magnetic spectrum to gather intelligence in support of warfare. Consequently, many measures to prevent enemy radar from being effective have been developed. One of these is electronic countermeasures (ECM), or more commonly called, jamming. Aircrafts and ground systems capable of using jamming techniques can confuse defense radar and allow raids to be performed with greater safety. It is therefore not surprising that jamming has today become an integral part of any tactical or strategic operation, including multi-player PE games.

The contribution of this paper is an analysis of the three challenges. A sub-optimal decentralized approach to multi-pursuer multi-evader PE games in adversarial noisy environments in which jamming might exist is proposed. Both logical analysis and simulations confirm the approach's reasonableness and acceptable computation complexity. Multi-player game solutions can be obtained in practical applications in a real-time mode. Moreover, jamming confrontation is considered in the solutions, which can help in analyzing modern pursuer-evader games under complex adversarial noisy environments.

The paper is organized as follows. Section II discusses the strategies for challenge 1. Section III investigates the strategy for capturing multiple superior evaders. Section IV analyzes jamming confrontation in PE games. Section V summarizes Section II-IV and provides the complete algorithm for the approach proposed in this paper. Section VI is devoted to simulations. Section VII is the conclusion.

II. Decentralization of Multi-pursuer Multi-evader PE Games

To facilitate computer implementation of the decentralized approach; in this section, the formulas have already been discretized.

A. Problem Formulation

There are N pursuers, identified as $p_1, \ldots, p_N$, and M evaders, identified as $e_1, \ldots, e_M$, in an open n-dimensional battlefield. Here "open" means there is no boundary and the battlefield occupies the whole n-dimensional Euclidean space. p_i's speed at timestep k is denoted as $u_i(k) = [u_{i,1}(k), \ldots, u_{i,n}(k)]^{\mathrm{T}}$, which is an n-dimensional column vector. p_i's location is also a n-dimensional column vector denoted as $x_i^p(k) = [x_{i,1}^p(k), \ldots, x_{i,n}^p(k)]^{\mathrm{T}}$. Correspondingly, $v_j(k) = [v_{j,1}(k), \ldots, v_{j,n}(k)]^{\mathrm{T}}$ is e_j's speed and $x_j^e(k) = [x_{j,1}^e(k), \ldots, x_{j,n}^e(k)]^{\mathrm{T}}$ is e_j's location. We assume that all pursuers and evaders have constant speeds. However, the directions of speeds can be arbitrary, and the constant speeds can be different for different pursuers or evaders.

$$\left\| u_i(k) \right\| = \sqrt{\sum_{r=1}^{n} u_{i,r}(k)^2} = c_i^p > 0 \tag{1}$$

$$\left\| v_j(k) \right\| = \sqrt{\sum_{r=1}^{n} v_{j,r}(k)^2} = c_j^e > 0 \tag{2}$$

where c_j^p and c_j^e are constants. There might exist one or more $j \in \{1, 2, \ldots, M\}$ such that

$$\left\| v_j(k) \right\| \geq \left\| u_i(k) \right\|, \quad \text{for any } \ i \in \{1, 2, \ldots, N\} \tag{3}$$

The discrete time update of the system is

$$\begin{aligned} x_i^p(k+1) &= x_i^p(k) + u_i(k) \\ x_j^e(k+1) &= x_j^e(k) + v_j(k) \end{aligned} \tag{4}$$

Note that $u_i(k)$ and $v_j(k)$ may be coupled due to control strategy calculations.

The purpose of pursuers is to capture evaders, and the purpose of evaders is to escape. Individual goals might be different according to different priorities, such as capturing as many evaders as possible, or maximizing captured value as possible so that high-value evaders will be first considered as targets of pursuers. Different applications will choose different priorities. If the distance between the jth evader and the ith pursuer is less than one unit at timestep k, the jth evader is captured by the ith pursuer. One evader may be captured by multiple pursuers simultaneously. Note also that one pursuer may capture multiple evaders simultaneously.

B. Decentralization

It is usually difficult to obtain an optimal solution for multi-player PE games. Some situations accept sub-optimal solutions; however, the computation of a centralized algorithm is often prohibitive for real-world applications. It takes about 14 hours to calculate 300 timesteps for a two-pursuer three-evader case (Matlab, 2G CPU, 512M memory). In this paper, we will use a decentralized algorithm requiring much less computation time to find sub-optimal solutions so that real-time applications are possible.

To obtain a decentralized algorithm, the first step is to decompose a multi-pursuer multi-evader game into one-pursuer-one-evader problems and multi-pursuer one-evader problems. Based on this decomposition, many results for one-pursuer one-evader problems can be applied.

Note that the decentralized algorithm proposed in this paper is different from ordinary decentralization approaches. Ordinary decentralization approaches tend to do the assignment at one time (the beginning of the game). That is, assign each pursuer a sequence of evaders so that when the pursuer finishes capturing the current evader it directly continues to chase the next evader arranged in its sequence. In this way, the problem can be stated as a problem similar to the Mixed Integer Linear Programming (MILP) problem,[8,11–14] assuming all evaders can be captured and the game value function can be obtained for multi-pursuer-multi-evader situations. However, the value function required is extremely difficult to calculate due to both theoretical and practical difficulties, especially for situations with multiple superior evaders. The second problem is that since there are time sequences, even if we exploit some approximate value functions (which might still be computation-intensive, according to simulation results)[8], the search space is daunting when the number of platforms increases.

Our approach is to divide the game procedure into stages. The division between two stages is the timestep that one or more evaders are captured. At the start of each stage, a coordination will be implemented whereby each pursuer will be assigned to exactly one non-captured evader. More than one pursuer might chase the same superior evader so that its free space can be squeezed. Each pursuer (or evader) is numbered at the beginning of the game, and it knows the numbers of all pursuers and evaders. The numbers will be fixed during the entire game. At each coordination stage divider, each pursuer and evader will independently do all the calculations according to the same assignment algorithm. As a result of the numbering mechanism, all the results will be the same and each pursuer or evader will know its role. In this way, team coordination in decentralized mode is ensured with minimal communication. For the cases where each platform (pursuer or evader) has only partial information of the battlefield, additional communication operations will be required so that all team members have the same level of information. By reducing the communications to a level low, synchronous methods such as the Ring Algorithm for decentralized networks can be applied.[15,16,18]

The number of coordination stages in our approach is more than the MILP algorithm (which has only one main coordination stage at the beginning of the game). However, compared to the ordinary MILP approaches, the computation time required is much less for most situations. It is anticipated that this decentralized approach is more suitable for real-time applications.

III. Capturing Superior Evaders

The focus of our decomposition is the assignment problems involving superior evaders. For a superior evader, capture is not guaranteed. However, a necessary condition can be determined for multi-pursuer single-evader cases,[8] although it is still very complex and difficult to use in actual algorithms.

Assuming $v > u_1$ holds in Fig. 1. If the pursuer can capture the evader in the way shown in Fig. 1, we have

$$\frac{\sin\alpha}{\sin\beta} = \frac{v}{u_1} \quad or \quad \sin\beta = \left(\frac{u_1}{v}\sin\alpha\right) \leq \frac{u_1}{v} \tag{5}$$

Clearly, if evader chooses any $\beta = \angle p_1 e E \in [0, \pi/2]$ such that $\sin\beta \leq (u_1/v)$ holds, the pursuer can always find a α so that in this time interval there exists an intersection, which means capture. However, if evader chooses a β

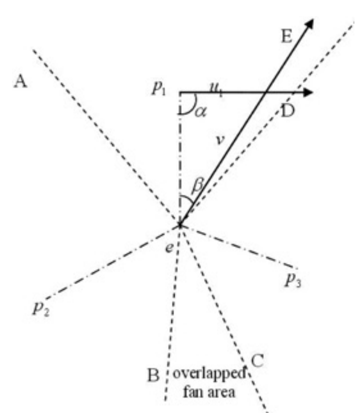

Fig. 1 Geometric capture on a plane.

such that $\sin\beta > (u_1/v)$ holds, the pursuer can not find an α such that (5) holds, which implies that capture is not guaranteed. It might be more illustrative to think that the pursuer only "controls the fan-like area within the angle $2\beta^*$", where $\beta^* = \angle p_1 e D = \max\beta = \arcsin(u_1/v)$. If the evader chooses an escape direction which is included in the angle $2\beta^*$, it will be captured by the pursuer. Otherwise, it escapes. We can call such a fan-like area as a *control fan*. If all possible moving directions of the evader lie in fan-like areas controlled by some pursuer(s) (such as $\angle AeC$, $\angle AeD$, and $\angle DeB$ in Fig. 1), the evader can not escape if such pursuers cooperate to squeeze the free space of the evader. Based on this, a necessary condition for the multi-pursuer single-evader situation can be specified.

Theorem 3.1. *If the evader is guaranteed to be captured, then at any time t, $0 \le t \le T$, there exists one pursuer such that the angle β with respect to that pursuer satisfies* (5).[8]

Proof. It is easy to explain and prove this necessary condition by contradiction: Suppose there is a "gap" between the control fans. The evader can definitely escape through the gap via a straight-line route. ■

Note that Theorem 3.1 can be extended to higher dimensional space. This theorem can be directly used in a centralized way, which requires coordination at each timestep. However, such a centralized approach often implies intractable computation and too much dependence on communication. It is still an open question to find an applicable criterion for the general case so that the decentralized algorithm can easily be applied.

Deriving a sufficient condition for capturing superior evader(s) is expected to be much more complicated due to the positions, dynamics, and control strategies of platforms. For special cases, in which the pursuers have same speed as the evader (close to the lion-man problem), we propose a sufficient condition as stated in Theorem 3.2.

Theorem 3.2. *Assume in an n-dimensional open space there is one evader and $n+1$ pursuers, and all the pursuers have the same speed as the evader. Assume the $n+1$ pursuers construct $(n+1)$-vertex convex object. Suppose the evader and pursuers move alternately. If the evader initially lies in the inner side of the $(n+1)$-vertex object (not including the boundaries except the vertexes), the evader is guaranteed to be captured within a finite number of steps. The maximum timestep needed is bounded by*

$$T_{\max} \le \frac{(n+1)\max(d_i)}{2v\Delta t\cos(\max(\theta_{ij}/2))}, \quad \theta_{ij} < \pi, \ \ i \ne j \tag{6}$$

where d_i is the distance between the ith pursuer and the evader. θ_{ij} is the angle $\angle p_i e p_j$ in Fig. 2. Δt is the length of one timestep.

Proof. The proof logic is somewhat related to the approach for solving the previously long-standing lion-man problem.[2] The one-dimensional case ($n = 1$) is trivial, because the two pursuers and the evader can only move along a straight line and the "2-vertex object" will be a straight-line segment. If the evader is on this line segment, for

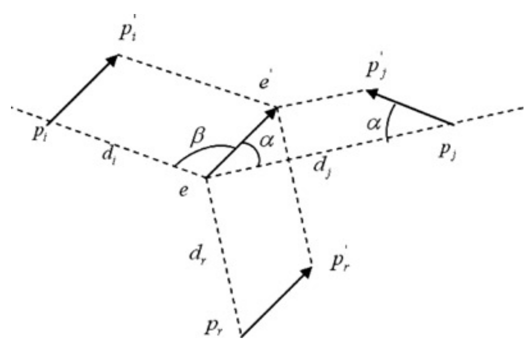

Fig. 2 Illustration of Theorem 3.2.

sure the two pursuers at the two ends can capture it. We prove the two-dimensional situation, and the proof can be extended to higher dimensional situations.

For two-dimensional ($n = 2$) situation, it is illustrative to consider the $(n + 1)$-vertex convex object constructed by the pursuers as a triangle, with the evader lying in the triangle. When the evader moves, the pursuers can take the following motions. Each of the two pursuers whose inner angle $\angle p_i e p_j < \pi$ can include (or "hold") the direction of the motion of the evader goes along the direction that has similar angle deviating from the line connecting the pursuer and the evader. If there is more than one direction satisfying this requirement, the pursuer chooses the direction that results in a closer distance between itself and the evader. All other pursuers move along directions parallel to the moving direction of the evader. In this way, after this stage, no angles between any pursuers and evader will change, and all future analyses for this step can be applied to all later steps in a similar way.

Denote the speed of all platforms as v. We consider two possible situations.

Situation 1. Assume$\beta \geq \pi/2$ holds. Since $\angle p_i e p_j = \theta_{ij} = \alpha + \beta < \pi$ holds, $\alpha = \theta_{ij} - \beta \leq \theta_{ij} - \pi/2$ will hold. This in turn implies $\|e' p'_j\| = d_j - 2v\Delta t \cos\alpha \leq d_j - 2v\Delta t \cos(\theta_{ij}/2)$ holds. This is because $\theta_{ij} < \pi \Rightarrow \theta_{ij} - \pi/2 \leq \theta_{ij}/2$.

Situation 2. Assume both $\beta < \pi/2$ and $\alpha < \pi/2$ hold. Clearly, $\|e' p'_j\| = d_j - 2v\Delta t \cos\alpha$ and $\|e' p'_i\| = d_i - 2v\Delta t \cos\beta$ hold simultaneously. We claim that at least one of the two formulas (7) and (8) holds

$$\left\|e' p'_j\right\| \leq d_j - 2v\Delta t \cos(\theta_{ij}/2) \tag{7}$$

$$\left\|e' p'_i\right\| \leq d_i - 2v\Delta t \cos(\theta_{ij}/2) \tag{8}$$

This claim is proved by contradiction. Suppose both do not hold, the following two inequalities, (9) and (10), will hold simultaneously.

$$\cos\alpha < \cos(\theta_{ij}/2) \text{ (or } \alpha > \theta_{ij}/2) \tag{9}$$

$$\cos\beta < \cos(\theta_{ij}/2) \text{ (or } \beta > \theta_{ij}/2) \tag{10}$$

Adding (9) to (10) we get $\alpha + \beta > \theta_{ij}$, which is a contradiction of the assumption $\theta_{ij} = \alpha + \beta$.

For both situations, there is at least one of the distances which are between the evader and pursuers reducing by no less than $2v\Delta t \cos(\theta_{ij}/2)$. Since the number of pursuers and the initial distances are finite, after a finite number of steps there must be at least one pursuer whose distance from the evader is less than or equal to one step unit, thus the evader is guaranteed to be captured. Considering the worst case in which each timestep only the distance between the evader and a different pursuer reduces, the upper bound $T_{\max} \leq \dfrac{(n+1)\max(d_i)}{2v\Delta t \cos(\max(\theta_{ij}/2))}$ will be natural and self-explanatory. ■

Theorem 3.1 and Theorem 3.2 provide recipes for chasing superior evaders. Simulations confirm that integrating such instructions into greedy algorithm[20] is reasonable for a large range of situations. If none of the remaining superior evaders satisfies any of the conditions, the algorithm will give up capturing such "no-hope" evaders and assign extra pursuers to inferior evaders. Here an inferior evader is an evader that is slower than at least one pursuer.

IV. Jamming Confrontation

In addition to problems caused by multiple superior evaders, there might exist jamming-estimation confrontation[17,19] between the players in complex PE games. In this section, we consider a min-max jamming-estimation confrontation between pursuers and evaders. Note that to theoretically maximize possible application of the double-sided jamming model, in this section, the analysis is continuous and in time-varying domain. When applied to PE game, after similar discrete approach for ordinary differential games discussed in section II, sub-optimal solutions can be obtained via computer implementation.

Suppose both pursuers and evaders believe that it is good to let the enemy's estimation error grow as large as possible. Similar to[17] and,[19] which built up unilateral min-max jamming models, we consider the following double-sided coupled time-variant system (assume all matrices have appropriate dimensions—which is based on Kalman Filter estimation)

$$
\begin{aligned}
\dot{x}^p(t) &= A^p(t)x^p(t) + B^p w^p(t) + \delta^e(t) \\
\dot{x}^e(t) &= A^e(t)x^e(t) + B^e w^e(t) + \delta^p(t) \\
y^p(t) &= C^p(t)x^p(t) + m^p(t) \\
y^e(t) &= C^e(t)x^e(t) + m^e(t) \\
\delta^e(t) &= D^e(t)(C^e(t)x^p(t) - C^e(t)\hat{x}^p(t) + m^e_\delta(t)) \\
\delta^p(t) &= D^p(t)(C^p(t)x^e(t) - C^p(t)\hat{x}^e(t) + m^p_\delta(t)) \\
\dot{\hat{x}}^p(t) &= A^p(t)\hat{x}^p(t) + K^p(t)(y^p(t) - C^p(t)\hat{x}^p(t)) \\
\dot{\hat{x}}^e(t) &= A^e(t)\hat{x}^e(t) + K^e(t)(y^e(t) - C^e(t)\hat{x}^e(t))
\end{aligned}
\tag{11}
$$

with the simplified initial conditions

$$
\begin{aligned}
\hat{x}^p(0) &= 0 \\
\hat{x}^e(0) &= 0
\end{aligned}
\tag{12}
$$

where $x^g(t)$ $(g \in \{p, e\})$ is the system state variable column vector, $\hat{x}^p(t)$ and $\hat{x}^e(t)$ are the corresponding estimates for pursuers and evaders, respectively. $A^g(t)$ $(g \in \{p, e\})$ is the system dynamics matrix. $C^p(t)$ and $C^e(t)$ are time-varying measurement matrices. $\delta^p(t)$ and $\delta^e(t)$ are the inputs generated by pursuers and evaders to interfere with the enemy's estimation, respectively. $y^p(t)$ and $y^e(t)$ are the observed measurements of pursuers and evaders, respectively. $m^p(t)$, $m^p_\delta(t)$, $m^e(t)$, and $m^e_\delta(t)$ are the standard white noise disturbances, not necessarily Gaussian. $B^g w^g(t)$ $(g \in \{p, e\})$ are the system disturbance inputs, respectively. $D^e(t)$, $D^p(t)$, $K^e(t)$ and $K^p(t)$ are the time-varying gain matrices which will be chosen by both sides according to a min-max differential game whose objective function is defined as

$$
\begin{aligned}
J = trace\{&M^p(t_f)E[e^p_1(t_f)(e^p_1(t_f))^{\mathrm{T}} - e^p_2(t_f)(e^p_2(t_f))^{\mathrm{T}}] \\
&+ M^p(t_f)\int_0^{t_f} E[e^p_1(\tau)(e^p_1(\tau))^{\mathrm{T}} - e^p_2(\tau)(e^p_2(\tau))^{\mathrm{T}}]d\tau\} \\
&- trace\{M^e(t_f)E[e^e_1(t_f)(e^e_1(t_f))^{\mathrm{T}} - e^e_2(t_f)(e^e_2(t_f))^{\mathrm{T}}] \\
&+ M^e(t_f)\int_0^{t_f} E[e^e_1(\tau)(e^e_1(\tau))^{\mathrm{T}} - e^e_2(\tau)(e^e_2(\tau))^{\mathrm{T}}]d\tau\}
\end{aligned}
\tag{13}
$$

where $M^e(t)$ and $M^p(t)$ are game parameters that are any positive definite matrices and can be chosen as identity matrices. e^p_1, e^p_2 are defined as

$$e^p = x^p - \hat{x}^p = e^p_1 + e^p_2 \tag{14a}$$

$$e^e = x^e - \hat{x}^e = e^e_1 + e^e_2 \tag{14b}$$

where

$$
\begin{aligned}
\dot{e}^p_1 &= (A^p(t) - K^p(t)C^p(t) + D^e(t)C^e(t))e^p_1 + B^p w^p(t) - K^p(t)m^p(t) \\
e^p_1(0) &= x^p_0 \\
\dot{e}^p_2 &= (A^p(t) - K^p(t)C^p(t) + D^e(t)C^e(t))e^p_2 + D^e(t)m^e(t) \\
e^p_2(0) &= 0
\end{aligned}
\tag{15}
$$

e_1^e and e_2^e are defined symmetrically. After some manipulation, it can be shown that

$$\begin{aligned} \dot{e}^p &= (A^p(t) - K^p(t)C^p(t) + D^e(t)C^e(t))e^p + B^p w^p(t) - K^p(t)m^p(t) + D^e(t)m_\delta^e(t) \\ \dot{e}^e &= (A^e(t) - K^e(t)C^e(t) + D^p(t)C^p(t))e^e + B^e w^e(t) - K^e(t)m^e(t) + D^p(t)m_\delta^p(t) \end{aligned} \tag{16}$$

The two team players are playing a min-max game, that is, the pursuer team will chose (K^p, D^p) and the evader team will chose (K^e, D^e) according to the following formulas

$$(K^{p*}, D^{p*}) = \arg \min_{(K^p, D^p)} \left\{J(K^p, D^p, K^{e*}, D^{e*})\right\} \tag{17}$$

$$(K^{e*}, D^{e*}) = \arg \max_{(K^e, D^e)} \left\{J(K^{p*}, D^{p*}, K^e, D^e)\right\} \tag{18}$$

Note that at first sight the cost function might not look intuitive. However, in jamming-estimating problems,[17] it is a common setting and can be explained as representing the case that both players are trying to maximally interfere with the enemy's estimation at some penalty such as energy cost or computation cost.

We have Theorem 4.1 and Theorem 4.2 for the solution of this confrontation game.

Theorem 4.1. *The cost function in (13) is equivalent to*

$$\begin{aligned} J &= trace[M^p(t_f)Q^p(t_f) + \int_0^{t_f} M^p(\tau)Q^p(\tau)d\tau] \\ &\quad - trace[M^e(t_f)Q^e(t_f) + \int_0^{t_f} M^e(\tau)Q^e(\tau)d\tau] \end{aligned} \tag{19}$$

Where $Q^p(t)$ and $Q^e(t)$ are solved differentially via

$$\begin{aligned} \dot{Q}^p(t) &= (A^p(t) - K^p(t)D^e(t) + D^e(t)C^e(t))Q^p(t) \\ &\quad + Q^p(t)(A^p(t) - K^p(t)D^e(t) + D^e(t)C^e(t))^{\mathrm{T}} + B^p(t)B^{pT}(t) + K^p(t)K^{pT}(t) - D^e(t)D^{eT}(t) \\ \dot{Q}^e(t) &= (A^e(t) - K^e(t)D^p(t) + D^p(t)C^p(t))Q^e(t) \\ &\quad + Q^e(t)(A^e(t) - K^e(t)D^p(t) + D^p(t)C^p(t))^{\mathrm{T}} + B^e(t)B^{eT}(t) + K^e(t)K^{eT}(t) - D^p(t)D^{pT}(t) \end{aligned} \tag{20}$$

with initial conditions

$$\begin{aligned} Q^p(0) &= E(x_0^p x_0^{pT}) \\ Q^e(0) &= E(x_0^e x_0^{eT}) \end{aligned} \tag{21}$$

Proof. Let $Q^p(t) = Q_1^p(t) - Q_2^p(t)$, where

$$Q_1^p(t) = E(e_1^p e_1^{pT}) \tag{22a}$$

$$Q_2^p(t) = E(e_2^p e_2^{pT}) \tag{22b}$$

We can see that

$$\begin{aligned} \dot{Q}_1^p(t) &= (A^p(t) - K^p(t)D^e(t) + D^e(t)C^e(t))Q_1^p(t) \\ &\quad + Q_1^p(t)(A^p(t) - K^p(t)D^e(t) + D^e(t)C^e(t))^{\mathrm{T}} + B^p(t)B^{pT}(t) + K^p(t)K^{pT}(t) \end{aligned} \tag{23}$$

$$\begin{aligned} \dot{Q}_2^p(t) &= (A^p(t) - K^p(t)D^e(t) + D^e(t)C^e(t))Q_2^p(t) \\ &\quad + Q_2^p(t)(A^p(t) - K^p(t)D^e(t) + D^e(t)C^e(t))^{\mathrm{T}} + D^e(t)D^{eT}(t) \end{aligned} \tag{24}$$

Subtract (24) from (23), the theorem follows. ∎

Theorem 4.2. *The following equilibrium solve* (17)–(18)

$$K^{p*}(t) = P^p(t)C^p(t) \tag{25a}$$

$$D^{p*}(t) = P^e(t)C^p(t) \tag{25b}$$

$$K^{e*}(t) = P^e(t)C^e(t) \tag{25c}$$

$$D^{e*}(t) = P^p(t)C^e(t) \tag{25d}$$

where $P^p(t)$ *and* $P^e(t)$ *are calculated as*

$$\begin{aligned} \dot{P}^p &= A^p P^p + P^p A^p - P^p(C^{pT}C^p - C^{eT}C^e)P^p + B^p B^{pT} \\ \dot{P}^e &= A^e P^e + P^e A^e - P^e(C^{eT}C^e - C^{pT}C^p)P^e + B^e B^{eT} \end{aligned} \tag{26}$$

With initial conditions as

$$\begin{aligned} P^p(0) &= Q^p(0) \\ P^e(0) &= Q^e(0) \end{aligned} \tag{27}$$

Proof. For simplicity, we drop the "(t)" attached to a symbol if it does not cause confusion. Denote

$$\Delta^p = P^p - Q^p \tag{28}$$

After some manipulation, we get the following equation

$$\begin{aligned} &(K^p - K^{p*})(K^p - K^{p*})^{\mathrm{T}} - (D^e - D^{e*})(D^e - D^{e*})^{\mathrm{T}} \\ &\quad = K^p K^{pT} - D^e D^{eT} + P^p(C^{pT}C^p - C^{eT}C^e) - K^p C^p P^p - P^p C^p K^p + D^e C^e P^p + P^p C^{eT} D^{eT} \\ &\quad = K^p K^{pT} - D^e D^{eT} + B^p B^{pT} + (A^p - K^p C^p + D^e C^e)Q^p + Q^p(A^p - K^p C^p + D^e C^e)^{\mathrm{T}} \\ &\qquad + (A^p - K^p C^p + D^e C^e)\Delta^p + \Delta^p(A^p - K^p C^p + D^e C^e)^{\mathrm{T}} - \dot{P}^p \\ &\quad = \dot{Q}^p - \dot{P}^p + (A^p - K^p C^p + D^e C^e)\Delta^p + \Delta^p(A^p - K^p C^p + D^e C^e)^{\mathrm{T}} \end{aligned} \tag{29}$$

which implies

$$\begin{aligned} \dot{\Delta}^p = {} & (A^p - K^p C^p + D^e C^e)\Delta^p + \Delta^p(A^p - K^p C^p + D^e C^e)^{\mathrm{T}} \\ & - (K^p - K^{p*})(K^p - K^{p*})^{\mathrm{T}} + (D^e - D^{e*})(D^e - D^{e*})^{\mathrm{T}} \end{aligned} \tag{30}$$

Consider $D^e = D^{e*}$, we will have

$$\dot{\Delta}^p = (A^p - K^p C^p + D^e C^e)\Delta^p + \Delta^p(A^p - K^p C^p + D^e C^e)^{\mathrm{T}} - (K^p - K^{p*})(K^p - K^{p*})^{\mathrm{T}} \tag{31}$$

According to,[21] we know $\Delta^p \le 0$ for $t \in [0, t_f]$, which implies

$$Q^p(K^p, D^{e*}) \ge P^p \tag{32}$$

Similarly, via letting $K^p = K^{p*}$, we have

$$Q^p(K^{p*}, D^e) \le P^p \tag{33}$$

Since when $D^e = D^{e*}$ and $K^p = K^{p*}$ hold simultaneously we have $P^p = Q^p$, we have

$$Q^p(K^{p*}, D^e) \le Q^p(K^{p*}, D^{e*}) \le Q^p(K^p, D^{e*}) \tag{34}$$

Since the two team players are symmetric, following a similar procedure and letting

$$\Delta^e = P^e - Q^e \tag{35}$$

we can have

$$Q^e(K^{e*}, D^p) \le Q^e(K^{e*}, D^{p*}) \le Q^e(K^e, D^{p*}) \tag{36}$$

or equivalently

$$-Q^e(K^e, D^{p*}) \le -Q^e(K^{e*}, D^{p*}) \le -Q^e(K^{e*}, D^p) \tag{37}$$

SinceM^p and M^e are positive definite matrices, by multiplying these two matrices with (34) and (37) respectively and add them, after some manipulations, the theorem readily follows. ■

The pursuers and evaders control strategies include applying the $K^{p*}(t)$, $D^{p*}(t)$, $K^{p*}(t)$, and $D^{p*}(t)$ calculated from Theorem 4.2 to (11) to calculate $\delta^p(t)$, $\delta^e(t)$, $\hat{x}^p(t)$, and $\hat{x}^e(t)$. When we substitute such estimated $\hat{x}^p$ and $\hat{x}^e$ for x^p and x^e in section II, suboptimal real-time solution can be obtained. The corresponding experiments are in section VI.

V. The Complete Algorithm

Section II–IV analyzed challenge 1–3 explained in Section I, respectively. Thus when we face a multi-pursuer multi-evader (there might be superior evaders) PE games under complex noisy environments with jamming confrontation, if we want to catch as many evaders as we can, the suggested complete strategy can be described by Algorithm 5.1. The algorithm can be modified according to different goals in the real world applications. Note that once an evader is assigned as the target of one or more pursuers, this evader and corresponding pursuers are called "assigned". Initially all evaders and pursuers are unassigned. An unassigned pursuer is designed as an available pursuer.

Algorithm 5.1

Step 1: According to initial values and 25a–25d, calculate the initial position estimations and emitting signals for all evaders and pursuers.

Step 2: If there is no non-captured evaders, go to Step 8. Else rank all non-captured superior evaders from the highest value to the lowest value. Starting from the highest value evader, according to Theorem 3.2, evaluate whether each superior evader can be guaranteed to be captured by unassigned pursuers (that is, available pursuers). When a superior evader's capture can be guaranteed, find out the least number of necessary pursuers and assign this superior evader to such pursuers as target. Label such pursuers as Cooperative.

Step 3: For each pursuer that has not any target, find the non-captured evader who is inferior to it but with the least absolute value of speed difference. If such evader exists, assign it to the corresponding pursuer as target. Check all evaders and if no evaders are assigned, go to Step 8.

Step 4: If there are still pursuers that have not any target, evaluate each unassigned non-captured superior evader to see whether some evader and unassigned pursuers satisfy Theorem 3.1, one evader by one evader. If yes, find out the least number of necessary pursuers and assign the evader as the target of the corresponding pursuers.

Step 5: If there is still pursuer(s) that has not any target, assign the nearest superior non-captured evader who is not assigned in Step 2 it as target. If no such evader exists, assign the nearest non-captured evader to it. Note that here we do not label such pursuer(s) as Guarantees. If the targets of all pursuers

Step 6: All Cooperative evaders move according to the strategies described by the proof of Theorem 3.2. All other pursuers move directly toward the target's current estimation of positions.

Step 7: According to 25a–25d, calculate the current position estimations and emitting signals for all evaders and pursuers. If there is no evader is captured at this timestep, go to Step 6. Else label captured evaders as captured, reset all other evaders and all pursuers as unassigned, and go to Step 2.

Step 8: PE game ends.

VI. Experiments

The scenario has four pursuers and four evaders, among whom there are two superior evaders. "Pi" stands for the ith pursuer, and "Ej" stands for the jth evader. The arrows are illustrations for initial moving directions of different platforms. Blue arrows stand for pursuers, and red arrows are for evaders. The initial positions of all platforms are the same for all figures so that comparisons are more straightforward. A pursuer's waypoint is denoted as a circle, and an evader's waypoint is a triangle. Only $E1$ and $E4$ are superior evaders. According to Theorem 3.1 and Theorem 3.2, both $E1$ and $E4$ are possible to be captured. If an evader is captured, it will not move any more. However, the pursuer who captures it can continue to chase other evader(s).

Scenarios with and without jamming are simulated and compared. To reflect the intuitive comprehension that in some public PE games, such as police-criminal situations, evaders often pay more attention to confusing pursuing enemies than pursuers do, M^e has larger norms than M^p in the weighted jamming min-max objective function.

Fig. 3 records the procedure of perfect-state 4-pursuer 4-evader game. This is a deterministic situation and no noise or jamming is applied. Initially the algorithm assigns $P1$, $P3$, and $P2$, which are the "fan-controlling" pursuers (see Fig. 1), to chase $E1$ first. After a while, $P2$ successfully captured $E1$. After $E1$ is captured, the pursuers are assigned to new tasks, and finally $E2$ and $E3$ are captured, too. However, $E4$ escapes. This is because chasing $E1$ needs most pursuers (three out of four, according to Theorem 3.2) and there are not enough pursuers for chasing $E4$. At the time when pursuers are chasing $E1$, $E4$ smartly goes to a place which does not satisfy Theorem 3.1 or Theorem 3.2 any more. Consequently, when pursuers finished capturing $E1$, they lose the chance of capturing $E4$. Thus after capturing other three inferior evaders, the pursuers chose to stop the endless chasing game.

Fig. 4 records the chasing procedure integrated with jamming confrontation. In this case the chasing procedure looks somewhat similar to the chasing procedure without jamming. The result of this case is also that all evaders except $E4$ are captured. However, the jamming increases the difficulty faced by both sides. The noises made the chasing paths twisted, and sometimes the estimated locations are not even "continuous enough". In addition, the increased efforts spent on jamming the enemy by evaders are paid since the filtered locations of evaders looks more fuzzy than pursuers'.

Fig. 5 records a slightly more complex jamming case in which the jamming successfully enables $E1$ to escape from the evaders. In this case, initially the algorithm also assigns $P1$, $P3$, and $P2$, which are the "fan-controlling" pursuers (see Fig. 1), to squeeze the free space of $E1$. However, this time the jamming-noise effect makes the pursuers "miss" $E1$ (see the center point of Fig. 5). Once after this missing, $E1$ gets out of the trap constructed by $P1$, $P3$, and $P2$ and the pursuers have to give up chasing $E1$ so that resources would not be wasted. Note that when $P1$, $P3$,

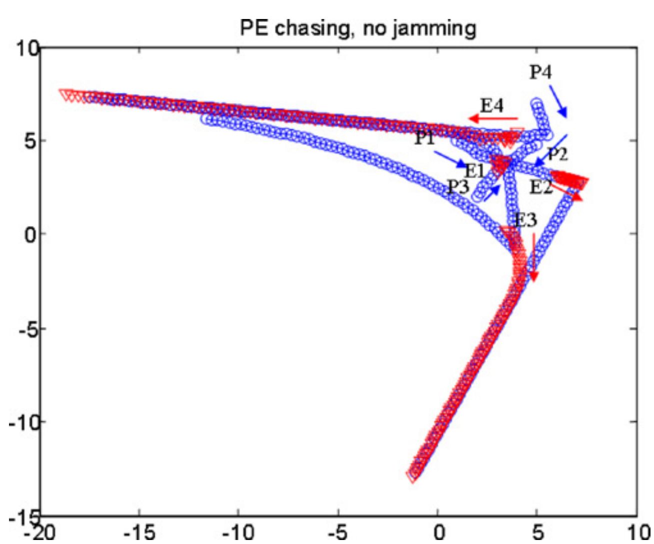

Fig. 3 PE chasing without jamming.

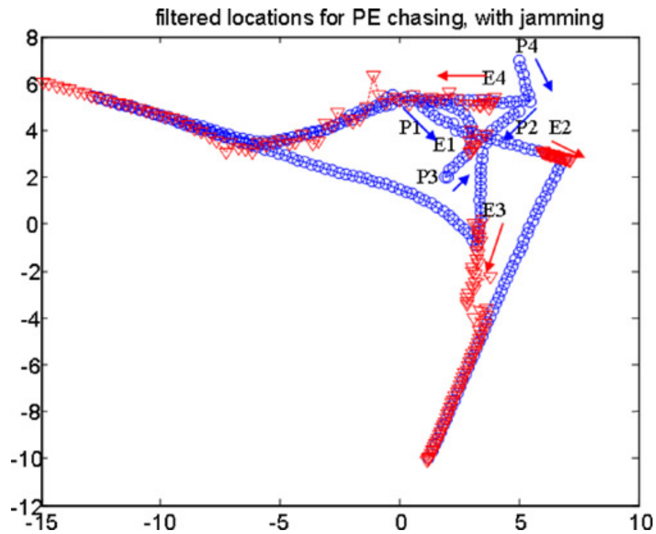

Fig. 4 PE chasing with jamming (three evaders are captured).

and $P2$ try to capture $E1$, $E4$ gets a chance to move to safe area thus after giving up capturing $E1$ pursuers also gave up chasing $E4$. Finally, only $E2$ and $E3$, which are all inferior evaders, are captured.

Fig. 6a and 6b record two computation time plots at each timestep, corresponding to Fig. 3 and Fig. 5, respectively. The horizontal axes are for timesteps. The vertical axes are for the calculation time at each timestep. Clearly, the peaks correspond to the coordination stages. The jamming calculations will consume more computation time, which is reflected in the comparison of this two time subplots.

Table 1 is a numerical comparison of total calculation time. It confirms that decentralized approaches require much less computation time than the centralized approach. Here the centralized approach is described in.[8]

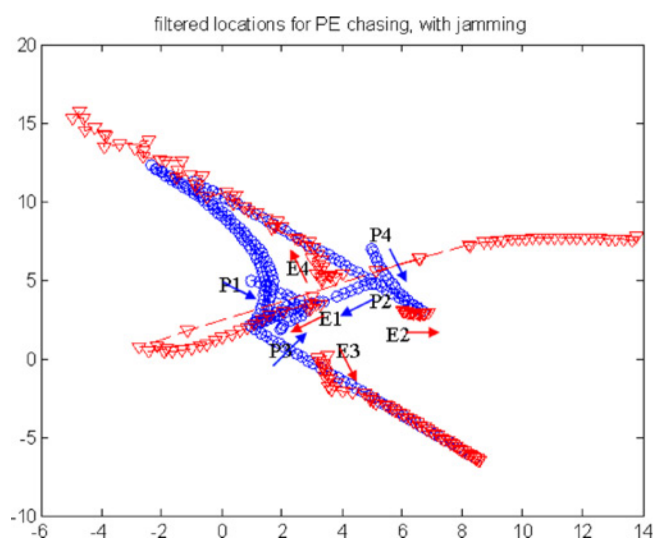

Fig. 5 PE chasing with jamming (only two evaders are captured).

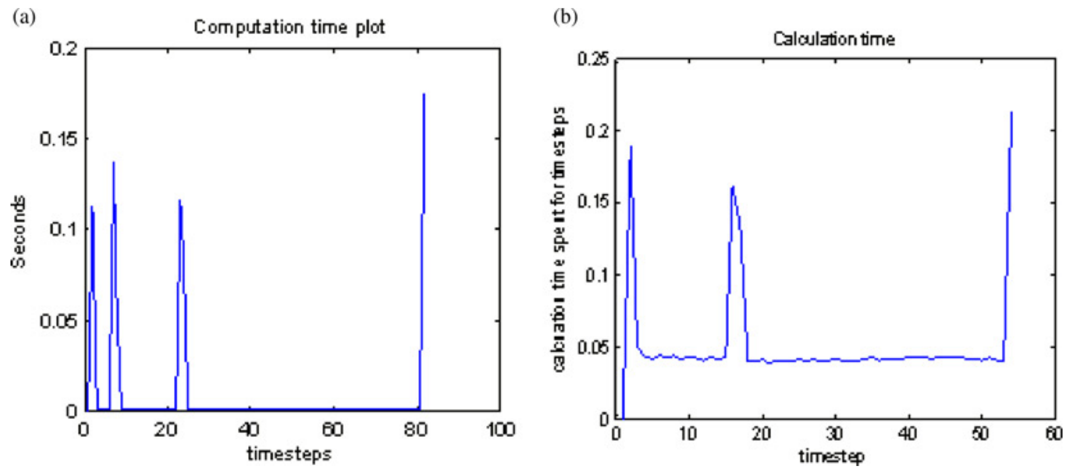

Fig. 6 (a) Calculation time plot, (b) Calculation time plot.

Table 1 Numerical comparison of calculation time.

Approach	With jamming?	Total timesteps	Computation time
Centralized approach	no	300	About 14 hours
Decentralized approach	no	82	0.7068 seconds
	yes	54	2.7407 seconds

VII. Conclusions

We proposed a decentralized approach to multi-player Pursuer-Evader Games with multiple superior evaders and jamming confrontation. The focus of our research is to deal with multiple superior evaders via a mechanism for real-time applications in complex jamming confrontation environments. We provided a sufficient condition for capturing superior evaders and modeled time-varying jamming confrontation in multi-player PE games. Simulation results confirm the advantages of our approach. Future research will extend the model so that pursuit-evasion strategies with jamming confrontation embedded can embrace situations with imperfect coordination among multiple pursuers.

Acknowledgment

This work was supported in part by the US ARMY under contract number W911NF-06-C-0016. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ARMY. The authors thank Dr. Mou-Hsiung Chang for his high degree of technical involvement in the project and support throughout the effort.

The authors appreciate and acknowledge the help of Ms. Xu Wang from the Ohio State University.

References

[1]Isaacs, Ruffs, *Differential Games*, John Wiley & Sons, Inc., New York, 1965.

[2]Sgall, J., "Solution of David Gale's Lion and Man Problem," available at http://www.math.cas.cz/~sgall/ps/lion.ps.

[3]Lee, J. H., Park, S. M., and Chwa, K. Y., "Searching a Polygonal Room with One Door by a 1-Searcher," *Int. J. Computat. Geom. Applicant.*, Vol. 10, No. 2, 2000, pp. 201–220.

[4]Isler, V., Kannan, S., and Khanna, S., "Randomized Pursuit-Evasion in a Polygonal Environment," *IEEE Transactions on Robotics*, Vol. 21, No. 5, October 2005.

[5]Mitchell, I. M. and Tomlin, C. J., "Approximating Reachable Sets by Hamilton-Jacobi Projections," *Journal of Scientific Computing*, Vol. 19, Nos. 1–3, December 2003.

[6]Kolmogorov, V. and Zabih, R., "Computing Visual Correspondence with Occlusions via Graph Cut," *IEEE Proc. of Int. Conference on Computer Vision*, Vol. II, 2001, pp. 508–515.

[7]Ficici, S. G. and Pollack, J. B., "Statistical Reasoning Strategies in the Pursuit and Evasion Domain," In D. Floreano, J. D. Nicoud, and F. Mondada, editors, *Fifth European Conference on Artificial Life*, 1999, pp. 79–88, Springer.

[8]Li, D., Cruz, J. B., Jr., Chen, G., Kwan, C., and Chang, M.-H., "A Hierarchical Approach To Multi-Player Pursuit-Evasion Differential Games", *Proceedings of Joint IEEE Conference on CDC-ECC05*, Spain, 2005.

[9]Zemskov, K. A. and Pashkov, A. G., "Construction of Optimal Position Strategies in a Differential Pursuit-Evasion Game with One Pursuer and Two Evaders," *Journal of Application of Maths and Mechs*, Vol. 61, No. 2, 1997, pp. 391–399.

[10]Pashkov, A. G. and Sinitsyn, A. V., "Construction of the Value Function in a Pursuit-Evasion Game with Three Pursuers and One Evader,"*Journal of Application of Maths and Mechs*, Vol. 59, No. 6, 1995, pp. 941–949.

[11]Maros, I., Nieminen, K, Ruuth, S., "Genetic Algorithm for Finding the First Integer Solution in Branch and Bound Algorithms," *20th IFIP Conference on System Modeling and Optimization*, Trier, Germany, July 2001.

[12]Nemhauser, G. L. and Wolsey, L. A., *Integer and Combinatorial Optimization*, John Wiley and Sons, Inc., New York, 1988.

[13]Nieminen, K., "Development of Mixed Integer Programming Methods," *Master's Thesis*, Systems Analysis Laboratory, Helsinki University of Technology, 2002.

[14]Taha, H., *Integer Programming*, Academic Press, Inc., New York, 1975.

[15]Crooks, P. and Perrott, R. H., "Language Construct for Data Partitioning and Distribution," *Technical Report*, Department of Computer Science, The Queen's University of Belfast, Belfast BT7 INN, Northern Ireland, 1994.

[16]Dongarra, J. J., Geijn, R. V., and Whaley, R. C., "Two Dimensional Basic Linear Algebra Communication Subprograms," editors, *Environments and Tools for Parallel Scientific Computing*, pp. 17–29, Elsevier, September 1992.

[17]Simon, D., "A Game Theory Approach to Constrained Minimax State Estimation," *IEEE Transactions on Signal Processing*, Vol. 54, No. 2, February 2006.

[18]Jain, A. K. and Dubes, R. C., *Algorithms for Clustering Data (Prentice Hall Advanced Reference Series: Computer Science)*, 1977, Prentice Hall.

[19]Yaesh, I. and Shaked, U., "Game Theory Approach to Optimal Linear State Estimation and Its Relation to the Minimum H-infinity norm Estimation," *IEEE Transactions on Automatic Control*, Vol. 37, No. 6, June 1992.

[20]Davis, G., Mallat, S., and Avellaneda, M., "Greedy Adaptive Approximation," *Journal of Construction and Approximation*, Vol. 13, 1997, pp. 57–98.

[21]Wonham, W. M., "On a Matrix Riccati Equation of Stochastic Control," *SIAM J. Contr.*, Vol. 6, 1968, pp. 681–697.

[22]Pineau, J., Gordon, G., Thrun, S., "Point-Based Value Iteration: An Anytime Algorithm for POMDPs," *International Joint Conference on Artificial Intelligence*, 2003.

[23]Smith, T. and Simmons, R., "Heuristic Search Value Iteration for POMDPs," *Proceedings of UAI*, 2004.

[24]Emery-Montemerlo, R., Gordon, G., Schneider, J., Thrun, S., "Approximate Solutions for Partially Observable Stochastic Games with Common Payoffs," *International Conference on Autonomous Agents and Multi-Agent Systems*, 2004.

[25]Li, D. and Cruz, J. B., Jr., "Better Cooperative Control with Limited Look Ahead," *Proceedings, 2006 American Control Conference*, Minneapolis, MN, June 14–16, 2006, CD ROM.

[26]Li, D. and Cruz, J. B., Jr., "Cooperative Pursuit of Multiple Evaders: On the Structure of Look-ahead Improvement and Decentralized Implementation," *Proceedings 45th IEEE Conference on Decision and Control*, San Diego, CA, December 2006.

Eytan Modiano
Associate Editor