A Game Theoretical Formulation of a Decentralized Cooperative Multi-Agent Surveillance Mission

Paulo E. U. de Souza, Caroline P. C. Chanel
Univesité de Toulouse – ISAE-SUPAERO
Institut Supérieur de l'Aéronautique et de l'Espace
31055 Toulouse Cedex 4, FRANCE
name.surname@isae.fr

Sidney Givigi
Royal Military College of Canada
PO Box 17000, Station Forces
Kingston, Ontario, CANADA
sidney.givigi@rmc.ca

Abstract

This paper presents a multi-aerial-robot coordination game theoretical approach to perform a surveillance mission in a well-structured environment. Such a mission consists in constantly visiting a set of points of interest while minimizing the time interval between successive visits (idleness). The proposed approach optimizes the agents' action selection based on an N-player (cooperative) game framework. The main contributions are: (i) the formulation of an original player's utility function composed of parameters that are independent from the action choices of the others players; (ii) the demonstration that the game solution is the Nash equilibrium, and this equilibrium can be obtained by optimizing separately/individually the single player's action choice; (iii) the proposal of a decentralized algorithm used to conduct the mission, which works considering minimum communication among players. Simulations evaluate the different policies obtained, which are compared using as metric the average idleness of all points of interest. The proposed framework allows for the decrease of the idleness of watched points compared to random action selection, while keeping some kind of randomness of motion (measured by a predictability metric), which can likely be desired to curb the prediction of the team surveillance strategy by an intruder.

Introduction

The recent advancement in decision making techniques for aerial robots, also known as drones, has significantly increased the number of applications for a team of autonomous agents. In certain scenarios, multi-robot systems are more desirable than a single robot due to their robustness, stability, adaptability, and scalability (Meng 2008). For instance, search and rescue missions (Murphy et al. 2008; Suarez and Murphy 2011; Xue, Zeng, and Zhang 2011), autonomous infrastructure inspection (Scaramuzza et al. 2014), or autonomous patrolling systems (Amigoni, Basilico, and Gatti 2009; Portugal and Rocha 2011; Hernández et al. 2013).

In particular, the multi-aerial-robot autonomous patrolling or surveillance problem is very challenging: the robots must navigate through the environment so different locations that are scattered in the operational space, and they have to coordinate their actions in order to optimize the time spent to cover all the desired points of interest (Portugal and Rocha 2013a). One of the key issues of a surveillance mission for a multi-robot system is how to coordinate their behaviors in order to optimize the global performance (Meng 2008). For example, monitoring an area of interest requires that the robots move repeatedly through the environment, and the difficulty is to decide on the paths while optimizing some performance criteria (Pasqualetti, Franchi, and Bullo 2010). Moreover, since surveillance implies the maximization of the number of visits to each node in a given environment, a good surveillance strategy must reduce the time interval between visits to the same location (Chevaleyre 2004).

With the aim of evaluating surveillance stategies, a comparative study using distinct topological environments and different team sizes is presented in (Portugal and Rocha 2013b). This work analyzes the performance and scalability of each patrolling approach. For that, (Portugal and Rocha 2013b) proposed as an evaluation metric the *average idleness of the graph* (Idl_G). In the same point of view, (Chevaleyre 2004) demonstrates that minimizing *worst idleness* will also lead to a smaller average idleness. In any case, the smallest the idleness, the better is the performance.

Another key point argued by some authors is that, for security reasons, it should be suitable to consider irregular time intervals to perform visits on desired locations while optimizing the strategy, in order to avoid that a potential intruder could observe the movement of the patrol members for some time and derive an accurate belief of their strategies (Hernández et al. 2013; Amigoni et al. 2010). The key idea is to make it more difficult for an intruder to predict the motion strategy of the team members.

In this kind of surveillance application, it is well known that the minimal refresh time patrolling problem is *NP-hard* (Pasqualetti, Franchi, and Bullo 2010; Portugal and Rocha 2011; Zhang and Kingston 2015). This means that to update the state of each position at each time step would be computational and memory expensive and impractical in real-world scenarios (Meng 2008; Portugal and Rocha 2011). This is so because in order to improve the efficiency of the collective searching strategy, the action of each robot does not only depend on its own situation, but also on other robots decisions.

In this sense, recent papers have based their approaches on *Game Theory* (Amigoni et al. 2010; Hernández et al. 2013; Meng 2008; Peshkin et al. 2000; An et al. 2012; Khan 2007), which is an elegant way to model an agent's decision making process based on the others agents decisions in a decentralized and distributed way. An example of such a Game Theory application is presented in (Hernández et al. 2013), where Game Theory models of the multi-robot patrolling problem are solved with the use of dynamic and decentralized collaborative approach. Another interesting solution is proposed by (Amigoni et al. 2010) based on Game Theory, which develops a surveillance strategy to drive mobile robots around a known environment in order to avoid intrusions while implementing a non deterministic strategy for their movements in order to make more difficult the task of intruders for they do not know a priori the stochastic distribution of such motions. Others examples can be found in: (Meng 2008), which proposed an N-agent cooperative nonzero-sum game to achieve an optimal overall robots behaviors; (Peshkin et al. 2000) described a gradient-descent policy-search algorithm for cooperative multi-agent domains, where they all share a common payoff; and, (An et al. 2012) that investigated the use of zero-sum games for the protection of critical infrastructures.

For the purpose of a cooperative surveillance mission based on Game Theory, this work addresses the problem of monitoring a closed area by a team of drones minimizing the time to revisit the points of interest (idleness) while keeping some kind of randomness of motion in order to render movements less predictable. Note that, this is neither a coverage problem nor a adversarial problem, but a mix of them. The issue is the development of a dynamic and decentralized approach to multi-aerial-robot cooperation in order to solve the patrolling problem by implementing game theoretical models. In this sense, the main contributions of this work are:

- the formulation of an original player's utility function composed by three parameters that are independent from the action choices of the others players;
- the demonstration that the game solution is a Nash equilibrium, and that this equilibrium can be obtained by optimizing separately and individually the single player's action choice;
- the proposal of a decentralized algorithm used to conduct the mission, which works considering minimum communication among players.

In other words, an original heuristic utility function is presented, where not only the path travel cost is considered, but also the current positions of the other players and the last time since each point of interest was visited. And, based on this utility function, a coordinated game is generated, where the Nash Equilibrium solution guides the player's behavior toward the team goal. In order to reduce the computational complexity the following approach for the solving algorithm is proposed: (1) a fixed path between nodes in the graph and its cost are generated off-line considering the graph does not change during the mission; and (2) the communication between agents and a new game occur only when the destination of each drone has been achieved, instead of at every time step (i.e. the communication is asynchronous).

This work is organized as follows: the considered surveillance problem, its game formulation and the decentralized algorithm proposed to solve this game is presented in Section . Simulation experiments results are shown in Section to evaluate the parameters of the single player's utility function, and their different policies are compared using as metric the average idleness of all points of interest and the overall randomness of the aerial robots' movements. Finally conclusions and future work are discussed in Section .

Problem formulation

This mission can be defined as a frequent visitation problem of all preset points for an aerial robot (here also called drone) team in the lowest possible time interval without having a cycling behavior in order to make the motion model less predictable.

The idea of this paper is to present a method of coordination between drones, based on Game Theory, that is capable of carrying out a monitoring mission on a known *topological model* represented as a graph $\mathbb{G} = (\mathbb{S}, \mathbb{E})$. In this graph $\mathbb{G}$, $\mathbb{S}$ is the set of nodes representing the points of interest in the environment (i.e. positions), and where the edges $\mathbb{E} \subseteq \mathbb{S} \times \mathbb{S}$ define adjacency relationships between the nodes $\mathbb{S}$, i.e., the possible paths between positions or points of interest. Each edge has a cost that represents the time required to move from one node to another. These costs are fixed.

To define the game problem, some assumptions were taken:

- For simplicity, time was discretized in turns;
- Each node can be considered as a point of interest that should be observed, i.e. looking for an intruder;
- Each destination node is a point where the communication among the drones team arises.
- Each drone will select, only once it reaches its destination point, the next point to visit, based on the available information of the others. This means that a new action selection problem will be considered by a drone only when this one has reached the destination point, instead of each time step;
- The drones are defined as "Conscientious Cognitive" agents (Portugal and Rocha 2011), i.e., they choose the next point to visit in the global graph, instead of

in their neighborhood. So, at each time interval, each drone can move from one node to another adjacent, without necessarily selecting a new point of interest;

- All drones have perfect knowledge of the graph model, of their own positions in the graph, the last position informed by the others and their destinations in the graph;
- We assume that each drone can avoid obstacles and collisions;
- The horizon of the mission is considered as infinite.

Therefore, under these assumptions, a Game Theory formulation of the problem is proposed.

Game theory problem formulation

The surveillance mission is defined as a *dynamic game*, where the costs at each time step depend on: the minimal distance between points of interest represented as nodes in a graph, the actual position (node) of the robots and the last time since points of interest (nodes) were visited. Formally, it can be defined as a N-player finite game $\Gamma = (N, \mathbb{A}, u)$, where:

- $N = \{1, \cdots, n\}$ is the finite set of n players, indexed by i;
- $\mathbb{A} = A_1 \times A_2 \times \cdots \times A_n$ represents all possible actions to be taken by all drones;
- $u = h(u_1, \cdots, u_i, \cdots, u_n)$ is the payoff function which is function of the payoff of each single player, with $u : \mathbb{A} \rightarrow \mathbb{R}$, and $u_i : A_i \rightarrow \mathbb{R}$ for each player i.

Players' actions. In conformity with the Game Theory formulation, $\bar{a} = [a_1, a_2, \cdots, a_n]$ is defined as the vector of actions for all $n \in N$ drones and $A_i = \{a_i^1, a_i^2, \cdots, a_i^q\}$, where q is the number of actions at the disposal of the i^{th} drone. Observe that the sets of actions A_i do not need to be equal for all drones; however, in the scenario we are modeling, we will consider the possible actions to be all equal. Then, one may conclude that $\mathbb{A} = A^n$ and the cardinality $|A| = q$. Another point is that, the set A is equal to the subset of states $S = R \subset \mathbb{S}$ meaning that the drone can choose as an action any node $s_k \in S$, then $A = S$.

Payoff function. According to the positions/destinations of drones at time step t, the utility function μ^t can be calculated. The utility is defined as the summation of the utilities of all players involved in the game, i.e.,

$$\mu^t(\bar{a}, \bar{s}^t) = \sum_{\bar{a} \in \mathbb{A}} \mu_i^t(\bar{a}, \bar{s}^t) \tag{1}$$

where $\bar{a}$ is the vector of actions and $\bar{s}^t$ is the state of the drones at time t.

The utility functions for each drone $i \in N$ at time t is defined as

$$\mu_i^t(\bar{a}, \bar{s}^t) = \delta_i(a_i, s_i^t) + \lambda_{-i}(a_i, \bar{s}_{-i}^t) - \rho_i^t(a_i) \tag{2}$$

where:

- $\delta_i(\cdot)$ is the *cost to go* for the drone i from the current position, i.e., the distance for the drone to move from its current position s_i^t to all its possible future locations $a_i^k \in A_i$, with $k \in \{1, \cdots, q\}$. Therefore, considering that $f^*(s_i^t, a_i^k)$ is the optimal distance cost that refers to the optimal (or sub-optimal, when the optimal cannot be calculated) path from node s_i^t to a_i^k, one gets:

$$\delta_i(a_i^k, s_i^t) = f^*(s_i^t, a_i^k) \tag{3}$$

- $\lambda_{-i}(\cdot)$ is the weighted sum of all other drones *inverted distance* ($\Psi(\cdot)$), where *inverted distance* is defined as a value that is equal to the maximum distance for the nearest point and decreases with the distance. Therefore, for the chosen action $a_i^k \in A_i$ we have:

$$\Psi_j(s_j^t, a_i^k) = \max_{a_j^p \in A_j} (\delta_j(a_j^p, s_j^t)) + \min_{a_j^r \in A_j} (\delta_j(a_j^r, s_j^t)) - \delta_j(a_i^k, s_j^t). \tag{4}$$

The idea here is to make the points more distant from the other drones more attractive for drone i, then, $\lambda_{-i}(\cdot)$ for a determined action $a_i^k \in A_i$ is given by:

$$\lambda_{-i}(a_i^k, \bar{s}_{-i}^t) = \frac{\sum_{j=1}^{n-1} \{\Psi_j(s_{-i,j}^t, a_i^k)\}}{n-1} \mid j \neq i. \tag{5}$$

- Finally, $\rho_i^t(a_i^k)$ is the *expected reward* to reach the node a_i^k. These values are collected (turn into zero) when a drone passes over the position and increase by a factor γn each time step that they are not visited, where γ is a normalizer constant and n is the number of drones:

$$\rho_i^{t+1}(a_i^k) = \rho_i^t(a_i^k) + (\gamma n) \mid \gamma \in [0, 1] \tag{6}$$

Note that since all action sets A_i are equal to A, the *expected reward* is equal for all drones.

Based on the definition of the utilities, the minimal global cost for this game would be:

$$\mu^{t*}(\bar{a}, \bar{s}^t) = \min_{\bar{a} \in \mathbb{A}} \mu^t(\bar{a}, \bar{s}^t) \tag{7}$$

Notice that the utilities for each drone i, $\mu_i^t(\cdot)$, is only directly dependent on a_i and only indirectly takes into consideration the actions of all other drones (through $\lambda_{-i}(\cdot)$). So, individual's utility functions are composed by three parameters that are, by definition, independent from the action choices of the others players. In this sense, (7) may be rewritten as:

$$\mu^{t*}(\bar{a}, \bar{s}^t) = \min_{a_1 \in A_1} \mu_1^t(a_1, \bar{s}^t) + \cdots + \min_{a_n \in A_n} \mu_n^t(a_n, \bar{s}^t) \tag{8}$$

Therefore, the minimal global cost strategy solution for drone i, $\sigma_i^{t^*}$, is adopted for the decoupled game as described in:

$$\sigma_i^{t^*} = \operatorname*{argmin}_{a_i \in A_i} \mu_i^t(a_i, \bar{s}^t) \tag{9}$$

It means that for this formulation the action choice for drone i is independent from the action choices of the others drones. We are now ready to enunciate and prove the following theorem:

This result is summarized in the following theorem.

Theorem 1. *The N-player finite game* $\Gamma = (N, \mathbb{A}, u)$ *with utility functions defined in* (1) *and* (2) *possess a pure-strategy equilibrium.*

Proof. Let us consider a *Wonderful Life Utility* for drone i.

$$WLU_i = \phi(z) - \phi(z_{-i})$$

where z is the collection of all players and z_{-i} is the collection of all players except player i. It is clear, that if one considers $\phi = \mu^t(\cdot)$, then

$$WLU_i = \mu_i(\cdot)$$

Therefore, the game Γ becomes a *Potential Game*, i.e., the drones' utilities $\mu_i(\cdot)$ are aligned to the global utility $\mu(\cdot)$. Therefore, it is guaranteed to have a pure-strategy equilibrium according to *Corollary 2.2* of (Monderer and Shapley 1996). □

Moreover, it may be verified that this pure-strategy equilibrium is indeed the Nash equilibrium of the game (Philip, Givigi Jr, and Schwartz 2014), for:

$$\mu^t(\bar{a}^*, \bar{s}^t) \leq \mu^t([a_1^*, \cdots, a_{j-1}^*, a_j, a_{j+1}^*, \cdots, a_n^*], \bar{s}^t), \ \forall j \in N.$$

Finally, notice that this decentralized approach, where the action selection is formalized as a potential game, allows to drones to take decision in an asynchronous way, as each drone selects the next action only once it reaches the destination point based only on available (last) information.

Algorithm for coordination

Algorithm 1 presents the process inside each drone. To better explain this algorithm we introduce two execution status on which drones' action selection relies. Before a drone starts to move it changes its status to *Busy* and when it arrives at the destination point it changes to *NotBusy*.

When one of then is *NotBusy*, i.e when it reaches a destination point, (line 2 of Alg. 1), it sends a message of its position and updates its knowledge of the position of the *NotBusy* drones and the destination position for the *Busy* ones (lines 3 and 4). Then, the drone changes the cost value of its current position s_i^t to ∞ which forces it to move to somewhere else (line 5). After, it proceeds all calculations for compute the cost vector μ_i^t, and it selects the minimal cost strategy (lines 6 and 7) applying the proposed approach. The concerned drone computes the global *minimal cost* knowing that the others will do the same. In this way, using game theory to predict what others will do, coordination arises. Finally, it informs its next destination to the others, change its status, and starts to navigate again (lines 8-10).

Algorithm 1 Patrol mission for $Drone_i$

1: **while** True **do**
2:   **if** status == $NotBusy$ **then**
3:     **report** current position
4:     **read** messages
5:     **assign** infinity to current position cost - $f^*(s_i^t, s_i^t) = \infty$
6:     **compute** the cost vector μ_i^t (Eq. (2))
7:     **find** and **select** the minimal cost strategy (Eq. (9))
8:     **report** destination
9:     **assign** $Busy$ to its current status
10:     **start** navigation
11:   **else**
12:     **if** position == destination **then**
13:       **assign** $NotBusy$ to its current status
14:     **else**
15:       **continue** navigation
16:     **end if**
17:   **end if**
18: **end while**

When the drone is *Busy*, it only continuously verifies if the destination point is reached, if is the case, it changes its status to *NotBusy*, if not, it continues to navigate (lines 11-15).

To evaluate the proposed approach an application case is presented next.

Simulation Experiments

Setup

The topological model considered for experiments is shown in Figure 1. This topological model is represented by the graph $\mathbb{G} = (\mathbb{S}, \mathbb{E})$ in that the nodes $\mathbb{S} = R \cup D$ represent some positions in the environment, with $R = \{r_1, r_2, ..., r_q\}$ the set of positions inside the rooms and corridors (points of interest) and D the set of doors. The edges $\mathbb{E} \subseteq \mathbb{S} \times \mathbb{S}$ define adjacency relationships between the nodes $\mathbb{S}$, i.e., the possible paths. Each edge has a fixed cost associated with, here, the time required to move from one node to another.

To evaluate the approach, a patrol simulator has been developed in Python 2.7.8. In this simulation model there are 25 *points of interest* (R), the 7 doors are considered as connection points (D) and 60 edges, i.e. $|R| = 25, |D| = 7, |\mathbb{E}| = 60$ respectively, as shown in Figure 1. Please note that, the set S of possible locations is equal to R (we do not consider doors - these specific connection points - as points of interest), and the set of actions A_i of each drone is equal to S.

Figure 2 shows a moment during the mission with three drones. In this simulation, the color of the floor is related to the *idleness* of the point, the blue areas are associated with greater rewards ρ^t.

We note that, as commented before, the approach presented in this paper is neither a coverage problem nor an adversarial problem, but a mix of them. The

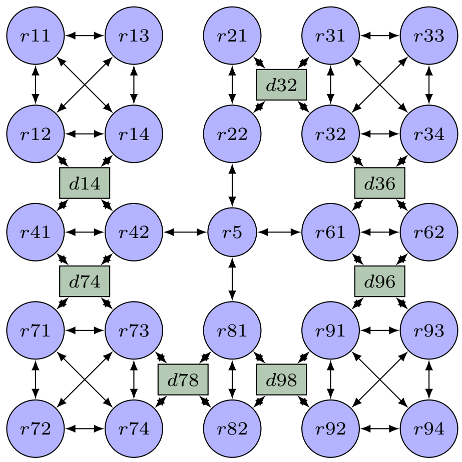

Figure 1: Topological model with the points of interest and all possible transitions.

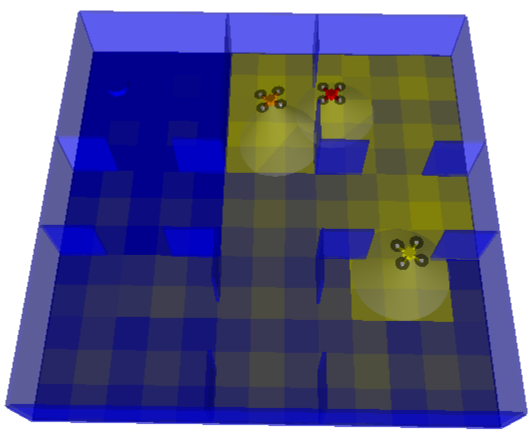

Figure 2: Geometric model.

mixed problem proposed, as far as we know, is for the first time studied, and for this reason a comparison to previous approaches is not straightforward possible.

In this context and in order to verify the performance of the patrolling algorithm considering different numbers of drones in the team and the influence of each component of the utility function (δ, λ and ρ from eq. 2), five scenarios were designed:

- *complete Utility* - where all components of the utility function were used;
- *no Reward* - where ρ was removed from the utility function;
- *no Inverse* - where λ was removed from the utility function;
- *only Distance* - where ρ and λ were removed from the utility function;
- *random* - where the drones select their destinies randomly.

1000 missions for each scenario were played and each patrol mission ran until each point of interest was visited at least fifty times.

Metrics

This study has been focused on (1) the interval between visits (*idleness*) and (2) the difficulty of prediction of the next position of the patrols (*predictability*). For the first one the *average idleness of the graph* Idl_G (Portugal and Rocha 2013b) was used as a metric, and for the second one, the Ljung-Box test (Box, Jenkins, and Reinsel 2008) results were considered.

The *average idleness of the graph* (Idl_G) proposed by (Portugal and Rocha 2013b) is defined as:

Starting with the *instantaneous idleness* (Idl_{t_k}) of a position $s_i \in S$ in the time step t_k:

$$Idl_{t_k}(s_i) = t_k - t_{last_{visit}} \tag{10}$$

where $t_{last_{visit}}$ corresponds to the last time step when that point s_i was visited by a drone. Consequently, the *average idleness* (Idl_m) of a point s_i in a total time T is defined as:

$$Idl_m(s_i) = \frac{\sum_{k=0}^{T} Idl_{t_k}(s_i)}{T} \tag{11}$$

And, finally, the *average idleness of the graph* (Idl_G) is defined as:

$$Idl_G = \frac{\sum_{i=0}^{|S|} Idl_m(s_i)}{\mid S \mid} \tag{12}$$

where $\mid S \mid$ represents the cardinality of the set S.

On the other hand, to evaluate how "unpredictable" the drone paths were, the *Ljung-Box test* was used. This statistical test allows the measurement of the "overall randomness" based on a number of lags of a time series by means of a single value Q:

$$Q = p(p+2)\sum_{l=1}^{m} \frac{\hat{\rho}_l^2}{p-l} \tag{13}$$

and:

$$\hat{\rho}_l = \frac{\sum_{k=1}^{p-l} (Y_i - \bar{Y})(Y_{i+l} - \bar{Y})}{\sum_{k=1}^{p} (Y_i - \bar{Y})^2} \tag{14}$$

where p is the sample size, m is the number of lags being tested, $\hat{\rho}_l$ is the autocorrelation function (ACF) at lag l and $Y = (Y_1, \cdots, Y_p)$ are the measurements. For a significance level α, the critical region for rejection of the hypothesis of randomness is given by the percentile $(1-\alpha)$ of the chi-squared distribution with m degrees of freedom:

$$Q > \chi^2_{1-\alpha,m} \tag{15}$$

Thus, if Eq. 15 is *TRUE* it is possible to say that exists a linear correlation, in other words, the information of past positions allows an inference of future positions. Moreover, Q weights the correlation process, i.e., the higher the value the greater the correlation.

Obviously, all tested scenarios have a high degree of autocorrelation between adjacent and near-adjacent positions, due to the movement model of the drones. Even though, Q can identify an appropriate time series model even when the data are not random.

In the end, in order to use this values as a metric of predictability (π) in the present work, Q for each scenario c was normalized by the worst value (per number of drones n):

$$\pi_c^n = \frac{Q_c^n}{max(Q^n)} \tag{16}$$

In this work, for a specific number of drones, the degrees of freedom m were selected among all scenarios as the smallest median number of steps necessary to complete a cycle (i.e., to visit all positions at least once) with an $\alpha = 0.05$.

Results

Figure 3 shows that increasing the number of drones implies the convergence of idleness, which will be zero when the number of drones reaches the number of points of interest. Nevertheless, looking to these charts it is possible to infer the minimal number of drones to achieve the goal of the mission in an efficient way, defined as the ratio $\frac{|N|}{Idl_G}$ (best cost-benefit ratio). Interestingly, in the *no Inverse* scenario, differently from the others, the idleness seems to be almost steady with two drones or more. The reason for that must be interpreted with caution, but it seems that when they do not need to coordinate their moves (and that is in essence what λ do), they can reach a local optimum very fast; however, these values will eventually decrease to zero. Also, it can be seen that the variance decreases with the number of drones, except for the *no Inverse* scenario. Together these results provide important insights into the approach presented. It is easy to observe the importance of each cost variable and their contribution for idleness.

The increase of the mission performance with the rise in the number of drones in all scenarios for both metrics, *idleness* and *predictability*, is shown in Figure 4. The results also indicate that when ρ was not used (*no Reward*) the paths became more predictable (greater values of π). With a single drone the scenarios *no Reward* and *only Distance* achieved the same value, as expected, since, in this case, they have the same utility function.

On the other hand, still looking to the single drone case, a very predictable path can be identified for *no Inverse* and *complete Utility* scenarios. A possible explanation for these results may be that they tried to maximize the reward earned at each iteration. Interestingly, for more than one drone, the *no Reward* scenario appears to maintain predictable paths. Overall, these charts indicate that the best scenario is the *complete Utility*.

The charts in Figure 5 present a slice of the surveillance mission for three drones with 100 arbitrarily collected steps from all scenarios, where each line represents the path of a drone. What is interesting here is that in *complete Utility*, *no Reward* and *only Distance* the drones tend to maintain themselves in a separated sector from the others. The *Random* scenario presented, as expected, the worst results as the drones moved randomly around the environment. In *complete Utility* and *no Inverse*, the path were longer than the others and with almost no local cycles, indicating global movement in contrast with some “sawtooth” path in the others charts. Another interesting behavior is observed in the *no Inverse* scenario where it seems like that the drones are following each other, maintaining almost the same path. The most striking observation to emerge from the data comparison is that the *complete Utility* generated longer and clearer paths, maintaining the drones separated for almost all time and changing the patrol sectors once in a while.

Conclusion and future work

A multi-aerial-robot game theoretical surveillance approach is proposed in this paper. The main contributions are the development of a dynamic and decentralized approach to cooperation in order to solve the patrol problem by implementing game theoretical models. In this way, a heuristic utility function is presented, where not only the travel cost is considered, but also the current positions of the other team members as well as the last time each point of interest was visited. Based on this utility function, an *N-player game* is played inside each drone, wherein the *Nash Equilibrium* was applied to the drones in order to make their decisions. To improve the real-time performance, the game is played only in the destination points of each drone. Five scenarios and two metrics were designed and used to evaluate the proposed model. Overall, the results indicate that the proposed *utility function* can minimize the *idleness* while also minimizing the patrol *predictability*.

There is abundant room for further progress in this proposed model. Future studies should consider:

- an unreliable human operator in the control loop;
- an intruder and different payoff values for drones and positions;
- leader-follower equilibria;
- uncertainty in the movements and in the detection of the evader;
- imperfect and not cost-free communication.

As it is known, depending on the type of game used, the computational complexity would become intractable with a large-scale team. This was the reason why a potential game was proposed. Furthermore, in the near future we intend to investigate the scalability of this approach. Also, new models for the utilities of the drones will be the focus of future investigations.

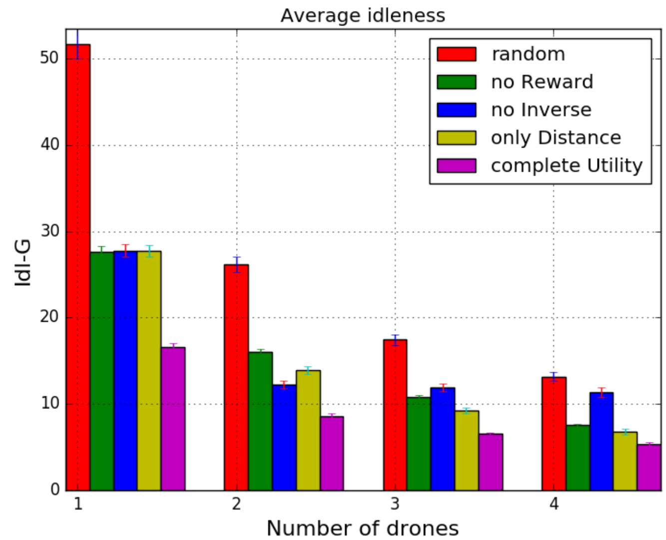

Figure 3: Average idleness of the graph per number of drones.

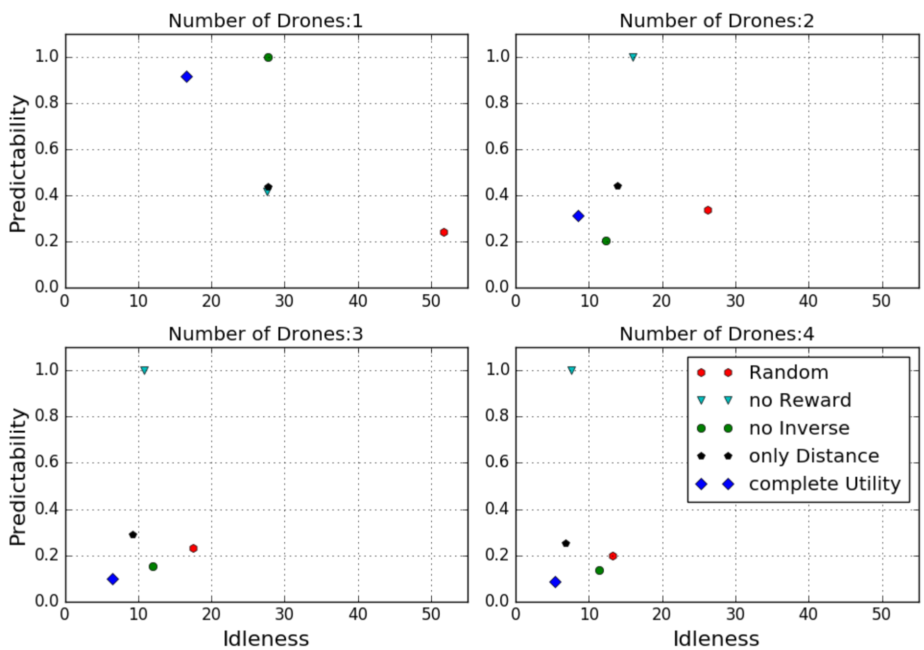

Figure 4: Predictability versus idleness.

References

Amigoni, F.; Basilico, N.; and Gatti, N. 2009. Finding the optimal strategies for robotic patrolling with adversaries in topologically-represented environments. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, 819–824. IEEE.

Amigoni, F.; Basilico, N.; Gatti, N.; Saporiti, A.; and Troiani, S. 2010. Moving game theoretical patrolling strategies from theory to practice: An usarsim simulation. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 426–431. IEEE.

An, B.; Kempe, D.; Kiekintveld, C.; Shieh, E.; Singh, S.; Tambe, M.; and Vorobeychik, Y. 2012. Security games with limited surveillance. *Ann Arbor* 1001:48109.

Box, G.; Jenkins, G.; and Reinsel, G. 2008. *Time Series Analysis: Forecasting and Control*. Wiley Series in Probability and Statistics. Wiley.

Chevaleyre, Y. 2004. Theoretical analysis of the multi-agent patrolling problem. In *Intelligent Agent Technology, 2004. (IAT 2004). Proceedings. IEEE/WIC/ACM International Conference on*, 302–308.

Hernández, E.; Cerro, J. d.; Barrientos, A.; et al. 2013. Game theory models for multi-robot patrolling of infrastructures. *International Journal of Advanced*

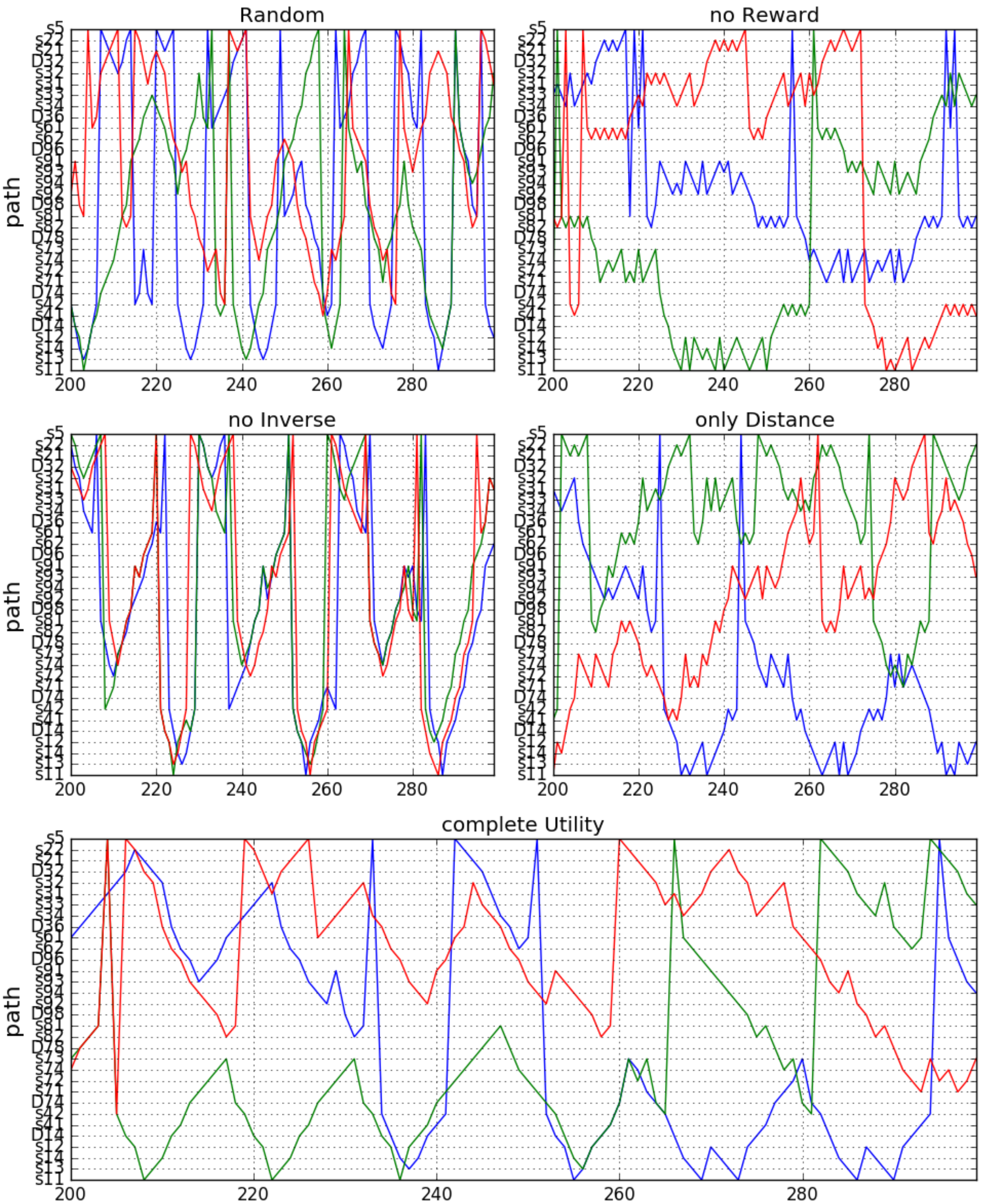

Figure 5: Paths generated with three drones. The "path" axis contains the nodes of the graph.

Robotic Systems 10(181).

Khan, M. E. 2007. Game theory models for pursuit evasion games. Technical report, Technical report, University of British Columbia, Vancouver.

Meng, Y. 2008. Multi-robot searching using game-theory based approach. *International Journal of Advanced Robotic Systems* 5(4):341–350.

Monderer, D., and Shapley, L. S. 1996. Potential games. *Games and Economic Behavior* 14(1):124 – 143.

Murphy, R. R.; Tadokoro, S.; Nardi, D.; Jacoff, A.; Fiorini, P.; Choset, H.; and Erkmen, A. M. 2008. Search and rescue robotics. In *Springer Handbook of Robotics*. Springer. 1151–1173.

Pasqualetti, F.; Franchi, A.; and Bullo, F. 2010. On optimal cooperative patrolling. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, 7153–7158. IEEE.

Peshkin, L.; Kim, K.-E.; Meuleau, N.; and Kaelbling, L. P. 2000. Learning to cooperate via policy search. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, 489–496. Morgan Kaufmann Publishers Inc.

Philip, G.; Givigi Jr, S. N.; and Schwartz, H. M. 2014. Cooperative navigation of unknown environments using potential games. *International Journal of Mechatronics and Automation* 4(3):173–187.

Portugal, D., and Rocha, R. 2011. A survey on multi-robot patrolling algorithms. In *Technological Innovation for Sustainability*. Springer. 139–146.

Portugal, D., and Rocha, R. P. 2013a. Distributed multi-robot patrol: A scalable and fault-tolerant framework. *Robotics and Autonomous Systems* 61(12):1572–1587.

Portugal, D., and Rocha, R. P. 2013b. Multi-robot patrolling algorithms: examining performance and scalability. *Advanced Robotics* 27(5):325–336.

Scaramuzza, D.; Achtelik, M. C.; Doitsidis, L.; Friedrich, F.; Kosmatopoulos, E.; Martinelli, A.; Achtelik, M. W.; Chli, M.; Chatzichristofis, S.; Kneip, L.; et al. 2014. Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in gps-denied environments. *Robotics & Automation Magazine, IEEE* 21(3):26–40.

Suarez, J., and Murphy, R. 2011. A survey of animal foraging for directed, persistent search by rescue robotics. In *Safety, Security, and Rescue Robotics (SSRR), 2011 IEEE International Symposium on*, 314–320. IEEE.

Xue, S.; Zeng, J.; and Zhang, G. 2011. A review of autonomous robotic search. In *Electrical and Control Engineering (ICECE), 2011 International Conference on*, 3792–3795. IEEE.

Zhang, M., and Kingston, D. 2015. Time-space network based exact models for periodical monitoring routing problem. In *American Control Conference (ACC), 2015*, 5264–5269. IEEE.