Speed vs Accuracy in Goal Recognition for Time-Sensitive Applications: a Game-Theoretic Approach

Sara Bernardini
University of Oxford
Oxford, United Kingdom
sara.bernardini@cs.ox.ac.uk

Fabio Fagnani
Politecnico di Torino
Torino, Italy
fabio.fagnani@polito.it

Santiago Franco
Royal Holloway University of London
London, United Kingdom
santiago.francoaixela@rhul.ac.uk

ABSTRACT

This work addresses a specific instance of Goal Recognition (GR), termed *time-sensitive GR*, where a malicious actor (the *attacker*) seeks to reach and damage one of several sensitive targets, while the observer (the *defender*) must identify the attacker's target and allocate limited resources to protect it. Focusing on real-world physical and cyber security scenarios, the defender faces a trade-off between acting early, with limited information, or waiting for more data but risking insufficient time to defend. Our contributions include introducing a game-theoretic formulation of this instance of GR, which captures the time-sensitive nature of these scenarios, and providing an efficient method to compute Nash equilibria using the fictitious play learning scheme. Experimental results confirm that our method equips the defender with robust policies, outperforming less adaptable strategies.

KEYWORDS

AI Planning, Multi-Agent Planning, Goal Recognition, Game Theory, Security Applications, Time-Sensitive Applications

ACM Reference Format:
Sara Bernardini, Fabio Fagnani, and Santiago Franco. 2025. Speed vs Accuracy in Goal Recognition for Time-Sensitive Applications: a Game-Theoretic Approach. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 9 pages.

1 INTRODUCTION

Goal recognition (GR) refers to the problem where an agent, known as the observer, monitors the actions of another agent, the actor, to infer its final goal. Since its inception as a subproblem of plan recognition [37], GR has been extensively studied [21, 40] due to its applications in various fields (e.g. [15, 19, 23, 24, 35]).

In this work, we study a new variant of GR, which we refer to as *time-sensitive GR* (TSGR), where the actor is a malicious agent - referred to as the *attacker* - that navigates an environment to reach and damage one of several sensitive targets. The observer, termed the *defender*, aims to identify the attacker's intended target as early as possible by analyzing its actions, allowing the defender to reinforce the target's defense. We focus on real-world physical and cyber security scenarios where the defender has limited resources to protect targets and must relocate those resources from an initial position to the selected target. A military example is a defender moving troops from a base to a threatened location (e.g., [36]). Because moving resources takes time, delayed action by the defender may result in failure to secure the target before the attacker reaches it. Depending on the environment's layout and agents' behaviors, the defender must eventually make an irrevocable decision on which target to protect. Procrastination risks leaving the defender unable to mount a sufficient defense.

In many cases, the defender will need to choose a target after observing only a portion of the attacker's trajectory. Early decisions give the defender more time to reposition resources while the attacker is still distant from the target. However, premature decisions risk inaccuracies, as the observed portion of the attacker's path might still be consistent with multiple potential targets. On the other hand, waiting for more information would allow the defender to make better predictions but could leave insufficient time to act. The defender's challenge, therefore, is to choose the right moment to commit resources—when confidence is high enough to ensure the correct target is chosen while there is still time to defend it.

To address this trade-off and support timely decision-making, after defining the TSGR problem formally, we model it as a *two-player zero-sum game*. We introduce a *protection success probability* for each target, which depends on the distance between the target and the attacker's current position. This probability is factored into the agents' rewards. We construct *defender policies* that map prefixes of the attacker's observed paths to specific targets. In our formulation, the attacker's strategy is to choose a path to a target, while the defender's strategy is to determine a prefix of each possible path and assign a target to it.

By playing multiple iterations of this game, both the attacker and defender refine their strategies to maximize their rewards. The existence of Nash equilibria [25] in this game follows directly from classical min-max theorems, although computing these equilibria is challenging due to the large action sets available to both players. To overcome this, we provide a *combinatorial* characterization of the players' best response sets and show that computing the defender's best responses is equivalent to solving an *optimal stopping time* problem [7]. We then apply the *fictitious play learning scheme* [33], which converges asymptotically to a Nash equilibrium. Our experimental results demonstrate that this approach outperforms less flexible strategies.

As we will explore in the next section, our work stands apart from existing GR literature by focusing on time-critical scenarios, where the defender must make decisions within stringent time constraints. It also differs from traditional attacker-defender games, where the defender lacks the ability to observe the attacker's behavior over time. Therefore, our key contributions lie in introducing

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), A. El Fallah Seghrouchni, Y. Vorobeychik, S. Das, A. Nowe (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

a novel instance of GR in time-sensitive domains, which captures the demands of critical real-world security challenges, offering a game-theoretic formulation of it and developing an efficient method for finding Nash equilibria, empowering the defender with robust policies to counter the attacker.

2 RELATED WORK

Game theory offers a sound mathematical approach to modeling various problems across many disciplines. In the security field, game theory is popular [3, 11, 41] because it allows us to reason on how to deploy limited security resources to maximize their effectiveness.

Our work follows this tradition. In particular, our approach is related to *attacker-defender games* [12], which focus on studying strategic interaction between defenders and adversaries via game theory. Instances of attacker-defender games are *infrastructure/asset protection games*, which deal with the allocation of defensive resources among multiple targets and the target selection decisions that adversaries make after observing the resource allocation [4, 9, 29, 34]. Typically, governments want to protect vulnerable assets (e.g., airports, ports, bridges, etc.) that an adversary wants to take down. These games are played sequentially, with the defender moving first by allocating resources to targets and the attacker moving second by deciding which target to damage, and are solved for Nash equilibria. *Patrolling games*, on the other hand, typically focus on the identification of optimal patrol routes, schedules, and postures [26, 38]. These problems have often been modeled as Stackleberg games [1, 14] and deployed in real-world scenarios, e.g., Los Angeles airport and Port of Boston [41]. A further specialization of Stackleberg games is represented by *security games*, which are non-zero-sum games helpful to model situations in which the defender and attacker attach different importance to the targets [13, 28, 41].

Our framework fundamentally differs from attacker-defender games because it focuses on a defender that can observe the attacker's behavior and modify its strategy based on its inference of the attacker's intentions. This element of goal recognition is not present in attacker-defender games, which focus on the opposite protocol, with the defender choosing first and the attacker responding to this choice.

GR [40] is the other stream of work to which our approach is inspired. It involves an agent (the observer) inferring another agent (the actor)'s goal by observing its actions. GR is a sub-problem of plan recognition (PR), which can be classified into three categories [6]: *keyhole* recognition, in which the actor does not change its behavior because of being observed; *intended* recognition, whereby the actor attempts to reveal its real goal to the observer; and *adversarial* recognition when the actor tries to hide its real goal. Most work has been performed in keyhole PR and GR, and various approaches have been developed, from heuristic classical search and graph theory [2, 10, 27, 31, 32] to machine learning [8, 22, 23, 44].

A few game-theoretic approaches to adversarial PR have been developed. Braynov [5] offers a conceptual framework that models the interaction between adversarial planning and adversarial PR as a two-player zero-sum game over *attack graphs* representing the attacker's possible plans. In Lisy et al. [16], the problem of adversarial PR is defined as an imperfect information two-player zero-sum game between an actor and an observer. However, this method requires that a plan library is provided explicitly.

The literature on game-theoretic solutions to adversarial GR is very limited. Ang et al. [2] present an approach in which they model GR as a stochastic game with incomplete information. Their approach concentrates on the defender identifying targets based on the current attacker's position during an online search. We offer a flexible and scalable solution to TSGR by providing the defender with a policy calculated offline that allows it to make timely decisions on which target to protect based on the entire attacker's behavior until the time of the decision.

3 PROBLEM STATEMENT

3.1 Preliminaries

We consider directed graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a (finite) set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of edges. A *walk* in $\mathcal{G}$ from a node v to a node w is any finite sequence of nodes $\gamma = (\gamma_0 = v, \gamma_1, \ldots, \gamma_l = w)$, where $(\gamma_i, \gamma_{i+1}) \in \mathcal{E}$ for every $i = 0, \ldots, l-1$. The symbol $E(\gamma) = w$ denotes the *end node* of the walk γ and $l(\gamma)$ its length l.

We define a *weight* matrix $W \in \mathbb{R}_+^{\mathcal{V}\times\mathcal{V}}$ on $\mathcal{G}$ such that, for every $(v, w) \in \mathcal{E}$, $W_{vw} > 0$. Given a walk γ, we put $W(\gamma) = \sum_{i=0}^{l-1} W_{\gamma_i\gamma_{i+1}}$.

Given two walks $\gamma' = (\gamma'_0, \gamma'_1, \ldots, \gamma'_{l'})$ and $\gamma'' = (\gamma''_0, \gamma''_1, \ldots, \gamma''_{l''})$ with $\gamma'_{l'} = \gamma''_0$, we indicate their *concatenation* as $\gamma' \perp \gamma'' = (\gamma'_0, \ldots, \gamma'_{l'}, \gamma''_1, \ldots, \gamma''_{l''})$. Given a walk $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_l)$ in $\mathcal{G}$ and a non negative integer $k \leq l$, we call $\gamma^{(k)} = (\gamma_0, \gamma_1, \ldots, \gamma_k)$ the *prefix* of γ of length k and $\gamma^{(k+)} = (\gamma_k, \ldots, \gamma_l)$ the *suffix* of γ of length $l - k$. Note that $\gamma = \gamma^{(k)} \perp \gamma^{(k+)}$. Given a set of walks $\mathcal{P}$ and a non-negative integer k, we denote the set of prefixes of length k of all walks in $\mathcal{P}$ having a length not smaller than k by $\mathcal{P}^{(k)}$.

We call $\mathcal{L}_\mathcal{P} = \bigcup_{k\geq 0} \mathcal{P}^{(k)}$ the *language* generated by $\mathcal{P}$, namely the set of all prefixes of walks in $\mathcal{P}$. Given a prefix $\alpha = (\alpha_0, \ldots, \alpha_l) \in \mathcal{L}_\mathcal{P}$, we indicate with $\mathcal{P}_\alpha^+$ the set of suffixes γ that start in α_l and are possible continuations of α, namely $\alpha \perp \gamma \in \mathcal{P}$. Finally, we put $\mathcal{P}_\alpha = \{\alpha \perp \gamma \mid \gamma \in \mathcal{P}_\alpha^+\}$.

Given a set A and $\bar{a} \in A$, we denote by $\delta^{\bar{a}} \in \mathbb{R}_+^A$ the probability vector that is always 0 except in position $\bar{a}$, where its value is 1.

3.2 Time-Sensitive Goal Recognition

To define our problem, we build on the path-planning goal recognition problem introduced and investigated in depth by Masters and Sardina [17–20]. Following them, we start considering a *path-planning* domain $\mathcal{D} = \langle \mathcal{G}, W \rangle$, where $\mathcal{G}$ is a directed graph and W is a weight matrix. $\mathcal{G}$ models the environment where the agents move, and the weight matrix represents the cost an agent incurs for moving along the graph's edges. A *path-planning goal recognition* (PPGR) problem [17] is a tuple $\mathcal{R} = \langle \mathcal{D}, s, \mathcal{T}, \mathcal{O}, \lambda \rangle$, where $\mathcal{D}$ is a path-planning domain, $s \in \mathcal{V}$ is the origin node, $\mathcal{T} \subseteq \mathcal{V} \setminus \{s\}$ is a set of targets, $\mathcal{O} = (o_1, \ldots, o_k)$, where $k \geq 0$ and $o_i \in \mathcal{V}$, is a sequence of observations, with $o_1 \neq s$, and λ is an apriori probability distribution on the set of targets $\mathcal{T}$. We assume that nodes in $\mathcal{T}$ have only incoming edges, the origin s has only outgoing edges, and, for every $d \in \mathcal{T}$, there exists at least a walk in $\mathcal{G}$ from s to d. The PPGR tuple $\mathcal{R}$ has the following interpretation. Two agents are present in the environment: i) an actor that, starting from the origin node s, moves over the graph $\mathcal{G}$ to reach a destination in

$\mathcal{T}$ according to distribution λ, and ii) an observer that monitors the actor's movements and aims to establish which target in $\mathcal{T}$ the actor wants to reach based on the gathered observations O. The solution to a PPGR problem $\mathcal{R}$ is a posterior probability distribution over $\mathcal{T}$ given a sequence of observations O: $Pr(\mathcal{T}|O)$.

To model our TSGR problem, we augment PPGR by adding two new elements, a set of *candidate walks* $\mathcal{P}$ and a *protection success probability* q, obtaining the tuple $\langle \mathcal{R}, \mathcal{P}, q \rangle$, where $\mathcal{R}$ is a PPGR problem, $\mathcal{P}$ is a fixed set of walks in $\mathcal{G}$, namely $\mathcal{P} = \bigcup_{d \in \mathcal{T}} \mathcal{P}(d)$, with $\mathcal{P}(d)$ being a set of walks in $\mathcal{G}$ from s to target $d \in \mathcal{T}$, and q is a function $q : \mathbb{R}_+ \to [0, 1]$. The new tuple can be understood as follows.

In our setting, the actor is an *attacker* that moves over the graph $\mathcal{G}$ by choosing one of the walks in $\mathcal{P}$ to reach and attack one of the targets in $\mathcal{T}$. We assume the attacker has a specific resource budget and generates only walks $\mathcal{P}$ within it. The attacker is goal-oriented, i.e., it has a predefined target in $\mathcal{T}$ it wants to attack and will not change that over time.

The observer is a *defender* tasked with protecting the targets $\mathcal{T}$. Its objective is to identify the attacker's intended target as quickly as possible, so it can reinforce the defense at that location. Once the defender gathers enough evidence to confidently infer the attacker's target, it reallocates resources to fortify that target. The defender's success depends on correctly identifying the attacker's target before the attacker reaches it, as well as the distance between the attacker and the target at the time of the defender's decision.

More formally, we encode the defender's capacity to defend a target as the protection success probability $q : \mathbb{R}_+ \to [0, 1]$, where the value $q(x)$ is the probability of the defender successfully defending a target when the attacker is at a distance x from it. This interpretation reflects the fact that we focus on scenarios in which the defender needs to relocate resources to protect a target, and its chances of doing so on time depend on how far the attacker is from it. If the attacker chooses a walk γ and the defender decides to protect target d after k steps, the reward the defender obtains is $q(W(\gamma^{(k+)}))$ if it guesses the right target, i.e. $d = E(\gamma)$, and 0 if not, i.e. $d \neq E(\gamma)$.

We assume that the walks $\mathcal{P}$ and the probability distribution λ are known to both agents and that, while the defender does not know which specific target the attacker is aiming for, it can precisely monitor the attacker's movements. In consequence, observations O are elements of $\mathcal{L}_{\mathcal{P}}$ and, from now on, we just refer to the latter.

To define the solution to a TSGR problem, we first introduce the concept of prefix cut.

Definition 1. *A* prefix cut *for $\mathcal{P}$ is any subset $C \subseteq \mathcal{L}_{\mathcal{P}}$ such that $|C \cap \{\gamma^{(k)} \mid k = 0, 1, \ldots, l(\gamma)\}| = 1$ for each $\gamma \in \mathcal{P}$.*

A solution to a TSGR problem consists of a pair (C, T), where C is a prefix cut of $\mathcal{P}$ and T is a map such that $T : C \to \mathcal{T}$. The pair (C, T) has the following interpretation: the defender will protect target $T(c)$ after observing prefix $c \in C$ of the attacker's trajectory. We refer to the map T as a *deterministic $\mathcal{T}$-estimator*.

Given a prefix cut C, the optimality of T corresponds to a generalized maximum a-posteriori estimator that considers the distance to the targets. The challenge is the determination of the prefix cut C, which comes from trading time for accuracy in the choice of the defender's decision time. Both elements C and T are dependent on the form of the probability function q. A pair (C, T) fixes a policy that, for any walk in $\mathcal{P}$ that the attacker can take, establishes how many observations the observer should make along it before deciding which target to protect. The goal is to maximize the probability that the defender chooses the right target when it still has sufficient time to defend it. Note the difference with PPGR, which provides a mapping from a specific set of observations to the most probable target but does not compute a policy for any choice of the actor and does not consider any temporal element in the GR process.

4 A GAME-THEORETIC FORMULATION

To solve the TSGR problem, we cast it as a *two-player strategic game* where one player is the attacker, and the other is the defender. Our problem naturally lends itself to being modeled within a game-theoretic framework because it involves two opponents with opposite goals. Game theory allows the defender to reason about how to deploy limited security resources in a timely manner to maximize their effectiveness.

4.1 An Attacker-Defender Game

An *attacker's strategy* consists in choosing a subset Q of $\mathcal{P}$ such that $|Q \cap \mathcal{P}(d)| = 1 \quad \forall d \in \mathcal{T}$, i.e. Q is the set of walks that the attacker will use to reach one of the possible targets.

A *defender's strategy* coincides with a TSGR solution, so it is a pair (C, T), where C is a prefix cut of $\mathcal{P}$ and T is a deterministic $\mathcal{T}$-estimator.

Remark 2. A simple defender's strategy is to make a decision at time 0 when the attacker is yet to start moving and pick a specific target d (we will discuss which d the defender might choose below). We notice that this corresponds to choosing the prefix cut $C_o = \{o\}$ and the deterministic $\mathcal{T}$-estimator T_o such that $T_o(o) = d$.

We formalize the game as below.

Definition 3. *We consider an adversarial game defined as follows. The set of the attacker's strategies is defined as*

$$\mathbb{M}^A = \{Q \subseteq \mathcal{P} \mid |Q \cap \mathcal{P}(d)| = 1 \quad \forall d \in \mathcal{T}\}$$

The set of the defender's strategies is defined as

$$\mathbb{M}^D = \{(C, T) \mid C \text{ prefix cut of } \mathcal{P},\ T : C \to \mathcal{T}\}$$

Given $Q \in \mathbb{M}^A$ and $(C, T) \in \mathbb{M}^D$, we define, for every $\gamma \in Q$, $c_\gamma = \gamma^{(k)} \in C$ to be the only prefix of γ in C. We then put

$$q_\gamma^{(C,T)} = \begin{cases} q(W(\gamma^{(k+)})) & \text{if } E(\gamma) = T(c_\gamma) \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

$q_\gamma^{(C,T)}$ represents the defender's reward when it uses the deterministic strategy (C, T) and the attacker chooses the motion trajectory γ. Averaging over the set Q, we obtain the defender's reward when it plays (C, T) and the attacker plays Q. Formally, we define:

$$\phi(Q, (C, T)) = \sum_{\gamma \in Q} \lambda(E(\gamma)) q_\gamma^{(C,T)} \tag{2}$$

We can interpret the setup as a zero-sum, two-player game in strategic form. The two players have, respectively, action sets $\mathbb{M}^A$ and $\mathbb{M}^D$. Given a pair of actions $(Q, (C, T))$, the utility of the first player, the attacker, is $-\phi(Q, (C, T))$, whereas the utility of the second player, the defender, is $\phi(Q, (C, T))$.

To solve the TSGR problem, we let the two players play and determine a Nash equilibrium offline by applying the algorithm presented below. Both players know the game rules and can run the algorithm independently. Once the players have computed their strategies, they merely play them in any online game run and cannot modify their actions as time passes. Hence, our approach differs from setups where the two players react online to the adversary's choice and modify their actions over time. An online version of the TSGR problem, where the two players learn the optimal policy by reacting to the adversary's current choice, is left for future research.

4.2 Mixed Strategies and Nash Equilibria

Nash equilibria for zero-sum games only exist as *mixed strategies*. We define $\mathbb{S}^A$ and $\mathbb{S}^D$ as the set of probability distributions over, respectively, the sets $\mathbb{M}^A$ and $\mathbb{M}^D$. Respectively, elements will be denoted with symbols σ_A and σ_D. The mixed strategy extension of the 2-player game in Def. 3 is obtained by averaging the reward function defined in Eq. (2) over the probabilistic strategies of the two players. Formally, the mixed strategy defender's reward is a function $\Phi : \mathbb{S}^A \times \mathbb{S}^D \to \mathbb{R}$ given by the expression

$$\Phi(\sigma_A, \sigma_D) := \sum_{Q \in \mathbb{M}^A} \sum_{(C,T) \in \mathbb{M}^D} \sigma_A(Q)\sigma_D(C,T)\phi(Q,(C,T)) \quad (3)$$

Deterministic strategies can be thought as particular mixed strategies: if $Q \in \mathbb{M}^A$ and $(C,T) \in \mathbb{M}^D$, we indicate with the symbol δ^Q and $\delta^{(C,T)}$ the corresponding mixed strategies defined as delta distributions on, respectively, the strategy Q and the strategy (C,T). Following this interpretation, we will consider the inclusions $\mathbb{M}^A \subseteq \mathbb{S}^A$ and $\mathbb{M}^D \subseteq \mathbb{S}^D$.

Best response sets gather the best actions that one of the players can take in response to a specific action of the other player. Formally, for every attacker's strategy $\sigma_A \in \mathbb{S}^A$ and defender's strategy $\sigma_D \in \mathbb{S}^D$, we put

$$\begin{aligned} \mathcal{B}_D(\sigma_A) &= \operatorname*{argmax}_{\sigma_D \in \mathbb{S}^D} \Phi(\sigma_A, \sigma_D) \\ \mathcal{B}_A(\sigma_D) &= \operatorname*{argmin}_{\sigma_A \in \mathbb{S}^A} \Phi(\sigma_A, \sigma_D) \end{aligned} \quad (4)$$

We also define *deterministic best response sets*:

$$\mathcal{B}_D^{det}(\sigma_A) = \mathcal{B}_D(\sigma_A) \cap \mathbb{M}_D, \qquad \mathcal{B}_A^{det}(\sigma_D) = \mathcal{B}_A(\sigma_D) \cap \mathbb{M}_A \quad (5)$$

It is well known that $\mathcal{B}_D(\sigma_A)$ and $\mathcal{B}_A(\sigma_D)$ can be represented as convex hulls of, respectively, $\mathcal{B}_D^{det}(\sigma_A)$ and $\mathcal{B}_A^{det}(\sigma_D)$.

A pair $(\sigma_A, \sigma_D) \in \mathbb{S}^A \times \mathbb{S}^D$ is called a *Nash equilibrium* if, simultaneously, $\sigma_D \in \mathcal{B}_D(\sigma_A)$ and $\sigma_A \in \mathcal{B}_A(\sigma_D)$. In this case, neither of the two players is incentivized to modify their behavior unilaterally. We indicate with $\mathcal{N}$ the set of Nash equilibria of our game. Von Neumann's Minimax theorem [42] guarantees the existence of Nash equilibria and their characterization as min-max optimizers. Precisely, if we define

$$\mathbb{S}^{D*} = \operatorname*{argmax}_{\rho \in \mathbb{S}^D} \min_{\mu \in \mathbb{S}^A} \Phi(\mu, \rho) \qquad \mathbb{S}^{A*} = \operatorname*{argmin}_{\mu \in \mathbb{S}^A} \max_{\rho \in \mathbb{S}^D} \Phi(\mu, \rho) \quad (6)$$

the set of all Nash equilibria is $\mathcal{N} = \mathbb{S}^{A*} \times \mathbb{S}^{D*}$ and Φ is constant on $\mathcal{N}$. We denote with Φ^* this value of Φ on the set $\mathcal{N}$. By construction, playing a Nash equilibrium gives a player a guaranteed reward, regardless of what the other player does. In particular,

$$\Phi(\sigma_A, \sigma_D^*) \geq \Phi(\sigma_A^*, \sigma_D^*), \quad \forall (\sigma_A^*, \sigma_D^*) \in \mathcal{N}, \quad \forall \sigma_A \in \mathbb{S}^A$$

We aim to obtain a computationally efficient algorithm to compute a Nash equilibrium for the above adversarial game. While such quadratic Minimax problems are classic and many algorithms are in principle available, including a well-known reduction to a linear programming problem [30], the challenge is dealing with the size of the defender's strategy space. Even for target sets with bounded sizes, the set of walks $\mathcal{P}$ and the set of prefixes will grow exponentially in the number of nodes n. In some cases, the set of prefix cuts might have a hyper-exponential size, and it is challenging to operationalize it. Our approach is to use a classical learning technique, the *fictitious play* algorithm [33], based on an alternate, iterative computation of best response strategies by the two players. We will show that computing the best responses for the attacker is relatively simple while, for the defender, it is equivalent to solving an *optimal stopping time* problem [7]. This can be done by applying a backward induction method that leads to an optimal prefix cut without requiring a general characterization of all prefix cuts.

4.3 The Analysis of an Example

We now analyze a simple example for which Nash equilibria can be computed analytically.

First, we restrict the defender's strategies; in particular, we say that a prefix cut C_1 is *minimal* if we cannot find another prefix cut C_2 and a bijection $\theta : C_1 \to C_2$ such that $\theta(c)$ is a prefix of c for every $c \in C_1$. In adddition, given a prefix cut $C \subseteq \mathcal{L}_{\mathcal{P}}$, a $\mathcal{T}$-estimator $T : C \to \mathcal{T}$ is called *natural* if, for any $\alpha \in C$ such that $\mathcal{T}(\alpha) = \{\bar{d}\}$, it holds that $T(\alpha) = \bar{d}$. The defender has no advantage in using strategies (C,T) such that C is not minimal and T is not natural. In game theory, such strategies are called *weakly dominated*, which means that we can always find another strategy (C',T') such that $\Phi(\sigma_A, \delta^{(C,T)}) \leq \Phi(\sigma_A, \delta^{(C',T')})$ for every $\sigma_A \in \mathbb{S}^A$, with sharp inequality for at least one σ_A. While, in general, Nash equilibria containing weakly dominated strategies might exist, it is always possible to find Nash equilibria that do not contain weakly dominated strategies. In the analysis of the example, we focus on the defender's strategies (C,T) with C minimal and T natural. They are denoted by $\mathbb{M}^D_{und}$.

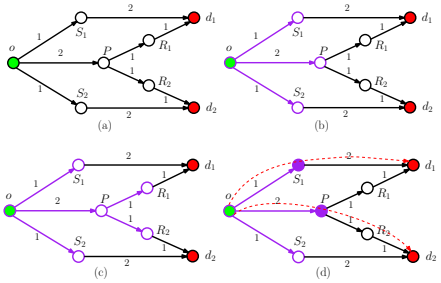


Figure 1: In Fig. (a), a graph $\mathcal{G}$ is depicted, with the origin o in green and the targets d_1 and d_2 in red. In Fig. (b) and (c), the subgraphs colored in purple represent the prefix cuts C_1 and C_2, respectively. Fig. (d) visualizes prefix-cut C_1 against the strategy chosen by the attacker. In solid purple, we show the nodes where the defender will make a decision if it decides to employ prefix cut C_1.

Consider the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ depicted in Figure 1(a), where the values of the weight matrix W are the numbers indicated close to the edges, and the origin is in green color. The set of targets is $\mathcal{T} = \{d_1, d_2\}$ (in red color in Figure 1). We assume that $\lambda(d_1) = \lambda(d_2) = 1/2$. We consider the family of walks $\mathcal{P} = \{oS_1d_1, oPR_1d_1, oS_2d_2, oPR_2d_2\}$. We consider two different successful protection probabilities $q(x) = x/4$ and $q(x) = \min\{x/3, 1\}$. In the first case, the probability decreases linearly, and its maximum value of 1 is reached when the attacker is at a distance of 4 from the targets. The second function has a saturation effect: when the attacker is at a distance ≥ 3, the protection probability is 1.

The set of the attacker's deterministic strategies is

$$\mathbb{M}^A = \{ \begin{array}{l} Q_1 = \{oS_1d_1, oS_2d_2\}, Q_2 = \{oPR_1d_1, oS_2d_2\}, \\ Q_3 = \{oS_1d_1, oPR_2d_2\}, Q_4 = \{oPR_1d_1, oPR_2d_2\}\} \end{array} \tag{7}$$

The set of prefixes is $\mathcal{L}_{\mathcal{P}} = \{o, oS_1, oS_1d_1, oS_2, oS_2d_2, oP, oPR_1, oPR_1d_1, oPR_2, oPR_2d_2\}$. The minimal prefix cuts and corresponding defender's deterministic strategies are given by

$$\begin{array}{l} C_o = \{o\}, C_1 = \{oS_1, oS_2, oP\}, C_2 = \{oS_1, oS_2, oPR_1, oPR_2\} \\ \mathbb{M}^D_{und} = \{(C_o, T_o^1),\ (C_o, T_o^2),\ (C_1, T_1^1),\ (C_1, T_1^2),\ (C_2, T_2)\} \end{array} \tag{8}$$

with $T_o^j(o) = d_j, T_1^j(oS_k) = d_k, T_1^j(oP) = d_j, T_2(oS_k) = T_2(oPR_k) = d_k$ for $j, k = 1, 2$. Prefix cuts C_1 and C_2 are depicted in purple in Figure 1(b) and (c), respectively.

Finally, we introduce two mixed strategies for the defender, which correspond to choosing prefix cut, respectively C_0 and C_1, with a uniformly random $\mathcal{T}$-estimator. They will be useful in representing the Nash equilibria.

$$\sigma_D^{C_0} = 1/2\left(\delta^{(C_0,T_0^1)} + \delta^{(C_0,T_0^2)}\right),\ \sigma_D^{C_1} = 1/2\left(\delta^{(C_1,T_1^1)} + \delta^{(C_1,T_1^2)}\right) \tag{9}$$

The following result describes the Nash equilibria of this game for the two q functions introduced above.

Proposition 4. *The following results hold:*

(i) *Assume that $q(x) = x/4$. Then (σ_D, σ_A) is a Nash equilibrium if and only if*

$$\begin{array}{ll} \sigma_D = (2/3)\sigma_D^{C_0} + p_1\sigma_D^{C_1} + p_2\delta^{(C_2,T_2)} & p_1 + p_2 = 1/3 \\ \sigma_A = (2/3)\delta^{Q_1} + (1/3)\delta^{Q_4} & \end{array}$$

(ii) *Assume that $q(x) = \min\{x/3, 1\}$. Then, (σ_D, σ_A) is a Nash equilibrium if and only if*

$$\begin{array}{lll} \sigma_D = p_0^1\delta^{(C_0,T_0^1)} + p_0^2\delta^{(C_0,T_0^2)} & p_0^1 + p_0^2 = 1 & \\ \sigma_A = q_1\delta^{Q_1} + q_4\delta^{Q_4} & 0 < q_1 < 1/2 & q_1 + q_4 = 1 \end{array}$$

Proposition 4's proof is in the appendix at GithubLink.

At a Nash equilibrium, the attacker employs a combination of two strategies: Q_1, which favors shorter but less ambiguous walks, and Q_4, which favors longer and more ambiguous walks. In the first scenario ($q(x) = x/4$), the attacker chooses Q_1 two-thirds of the time, and the defender makes a decision when the attacker is at the origin two-thirds of the time and for the rest uses a combination of prefix cuts C_1 and C_2. In contrast, in the second scenario ($q(x) = \min\{x/3, 1\}$), the attacker adopts the Q_4 strategy at least half the time, and the defender always makes a decision when the attacker is at the origin. The saturation effect in the second scenario diminishes the advantage that the shorter walks in Q_1 offer to the attacker in the first scenario, making them no more favorable than the longer ones in Q_4. Given this, the defender's best response is to make an immediate decision when the attacker is still at the origin rather than applying the prefix cuts C_1 and C_2 as in the first case.

5 BEST RESPONSES AND NASH EQUILIBRIA

This section offers insights into the characterization and computation of the deterministic best response sets given in Eqs. (5). We need to introduce a few preliminary concepts before doing that.

First, we notice that the pair composed of the probability distribution λ on $\mathcal{T}$ and an attacker's strategy $\sigma_A \in \mathbb{S}_A$ gives rise to a probability distribution μ_{σ_A} on $\mathcal{P}$ as follows

$$\mu_{\sigma_A}(\gamma) = \lambda(E(\gamma)) \sum_{Q\,:\,Q \ni \gamma} \sigma_A(Q), \qquad \gamma \in \mathcal{P} \tag{10}$$

In other words, the probability that the attacker will choose walk γ is given by the probability of selecting target $E(\gamma)$ and the probability of selecting a strategy Q containing γ.

Any probability distribution μ on $\mathcal{P}$ can be extended to all prefixes via saturation: given a prefix $\alpha \in \mathcal{L}_{\mathcal{P}}$, we put $\mu(\alpha) = \sum_{\gamma \in \mathcal{P}_\alpha} \mu(\gamma)$. Below, we will need to consider various conditioned versions of such distributions, and hence, we introduce the following notation. Assume μ is a probability distribution on $\mathcal{P}$, then

- Given a target d, $\gamma \in \mathcal{P}(d)$ and $\alpha \in \mathcal{L}_{\mathcal{P}(d)}$, we put
$$\mu^{|d}(\gamma) = \mu(\gamma)/\mu(\mathcal{P}(d)), \quad \mu^{|d}(\alpha) = \mu(\alpha)/\mu(\mathcal{P}(d))$$
- Given a prefix $\alpha \in \mathcal{L}_{\mathcal{P}}$, $\gamma \in \mathcal{P}_\alpha^+$ and a target d, we define the conditional probabilities on $\mathcal{P}_\alpha^+$ as
$$\mu^{|\alpha}(\gamma) = \mu(\alpha\perp\gamma)/\mu(\alpha), \quad \mu^{|d,\alpha}(\gamma) = \mu^{|d}(\alpha\perp\gamma)/\mu^{|d}(\alpha)$$
- Given a prefix $\alpha \in \mathcal{L}_{\mathcal{P}}$, we define the a-posteriori probability on targets by setting $\lambda_\mu^{|\alpha}(d) = \sum_{\gamma \in \mathcal{P}_\alpha(d)} \mu^{|\alpha}(\gamma)$.

We also associate a function ρ_{σ_D} on $\mathcal{L}_{\mathcal{P}} \times \mathcal{T}$ to every defender's strategy $\sigma_D \in \mathbb{S}_D$ defined as follows

$$\rho_{\sigma_D}(\alpha, d) = \sum_{\substack{C\ \ni\ \alpha \\ T(\alpha) = d}} \sigma_D(C, T) \tag{11}$$

Given a walk $\gamma \in \mathcal{P}$, for every α that is a prefix of γ and for every target $d \in \mathcal{T}$, the term $\rho_{\sigma_D}(\alpha, d)$ represents the probability that, if the attacker uses γ, the defender equipped with strategy σ_D will make a decision when the attacker has reached α and will choose to defend target d.

5.1 Deterministic Best Responses

Finding deterministic best response strategies for the attacker is relatively straightforward from a computational point of view because the attacker's minimization problem decouples along the set of trajectories to the targets. Formally, we have the following result.

Proposition 5. *Given a defender's strategy $\sigma_D \in \mathbb{S}^D$, a deterministic strategy $Q = \{\gamma_d \in \mathcal{P}(d) \mid d \in \mathcal{T}\}$ is in $\mathcal{B}_A^{det}(\sigma_D)$ if and only if, for every $d \in \mathcal{T}$, it holds that*

$$\gamma_d \in \operatorname*{argmin}_{\gamma \in \mathcal{P}(d)} \sum_{k=0}^{l(\gamma)} \rho_{\sigma_D}(\gamma^{(k)}, d) q(w(\gamma^{(k+)})) \tag{12}$$

Proof From Eqs. (2), (3) and (11), the reward can be rewritten as follows:

$$\begin{aligned}\Phi(\delta^{\mathcal{Q}},\sigma_D) &= \sum_{d\in\mathcal{T}}\lambda(d)\sum_{(C,T)\in\mathbb{M}_D}\sigma_D(C,T)q_{\gamma_d}^{(C,T)}\\ &= \sum_{d\in\mathcal{T}}\lambda(d)\sum_{k=0}^{l(\gamma_d)}\sum_{\substack{C\ni\gamma^{(k)}\\ T(\alpha)=d}}\sigma_D(C,T)q(w(\gamma^{(k+)}))\\ &= \sum_{d\in\mathcal{T}}\lambda(d)\sum_{k=0}^{l(\gamma_d)}\rho_{\sigma_D}(\gamma^{(k)},d)q(w(\gamma^{(k+)}))\end{aligned} \tag{13}$$

This yields the thesis. ∎

We now focus on the computation of the deterministic best responses for the defender. This is a more complex problem because of various reasons. First, the family of prefix cuts suffers from an exponential blow-up in the length of walks in $\mathcal{P}$ even when we limit the cardinality of $\mathcal{P}$. Second, the prefix cuts are composed of words that can simultaneously be prefixes of more than one walk used by the attacker, and thus, it is not straightforward to show how the optimization problem in Eqs. (5) can be geometrically decoupled. Our main result shows that the optimization problem in the pair (C,T) can indeed be decoupled into a *Maximum-a-Posteriori (MAP) problem* to determine T and an *optimal stopping problem* to determine C.

Our first goal is to rewrite the performance function in Eq. (3) in an equivalent fashion but adapted to the defender's decision mechanism. This is the content of the next result.

Proposition 6. *Given an attacker's strategy $\sigma_A\in\mathbb{S}^A$ and a defender's deterministic strategy $(C,T)\in\mathbb{M}^D$, we have that*

$$\Phi(\sigma_A,\delta^{(C,T)}) = \sum_{\alpha\in\mathcal{L}_{\mathcal{P}}}\mu_{\sigma_A}(\alpha)\mathbb{1}_{\{\alpha\in C\}}\sum_{d\in\mathcal{T}}\mathbb{1}_{\{d=T(\alpha)\}}\bar{q}(\alpha,d) \tag{14}$$

where,

$$\bar{q}(\alpha,d) = \lambda_{\sigma_A}^{|\alpha}(d)\sum_{\gamma\in\mathcal{P}_\alpha^+(d)}\mu_{\sigma_A}^{|d,\alpha}(\gamma)q\left(w\left(\gamma\right)\right),\quad \lambda_{\sigma_A}^{|\alpha}=\lambda_{\mu_{\sigma_A}}^{|\alpha} \tag{15}$$

Proof We first rewrite Eq. (3) using Eq. (2) and the definition of μ_{σ_A} in Eq. (10) as

$$\Phi(\sigma_A,\delta^{(C,T)}) = \sum_{\gamma\in\mathcal{P}}\mu_{\sigma_A}(\gamma)q_\gamma^{(C,T)} \tag{16}$$

We now parameterize walks in $\mathcal{P}$ by first fixing a prefix, then a target, and finally a suffix compatible with that target. First note that, if $\alpha\in\mathcal{L}_{\mathcal{P}}$ and $\gamma\in\mathcal{P}_\alpha^+(d)$, it holds that $\mu_{\sigma_A}(\alpha\perp\gamma) = \mu_{\sigma_A}(\alpha)\lambda_{\sigma_A}^{|\alpha}(d)\mu_{\sigma_A}^{|d,\alpha}(\gamma)$. Using this and Eq. (1), we then compute as follows

$$\begin{aligned}&\Phi(\sigma_A,\delta^{(C,T)}) = \sum_{\alpha\in\mathcal{L}_{\mathcal{P}}}\sum_{d\in\mathcal{T}}\sum_{\gamma\in\mathcal{P}_\alpha^+(d)}(\\ &\mu_{\sigma_A}(\alpha)\lambda_{\sigma_A}^{|\alpha}(d)\mu_{\sigma_A}^{|d,\alpha}(\gamma)\mathbb{1}_{\{\alpha\in C\}}\mathbb{1}_{\{d=T(\alpha)\}}\,q(w(\gamma)))\end{aligned} \tag{17}$$

Reassembling the terms, we get the thesis. ∎

The term $\bar{q}(\alpha,d)$, defined in Eq. (15), represents the expected reward conditioned to the fact that prefix α has been observed and that the defender has picked target d. It follows from Eq. (14) that when the defender follows a best response strategy, it will necessarily select targets that maximize the quantity $\bar{q}(\alpha,d)$. Hence, we have the following result.

Corollary 7. *Given an attacker's strategy $\sigma_A\in\mathbb{S}^A$, a defender's deterministic strategy (C^*,T^*) is in $\mathcal{B}_D^{det}(\sigma_A)$ if and only if*

(i) *T^* is a Maximum-a-Posteriori (MAP) estimator, namely,*

$$T^*(\alpha)\in\operatorname*{argmax}_{d'\in\mathcal{T}}\bar{q}(\alpha,d')\quad\forall\alpha\in\mathcal{L}_{\mathcal{P}} \tag{18}$$

(ii) *Let us put $\bar{q}(\alpha)=\max_{d'\in\mathcal{T}}\bar{q}(\alpha,d')$, then we have*

$$C^*\in\operatorname*{argmax}_{C\ cut}\sum_{\alpha\in C}\mu_{\sigma_A}(\alpha)\bar{q}(\alpha) \tag{19}$$

While the maximum problem in Eq. (18) is relatively easy to solve as the target set $\mathcal{T}$ has small cardinality, the maximum problem in Eq. (19) is calculated over the set of prefix cuts, which, based on the graph, can grow exponentially big in the number of graph nodes.

The maximum problem in Ex. (19) can be interpreted as an *optimal stopping time problem* [7], for which the solution can be described through a backward induction scheme as walks in $\mathcal{P}$ have finite length. We first embed our problem into a family of related maximization problems, replacing the origin o with any possible prefix $s\in\mathcal{L}_{\mathcal{P}}$ and assuming that the attacker's motion starts from that point. Specifically, given $s\in\mathcal{L}_{\mathcal{P}}$ such that $\mu_{\sigma_A}(s)>0$, we consider the set of walks $\mathcal{P}_s^+$ that are the possible continuations of prefix s in $\mathcal{P}$ equipped with the probability measure $\mu_{\sigma_A}^{|s}$ on $\mathcal{P}_s^+$ and define

$$\Phi^*(s) = \max_C\sum_{\alpha\in C}\mu_{\sigma_A}^{|s}(\alpha)\bar{q}(\alpha)$$

where C is assumed to vary among all possible prefix cuts relative to $\mathcal{P}_s^+$. In other words, $\Phi^*(s)$ is the maximum reward the defender can obtain when it starts observing the attacker's motion after the prefix s and assumes the attacker chooses trajectories according to $\mu_{\sigma_A}^{|s}$, which is the original distribution μ conditioned to having followed prefix s. In particular, $\Phi^*(o)$ coincides with the maximum reward when the motion is observed from the origin. The values $\Phi^*(s)$ can be computed through a recursive algorithm on a Direct Acyclic Graph (DAG) whose nodes are prefixes in $\mathcal{L}_{\mathcal{P}}$. More precisely, we consider the graph $\mathcal{H}=(\mathcal{S},\mathcal{F})$, where $\mathcal{S}=\{s\in\mathcal{L}_{\mathcal{P}}\mid\mu_{\sigma_A}(s)>0\}$ and where $\mathcal{F}=\{(s,s')\in\mathcal{S}\times\mathcal{S}\mid\exists v\in\mathcal{V}\text{ with }s'=s\perp(v)\}$. The graph $\mathcal{H}$ is a DAG rooted in (o) and having the maximal prefixes, i.e. the original walks in $\mathcal{P}$, as leaves. We now consider a transition matrix Λ adapted to $\mathcal{H}$. If $s,s'\in\mathcal{S}$, we put

$$\Lambda_{ss'} = \begin{cases}\frac{\mu_{\sigma_A}(s')}{\mu_{\sigma_A}(s)} & \text{if }(s,s')\in\mathcal{F}\\ 0 & \text{otherwise}\end{cases} \tag{20}$$

We have the following result.

Theorem 8. *Facts (1), (2), and (3) below hold.*

(1) *The values $\Phi^*(s)$, as s varies in $\mathcal{S}$, satisfy the following backward induction relation:*

$$\begin{cases}\Phi^*(s)=\max\left\{\bar{q}(s),\sum_{s'\in\mathcal{L}_{\mathcal{P}}}\Lambda_{ss'}\Phi^*(s')\right\} & \forall s\in\mathcal{S}\setminus\mathcal{P}\\ \Phi^*(s)=\bar{q}(s)=q(0)\ \forall s\in\mathcal{P}\mathcal{S}\end{cases} \tag{21}$$

(2) *The set C^* of minimal prefixes s for which $\Phi^*(s)=\bar{q}(s)$ is a prefix cut of $\mathcal{P}$.*

(3) *Let $T^* : C^* \to \mathcal{T}$ be any deterministic MAP estimator (i.e., it satisfies condition* (18)*). We have that $(C^*, T^*) \in \mathcal{B}_D^{det}(\mu_{\sigma_A})$.*

Proof *Statement (1).* For simplicity, we put $\mu = \mu_{\sigma_A}$. We denote with $\mathbb{K}(s)$ the set of prefix cuts relative to the set of walks $\mathcal{P}_s^+$ that are continuations of prefix s. We notice that $\{s\} \in \mathbb{K}(s)$ and put $\mathbb{K}_>(s) = \mathbb{K}(s) \setminus \{\{s\}\}$. For every $s \in \mathcal{S}$ and $C \in \mathbb{K}(s)$, we define $F(s, C) = \sum_{\alpha \in C} \mu^{|s}\alpha)\bar{q}(s \perp \alpha)$. If $C \in \mathbb{K}_>(s)$, we can represent

$$C = \bigcup_{s' : \Lambda_{ss'} > 0} C^{(s')}$$

for some $C^{(s')} \in \mathbb{K}(s')$ so that

$$\begin{aligned} F(s, C) &= \sum_{s' \in \mathcal{S}} \Lambda_{ss'} \sum_{\alpha \in \mathcal{L}_{\mathcal{P}_{s'}^+}} \mu^{|s'}(\alpha)\bar{q}(s' \perp \alpha) \mathbb{1}_{\{\alpha \in C^{(s')}\}} \\ &= \sum_{s' \in \mathcal{S}} \Lambda_{ss'} F(s, C^{(s')}) \end{aligned} \tag{22}$$

where we use the fact that, by definition,

$$\mathcal{L}_{\mathcal{P}_s^+} = \{s\} \bigcup_{s' : \Lambda_{ss'} > 0} s' \perp \mathcal{L}_{\mathcal{P}_{s'}^+}$$

and, if $s' \in \mathcal{S}$ is such that $s' = s \perp v$ and $\alpha \in \mathcal{L}_{\mathcal{P}_{s'}^+}$,

$$\mu^{|s}(v \perp \alpha) = \frac{\mu(s \perp v \perp \alpha)}{\mu(s)} = \frac{\mu(s' \perp \alpha)}{\mu(s')} \frac{\mu(s')}{\mu(s)} = \mu^{|s'}(\alpha)\Lambda_{ss'}$$

Eq. (22) and the fact that $F(s, \{s\}) = \bar{q}(s)$ yield

$$\Phi^*(s) = \max_{C \in \mathbb{K}(s)} = \max\{\bar{q}(s), \sum_{s'} \Lambda_{ss'} \max_{C \in \mathbb{K}(s')} F(s, C^{(s')})$$

which in turn yields the top relation in Eq. (21). Notice that if $s \in \mathcal{P}$, $\mathcal{P}_s^+ = \{\epsilon\}$ and the only prefix cut in $\mathbb{K}(s)$ is $C = \{s\}$. This yields $\Phi^*(s) = \bar{q}(s) = q(0)$. This completes the proof of statement (1).

Statement (2). Consider the set C^* as defined in statement (2). Given any $\gamma \in \mathcal{P}$, there can be at most one prefix of γ in C^* by the minimality assumption. Since the set of prefixes s of γ for which $\Phi^*(s) = \bar{q}(s)$ is non-empty, as by construction $s = \gamma$ is one of them, there must exist a minimal one. This proves the statement.

Statement (3). Given Corollary (7), we only need to prove that the constructed C^* is optimal, meaning that $\Phi^*(o) = F(o, C^*)$. We first notice that, if C is any prefix cut and $s \in C$, if we pick any

$$\bar{C}_s \in \operatorname*{argmax}_{C \in \mathbb{K}(s)} F(s, C)$$

and denote $C^{(s)} = (C \setminus \{s\}) \cup \bar{C}_s$, we have that

$$F(o, C^{(s)}) - F(o, C) = \mu(s)[\Phi^*(s) - \bar{q}(s)] \geq 0 \tag{23}$$

This implies that if C is optimal, i.e. $\Phi^*(o) = F(o, C)$, necessarily it must be that $\Phi^*(s) = \bar{q}(s)$. Suppose now that C^* is not optimal and let $\bar{C}$ be an optimal prefix cut for which $|C^* \setminus \bar{C}| > 0$ is as small as possible. Let $s \in C^* \setminus \bar{C}$ and notice that, given the way C^* is defined, no proper prefix of s can be in $\bar{C}$, hence the only possibility is that $\bar{C}$ contains a subset of prefixes $\bar{C}^{(s)}$ that is in $\mathbb{K}_{>s}$. Since $\Phi^*(s) = \bar{q}(s)$, the computation in Eq. (23) implies that, if we replace $\bar{C}^{(s)}$ with $\{s\}$ inside $\bar{C}$, we obtain another optimal prefix cut. This contradicts how $\bar{C}$ is chosen and completes the proof. ■

Theorem 8 allows efficient computation of the defender's best responses based on a backward induction algorithm on the set of prefixes $\mathcal{L}_{\mathcal{P}}$ without the need for an a priori explicit characterization of prefix cuts. As discussed in the next subsection, the theorem lays the foundation for employing the fictitious play algorithm.

5.2 Fictitious Play Learning Scheme

Fictitious play [33] is a well-known recursive learning scheme used to obtain, asymptotically, a Nash equilibrium of a 2-player 0-sum game. It offers the flexibility to stop at any time with bounded approximations. For large problems (e.g., 50×50 grids), fictitious play is often faster and simpler than alternatives like the simplex method, providing sufficiently accurate solutions in less time. We employ this algorithm because it does not need an explicit description of the strategy space of the two players; it only relies on computing the best responses.

Every player remembers the distribution of actions played by the opponent in the past and, at every time step, selects a best response against it. More formally, first, initial conditions are set:

(1) Choose arbitrarily $Q_1 \in \mathbb{M}^A$ and $(C_1, T_1) \in \mathbb{M}^D$.
(2) Put $\sigma_A(1) = \delta^{Q_1}$ and $\sigma_D(1) = \delta^{(C_1, T_1)}$.

Then, the following recursive scheme is implemented for $t \geq 2$:

(3) Pick $(C_t, T_t) \in \mathcal{B}_D^{det}(\sigma_A(t-1))$ and $Q_t \in \mathcal{B}_A^{det}(\sigma_D(t-1))$ as outlined in Proposition 5 and Theorem 8.
(4) Put

$$\begin{aligned} \sigma_A(t) &= \left(1 - \tfrac{1}{t}\right) \sigma_A(t-1) + \tfrac{1}{t} \delta^{Q_t} \\ \sigma_D(t) &= \left(1 - \tfrac{1}{t}\right) \sigma_D(t-1) + \tfrac{1}{t} \delta^{(C_t, T_t)} \end{aligned}$$

We have the following classical result, proven in [33].

Theorem 9. *For $t \to +\infty$, the sequence $(\sigma_A(t), \sigma_D(t))$ converges to a Nash equilibrium of the game.*

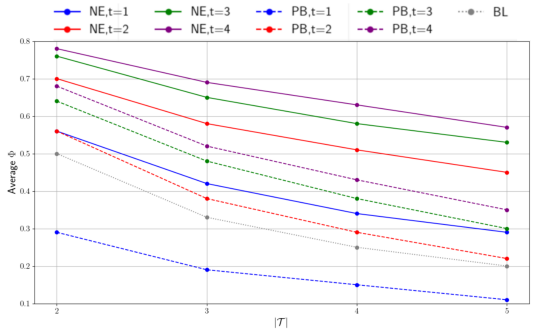
Theorem 9 ensures that the fictitious play algorithm outlined above always converges to a Nash equilibrium of the game. We notice that the algorithm's iterative step consists of the computation of a deterministic best response action for both players. As this can be accomplished through the algorithms developed in the previous subsection, we have reached a computational scheme for our min-max problems. In the next section, we will analyze, through a set of simulations, the performance of this method for our game.

6 EXPERIMENTS

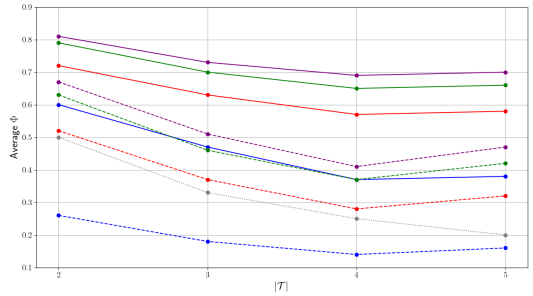
We conducted experiments on two graphs. The first is a 30x30 grid graph, with connectivity degree 4 excluding boundary nodes. The second is the Shanghai city map, with degree 8, obtained from the Moving-AI 2D pathfinding benchmarks [39], representing a city fragment as a 256x256 grid with obstacles. All edges have unit cost. These are standard benchmarks used in AI and robotics and are considerably larger than those used in related work (e.g., [2]).

We consider an increasing number of targets, $|\mathcal{T}| = 2, 3, 4, 5$. Origins and targets are selected randomly and uniformly for each problem. In each scenario, we generate a set of walks $\mathcal{P}$ containing up to a thousand walks per target using the Yen algorithm [43].

We use a protection success probability $q(x) = \min(1.0, xt/M)$, where $M = \max_{\gamma \in \mathcal{P}} W(\gamma)$ is the cost of the longest walk considered

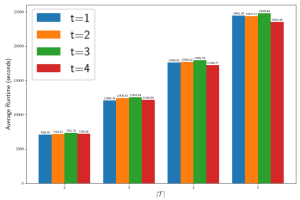


(a) 30x30 Orthogonal Grid

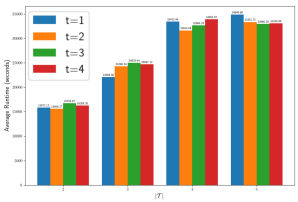


(b) 256x256 Shanghai Map

Figure 2: Average reward $\Phi(\sigma_A, \sigma_D)$ against number of targets ($|\mathcal{T}| = 2, 3, 4, 5$) for four values of the saturation parameter ($t = 1, 2, 3, 4$).



(a) 30x30 Orthogonal Grid



(b) 256x256 Shanghai Map

Figure 3: Computational cost for the FP algorithm.

and t is a parameter that takes integer values from 1 to 4. When $t = 1$, $q(x)$ is proportional to the attacker's distance from the target. As t increases, the ambiguity of the walks chosen by the attacker influences the defender's strategy more than their length, as illustrated in the example in Section 4.3.

We ran two hundred experiments for each of the two graphs and each number of targets $|\mathcal{T}| = 2, 3, 4, 5$. We performed ten thousand iterations per problem for the first graph and a thousand iterations for the second one. We used a server equipped with Intel E5-2583 cores at 2.10 GHz, with a memory cap of 4 GB per process and no time limit. See GithubLink for code and supplementary data .

We compare the reward obtained by the defender via the Fictitious Play algorithm (denoted **FP**) against the reward it would get when using one of the following two suboptimal strategies, explained below: (1) a baseline strategy (**BL**), and (2) a perimeter-based strategy (**PB**). Strategy **BL** prescribes that the defender makes a decision when the target is at the origin by choosing the furthest target possible. This corresponds to the strategy illustrated in Remark 2: $(C_o, T_o^{\bar{d}})$ where $\bar{d} = \text{argmax}_d\, D(o, d)$. Strategy **PB** instead is defined as follows. For each target d, we consider the neighborhood consisting of all vertices v whose distance from d is not larger than half the distance from o to d, namely $\mathcal{N}_d = \{v \in \mathcal{V} \mid D(v, d) \leq D(o, d)/2\}$. The defender's strategy is to make a decision the first time the attacker enters one of such neighborhoods by choosing the corresponding target. If such neighborhoods intersect and the attacker simultaneously enters multiple of them, the selection among them is made uniformly at random. We cannot directly compare our technique against traditional GR methods, including PPGR ones, because the two frameworks are fundamentally different.

Figure 2 displays plots of the average defender's reward Φ against the number of targets for the two graphs and the four values of the saturation parameter t. Our algorithm FP is compared with the other two, BL and PB. Note that the BL plot does not indicate t as, by construction, it does not depend on this parameter. The comparison shows the significant suboptimality of the two strategies, BL and PB, with respect to our solution, particularly when the number of targets increases. As expected, the reward grows monotonically with t because, as we increase t, the protection success probability does not decrease. Typically, the reward decreases as the number of targets increases. Interestingly, for the Shanghai map, increasing the number of targets from 4 to 5 leads to a slight increase in the defender's reward for both the FP and PB strategies. This is related to the topology of the Shanghai map and the obstacle positions.

Finally, Figure 3 shows the average runtime of our technique per problem. Even when keeping the number of walks per target constant, larger grids are computationally more demanding. This is due to the need to calculate more and longer prefixes. In addition, the overall runtime generally increases linearly with the number of targets for the obstacle-free 30x30 grids and, to a lesser extent, for the Shanghai map.

7 CONCLUSION AND FUTURE WORK

We consider security scenarios in which a malicious attacker enters an environment to attack one of a set of vulnerable targets that a defender wishes to protect. We model the decision-making of the two players as a zero-sum strategic game. The computation of Nash equilibria is complex in this setting because of the size of the agents' action sets. We offer a combinatorial formulation of the players' best response sets and use the classical fictitious play learning scheme to achieve a Nash equilibrium asymptotically. We validate our methodology through a series of experiments, confirming that our approach outperforms other less dynamic strategies in terms of effectiveness.

We aim to analyze several directions for extending our model in the future. Although, in our approach, all targets are equally important to the defender, we could easily expand our techniques to accommodate heterogeneity. A more challenging extension is to allow the defender to simultaneously defend two or more targets, making the protection success probability depend on the set of chosen targets. Finally, it would be interesting to consider scenarios where the defender only has partial information on the attacker's movements.

REFERENCES

[1] Bo An, Milind Tambe, and Arunesh Sinha. 2017. Stackelberg security games (ssg) basics and application overview. *Improving Homeland Security Decisions* 2 (2017), 485.

[2] Samuel Ang, Hau Chan, Albert Xin Jiang, and William Yeoh. 2017. Game-theoretic goal recognition models with applications to security domains. In *Decision and Game Theory for Security: 8th International Conference, GameSec 2017, Vienna, Austria, October 23-25, 2017, Proceedings*. Springer, 256–272.

[3] Vicki M Bier and M Naceur Azaiez. 2008. *Game theoretic risk analysis of security threats*. Vol. 128. Springer Science & Business Media.

[4] Vicki M. Bier, Naraphorn Haphuriwat, Jaime Menoyo, Rae Zimmerman, and Alison M. Culpen. 2008. Optimal Resource Allocation for Defense of Targets Based on Differing Measures of Attractiveness. *Risk Analysis* 28, 3 (2008), 763–770.

[5] Sviatoslav Braynov. 2006. Adversarial planning and plan recognition: Two sides of the same coin. In *Secure Knowledge Management Workshop*, Vol. 3. 67–70.

[6] Sandra Carberry. 2001. Techniques for plan recognition. *User modeling and user-adapted interaction* 11 (2001), 31–48.

[7] Yuan Shih Chow, Herbert Robbins, and David Siegmund. 1991. *The theory of optimal stopping*. Dover.

[8] Christopher W Geib and Robert P Goldman. 2009. A probabilistic plan recognition algorithm based on plan tree grammars. *Artificial Intelligence* 173, 11 (2009), 1101–1132.

[9] Kjell Hausken and Fei He. 2016. On the Effectiveness of Security Countermeasures for Critical Infrastructures. *Risk Analysis* 36, 4 (2016), 711–726.

[10] Jun Hong. 2001. Goal recognition through goal graph analysis. *Journal of Artificial Intelligence Research* 15 (2001), 1–30.

[11] Linan Huang and Quanyan Zhu. 2018. Dynamic Bayesian Games for Adversarial and Defensive Cyber Deception. *CoRR* abs/1809.02013 (2018). arXiv:1809.02013 http://arxiv.org/abs/1809.02013

[12] Kyle Hunt and Jun Zhuang. 2024. A review of attacker-defender games: Current state and paths forward. *European Journal of Operational Research* 313, 2 (2024), 401–417.

[13] Sushil Jajodia, Anup K Ghosh, VS Subrahmanian, Vipin Swarup, Cliff Wang, and X Sean Wang. 2012. *Moving Target Defense II: Application of Game Theory and Adversarial Modeling*. Vol. 100. Springer Science & Business Media.

[14] Dmytro Korzhyk, Zhengyu Yin, Christopher Kiekintveld, Vincent Conitzer, and Milind Tambe. 2011. Stackelberg vs. Nash in security games: An extended investigation of interchangeability, equivalence, and uniqueness. *Journal of Artificial Intelligence Research* 41 (2011), 297–327.

[15] Seung Y. Lee, Bradford W. Mott, and James C. Lester. 2012. Real-Time Narrative-Centered Tutorial Planning for Story-Based Learning. In *Intelligent Tutoring Systems*, Stefano A. Cerri, William J. Clancey, Giorgos Papadourakis, and Kitty Panourgia (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 476–481.

[16] V Lisy, R Pibil, J Stiborek, B Bosansky, and M Pechoucek. 2012. Game-theoretic approach to adversarial plan recognition. In *Proceedings of the 20th European Conference on Artificial Intelligence*. 546–551.

[17] Peta Masters and Sebastian Sardina. 2017. Cost-based goal recognition for path-planning. In *Proceedings of the 16th conference on autonomous agents and multiagent systems*. 750–758.

[18] Peta Masters and Sebastian Sardina. 2017. Deceptive Path-Planning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 4368–4375. https://doi.org/10.24963/ijcai.2017/610

[19] Peta Masters and Sebastian Sardina. 2019. Cost-based goal recognition in navigational domains. *Journal of Artificial Intelligence Research* 64 (2019), 197–242.

[20] Peta Masters and Sebastian Sardina. 2021. Expecting the unexpected: Goal recognition for rational and irrational agents. *Artificial Intelligence* 297 (2021), 103490.

[21] Felipe Rech Meneguzzi and Ramon Fraga Pereira. 2021. A survey on goal recognition as planning. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI), 2021, Canada*. 4524–4532.

[22] Wookhee Min, Eun Ha, Jonathan Rowe, Bradford Mott, and James Lester. 2014. Deep learning-based goal recognition in open-ended digital games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Vol. 10. 37–43.

[23] Wookhee Min, Bradford W Mott, Jonathan P Rowe, Barry Liu, and James C Lester. 2016. Player Goal Recognition in Open-World Digital Games with Long Short-Term Memory Networks.. In *IJCAI*. 2590–2596.

[24] Bradford Mott, Sunyoung Lee, and James Lester. 2006. Probabilistic goal recognition in interactive narrative environments. In *Proceedings of the National Conference on Artificial Intelligence*, Vol. 21. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 187.

[25] John Nash. 1951. Non-Cooperative Games. *Annals of Mathematics* 54, 2 (1951), 286–295.

[26] Katerina Papadaki, Steve Alpern, Thomas Lidbetter, and Alec Morton. 2016. Patrolling a Border. *Operations Research* 64, 6 (2016), 1256–1269. https://doi.org/10.1287/opre.2016.1511

[27] Ramon Pereira, Nir Oren, and Felipe Meneguzzi. 2017. Landmark-based heuristics for goal recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31. 3622–3628.

[28] Robert Powell. 2007. Defending against terrorist attacks with limited resources. *American Political Science Review* 101, 3 (2007), 527–541.

[29] Robert Powell. 2009. Sequential, nonzero-sum "Blotto": Allocating defensive resources prior to attack. *Games and Economic Behavior* 67, 2 (2009), 611–615.

[30] TES Raghavan. 1994. Zero-sum two-person games. *Handbook of game theory with economic applications* 2 (1994), 735–768.

[31] Miquel Ramırez and Hector Geffner. 2009. Plan recognition as planning. In *Proceedings of the 21st international joint conference on Artificial intelligence. Morgan Kaufmann Publishers Inc.* Citeseer, 1778–1783.

[32] Miguel Ramírez and Hector Geffner. 2010. Probabilistic plan recognition using off-the-shelf classical planners. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 24. 1121–1126.

[33] Julia Robinson. 1951. An iterative method of solving a game. *Annals of mathematics* (1951), 296–301.

[34] Violetta Rostobaya, Yue Guan, James Berneburg, Michael R. Dorothy, and Daigo Shishika. 2023. Deception by Motion: The Eater and the Mover Game. *IEEE Control. Syst. Lett.* 7 (2023), 3157–3162. https://doi.org/10.1109/LCSYS.2023.3291385

[35] Patrice C Roy, Abdenour Bouzouane, Sylvain Giroux, and Bruno Bouchard. 2011. Possibilistic activity recognition in smart homes for cognitively impaired people. *Applied Artificial Intelligence* 25, 10 (2011), 883–926.

[36] Daisaku Sakaguchi. 2011. Distance and military operations: Theoretical background toward strengthening the defense of offshore islands. *NIDS journal of defense and security* 12 (2011), 83–105.

[37] Charles F. Schmidt, Natesa S. Sridharan, and John L. Goodson. 1978. The Plan Recognition Problem: An Intersection of Psychology and Artificial Intelligence. *Artificial Intelligence* 11 (1978), 45–83.

[38] Eric Shieh, Albert Xin Jiang, Amulya Yadav, Pradeep Varakantham, and Milind Tambe. 2015. An extended study on addressing defender teamwork while accounting for uncertainty in attacker defender games using iterative dec-mdps. *Multiagent and Grid Systems* 11, 4 (2015), 189–226.

[39] Nathan R. Sturtevant. 2012. Benchmarks for Grid-Based Pathfinding. *IEEE Trans. Comput. Intell. AI Games* 4, 2 (2012), 144–148.

[40] Gita Sukthankar, Christopher Geib, Hung Hai Bui, David Pynadath, and Robert P Goldman. 2014. *Plan, activity, and intent recognition: Theory and practice*. Newnes.

[41] Milind Tambe. 2011. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge University Press.

[42] John Von Neumann and Oskar Morgenstern. 1947. *Theory of games and economic behavior, 2nd rev.* Princeton University Press.

[43] Jin Y Yen. 1971. Finding the k shortest loopless paths in a network. *management Science* 17, 11 (1971), 712–716.

[44] Tan Zhi-Xuan, Jordyn L. Mann, Tom Silver, Joshua B. Tenenbaum, and Vikash K. Mansinghka. 2020. Online Bayesian goal inference for boundedly-rational planning agents. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Vancouver, BC, Canada) *(NIPS '20)*. Curran Associates Inc., Red Hook, NY, USA, Article 1614, 13 pages.