

## AN EFFICIENT ALGORITHM FOR A VISIBILITY-BASED SURVEILLANCE-EVASION GAME\*

RYO TAKEI<sup>†</sup>, RICHARD TSAI<sup>‡</sup>, ZHENGYUAN ZHOU<sup>§</sup>, AND YANINA LANDA<sup>¶</sup>

**Abstract.** We present an algorithm which computes the value function and optimal paths for a two-player *static* game, where the goal of one player is to maintain visibility of an adversarial player for as long as possible, and that of the adversarial player is to minimize this time. In a static game both players choose their controls at initial time and run in open-loop for  $t > 0$  until the end-game condition is met. Closed-loop (feedback strategy) games typically require solving PDEs in high dimensions and thus pose insurmountable computational challenges. We demonstrate that, at the expense of a simpler information pattern that is more conservative towards one player, more memory and computationally efficient static games can be solved iteratively in the state space by the proposed PDE-based technique. In addition, we describe how this algorithm can be easily generalized to games with multiple evaders. Applications to target tracking and an extension to a feedback control game are also presented.

**Key words.** Multi-player differential games, visibility, target tracking, motion planning, closed-loop v.s. open-loop, model predictive control.

**AMS subject classifications.** 91A23, 91A24, 91A25, 68T40, 49K35, 93A14, 93B52, 93B40.

### 1. Introduction

Consider the setting where a mobile agent is required to visually track a mobile target in an environment containing obstacles that block the agent's line of sight. For the mobile agent to maximize the time interval in which it maintains visibility of the target, a conservative, worst-case, strategy is to assume that the target actively seeks occlusion behind the obstacles. Hence, the problem may be posed as a pursuit-evasion-type game: the game terminates the first instance in time the pursuer loses sight of the evader, the pursuer's objective is to prolong the game as much as possible, and the evader's objective is to terminate the game as quickly as possible. We shall call this type of game a *visibility-based surveillance-evasion game*, or simply a surveillance-evasion game.

In this work, we investigate optimal motion strategies for two or more players moving with inhomogeneous and/or anisotropic velocities in a domain containing obstacles. The main contribution is a computationally and memory efficient algorithm for constructing the players' optimal paths for the *static* surveillance-evasion game. In this game the pursuer initially chooses its control for the remaining time, assuming that the evader would always counter with its best control. As we shall see, the particular information pattern of the static game allows for a decomposition of the problem into simpler optimal control problems in the underlying domain, which can be implemented in parallel. Hence, the feedback strategy games which are completely characterized by the Hamilton-Jacobi-Isaacs (HJI) partial differential equations are now decomposed into a series of optimal control problems, which are characterized

---

\*Received: January 24, 2012; accepted (in revised form): March 31, 2014. Communicated by Peter Smereka.

<sup>†</sup>Department of Electrical Engineering and Computer Sciences, UC Berkeley, USA (rrtakei@gmail.com).

<sup>‡</sup>Institute for Computational Engineering and Sciences, The University of Texas at Austin, USA (ytsai@ices.utexas.edu).

<sup>§</sup>Department of Electrical Engineering and Computer Sciences and Mathematics, UC Berkeley, USA (zhengyuan@berkeley.edu).

<sup>¶</sup>(yanalanda@gmail.com).

by the Hamilton-Jacobi-Bellman (HJB) equations, at the expense of a simpler and static information pattern. This decomposition can greatly save computational costs because solving a feedback strategy game, which amounts to solving an HJI equation in the joint configuration space, entails a computational complexity that increases exponentially with the number of participating players, whereas solving an HJB equation can be achieved in the configuration space in a single player.

The efficiency of the algorithm allows for various applications and extensions. For example, the algorithm generalizes easily to the game involving multiple evaders while maintaining a complexity which is linear in the number of players. Furthermore, our algorithm may be applied iteratively in small time increments to provide efficient solutions to a class of dynamic target tracking problems.

### 1.1. Previous work.

#### 1.1.1. Related problems involving surveillance and visibility.

Surveillance-evasion games were studied geometrically in [24, 20, 15], where the game terminates when the evader exits the pursuer's local "detection circle". It is unclear, however, how these geometric formulations could be applied in the context of the more general visibility-based surveillance-evasion game. Furthermore, the geometric nature of the derivations makes generalizations to arbitrary players' dynamics and the presence of state constraints difficult to analyze.

In a domain with polygonal obstacles, the numerical methods for the visibility based surveillance-evasion game with constant, isotropic velocities were studied in [5]. There, the authors geometrically characterized sets called "decidable regions" consisting of all possible pursuer initial states where the outcome of the game is known, given the initial state of the evader. The same authors, in [6], also considered computing optimal trajectories of the surveillance-evasion game backwards in time by tracing particular characteristic curves from the terminal condition set.

**1.1.2. Pursuit-evasion type game.** One of the initial applications of the pursuit-evasion games, formulated by R. Isaacs in the celebrated monograph [18], was missile guidance systems. Other applications of this problem include security and surveillance systems as well as search/rescue efforts. The proposed strategies could be used by autonomous systems or by human searchers.

In Section 2.2, we outline how the connection between differential games and HJI equations is established. The key idea is to introduce a value function of the game, which formally satisfies a particular PDE, known as the Hamilton-Jacobi-Isaacs (HJI) equation. The characteristic curves of the HJI equation correspond to the optimal paths of the players. The first treatment of such equations appeared in the monologue [18]. Subsequently, with the advent of the notion of viscosity solutions [8], rigorous theories regarding the existence and uniqueness of solutions (the value functions) to HJI equations were established; see [10] and the references therein. Numerical methods have been developed to compute the value and optimal paths for various games; see e.g. [4]. In practice, the high dimensionality of the HJI equation makes this approach tractable only on a modestly sized grid with a small number of players – usually two players in at most a two dimensional domain.

We also mention another class of problems, called visibility based pursuit-evasion, which concerns the motion planning of an agent that seeks out an initially hidden mobile target in a simply connected polygonal domain [33, 37, 13]. The simplified representation of the environment using polygons is a major limitation of the algorithms based on computational geometry and combinatorics. Furthermore, the extension of

these algorithms to three dimensional problems may be extremely complicated.

Probabilistic methods using a sampling-based approach has also been studied [19] in related but different contexts. There, they have considered a problem where an evader intends to reach a target under an adversarial pursuer whose goal is to intercept the evader. The aforementioned work is closer to the recent work [42] (involving two authors of the present manuscript), where a deterministic algorithm based on the modified fast marching methods is developed.

**1.1.3. Target tracking.** A related research area involving target tracking consists of a combination of different challenging problems that have been separately studied in robotics. Visibility and collision constraints must be satisfied in the presence of uncertainties in the positions of the observer and the target [21, 12, 25, 7, 40]. The observer would like to maintain visibility of the target while avoiding collisions with the obstacles [21]. When the target is fully predictable, that is, its trajectory is known, a dynamic programming approach [3] can be used to compute the shortest path for the observer. If the target is only partially predictable, this approach fails. Instead, the observer chooses the motion command which would maximize the likelihood that the target will remain visible during some time in the future [21, 12, 25]. In [22], a more complicated problem of tracking unpredictable targets in unknown environments is considered. The observer is able to reconstruct a local map of the environment based on range data. Then, a new velocity is determined using a combinatorial algorithm to minimize the risk of target escaping the observer's view by crossing an occlusion ray created by the obstacle. The general problem of inferring where the unpredictable moving targets could be as they pass out of view is presented in [40]. The main idea behind [40] is to introduce information spaces that extract and maintain combinatorial data obtained by robots that carry sensors. The proposed strategy [40] can be easily generalized to two or three dimensional, known or unknown, multiply connected domains, as long as the connected components of the shadow region are maintained.

The problem of target tracking in an arbitrary domain using a variational model was proposed in [9]. There, the pursuer's control was determined by optimizing a functional composed of a linear combination of the distance between the two players and the visibility area. Gradient descent was employed to lead the pursuer to a local minimizer of the functional.

**1.1.4. Optimal control and visibility in the Hamilton-Jacobi-Bellman framework.** Our approach to the visibility based surveillance-evasion game is based on decomposing the problem into optimal control problems. Thus, its efficiency hinges on that of the solution for the corresponding optimal control problems. To this end, we exploit recent developments in numerical Hamilton-Jacobi-Bellman equation solvers to determine the minimum time-to-reach function associated with the optimal control problem [35, 31, 1, 30, 38, 32, 14, 11]; optimal controls and paths can then be extracted from the minimum time-to-reach function by means of solving an ordinary differential equation along the characteristic curves; see the appendix of [1].

Another useful tool, the level-set based continuous visibility function  $\varphi(\cdot, \cdot)$  [9, 34] has the property that for any two points  $x, x_0$ ,

$$\varphi(x, x_0) \leq 0 \iff x \text{ is occluded from } x_0,$$

when obstacles are suitably defined implicitly as a level set function; i.e.  $\varphi$  is zero on the obstacle's boundary, negative inside, and positive outside. This formulation allows

for accurate description with subcell resolution of the the visibility information that is convenient for Boolean operations of the described regions as well as for integration and differentiation. Numerically,  $\varphi$  can be computed in  $O(N)$  complexity, on a grid with  $N$  nodes, by solving a Hamilton-Jacobi-Bellman equation with special sweeping techniques [34]. In the appendix, we extend the notion of a visibility function to that of a shadow function:

$$\xi(x, x_0) \leq 0 \quad \Leftrightarrow \quad x \text{ is occluded from } x_0 \text{ and } x \text{ is not inside an obstacle.}$$

**1.2. Outline of article.** The article will proceed as follows. In Section 2, we lay out all necessary assumptions as well as give a precise definition of a static information pattern for a two-player game. Next, in Section 3, we present the theory and algorithm for the static surveillance-evasion game, as well as numerical results demonstrating its performance. Applications and extensions of our algorithm to multiple evaders, target tracking, and feedback control games are presented in Section 4.

## 2. Information patterns and game definitions

**2.1. Preliminary definitions.** We partition an open and bounded ambient space  $\Omega$  into the *freespace* and *obstacles*:  $\Omega = \Omega_{\text{free}} \cup \Omega_{\text{obs}}$ , where  $\Omega_{\text{free}}$  is open;  $\Omega_{\text{obs}}$  serves as impenetrable obstacles to all players and blocks the line of sight between any two players. For a vantage point  $x \in \Omega_{\text{free}}$ , we denote  $\mathcal{D}_x \subset \Omega_{\text{free}}$  to be the relatively closed<sup>1</sup> subset in  $\Omega_{\text{free}}$  of all points that are not visible from  $x$ .

We shall refer to the two competing players as E (for ‘Evader’) and P (for ‘Pursuer’), with fixed initial states  $x_E^0, x_P^0 \in \Omega_{\text{free}}$ , respectively. Let  $A = \{\hat{\sigma} \mid \|\hat{\sigma}\|_2 = 1 \text{ or } \hat{\sigma} = \mathbf{0}\} \subset \mathbb{R}^n$  be the compact set of control values and  $\mathcal{A} = \{\text{measurable } \sigma: [0, \infty) \rightarrow A\}$  be the set of time-parametrized controls. Let  $x_E: [0, \infty) \rightarrow \Omega_{\text{free}}$  and  $x_P: [0, \infty) \rightarrow \Omega_{\text{free}}$  represent the time-evolution of E and P’s states, respectively. We assume the players move according to the following geometric dynamics:

$$\begin{cases} \dot{x}_E(t) = f_E(x_E(t), \sigma_E(t))\sigma_E(t), \\ \dot{x}_P(t) = f_P(x_P(t), \sigma_P(t))\sigma_P(t), \end{cases} \quad \text{for } t > 0, \quad \begin{cases} x_E(0) = x_E^0, \\ x_P(0) = x_P^0, \end{cases} \quad (2.1)$$

where  $f_E: \Omega_{\text{free}} \times A \rightarrow \mathbb{R}$ ,  $f_P: \Omega_{\text{free}} \times A \rightarrow \mathbb{R}$  are speeds assumed to satisfy the following:

- (A1)  $f_E, f_P$  are Lipschitz continuous.
- (A2) There exist constants  $F_1, F_2 > 0$  such that  $F_1 < f_E(x, \hat{\sigma}), f_P(x, \hat{\sigma}) < F_2$  for all  $x \in \Omega_{\text{free}}$  and  $\hat{\sigma} \in A$ .
- (A3) For every  $x_E, x_P \in \Omega_{\text{free}}$  the interior of the *vectograms*  $\{f_E(x_E, \hat{\sigma}_E)\hat{\sigma}_E \mid \hat{\sigma}_E \in A \setminus \{\mathbf{0}\}\}, \{f_P(x_P, \hat{\sigma}_P)\hat{\sigma}_P \mid \hat{\sigma}_P \in A \setminus \{\mathbf{0}\}\}$  are convex.

Assumption (A1) implies a unique solution to (2.1) for a pair of controls  $\sigma_E \in \mathcal{A}, \sigma_P \in \mathcal{A}$  up to some finite time; we refer to these solutions  $x_E(\cdot), x_P(\cdot)$  as *paths* for E and P, respectively. To avoid cumbersome notation, we will not write the explicit dependence of  $x_E(\cdot)$  on  $\sigma_E$  and  $x_P(\cdot)$  on  $\sigma_P$  and  $x_P^0$ . Assumption (A2) ensures that the minimum time-to-reach functions

$$u_E(x) = \inf_{\sigma_E \in \mathcal{A}} \inf\{t \geq 0 \mid x_E(t) = x\}, \quad (2.2)$$

<sup>1</sup>two players are considered occluded if they lie on each other’s *visibility horizon*, i.e. the line segment connecting the players that is tangent to an obstacle boundary.

$$u_P(x) = \inf_{\sigma_P \in \mathcal{A}} \inf\{t \geq 0 \mid x_P(t) = x\}, \tag{2.3}$$

for the single player optimal control problems, are continuous. Finally, the convexity assumptions of the interior of the vectograms in (A3) guarantee the existence of optimal controls in (2.2) and (2.3) and closedness of optimal paths (under uniform convergence over bounded-time intervals) in an unrestricted state space; for a remark on the existence of optimal controls for a restricted state space, see Remark 2.2. Note that the *zero control value*  $\mathbf{0} \in A$  gives the players the option to stop. As we shall see later, the existence of the zero control crucial is when characterizing the optimal control for P.

REMARK 2.1. It can be shown by a formal derivation that  $u_E$  satisfies the Hamilton-Jacobi-Bellman partial differential equation (PDE)

$$\max_{\sigma \in A} \{\nabla u_E(x) \cdot f_E(x, \sigma)\sigma\} = 1, \tag{2.4}$$

in  $\Omega_{\text{free}} \setminus \{x_E^0\}$ , and similarly for  $u_P$ . It is known that in general, even under the assumptions (A1)-(A3), Lipschitz continuous solutions of (2.4) with typical Dirichlet boundary conditions are non-unique. However, it is possible to characterize a generalized solution, called a *viscosity solution* [8]. It has been shown that the viscosity solution to (2.4) with  $u_E(x_E^0)$  coincides with the minimum time-to-reach function  $u_E$  of the corresponding optimal control problem [23]. The advantage of characterizing the minimum time-to-reach function as a HJB PDE is the availability of a plethora of efficient numerical methods known to converge to the viscosity solution, including fast sweeping-type [35, 30], fast marching-type [31, 38, 32], and semi-Lagrangian methods [14, 11]. See also [27] for a publicly available Matlab toolbox for computing solutions to a wide class of HJB equations.

REMARK 2.2. In a restricted state space (i.e.  $\Omega_{\text{free}} \neq \mathbb{R}^n$ ), the optimal controls in (2.2) and (2.3) may not exist, even under the assumptions (A1)-(A3). However, from the definition of the minimum time-to-reach function, there will always exist controls that are  $\epsilon$ -suboptimal, provided the minimum time-to-reach function is finite. For instance, given  $\epsilon > 0$ , there exists  $\sigma^* \in \mathcal{A}$  such that the corresponding path  $x_E^*(\cdot)$  for E satisfies

$$u_E(x) > \inf\{t \geq 0 \mid x_E^*(t) = x\} - \epsilon.$$

To simplify the presentation, we will loosely refer to an “optimal path” as a path that corresponds to an optimal control, if such a control exists, or an  $\epsilon$ -suboptimal control, if no optimal control exists. For detailed discussions on numerical computations of optimal trajectories and empirical demonstrations of its computation in a restricted state space, we refer the reader to Appendix A in [1].

**2.2. Feedback strategies and the value function.** E’s objective is to choose an optimal  $\sigma_E^* \in \mathcal{A}$  in order to minimize the payoff, while P’s objective is to choose an optimal  $\sigma_P^* \in \mathcal{A}$  to maximize the payoff. For instance, P’s strategy may be to move in the direction that locally increases the visibility area from the vantage point  $x_P(t)$  [9].

A *feedback strategy* at a time  $t \geq 0$  is a response rule of a player based on the system states up to  $t$ . For notational simplicity, write  $\mathbf{x}(\cdot) := (x_E(\cdot), x_P(\cdot))$  as the ordered pair of E’s and P’s paths. Let  $\mathcal{P} = \{\mathbf{x}(\cdot) : [0, \infty) \rightarrow \Omega_{\text{free}}^2 \mid \mathbf{x}(0) = (x_E^0, x_P^0)\}$  denote the set of paths for E and P from their respective initial states. Then, a feedback strategy for P is a map  $\zeta : \mathcal{P} \rightarrow \mathcal{A}$  that satisfies

1.  $\mathbf{x}(s) = \tilde{\mathbf{x}}(s)$  for all  $s \in [0, t]$  implies  $\zeta[\mathbf{x}](s) = \zeta[\tilde{\mathbf{x}}](s)$  for all  $s \in [0, t]$ ,
2. for all  $\sigma_E \in \mathcal{A}$  and  $T > 0$ , there exists a unique solution to (2.1) with  $\sigma_P(t) = \zeta[\mathbf{x}](t)$  for  $t \in (0, T]$ .

Let  $\mathcal{F}_P$  be the set of all feedback strategies of P, and let  $(x_E(\cdot), x_P(\cdot; \sigma_E, \zeta))$  be the solution to (2.1) corresponding to  $\sigma_E \in \mathcal{A}$  and  $\sigma_P = \zeta(x_E)$ , where  $\zeta \in \mathcal{F}_P$ . We then define the payoff function

$$\mathcal{J}_{\text{feedback}}(x_E^0, x_P^0, \sigma_E, \zeta) = \inf\{t \geq 0 \mid x_E(t) \in \mathcal{D}_{x_P(t; \sigma_E, \zeta)}\}.$$

In a *feedback strategy game*, at each time  $t \geq 0$ , one player chooses its control at  $t$  knowing the opponent's state from times 0 up to and including  $t$ . For the moment, we assume the player choosing the control is P.

DEFINITION 2.3. *The (lower) value function  $u: \Omega_{\text{free}}^2 \rightarrow \mathbb{R}$  for the feedback strategy game is*

$$u(x_E^0, x_P^0) := \sup_{\zeta \in \mathcal{F}_P} \inf_{\sigma_E \in \mathcal{A}} \mathcal{J}_{\text{feedback}}(x_E^0, x_P^0, \sigma_E, \zeta). \tag{2.5}$$

Here, for any of P's feedback strategies, E chooses an optimizing control, and P then picks out one strategy that has the best payoff. It can be shown that under certain conditions the value for such games is equal regardless of which player, E or P, chooses the control and which reacts by choosing feedbacks. Furthermore,  $u$  is a viscosity solution to the *Hamilton-Jacobi-Issacs (HJI) equation* [1, 8, 10]

$$1 = \max_{\hat{\sigma}_P \in \mathcal{A}} \min_{\hat{\sigma}_E \in \mathcal{A}} \{-f_E(x_E, \hat{\sigma}_E) \hat{\sigma}_E \cdot \nabla_{x_E} u - f_P(x_P, \hat{\sigma}_P) \hat{\sigma}_P \cdot \nabla_{x_P} u\} \quad \text{on } \Omega_{\text{free}}^2 \setminus \mathcal{T}, \tag{2.6}$$

where  $u = 0$  in  $\mathcal{T} = \{(x, y) \in \Omega_{\text{free}}^2 \mid x \in \mathcal{D}_y\}$  and  $\nabla_x u$  is the gradient of  $u$  with respect to the variable  $x \in \Omega_{\text{free}}$ .

In contrast to the feedback strategy game, a game where the feedback strategy is invoked only at  $t = 0$  is called a *static game*. That is, at the start of the game, one player chooses its control for all  $t \geq 0$  by countering the control chosen by the other player for all  $t \geq 0$ , and the other player optimizes knowing this response. Indeed, in such games, the optimal occlusion time may differ depending on which player (E or P) plays in response to the other. In this article, we focus on the case where P optimizes assuming that E will always optimally counter P's strategies. For initial states  $x_E^0, x_P^0$  and controls  $\sigma_E, \sigma_P$ , consider the payoff function

$$\mathcal{J}(x_E^0, x_P^0, \sigma_E, \sigma_P) = \inf\{t \geq 0 \mid x_E(t) \in \mathcal{D}_{x_P(t)}\}. \tag{2.7}$$

Here and throughout this article, we adopt the convention that the infimum of the empty set is infinity.

DEFINITION 2.4. *Define the (lower) value function  $v_s: \Omega_{\text{free}}^2 \rightarrow \mathbb{R}$  for the static game as*

$$v_s(x_E^0, x_P^0) := \sup_{\sigma_P \in \mathcal{A}} \inf_{\sigma_E \in \mathcal{A}} \mathcal{J}(x_E^0, x_P^0, \sigma_E, \sigma_P). \tag{2.8}$$

In general, it can be shown that  $v_s \leq u$  [2]. Our central result, presented in Section 3, is an efficient algorithm which computes  $v_s(x_E^0, x_P^0)$  and the paths corresponding to the optimal controls (or  $\epsilon$ -suboptimal controls, see Remark 2.2), formally written as

$$\sigma_P^* \in \arg \sup_{\sigma_P \in \mathcal{A}} \inf_{\sigma_E \in \mathcal{A}} \mathcal{J}(x_E^0, x_P^0, \sigma_E, \sigma_P), \tag{2.9}$$

$$\sigma_E^* \in \arg \inf_{\sigma_E \in \mathcal{A}} \mathcal{J}(x_E^0, x_P^0, \sigma_E, \sigma_P^*), \tag{2.10}$$

provided the game ends in finite time. For brevity, we shall refer to the static visibility-based surveillance-evasion game simply as the *static game* and the corresponding value (2.8) as the *static value*.

REMARK 2.5. Note that the value function can be analogously defined for a “dual” static game where E optimizes by assuming that P will always optimally counter, by swapping the “ $\inf_{\sigma_E \in \mathcal{A}}$ ” and “ $\sup_{\sigma_P \in \mathcal{A}}$ ” in (2.8); this will yield a(n) (upper) value function which will be greater than or equal to  $v_s$  and  $u$  [2].

In the differential game theory literature, a pursuit-evasion type game typically refers to (2.6), i.e. the boundary value PDE problem whose solution corresponds to the feedback strategy value function. The HJI equation (2.6) immediately suggests a numerical algorithm to solve the feedback strategy value function. However, because (2.6) is solved on the joint configuration space  $\Omega_{\text{free}}^2$ , even the simplest (non-trivial) setting of two players in a two dimensional ambient space leads to a problem in at least four dimensions. For problems with additional players, the dimension of the problem scales exponentially in the number of players. The high-dimensionality of the problem is attributed to the information contained in a single value function: the optimal control can be extracted from  $u$  for every pair of initial starting positions on the domain.

We emphasize that the objective of this article is not to solve  $u$ , but to solve the static game value function  $v_s$  (and the associated optimal paths), which is not known to be a solution to a PDE. For a problem of finding optimal controls for a single prescribed initial pair of positions  $(x_E^0, x_P^0)$ , solving (2.6) involves excessive amounts of computational time and data memory.

We now give a condition such that the static value function (2.8) is finite.

PROPOSITION 2.6. Assume that  $\Omega_{\text{free}}$  is not a star domain, and  $\sup_{x \in \Omega_{\text{free}}} u_E(x) < \infty$ . Then  $v_s(x_E^0, x_P^0) < \infty$ .

*Proof.* For  $\epsilon > 0$  small, let  $t_{\text{max}} = \sup_{x \in \Omega_{\text{free}}} u_E(x) + \epsilon$ . By the existence of  $\epsilon$ -suboptimal controls, for any  $y \in \Omega_{\text{free}}$ , there is a path  $x_E(\cdot)$  such that  $x_E(0) = x_E^0$  and  $x_E(t_{\text{max}}) = y$ .

Choose any  $\sigma_P^* \in \mathcal{A}$ , and let  $x_P^*(\cdot)$  be the corresponding path for P. Because the domain is not star shaped, there exists  $y \in \Omega_{\text{free}}$  such that  $x_P^*(t_{\text{max}}) \in \mathcal{D}_y$ . If in response, E chooses  $\sigma_E^* \in \mathcal{A}$  corresponding to a path  $x_E^*(\cdot)$  such that  $x_E^*(t_{\text{max}}) = y$  (which exists by the earlier argument), then  $\mathcal{J}(x_E^0, x_P^0, \sigma_E^*, \sigma_P^*) \leq t_{\text{max}}$ . Because E has a response control such that  $\mathcal{J} \leq t_{\text{max}}$  for any control choice of P, we conclude that  $v_s(x_E^0, x_P^0) \leq t_{\text{max}} < \infty$ . □

This proposition says that the static game will terminate in finite time, for example, if E travels with positive speed on a bounded, path connected  $\Omega_{\text{free}}$  that is not a star domain. This rather strong statement results from the fact that the entire future control values  $\sigma_P(t)$  and  $\sigma_E(t)$  for both players are chosen *only once at initial time*, and no future revision is allowed. Looking ahead, the last proposition is useful in justifying the use of our algorithm in the numerical examples discussed in Section 3. In Section 4.2, we present a scenario in which optimal controls for both players are allowed to be revised at discrete time increments as the game progresses.

We close this section by proving a simple observation, to be used later.

PROPOSITION 2.7. For a stationary  $P$ , i.e.  $x_P(t) = x_P^0, \forall t \geq 0$ , the optimal control for  $E$  for the static game is the time optimal control from  $x_E^0$  to the set  $\mathcal{D}_{x_P^0}$ .

*Proof.* Fix  $P$ 's control  $\sigma_P^0$  to have zero control value  $\mathbf{0} \in A$  for all  $t \geq 0$ . Then,

$$\begin{aligned} v_s(x_E^0, x_P^0) &= \inf_{\sigma_E \in \mathcal{A}} \mathcal{J}(x_E^0, x_P^0, \sigma_E, \sigma_P^0) \\ &= \inf_{\sigma_E \in \mathcal{A}} \inf\{t \geq 0 \mid x_E(t) \in \mathcal{D}_{x_P^0}\}, \end{aligned}$$

which precisely characterizes the optimal control of  $E$  as desired.  $\square$

**2.3. Notations and game assumptions.** Because we will consider ‘‘optimal paths/controls’’ in the context of both the differential game and the related optimal control problems, we choose to call *optimal paths/controls* (or  $\epsilon$ -suboptimal paths/controls) those which correspond to the static game, and *time optimal paths/controls* (or  $\epsilon$ -time suboptimal paths/controls) those which correspond to minimum arrival time optimal control problems, such as in (2.2) and (2.3).

Throughout this article, we use the following notations:

- $\Omega \subset \mathbb{R}^n$  is an open and bounded ambient space; we focus particularly with  $n = 2$ , but the theory applies naturally to higher dimensions.
- $\Omega_{\text{free}} \cup \Omega_{\text{obs}} = \Omega$  partitions the ambient space into the freespace and obstacles. We assume that  $\Omega_{\text{free}}$  is open.
- $E$  and  $P$  are the two competing players, representing the evader and the pursuer, respectively.
- $x, y$  are generic points in  $\Omega$ .
- $\mathcal{D}_x \subset \Omega_{\text{free}}$  is the set of occluded points from the vantage point  $x \in \Omega_{\text{free}}$ .
- $\sigma_E, \sigma_P: t \in [0, \infty) \rightarrow A$  are the controls of  $E$  and  $P$  at time  $t \geq 0$ , respectively.
- $x_E, x_P: t \in [0, \infty) \rightarrow \Omega_{\text{free}}$  are  $E$  and  $P$ 's respective positions at time  $t \geq 0$ ; we suppress their dependence on the initial states  $x_E^0, x_P^0$  and controls  $\sigma_E, \sigma_P$ .
- $x_E^0, x_P^0 \in \Omega_{\text{free}}$  are the fixed initial positions of  $E$  and  $P$ , respectively.
- $x_E^f, x_P^f \in \Omega_{\text{free}}$  are the final positions of  $E$  and  $P$ , respectively, to be computed.
- $v_s: \Omega_{\text{free}}^2 \rightarrow \mathbb{R}^+ \cup \{0\}$  is the value of the static game; see (2.8).

Furthermore, we shall make the following assumptions on the games introduced henceforth:

- (B1)  $x_E^0 \notin \mathcal{D}_{x_P^0}$ , i.e. the players are initially visible to each other.
- (B2) both players are aware of their opponent's dynamics,
- (B3) both players know their opponent's positions, if and only if they are visible to each other.

REMARK 2.8. Given a vantage point  $x \in \Omega_{\text{free}}$ , a level set function  $\xi(\cdot; x): \Omega \rightarrow \mathbb{R}$  representing  $\mathcal{D}_x \subset \Omega$ , where

$$\begin{cases} \xi(y; x) \leq 0, & \text{if } y \in \mathcal{D}_x, \\ \xi(y; x) > 0, & \text{if } y \in \Omega \setminus \mathcal{D}_x, \end{cases}$$

can be computed in  $O(N)$  time on a grid with  $N$  nodes, using the level set visibility function [34]. The details are described in Appendix A.



**3. An algorithm for the static value function**

In this section, we present an algorithm which computes the value function and optimal paths corresponding to the optimal controls (2.9) and (2.10). Our algorithm depends crucially on a maximal set  $R^* \subset \Omega_{\text{free}}$  and two functions  $t^*(\cdot; x_E^0)$  and  $w_{R^*}(\cdot; x_P^0)$  which, in some sense, decouple the analysis of the dynamics of E and P.

DEFINITION 3.1. *Given  $x \in \Omega_{\text{free}}$ ,*

$$t^*(x; x_E^0) := \inf_{\sigma_E \in \mathcal{A}} \inf\{t \geq 0 \mid x_E(t) \in \mathcal{D}_x, x_E(0) = x_E^0\}. \tag{3.1}$$

That is,  $t^*(x; x_E^0)$  is the shortest time for E, starting from  $x_E^0$ , to become occluded from a *stationary vantage point*  $x \in \Omega_{\text{free}}$ . Note that the two players must be initially visible to each other by assumption (B1), so  $t^*(x_P^0; x_E^0) > 0$ .

DEFINITION 3.2. *For any subset  $R \subset \Omega_{\text{free}}$  containing  $x_P^0$ , let*

$$w_R(x; x_P^0) = \inf_{\sigma_P \in \mathcal{A}} \inf\{t \geq 0 \mid x_P(0) = x_P^0, x_P(t) = x, x_P(s) \in R, \forall s \in [0, t]\}. \tag{\Delta R}$$

That is,  $w_R(x; x_P^0)$  is the minimum time for P to reach  $x$  from  $x_P^0$  by traveling along a path that is contained in  $R$ . By the small-time controllability assumption (A2), the function  $w_R(\cdot; x_P^0)$  is finite and continuous in any pathwise connected subset of  $R$  containing  $x_P^0$ . Next, we relate  $t^*$  and  $w_R$  by an inequality over a maximal set  $R^*$ .

DEFINITION 3.3. *Let  $R^* \subset \Omega_{\text{free}}$  be the maximal set containing  $x_P^0$  such that*

$$w_{R^*}(y; x_P^0) < t^*(y; x_E^0), \text{ for all } y \in R^*. \tag{3.2}$$

*In other words, for any  $R \supsetneq R^*$  and  $y \in R \setminus R^*$ , we have  $w_R(y; x_P^0) \geq t^*(y; x_E^0)$ .*

We remark that the maximality of  $R^*$  implies its uniqueness. The existence follows from the fact that property (3.2) holds trivially for the subset  $\{x_P^0\} \subset \Omega_{\text{free}}$ .

Roughly speaking, The functions  $t^*$  and  $w_{R^*}$  defined above quantify the reachability of the two players by time under optimal choice of controls; this allows us to characterize P’s possible controls (or paths) that maintain visibility of E under worst case scenarios; see figure 3.1.

**3.1.  $\tau$ -admissible pursuit paths.** By Proposition 2.7, assuming a stationary P, an optimal strategy for E is to move towards the occlusion set  $\mathcal{D}_{x_P^0}$  via a time-optimal path, provided it exists. In this case, it will take  $t^*(x_P^0; x_E^0)$  time for an optimally moving E to become occluded from P. This also implies that  $x_E(\tau) \notin \mathcal{D}_{x_P^0}$  for  $0 \leq \tau < t^*(x_P^0; x_E^0)$  and  $\sigma_E \in \mathcal{A}$ . To consider the case of a non-stationary P, we introduce the following notion.

DEFINITION 3.4 ( $\tau$ -admissible pursuit path). *For a  $\tau > 0$ , we say that a path  $x_P(\cdot)$  of P is a  $\tau$ -admissible pursuit path if*

$$t < t^*(x_P(t); x_E^0) \text{ for all } t \in [0, \tau). \tag{3.3}$$

Set  $T := v_s(x_E^0, x_P^0)$ , where  $v_s$  is defined in (2.8). Then, an optimal path  $x_P(\cdot)$  for P is a  $T$ -admissible pursuit path. If not,  $s \geq t^*(x_P(s); x_E^0)$  at some  $s < T$ ; this implies that there exists a control  $\sigma_E \in \mathcal{A}$  such that  $x_E(s) \in \mathcal{D}_{x_P(s)}$ , thereby prematurely terminating the game no later than  $s$ . Based on this observation, P’s optimal path is a  $\tau$ -admissible pursuit path with the largest  $\tau$ ; this corresponds to *reaching a final*

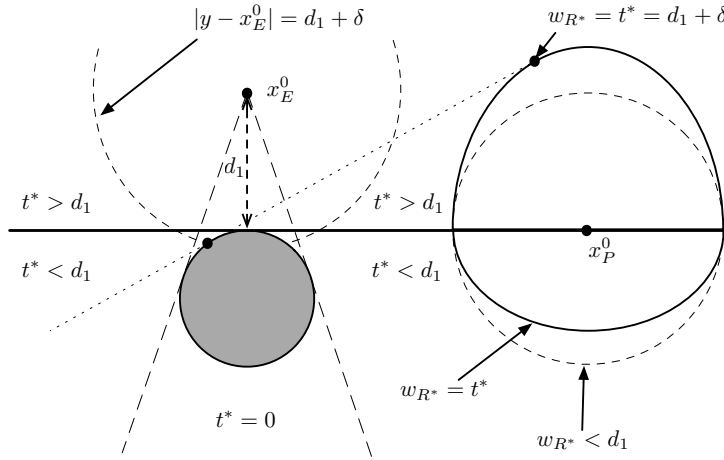


FIG. 3.1. An illustration of the relations between  $t^*(y; x_E^0)$  and  $w_{R^*}(y; x_P^0)$ . Both players are assumed to have unit speeds. The shaded circle represents the obstacle,  $d_1$  is the distance between  $x_E^0$  and the obstacle, and  $\delta$  is some small positive constant. The precise shape of the curve that corresponds to  $w_{R^*} = t^*$  is not drawn accurately in this illustration. In this scenario, the game will last until  $t > d_1$  because  $E$  will not be able to reach the shadow boundary before  $P$  pushes the shadow boundary further away.

point  $x_P^f$  which maximizes  $t^*(\cdot; x_E^0)$ . As we shall show later (Lemma 3.6), the set  $R^*$  coincides with the set of points reachable by  $\tau$ -admissible pursuit paths. Therefore, the static value can be related to  $R^*$  and  $t^*(\cdot; x_E^0)$  as follows.

**THEOREM 3.5.**  $v_s(x_E^0, x_P^0) = \sup_{x \in R^*} t^*(x; x_E^0)$ , provided both sides are finite.

We note that it is possible for  $t^*(\cdot; x_E^0)$  to be infinite, for example, if  $\Omega_{\text{free}}$  is star shaped (see Proposition 2.6). While the equality in Theorem 3.5 may still be valid in such cases, we shall not pursue them further. We give the proof of Theorem 3.5 at the end of this subsection. First, as claimed earlier, we show that all points in  $R^*$  are precisely the points “reachable” by  $\tau$ -admissible pursuit paths that start from  $x_P^0$ .

**LEMMA 3.6.**

1. For any arbitrary point  $y \in R^*$ , there exists a  $\tau$ -admissible pursuit path  $x_P(\cdot)$  such that  $x_P(t) = y$  for some  $t \in [0, \tau)$ , if there exists a time-optimal path in  $R^*$  from  $x_P^0$  to  $y$ .
2. For any  $y \in \Omega_{\text{free}} \setminus R^*$ , there does not exist a  $\tau$ -admissible pursuit path  $x_P(\cdot)$  such that  $x_P(t) = y$  for some  $t \in [0, \tau)$ .

*Proof.* 1. We construct a  $\tau$ -admissible pursuit path from a time optimal path for the minimum arrival time problem corresponding to  $(\Delta_{R^*})$ . Choose a point  $y \in R^*$ , and define  $T, \tau > 0$  as

$$T := w_{R^*}(y; x_P^0) < t^*(y; x_E^0) =: \tau. \tag{3.4}$$

Let  $x_P(\cdot)$  be the time optimal path from  $x_P^0 = x_P(0)$  to  $y = x_P(T)$  such that  $\{x_P(t) |$

$t \in [0, T] \} \subseteq R^*$ . Then

$$t = w_{R^*}(x_P(t); x_P^0) < t^*(x_P(t); x_E^0), \quad \text{for } t \in [0, T]. \tag{3.5}$$

The first equality follows from a property of time-optimal paths; cf. Corollary 2.3 in [39]. For  $t > T$ , set  $x_P(t) = y$ , i.e. P is stationary at  $y$ . Then, from Definition (3.1),

$$t < t^*(y; x_E^0) = t^*(x_P(t); x_E^0), \quad \text{for } t \in [T, \tau).$$

Thus,  $x_P(\cdot)$  is a  $\tau$ -admissible pursuit path such that  $x_P(T) = y$  for  $T \in [0, \tau)$ .

2. If such a  $\tau$ -admissible pursuit path existed, then  $w_{R^*}(y; x_P^0) < t^*(y; x_E^0)$ ; this contradicts the maximality of  $R^*$ .  $\square$

**REMARK 3.7.** Suppose part 1 of Lemma 3.6 did not assume the existence of a time-optimal path from  $x_P^0$  to  $y$  in  $R^*$ . In the spirit of Remark 2.2, we can modify Definition 3.3 and the proof in Lemma 3.6 to characterize a sequence of paths in  $R^*$  satisfying (3.5) in the limit. Namely, for fixed  $\delta > 0$  define  $R_\delta^* \subset \Omega_{\text{free}}$  to be the maximal set containing  $x_P^0$  such that

$$w_{R_\delta^*}^\delta(x; x_P^0) < t^*(x; x_E^0) - \delta, \quad \text{for all } x \in R_\delta^*. \tag{3.6}$$

It can be verified that  $\bigcup_{\delta > 0} R_\delta^* = R^*$ . Then, consider  $x_P^{\epsilon, \delta}(\cdot)$ , an  $\epsilon$ -time suboptimal path in  $R_\delta^*$ . It can be shown that

$$t < t^*(x_P^{\epsilon, \delta}(t); x_E^0) - \delta/2, \quad \text{for } t \in [0, T].$$

Thus, the limiting path as  $\epsilon \rightarrow 0$  followed by  $\delta \rightarrow 0$  satisfies the conditions in (3.5). The rest of the proof follows with the same argument as in the earlier proof.

We close this subsection with the proof of Theorem 3.5.

*Proof.* [Proof of Theorem 3.5.] By Lemma 3.6 and Remark 3.7, all  $[0, \tau)$  time portions of  $\tau$ -admissible pursuit paths lie inside  $R^*$ . Hence, P’s static value is bounded above by  $\sup_{x \in R^*} t^*(x; x_E^0)$ , i.e.  $v_s(x_E^0, x_P^0) \leq \sup_{x \in R^*} t^*(x; x_E^0)$ .

We claim that  $v_s(x_E^0, x_P^0) \geq \sup_{x \in R^*} t^*(x; x_E^0)$ . Consider a sequence  $\{y_i\}_{i=0}^\infty$  in  $R^*$  converging to  $y \in \overline{R^*}$  so that  $\tau_i := t^*(y_i; x_E^0) \rightarrow \sup_{x \in R^*} t^*(x; x_E^0)$  as  $i \rightarrow \infty$ . By Lemma 3.6, there is a  $\tau_i$ -admissible pursuit path for each  $i = 0, 1, 2, \dots$ . This implies that  $v_s(x_E^0, x_P^0) \geq \tau_i$  for all  $i$ . Taking  $i \rightarrow \infty$ , we have the desired inequality.  $\square$

**3.2. Optimal paths.** In this section, we discuss a characterization of optimal paths of P and E for the static game.

From the proof of Theorem 3.5, each of P’s optimal paths will reach an optimal final point  $x_P^f \in \overline{R^*}$  such that there exists a sequence  $\{y_i\}_{i=0}^\infty$  in  $R^*$  converging to  $x_P^f$ , and  $t^*(y_i; x_E^0) \rightarrow \sup_{x \in R^*} t^*(x; x_E^0)$  as  $i \rightarrow \infty$ . We denote all possible optimal final points as “ $\text{argsup}_{x \in R^*} t^*(x; x_E^0)$ ”.

As for E’s optimal paths, we first recall that the optimal control value function  $u_E$  is defined in (2.2). Because E has a strategy to become occluded from P at time  $v_s(x_E^0, x_P^0)$  (see Theorem 3.5), the final state  $x_E^f$  would be any point that is *reachable from  $x_E^0$  within time  $v_s(x_E^0, x_P^0)$*  and is *occluded from the point  $x_P^f$* , i.e.

$$x_E^f \in \mathcal{D}_{x_P^f} \cap \{x \mid u_E(x) \leq v_s(x_E^0, x_P^0)\}.$$

The set on the right hand side is non-empty:  $t^*(x_P^f; x_E^0) = v_s(x_E^0, x_P^0)$  implies that there exists a  $\sigma_E \in \mathcal{A}$  for E to occlude from the vantage point  $x_P^f$  at time  $v_s(x_E^0, x_P^0)$ . The optimal path for E is therefore its time-optimal path from  $x_E^0$  to  $x_E^f$  in  $\Omega_{\text{free}}$ .

**3.3. Construction of  $R^*$ .** Recall that  $R^*$  is the maximal set such that the function  $w_{R^*}$  satisfies the inequality (3.2). This characterization, however, is not constructive. To this end, consider a sequence of sets  $\{R_i\}_{i=0}^\infty$  defined as follows:

$$\begin{aligned} R_0 &:= \{x \in \Omega_{\text{free}} \mid u_P(x) < t^*(x; x_E^0)\}, \\ R_{i+1} &:= \{x \in \Omega_{\text{free}} \mid w_{R_i}(x; x_P^0) < t^*(x; x_E^0)\}, \end{aligned} \tag{3.7}$$

where  $u_P(x)$  is the minimum time for P to arrive at  $x$  from the starting location  $x_P^0$ , defined as per (2.3).

LEMMA 3.8.  $R_0 \supseteq R^*$ .

*Proof.* We prove by a contrapositive argument. Suppose we have  $u_P(x; x_P^0) \geq t^*(x; x_E^0)$ , i.e.  $x \in \Omega_{\text{free}} \setminus R_0$ . Then, no path for P can reach  $x$  before  $t^*(x; x_E^0)$ , the time necessary for E to become occluded from  $x$ . This implies that no point on the  $[0, \tau)$  portion of any  $\tau$ -admissible pursuit path would coincide with  $x$ . Thus, by Lemma 3.6,  $x \in \Omega_{\text{free}} \setminus R^*$ .  $\square$

The following theorem proves that  $R_i$  converges monotonically to a set that is contained in  $\overline{R^*}$  and that contains  $R^*$ :

THEOREM 3.9. *The sequence  $\{R_i\}_{i=0}^\infty$  satisfies*

1.  $R_{i+1} \subseteq R_i$  for all  $i = 0, 1, 2, \dots$ , and
2.  $R^* \subseteq \bigcap_{i=1}^\infty R_i \subseteq \overline{R^*}$ .

REMARK 3.10. For numerical purposes, when the set  $R_\infty := \bigcap_{i=1}^\infty R_i$  is approximated by discrete points on a finite grid,  $R^*$  and  $\overline{R^*}$  are indistinguishable. Therefore, when applying Theorem 3.5 in the forthcoming algorithm (see Algorithm 1), we are justified in treating  $R_\infty$  to be  $R^*$ .

*Proof.* [Proof of Theorem 3.9]

1. By Lemma 3.8, the initial set  $R_0$  is a superset of  $R^*$ . Then we note that

$$\begin{aligned} R_{i+1} = \{x \in \Omega_{\text{free}} \mid w_{R_i}(x; x_P^0) < t^*(x; x_E^0)\} &\subseteq \{x \in \Omega_{\text{free}} \mid w_{R_i}(x; x_P^0) < \infty\} \\ &\subseteq R_i, i = 0, 1, 2, \dots \end{aligned} \tag{3.8}$$

The last inclusion follows from the fact that if  $x \notin R_i$ , then  $w_{R_i}(x; x_P^0)$  is infinite. Define  $R_\infty := \bigcap_{i=0}^\infty R_i$ .

2. To see that  $R_\infty \supseteq R^*$ , we show by induction that  $R_i \supseteq R^*$  for all  $i = 0, 1, 2, \dots$ . The base case  $i = 0$  was proved in Lemma 3.8. Assume  $R_i \supseteq R^*$ . Then,  $w_{R_i}(x; x_P^0) \leq w_{R^*}(x; x_P^0)$  for all  $x \in R^*$ . From Definition 3.3,  $w_{R^*}(y; x_P^0) < t^*(y; x_E^0)$  for all  $y \in R^* \subseteq R_i$ . The last two facts imply  $w_{R_i}(y; x_P^0) < t^*(y; x_E^0)$ , so  $y \in R_{i+1}$ . Therefore,  $R_{i+1} \supseteq R^*$ .

Next, we show  $R_\infty \subseteq \overline{R^*}$ . Fix a  $\delta > 0$  small and define the sequence of sets  $\{R_i^\delta\}_{i=0}^\infty$  as follows:

$$\begin{aligned} R_0^\delta &:= \{x \in \Omega_{\text{free}} \mid u_P(x; x_P^0) < t^*(x; x_E^0) - \delta\}, \\ R_{i+1}^\delta &:= \{x \in \Omega_{\text{free}} \mid w_{R_i^\delta}(x; x_P^0) < t^*(x; x_E^0) - \delta\}, \quad i = 0, 1, 2, \dots \end{aligned} \tag{3.9}$$

We define  $R_\infty^\delta := \bigcap_{i=0}^\infty R_i^\delta$ . The inclusions  $R_{i+1}^\delta \subseteq R_i^\delta$  can be shown via an argument similar to that in (3.8). We claim that

$$R_\infty^\delta = \{x \in \Omega_{\text{free}} \mid w_{R_\infty^\delta}(x; x_P^0) < t^*(x; x_E^0)\}. \tag{3.10}$$

Thus, by the maximality of  $R^*$ , we have  $R_\infty^\delta \subseteq R^*$ . Because  $R_\infty \subseteq \overline{\bigcup_{\delta>0} R_\infty^\delta}$  and  $\bigcup_{\delta>0} R_\infty^\delta \subseteq R^*$ , we conclude that

$$R_\infty \subseteq \overline{\bigcup_{\delta>0} R_\infty^\delta} \subseteq \overline{R^*},$$

as desired. To show (3.10), first note that

$$\begin{aligned} R_\infty^\delta &= \bigcap_{i=0}^\infty \{x \in \Omega_{\text{free}} \mid w_{R_i^\delta}(x; x_P^0) < t^*(x; x_E^0) - \delta\} \\ &\subseteq \{x \in \Omega_{\text{free}} \mid w_{R_\infty^\delta}(x; x_P^0) \leq t^*(x; x_E^0) - \delta\} \\ &\subseteq \{x \in \Omega_{\text{free}} \mid w_{R_\infty^\delta}(x; x_P^0) < t^*(x; x_E^0)\}. \end{aligned}$$

Next, to show  $R_\infty^\delta \supseteq \{x \in \Omega_{\text{free}} \mid w_{R_\infty^\delta}(x; x_P^0) < t^*(x; x_E^0)\}$ , take any  $y \in \Omega_{\text{free}}$  such that  $w_{R_\infty^\delta}(y; x_P^0) < t^*(y; x_E^0)$ ; clearly  $w_{R_\infty^\delta}(y; x_P^0)$  is finite, and therefore  $y \in R_\infty^\delta$ . This proves (3.10).  $\square$

We note that empirically the convergence rate is quite fast. For all simulation examples (with varying domain complexities) in this paper, it takes fewer than 10 iterations for  $R_i$  to numerically converge to  $R^*$ .

**3.4. Algorithm and examples.** We summarize the steps described in the previous sections in Algorithm 1. Note that we have omitted details on computing optimal control value functions and time-optimal paths within the algorithm; this is in the spirit of both simplifying the presentation and emphasizing the independence of the algorithm to particular computational methods.

If state-of-the-art numerical HJB solvers are employed in Step 1 to solve  $u_E(\cdot)$ ,  $u_P(\cdot)$ , and  $t^*(\cdot; x_E^0)$ , then the bulk of the computation would be spent in the latter function; it involves the computation of  $\mathcal{D}_x$  and the minimum arrival time function to  $\mathcal{D}_x$  from  $y$  for each  $x, y \in \Omega_{\text{free}}$ . If  $\Omega_{\text{free}} \subset \mathbb{R}^2$  is discretized on a  $N \times N$  grid, this would involve at least  $O(N^4)$  operations:  $O(N^2)$  for computing the level set function representation for  $\mathcal{D}_x$  (see Appendix A) and at least  $O(N^2)$  for solving the minimum arrival time to  $\mathcal{D}_x$ ; cf. [35, 17, 26, 28]. While the worst-case computational complexity is comparable to solving the feedback strategy game via a Hamilton-Jacobi-Isaacs equation, we mention a few advantages of solving the static game using Algorithm 1:

1. *The most costly part of the algorithm is easily parallelizable:*  $t^*(x; x_E^0)$  can be evaluated independently for each discrete node  $x$  in  $\Omega_{\text{free}}$ .
2. *Small memory footprint:* the memory complexity is only  $O(N^2)$  because for each node  $x$ , the level-set representations of  $\mathcal{D}_x$  and the corresponding minimum arrival time function need not be stored once  $t^*(x; x_E^0)$  is evaluated.
3. *Exact solutions:* for domains with simple geometry (e.g. circles, polygonal obstacle boundaries) and constant speeds, it may be possible to derive an analytical formula for  $t^*(x; x_E^0)$  for  $x \in \Omega_{\text{free}}$ ; see examples 1 and 2 below.

Next, we present three examples demonstrating our algorithm. In all test cases,  $\Omega = [-1, 1]^2$ . Computations were performed on a desktop running on a 3.33 GHz dual-core CPU with 4 GB of RAM memory. For examples 1 and 2, computations took less than a second because an exact formula for  $t^*(\cdot; x_E^0)$  was employed. For Example 3, the complete computation took approximately 5 minutes.

---

**Algorithm 1:** Algorithm for the static visibility-based surveillance-evasion game.

---

**Input :** Domain information  $\Omega = \Omega_{\text{free}} \cup \Omega_{\text{obs}}$ .  
Initial states  $x_E^0, x_P^0$  and dynamics  $f_E, f_P$ .

**Output:** Optimal occlusion time  $T = v_s(x_E^0, x_P^0)$ .  
Optimal paths  $x_E^*, x_P^* : [0, T] \rightarrow \Omega_{\text{free}}$ .

**Step 1**

Compute functions  $u_E(x), u_P(x), t^*(x; x_E^0)$  on  $x \in \Omega_{\text{free}}$ ;

**Step 2 (construct  $R^*$ )**

$R_0 := \{x \mid u_P(x) \leq t^*(x; x_E^0)\}$

**foreach**  $i=0, 1, 2, \dots$  *until*  $R_i$  *stop changing* **do**

    Compute  $w_{R_i}$  on  $R_i$ ;

$R_{i+1} := \{x \mid w_{R_i}(x) < t^*(x; x_E^0)\}$ ;

$R^* := R_{i+1}$ ;

**Step 3 (extract static value and optimal paths)**

Return  $T := \sup_{x \in R^*} t^*(x; x_E^0)$ ;

Find  $x_P^f \in \text{argsup}_{x \in R^*} t^*(x; x_E^0)$ ;

Return  $x_P^*(\cdot)$ , the time-optimal path from  $x_P^0$  to  $x_P^f$  in  $R^*$ ;

Choose  $x_E^f \in \mathcal{D}_{x_P^f} \cap \{x \mid u_E(x) \leq T\}$ ;

Return  $x_E^*(\cdot)$ , the time-optimal path from  $x_E^0$  to  $x_E^f$  in  $\Omega_{\text{free}}$ ;

---

**3.4.1. Example 1: Circular obstacle.** The first example concerns a single circular obstacle, with E and P placed as shown in figure 3.2 (a).

Due to the simple geometry, it is possible to analytically derive a formula for  $t^*(x; x_E^0)$  for  $x \in \Omega_{\text{free}}$ . All other functions,  $u_E, u_P$ , and  $w_{R_i}$  ( $i=0, 1, \dots$ ) were approximated on a uniform Cartesian grid of size  $400^2$ . The function  $t^*(\cdot; x_E^0)$  is shown in figure 3.3 (a). In figure 3.3 (b), the computed  $R^*$  using the algorithm 3.7 is plotted, as well as the optimal paths of E and P.

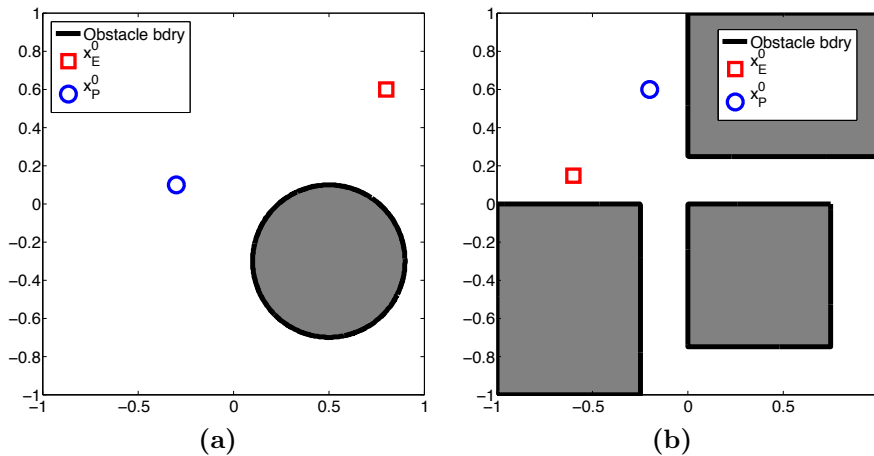


FIG. 3.2. Obstacle (shaded) and initial states for (a) Example 1 and (b) Example 2.

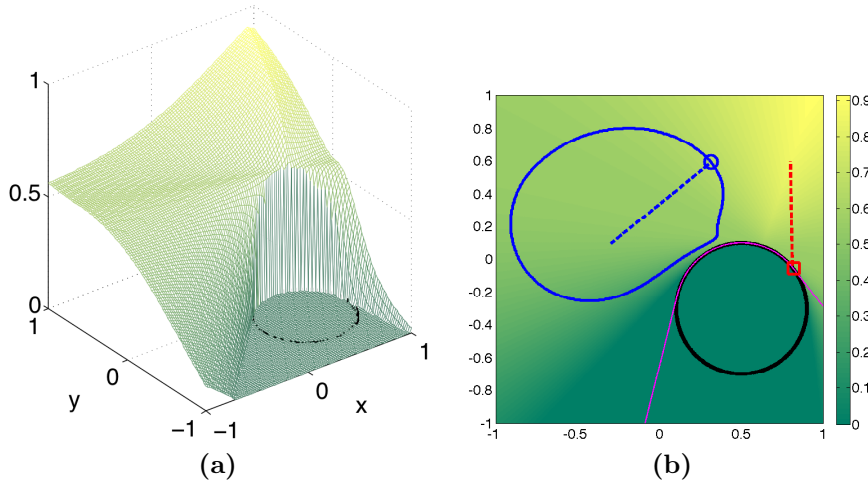


FIG. 3.3. *Example 1. (a) Surface plot of  $t^*(\cdot; x_E^0)$ , where  $x_E^0 = (0.8, 0.6)$ . (b) A plot illustrating  $R^*$  (blue curve),  $t^*(\cdot; x_E^0)$  (background),  $x_E^f$  (red square),  $x_P^f$  (blue circle), and the optimal paths of  $P$  (blue dotted curve) and  $E$  (red dotted curve), where the isotropic speeds are  $f_E = 1, f_P = 1.2$ . The purple curve represents the visibility region from the vantage point  $x_P^f$ .*

**3.4.2. Example 2: Two corridors.** Next we consider a case where  $x_E^0, x_P^0$ , and  $\Omega_{\text{free}}$  are shown in figure 3.2 (b). Again, due to the simplicity of the geometry of the domain, we derived an analytical formula for  $t^*(x, x_E^0)$  for  $x \in \Omega_{\text{free}}$ .

For the sake of analyzing the solution, we set  $S$  to be the square  $(x, y) \in [-0.25, 0] \times [0, 0.25]$  (the north-east, south-west, and south-east vertices of  $S$  coincide with the vertices near the origin of the three obstacles). Also, we call  $C_1$  the set of points  $(x, y) \in [0, 1] \times [0, 0.25]$  in the “horizontal corridor” between the upper right and bottom right rectangular obstacles. Likewise, we call  $C_2$  the set of points  $(x, y) \in [-0.25, 0] \times [-1, 0]$  in the “vertical corridor” between the lower left and bottom right rectangular obstacles. Note that the set of states where  $P$  has the greatest visibility of domain is  $S$ ; this is reflected by the peak in  $t^*(x; x_E^0)$  for  $x \in S$ , shown in figure 3.4 (a) and (b). This prompts a conservative  $P$  to move towards  $S$  via a time-optimal path, and remain stationary until the game terminates. Indeed, if  $P$  “commits” to entering  $C_1$  then  $E$  would counter by moving into  $C_2$ , thereby ending the game prematurely; a similar argument follows if  $P$  commits to entering  $C_2$ . Consequently,  $E$ ’s final state  $x_E^f$  to become occluded from a  $P$  in  $S$  would either be the south-west or north-east vertex of the bottom right obstacle, whichever is closer to  $x_E^0$ ; in the present example,  $x_E^f$  is the former.

**3.4.3. Example 3: inhomogeneous speeds.** In figure 3.5, we illustrate how inhomogeneous speed functions  $f_E$  and  $f_P$  can be handled using the proposed algorithm. The following isotropic velocity functions were used: for all  $\hat{\sigma}_E \in A$  and  $\hat{\sigma}_P \in A$ ,

$$f_E(x, \hat{\sigma}_E) = f_P(x, \hat{\sigma}_P) = f(x) = 1 - 0.8 \sin(4\pi x_1) \sin(4\pi x_2), \quad \text{where } x = (x_1, x_2).$$

Because an analytical formula for  $t^*(\cdot; x_E^0)$  is too complicated to derive, it was constructed numerically via Definition 3.1. All computations were performed on a  $200^2$

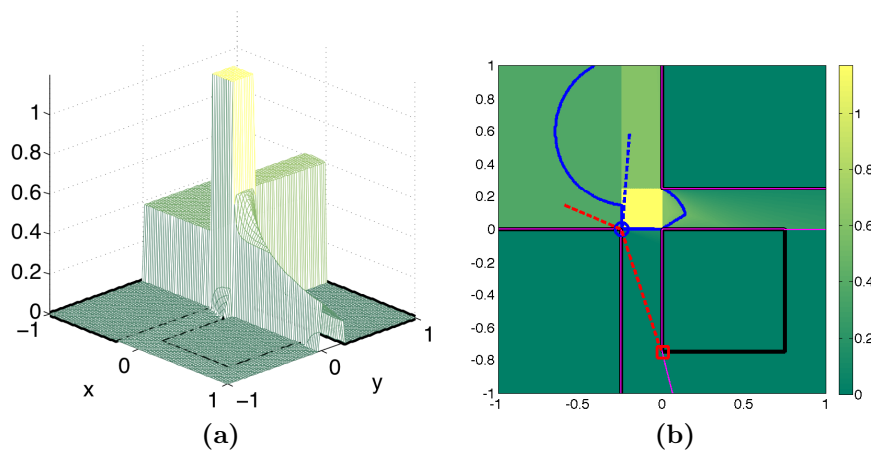


FIG. 3.4. Example 3. (a) Surface plot of  $t^*(\cdot; x_E^0)$ , where  $x_E^0 = (-0.6, 0.15)$ . (b) A plot illustrating  $R^*$  (blue curve),  $t^*(\cdot; x_E^0)$  (background),  $x_E^f$  (red square),  $x_P^f$  (blue circle), and optimal paths of  $P$  (blue dotted curve) and  $E$  (red dotted curve), where the isotropic speeds are  $f_E = 1, f_P = 1.2$ . The purple curves represent the visibility region from the vantage point  $x_P^f$ .

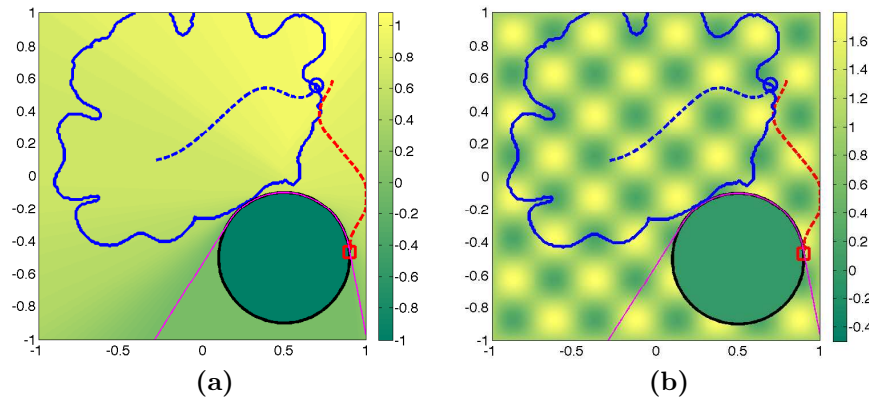


FIG. 3.5. Example 2. (a) A plot illustrating  $R^*$  (blue curve),  $t^*(\cdot; x_E^0)$  (background),  $x_E^f$  (red square),  $x_P^f$  (blue circle), and optimal paths of  $P$  (blue dotted curve) and  $E$  (red dotted curve), with an intensity plot of  $t^*(\cdot; x_E^0)$  in the background, with the speed function  $f(x)$ . (b) Same plot as (a), except with the speed function  $f(x)$  in the background.

domain. Note that, by observing figure 3.5 (b), both players travel where their speeds are the greatest (yellow regions).

#### 4. Applications and extensions

In this section, we present several immediate applications and extensions of the static game algorithm. In Section 4.1, we show that for a generalized version of the visibility pursuit-evasion game with multiple evaders, the computation scales linearly with the number of players. In Section 4.2, we present a feedback control strategy by invoking the static game algorithm iteratively over short time intervals.



**4.1. Multiple evaders.** A natural extension of the present algorithm is to the visibility based single-pursuer, multiple-evader (static) game. Suppose there are  $k$  evaders,  $E_1, E_2, \dots, E_k$ , where  $E_i$ 's motion satisfies the initial value problem

$$\dot{x}_{E_i}(t) = f_{E_i}(x_{E_i}(t), \sigma_{E_i}(t))\sigma_{E_i}(t), \quad x_{E_i}(0) = x_{E_i}^0 \in \Omega_{\text{free}} \quad \text{for each } i = 1, 2, \dots, k.$$

For each  $i$ , the control  $\sigma_{E_i} \in \mathcal{A}$  is chosen as with the two player game. We assume, for each  $i$ , that the velocity  $f_{E_i}$  satisfies the conditions (A1)-(A3).

Given  $x_P^0, \{x_{E_i}^0\}_{i=1}^k$  and  $\sigma_P \in \mathcal{A}, \{\sigma_{E_i} \in \mathcal{A}\}_{i=1}^k$ , consider the payoff function

$$\mathcal{J}_k(x_{E_1}^0, \dots, x_{E_k}^0, x_P^0, \sigma_{E_1}, \dots, \sigma_{E_k}, \sigma_P) = \min_{i=1, \dots, k} \mathcal{J}(x_{E_i}^0, x_P^0, \sigma_{E_i}, \sigma_P), \quad (4.1)$$

where  $\mathcal{J}$  is the payoff function for the two-player game (2.7). Here, the game terminates the first instance that at least one evader becomes occluded from the pursuer. The key property is that each  $E_i$  plays independently against P, thus solving this game amounts to  $k$  instances (for each  $E_i$ ) of the two-player game. As with the two-player game, all players choose their complete controls at the start of the game.

Let  $t^*(\cdot; x_{E_i}^0)$  be the function defined as per (3.1) with initial state  $x_{E_i}^0$ . Then, analogous to (3.3), the  $\tau$ -admissible pursuit path  $x_P(\cdot)$  for P among the  $E_i$ 's must satisfy  $t < t^*(x_P(t); x_{E_i}^0)$  for all  $t \in [0, \tau)$  and  $i = 1, \dots, k$ , or equivalently,

$$t < \min_{i=1, \dots, k} t^*(x_P(t); x_{E_i}^0) \quad \text{for all } t \in [0, \tau).$$

The rest of the algorithm follows the same procedure as with the two-player game, where  $t^*(x; x_E^0)$  is replaced by  $\min_{i=1, \dots, k} t^*(x; x_{E_i}^0)$ ; the definitions of  $R^*, w_{R^*}$ , etc. can be defined accordingly. For example, the static value function for  $k$  evaders becomes

$$v_k(x_{E_1}, \dots, x_{E_k}, x_P^0) = \sup_{x \in R^*} \min_{i=1, \dots, k} t^*(x; x_{E_i}^0),$$

and the final state of P is  $x_P^f = \text{argsup}_{x \in R^*} \min_{i=1, \dots, k} t^*(x; x_{E_i}^0)$ . Furthermore, there exists  $i \in \{1, \dots, k\}$  and (a final state)  $x_{E_i}^f \in \Omega_{\text{free}}$  such that  $x_{E_i}^f$  is reachable by  $E_i$  (via a time-optimal path) in time  $v_k(x_{E_1}, \dots, x_{E_k}, x_P^0)$  and occluded from  $x_P^f$ .

REMARK 4.1. Because each  $t^*(\cdot; x_{E_i}^0)$  can be computed independently on the underlying domain, the computational complexity for multiple evaders scales linearly with the number of players.

**4.2. Applications to a feedback control game and target tracking.** Let us return to the two-player game setting, E versus P. In realistic settings, controls of one or more players are chosen using feedback controls, rather than open loop controls. To model a feedback control strategy, we adopt an iterative open loop approach in the spirit of Model Predictive Control (MPC) [29]: one advances the players over short time intervals according to the optimal controls computed by Algorithm 1, starting at the current states.

Suppose the player states are sampled at discrete times  $\Delta = (0 = t_0, t_1, t_2, \dots, t_n)$ , where  $t_i < t_{i+1}$  for all  $i = 0, 1, 2, \dots, n$ . The aforementioned feedback control strategy can be stated as follows. Suppose E and P initially start at states  $x_E^0$  and  $x_P^0$ , respectively. We write the player states in each time interval as

$$\begin{cases} x_E(t) = x_E^{(i)}(t - t_i), \\ x_P(t) = x_P^{(i)}(t - t_i), \end{cases} \quad t \in [t_i, t_{i+1}).$$

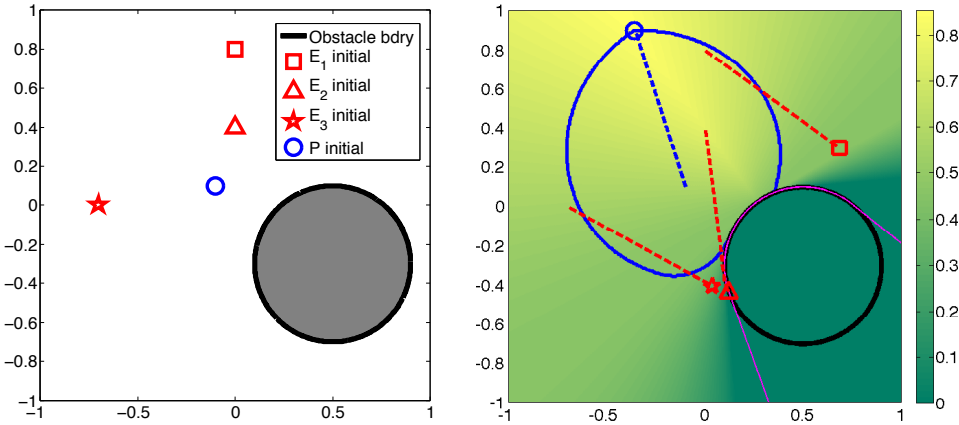


FIG. 4.1. The solution to a visibility single-pursuer, three-evader static game.

In the time interval  $[t_i, t_{i+1}]$ , the states satisfy the dynamics

$$\begin{cases} \dot{x}_E^{(i)}(t) = f_E(x_E^{(i)}(t), \Phi(t))\Phi(t), \\ \dot{x}_P^{(i)}(t) = f_P(x_P^{(i)}(t), \Psi(t))\Psi(t), \end{cases} \quad t \in [0, t_{i+1} - t_i], \quad (4.2)$$

with initial conditions defined recursively as

$$(x_E^{(i)}(0), x_P^{(i)}(0)) = \begin{cases} (x_E^{(i-1)}(t_i - t_{i-1}), x_P^{(i-1)}(t_i - t_{i-1})), & \text{if } i \geq 1, \\ (x_E^0, x_P^0), & \text{if } i = 0. \end{cases} \quad (4.3)$$

For fixed states  $x_E, x_P \in \Omega_{\text{free}}$ , the feedback controls  $\Phi(t) = \Phi(x_E, x_P; \cdot) \in \mathcal{A}$  and  $\Psi(t) = \Psi(x_E, x_P; \cdot) \in \mathcal{A}$  are the static game controls computed from the joint state configuration at the discrete time sets:

$$\begin{aligned} \Psi(x_E, x_P; \cdot) &\in \arg \sup_{\sigma_P \in \mathcal{A}} \inf_{\sigma_E \in \mathcal{A}} \mathcal{J}(x_E, x_P, \sigma_E, \sigma_P), \\ \Phi(x_E, x_P; \cdot) &\in \arg \inf_{\sigma_E \in \mathcal{A}} \mathcal{J}(x_E, x_P, \sigma_E, \Psi(x_E, x_P; \cdot)). \end{aligned} \quad (4.4)$$

In figure 4.2, we show a comparison of the static and feedback control games on a simple example. The players both have unit speeds in a square domain  $[-1, 1]^2$  with a single circular obstacle. Here, we sampled and recomputed the controls at equal intervals  $\Delta = (0, \Delta t, 2\Delta t, 3\Delta t, \dots)$ , where  $\Delta t = 0.025$ ; the iterative procedure was performed until E became occluded from P. Notice how the feedback strategy game prolonged the game duration compared to the static game. This makes intuitive sense: because the static value (2.8) corresponds to the most conservative strategy for P, the iterative procedure relaxes this conservatism and thus yields a more favorable outcome for P.

REMARK 4.2. In general, the static game in each interval does not have a unique optimal control. Take for instance Example 2 (the two-corridor example): all time-optimal controls for P that reach a (final) point  $x_P^f \in S$  are optimal controls for the static game. While the static values are all equal for all such optimal controls, some choices may be more advantageous because they will directly affect the feedback

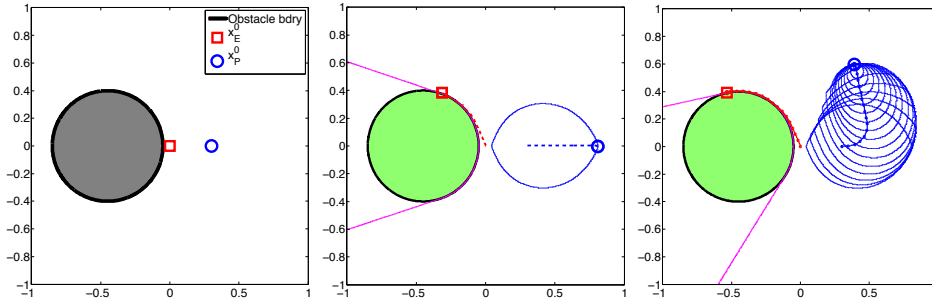


FIG. 4.2. A comparison between the static and feedback control games. From left to right: the initial states and obstacles, the optimal static game paths, and the optimal feedback control game paths. In the feedback control game, we have plotted the sequence of positions of the players as well as the  $R^*$  at each iteration.

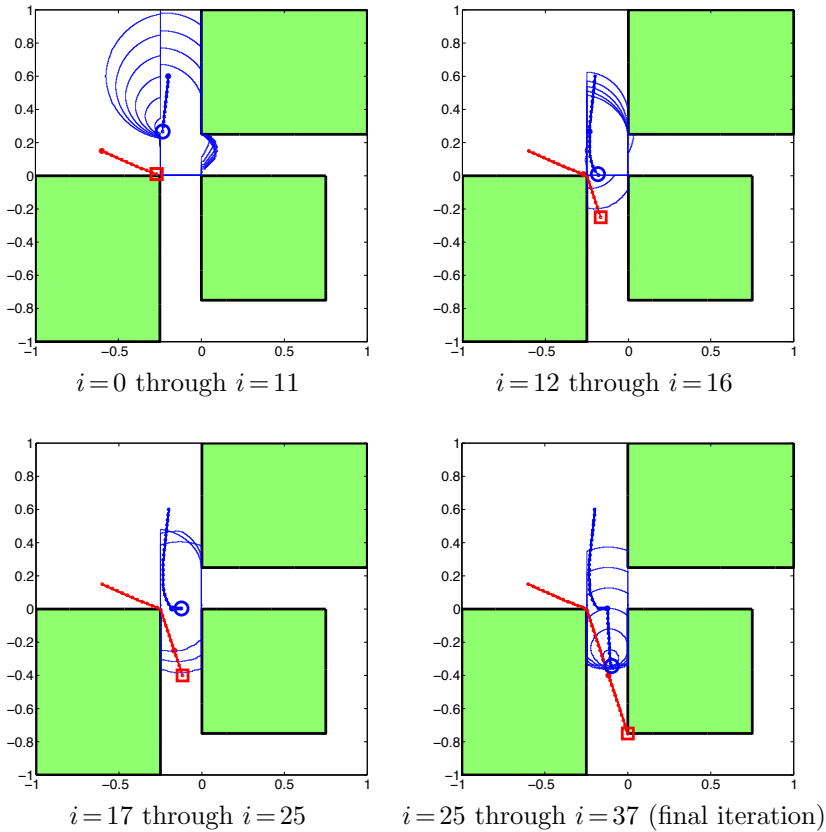


FIG. 4.3. Snapshots of the iterative static game algorithm.

control in subsequent iterations. As a sensible strategy to uniquely choose among the optimal state game controls, at each step  $i$ , we implement P to choose the control corresponding to the final point  $x_P^f$  such that  $\|x_P^f - x_E(t_i)\|_2$  is minimized, i.e. choose the control to move to the point closest to the current state of E.

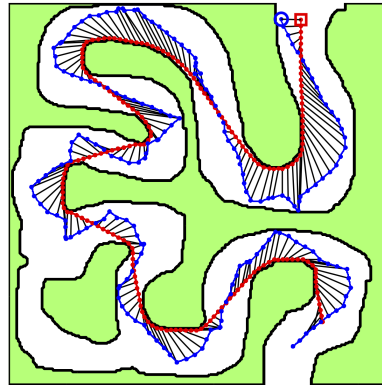


FIG. 4.4. A target tracking example. Both players start at the bottom right entrance to the “cave”. The target’s path (in red) is predefined, heading towards to the cave exit at the top right corner. the tracker’s path (in blue) is computed using a sequence of controls computed by Algorithm 1 from the current states at fixed time intervals. The black line segment connects the target and tracker states at the same discrete time instances  $t_i$ .

As another illustration, in figure 4.3 we show snapshots of the optimal paths computed using the iterative static information pattern described above. The state constraints, dynamics, and initial conditions are the same as in figure 3.4 (the two-corridor example). Notice that between  $i = 16$  and  $i = 25$ , P is forced to stay within  $S$ , owing to P’s conservative nature in the static game; indeed, if P commits down the vertical corridor  $C_2$  at  $i = 16$ , then E would counter by reversing course and occlude behind the corner at  $(0,0)$ . It is only after E has passed the midway point of  $C_2$  at  $i = 25$ , that P deems his commitment down  $C_2$  to be worthwhile.

**4.2.1. Target tracking.** In a realistic setting, one of P or E may move sub-optimally. In the case where E is sub-optimal, the problem can be seen as a target-tracking problem, where E is the (moving) target and P is the observer. This can be modeled by modifying the formulation (4.2)-(4.4) so that  $\Phi = \Phi(t) \in \mathcal{A}$  is some arbitrary prescribed control for E (the oblivious target). That is, at each discrete time  $t_i$ , only P chooses its optimal static game control based on the current states of the system. An numerical result of this approach is shown in figure 4.4.

## 5. Conclusions and future work

In this article, we have presented an efficient algorithm to compute the value and optimal controls for a class of pursuit-evasion games, called the visibility-based surveillance-evasion game. The novelty of the algorithm is the decomposition of a game into optimal control problems, which can be solved on a lower dimensional domain. This decomposition is made possible by the decoupled player dynamics and the so called static information pattern. Generalizations to inhomogenous speeds and multiple evaders were presented, as well as applications to target tracking and feedback control games.

The present technique of solving the static game using optimal control was recently applied to another problem in [36]. There, the upper static value for the game known as *capture-the-flag* [16] was considered. We are currently investigating a more general class of pursuit-evasion type games that can be solved via a similar decomposition technique [41].

We are also investigating a more efficient method for computing  $t^*(\cdot; x_E^0)$ , which is the bottleneck to our algorithm in terms of computation time. Preliminary tests have shown that for cases with relatively small  $R^*$ , the computation time can be significantly reduced. For instance, the test case shown in figure 4.4 can be produced at, on average, approximately 3 seconds per iteration (i.e. for each  $t_i$ ) using the more efficient approach. Another useful result, in the spirit of Example 2, would be an algorithm that computes the exact formula for  $t^*(\cdot; x_E^0)$  in a *polygonal* domain for an E with a constant, isotropic speed.

**6. Acknowledgement.** Takei was supported by ONR grants N00014-07-1-0810, N0014-03-1-0071, N00014-07-1-0810, DOE grant DE-FG02-05ER25710, an ARO MURI through Rice University, and the CHASE MURI grant 556016. Tsai was supported by National Science Foundation grants DMS-0914465, DMS-0914840, and a MURI sub-contract from Univ. of S. Carolina grant No. W911NF-07-1-0185. Zhou was supported by the National Science Foundation under grant CNS-0931843.

**Appendix A. The level set visibility and shadow functions.** In this appendix, we describe a fast sweeping type method for constructing an implicit representation for  $\mathcal{D}_{x_0}$  for an arbitrary vantage point  $x_0 \in \Omega_{\text{free}}$ . This implicit representation uses a continuous level set function, which we write as  $\xi(\cdot; x_0): \Omega \rightarrow \mathbb{R}$ , holding the property

$$\xi(x; x_0) \leq 0 \Leftrightarrow x \in \mathcal{D}_{x_0}. \tag{A.1}$$

We shall call  $\xi(\cdot; x_0)$  the *shadow function* from the vantage point  $x_0 \in \Omega_{\text{free}}$ . The minimum arrival time function to  $\mathcal{D}_x = \{x \in \Omega_{\text{free}} \mid \xi(x; x_0) \leq 0\}$ , denoted by  $t^*$ , can then be conveniently computed as per the definition (3.1) by solving an eikonal equation with the boundary conditions defined by  $\xi$ .

We begin by assuming that the obstacles are represented as a level set function  $\psi: \Omega \rightarrow \mathbb{R}$ ,

$$\psi(x) \leq 0 \Leftrightarrow x \in \Omega_{\text{obs}}. \tag{A.2}$$

We shall refer to such a function as an *occluder function*. Let  $\Gamma = \{x \mid \psi(x) = 0\}$  be the boundary of the obstacles. For what follows, we assume that  $\Gamma$  is smooth, and  $\nabla\psi(x) \neq 0$  on  $\Gamma$ . Consequently,  $\nabla\psi(x)/\|\nabla\psi(x)\| = \mathbf{n}$  defines the unit outer normal of  $\Omega_{\text{obs}}$  on  $\Gamma$ .

Next, we adopt the framework of [34] by defining the *visibility function* associated with the occluder function  $\psi$  and the vantage point  $x_0 \in \Omega$ :

$$\varphi(x; x_0) = \min_{y \in \mathcal{L}(x, x_0)} \psi(y), \tag{A.3}$$

where, we take  $\mathcal{L}(x, x_0)$  as the line segment<sup>2</sup> between  $x$  and  $x_0$ . Note that, by definition,  $\varphi(x, y) = \varphi(y, x)$ . Using this function, we define the visibility of  $x$  from the vantage point  $x_0$ :

$$\varphi(x; x_0) > 0 \Leftrightarrow x \text{ is visible from } x_0. \tag{A.4}$$

A graphical illustration of the zero sub-level set of  $\varphi(\cdot, x_0)$  is shown in figure A.2 (b).

---

<sup>2</sup>In general,  $\mathcal{L}(x, x_0)$  is the curve representing the path of a ray of light connecting  $x$  and  $x_0$  influenced by refraction and/or reflection.

REMARK A.1. We point out that, while it may appear that  $f(x; x_0) = \max\{\varphi(x; x_0), -\psi(x)\}$  is a candidate definition for the shadow function, it does not satisfy (A.1). Observe that  $f(x; x_0) = 0$  for all  $x \in \Gamma$  and in general,  $\Gamma \not\subset \mathcal{D}_{x_0} \cup \partial\mathcal{D}_{x_0}$ ; consider the points  $x \in \Gamma$  that are “visible” from the vantage point, i.e.  $(x_0 - x) \cdot \nabla\psi(x) > 0$ ; see also Case V below.

We now describe a method for constructing the shadow function  $\xi(\cdot; x_0)$  by performing a few Boolean operations on sets defined by some auxiliary level set functions. Fix a vantage point  $x_0 \in \Omega_{\text{free}}$ . Define the auxiliary function  $g(\cdot; x_0): \Omega \rightarrow \mathbb{R}$  as

$$g(x; x_0) = \max\{\psi(x), (x_0 - x) \cdot \nabla\psi(x)\}.$$

Thus  $g$  describes the portions of  $\partial\Omega_{\text{obs}}$  that cannot be visible from  $x_0$ . Let  $\tilde{g}(\cdot; x_0): \Omega \rightarrow \mathbb{R}$  be the visibility function associated with the occluder function  $\tilde{\psi}(x) = g(x; x_0)$ . Then, a candidate shadow function is given by

$$\xi(x; x_0) = \max\{-\psi(x), \tilde{g}(x; x_0)\}. \tag{A.5}$$

We illustrate the auxiliary function  $g$  and the shadow function  $\xi$  in figure A.2 (c) and (d). Note that this construction of  $\xi$  involves solving the visibility function twice and a pointwise maximum operation twice. Thus, the calculation is  $O(N)$  for a mesh with  $N$  grid points.

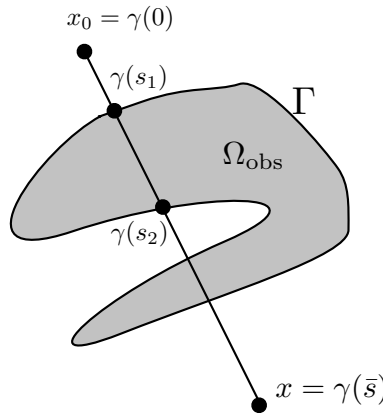


FIG. A.1. An example illustrating  $\mathcal{L}(x, x_0) = \{\gamma(s) \mid s \in [0, \bar{s}]\}$  where  $\gamma(x_0) = x_0$  is the vantage point.

Now, we show that (A.1) is satisfied by considering all possible cases for  $x \in \Omega$ . For the purpose of justifying the construction of  $\xi$  above, we define a line  $\gamma: [0, \bar{s}] \rightarrow \Omega$ , parametrized by arc length, where

$$\mathcal{L}(x, x_0) = \{\gamma(s) \mid s \in [0, \bar{s}]\}, \quad \gamma(0) = x_0, \gamma(\bar{s}) = x.$$

If  $x \in \mathcal{D}_{x_0}$ , we also define  $s_1, s_2 \in [0, \bar{s}]$  such that  $\gamma(s_1)$  is the first entry point into  $\Omega_{\text{obs}}$  and  $\gamma(s_2)$  is the first exit point from  $\Omega_{\text{obs}}$ :

$$s_1 = \arg \min_{s \in [0, \bar{s}]} \{\gamma(s) \in \Omega_{\text{obs}}\}, \quad s_2 = \arg \min_{s \in [s_1, \bar{s}]} \{\gamma(s) \in \Gamma\} \geq s_1.$$

See figure A.1 depicting  $\gamma$  on a simple example. Note that, if  $s_1 = s_2$ , then  $\mathcal{L}(x, x_0)$  is tangent to  $\Omega_{\text{obs}}$  at  $\gamma(s_1)$ .

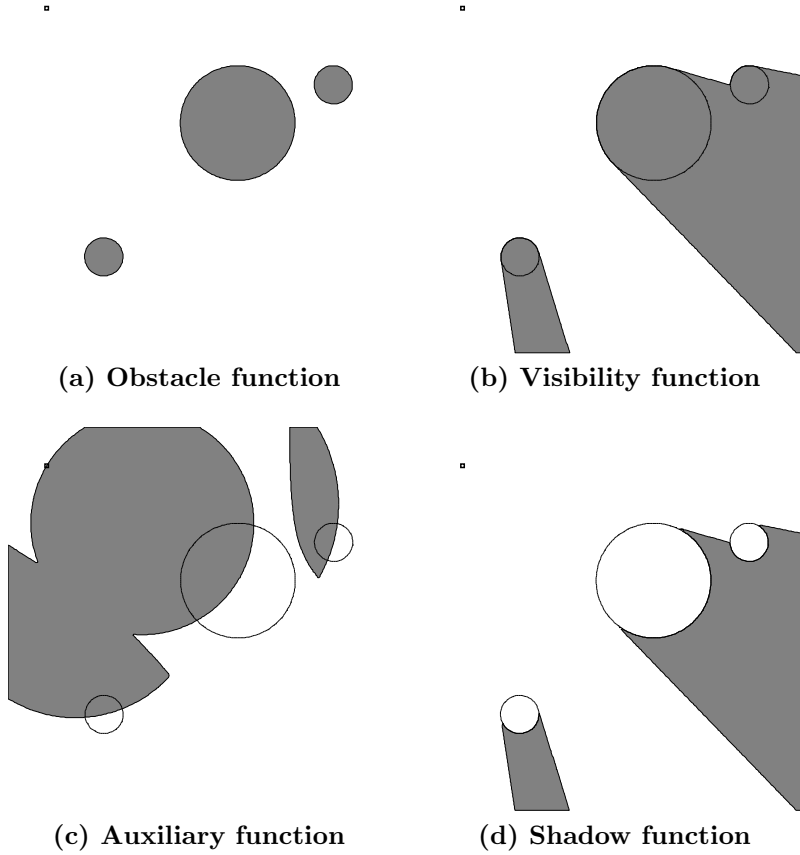


FIG. A.2. Illustration of the construction of the shadow function in  $\mathbb{R}^2$ . The shaded area is where each function is negative; the small square on the top left of each plot is the vantage point  $x_0$ . The boundary of the obstacles  $\Gamma$  are shown as black contours. (a) The obstacle function  $\psi$ , (b) visibility function  $\varphi(\cdot, x_0)$ , (c) the auxiliary function  $g(\cdot, x_0)$ , and (d) the shadow function  $\xi(\cdot, x_0)$ .

Thus, the consistency of (A.5) with (A.1) can be justified by considering the following five cases for  $x \in \Omega$ :

- Case I.**  $x \in \text{int}(\Omega_{\text{obs}})$ . Because  $\psi(x) < 0$  we have  $\xi(x; x_0) > 0$ , as desired.
- Case II.**  $x \notin \mathcal{D}_{x_0}$ . In this case, we have  $\mathcal{L}(x, x_0) \subset \Omega_{\text{free}}$ , so  $g(y; x_0) \geq \psi(y) > 0$  for all  $y \in \mathcal{L}(x, x_0)$ . Thus,  $\tilde{g}(x; x_0) > 0$ . By (A.5), we conclude that  $\xi(x; x_0) > 0$ , as desired.
- Case III.**  $x \in \mathcal{D}_{x_0}$ . Note that at  $y = \gamma(s_2) \in \Gamma$ , we have  $(x_0 - y) \cdot \nabla \psi(y) \leq 0$  and  $\psi(y) = 0$ . Therefore,  $g(y; x_0) = 0$  and

$$\tilde{g}(x; x_0) = \min_{s \in [0, \bar{s}]} g(\gamma(s); x_0) \leq g(y; x_0) = 0.$$

Furthermore, because  $-\psi(y) < 0$ , we have that  $\xi(x; x_0) \leq 0$  as desired.

- Case IV.**  $x \in \Gamma \cap \partial \mathcal{D}_{x_0}$ . Here, we necessarily have  $x = \gamma(s_2)$ . Then, by the same argument as Case III, we have  $g(x; x_0) = \tilde{g}(x; x_0) = \psi(x) = 0$ . Thus,  $\xi(x; x_0) = 0$  as desired.

**Case V.**  $x \in \Gamma \setminus \partial \mathcal{D}_{x_0}$ . In this case,  $x = \gamma(s_1)$  and  $s_1 < s_2$ . Then,  $(x_0 - x) \cdot \nabla \psi(x) > 0$ , so  $g(x; x_0) > 0$ . Furthermore, note that  $g(\gamma(s); x_0) \geq \psi(\gamma(s)) > 0$  for  $s \in [0, s_1]$ . Therefore,  $\tilde{g}(x; x_0) > 0$ , and we conclude that  $\xi(x; x_0) > 0$ .

## REFERENCES

- [1] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Systems & Control: Foundations & Applications. Birkhäuser Boston Inc., Boston, MA, 1997.
- [2] M. Bardi, M.G. Crandall, L.C. Evans, H.M. Soner, and P.E. Souganidis, *Viscosity Solutions and Applications*, in Lecture Notes in Mathematics, Springer-Verlag, Berlin, 1660, 1997.
- [3] D.P. Bertsekas, *Dynamic Programming: Deterministic and Stochastic Models*, Prentice-Hall, 1986.
- [4] M. Bardi, M. Falcone, and P. Soravia, *Numerical methods for pursuit-evasion games via viscosity solutions*, in Stochastic and Differential Games, Ann. Internat. Soc. Dynam. Games, Birkhäuser Boston, Boston, MA, 4, 105–175, 1999.
- [5] S. Bhattacharya and S. Hutchinson, *Approximation schemes for two-player pursuit evasion games with visibility constraints*, in Proceedings of Robotics: Science and Systems IV, Zurich, Switzerland, June 2008.
- [6] S. Bhattacharya and S. Hutchinson, *On the existence of Nash equilibrium for a visibility based pursuit evasion game*, in Workshop on Algorithmic Foundations of Robotics, 2008.
- [7] T. Bandyopadhyay, Y. Li, M.H. Ang Jr., and D. Hsu, *Stealth tracking of an unpredictable target among obstacles*, in Algorithmic Foundations of Robotics VI, Springer-Verlag, 43–58, 2005.
- [8] M.G. Crandall and P.L. Lions, *Viscosity solutions of Hamilton-Jacobi equations*, Trans. Amer. Math. Soc., 277(1), 1–42, 1983.
- [9] L.T. Cheng and Y.H. Tsai, *Visibility optimization using variational approaches*, Commun. Math. Sci., 3(3), 425–451, 2005.
- [10] L.C. Evans and P.E. Souganidis, *Differential games and representation formulas for solutions of Hamilton-Jacobi-Isaacs equations*, Indiana Univ. Math. J., 33(5), 773–797, 1984.
- [11] M. Falcone and R. Ferretti, *Semi-Lagrangian schemes for Hamilton-Jacobi equations, discrete representation formulae and Godunov methods*, J. Comput. Phys., 175(2), 559–575, 2002.
- [12] P. Fabiani, H.H. González-Baños, J.C. Latombe, and D. Lin, *Tracking a partially predictable target with uncertainties and visibility constraints*, J. Autonomous Robots, 38(1), 31–48, 2001.
- [13] L.J. Guibas, J.C. Latombe, S.M. LaValle, D. Lin, and R. Motwani, *Visibility-based pursuit-evasion in a polygonal environment*, in International Journal of Computational Geometry and Applications, Springer-Verlag, 17–30, 1997.
- [14] R. Gonzalez and E. Rofman, *On deterministic control problems: An approximation procedure for the optimal cost in the stationary problem*, SIAM J. Cont. Optim., 23(2), 242–266, 1985.
- [15] I. Greenfeld, *A differential game of surveillance evasion of two identical cars*, J. Optim. Theory Appl., 52(1), 1987.
- [16] H. Huang, J. Ding, W. Zhang, and C.J. Tomlin, *A differential game approach to planning in adversarial scenarios: A case study on capture-the-flag*, in Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, May, 2011.
- [17] J. Hershberger and S. Suri, *An optimal algorithm for Euclidean shortest paths in the plane*, SIAM J. Comput., 28, 2215–2256, 1999.
- [18] R. Isaacs, *Differential Games, A Mathematical Theory, with Applications to Warfare and Pursuit, Control and Optimization*, John Wiley and Sons, Inc., New York, 1965.
- [19] S. Karaman and E. Frazzoli, *Incremental sampling-based algorithms for a class of pursuit-evasion games*, in Algorithmic Foundations of Robotics IX, Springer Tracts in Advanced Robotics, Springer Berlin Heidelberg, 68, 71–87, 2011.
- [20] J. Lewin and J.V. Breakwell, *The surveillance-evasion game of degree*, J. Optim. Theory Appl., 16(3/4), 339–353, 1975.
- [21] S.M. LaValle, H.H. González-Baños, C. Becker, and J.C. Latombe, *Motion strategies for maintaining visibility of a moving target*, in Proceedings IEEE International Conference on Robotics and Automation, 731–736, 1997.
- [22] C.Y. Lee, H.H. González-Baños, and J.C. Latombe, *Real-time tracking of an unpredictable target amidst unknown obstacles*, in ICARCV, 596–601, 2002.
- [23] P.L. Lions, *Generalized Solutions of Hamilton-Jacobi Equations*, Pitman, Boston, 1982.
- [24] J. Lewin and G.J. Olsder, *Conic surveillance evasion*, J. Optim. Theory Appl., 27(1), 107–125, 1979.



- [25] R. Murieta-Cid, H. González-Baños, and B. Tovar, *A reactive motion planner to maintain visibility of unpredictable targets*, in Proceedings IEEE International Conference on Robotics and Automation, 4242–4248, 2002.
- [26] J.S.B. Mitchell, *Shortest paths among obstacles in the plane*, in Proceedings of the Ninth Annual Symposium on Computational Geometry, SCG '93, New York, NY, USA, 308–317, 1993.
- [27] I. Mitchell, *The flexible, extensible and efficient toolbox of level set methods*, J. Sci. Comput., 35(2-3), 300–329, 2008.
- [28] J.S.B. Mitchell, D.M. Mount, and C.H. Papadimitriou, *The discrete geodesic problem*, SIAM J. Comput., 16(4), 647–668, 1987.
- [29] J.B. Rawlings and D.Q. Mayne, *Model Predictive Control: Theory and Design*, Nob Hill Publishing, Madison, WI, 2009.
- [30] E. Rouy and A. Tourin, *A viscosity solutions approach to shape-from-shading*, SIAM J. Numer. Anal., 29(3), 867–884, 1992.
- [31] J.A. Sethian, *A fast marching level set method for monotonically advancing fronts*, in Proc. Nat. Acad. Sci., 1591–1595, 1995.
- [32] J.A. Sethian and A. Vladimirsky, *Ordered upwind methods for static Hamilton-Jacobi equations: Theory and algorithms*, SIAM J. Numer. Anal., 41(1), 325–363, 2003.
- [33] I. Suzuki and M. Yamashita, *Searching for a mobile intruder in a polygonal region*, SIAM J. Comput., 21(5), 863–888, 1992.
- [34] Y.H.R. Tsai, L.T. Cheng, S. Osher, P. Burchard, and G. Sapiro, *Visibility and its dynamics in a PDE based implicit framework*, J. Comput. Phys., 199(1), 260–290, 2004.
- [35] Y.H.R. Tsai, L.T. Cheng, S. Osher, and H.K. Zhao, *Fast sweeping algorithms for a class of Hamilton-Jacobi equations*, SIAM J. Numer. Anal., 41(2), 673–694, 2003.
- [36] R. Takei, H. Huang, J. Ding, and C.J. Tomlin., *Time-optimal multi-stage motion planning with guaranteed collision avoidance via an open-loop game formulation*, in Proceedings of the IEEE International Conference on Robotics and Automation, 323–329, 2012.
- [37] B. Tovar and S.M. LaValle, *Visibility-based pursuit evasion with bounded speed*, Int. J. Robotics Research, 27, 1350, 2008.
- [38] J.N. Tsitsiklis, *Efficient algorithms for globally optimal trajectories*, IEEE Transactions on Automatic Control, 40(9), 1528–1538, 1995.
- [39] A. Vladimirsky, *Static pdes for time-dependent control problems*, Inter. Free Bound., 8, 281–300, 2006.
- [40] J. Yu and S.M. LaValle, *Tracking hidden agents through shadow information spaces*, in Proceedings IEEE International Conference on Robotics and Automation, 2008.
- [41] Z. Zhou, H. Huang, J. Ding, R. Takei, and C.J. Tomlin., *An open-loop framework for reach-avoid games*, submitted.
- [42] Z. Zhou, R. Takei, H. Huang, and C.J. Tomlin, *A general, open-loop formulation for reach-avoid games*, in Decision and Control (CDC), 2012 IEEE 51st Annual Conference on, 6501–6506, 2012.