Uncertainties in Adversarial Patrol

Noa Agmon, Sarit Kraus and Gal A. Kaminka*
Department of Computer Science
Bar Ilan University, Israel
{segaln, sarit, galk}@cs.biu.ac.il

Abstract

In this paper, we study the problem of multi-robot perimeter patrol in adversarial environments, under uncertainty. In this problem, the robots patrol around a closed area, where their goal is to patrol in a way that maximizes their chances of detecting an adversary trying to penetrate into the area. Uncertainties may rise in different aspects in this domain, and herein our focus is twofold. First, uncertainty in the robots' sensing capabilities, and second uncertainty of the adversary's knowledge of the patrol's weak points. Specifically, in the first part of the paper, we consider the case in which the robots have realistic, and thus imperfect, sensors. These cannot always detect the adversary, and their detection capability changes with their range. In the second part of the paper, we deal with different possible choices of penetration spots by the adversary, and find an optimal solution for the patrolling robots in each such case.

1 Introduction

The problem of multi-robot patrol has gained growing interest in the past years (e.g. [7; 4; 2]). In this problem, robots are required to repeatedly visit a target area while monitoring it in order to detect some change in the area's state. Most researches have concentrated on assuring some point-visit frequency criteria by the patrol algorithm. Agmon et al. presented a new approach for multi-robot patrol in [2], where they addressed the problem of multi-robot patrol in adversarial environments, in which the robots' goal is to patrol in a way that maximizes their chances of detecting an adversary trying to penetrate through the patrol path. They have shown that this problem is inherently different from the frequency driven patrol problem, and discussed optimality of patrol algorithms in different adversarial environments.

Generally, when dealing with a system of robots, it is necessary to consider deviation from the expected behavior of the robots in order to adapt the system to real world constraints settings (e.g. [11; 12]). One of the aspects of interest is uncertainty in the robots' sensing [11]. In reality, it is rarely the case that robots sense successfully and accurately everything they are supposed to detect, and in this case it is important to address the probability that their sensing will fail. A second aspect is uncertainty in the capability and knowledge of the adversary (depending, for example, on its confidence in the information it attained on the patrolling robots). In this paper we address these two different aspects, model the behavior of the system in each case and describe polynomial-time optimal theoretical solutions to the patrol algorithm of the robots that correspond to each scenario.

A first attempt to deal with perception uncertainties in adversarial patrol was given in [1], which altered the model of multi-robot *open fence* (polyline) patrol to deal with cases in which the robots will detect penetration in their current location only with some probability $p_d \leq 1$. In this paper we expand this result in several manners. First, we describe a solution for this case in *perimeter* (closed polyline) patrol. Second, we consider the case in which the robots can sense beyond their current physical location. We find an optimal patrol algorithm for cases where the sensorial capabilities changes as a function of distance from the robot.

Previous work in adversarial patrol has shown that the optimality of patrol algorithm depends on the adversarial model, specifically the knowledge obtained by the adversary on the patrolling robots. Theoretical optimality results were proven for a zero-knowledge adversary, in which the adversary choose as its penetration spot at random with uniform distribution, and for a full knowledge adversary, that is assumed to choose the weakest point of the patrol as its penetration spot. In the latter case the penetration spot is well defined. However, since the calculation of probability of penetration detection throughout the perimeter is not trivial, it is likely that the adversary will choose to penetrate through one of the weakest spots, and not through the exact optimal spot. On the other hand, the adversary might choose to penetrate through some physical proximity to the weakest spot. In this paper we analyze both cases and provide optimal patrol algorithms that deal with both options of uncertainty in penetration spots.

2 Background

Systems of multiple robots, working together in order to patrol in some target area, have been studied in various contexts, concentrating on optimizing frequency criteria (e.g. [7]), or

*This research was supported in part by ISF grant #$1357/07$ and #$1685/07$.

on patrol in adversarial environments [2; 3; 4], which is the focus of this paper.

Agmon et al. [2; 3] introduced the multi-robot *adversarial* perimeter patrol along with the robotic model we base our work upon. In their work, they consider three adversarial models, which differ in the amount of knowledge the adversary obtained on the patrol algorithm: full knowledge, zero knowledge and patrial knowledge. They provide optimal patrol algorithms for the first two cases, and consider a heuristic algorithm for the last. In this paper we continue the research on the patrial-knowledge adversarial model, and provide a theoretical discussion and approaches towards determining an optimal patrol algorithm in this scenario.

Agmon et al. suggest a solution to the adversarial patrol along an *open fence* in [1], where they refer also to the imperfect sensorial capabilities of the robots. However, their solution is limited only for open fences, and to imperfect sensing in one segment the robot currently resides on. Here we extend their results also to perimeter patrol and to handle also extended sensing range with possible different sensing capabilities as the distance from the robot grows.

Elmaliach et al. [7] studied the problem of frequency-based fence patrol, where they concentrated on optimizing point-visit frequency criteria in realistic multi-robot systems. They considered in their analysis of the system possible uncertainties in the movement and velocity of the robots. In our work we also model uncertainties in the system, however our main goal is different - to optimize probability of detecting penetrations, rather than optimize frequency criteria, hence we also concentrate on sensing uncertainties.

Other closely related work is the work by Paruchuri et al. [9; 8], which consider the problem of placing security checkpoints in adversarial environments. They use policy randomization for the agents behavior in order to maximize their rewards. In their work, the adversary has full knowledge of the agents' behavior, therefore it can use it in order to minimize its probability of being caught in some checkpoint. They again do not consider sensorial scenarios which depend on different sensorial models of the robots. Pita et al. [10] continue this research to consider the case in which the adversary makes their choice based on their bounded rationality or uncertainty, rather than make the optimal game-theoretic choice. They consider three different types of uncertainty over the adversary's choices, and provide new algorithms that deal with these types of uncertainties. In our work we discuss both uncertainty in sensing and uncertainty in adversary's choice, and provide optimal polynomial-time solutions for both cases.

Amigoni et. al. [4] also used a game-theoretic approach for determining the optimal strategy for patrolling agents, using leader-follower games. They consider an environment in which a robot can move between any two nodes in a graph, as opposed to the perimeter model we focus on. Their solution is suitable for one robot, and since the computation of the optimal strategy is exponential, they described a heuristic approach for finding a solution.

3 Robot and environment model

We are given a team of k homogenous robots, required to patrol around a closed area (perimeter). The perimeter is divided into N segments, where the travel time of each robot through a segment is uniform, i.e., all robots travel through one segment per time cycle . Hence the segments' length is uniform in time, but not necessarily in distance.

The robots have directionality associated with their movement, i.e, if they go backwards they have to physically turn around. We model the cost of turning around in time, and denote time it takes the robots to turn around and stabilize in their new direction by τ.

The system of perimeter patrol is linear, meaning that at each time step the robots have one of two options: go straight or turn around. Therefore the robot's patrol algorithm is characterized by a probability p, i.e., at each time step go straight forward with probability p, or turn around with probability $q = 1 - p$.

Motivated by the optimality proofs in [3; 2], we assume the robots are coordinated in the sense that if they decide to turn around they do it simultaneously. Moreover, we assume the robots are placed uniformly along the perimeter, with distance of $d = N/k$ segments between every two consecutive robots along the path. The optimality proof of these requirements (synchronization and uniform distance) is based on the fact that the probability of penetration detection decreases as the distance from the robot increases (see Lemma 3 in [2]). Therefore it is best to minimize the maximal distance between every two consecutive robots, and it is done by guaranteeing that the distance in time between every two consecutive robots is equal, and this is maintained by the requirement that the robots are coordinated.

In the adversarial models considered here, the adversary decides at time 0 (after possibly an observation period in which it studies the patrol) through which segment to penetrate, and the time it takes it to penetrate is not instantaneous, and lasts t time units.

The chances of the robots to detect an adversary passing through a segment s_i is defined as the *probability of penetration detection* at the segment, and denoted by ppd_i.

Finding the optimal patrol algorithm is divided into two stages. In the first stage, the ppd is calculated for all segments, where ppd_i is a function of p. In the second stage, these functions are manipulated in a way that will maximize a property suitable for the current adversarial model. For example, for an adversary choosing at random its penetration spot with uniform distribution, in the second stage the goal will be to maximize the expected ppd throughout the perimeter, using as input the ppd_i functions.

4 Uncertainty in sensing

In this section we deal with uncertainty in the robots' perception. Specifically, we consider the case in which the robot could have imperfect sensorial capabilities. We introduce three models of the robots' sensing abilities. In the first model, LRange, the robots have sensing abilities that exceed the segment they currently reside on. In the second model, ImpDetect, we present a solution to the case in which the robots' sensorial abilities might not be perfect, i.e., a robot might not detect an adversary even if it is under its sensorial range. This is an extension of the results presented in [1] to include also perimeter patrol. This leads us to the last and general model, ImpDetLRange, that combines both models and deals with the case that the robots' sensing abilities exceed their current location, yet the reliability of the detection is not perfect. In this case we assume that the ability to detect penetrations might change along the detection range, specifically it might decrease as the distance grows.

4.1 Extending sensing range

In this section we consider the LRange model, in which the sensorial range of a robot exceeds the segment it currently resides in. Denote the number of segments sensed by the robot beyond the segment currently occupied by it by L (see Figure 1). If $L > 0$, we refer to these L segments by *shaded segments*. Note that the location of the shaded segments depends on the direction of the robot shading them, and are always in the direction the robot is facing.

A trivial solution to dealing with such a situation is to enlarge the size of the segment, hence enlarge the length of the time unit used as base for the system, such that it will enforce L to be 0. However, in this case we lose accuracy of the analysis of the system, as the length of the time cycle should be as small as possible to suit also the velocity of the robots and the value of t.

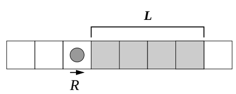

Figure 1: An illustration of the L segments shaded by robot R. Here, R is facing right, therefore the shaded segments are to its right.

In [3], the values of t that can be handled by the system are bounded by its relation to d (the distance between every two robots along the path). In case $L > 0$, this changes. Specifically, if $L = 0$, then the possible values of t considered are $\lceil d/2 \rceil + \tau \leq t \leq d - 1$ [3]. However, if $L > 0$, then it is possible to handle even smaller values of t, i.e., even if the penetration time of the adversary is short. Formally, the possible values of t are given in the following equation.

$$\lceil d/2 \rceil + \tau - L \leq t \leq d - L$$

If t is smaller than $\lceil d/2 \rceil + \tau - L$, then an adversary with full knowledge will manage to penetrate with probability 1, i.e., there is a segment unreachable within t time units. On the other hand, if t is greater than $d - L$, then a simple deterministic patrol algorithm will detect all penetrations with probability 1. We assume that during the τ time units the robot turns around, it can sense only its current segment.

Algorithm for finding ppd_i with shaded segments:

For each segment s_i, ppd_i is determined by the probability that some robot will visit this segment *plus* the probability that this segment is shaded by some robot. We use a dynamic-programming inspired rule, similar to the one described in [2], yet we expand it to include also the probability of being shaded by some robot. The main idea is that in each transition phase, the algorithm checks whether the state shades on an absorbing state, i.e., if the robot in its current location and direction shades the given segment (distance from it is smaller than L). See Algorithm 1 for a full description of the algorithm for calculating ppd_{loc}, $1 \leq loc \leq d$. The time complexity of the algorithm is $\mathcal{O}(dt)$, which is the time it takes to fill in the entire table.

Algorithm 1 FindPPDwShade(d, t, loc, L)

```
Create matrix M of size (2d+2) × (t+1), initialized with 0s.
Set M[0, loc^{cw}] ← 1.
Fill all entries in M gradually using the following rules.
for r ← 1 to t do
    for each entry M[r, s_i^{cw}] do
        Set v ← p · M[r−1, s_{i+1 mod d}^{cw}] + q · M[r−1, s_i^{cc}]
        Set M[r, s_i^{cw}] ← v
        if i + L ≥ d then
            Set M[r, s_{abs}] ← v {absorbing state}
    for each entry M[r, s_i^{cc}] do
        Set v ← p · M[r−1, s_{i+1 mod d}^{cw}] + q · M[r−1, s_i^{cc}]
        Set M[r, s_i^{cc}] ← v
        if i − L ≤ 0 then
            Set M[r, s_{abs}] ← v {absorbing state}
    for absorbing state M[r, s_{abs}] do
        Set M[r, s_{abs}] ← M[r−1, s_{abs}] + p · {M[r−1, s_1^{cw}] +
        M[r−1, s_d^{cc}]}
Return M[t, s_{abs}]
```

4.2 Imperfect detection

Uncertainty in the perception of the robots should be taken into consideration in practical multi-robot problems. Therefore we consider the realistic case in which the robots have imperfect sensorial capabilities, i.e., even if the adversary passes through the sensorial range of the robot, it still does not necessarily detect it.

We introduce the ImpDetect model, in which a robot travels through one segment per time cycle while monitoring it (i.e., $L = 0$), and has imperfect sensing. Denote the probability that an adversary penetrating through a segment s_i while it is monitored by some robot R and R will actually detect it by $p_d \leq 1$.

Note that in case $p_d < 1$, revisiting a segment by a robot could be worthwhile - it could increase the probability of detecting the adversary. Therefore the probability of detection in a segment s_i (ppd_i) is *not* equivalent to the probability of *first* arriving at s_i (as seen in [2]), but the probability of detecting the adversary during *some* visit y to s_i, $0 \leq y \leq t$. Denote the probability of the y'th visit of some robot to segment s_i by w_i^y. Therefore ppd_i is defined as follows.

$$\mathsf{ppd}_i = w_i^1 p_d + w_i^1(1-p_d) \times \{w_i^2 p_d + w_i^2(1-p_d) \times \{\ldots\{w_i^t \times p_d\}\}\} \quad (1)$$

In words, the probability of detecting the penetration is the probability that it was detected in the first visit ($w_i^1 \times p_d$) plus the probability that it was *not* detected then, but during later stages.

Note that after t time units, $w_i^t = 0$ for all currently unoccupied segments s_i, and if a robot resides in s_i, then w_i^t is exactly $(1 - p_d)^t$.

One of the building blocks upon which the optimal patrol algorithms are based, is the assumption that the probability of detection decreases or remains the same as the distance from a robot increases, i.e., it is a monotonic decreasing function. This fact was used in [2] in proving that in order to maintain optimal ppd, then the robots should be placed uniformly around the perimeter (with uniform time distance), and maintain this distance by being coordinated. We omit the detailed proof due to space constraints. The rational of the proof is that the probability of detection decreases as the distance from the location of the robot increases. Consequently, both minimal ppd and average ppd are maximized if the distance between the robots is as small as possible. Since the patrol path is cyclic, this is achieved only if the distance between every two consecutive robots is uniform, and remains uniform.

Theorem 1. *For both the full knowledge and zero knowledge adversarial models, a patrol algorithm in the* ImpDetect *model is optimal only if it satisfies these two conditions. a. The robots are placed uniformly around the perimeter. b. The robots are coordinated in the sense that if they turn around, they do it simultaneously. By assuring these two conditions, the robots preserve the uniform distance between them along the execution.*

Algorithm for finding ppd_i with imperfect sensorial detection:

We now describe Algorithm FindPPDwImpDetect that finds the probability of penetration detection in each segment (ppd_i). The algorithm computes the probability of all visits to a segment during t time units. The algorithm, similar to algorithm FindFunc [2], is dynamic programming inspired. As stated previously, the main difference between the algorithms is that FindFunc considers only the first visit to a segment, where FindPPDwImpDetect considers *all* visits to a segments and the probability of sensorial detection. Figure 2 describes a representation of transition between segments as a Markov chain. This is later translated into constructing gradually a table using a dynamic programming-inspired rules, as described in Algorithm FindPPDwImpDetect. The time complexity of the algorithm is $\mathcal{O}(dt)$, which is the time it takes to construct table M. Extracting the polynomial coefficient is done in time $\mathcal{O}(1)$.

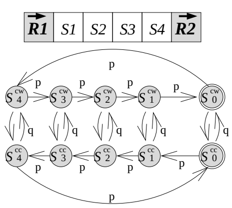

Figure 2: Representation of the system as a Markov chain along with state transition. The robots are initially placed at the external segments, heading right. State s_0 represents the segment currently occupied by a robot.

Theorem 2. *Algorithm* FindPPDwImpDetect(d, t, i) *computes* ppd_i.

4.3 Extending sensorial range along with imperfect detection

In many cases, the actual sensorial capabilities of the robot is composed of the two characteristics described in previous sections, i.e., the robot can sense beyond its current segment, however the sensing ability is imperfect. Therefore in this section we introduce the ImpDetLRange sensorial model, which is a combination of the LRange and the ImpDetect models. Here the robot can sense L segments beyond its current segment, yet the p_d in each segment varies and is not necessarily 1. We therefore describe an algorithm that deals

Algorithm 2 FindPPDwImpDetect(d, t, loc)

Create matrix M of size $(2d+2) \times (t+1)$, initialized with 0s.
Set $M[0, loc^{cw}] \leftarrow 1$.
Fill all entries in M gradually using the following rules.
for $r \leftarrow 1$ to t **do**
  for $i \leftarrow 1$ to d (all other states) **do**
    For each entry $M[r, s_i^{cw}]$ set value to $p \cdot M[r-1, s_{(i+1 \mod d)}^{cw}] + q \cdot M[r-1, s_i^{cc}]$.
    For each entry $M[r, s_i^{cc}]$ set value to $p \cdot M[r-1, s_{(i-1 \mod d)}^{cc}] + q \cdot M[r-1, s_i^{cw}]$.
  for s_0^{cw} and s_0^{cc} **do**
    Set $M[r, s_0^{cw}] \leftarrow f \times \{p \cdot M[r-1, s_1^{cw}] + q \cdot M[r-1, s_0^{cc}]\}$
    Set $M[r, s_0^{cc}] \leftarrow f \times \{p \cdot M[r-1, s_d^{cc}] + q \cdot M[r-1, s_0^{cw}]\}$
$w_{loc}^i \leftarrow$ polynomial coefficients of f^i from sum of $M[r, s_0^{cw}] + M[r, s_0^{cc}]$, for all $0 \leq r \leq t$, $1 \leq i \leq t$.
Return the result obtained by substituting the w_{loc}^i values in Equation 1.

with the most realistic form of sensorial capabilities [6]: imperfect, long range sensing.

The information regarding the sensorial capabilities of the robots includes two parameters. The first describes the quantity of the sensing ability, i.e., the number of segments that exceeds the current segment the robot resides in, in which the robot has *some* sensing abilities, denoted by L. The second parameter describes the quality of sensing in all segments the robot can sense. This is given in the form of a vector $V_S = \{v_0, v_1, \ldots, v_L\}$, where v_i is the probability that the robot residing in s_0 will detect a penetration that occurs in segment s_i. We assume that the values in V_S is monotonically non-increasing, i.e., as i increases, v_i decreases or remains the same.

In the ImpDetLRange model, the probability of penetration detection is more complex, and has to take into consideration also the possibility of being in the sensorial range of some robot and the probability of being detected there. Denote the probability that s_i is in distance $j <= L$ from some robot, i.e., within its sensorial range, for the j'th time by $w_i^j(e)$. Denote the probability that the adversary in s_i will not be detected at all by $\overline{\mathsf{ppd}_i}$. The probability that the adversary will be detected is actually the complementary of the probability that it will not be detected. Therefore ppd_i is defined as follows.

$$\mathsf{ppd}_i = 1 - \overline{\mathsf{ppd}_i} = 1 - \prod_{j=1}^{t} \prod_{e=1}^{L} \{w_i^j(e) \cdot (1 - v_e)\} \qquad (2)$$

In words, the probability of penetration detection is the complementary of the probability that the adversary is *not detected at all* during the t time units. This is the probability that it is not detected at any possible occurrence in any possible range (corresponding to a probability of detection) during those time unit. The overall number of components is, therefore, $L \times t$.

Algorithm for finding ppd_i with extended-range imperfect detection:

The algorithm for finding ppd_i in case we allow extended range ($L > 0$) and imperfect detection with changing probabilities of detection as a function of the distance from the robot is composed by two stages. In the first stage, we need to find the probability of being shaded with distance $1 \leq e \leq L$ from the robot for the j'th time, $1 \leq j \leq t$. This provides us with all values $w_i^j(e)$. We then substitute all the acquired values in Equation 2. The full description of the algorithm is presented in Algorithm FindComplexP below. Note that in Algorithm FindPPDwImpDetect we had one object f used for identifying the number of the visit to the segment. Here, since we have to consider all visits of all possible distances that are less or equal to L (shaded segments), we use $L+1$ objects, $f_0, \dots, f_L$. The time complexity of the algorithm is $\mathcal{O}(dt + Lt)$ - the time to construct the table M plus the time to extract all polynomial coefficients (respectively). Since $L < d$, this is again $\mathcal{O}(dt)$.

Algorithm 3 FindComplexP$(d, t, loc, L, V_S = \{v_0, \dots v_L\})$

Create matrix M of size $(2d+2) \times (t+1)$, initialized with 0s.
Set $M[0, loc^{cw}] \leftarrow 1$.
Set $Res \leftarrow 0$
Fill all entries in M gradually using the following rules.
for $r \leftarrow 1$ to t **do**
    for each entry $M[r, s_i^{cw}]$ **do**
        Set $u \leftarrow p \cdot M[r-1, s_{i+1 \mod d}^{cw}] + q \cdot M[r-1, s_i^{cc}]$
        if $i + L \geq d$ **then**
            $u \leftarrow u \times f_{d-i}$
            $Res \leftarrow Res + u$
        Set $M[r, s_i^{cw}] \leftarrow u$
    for each entry $M[r, s_i^{cc}]$ **do**
        Set $u \leftarrow p \cdot M[r-1, s_{i+1 \mod d}^{cw}] + q \cdot M[r-1, s_i^{cc}]$.
        if $i - L \leq 0$ **then**
            $u \leftarrow u \times f_i$
            $Res \leftarrow Res + u$
        Set $M[r, s_i^{cc}] \leftarrow u$
$w_i^j(e) \leftarrow$ polynomial coefficient of f_e^j of Res, for all $1 \leq j \leq t$, $0 \leq e \leq L$ (while substituting all other $f_{e'}^j$, $e' \neq e$ in the equation).
Return the result obtained by substituting the $w_i^j(e)$ values in Equation 2.

4.4 Applying the sensorial models in different adversarial models

Until now, we have presented three different algorithms for finding the ppd_i for each segment s_i in three different sensorial models - ImpDetect, LRange and ImpDetLRange. This information can be used by the team of robots in order to define their patrol algorithm, based on the adversarial environment they operate in. We describe here how the different sensorial models influence the patrol algorithm in two such environments. The first is the *full knowledge* adversarial environment [2], in which the adversary holds all the information concerning the patrol algorithm of the robots, hence chooses to penetrate through the segment with minimal ppd. In the second adversarial model, we assume the adversary has no knowledge of the patrol algorithm ([3]), hence it decides to penetrate through a currently vacant segment at random with uniform distribution.

If the adversary has full knowledge of the robots' patrol algorithm, we can use the ppd generated by the three algorithms described here — FindPPDwShade, FindPPDwImpDetect and FindComplexP, as input to the MaxiMin algorithm described in [2]. This algorithm finds the probability p in which the minimal ppd_i is maximized, and does it by identifying the maximal point in all ppd_i integral intersections.

A more interesting case is the zero-knowledge adversarial model, in which the adversary has no knowledge of the patrol algorithm, hence chooses at random with uniform distribution through what segment to penetrate. In [3], it was proven that the optimal patrol algorithm in this case *if the robots have perfect sensors* is the simple deterministic algorithm ($p = 1$). The rationale behind the optimality proof of the deterministic algorithm lies in the fact that it is not worthwhile to go back and revisit segments.

However, in case the probability of detecting the penetrator is imperfect, i.e., $p_d < 1$, this argument does not necessarily hold, i.e., revisiting a segment does have added value. In the following, we show the surprising result that even if $p_d < 1$, if the adversary chooses its penetration spot at random, it is still best to patrol deterministically around the perimeter. Moreover, we strengthen our result by showing that even if the robot makes a post analysis of its decision to go straight or turn around, it will also decide to keep on going straight.

Theorem 3. *In the* ImpDetect *model, the deterministic algorithm maximizes the expected* ppd *throughout the perimeter for all $p_d \leq 1$ in case the adversary chooses its penetration spot at random with uniform distribution.*

We omit the proof due to space constraints. The idea behind the proof lies in the fact that the gain (in probability of penetration detection) from visiting a new segment exceeds the gain from revisiting a segment, since by revisiting a segment we add also the probability of re-reaching it.

We strengthen this result by showing that it is beneficial for the robot to keep visiting new segments in case the adversary chooses its penetration spot randomly with uniform distribution (with probability $1/d$) even if the robot calculates its benefit post factum, i.e., after visiting a segment. Denote the probability that the adversary penetrated through a segment s_i by PN_i, and the probability that the robot visited s_i without detecting it by ND_i. Therefore, by conditional probability law, if $ND_i > 0$, $P(PN_i \mid ND_i) =$

$$\frac{PN_i \bigcap ND_i}{ND_i} = \frac{1/d(1-p_d)}{(d-1)/d + 1/d(1-p_d)} = \frac{1-p_d}{d-p_d}$$

On the other hand, the probability that the adversary chooses to penetrate through s_{i+1} given that the robot did not detect it in segment s_i is

$$\frac{1 - \frac{1-p_d}{d-p_d}}{d-1} = \frac{1}{d-p_d} > \frac{1-p_d}{d-p_d}$$

In other words, the probability of revealing new information in visiting a new segment is greater than the probability of revealing new information from revisiting a segment that was already visited at least once, even after knowing that the adversary was not caught in the revisited segment. The intuition is that by visiting a new segment, the probability of penetration detection grows by p_d, where if the robots revisits a segment, it carries along with it the probability of arriving there again, multiplied by p_d. Since the probability of arriving again is smaller than 1, the gain from revisiting a segment is smaller.

Theorem 4. *In the* LRange *model, the deterministic algorithm guarantees maximal expected* ppd *for random-uniform adversary if* $L \leq \tau$.

The proof of the theorem resembles the proof of Theorem 3 in [3], and we omit it due to lack of space.

5 Uncertainty in the adversary's perspective

In this section we turn to examine uncertainties in the adversary's point of view. Specifically, we try to bound the level of uncertainty the adversary has on the patrolling robots, and specifically on its optimal choice of penetration spot. Quantifying the uncertainty of the adversary is important in order to find optimal patrol algorithms that are suitable to the level of uncertainty of the adversary. In other words: Given a bounded region of the adversary's uncertainty, what is the patrol algorithm that maximizes the probability of penetration detection?

We suggest two general approaches for bounding the uncertainty level. In the first approach, we examine the case in which the adversary knows the probability p characterizing the patrol algorithm with some uncertainty. Unfortunately, we show that it is impossible to find an optimal patrol algorithm in this case.

We therefore suggest an alternative approach, in which the uncertainty is reflected by the choice of penetration spot. In this case, we do not necessarily assume that the adversary calculates the probability p, but tries to estimate the weakest spot using two estimation methods - physical proximity, or closeness to the minimal ppd.

5.1 Uncertainty of the adversary's knowledge of the patrol - negative result

In this section we bring an attempt to deal with the uncertainty of the adversary in the choice of the weakest spot of the algorithm. In this case, we try to quantify the information by the number of time cycles the adversary had to observe the system before it attempts to penetrate. The result of the knowledge obtained by the adversary is its assessment of the probability p characterizing the robots' patrol algorithm.

The problem of deducing the probability p can be considered as observing a Bernoulli trial, where a success is an event of going straight with probability p, and loss is turning around with probability $1-p$. We can use the Central Limit Theorem [5] that gives us bounds to the expected error from the real value of p after viewing it for t_v trials. Assuming the average of successes after viewing t_v trials is $\overline{p}$, its value is inside the boundaries $[p-\delta, p+\delta]$ with probability $p_{confidence}$, where δ is a function of t_v and depends on $p_{confidence}$. Therefore this bounds the uncertainty of the adversary on the real value of p to an interval around p, and we will try to use this interval in order to optimize the patrol algorithm of the robots.

A common way to handling uncertainties of systems is to assume that when having no knowledge, a random choice, with uniform probability, is made. In this domain, this approach was proven to be useful in an empirical evaluation in [3], where a patrol algorithm proven to be optimal for a random adversary performed substantially better than other algorithms for humans playing the role of an adversary that had no knowledge of the patrolling robots.

Therefore we tried to use a similar approach here, as we considered the following problem.

P-Interval **problem definition:** Let p be the probability characterizing the perimeter patrol algorithm of a team of robots. Assume the adversary have a bounded interval of uncertainty, i.e., the adversary knows that the real value of p is inside the interval $[p-\delta, p+\delta]$. Therefore it chooses its believed p_b at random with uniform probability inside this interval. Find the probability p characterizing the patrol of the robots such that it maximizes the expected ppd throughout the perimeter.

Unfortunately, we prove that this problem is unsolvable unless $\delta = 0$. We prove it by showing that the expected ppd function inside the interval $[0, 1]$ is monotonically increasing, i.e., as p grows the expected ppd grows, hence the optimal p does not converge unless $\delta = 0$.

Denote the number of times a robot switched directions during t time units by r, $r \geq 1$, and the ppd in segment s_j by R_0 after switching its direction r times by $\mathsf{ppd}_j^0(r)$.

Lemma 5. *Consider a sequence of* $2d$ *segments with one robot* R_0 *in the mid segment at time* 0. *Then* $\sum_{j=1}^{2d} \mathsf{ppd}_j^0(r) < \sum_{j=1}^{2d} \mathsf{ppd}_j^0(r-1)$ *for every* $\tau \geq 1$.

Proof. We first prove the lemma for $\tau = 1$. We divide the sequence of $2d$ segments into two: the sequence to the right of R_0 and to the left of R_0. For every j number of direction switches, let $\sum_{i=-t+j+1}^{-1} \mathsf{ppd}_i^0 = \delta(j)$, $\mathsf{ppd}_{-t+j}^0 = \delta'(j)$, $\sum_{i=1}^{t-j} \mathsf{ppd}_i^0 = \alpha(j)$ and $\mathsf{ppd}_{t-j}^0 = \alpha'(j)$.

The sum of ppd_i^0 for $r-1$ number of direction switches is $\delta(r-1)+\delta'(r-1)+\alpha(r-1)+\alpha'(r-1)$. For r switches, since the robots spend an extra time cycle for turning around, the two extreme segments with ppd > 0 are now unreachable, hence in this case $\delta'(r-1)$ and $\alpha'(r-1)$ no longer exist. Now, $\delta(r) + \delta'(r)$ is similar to changing the initial direction of the robot (by multiplying by $1-p$), and obtaining exactly $\alpha(r-1)$, hence $\delta(r) + \delta'(r) < (1-p)\alpha(r-1)$. Similarly, $\alpha(r)+\alpha'(r) < (1-p)\delta(r-1)$. Altogether, $\sum_{l=1}^{2d} \mathsf{ppd}_l^0(r) = \delta(r)+\delta'(r)+\alpha(r)+\alpha'(r) < (1-p)\alpha(r-1)+(1-p)\delta(r-1)$ and since $(1-p) < 1$ this is smaller than $\sum_{l=1}^{2d} \mathsf{ppd}_l^0(r-1)$. By the induction assumption, this is smaller than t.

The proof follows directly for $\tau > 1$, as the number of segments that become unreachable increases from 1 to τ for each direction switch, while the probability of penetration detection in other segments is the same. □

Denote the expected ppd for probability p (probability that the robots will continue straight in each time unit during the patrol), $0 \leq p \leq 1$ by $E_{\mathsf{ppd}}(p)$.

Lemma 6. *The expected* ppd, *as a function of* p, *is a monotonically decreasing function in the range* $[0, 1]$, *i.e., for all* $0 \leq p' < p \leq 1$, $E_{\mathsf{ppd}}(p') < E_{\mathsf{ppd}}(p)$

Proof. p represents the probability of going straight (and not switching directions) at each time unit. Denote the expected number of direction switches of robot R during t time units using probability p of going straight by $E_{switch}(p)$. Therefore $E_{switch}(p) = t(1-p)$ and $E_{switch}(p') = t(1-p')$, and since $p' < p$ it follows that $E_{switch}(p) < E_{switch}(p')$.

It remains to show that if a robot R is expected to switch its direction more times during t time units, then the expected ppd is smaller. Formally, for $0 \leq p' < p \leq 1$, $E_{switch}(p) < E_{switch}(p') \Rightarrow E_{\mathsf{ppd}}(p') < E_{\mathsf{ppd}}(p)$.

The expected ppd along the perimeter with r direction switches is $E^r_{\mathsf{ppd}} = 1/N \sum_{i=1}^{N} \sum_{j=1}^{k} \mathsf{ppd}_i^k(r)$. During $t < d$ time units the robot can influence only the ppd along at most $2d$, therefore the sum of each robot is not over all N segments, but only the neighboring d segments from each of its sides. Therefore, following Lemma 5, $E^r_{\mathsf{ppd}} < E^{r-1}_{\mathsf{ppd}}$. Therefore if $E_{switch}(p) < E_{switch}(p')$ then $E_{\mathsf{ppd}}(p') < E_{\mathsf{ppd}}(p)$. □

Theorem 7. P-Interval *is unsolvable unless* $\delta = 0$.

Proof. Assume, towards contradiction that $\delta > 0$, yet there exists $p*$ that maximizes the expected ppd throughout the perimeter. By the definition of P-Interval, the adversary deduces an interval around $p*$ in which it chooses its believed p at random inside the interval $[p*-\delta, p*+\delta]$. By Lemma 6, the expected ppd function is monotonically increasing, therefore the maximal expected ppd inside this interval is obtained in $p * +\delta$. This contradicts the assumption that $p*$ maximizes the expected ppd, unless $\delta = 0$. □

5.2 Uncertainty in the actual choice of penetration spot

When trying to bound the uncertainty of the adversary to its knowledge of the patrol, another option is to try and quantify the uncertainty in its choice of *penetration spot*. For several reasons, the adversary, even if knowing the patrol algorithm of the robots (specifically the probability p), might not choose to penetrate through the *exact* weakest spot. We present herein two possible deviations from the weakest spots, and hence two possible corresponding optimal ways of choosing the value of p in such cases.

The adversary, after studying the robots' patrol for a period of time, could result in several reasonable segments which the ppd values, as it calculates, are small enough. In this case it will choose at random, with a given probability distribution (for example uniform), the penetration spot between the v possible weakest segments. Hence the robots should choose p such that the expected ppd along the v segments with minimal ppd is maximal.

The second case is that the adversary might not choose to penetrate through the segment with the minimal ppd, but either through that segment, or through one of its neighboring segments at random. Hence in this case the robots should choose p such that the expected ppd along v neighboring with minimal ppd is maximized.

Note the difference between the two cases - in the first we are looking for the value $0 \leq p \leq 1$ such that the minimal v ppd's are maximized, and in the second case we are looking for p such that the weighted average of minimal possible v neighboring segments is maximized.

In both cases, the two extremities of uncertainties—full knowledge adversary (no uncertainty) and zero knowledge adversary (complete uncertainty)—match the results obtained by [2] and [3], respectively. If $v = 1$, i.e., there is no uncertainty in the choice of the weakest spot, then the algorithms are required to return exactly the value p such that the minimal ppd is maximized, similar to the MaxiMin algorithm presented in [2]. On the other hand, if $v = d$ and the probability distribution is uniform, then the algorithms will return the value p that maximized the expected ppd throughout the perimeter (=average ppd). As proven in [3], the optimal algorithm in this case is $p = 1$, i.e., the deterministic algorithm.

Note that the algorithms for finding optimal patrol uses the ppd_i functions. Hence this can be combined with any sensorial model presented in the previous section.

Algorithm 4 ComputeMinV$(v, V, \{\mathsf{ppd}_1, \ldots, \mathsf{ppd}_d\})$

Set $BufP \leftarrow \{0, 1\}$.
for every pair $\mathsf{ppd}_i, \mathsf{ppd}_j$, $1 \leq i, j \leq d$, $i \neq j$ **do**
  $Intersect_{i,j} \leftarrow$ all intersection points between ppd_i and ppd_j.
  $BufP \leftarrow BufP \bigcup Intersect_{i,j}$
Sort $BufP$ in ascending order
for $j \leftarrow 1$ to $|BufP|$ **do**
  Find v minimal function within $[BufP(j), BufP(j+1)]$, $f_{j_i}, f_{j_2}, \ldots, f_{j_v}$
  $f_{avg} \leftarrow \sum_{i=1}^{v} v_i \times f_{j_i}$
  $m \leftarrow f_{avg}(p*)$ such that $\forall p \in [BufP(j), BufP(j+1)], f^{(}_{avg}p*) \geq f_{avg}(p)$
  if $m > Res_f$ **then**
    $Res_f \leftarrow m$
    $Res_p \leftarrow p*$
Return Res_f

Maximizing expected ppd of v minimal segments

We first present Algorithm ComputeMinV that computes the value p such that the minimal v ppd's are maximized, given a probability distribution $V = \{v_i, v_2, \ldots, v_v\}$, where v_i is the probability that the adversary will choose to penetrate through the i'th weakest spot, $\sum_i = 1^v v_i = 1$. The algorithm works as follows. First, it identifies all intersection points between every pair of ppd_i, ppd_j functions ($1 \leq i, j \leq d$, $i \neq j$). Then it divides the range $[0, 1]$ to sections according to all intersection points. For each section $[p_a, p_b]$, the algorithm then identifies the minimal v curves between $[p_a, p_b]$, and finds the average curve, f_{avg} of these three curves between the points $[p_a, p_b]$. Since the adversary chooses to penetrate through one of the v segments with lowest ppd at random with the given distribution V, the *weighted average* (given weight v_i to the i'th minimal curve) of the v curves represent the *expected* ppd in that section. Last, ComputeMinV calculates the maximal value of $f_{avg}(a, b)$ in the section $[p_a, p_b]$, and reports the point p_{opt} that is maximal among all minimal points of the average functions. Figure 3 illustrates this algorithm.

The time complexity of Algorithm ComputeMinV is $\mathcal{O}(d^4 + d^3 \log d^3)$, compared to time complexity of d^3 of the original MaxiMin algorithm for full knowledge adversary.

Maximizing minimal ppd of v neighboring segments

As stated previously, the adversary might attempt to penetrate not only through the weakest segment, but through one of its

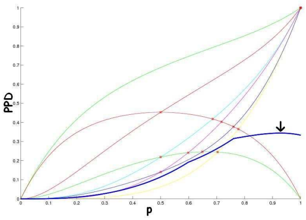

Figure 3: An illustration of Algorithm ComputeMinV for $d = 8, t = 6, v = 3$ with $v_i = 1/3$. The small stars mark the intersection points and the bold curve is the average curves of the 3 minimal curves at each section where the curves correspond to the ppd_i functions. The arrow marks the maximal point returned by ComputeMinV.

neighboring segments. Therefore the information can be used in order to find a p valuable more suitable for the situation. Algorithm ComputeNeighborV computes the weighted average of v neighboring segments (according to a distribution $V = \{v_1, \dots v_v\}$), then finds the maximin point of the new d curves. Note that if the robot currently resides inside the v-neighborhood of the segment, it is excluded, i.e., we average less segments for that case. The probability distribution can be used to express the fact that the adversary tends, for example, to try and penetrate through the segments further away from the robot in its current position. The time complexity of Algorithm ComputeNeighborV is $\mathcal{O}(d^3)$. Figure 4 illustrates the algorithm for $d = 8$, $t = 6$ and $v = 3$.

Algorithm 5 ComputeNeighborV$(v, V, \{\text{ppd}_1, \dots, \text{ppd}_d\})$

```
Set FuncSet ← ∅
for i ← 1 to d do
    i_e = min(d, i + v)
    FuncSet ← Σ_{j=i}^{i_e} v_{j−i+1} × ppd_j
p_opt ← MaxiMin(FuncSet, d)
Return p_opt
```

It is interesting to note the difference between the results of the three possible algorithms used in the illustrated examples of Figures 3 and 4. Note that in both cases $d = 8$, $t = 6$ and $v = 3$. The result returned by the MaxiMin algorithm (used in case of a full knowledge adversary, i.e., $v = 1$) is $p = 0.7037$. When the input is $v = 3$, the optimal p in case of maximizing minimal $v-$ neighborhood is $p = 0.7359$, and the optimal p for maximizing minimal v ppd values is obtained in $p = 0.9273$.

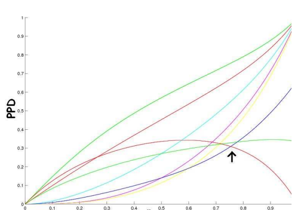

Figure 4: An illustration of Algorithm ComputeNeighborV for $d = 8, t = 6, v = 3$, $v_1 = v_2 = v_3 = 1/3$. All curves are *not* the original ppd_i functions, but the average of $v-$neighborhood of each segment. The arrow points to the maximin point of the new curves.

6 Conclusions and future work

In this paper we considered the problem of multi-robot patrol in adversarial environments, with the existence of uncertainties in the system. We discussed two types of uncertainties: Uncertainty in the robots' perception of their own capabilities (originated in imperfect sensing), and uncertainty of the adversary in its own choices (originated in the adversary's imperfect ability to asses the best penetration spot). In the first part of the paper, we presented three types of sensorial models of the robots, and shown how there models change the probability of penetration detection along the perimeter. In the second part of the paper, we analyzed the optimal patrol algorithm of the robots in case the adversary is uncertain of its optimal penetration spot.

In our future work we consider the following points. First, we would like to further examine possible sensing models and other possible errors in the robotic system, originated for example in faulty movement (rather than faulty sensing). We would also like to further investigate uncertainties of the adversary. As first step in this direction, we are interested in empirically evaluating the possible optimal algorithms we have presented here, and examine their performance in different adversarial models.

References

[1] N. Agmon, G. A. Kaminka, and S. Kraus. Multi-robot fence patrol in adversarial domains. In *IAS-10*, pages 193–201. IOS Press, 2008.

[2] N. Agmon, S. Kraus, and G. A. Kaminka. Multi-robot perimeter patrol in adversarial settings. In *ICRA*, 2008.

[3] N. Agmon, V. Sadov, S. Kraus, and G. A. Kaminka. The impact of adversarial knowledge on adversarial planning in perimeter patrol. In *AAMAS*, 2008.

[4] F. Amigoni, N. Gatti, and A. Ippedico. Game-theoretic approach to determining efficient patrolling strategies for mobile robots. In *IAT*, 2008.

[5] J. L. Devore. *Probability and Statistics for Engineering and the Sciences*. Brooks/Cole Publishing Company, 1991.

[6] M. F. Duarte and Y. H. Hu. Distance-based decision fusion in a distributed wireless sensor network. *Telecommunication Systems*, 26(2-4):339–350, 2004.

[7] Y. Elmaliach, A. Shiloni, and G. A. Kaminka. A realistic model of frequency-based multi-robot fence patrolling. In *AAMAS*, 2008.

[8] P. Paruchuri, J. P. Pearce, M. Tambe, F. Ordonez, and S. Kraus. An efficient heuristic approach for security against multiple adversaries. In *AAMAS*, 2007.

[9] P. Paruchuri, M. Tambe, F. Ordonez, and S. Kraus. Security in multiagent systems by policy randomization. In *AAMAS*, 2007.

[10] J. Pita, M. Jain, F. Ordonez, , M. Tambe, S. Kraus, and R. Magorii-Cohen. Effective solutions for real-world stackelberg games: When agents must deal with human uncertainties, to appear. In *AAMAS*, 2009.

[11] S. Thrun. Probabilistic algorithms in robotics. *AI Magazine*, 21:93–109, 2000.

[12] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Autonomous Robots*, 5(3-4), 1998.