Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Coordination of control in robot teams using game-theoretic learning

M Smyrnakis * **S M Veres** *

* *Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, S1 3JD, UK (e-mail:m.smyrnakis,s.veres@sheffield.ac.uk).*

Abstract: This paper presents a distributed decision making approach to the problem of control effort allocation to robotic team members. The objective is for a team of autonomous robots to coordinate their actions in order to efficiently complete a task. A novel controller design methodology is proposed which allows the robot team to work together based on a game-theoretic learning algorithms using fictitious play and extended Kalman filters. In particular each robot of the team predicts the other robots' planned actions while making decision to maximise its own expected reward that is dependent on the reward for joint successful completion of the task. After theoretical analysis the performance of the proposed algorithm is tested on a scenario of collaboration between material handling and patrolling robots in a warehouse.

1. INTRODUCTION

Teams of robots can be used in many domains such as mine detection [Zhang et al., 2001], exploration of unknown environments [Simmons et al., 2000, Madhavan et al., 2004], medication delivery in medical facilities [Evans and Krishnamurthy, 1998] and inspection of hazardous areas which contain materials that are harmful for humans. In these cases teams of intelligent robots should coordinate in order to accomplish a desired task. When autonomy is a desired property of a multi-robot system then coordination between the robots of the team is necessary.

Game theory has been used in the past to design optimal controllers when the objective is coordination. In these results the agents/robots would eventually reach the Nash equilibrium of a coordination game. Semsar-Kazerooni and Khorasani [2009, 2008] used local and global components in each agent's cost function and searched for the Nash equilibrium of the game. Bauso et al. [2006] also searched for the Nash equilibrium of games using agents' cost functions which were based on local components and assumed that the states of the other agents were constant.

In this work we propose a collaborative controller design methodology which is based on game theory and it overlaps with the topic of distributed optimisation [Chapman et al., 2011]. Each agent i strives to minimise a global control cost function through minimising its private control cost function which is associated with the global one. The private cost functions of an agent i incorporates terms that not only depend on agent i but also on costs associated with the actions of all the agents. As we are interested in autonomous robots we interpret the coordination problem as a distributed optimisation problem. It is well known that many decentralised optimisation tasks can be cast as potential games [Wolpert and Tumer, 2004, Arslan et al., 2007], and the search of an optimal solution can be seen as the task of finding Nash equilibria in a game. Thus it is feasible to use iterative learning algorithms from the game-theoretic literature, such as fictitious play, to solve decentralised optimisation problems.

In this paper we propose a learning algorithm based on fictitious play [Brown, 1951] which serves as the coordination mechanism of the controllers of team members. Instead of finding the Nash equilibrium of the game, which is not possible in polynomial time for some games [Daskalakis et al., 2006], we allow agents to learn how they will minimise their cost function through communication and interaction with other agents. Thus, in our proposed controller design methodology there is a coordination phase where agents learn other agents policies and then they use this knowledge to decide the action that minimise their cost functions.

Fictitious play is a learning process where players choose an action that maximises their expected rewards based on their beliefs about their opponents' strategies. The players update these beliefs after observing their opponents' actions. Even though fictitious play converges to the Nash equilibrium for certain categories of games [Fudenberg and Levine, 1998], this convergence can be very slow because of the assumption that players use a fixed strategy in the whole game. Speed up of the convergence can be facilitated by an alternative approach, which was presented by Smyrnakis and Leslie [2010], where opponents' strategies vary through time and players use particle filters to predict them. Though providing faster convergence, this approach has the drawback of high computational costs of the particle filters. In applications where the computational cost is important, as the coordination of many UAVs, the particle filters approach is intractable. The alternative that we propose is to use extended Kalman filters (EKF) instead to predict opponents' strategies. EKFs have much smaller computational costs than the particle filter variants of fictitious play algorithm that have been proposed by Smyrnakis and Leslie [2010]. Moreover in contrast to [Smyrnakis

⋆ This work was supported by EPSR Research Grant No EP/J011894/2: Distributed Sensing, Control and Decision Making.

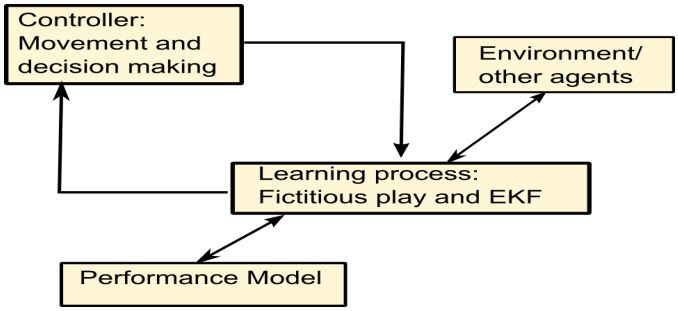

Fig. 1. A feedback loop controller for decision making of autonomous robots

and Leslie, 2010] we provide a proof of convergence to Nash equilbrium of the proposed learning algorithm for potential games.

From implementations' point of view, the controller methodology we propose can be considered as part of the "mental" capabilities of each software agent controlling a robot as in Lincoln et al. [2013]. While the agent focuses on its own duties such as collision detection and observing its environment and observing what the other robotic agents are doing, it can also estimate their anticipated actions within a generic rational agent architecture. Figure 1 outlines this collaborative skill of the agent in a block diagrammatical form.

The remainder of this paper is organised as follows. We start with a brief description of relevant game theory. Section 5 describes how we construct the performance model by casting the control problem as a game. Section 3 introduces the learning algorithm that we use in our controller, Section 5 contains the main theoretical results and Section 6 presents the simulation results we obtained. In the final section we conclude by a summary of results, open problems and future work.

2. GAME THEORY DEFINITIONS

In this section we will briefly present some basic definitions from game theory, since the learning block of our controller is based on it. A game Γ is defined by a set of players $\mathcal{I}$, $i \in 1, 2, \ldots, \mathcal{I}$, who can choose an action, s^i, from a finite discrete set S^i. We then can define the joint action s, $s = (s^1, \ldots, s^{\mathcal{I}})$, that is played in a game as an element of the set product $S = \times_{i=1}^{i=\mathcal{I}} S^i$. Each Player i receives a reward, r^i, after choosing an action s^i. The reward, utility, is a map from the joint action space to the real numbers, $r^i : S \to R$. We will often write $s = (s^i, s^{-i})$, where s^i is the action of Player i and s^{-i} is the joint action of Player i's opponents. When players select their actions using a probability distribution they use mixed strategies. The mixed strategy of a Player i, σ^i, is an element of the set Δ^i, where Δ^i is the set of all the probability distributions over the action space S^i. The joint mixed strategy, σ, is then an element of $\Delta = \times_{i=1}^{i=\mathbb{I}} \Delta^i$. Analogously to the joint actions we will write $\sigma = (\sigma^i, \sigma^{-i})$. The expected utility a player i will gain if he chooses a strategy σ^i (resp. s^i), when his opponents choose the joint strategy σ^{-i} is $r^i(\sigma^i, \sigma^{-i})$ (resp. $r^i(s^i, \sigma^{-i})$).

A common decision rule in game theory is best response (BR). Best response is defined as the action that maximises players' expected utility given their opponents' strategies. Thus for a specific mixed strategy σ^{-i} we evaluate the best response as:

$$BR^i(\sigma^{-i}) = \underset{s^i \in S}{\operatorname{argmax}} \quad r^i(s^i, \sigma^{-i}) \tag{1}$$

Nash [1950], showed that every game has at least one equilibrium, which is a fixed point of the best response correspondence, $\sigma^i \in BR(\sigma^{-i})$. This implies that if a mixed strategy $\hat{\sigma}$ is a Nash equilibrium then it is not possible for a player to increase his utility by unilaterally changing his strategy. When all the players in a game select their actions using pure strategies then the equilibrium actions are referred to as pure strategy Nash equilibria.

A class of games of particular interest are potential games because distributed optimisation tasks can be cast as potential games. In potential games the global utility and players' utilities has the following attribute:

$$r^i(s^i, s^{-i}) - r^i(\tilde{s^i}, s^{-i}) = \phi(s^i, s^{-i}) - \phi(\tilde{s^i}, s^{-i}) \tag{2}$$

where ϕ is a potential function and the above equality stands for every player i, for every action $s^{-i} \in S^{-i}$, and for every pair of actions s^i, $\tilde{s^i} \in S^i$. Moreover potential games have at least one pure Nash equilibria, hence there is at least one joint action s where no player can increase their reward, i.e. their potential function, through a unilateral action. For instance the ""wonderful life" utility [Wolpert and Tumer, 2004, Arslan et al., 2007] can be used as global utility to act as a potential for the system.

3. THE LEARNING PROCESS

In this section we present a combination of fictitious play and extended Kalman filters as the algorithms that we will use in the learning block of the proposed controller. We briefly present the classic fictitious play algorithm and how it can be combined with extended Kalman filters in a decision making algorithm.

3.1 Fictitious play

Fictitious play [Brown, 1951], is a widely used learning technique in game theory. In fictitious play each player chooses his action according to the best response of his beliefs about his opponent's strategy.

Initially each player has some prior beliefs about the strategy that his opponent uses to choose an action based on a weight function κ_t. The players, after each iteration, update the weighting functions, and therefore their beliefs, about their opponent's strategy and play again the best response according to their beliefs. More formally, in the beginning of a game players maintain some arbitrary non-negative initial weighting functions κ_0^j, $j = 1, \ldots, \mathcal{I}$, $j \neq i$that are updated using the formula:

$$\kappa_t^j(s^j) = \kappa_{t-1}^j(s^j) + \mathfrak{I}_{s_t^j = s^j} \tag{3}$$

for each j, where $\mathfrak{I}_{s_t^j = s^j} = \begin{cases} 1 \text{ if } s_t^j = s^j \\ 0 \text{ otherwise.} \end{cases}$.

Players assume that their opponents choose their actions using a fixed mixed strategy. It is natural then to use

a multinomial distribution to approximate an opponent's mixed strategy. The parameters of the multinomial distribution can be estimated using the maximum likelihood method. The mixed strategy of opponent j is then estimated from the following formula:

$$\sigma_t^j(s^j) = \frac{\kappa_t^j(s^j)}{\sum_{s' \in S^j} \kappa_t^j(s')}. \quad (4)$$

When (4) is used, the recent observations have the same weight as the initial ones, which can lead to poor adaptation when the other players choose to change their strategies.

3.2 Fictitious play as a state space model

A more realistic assumption is to presume that players are intelligent and change their strategies according to the other players' actions. We follow Smyrnakis and Leslie [2010] and will represent the fictitious play process as a state-space model. According to the state space model each player has a propensity $Q_t^i(s^i)$ to play each of their available actions $s^i \in S^i$, and then to form a strategy based on these propensities. Finally players can choose an action based on their strategy and the best response decision rule. Because players have no information about the evolution of their opponents' propensities, and under the assumption that the changes in propensities are small from one iteration of the game to another, we model propensities using a Gaussian autoregressive prior on all propensities zSmyrnakis and Leslie [2010]. We set $Q_0 \sim N(0, I)$, where I is the identity matrix, and recursively update the value of Q_t according to the value of Q_{t-1} as follows:

$$Q(s_t) = Q(s_{t-1}) + \eta_t \quad (5)$$

where $\eta_t \sim N(0, \chi^2 I)$. The propensities are connected with the measurements' layer, actions by the following sigmoid equation for every $s^i \in S^i$

$$Qm(s^i) = \frac{\exp\left(Q_t^i(s^i)/\tau\right)}{\sum_{\tilde{s} \in S^i} \exp\left(Q_t^i(\tilde{s})/\tau\right)}. \quad (6)$$

3.3 Fictitious play and EKF

For the rest of this paper we will only consider inference over a single opponent mixed strategy in fictitious play. Separate estimates will be formed identically and independently for each opponent. We therefore consider only one opponent, and we will drop all dependence on player i, and write s_t, σ_t and Q_t for player i's opponent's action, strategy and propensity respectively. Moreover for any vector x, $x[j]$ will denote the j_{th} element of the vector and for any matrix y, $y[i, j]$ will denote the $(i, j)_{th}$ element of the matrix.

We can use the following state space model to describe the fictitious play process:

$$Q_t = Q_{t-1} + \xi_{t-1}$$
$$Qm_t = h(Q_t) + \zeta_t$$

where $\xi_{t-1} \sim N(0, \Xi)$, is the noise of the state process and $\zeta_t \sim N(0, Z)$ is is the error of the observation state with zero mean and covariance matrix Z, which occurs because we approximate a discrete process like best responses (1), using a continuous function $h(\cdot)$. Hence we can combine the EKF with fictitious play as follows.

At time $t-1$ Player i has an estimation of his opponent's propensity using a Gaussian distribution with mean m_{t-1} and variance P_{t-1}, and has observed an action s_{t-1}. Then at time t he uses EKF prediction step to estimate his opponent's propensity. The mean and variance of $p(Q_t|s_{1:t-1})$ of the opponent's propensity approximation are:

$$m_t^- = m_{t-1}$$
$$P_t^- = P_{t-1} + \Xi$$

Player i then evaluates his opponents strategies using his estimations as:

$$\sigma_t(s_t) = \frac{exp(m_t^-[s_t]/\tau)}{\sum_{\tilde{s} \in S} exp(m_t^-[\tilde{s}]/\tau)}. \quad (7)$$

where $m_t^-[s_t]$ is the mean of Player i's estimation about the propensity of his opponent to play action s_t. Player i then uses the estimation of his opponent strategy (7) and best responses (1), to choose an action. After observing the opponent's action s_t, Player i correct his estimations about his opponent's propensity using the update equations of EKF process. The update equations are:

$$v_t = z_t - h(m_t^-)$$
$$S_t = H(m_t^-) P_t^- H^T(m_t^-) + Z$$
$$K_t = P_t^- H^T(m_t^-) S_t^{-1}$$
$$m_t = m_t^- + K_t v_t$$
$$P_t = P_t^- - K_t S_t K_t^T$$

where $h = \frac{exp(Q_t[s^{'}]/\tau)}{\sum_{\tilde{s} \in S} exp(Q_t[\tilde{s}]/\tau)}$, and τ is a temperature parameter. The Jacobian matrix $H(m_t^-)$ is defined as

$$[H(m_t^-)]_{j,j'} = \begin{cases} \dfrac{\sum_{j \neq j'} \exp(m_t^-[j]) \exp(m_t^-[j'])}{(\sum_j \exp(m_t^-[j]))^2} & \text{if } j = j' \\ -\dfrac{\exp(m_t^-[j]) \exp(m_t^-[j'])}{(\sum_j \exp(m_t^-[j]))^2} & \text{if } j \neq \text{j'} \end{cases}.$$

Table 1 summarises the fictitious play algorithm when EKF is used to predict opponents strategies.

4. THE MAIN RESULTS

In this section we present our convergence results for games with at least one pure Nash equilibrium. The results are valid for the EKF fictitious play algorithm of Table 1, when the covariance matrices Ξ and Z are defined as $\Xi = (\tilde{\xi} + \epsilon)I$ and $Z = (1/t)I$ respectively, where $\tilde{\xi}$ is a constant, ϵ is an arbitrarily small Gaussian random variable, $\epsilon \sim N(0, \Psi)$, t is the t_{th} iteration of fictitious play, and I is the identity matrix.

The EKF fictitious play algorithm has the following properties:

Proposition 1. If at iteration t of the EKF fictitious play algorithm, action s is played from Player i's opponent, then the estimation of his opponent propensity to play action s increases, $m_{t-1}[s] < m_t[s]$. Moreover if $\Delta[i] = m_t[i] - m_{t-1}[i]$, then $\Delta[s] > \Delta[j] \forall j \in S^j$, where S^j is the action space of the j_{th} opponent of Player i. Therefore since $\sum_{j \in S^j} \sigma_j = 1$, σ_t^j will be also increased.

(1) At time t Player i maintains some estimations about his opponents propensity up to time $t-1$, $p(Q_{t-1}|s1:t-1)$. Thus he has an estimation of the mean m_{t-1} and the covariance P_{t-1} of this distribution.
(2) Player i updates his estimations about his opponents' propensities $p(Q_t|s1:t-1)$ using equations, $m_t^- = m_{t-1}$, $P_t^- = P_{t-1} + \Xi_{t-1}$.
(3) Based on the weights of step 1 each player updates his beliefs about his opponents' strategies using $\sigma_t^j(s^j) = \frac{exp(m_t^-(j)/\tau)}{\sum_{j'} exp(m_t^-(j)/\tau)}$.
(4) Player i chooses an action based on the beliefs of step 3 and best response decision rule.
(5) He observes opponents' actions s_t.
(6) Update his estimations of opponents' propensities using $m_t = m_t^- + K_t v_t$ and $P_t = P_t^- - K_t S_t K_t^T$.

Table 1. EKF Fictitious Play algorithm

	L	R
U	1,1	0,0
D	0,0	1,1

Table 2. Simple coordination game

Proof. The proof of Proposition 1 is on Appendix A.

Proposition 1 implies that players, when they use EKF fictitious play, learn their opponent's strategy and eventually they will choose the action that will maximise their reward base on their estimation. Nevertheless there are cases where players may change their action simultaneously and become trapped in a cycle instead of converging in a pure Nash equilibrium. As an example we consider the game that is depicted in Table 2. This is a simple coordination game with two pure Nash equilibria the joint actions (U, L) and (D, R). In the case were the two players start from joint action (U, R) or (D, L) and they always change their action simultaneously then they will never reach one of the two pure Nash equilibria of the game.

Proposition 2. When the players of a game Γ use EKF fictitious play process to choose their actions, then with high probability they will not change their action simultaneously infinitely often.

Proof. The proof of Proposition 2 is on Appendix B.

Based on Proposition 1 and 2 we can infer the following propositions and theorems.

Proposition 3. (a) If s is a pure Nash equilibrium of a game Γ, and s is played at date t in the process of EKF fictitious play, s will be played at all subsequent dates. That is, pure Nash equilibria are absorbing for the process of EKF fictitious play.

(b) Any pure strategy steady state of EKF fictitious play must be a Nash equilibrium.

Proof. Consider the case where players beliefs $\hat{\sigma}_t$, are such that their optimal choices correspond to a pure Nash equilibrium $\hat{s}$. In EKF fictitious play process players' beliefs are formed identically and independently for each opponent based on equation (7). By Proposition 1 we know that players' estimations about their opponents' propensities and therefore their strategies will increase for the actions that are included in $\hat{s}$. Thus the best response to their beliefs $\hat{\sigma}_{t+1}$ will be again $\hat{s}$ and since $\hat{s}$ is a Nash equilibrium they will not deviate from it. Conversely, if a player remains at a pure strategy profile, then eventually the assessments will become concentrated at that profile, because of Proposition 1 and so if the profile is not a Nash equilibrium, one of the players would eventually want to deviate.

Proposition 4. Under EKF fictitious play, if the beliefs over each player's choices converge, the strategy profile corresponding to the product of these distributions is a Nash equilibrium.

Proof. Suppose that the beliefs of the players at time t, σ_t, converges to some profile $\hat{\sigma}$. If $\hat{\sigma}$ were not a Nash equilibrium, some player would eventually want to deviate and the beliefs would also deviate since based on Proposition 1 players eventually learn their opponents actions.

Based on the propositions (1-4) we can show that EKF fictitious play converges to the the Nash equilibrium of games with a better reply path. A game with a better reply path can be represented as a graph were its edges are the join actions of the game s and there is a vertex that connects s with s' iff only one player i can increasing his payoff by changing his action [Young, 2005]. Potential games have a better reply path.

Theorem 5. The EKF fictitious play process converges to the Nash equilibrium in games with a better reply path.

Proof. If the initial beliefs of the players are such that their initial joint action s_0 is a Nash equilibrium, from Proposition 3 and equation (7), we know that they will play the joint action which is a Nash equilibrium for the rest of the game.

Moreover in the case of the initial beliefs of the players are such that their initial joint action s_0 is not a Nash equilibrium based on Proposition 1 and Proposition 2 after a finite number of iterations because the game has a better reply path the only player that can improve his payoff by changing his actions will choose a new action which will result in a new joint action s. If this action is not the a Nash equilibrium then again after finite number of iterations the player who can improve his payoff will change action and a new joint action s' will be played. Thus after the search of the vertices of a finite graph, and thus after a finite number of iterations, players will choose a joint action which is a Nash equilibrium. the player that we know that after a finite number of iterations t_1 with high probability players at least one player will not change

his action simultaneously with the others, thus for a Player i, $s^i_{t1-1} = s^i_{t1}$. If the new joint action s_{t1} is not a Nash equilibrium, then at least one of the other players will deviate. Based on Proposition 2 and the fact that players estimate opponents' strategies independently at least one of them who will not change his action simultaneously with the others after $t2$ iterations which will result to a new joint action s_{t2} that will improve the current utility. Eventually after a finite number of time steps, T, the process will end up in a pure Nash equilibrium. The maximum number of iterations that is needed is the cardinality of the joint action set multiplied with the total number of iterations that is needed in order not to have simultaneous changes, $\binom{n}{2}(t1 + t2 + t3 + \ldots + T)$

5. THE PERFORMANCE MODEL

The controllers that we associate with each robot will be assumed to be able to carry out some dynamic tasks. We will consider $\mathcal{I}$ independent control systems, each of them associated with a robot, with the following dynamics:

$$\dot{x}_i = f_i(x_i, u_i) \tag{8}$$
$$y = g_i(x_i) \tag{9}$$

where x_i is the state of the system and u_i the control input of the system. A general cost function then can be expressed as $L_i(x_i, u_i) = J_i(x_i, u_i) + E_i(x_i)$ where $J_i(x_i, u_i)$ and $E_i(x_i, x_{-i})$ are the maneuverability and environmental costs respectively and can be defined as:

$$J_i(x_i^T, u_i) = c_m \int_0^T ||u_i(s)||ds \tag{10}$$

The states can be angular lateral possitions and their velocities and u_i can be actuator forces.

$$E_i(x_i) = c_e \sum_{j=1}^{k} \theta_i^j(x_i, x_{-i}) \tag{11}$$

where c_m and c_e are constants, u_s is the control input in order to move from from the initial state x_i^0 towards the terminal state x_i^T and $\theta_i^j(\cdot)$ are the environmental costs that depend on the final state of the agent x_i^T and all agents but i final state x_{-i}^T. In order to choose the actions that provide the best performance we should solve the following maximisation problem :

$$\begin{aligned} &\min \quad -L_i(x_i, u_i) \\ &s.t \\ &x_i^T \in X_i \end{aligned} \tag{12}$$

where X_i is the set of all possible final states for agent i.

In order to create the performance model of our controller we need to define a utility function. Utility functions have been used as metrics of the robots' coordination efficacy in applications,such as [Parker, 1998, Zlot et al., 2002, Timofeev et al., 1999, Botelho and Alami, 1999, Tsalatsanis et al., 2009]. Since the players of the game will maximise their expected reward the utility function can be seen as the negative of a cost function. Therefore we can include in the utility any maneuverability costs of the robots and environmental elements of the robots' tasks and the constraints that might arise from specific tasks. In the maneuverability part of the cost function we will take into account the spatial characteristics of the problem like the cost to the robot to move towards to a specific position. The environmental part of the cost function takes into account costs that arose from the nature of the coordination problem and can include the also the quality of the sensors of a robot, the aptness of the robots to perform specific tasks, etc.

The following case study serves as an example of the relationship between game theory and the control applications. We consider a team of robots who should coordinate in order to identify possible threats in an area A. We assume that in $\mathcal{N}$ regions of A there are some hazardous items. These items have different attributes of the following categories: flammable, chemical, radioactive and security sensitive. In each region there are items that belong up to two different categories, i.e. in the same region n we cannot place flammable, radioactive and chemical objects. Each of the $\mathcal{I}$ robots of the team is equipped with sensors that have different attributes and capabilities in order to identify different threats. Each robot i then can be equipped with up to four of the following sensors: fire detector, chemical detector, Geiger counter and vision system. Therefore robots with a fire detector should patrol areas with flammable objects, robots with Geiger should patrol areas with radioactive material etc.

We will use cooperative robot teams, robots who have different sensors and capabilities, but it is possible to use a similar controller in swarms of robots, i.e. robot teams that have identical specifications. The differences between the robots can be expressed in terms of their endurance in a specific environment, the efficiency with which they accomplish a specific task and the presence of the correct sensor to identify a specific threat. In order to quantify the efficacy of a robot in a specific task we use two fuzzy variables: endurance and efficiency. The values of endurance are short, fair, long and values of efficiency are low, medium and high. If a robot is not equipped with a specific sensor then its efficiency to detect an event is zero. Each robot, using a controller like the one that is depicted in Figure 1, should then coordinate with the other robots in order to efficiently choose a region to patrol.

The robots should coordinate and choose a region which they will patrol based on the possible threats that should be detected in each area, their sensors' specifications and the actions of the other robots. Each robot can choose only one region to patrol, but many robots can choose the same area. We will denote the efficacy of a robot i to sense a threat k in area n as $\mathcal{E}_{ink}$.

Each object in A is of different significance and therefore each region has a different value depending on the objects that are stored there which can be incorporated in $E_i(\cdot)$. In a game theoretic terminology we can define a potential game Γ with $\mathcal{I}$ players which have $\mathcal{N}$ available actions. The utility function that is produsedin a region n can be defined as:

$$r_n(s) = \sum_{\forall i, s^i = n} r_i^m(s^i) + \sum_{\forall s^i = n, n \in \mathcal{N}} r_i^e(s^i, s^{-i}).$$

where $r_i^m(s^i)$, is a function that depends on the initial position of the robot, the region n, $n \in \mathcal{N}$. In terms of the cost functions of the previous section ($r_i^m(s^i) = -J_i(x_i^T, u^i)$) and the final state x_i^T depends on the region

	Weak	Fair	Strong
Weak	1,1	0,0	0,0
Fair	0,0	1,1	0,0
Strong	0,0	0,0	1,1

Table 3. Symmetric game

which robot i chooses. The environmental cost of this choice,$r_i^e(s^i) = -E_i(x_i^T)$ depends on the actions, final states, of all the robots and is a function that depends on the sensors of the robots, the value an the types of the objects in region n that is represented as final the state x_i^T. The global utility that the robots will share is defined as:

$$r_g(s) = \sum_{n=1}^{n=\mathcal{N}} r_n(s) \tag{13}$$

6. SIMULATIONS

This section contains the results of two simulation scenarios we tested the performance of the proposed learning algorithm on. We set, in both games, $\Xi_t = (\tilde{\xi} + \epsilon)I$ and $Z_t = \tilde{\zeta}I$, where I is the identity matrix, $\tilde{\xi} = 0.1$, $\tilde{\zeta} = 1/t$ and ϵ is an arbitrary small Gaussian random number. This set of parameters of the EKF fictitious play algorithm were empirically found to provide good tracking of opponents' strategies.

We initially tested the performance of the proposed algorithm in the symmetric game that is presented in Table 3.

An example of how a symmetric game can be ensued from a realistic application comes from the area of material handling robots tasks. Consider the case where two robots should coordinate in order to move some objects to a desired destination. The robots can either push or pull the objects depending to the direction of the destination of the object. Moreover each robot can apply different forces to the objects. The amount of the force that a robot applies to an object can be represented as a fuzzy variable that can take the values: Weak, Fair and Strong. This set up can lead to either games where there is a dominant joint action which will gain the maximum reward, or to symmetric games like the games that are described in Tables 2 and 3. The controller which is depicted in Figure 1 can be used when robots have to take such decisions. We compared the quality of our robots' decisions with the max-sum algorithm [Farinelli et al., 2013]. Because of the symmetry in the utility function of the fictitious play and themax-sum algorithms, they do not always converge in one of the joint actions that maximise the global reward. In our simulations the two robots had to negotiate using one of the algorithms for 50 negotiation steps and then choose a joint action. Figure 2 depicts the percentage of the replications where the algorithms converged to one of the three optimal solutions. We can observe that, when we use EKF fictitious play, then robots always choose an optimal joint action, if the other two algorithms are used the robots choose an optimal joint action less than in 50% of the replications.

An advantage of the game-theoretic algorithms, and the proposed method in particular over the message passing algorithms is their communication cost. Even in the simple

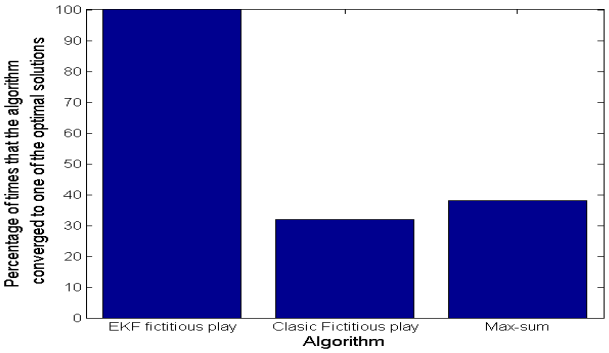

Fig. 2. Percentage of the times that each algorithm converged to one of the three optimum joint actions.

scenario where two robots have three available actions, when the robots use the message passing algorithm as a coordination mechanism they need to exchange more messages. In particular the robots that use fictitious play based algorithms should only share with the other robots only their action, in the simple case of the symmetric game each robot should exchange with all the other 50 messages in total. Each of these messages will consist of an integer number that will represent the action that he will choose. On the other hand when the max-sum algorithm is used as a coordination mechanism then each robot should exchange 100 messages and each of these messages consists of 3 real numbers. Even in the simple symmetric case the communication needs of the message passing algorithm are greater than the game-theoretic one, therefore in more complicated games with a large number of players and actions, message passing algorithms can be intractable because their communication cost increases exponentially [Farinelli et al., 2013].

We also examined the performance of EKF fictitious play in the task allocation scenario we described in Section 2. In particular in an area A we assume that there are ten regions, $\mathcal{N} = 10$, with hazardous objects. Each region contains two objects with different attributes. In A there are thirty robots, $\mathcal{I} = 30$, who should be allocated in one of the ten regions to patrol them. Each robot can move towards a region n with a velocity $\mathcal{V}_{in}$. In our case study we assume that the velocity of a robot i, towards a region n can be either slow or medium or fast. Therefore the time that a robot needs to reach a region n, T_{in}, depends both in the distance between the robot and the region and the velocity that the robot will choose to move towards n as well. The global reward that is shared among the agents is defined as:

$$\begin{aligned} r_g(s) = &-\sum_{n=1}^{\mathcal{N}} \sum_{i:s^i=n} c_1 T_{in} + c_2 \mathcal{V}_{in} \\ &+ \sum_{n=1}^{\mathcal{N}} \sum_{k \in K_n} V_{nk}(1 - \prod_{i:s^i=n} (1 - \mathcal{E}_{ink})) \end{aligned} \tag{14}$$

where $c_1 = 1$, $c_2 = 1/6$ and $\mathcal{E}_{ink}$ is a metric of robot i's efficacy to patrol a region n based on its endurance and efficiency to detect a threat k. It can also be seen as the accuracy of robot i's measurement in region n for threat

		Endurance		
		Short	Fair	Long
Efficiency	Low	0.2	0.3	0.5
	Medium	0.3	0.5	07
	High	0.5	0.7	0.9

Table 4. Efficacy that a robot efficiently patrol the area for a specific threat.

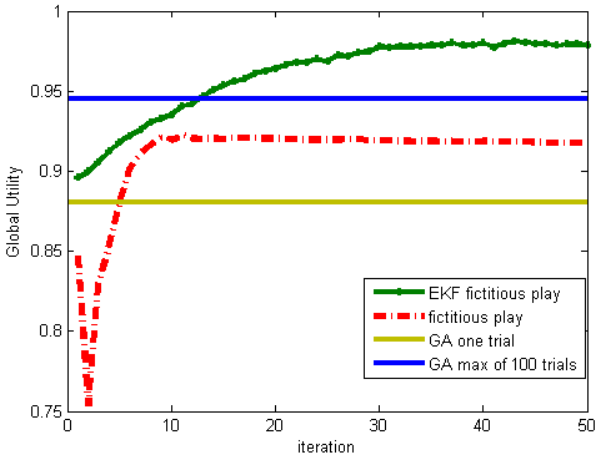

Fig. 3. Robot team task allocation performance

k. We define $\mathcal{E}_{ink}$ as a function of robot i's endurance and efficiency as it is presented in Table 4.

In this scenario we did not use the max-sum algorithm in our comparisons because of the scale of our problem. In particular robots should search over to 10^{30} possible joint actions which make the max-sum algorithm intractable for this scenario. Instead we compared the efficacy of the proposed controller with the allocation quality of a centralised approach that uses genetic algorithms, thus a central decision maker chooses in advance the room that each robot will patrol and the robot completes the pre-optimised allocation task. In our comparison we used the genetic algorithm function of Matlab's optimisation toolbox. Since genetic algorithms are stochastic processes, in each repetition of the same replication of the task allocation problem it will converge to a different solution Thus we used two instances of the genetic algorithm, in the former one we used a single repetition of the genetic algorithm per replication of our problem, in the latter one for each replication of the task allocation problem we chose the allocation of the maximum utility among 100 runs of the genetic algorithm.

In our simulations we used 100 replications of the task allocation problem. In order to be able to average across the 100 replications. we normalise the utility by the maximum global utility that is observed in a replication of the simulation scenario by the learning algorithms we used.

As it is shown in Figure 3, the genetic algorithm can be easily trapped in a local maximum point and therefore it is important to use more than one repetition of the algorithm. Moreover when EKF fictitious play is used in the optimisation block of our controller the allocation the robots can patrol the warehouse with bigger efficacy than the instances we used genetic algorithm and fictitious play.

7. CONCLUSIONS

A cooperative control methodology for a team of robots has been proposed using a game theoretical approach. The coordination of the robots has been cast as a potential game which has been used as a performance model of this distributed control problem. A learning algorithm has been used by each robot based on fictitious play and EKF as an implicit coordination mechanism between the robots. It has been shown that the collective learning algorithm converges to the Nash equilibrium of potential games. Simulations have been used to illustrate performance.

Our ongoing work includes further comparison with state of the art predictive control algorithms. Moreover we are planning to use the proposed controller design methodology on a team of flying and ground based robots in our laboratory followed by implementation by our industrial partners.

Appendix A. PROOF OF PROPOSITION 1

In the case where players have only 2 available actions an abstract representation of H and P are $\left(\begin{smallmatrix} a & -a \\ -a & a \end{smallmatrix}\right)$ and $\left(\begin{smallmatrix} b & k \\ k & b \end{smallmatrix}\right)$ respectively. The Kalman gain is estimated up to a multiplication constant

$$K_1 \sim \begin{pmatrix} c & -c \\ -d & d \end{pmatrix}$$

where $c = P_t^-[1,1] - P_t^-[1,2]$ and $d = P_t^-[2,2] - P_t^-[1,2]$. Then m increases for the recently observed action and decreases otherwise.

In the case where there are more than 2 available actions, when $1/t << 1$ then the covariance matrix $S_t \simeq H(m_t^-)P_t^- H^T(m_t^-)$ and then $K_t \simeq H(m_t^-)$. From the definition of $H(m_t^-$ we know that its diagonal elements are positive and all the of diagonal elements are negative. In particular we can write :

$$H(m_t^-)[i,i] = \sum_{j \in S/i} \sigma_{t-1}(s^i)\sigma_{t-1}(s^j)$$

$$H(m_t^-)[i,j] = -\sigma_{t-1}(s^i)\sigma_{t-1}(s^j).$$

Suppose that action s^j is played from Player i's opponent then the update of $m_t[j]$ is the following:

$$m_t[j] = m_t^- + H(m_t^-)[j,\cdot]y.$$

The only positive element of y is $y[j]$. The multiplication of $H(m_t^-)[j,\cdot]$ and y is the sum of $\mathbb{I}$ positive coefficients and therefore the value of $m_t[j]$ will be increased. In order to complete the proof we should show that $\Delta[j] > \Delta[i] \forall i \in S/i$, where $\Delta[\tilde{i}] = m_t[\tilde{i}] - m_{t-1}[\tilde{i}], \tilde{i} \in S$. For simplicity of notation for the rest of the proof we will write $H[i,j]$ instead of $H(m_t^-)[i,j]$ and $\sigma(i)$ instead of $\sigma_{t-1}(s^i)$.

$$\begin{aligned}\Delta[i] &= H[i,\cdot]y \\ &= H[i,1]\sigma(1) + H[i,2]\sigma(2) + \ldots + \\ &\quad H[i,i-1]\sigma(i-1) - H[i,i]\sigma(i) + \\ &\quad H[i,i+1]\sigma[i+1] + \ldots + \\ &\quad H[i,j-1]\sigma(j-1) - H[i,j](1-\sigma(j)) + \\ &\quad H[i,j+1]\sigma(j+1) + \ldots + H[i,\mathcal{I}]\sigma(\mathcal{I})\end{aligned} \tag{A.1}$$

$$\begin{aligned}\Delta[j] &= H[j,\cdot]y \\ &= H[j,1]\sigma(1) + H[j,2]\sigma(2) + \ldots + \\ &\quad H[j,j](1-\sigma(j)) + \ldots + H[i,\mathcal{I}]\sigma(\mathcal{I})\end{aligned} \tag{A.2}$$

$$\begin{aligned}\Delta[i] - \Delta[j] = {} & \sigma(1)(H[i,1] - H[j,1]) + \ldots + \\ & \sigma(i-1)(H[i,i-1] - H[j,i-1]) - \\ & \sigma(i)(H[i,i] + H[j,i]) + \\ & \sigma(i+1)(H[i,i+1] - H[i,i+1]) + \ldots + \\ & \sigma(j-1)(H[i,j-1] - H[j,j-1]) - \\ & (1-\sigma(j))(H[[i,j] + H[j,j]]) + \\ & \sigma(j+1)(H[i,j+1] - H[j,j+1]) + \ldots + \\ & \sigma(\mathcal{I})(H[i,\mathcal{I}] - H[j,\mathcal{I}])\end{aligned} \tag{A.3}$$

If we substitute $H[\cdot,\cdot]$ with its equivalent and we can write $\Delta[i] - \Delta[j]$ as:

$$\begin{aligned}\Delta[i] - \Delta[j] = {} & (\sigma(1))^2(\sigma(i) - \sigma(j)) + \ldots + \\ & (\sigma(i-1))^2(\sigma(i) - \sigma(j)) - \\ & (\sigma(i))^2((\sum_{\tilde{j}\in S/i} \sigma(\tilde{j})) + \sigma(j)) + \\ & (\sigma(i+1))^2(\sigma(i) - \sigma(j)) + \ldots + \\ & (\sigma(j-1))^2(\sigma(i) - \sigma(j)) - \\ & (1-\sigma(j))\sigma(j)((\sum_{\tilde{j}\in S/j} \sigma(\tilde{j})) + \sigma(i)) + \\ & (\sigma(j+1))^2(\sigma(i) - \sigma(j)) + \ldots + \\ & (\sigma(\mathcal{I}))^2(\sigma(i) - \sigma(j))\end{aligned} \tag{A.4}$$

solving the inequality $\Delta[i] - \Delta[j] < 0$ we obtain:

$$\begin{aligned}&\Delta[i] - \Delta[j] < 0 \Leftrightarrow \\ &(\sigma(i) - \sigma(j))((\sum_{\tilde{j}\in S/\{i,j\}} (\sigma(\tilde{j})^2))) < \\ &((\sigma(i))^2((\sum_{\tilde{j}\in S/i} \sigma(\tilde{j})) + \sigma(j)) + \\ &(1-\sigma(j))(\sigma(j))((\sum_{\tilde{j}\in S/j} \sigma(\tilde{j})) + \sigma(i)))\end{aligned} \tag{A.5}$$

In the case where $\sigma(i) < \sigma(j)$ the inequality is satisfied always because the left hand side of the inequality is always negative and the right hand side is always positive. In the case where $\sigma(i) > \sigma(j)$ inequality (A.5) we will show by contradiction that (A.5) is satisfied $\forall i, i \neq j$. Therefore we assume that:

$$\begin{aligned}&\Delta[i] - \Delta[j] \geq 0 \Leftrightarrow \\ &(\sigma(i) - \sigma(j))((\sum_{\tilde{j}\in S/\{i,j\}} (\sigma(\tilde{j})^2))) \geq \\ &((\sigma(i))^2 + (1-\sigma(j))(\sigma(j)))((\sum_{\tilde{j}\in S/i} \sigma(\tilde{j})) + \sigma(j))\end{aligned} \tag{A.6}$$

Since $((\sum_{\tilde{j}\in S/\{i,j\}}(\sigma(\tilde{j})^2))) < ((\sum_{\tilde{j}\in S/i}\sigma(\tilde{j})) + \sigma(j))$ in order to complete the proof we only need to show that:

$$\begin{aligned}\sigma(i) - \sigma(j) &\geq (\sigma(i))^2 + (1-\sigma(j))\sigma(j) \\ \sigma(i) - (\sigma(i))^2 &\geq 2\sigma(j) - (\sigma(j))^2 \\ \sigma(i)(1-\sigma(i)) &\geq \sigma(j)(2-\sigma(j))\end{aligned} \tag{A.7}$$

Inequality (A.7) will be satisfied if the following inequality is satisfied:

$$(1-\sigma(i)) \geq \frac{\sigma(j)}{\sigma(i)}(2-\sigma(j)) \tag{A.8}$$

Inequality (A.8) will be satisfied if

$$\begin{aligned}(1-\sigma(i)) &\geq (2-\sigma(j)) \\ \sigma(j) - \sigma(i) &\geq 1\end{aligned} \tag{A.9}$$

since $\sigma(i) > \sigma(j)$, (A.9), is false $\forall i$ and thus by contradiction $\Delta[i] - \Delta[j] < 0$, which completes the proof.

Appendix B. PROOF OF PROPOSITION 2

Similarly to the proof of Proposition 1 we consider only one opponent and a game with more than one pure Nash equilibria. We assume that a joint action $s = (s^1(j'), s^2(j))$ is played which is not a Nash equilibrium. Since players use EKF fictitious play because of Proposition 1 they will eventually change their action to $\tilde{s} = (s^1(\tilde{j'}]), s^2(\tilde{j}))$ which will be the best response to action $s^2(j)$ and $s^1(j')$ for players 1 and 2 respectively. But if this change is simultaneous their is no guarantee that the resulted joint action $\tilde{s}$ will increase the expected reward of the players. We will show that with high probability the two players will not change actions simultaneously infinitely often.

Without loss of generality we assume that at time t Player 2 change his action from $s^2(j)$ to $s^2(\tilde{j})$. We want to show that the probability that Player 1 will change his action to $s^1(\tilde{j'})$ with probability less than 1. Player 1 will change his action if he thinks that Player 2 will play action $s^2(\tilde{j})$ with high probability such that his utility will be maximised when he plays action $s^1(\tilde{j'})$. Therefore Player 1 will change his action to $s^1(\tilde{j'})$ if $\sigma^2(s^2(j)) > \lambda$, where $\sigma^2(\cdot)$ is the estimation of his Player 1 about Player 2's strategy. Thus we want to show

$$\begin{aligned}&p(\sigma^2 > \lambda) < 1 \Leftrightarrow \\ &p(\frac{\exp(m_{t-1}[j])}{\sum_{j''\in S^2}\exp(m_{t-1}[j''])} > \lambda) < 1 \Leftrightarrow \\ &p(m_{t-1}[j] > \ln\frac{\lambda}{1-\lambda} + \sum_{j''\in S^2/j}\exp(m_{t-1}[j''])) < 1 \Leftrightarrow \\ &p(m_{t-2}[j] + (K_{t-1}y_{t-1})[j] > \ldots \\ &\quad \ln\frac{\lambda}{1-\lambda} + \sum_{j''\in S^2/j}\exp(m_{t-1}[j''])) < 1\end{aligned} \tag{B.1}$$

we can expand $K_{t-1}y_{t-1}[j]$ as follows

$$\begin{aligned}(K_{t-1}y_{t-1})[j] &= ((P_{t-2} + (\tilde{\xi} + \epsilon)I)HS^{-1}y_{t-1})[j] \\ &= (P_{t-2}HS^{-1}y_{t-1})[j] + \\ &\quad ((\tilde{\xi} + \epsilon)HS^{-1}y_{t-1})[j]\end{aligned} \tag{B.2}$$

If we substitute $K_{t-1}y_{t-1}[j]$ in (B.1) with its equivalent we obtain the following inequality:

$$
\begin{aligned}
&p(m_{t-2}[j] + (P_{t-2}HS^{-1}y_{t-1})[j] + ((\tilde{\xi}+\epsilon)HS^{-1}y_{t-1})[j] > \dots \\
&\quad \ln\frac{\lambda}{1-\lambda} + \sum_{j''\in S^2/j} \exp(m_{t-1}[j''])) < 1 \Leftrightarrow \\
&p(((\tilde{\xi}+\epsilon)HS^{-1}y_{t-1})[j] > -m_{t-2}[j] - (P_{t-2}HS^{-1}y_{t-1})[j] \dots \\
&\quad \ln\frac{\lambda}{1-\lambda} + \sum_{j''\in S^2/j} \exp(m_{t-1}[j''])) < 1 \Leftrightarrow \\
&p(\epsilon > C) < 1
\end{aligned}
\tag{B.3}
$$

where C is defined as:

$$
\begin{aligned}
C = \frac{\ln\frac{\lambda}{1-\lambda}}{(HS^{-1}y_{t-1})[j]} + \\
\frac{\sum_{j''\in S^2/j}\exp(m_{t-1}[j''])}{(HS^{-1}y_{t-1})[j]} - \\
\frac{-m_{t-2}[j] - (P_{t-2}HS^{-1}y_{t-1})[j]}{(HS^{-1}y_{t-1})[j]} - \\
\frac{q(HS^{-1}y_{t-1})[j]}{(HS^{-1}y_{t-1})[j]}
\end{aligned}
\tag{B.4}
$$

Inequality (B.3) is always satisfied since ϵ is a Gaussian random variable. To conclude the proof we define χ_t as the event that both players change their action at time t simultaneously, and assume that the two players have change their actions simultaneously at the following iterations $t_1, t_2, \dots, t_T$, then the probability that they will also change their action simultaneously at time t_{T+1}, $P(\chi_{t_1}, \chi_{t_2}, \dots, \chi_{t_T}, \chi_{t_{T+1}})$ is almost zero for large but finite T.

REFERENCES

Gürdal Arslan, Jason R Marden, and Jeff S Shamma. Autonomous vehicle-target assignment: A game-theoretical formulation. *Journal of Dynamic Systems, Measurement, and Control*, 129(5):584–596, 2007.

D. Bauso, L. Giarre, and R. Pesenti. Mechanism design for optimal consensus problems. In *Decision and Control, 2006 45th IEEE Conference on*, pages 3381–3386, 2006.

S.C. Botelho and R. Alami. M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 2, pages 1234–1239 vol.2, 1999.

G. W. Brown. Iterative solutions of games by fictitious play. In T. C. Koopmans, editor, *Activity Analysis of Production and Allocation*, pages 374–376. Wiley, 1951.

A. C. Chapman, A. Rogers, N. R Jennings, and D. S Leslie. A unifying framework for iterative approximate best-response algorithms for distributed constraint optimization problems. *Knowledge Engineering Review*, 26(4): 411–444, 2011.

C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou. The complexity of computing a nash equilibrium. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 71–78, 2006.

J. Evans and B. Krishnamurthy. Helpmate, the trackless robotic courier: A perspective on the development of a commercial autonomous mobile robot. In *Autonomous Robotic Systems*, volume 236, pages 182–210. Springer London, 1998.

A Farinelli, A Rogers, and NR Jennings. Agent-based decentralised coordination for sensor networks using the max-sum algorithm. *Autonomous Agents and Multi-Agent Systems*, pages 1–44, 2013.

D. Fudenberg and D. Levine. *The theory of Learning in Games*. The MIT Press, 1998.

N. Lincoln, S. Veres, L. Dennis, M. Fisher, and A. Lisitsa. Autonomous asteroid exploration by rational agents. *Computational Intelligence Magazine, IEEE*, 8(4):25–38, 2013.

R. Madhavan, K. Fregene, and L. Parker. Distributed cooperative outdoor multirobot localization and mapping. *Autonomous Robots*, 17(1):23–39, 2004.

J. Nash. Equilibrium points in n-person games. In *Proceedings of the National Academy of Science, USA*, volume 36, pages 48–49, 1950.

L.E. Parker. Alliance: an architecture for fault tolerant multirobot cooperation. *Robotics and Automation, IEEE Transactions on*, 14(2):220–240, 1998.

E. Semsar-Kazerooni and K. Khorasani. Optimal consensus algorithms for cooperative team of agents subject to partial information. *Automatica*, 44(11):2766 – 2777, 2008.

E. Semsar-Kazerooni and K. Khorasani. Multi-agent team cooperation: A game theory approach. *Automatica*, 45 (10):2205 – 2213, 2009.

R. Simmons, D. Apfelbaum, W. Burgard, M. Fox, D. an Moors, S. Thrun, and H. Younes. Coordination for multi-robot exploration and mapping. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Austin, TX, 2000. AAAI.

M. Smyrnakis and D. S. Leslie. Dynamic Opponent Modelling in Fictitious Play. *The Computer Journal*, 2010.

A.V. Timofeev, F.A. Kolushev, and A.A. Bogdanov. Hybrid algorithms of multi-agent control of mobile robots. In *Neural Networks, 1999. IJCNN '99. International Joint Conference on*, volume 6, pages 4115–4118, 1999.

A. Tsalatsanis, A. Yalcin, and K.P. Valavanis. Optimized task allocation in cooperative robot teams. In *Control and Automation, 2009. MED '09. 17th Mediterranean Conference on*, pages 270–275, 2009.

D. Wolpert and K. Tumer. *.A survey of collectives.*, chapter Collectives and the Design of Complex Systems., pages 1–42. Springer, 2004.

H. P. Young. *Strategic Learning and Its Limits*. Oxford University Press, 2005.

Y. Zhang, M. Schervish, E. Acar, and H. Choset. Probabilistic methods for robotic landmine search. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '01)*, pages 1525 – 1532, 2001.

R. Zlot, A. Stentz, M B. Dias, and S. Thayer. Multi-robot exploration controlled by a market economy. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 3, 2002.