12th International Conference on Information Fusion
Seattle, WA, USA, July 6-9, 2009

Sensor Attack Avoidance: Linear Quadratic Game Approach

Dongxu Li
GM R&D
Warren, MI
dongxu.li@gm.com

Genshe Chen
DCM Res. Res. LLC
Germantown,MD
gchen@dcmresearch-resources.com

Erik Blasch
AFRL/RYAA
WPAFB, OH
Erik.blasch@wpafb.af.mil

Khanh Pham
AFRL/RVSV
Kirtland AFB, NM
Khanh.Pham@kirtland.af.mil

Abstract − *For reliable and sustainable decision making, it is essential to perform intelligent sensing and data collection at scalable network resources costs. The sensor platforms used in a warfare may be under attacks from adversarial forces, which will largely impact the overall performance of surveillance systems. Thus, it is crucial that each intelligent sensor have the capability of detecting and avoiding possible attacks. In this paper, we study an attack-avoidance problem under the framework of a LQ game formulation. This is a first attempt to solve such kind of problems. From a practical point of view, the inherent hard constraints have been approximated and replaced by soft constraints with a fixed optimization horizon. For implementation, a receding horizon scheme has been used in junction with the LQ strategies. Overall, the LQ strategies can provide good control guidance laws for the players.*

Keywords: Tracking, game, linear quadratic, equilibrium.

1 Introduction

In modern military operations, it is desired to have heterogeneous sensor platforms and distributed warfare assets, which are strategically responsive, sustainable and survivable, and provide surveillance and situation awareness. It is therefore essential to perform intelligent sensing and data collection for reliable and sustainable decision making, at scalable costs to the network resources. To date, recent work seeks to reduce sensor management noise, communication overhead, computation complexity and scalability [1, 2]. However, the sensor platforms used in a warfare may be under attacks from adversarial forces, which will largely impact the overall performance of surveillance systems. Thus, it is crucial that each intelligent sensor have the capability of detecting and avoiding possible attacks, advocating for novel sensor fusion approaches to threat assessment (typically called Level 3 fusion) that account for sensor management constraints (typically called Level 4 fusion).

To avoid the complexity involved in networked sensors, as a first attempt, we study an attack-avoidance problem with only one sensor. The problem involves four entities: sensor, environment, target and attacker. Here, the sensor tracks a target in a given environment, which may be stationary or moves along its predetermined trajectory. An attacker wants to collide with the sensor to destroy it, while the sensor tries to avoid. This type of attacker-avoidance problem is new. Although sharing similarities with conventional tracking and object avoidance problems, it certainly has more ingredients. It involves both tracking and the conflict of pursuit and evasion. Control strategies designed purely for tracking or object avoidance becomes irrelevant.

This attacker-avoidance problem also shares some similarities with pursuit-evasion (PE) games. In a typical PE game, two players are present, i.e., a pursuer and an evader [1]. To study the optimal pursuit or evasion strategy, it is formulated as a zero-sum game. The pursuer tries to minimize a prescribed cost functional while the evader tries to maximize the same functional [3, 4]. Dynamic Programming (DP) is the a general method for solving such games. In the literature, a number of formal solutions regarding optimal strategies in particular PE problems have been achieved [3, 4, 5, 6]. Due to the development of Linear Quadratic (LQ) optimal control theory, a large portion of the literature focuses on PE differential games with a performance criterion in a quadratic form and linear dynamics [5, 7].

However, with an additional attacker, the existing results on conventional PE games are largely not applicable to the sensor attack-avoidance problem. Here, the point of interest is no longer pure pursuit or evasion. The refined strategy for the sensor is to continuously track the target while avoiding possible attacks (within certain period of time). To our knowledge, there is little direct literature on the attack-avoidance problem.

[1] Readers should be aware that pursuit-evasion games involving multiple pursuers and evaders have been studied in the literature.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JUL 2009		**06-07-2009 to 09-07-2009**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Sensor Attack Avoidance: Linear Quadratic Game Approach	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **GM R&D, ,Warren,MI**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002299. Presented at the International Conference on Information Fusion (12th) (Fusion 2009). Held in Seattle, Washington, on 6-9 July 2009. U.S. Government or Federal Rights License.

14. ABSTRACT
For reliable and sustainable decision mak- ing, it is essential to perform intelligent sensing and data collection at scalable network resources costs. The sensor platforms used in a warfare may be under attacks from adversarial forces, which will largely impact the overall performance of surveillance systems. Thus it is crucial that each intelligent sensor have the capa- bility of detecting and avoiding possible attacks. In this paper, we study an attack-avoidance problem under the framework of a LQ game formulation. This is a first attempt to solve such kind of problems. From a prac- tical point of view, the inherent hard constraints have been approximated and replaced by soft constraints with a fixed optimization horizon. For implementation, a receding horizon scheme has been used in junction with the LQ strategies. Overall, the LQ strategies can pro- vide good control guidance laws for the players.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Public Release**	**8**	

Missile guidance and navigation is another related research area. In a conventional navigation problem, control laws have been designed for an interceptor to track a moving target (with no attacker). The proportional navigation guidance law and its variants have been the most widely employed techniques for non-maneuvering targeting due to their simplicity and ease of implementation [8]. Another large class of guidance laws relevant to this problem are those designed based on optimal control theory, of which many are applications of LQ optimal control theory [8].

In this paper, we formulate the attack-avoidance problem as a zero-sum game between the sensor and the attacker. This is a first attempt to such a problem, and to avoid theoretical difficulties, we adopt a LQ formulation to make use of the existing LQ game theory. In particular, as a practical approach, terminal penalty terms are used as *soft constraints* in the adopted game completion. With additional assumptions on the linear dynamics of the players, LQ differential game theory is applicable. Furthermore, a practical approach to this emerging problem is developed with a sequential implementation scheme. The performance of the algorithm is demonstrated through simulations which validates the usefulness of the approach.

In the proposed LQ game approach, the key assumption and the main limitation is that the trajectory of the target (at least in the immediate future) is known to both the sensor and the attacker, which in many case, is not valid in sensor applications, due to the concern that collecting information is the main goal. However, the approach is still worth considering because the approach offers an opportunity of avoiding attacks while keeping track of a target for the sensor. In this sense, the assumption can be interpreted as the sensor's prediction of the target's movement. In a broader sense, the target here can also represent an uncertain area to be searched, and the "known trajectory" represents the areas of the highest interest.

The paper is organized as follows. In Section 2, an attack-avoidance game is formulated with linear dynamics and a quadratic objective based on soft constraints. Equilibrium strategies of the players are derived in section 3. An implementation scheme is then introduced in Section 4 to fill the gap between the LQ formulation and real-world attack-avoidance problems. In Section 5, we evaluate the performance of the proposed strategies by simulations and comparisons are drawn with the existing strategies. Concluding remarks are provided in Section 6.

2 Linear Quadratic Formulation with Soft Constraints

In this section, we formulate the attack-avoidance problem using soft constraints under the LQ framework with a fixed horizon. Consider a sensor, an attacker and a target in an n_S-dimensional space $S \subseteq \mathbb{R}^{n_S}$ with $n_S \in \mathbb{N}$. Let $x_s \in \mathbb{R}^{n_s}$, $x_a \in \mathbb{R}^{n_a}$ and $x_t \in \mathbb{R}^{n_t}$ be the state variables of the sensor, the attacker and the target respectively, with $n_s, n_a, n_t \geq n_S$. Suppose that each player in the game has its independent dynamics, which is described by the following linear equations respectively.

$$\dot{x}_s(t) = A_s x_s(t) + B_s' u_s(t) \quad \text{with} \quad x_s(t_0) = x_{s0} \tag{1a}$$
$$\dot{x}_a(t) = A_a x_a(t) + B_a' u_a(t) \quad \text{with} \quad x_a(t_0) = x_{a0} \tag{1b}$$
$$\dot{x}_t(t) = A_t x_t(t) + B_t' u_t(t) \quad \text{with} \quad x_t(t_0) = x_{t0} \tag{1c}$$

Here, $x_s(t) \in \mathbb{R}^{n_s}$, $x_a(t) \in \mathbb{R}^{n_a}$ and $x_t(t) \in \mathbb{R}^{n_t}$ for $t \geq t_0$; $u_s(t) \in U_s$, $u_a(t) \in U_a$ and $u_t(t) \in U_t$ are control inputs; $A_s, A_a, A_t, B_s', B_a', B_t'$ are real matrices with proper dimensions. Suppose that the first n_S elements of x_s (x_a, x_t) stand for the physical position of the sensor (attacker, target) in S. We can define a projection operator $P : \mathbb{R}^{n_s} \mapsto S$ for the sensor as

$$P(x_s) = [x_{s1}, \cdots, x_{sn_S}]^{\mathrm{T}} \in S. \tag{2}$$

That is, $P(x_s)$ gives the sensor's position in S. Similar operators can also be defined for both the attacker and the target, and here we use the same notation P.

For simplicity, we use the following aggregate dynamic equation.

$$\dot{\bar{x}}(t) = \bar{A}\bar{x}(t) + \bar{B}_s u_s(t) + \bar{B}_a u_a(t), \tag{3}$$

where

$$\bar{x} = \begin{bmatrix} x_s \\ x_a \end{bmatrix}, \bar{A} = \begin{bmatrix} A_s & 0 \\ 0 & A_a \end{bmatrix}, \bar{B}_s = \begin{bmatrix} B_s' \\ 0 \end{bmatrix},$$
$$\bar{B}_a = \begin{bmatrix} 0 \\ B_a' \end{bmatrix}.$$

Define $x \triangleq [x_s^{\mathrm{T}}, x_a^{\mathrm{T}}, x_t^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^n$ with $n = n_s + n_a + n_t$. We assume that each player can access the state x at any time t, and in this paper, feedback strategies are considered. Let $\gamma_s : \mathbb{R}^n \times \mathbb{R} \mapsto U_s$ and $\gamma_a : \mathbb{R}^n \times \mathbb{R} \mapsto U_a$ denote the strategy of the sensor and the attacker respectively. Given $x \in \mathbb{R}^n$ and time $0 \leq t < T$, $\gamma_s(x,t) \in U_s$, $\gamma_a(x,t) \in U_a$. Denote by Γ_s, Γ_a the set of admissible feedback strategies for each player.

We consider the objective functional of the following form.

$$\begin{aligned} J(\gamma_s, \gamma_a; x_0) = \int_0^T \Big(& u_s(\tau)^{\mathrm{T}} u_s(\tau) - u_a^{\mathrm{T}}(\tau) u_a(\tau) \\ & + w_s^I \|P(x_s(\tau)) - P(x_t(\tau))\|^2 \\ & - w_a^I \|P(x_a(\tau)) - P(x_s(\tau))\|^2 \Big) \mathrm{d}t \\ & + w_s \|P(x_s(T)) - P(x_t(T))\|^2 \\ & - w_a \|P(x_a(T)) - P(x_s(T))\|^2 \end{aligned} \tag{4}$$

In (4), γ_s, γ_a are feedback strategies; u_s, u_a are the control inputs associated with each corresponding strategy;

$w_s > 0$,$w_a > 0$, w_s^I and $w_a^I > 0$ are weighting scalars associated with the relevant costs induced by the distance between the sensor and the target and that between the attacker and the sensor. In this formulation, the distance between the attacker and the sensor is used as a penalty term to approximate the "hard constraint" of the problem, which mandates that the sensor stay out of reach of the attacker. The use of a penalty term is a common approach in optimal control and differential game theory to deal with hard constraints, especially under the LQ framework [9]. Note that a penalty on the distance between the sensor and the target is also included, which enables the sensor to closely track the target (while avoiding the attacker) under the resulting strategy. The fixed time duration T and scalars w_s,w_a,w_e^I, w_p^I are design parameters, and their values are case dependent.

The objective J in (4) can be rewritten in a quadratic form with respect to x, u_s and u_a, i.e.,

$$J(\gamma_s,\gamma_a;x_0) = \int_0^T \Big(u_s(\tau)^{\mathrm{T}}u_s(\tau) - u_a^{\mathrm{T}}(\tau)u_a(\tau) + x^{\mathrm{T}}(\tau)Qx(\tau)\Big)\mathrm{d}t + x^{\mathrm{T}}(T)Q_f x(T), \quad (5)$$

where Q can be defined through mapping $\hat{Q}_2$: $Q = \hat{Q}_2(w_s^I, w_a^I)$, $Q_f = \hat{Q}_2(w_s, w_a)$, where $\hat{Q}_2 : \mathbb{R}\times\mathbb{R} \mapsto \mathbb{R}^{n\times n}$ ($n = n_s + n_a + n_t$) is defined in (6) below.

$$\hat{Q}_2(w_s,w_a) = \left[\begin{array}{cc|c} (w_s-w_a)I^{n_S}_{n_s\times n_s} & w_a I^{n_S}_{n_s\times n_a} & -w_s I^{n_S}_{n_s\times n_t} \\ w_a I^{n_S}_{n_a\times n_s} & -w_a I^{n_S}_{n_a\times n_a} & 0_{n_a\times n_t} \\ \hline -w_s I^{n_S}_{n_t\times n_s} & 0_{n_t\times n_a} & w_s I^{n_S}_{n_t\times n_t} \end{array}\right] \quad (6)$$

In (6), $I^{n_0}_{n_1\times n_2}$ is an $n_1\times n_2$ matrix, in which the first n_S rows and n_S columns form an identity matrix, and the rest of the entries are zero.

This attack-avoidance game is a zero-sum game, where the sensor seeks a strategy $\gamma_s \in \Gamma_s$ to minimize J subject to (3), while the attacker tries to maximize J with $\gamma_a \in \Gamma_a$. The game can be viewed as a dual tracking problem, where the sensor wants to track the target but to avoid the attacker, and at the same time, the attacker needs to follow the sensor closely.

3 Game Solution for the Attack-Avoidance Problem

3.1 Review of LQ Game Theory

We first introduce players' saddle-point equilibrium strategies in a two-player LQ different game. This will be the major tool that we rely on in this paper. Let us consider a game involving two players with the following linear dynamics

$$\dot{x}(t) = Ax(t) + B_1u_1(t) + B_2u_2(t). \quad (7)$$

Note that here x is the state variable in this game, and different from aggregate state x defined above. The meaning of this notation is only true in this section. The objective function is given as

$$J = \int_0^T \Big(u_1(\tau)^{\mathrm{T}}u_1(\tau) - u_2^{\mathrm{T}}(\tau)u_2(\tau) + x^{\mathrm{T}}(\tau)Qx(\tau)\Big)\mathrm{d}\tau + x^{\mathrm{T}}(T)Q_f x(T). \quad (8)$$

The following LQ theorem specifies saddle-point equilibrium feedback strategies for both players.

Theorem 1. *The game with players' dynamics in (7) and the objective J in (8) admits a feedback saddle-point solution given by $u_1^*(t) = \gamma_1^*(x(t),t) = K_1^*(t)x(t)$ and $u_2^*(t) = \gamma_2^*(x(t),t) = K_2^*(t)x(t)$ with $K_1^*(t) = -B_1^TZ(t)$ and $K_2^*(t) = B_2^TZ(t)$, where $Z(t)$ is bounded, symmetric and satisfies*

$$\dot{Z} = -A^TZ - ZA - Q + Z(B_1B_1^T - B_2B_2^T)Z \quad \text{with } Z(T) = Q_f. \quad (9)$$

Readers can refer to [4] and [10] for a detailed proof.

3.2 Game Solution for the Attack-Avoidance Problem

In the attack-avoidance game, we consider that the target that moves along a predetermined trajectory $x_t(\cdot)$ in $\mathbb{R}^{n_S}$. The movement of the target is known to both the sensor and the attacker.

Consider the dynamics of the sensor and the attacker in (1a)-(1b). Note that in (1), the target's control is known (with a known trajectory), and the game is played between the sensor and the attacker. By inspection of the objective (4), we find that an attack-avoidance game with an arbitrarily moving target is closely related to a LQ regulator problem with a reference state trajectory [11]. In what follows, we make use of this analogy to solve the game. The following theorem provides saddle-point strategies of the players.

Theorem 2. *Suppose that the target trajectory $x_t(t)$ is known. The attack-avoidance game with the dynamics in (1a) and (1b) and the objective J in (4) admits a feedback saddle-point solution under the strategies*

$$u_s^* = \gamma_s^*(x,t) = -\bar{B}_s^TZ_{11}\bar{x} - \bar{B}_s^Tb; \quad (10)$$

$$u_a^* = \gamma_a^*(x,t) = \bar{B}_a^TZ_{11}\bar{x} + \bar{B}_a^Tb, \quad (11)$$

where $\bar{B}_s, \bar{B}_a, \bar{x}$ are defined in (3); the $\bar{n}\times\bar{n}$ ($\bar{n} = n_s + n_a$) matrix Z_{11} are bounded and satisfies

$$\dot{Z}_{11} + \bar{A}^TZ_{11} + Z_{11}\bar{A} + Q_{11} - Z_{11}(\bar{B}_s\bar{B}_s^T - \bar{B}_a\bar{B}_a^T)Z_{11} = 0, \text{ with } Z_{11}(T) = Q_{f\,11}. \quad (12)$$

Here $Q_{11}, Q_{12}, Q_{f\,11}, Q_{f\,12}$ are the corresponding submatrices of the matrices Q and Q_f partitioned as

$$Q = \left[\begin{array}{c|c} Q_{11} & Q_{12} \\ \hline Q_{12}^T & Q_{22} \end{array}\right] \text{ and } Q_f = \left[\begin{array}{c|c} Q_{f\,11} & Q_{f\,12} \\ \hline Q_{f\,12}^T & Q_{f\,22} \end{array}\right],$$

with Q is defined in (5). Matrices Q_{11} and Q_{f11} are $\bar{n}\times\bar{n}$ matrices; $\bar{A}$ is given in (3); the time-varying vector b is specified by

$$\dot{b}(t) = \left[-\bar{A}^T + Z_{11}(\bar{B}_s\bar{B}_s^T - \bar{B}_a\bar{B}_a^T)\right] b(t) - Q_{12}x_t \quad (13)$$

with $b(T) = Q_{f12}x_t(T)$.

Proof. We use Theorem 1 to prove the theorem. For the time being, we temporarily assume that the trajectory of the target $x_t(\cdot)$ is generated by an autonomous linear system (without control) as

$$\dot{x}_t = A_t x_t \text{ with } x_{t0} \text{ given.} \quad (14)$$

Later, we will show that this assumption is not necessary.

Combining the dynamic equations (1a)-(1b) with (14), we can write an aggregate dynamic equation as

$$\dot{x}(t) = Ax(t) + B_s u_s(t) + B_a u_a(t), \quad (15)$$

where x, A, B_s, B_a are defined as

$$x = \begin{bmatrix} x_s \\ x_a \\ x_t \end{bmatrix}, A = \begin{bmatrix} A_s & 0 & 0 \\ 0 & A_a & 0 \\ 0 & 0 & A_t \end{bmatrix},$$
$$B_s = \begin{bmatrix} B_s' \\ 0 \\ 0 \end{bmatrix} \text{ and } B_a = \begin{bmatrix} 0 \\ B_a' \\ 0 \end{bmatrix}.$$

The objective is still the same as (5).

For a game with the objective in (5) and the players' dynamics in (15), Theorem 1 is applicable. That is, if the following Riccati equation

$$\dot{Z} = -A^{\mathrm{T}}Z - ZA - Q + Z(B_sB_s^{\mathrm{T}} - B_aB_a^{\mathrm{T}})Z \quad \text{with } Z(T) = Q_f \quad (16)$$

admits a solution Z over the interval $[0, T]$, the saddle-point strategies of the sensor and the attacker are given by

$$u_s^*(t) = -B_s^{\mathrm{T}}Z(t)x(t) \text{ and } u_a^*(t) = B_a^{\mathrm{T}}Z(t)x(t). \quad (17)$$

In (17), Z is a $n \times n$ matrix ($n = n_s + n_a + n_t$), and we now partition Z in the following way,

$$Z = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{12}^{\mathrm{T}} & Z_{22} \end{bmatrix}.$$

Here, Z_{11} is an $n_1 \times n_1$ matrix with $n_1 = n_s + n_a$; Z_{12} is an $n_1 \times n_t$ matrix; and accordingly, Z_{22} is an $n_t \times n_t$ matrix. Matrices Q and Q_f can also be partitioned in the same way into submatrices Q_{ij} and Q_{fij} with the same dimensions of Z_{ij} ($i, j \in \{1, 2\}$). Note the difference between A here and $\bar{A}$ in (3) as well as those between B_s, B_a and $\bar{B}_s, \bar{B}_a$. With the submatrices defined above, the Riccati equation (16) can be presented separately in terms of Z_{ij}, Q_{ij} and Q_{fij} as

$$\dot{Z}_{11} + \bar{A}^{\mathrm{T}}Z_{11} + Z_{11}\bar{A} + \bar{Q} - Z_{11}(\bar{B}_s\bar{B}_s^{\mathrm{T}} - \bar{B}_a\bar{B}_a^{\mathrm{T}})Z_{11} = 0, Z_{11}(T) = Q_{f11}; \quad (18)$$

$$\dot{Z}_{12} + Z_{12}A_t + \bar{A}^{\mathrm{T}}Z_{12} + Q_{12} - Z_{11}(\bar{B}_s\bar{B}_s^{\mathrm{T}} - \bar{B}_a\bar{B}_a^{\mathrm{T}})Z_{12} = 0, Z_{12}(T) = Q_{f12}; \quad (19)$$

$$\dot{Z}_{22} + Z_{22}A_t + A_t^{\mathrm{T}}Z_{22} + Q_{22} - Z_{12}(\bar{B}_s\bar{B}_s^{\mathrm{T}} - \bar{B}_a\bar{B}_a^{\mathrm{T}})Z_{12} = 0, Z_{22}(T) = Q_{f22}. \quad (20)$$

The advantage of partitioning the Riccati equation in this way is that the saddle-point strategy of the sensor (or the attacker) can be decomposed into two parts. Note that $x^{\mathrm{T}} = [\bar{x}^{\mathrm{T}}, x_t^{\mathrm{T}}]$. Accordingly, the sensor's optimal control in (17) can also be written as

$$u_s^* = -\bar{B}_s^{\mathrm{T}}Z_{11}\bar{x} - \bar{B}_s^{\mathrm{T}}Z_{12}x_t. \quad (21)$$

Next, we define $b \triangleq Z_{12}x_t$. Take the time derivative of b. Based on $\dot{Z}_{12}$ in (19), we obtain the differential equation of $b(t)$ below.

$$\begin{aligned} \dot{b}(t) &= \frac{\mathrm{d}}{dt}(Z_{12}x_t) = \dot{Z}_{12}x_t + Z_{12}\dot{x}_t \\ &= Z_{12}(A_tx_t) - (Z_{12}A_t + \bar{A}^{\mathrm{T}}Z_{12} + Q_{12})x_t \\ &\quad + Z_{11}(\bar{B}_s\bar{B}_s^{\mathrm{T}} - \bar{B}_a\bar{B}_a^{\mathrm{T}})Z_{12}x_t \\ &= \left[-\bar{A}^{\mathrm{T}} + Z_{11}(\bar{B}_s\bar{B}_s^{\mathrm{T}} - \bar{B}_a\bar{B}_a^{\mathrm{T}})\right] Z_{12}x_t - Q_{12}x_t \\ &= \left[-\bar{A}^{\mathrm{T}} + Z_{11}(\bar{B}_s\bar{B}_s^{\mathrm{T}} - \bar{B}_a\bar{B}_a^{\mathrm{T}})\right] b(t) - Q_{12}x_t \end{aligned} \quad (22)$$

The initial conditions for equation (22) is $b(T) = Z_{12}(T)x_t(T) = Q_{f12}x_t(T)$ can be easily derived. Here, $b(t)$ can be completely determined by solving (22). Thus, the saddle-point strategy in (21) is

$$u_s^* = -\bar{B}_s^{\mathrm{T}}Z_{11}\bar{x} - \bar{B}_s^{\mathrm{T}}b. \quad (23)$$

By inspection of (23) and (22), the saddle-point equilibrium strategy actually does not depend on the assumption of the target's linear dynamics given in (14). Finally, the saddle-point equilibrium strategy of the attacker u_a^* in (17) can be further derived to obtain

$$u_a^* = \bar{B}_a^{\mathrm{T}}Z_{11}\bar{x} + \bar{B}_a^{\mathrm{T}}b.$$

□

Remark 1. *The theorem can also be proved by solving the corresponding Hamilton-Jacobi-Isaacs equation associated with the game without the auxiliary assumption in (14).*

In Theorem 2, the saddle-point strategy of the sensor (or the attacker) has two terms. The first term is feedback that depends on the state variables of both the attacker (or the sensor) and itself. This part is the game strategy that is coupled with the attacker's (sensor's) reaction. The second term in is a feedforward term that solely depends on the target's motion.

4 On Implementation of the LQ Strategies in Practice

The discussion under the framework of LQ game above takes advantage of the availability of analytical solutions. However, its usefulness remains to be tested. The gap between the LQ game approach and a real-world attack-avoidance game lies in the fixed terminal time T in the formulation, which imposes a soft constraint on distances. To demonstrate the usefulness of the LQ formulation in practice, we propose a sequential implementation scheme as follows.

We choose $\Delta t > 0$ as the sampling time interval. At each sampling time $t_k = t_0 + k\Delta t$ for $k \in \{0, 1, 2, \cdots\}$, saddle-point equilibrium strategies γ_s^*, γ_a^* are solved over the interval $[t_k, t_k + T_k]$, where $T_k > \Delta t$ is the optimization horizon used in the quadratic objective (4). We will discuss shortly the choice of T_k and the related issue about the existence of solutions for the corresponding Riccati equation. The game strategies γ_s^*, γ_a^* are implemented for only the next Δt interval, i.e., $[t_k, t_k + \Delta t)$. At the following sampling time $t_k + \Delta t$, the same procedure is repeated. We call this implementation scheme LQ Receding Horizon Algorithm (LQRHA). The detailed calculation at each time t_k is given in Table 1, where w_s, w_a, w_s^I, w_a^I are the design parameters.

Table 1: Procedure at Each t_k in the LQRHA

1. Input: state x at time t_k
2. Obtain the parameters w_s, w_a (w_s^I, w_a^I) and T_k
3. Solve the saddle equilibrium feedback strategies γ_s^*, γ_a^* over the time interval $[t_k, t_k + T_k)$
4. Output: γ_s^*, γ_a^*

We now discuss how to choose a proper T_k, such that the corresponding Riccati equation (9) admits a bounded solution on $[0, T_k]$, or in other words, the interval $[0, T_k]$ contains no *escape time* [12]. A finite escape time (if it exists) of a Riccati equation can be determined in the way suggested by the following theorem. Note that since the problem here is time invariant, the existence of solutions over $[0, T_k]$ is essentially the same as that over $[t_k, t_k + T_k]$ regardless of t_k.

Theorem 3. *The Riccati Differential Equation (RDE) (9) has a bounded solution over $[0, T]$ if and only if the following matrix linear differential equation*

$$\begin{bmatrix} \dot{X}(t) \\ \dot{Y}(t) \end{bmatrix} = \begin{bmatrix} A & -S \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} X(t) \\ Y(t) \end{bmatrix}, \quad \begin{bmatrix} X(T) \\ Y(T) \end{bmatrix} = \begin{bmatrix} I_n \\ Q_f \end{bmatrix} \tag{24}$$

has a solution on $[0, T]$ with $X(\cdot)$ nonsingular over $[0, T]$. In (24), A, Q and $S = B_1 B_1^T - B_2 B_2^T$ are the corresponding matrices in (9). Moreover, if $X(\cdot)$ is invertible, $Z(t) = Y(t)X^{-1}(t)$ is a solution of (9).

Refer to [10], pp. 194 or [12], pp. 354 for a proof.

According to Theorem 3, we define a finite escape time $T_e > 0$ (if it exists) such that $T - T_e$ is the smallest time such that the matrix $X(T - T_e)$ at time $T - T_e$ is singular[2]. The escape time can help determine the optimization horizon T_k. Suppose that we know how to choose the optimization horizon $\hat{T}_k$ (a design variable in the LQ design approach) based on the system states without considering the existence of solutions for the Riccati equation, e.g., $\hat{T}_k = \mathcal{T}(x_k)$. Then, by solving the linear differential equation in (24), it may be checked whether $T_e \in [0, \hat{T}_k]$. If $T_e \notin [0, \hat{T}_k]$, then T_k can be chosen as $\hat{T}_k$; otherwise, T_k can be set as $T_k = T_e - \delta$ for some $\delta > 0$. With T_k chosen in this way, the Riccati equation in (9) is guaranteed to have a bounded solution over $[0, T_k]$. Here, T_e only needs to be calculated once because the equation (9) is not state dependent. On the other hand, we need to choose a proper sampling time Δt such that $T_k > \Delta t$.

5 A Numerical Example

In this section, we demonstrate the usefulness of the LQ strategies by solving a selected attack-avoidance game in $\mathbb{R}^2$.

5.1 Players with Simple Motion

Suppose that the sensor and the attacker have the following simple motion dynamics in $\mathbb{R}^2$, which are given in an $\bar{x}$-$\bar{y}$ coordinate as

$$\begin{cases} \dot{\bar{x}}_s = v_s \bar{u}_s \cos(\theta_s) \\ \dot{\bar{y}}_s = v_s \bar{u}_s \sin(\theta_s) \end{cases}; \begin{cases} \dot{\bar{x}}_a = v_a \bar{u}_a \cos(\theta_a) \\ \dot{\bar{y}}_a = v_a \bar{u}_a \sin(\theta_a) \end{cases} \tag{25}$$

and the initial states are known. Define $x_\varsigma = [\bar{x}_\varsigma, \bar{y}_\varsigma]^{\mathrm{T}}$ as an aggregate state, and the subscript $\varsigma \in \{s, a\}$ stands for sensor or attacker. In (25), $\bar{x}_\varsigma, \bar{y}_\varsigma$ are the displacements along the $\bar{x}$ and $\bar{y}$ axis; v_ς is the speed, which is a constant; $\bar{u}_\varsigma, \theta_\varsigma$ are the control inputs, where $\bar{u}_\varsigma \in [0, 1]$ is a scalar that determines the player's moving speed from 0 up to v_ς, and θ_ς is the moving orientation. To make use of the LQ approach, in the following, we use an equivalent dynamics in the following form.

$$\begin{bmatrix} \dot{\bar{x}}_s \\ \dot{\bar{y}}_s \end{bmatrix} = \begin{bmatrix} v_s & 0 \\ 0 & v_s \end{bmatrix} \begin{bmatrix} u_{s\bar{x}} \\ u_{s\bar{y}} \end{bmatrix}, \quad \begin{bmatrix} \dot{\bar{x}}_a \\ \dot{\bar{y}}_a \end{bmatrix} = \begin{bmatrix} v_a & 0 \\ 0 & v_a \end{bmatrix} \begin{bmatrix} u_{a\bar{x}} \\ u_{a\bar{y}} \end{bmatrix}. \tag{26}$$

In (26), $(u_{\varsigma\bar{x}}, u_{\varsigma\bar{y}})$ are the control inputs with the constraint $\sqrt{u_{\varsigma\bar{x}}^2 + u_{\varsigma\bar{y}}^2} \le 1$. Clearly, $(\bar{u}_\varsigma, \theta_\varsigma)$ and $(u_{\varsigma\bar{x}}, u_{\varsigma\bar{y}})$ forms a one-to-one mapping with $\theta_\varsigma \in [0, 2\pi)$.

[2]Note that the Riccati equation (9) is solved backwards since its value is given at the final time T. Hence, $T - T_e$ is used here.

The dynamics in (26) are linear in the inputs $u_\varsigma \triangleq [u_{\varsigma\bar{x}}, u_{\varsigma\bar{y}}]^{\mathrm{T}}$ but with an additional constraint on the boundedness. We still rely on the LQ approach to design the feedback control law γ_ς. To ensure the boundedness, the following nonlinear function $\varphi(\cdot)$ is used.

$$\varphi(r) = \begin{cases} r & \text{if } \|r\| \le 1 \\ r/\|r\| & \text{if } \|r\| > 1 \end{cases} \quad \text{for } r \in \mathbb{R}^m \text{ with } m \ge 1 \tag{27}$$

In the simulations, the actual control u_ς applied is $u_\varsigma = \varphi(\gamma_\varsigma(x))$.

5.2 Attack Avoidance Game

We consider an attacker-avoidance problem with a mobile target that moves along a specific trajectory, which is known to both the sensor and the attacker. The movement of the target is described by the following equation.

$$\begin{cases} \bar{x}_t = 0.5t; \\ \bar{y}_t = -0.5t - 5\sin(\frac{\pi}{5}t). \end{cases}$$

The speeds and the initial positions of the players are specified in Table 2. The units of the parameters can be arbitrary and not specified here.

Table 2: Simulation Parameters

	Sensor	Attacker	Target
Speed	1	1	
Initial Position	(-9, -4)	(-9, -9)	(2,2)

We apply the LQRHA algorithm to determine both the sensor's and the attacker's strategies. Let the sampling time interval $\Delta t = 0.1$ second. At each sampling time $t_k = t_0 + k\Delta t$, the optimization horizon T_k is chosen as 15 seconds for all k. The parameters in the objective functional (5) are chosen as $w_s = w_s^I = 10$ and $w_a = w_a^I = 100$. Here, $w_\varsigma^I = 10$ indicates that the distances between players are much more important than the control energy needed in the LQ formulation. The relative numbers between w_a, w_a^I and w_s, w_s^I are chosen to reflect the fact that avoidance of attacks is of greater importance than tracking.

Figure 1 depicts the players' trajectories under the LQ game strategies. Here, the arrows indicate the instantaneous moving directions of the trajectories at the end of the simulation. With the LQ game strategy, the sensor is able to follow the target and stay away from the attacker. On the other hand, the attacker can closely follow the sensor and well position itself between the sensor and the target. This is important because the attacker may lose its ability of reaching the sensor if it follows the sensor too closely, and we will see the case shortly where the attacker uses other strategies.

Next, we compare the LQ game strategies with alternative tracking strategies (purely designed based on

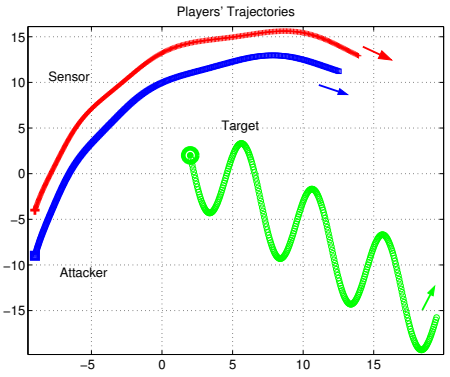

Figure 1: Players' Trajectories with the LQ Game Strategies

tracking problems) adopted by both the sensor and the attacker respectively. In each case, one player keeps its current LQ game strategy unchanged while the other player switches to another strategy. Two alternative strategies are considered, and both are well-known navigation strategies [8]. One is a LQ tracking strategy that is directly obtained by following the same procedure described in this paper, i.e., by solving an optimization problem with the objective function (4) where the weights w_a^I, w_a (or w_s^I, w_s) are set zero. With the zero weights on the selected penalty terms, it is no longer a game problem but an optimal tracking problem between the sensor and the target (or the attacker and the sensor). The other strategy to be considered is the so-called Line-Of-Sight (LOS) strategy, which will be specified shortly. Both of the strategies are designed to merely track an object of interest. Besides the changes in the players' strategies, all other parameters of the game remain the same, and a same length of the game time duration is used in the simulations.

We first simulate a scenario where the sensor still uses the same game strategy determined earlier, while the attacker switches to other strategies. The game result in Figure 2 shows the case where the attacker uses the LQ tracking strategy.

On the other hand, a possible LOS feedback strategy of the attacker is defined as follows.

$$u_a^{LOS} = v_a \frac{x_s - x_a}{\|x_s - x_a\|} \quad (x_s \neq x_a).$$

Figure 3 illustrates the simulation result when the attacker uses this LOS strategy.

In the both cases above, without prediction of the sensor's movement based on the motion of the target, the attacker loses its capability of intercepting the sensor considering that it moves at the same speed as the sensor.

Similar comparisons have also been drawn if the sensor deviates from its LQ game strategy. At this time, the attacker adopts the same LQ game strategy while

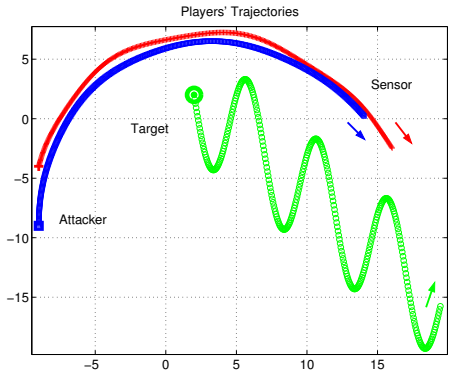

Figure 2: Players' Trajectories When the Attacker Uses the LQ Tracking Strategy

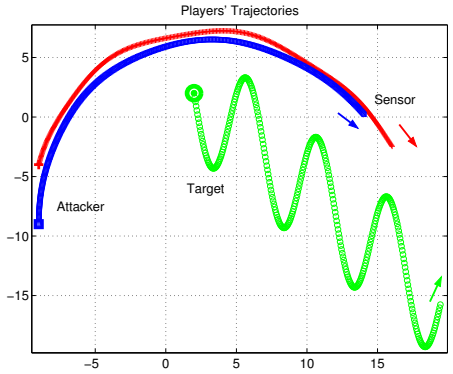

Figure 3: Players' Trajectories When the Attacker Uses the LOS Strategy

the sensor uses both the LQ tracking and the LOS strategy. The game results are plotted in Figure 4 and Figure 5 respectively. The LOS strategy for the sensor is given as

$$u_a^{LOS} = v_s \frac{x_t - x_s}{\|x_t - x_s\|} \quad (x_s \neq x_t).$$

In both cases, the sensor is intercepted by the attacker within the simulation time. Without considering the attacker, other tracking strategies should lead to a similar result.

Finally, Figure 6 shows the players' trajectories when the sensor implements an escaping strategy that is determined by solving the same game problem with the weights $w_s^I, w_s = 0$ in the objective function (4). Note that since $w_s^I, w_s = 0$, i.e., with no penalties on tracking the target, the main objective of the sensor is to escape from the attacker. Here, the attacker uses the same LQ game strategy.

Based on the simulation examples above, it is clear that the LQ game design provides for the sensor with a better compromised strategy between tracking and avoiding attacks. From the sensor's perspective, implementing both pure target tracking and escaping strategies has obvious disadvantages. Each represents an extreme of the entire spectrum of possible strategies from mere escaping to tracking. As seen in the simulations, without considering the attacker, the sensor under a pure tracking strategy is likely to be destroyed by the attacker. On the other hand, with an escaping strategy, tracking of the target has been given up. In both cases, the mission of tracking could be failed. The advantage of the LQ game approach is that the knowledge or a prediction of the target's future movement provides a

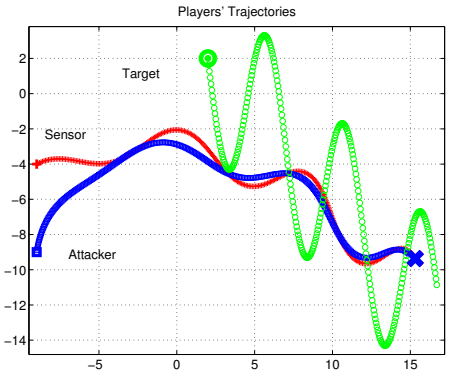

Figure 4: Players' Trajectories When the Sensor Uses the LQ Tracking Strategy

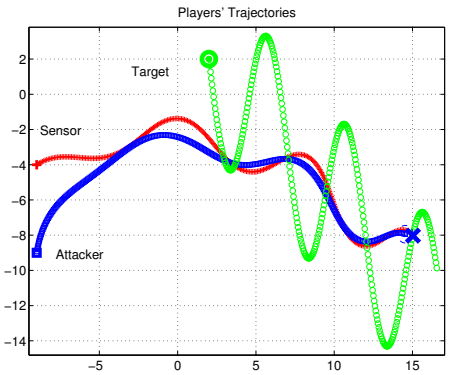

Figure 5: Players' Trajectories When the Sensor Uses the LQ Tracking Strategy

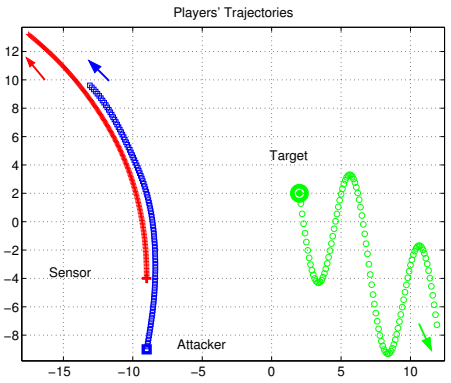

Figure 6: Players' Trajectories When the Sensor Uses the LQ Tracking Strategy

better chance for the sensor to avoid possible attacks while keeping the track of the target. When the danger of being attacked is eliminated, the sensor is still in a good position for tracking tasks under normal conditions.

Another observation is that the LQ game strategy also provides a better attacking strategy for the attacker. From the simulations, blindly going after the sensor can cost the attacker the chance of intercepting the sensor. The game strategy somehow predicts the sensor's movement and better aligns the attacker's movement with the sensor and the target in this three-entity game situation.

Based on a number of simulations, it is clear that the LQ strategy with the LQRHA implementation can provide fairly good guidance laws for both the sensor and the attacker.

6 Conclusions

In this paper, we have studied a attack-avoidance problem under the framework of a LQ game formulation. From a practical point of view, inherent hard constraints have been approximated and replaced by the soft constraints with a fixed optimization horizon. We have derived equilibrium strategies for both the sensor and the attacker. For implementation, a receding horizon algorithm called LQRHA has been proposed for application of the LQ strategies. Simulations have shown that this LQ game design can successfully provide for the sensor with a better compromised strategy between tracking and avoiding attacks, for which a traditional design can fail. Overall, the LQ strategies based on the LQRHA implementation can provide good control guidance laws for both players in this problem.

The main limitation of the approach is the assumption that the trajectory of the target is known to both the sensor and the attacker. In practice, this assumption can be interpreted as the sensor's prediction of the target's movement. In a broader sense, the target here can also represent an uncertain area to be searched, and the "known trajectory" may represent the areas of the highest interest.

Acknowledgement

This work was supported in part by the US Air Force under contracts FA9453-09-M-0061and FA9453-09-C-175.

References

[1] C. Kreucher, K. Kastella, and A. Hero, "Sensor management using an active sensing approach," *Signal Processing*, vol. 85, pp. 607–624, 2005.

[2] F. Zhao, J. Liu, L. Guibas, and J. Reich, "Collaborative signal and information processing: An information-directed approach," *Proceedings of IEEE*, vol. 91, pp. 1199–1209, 2003.

[3] R. Isaacs, *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit*. New York: John Wiley & Sons, Inc., 1965.

[4] T. Başar and G. Olsder, *Dynamic Noncooperative Game Theory*. Philadelphia: SIAM, second ed., 1998.

[5] Y. C. Ho, A.E. Bryson, Jr., and S. Baron, "Differential games and optimal pursuit-evasion strategies," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 385–389, 1965.

[6] W. Willman, "Formal solutions for a class of stochastic pursuit-evasion games," *IEEE Transactions on Automatic Control*, vol. 14, no. 5, pp. 504–509, 1969.

[7] V. Turetsky and J. Shinar, "Missile guidance laws based on pursuit-evasion game formulations," *Automatica*, vol. 39, no. 4, pp. 607–618, 2003.

[8] P. Zarchan, *Tactical and Strategic Missile Guidance*. Reston, Virginia: American Institute of Aeronautics and Astronautics, Inc., 2007.

[9] A.E. Bryson, Jr., *Dynamic Optimization*. Menlo Park, California: Addison-Wesley Longman, Inc., 1999.

[10] J. Engwerda, *LQ Dynamic Optimization and Differential Games*. NJ: John Wiley & Sons, Ltd, 2005.

[11] B. Anderson and J. Moore, *Optimal Control: Linear Quadratic Methods*. New Jersey: Prentice-Hall Inernational, Inc., 1989.

[12] T. Başar and P. Bernhard, *H^∞-optimal Control and Related Minimax Design Problems*. Boston: Birkhäuser, 2nd ed., 1995.