Multi-agent learning using Fictitious Play and Extended Kalman Filter

Michalis Smyrnakis\textsuperscript{*1}

\textsuperscript{1}Complex Systems and Statistical Physics Group, School of Physics and Astronomy, University of Manchester, UK

December 11, 2013

Abstract

Decentralised optimisation tasks are important components of multi-agent systems. These tasks can be interpreted as n-player potential games: therefore game-theoretic learning algorithms can be used to solve decentralised optimisation tasks. Fictitious play is the canonical example of these algorithms. Nevertheless fictitious play implicitly assumes that players have stationary strategies. We present a novel variant of fictitious play where players predict their opponents’ strategies using Extended Kalman filters and use their predictions to update their strategies.

We show that in 2 by 2 games with at least one pure Nash equilibrium and in potential games where players have two available actions, the proposed algorithm converges to the pure Nash equilibrium. The performance of the proposed algorithm was empirically tested, in two strategic form games and an ad-hoc sensor network surveillance problem. The proposed algorithm performs better than the classic fictitious play algorithm in these games and therefore improves the performance of game-theoretical learning in decentralised optimisation.

Keywords: Multi-agent learning, game theory, fictitious play, decentralised optimisation, learning in games, Extended Kalman filter.

1 Introduction

Recent advance in technology render decentralised optimisation a crucial component of many applications of multi agent systems and decentralised control. Sensor networks (Kho et al. 2009), traffic control (van Leeuwen et al. 2002) and scheduling problems (Stranjak et al. 2008) are some of the tasks where decentralised optimisation can be used. These tasks share common characteristics such as large scale, high computational complexity and communication constraints that make a centralised solution intractable. It is well known that many decentralised optimisation tasks can be cast as potential games (Wolpert and Turner 1999, Arslan et al. 2006), and the search of an optimal solution

\textsuperscript{*}michalis.smyrnakis@manchester.ac.uk
can be seen as the task of finding Nash equilibria in a game. Thus it is feasible to use iterative learning algorithms from game-theoretic literature to solve decentralised optimisation problems.

A game theoretic learning algorithm with proof of convergence in certain kinds of games is fictitious play (Fudenberg and Levine, 1998; Monderer and Shapley, 1996). It is a learning process where players choose an action that maximises their expected rewards according to the beliefs they maintain about their opponents' strategies. The players update their beliefs about their opponents' strategies after observing their actions. Even though fictitious play converges to Nash equilibrium, this convergence can be very slow. This is because it implicitly assumes that other players use a fixed strategy in the whole game. Smyrnakis and Leslie (2010) addressed this problem by representing the fictitious play process as a state space model and by using particle filters to predict opponents' strategies. The drawback of this approach is the computational cost of the particle filters that render difficult the application of this method in real time applications.

The alternative that we propose in this article is to use instead of particle filters, extended Kalman filters (EKF) to predict opponents' strategies. Therefore the proposed algorithm has smaller computational cost than the particle filter variant of fictitious play algorithm that proposed by Smyrnakis and Leslie (2010). We show that the EKF fictitious play algorithm converges to a pure Nash equilibrium, in 2 by 2 games with at least one pure Nash equilibrium and in potential games where players have two available actions. We also empirically observe, in a range of games, that the proposed algorithm needs less iterations than the classic fictitious play to converge to a solution. Moreover in our simulations, the proposed algorithm converged to a solution with higher reward than the classic fictitious play algorithm.

The remainder of this paper is organised as follows. We start with a brief description of game theory, fictitious play and extended Kalman filters. Section 2 introduces the proposed algorithm that combines fictitious play and extended Kalman filters. The convergence results we obtained are presented in Section 3. In Section 4, we propose some indicative values for the EKF algorithm parameters. Section 5 presents the simulation results of EKF fictitious play in a 2×2 coordination game, a three player climbing hill game and an ad-hoc sensor network surveillance problem. In the final section we present our conclusions.

2 Background

In this section we introduce some definition from game theory that we will use in the rest of this article and the relation between potential games and decentralised optimisation. We also briefly present the classic fictitious play algorithm and the extended Kalman filter algorithm.

2.1 Game theory definitions

We consider a game Γ with ℓ players, where each player $i, i = 1, 2, \ldots, ℓ$, choose his action, $s^i$, from a finite discrete set $S^i$. We then can define the joint action that is played in a game as the set product $S = \times_{i=1}^{ℓ} S^i$. Each Player $i$ receive a reward, $u^i$, after choosing an action. The reward is a map from the joint action
space to the real numbers, \( u^i : S \rightarrow R \). 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 \( i \), \( \sigma^i \), is an element of the set \( \Delta^i \), where \( \Delta^i \) is the set of all the probability distributions over the action space \( S^i \). The joint mixed strategy, \( \sigma \), is then an element of \( \Delta = \times_{i=1}^n \Delta^i \). Analogously to the joint actions we will write \( \sigma = (\sigma^i, \sigma^{-i}) \). In the special case where the players choose an action with probability one we will say that players choose their actions using pure strategies. The expected utility a player \( i \) will gain if he chooses a strategy \( \sigma^i \) (resp. \( s^i \)), when his opponents choose the joint strategy \( \sigma^{-i} \) is \( u^i(\sigma^i, \sigma^{-i}) \) (resp. \( u^i(s^i, \sigma^{-i}) \)).

A common decision rule in game theory is best response (BR). The best response is defined as the action that maximizes players’ expected utility given their opponents’ strategies. Thus for a specific opponents’ strategy \( \sigma^{-i} \) we evaluate the best response as:

\[
BR^i(\sigma^{-i}) = \arg\max_{s^i \in S^i} u^i(s^i, \sigma^{-i}) \tag{1}
\]

Nash (1950) showed that every game has at least one equilibrium, which is a fixed point of the best response correspondence, \( \sigma^i \in BR(\sigma^{-i}) \). Thus when a joint mixed strategy \( \hat{\sigma} \) is a Nash equilibrium then:

\[
u^i(\hat{s}^i, \hat{s}^{-i}) \geq u^i(s^i, \hat{s}^{-i}) \quad \text{for all } s^i \in S^i \tag{2}
\]

Equation 2 implies that if a 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 as pure strategy Nash equilibria. A pure equilibrium is strict if each player has a unique best response to his opponents actions.

### 2.2 Decentralised optimisation tasks as potential games

A class of games that are of particular interest in multi agent systems and decentralised optimisation tasks are potential games, because of their utility structure. In particular in order to be able to solve an optimisation task decentraly the local functions should have similar characteristics with the global function that we want to optimise. This suggests that an action which improves or reduces the utility of an individual should respectively increase or reduce the global utility. Potential games have this property, since the potential function (global function) depict the changes in the players’ payoffs (local functions) when they unilaterally change their actions. More formally we can write

\[
u^i(s^i, s^{-i}) - u^i(s^i, s^{-i}) = \phi(s^i, s^{-i}) - \phi(s^i, s^{-i})
\]

where \( \phi \) 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 \( s^i, s^i \in S^i \), where \( S^i \) and \( S^{-i} \) represent the set of all available actions for Player \( i \) and his opponents respectively. Moreover potential games has at least one pure Nash equilibrium, hence there is at least one joint action \( s \) where no player can increase their reward, therefore the potential function, through a unilateral deviation.
It is feasible to choose an appropriate form of the agents’ utility function in order for the global utility to act as a potential of the system. Wonderful life utility is a utility function that introduced by [Wolpert and Turner] (1999) and applied by [Arslan et al.] (2006) to formulate distributed optimisation tasks as potential games. Player $i$’s utility, when wonderful life utility is used, can be defined as the difference between the global utility $u_g$ and the utility of the system when a reference action is used as player’s $i$ action. More formally when player $i$ chooses an action $s^i$, we write

$$u^i(s^i) = u_g(s^i, s^{-i}) - u_g(s^i_0, s^{-i})$$

where $s^i_0$ denotes the reference action of player $i$. Hence the decentralised optimisation problem can be cast as a potential game and any algorithm that is proved to converge to a Nash equilibrium of a potential game, which is a local or the global optimum of the optimisation problem, will converge to a joint action from which no player can increase the global reward through unilateral deviation.

### 2.3 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 to his beliefs about his opponents’ joint mixed strategy $\sigma^{-i}$.

Initially each player has some prior beliefs about the strategy that each of his opponents uses to choose an action based on a weight function $\kappa_t$. The players, after each iteration, update the weight function and therefore their beliefs about their opponents’ strategies and play again the best response according to their beliefs. More formally in the beginning of a game Player $i$ maintains some arbitrary non-negative initial weight functions $\kappa^0_j$, $\forall j \in [1, l] \setminus \{i\}$, that are updated using the formula:

$$\kappa^t_j(s^j) = \kappa^{t-1}_j(s^j) + \mathcal{I}_{s^j = s^j}$$

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

The mixed strategy of opponent $j$ is estimated from the following formula:

$$\sigma^i_j(s^j) = \frac{\kappa^t_j(s^j)}{\sum_{s' \in S^j} \kappa^t_j(s')}.$$  \hspace{1cm} (3)

Player $i$ based on his beliefs about his opponents’ strategies, chooses the action which maximises his expected payoffs. When player $i$ uses equation (3) to update the beliefs about his opponents’ strategies he treats the environment of the game as stationary and implicitly assumes that the actions of the players are sampled from a fixed probability distribution. Therefore the recent observations have the same weight as the initial ones. This approach leads to poor adaptation when the other players choose to change their strategies.

### 2.4 Fictitious play as a state space model

We follow [Smyrnakis and Leslie] (2010) and we will represent fictitious play process as a state-space model. According to this state space model each player
has a propensity $Q^i_t(s^i)$ to play each of his available actions $s^i \in S^i$, and then he forms his strategy based on these propensities. Finally he chooses his actions based on his 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. We set $Q_0 \sim N(0, I)$ and recursively update the value of $Q_t$ according to the value of $Q_{t-1}$ as follows:

$$Q(s^i_t) = Q(s^i_{t-1}) + \eta_t$$

where $\eta_t \sim N(0, \chi^2 I)$. The action of a player then is related to his propensity by the following sigmoid equation for every $s^i \in S^i$

$$s^i = \frac{e^{Q^i_t(s^i_t)/\tau}}{\sum_{i \in S^i} e^{Q^i_t(s^i_t)/\tau}}.$$

Therefore players will assume that at every iteration $t$ their opponents have a different strategy $\sigma_t$.

### 2.5 Kalman filters and Extended Kalman filters

Our objective is to estimate player $i$’s opponent propensity and thus to estimate the marginal probability $p(Q_t, s_{1:t})$. This objective can be represented as a Hidden Markov Model (HMM). HMMs are used to predict the value of an unobserved variable $x_t$, the hidden state, using the observations of another variable $z_{1:t}$. There are two main assumptions in the HMM representation. The former one is that the probability of being at any state $x_t$ at time $t$ depends only at the state of time $t - 1$, $x_{t-1}$. The latter one is that an observation at time $t$ depends only on the current state $x_t$. One of the most common methods to estimate $p(x_{1:t}, z_{1:t})$ is Kalman filters and its variations. Kalman filter (Kalman et al., 1960) is based on two assumptions, the first is that the state variable is Gaussian. The second is that the observations are the result of a linear combination of the state variable. Hence Kalman filters can be used in cases which are represented as the following state space model:

$$x_t = Ax_{t-1} + \xi_{t-1} \text{ hidden layer}$$

$$y_t = Bx_t + \zeta_t \text{ observations}$$

where $\xi_t$ and $\zeta_t$ follow a zero mean normal distribution with covariance matrices $\Xi = q_t I$ and $\Sigma = r_t I$ respectively, and $A, B$ are linear transformation matrices. When the distribution of the state variable $x_t$ is Gaussian then $p(x_t | y_{1:t})$ is also a Gaussian distribution, since $y_t$ is a linear combination of $x_t$. Therefore it is enough to estimate its mean and variance to fully characterise $p(x_t | y_{1:t})$.

Nevertheless in the state space model we want to implement, the relation between Player $i$’s opponent propensity and his actions is not linear. Thus we should use a more general form of state space model such as:

$$x_t = f(x_{t-1}) + \xi_t$$

$$y_t = h(x_t) + \zeta_t \quad (4)$$
where $\xi_t$ and $\zeta_t$ are the hidden and observation state noise respectively, with zero mean and covariance matrices $\Xi = q_t I$ and $Z = r_t I$ respectively. The distribution of $p(x_t | y_{1:t})$ is not a Gaussian distribution because $f(\cdot)$ and $h(\cdot)$ are non-linear functions. A simple method to overcome this shortcoming is to use a first order Taylor expansion to approximate the distributions of the state space model in (4). In particular we let $x_t = m_{t-1} + \epsilon$, where $m_t$ denotes the mean of $x_t$ and $\epsilon \sim N(0, P)$. We can rewrite (4) as:

$$
\begin{align*}
x_t &= f(m_{t-1} + \epsilon) + w_{t-1} = f(m_{t-1}) + F_x(m_{t-1})\epsilon + \xi_{t-1} \\
y_t &= h(m_t + \epsilon) + \zeta_t = h(m_t) + H_z(m_t)\epsilon + \zeta_t
\end{align*}
$$

(5)

where $F_x(m_{t-1})$ and $H_z(m_t)$ is the Jacobian matrix of $f$ and $h$ evaluated at $m_{t-1}$ and $m_t$, respectively. If we use the transformations in (5) then $p(x_t | y_{1:t})$ is a Gaussian distribution.

Since $p(x_t | y_{1:t})$ is a Gaussian distribution to fully characterise it we need to evaluate its mean and variance. The EKF process (Jazwinski 1970; Grewal and Andrews 2011) estimates this mean and variance in two steps the prediction and the update step. In the prediction step at any iteration $t$ the distribution of the state variable is estimated based on all the observations until time $t-1$, $p(x_t | y_{1:t-1})$. The distribution of $p(x_t | y_{1:t-1})$ is Gaussian and we will denote its mean and variance as $m_t^-$ and $P_t^-$ respectively. During the update step the estimation of the prediction step is corrected in the light of the new observation at time $t$, so we estimate $p(x_t | y_{1:t})$. This is also a Gaussian distribution and we will denote its mean and variance as $m_t$ and $P_t$ respectively.

The prediction and the update steps of the EKF process (Jazwinski 1970; Grewal and Andrews 2011) to estimate the mean and the variance of $p(x_t | y_{1:t-1})$ and $p(x_t | y_{1:t})$ respectively are the following:

**Prediction Step**

$$
\begin{align*}
m_t^- &= f(m_{t-1}) \\
P_t^- &= F(m_{t-1})P_{t-1}F(m_{t-1}) + \Xi_{t-1}
\end{align*}
$$

where the $j, j'$ element of $F(m_t)$ is defined as

$$
[F(m_t^-)]_{j, j'} = \left. \frac{\partial f(x, r)}{\partial x_{j'}} \right|_{x=m_t^-, r=0}
$$

**Update Step**

$$
\begin{align*}
v_t &= z_t - h(m_t^-) \\
S_t &= H(m_t^-)P_t^-H^T(m_t^-) + Z \\
K_t &= P_t^-H^T(m_t^-)S_t^{-1} \\
m_t &= m_t^- + K_tv_t \\
P_t &= P_t^- - K_tS_tK_t^T
\end{align*}
$$

where $z_t$ is the observation vector (with 1 in the entry of the observed action and 0 everywhere else) and the $j, j'$ element of $H(m_t)$ is defined as:

$$
[H(m_t^-)]_{j, j'} = \left. \frac{\partial h(x, r)}{\partial x_{j'}} \right|_{x=m_t^-, r=0}
$$
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 drop all dependence on player $i$, and write $s_t$, $\sigma_t$ 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:

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

where $\xi_{t-1} \sim N(0, \Xi)$, is the noise of the state process and $\zeta_t$ is is the error of the observation state with zero mean and covariance matrix $\Sigma$, which occurs because we approximate a discrete process like best responses, equation (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:

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

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, equation (6), and best responses, equation (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:

$$v_t = z_t - h(m_t^{-})$$
$$S_t = H(m_t^{-})P_t^{-}H^T(m_t^{-}) + \Sigma$$
$$K_t = P_t^{-}H^T(m_t^{-})S_t^{-1}$$
$$m_t = m_t^{-} + K_tv_t$$
$$P_t = P_t^{-} - K_tS_tK_t^T$$

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

7
\[ [H(m_{-t})]_{j,j'} = \begin{cases} \frac{\sum_{j \neq j'} \exp(m_{-t}[j]) \exp(m_{-t}[j'])}{\left(\sum_{j} \exp(m_{-t}[j])\right)^2} & \text{if } j = j' \\ \frac{-\exp(m_{-t}[j]) \exp(m_{-t}[j'])}{\left(\sum_{j} \exp(m_{-t}[j])\right)^2} & \text{if } j \neq j' \end{cases} \]

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

---

At time \( t \)

1. Player \( i \) maintains some estimations about his opponents propensity up to time \( t-1 \), \( p(Q_{t-1}|s_1:t-1) \). Thus he has an estimation of the mean \( m_{t-1} \) and the covariance \( P_{t-1} \) of this distribution.

2. Then Player \( i \) is updating his estimations about his opponents propensities \( p(Q_t|s_1:t-1) \) using equations, \( m_t = m_{t-1}, P_t = P_{t-1} + W_{t-1} \).

3. Based on the weights of step 1 each player updates his beliefs about his opponents strategies using \( \sigma^j_t(s^t) = \exp\left(\frac{m_t-[j]}{\tau}\right) \sum_{j'} \exp\left(\frac{m_t-[j']}{\tau}\right) \).

4. Choose an action based on the beliefs of step 3 according to best response.

5. Observe opponent’s action \( s_t \).

6. Update the propensities estimates using \( m_t = m_t^- + K_t v_t \) and \( P_t = P_t^- - K_t S_t K_t^T \).

7. Set \( t=t+1 \)

---

Table 1: EKF Fictitious Play algorithm

### 4 Theoretical Results

In this section we present the convergence results we obtained for games with at least one pure Nash equilibrium and players who have 2 available actions, \( s = (1, 2) \). We will denote as \( -s \) the action that a player does not choose, for example if Player \( i \)'s opponent chooses action 1, \( s = 1 \) and hence \( -s = 2 \). Also we will denote as \( m[1] \) and \( m[2] \) the estimated means of opponent’s propensity of action 1 and 2 respectively. Similarly \( P[1, 1] \) and \( P[2, 2] \) will represent the variance of the propensity’s estimation of action 1 and 2 respectively, and \( P[1, 2], P[2, 1] \) their covariance.

The proposed algorithm has the following two 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, \( m_{t-1}[s] < m_t[s] \). Also the estimation of his opponent propensity to play action \( -s \) decreases, \( m_{t-1}[-s] > m_t[-s] \).

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

---

\[ 8 \]
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 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.** In a \(2 \times 2\) game where the players use EKF fictitious play process to choose their actions, and the variance of the observation state is set to \(Z = rI + \epsilon I\), with high probability the two players will not change their action simultaneously infinitely often. We define \(\epsilon\) as a random number from normal distribution with zero mean and arbitrarily small covariance matrix, \(I\) is the identity matrix.

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

We should mention here that the reason we set \(Z = rI + \epsilon I\) is in order to break any symmetries that occurred because the initialisation of the EKF fictitious play algorithm. Based on Proposition 1 and 2 we can infer the following propositions and theorems.

**Proposition 3.** (a) In a game where players have two available actions if \(s\) is a Nash equilibrium, and \(s\) is played at date \(t\) in the process of EKF fictitious play, \(s\) is played at all subsequent dates. That is, strict 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 strict Nash equilibrium \(\hat{s}\). In EKF fictitious play process players’ beliefs are formed identically and independently for each opponent based on equation (6). By Proposition 1 we know that players’ estimations about their opponents’ propensities and therefore their strategies, that each player maintains for the other players, will increase for the actions that are included in \(\hat{s}\) and will be reduced otherwise. 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 hence 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, $\sigma_t$, 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.

Theorem 1. The EKF fictitious play process converges to the Nash equilibrium in $2 \times 2$ games with at least one pure Nash equilibrium, when the covariance matrix of the observation space error, $Z$, is defined as in Proposition 3, $Z = rI + \epsilon I$.

Proof. We can distinct two possible initial states in the game. In the first players’ initial beliefs of the players actions are such that their initial joint action $s_0$ is a Nash equilibrium. From Proposition 3 and equation (6) we know that they will play the joint action which is a Nash equilibrium for all the iterations of the game.

The second case where the initial beliefs of the players are such that their initial joint action $s_0$ is not a Nash equilibrium is divided in 2 subcategories. The first include $2 \times 2$ games with only one pure Nash equilibrium. In this case, one of the two players has a dominant action, thus for all the iterations of the game he will choose the dominant action. This action maximises his expected payoff regardless the other player’s strategy and thus he will select this action in every iteration of the game. Therefore because of Proposition 1 the other player will learn his opponent’s strategy and players will choose the joint action which is the pure Nash equilibrium.

The second category includes $2 \times 2$ games with 2 pure Nash equilibria, like the simple coordination game that is depicted in Table 2. In this case players initial joint action $s_0 = (s^1, s^2)$ is not a Nash equilibrium. Then the players will learn their opponent’s strategy, Proposition 3 and Equation (6), and they will change their action. We know from Proposition 2 that in a finite time with high probability the players will not change their actions simultaneously, and hence they will end up in a joint action that will be one of the two pure Nash equilibria of the game.

We can extend the results of Theorem 1 in $n \times 2$ games with a better reply path. A game with a better reply path can be represented as a graph were its edges are the joint 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 2. The EKF fictitious play process converges to the Nash equilibrium in $n \times 2$ games with a better reply path when the covariance matrix of the observation space error, $Z$, is $Z = rI + \epsilon I$.

Proof. Similarly to the $2 \times 2$ games 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 (6), 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.

5 Simulations to define algorithm parameters \( \Xi \) and \( Z \).

The covariance matrix of the state space error \( \Xi = qI \) and the measurement error \( Z = rI \) are two parameters that we should define in the beginning of the EKF fictitious play algorithm and they affect its performance. Our aim is to find values, or range of values, of \( q \) and \( r \) that can efficiently track opponents’ strategy when it smoothly or abruptly change, instead of choosing \( q \) and \( r \) heuristically for each opponent when we use the EKF algorithm. Nevertheless it is possible that for some games the results of the EKF algorithm will be improved for other combinations of \( q \) and \( r \) than the ones that we propose in this section.

We examine the impact of EKF fictitious play algorithm parameters in its performance in the following two tracking scenarios. In the first one a single opponent chooses his actions using a mixed strategy which changes smoothly and has a sinusoidal form over the iterations of the tracking scenario. In particular for \( t = 1, 2, \ldots, 100 \) iterations of the game: \( \sigma_t(1) = \frac{\cos \left( \frac{2\pi n + 1}{2} \right)}{\sigma_t(2)} = 1 - \sigma_t(2) \), where \( n = 100 \). In the second toy example Player \( i \)'s opponent change his strategy abruptly and chooses action 1 with probability \( \sigma_t^2(1) = 1 \) during the first 25 and the last 25 iterations of the game and for the rest iterations of the game \( \sigma_t^2(1) = 0 \). The probability of the second action is calculated as: \( \sigma_t^2(2) = 1 - \sigma_t^2(1) \).

We tested the performance of the proposed algorithm for the following range of parameters \( 10^{-4} \leq q \leq 1 \) and \( 10^{-4} \leq r \leq 1 \). We repeated both examples 100 times for each of the combinations of \( q \) and \( r \). Each time we measured the absolute error of the estimated strategy against the real one. The combined average absolute error when both examples are considered is depicted on Figure 1. The darkest areas of the contour plot represent the areas where the average absolute error is minimised.

The average absolute error is minimised for a range of values of \( q \) and \( r \), that form two distinct areas. In the first area, the wide dark area of Figure 1 the range of \( q \) and \( r \) were \( 0.08 \leq q \leq 0.4 \) and \( 0.2 \leq r \leq 1 \) respectively. In the second area, the narrow dark area of Figure 1 the range of \( q \) and \( r \) were \( 0.001 \leq q \leq 0.025 \) and \( 0.08 \leq r \leq 0.13 \) respectively. The minimum error which we observed in our simulations was in the narrow area and in particular when \( \Xi = 0.01I \) and \( Z = 0.1I \), where \( I \) is the identical matrix.
Figure 1: Combined absolute error for both tracking scenarios. The range of both parameters, $q$ and $r$ is between $10^{-4}$ and 1.

Table 3: Climbing hill game with three players. Player 1 selects rows, Player 2 selects columns, and Player 3 selects the matrix. The global reward depicted in the matrices, is received by all players. The unique Nash equilibrium is in bold.

6 Simulation results

This section is divided in two parts. The first part contains results of our simulations in two strategic form games and the second part contains the results we obtained in an ad-hoc sensor network surveillance problem. In all the simulations of this section we set the covariance matrix of the hidden and the observations state to $\Xi = 0.01I$ and $Z = (0.1 + \epsilon)I$ respectively, where $\epsilon \sim N(0, 10^{-5})$ and $I$ is the identical matrix.

6.1 Simulations results in strategic form games

In this section we compare the results of our algorithm with those of fictitious play in two coordination games. These games are depicted in Tables 2 and 3.

The game that is depicted in Table 2 as it was described in Section 4, is a simple coordination game with two pure Nash equilibria, its diagonal elements. Table 3 presents an extreme version of the climbing hill game (Claus and Boutilier, 1998) in which three players must climb up a utility function in order to reach the Nash equilibrium where their reward is maximised.

We present the results of 50 replications of a learning episode of 50 iterations for each game. As it is depicted in Figures 2 and 3 the proposed algorithm performs better than fictitious play in both cases. In the simple coordination game that is shown in Table 2, the EKF fictitious play algorithm converges to
one of the pure equilibria after a few iterations. On the other hand fictitious play is trapped in a limit cycle in all the replications where the initial joint action was not one of the two pure Nash equilibria. For that reason the players’ payoff for all the iterations of the game was either 1 utility unit or 0 utility units depending to the initial joint action. In the climbing hill game, Table 3 the proposed algorithm converges to the Nash equilibrium after 35 iterations when fictitious play algorithm do not converge even after 50 iterations.

6.2 Ad-hoc sensor network surveillance problem.

We compared the results of our algorithm against those of fictitious play in a coordination task of a power constrained sensor network, where sensors can be either in a sense or sleep mode (Farinelli et al., 2008; Chapman et al., 2011). When the sensors are in sense mode they can observe the events that occur in their range. During their sleep mode the sensors harvest the energy they need in order to be able function when they are in the sense mode. The sensors then should coordinate and choose their sense/sleep schedule in order to maximise the coverage of the events. This optimisation task can be cast as a potential game. In particular we consider the case where I sensors are deployed in an area where E events occur. If an event e, e ∈ E, is observed from the sensors then it produce some utility V_e. Each of the sensors i = 1, ..., I should choose an action s_i = j, from one of the j = 1, ..., J time intervals which they can be in sense mode. Each sensor i when it is in sense mode can observe an event e, if it is in its sense range, with probability p_{ie} = \frac{1}{d_{ie}}, where d_{ie} is the distance between the sensor i and the event e. We assume that the probability each sensor has to observe an event is independent from the other sensors. If we denote as i_ν the sensors that are in sense mode when the event e occurs and e is in their sensing range, then we can write the probability an event e to be observed from
Figure 3: Probability of playing the (U,U,D) equilibrium for the EKF fictitious play (solid line) and fictitious play (dash line) for the three player climbing hill game

the sensors, $i_{in}$ as

$$1 - \prod_{i \in i_{in}} (1 - p_{ie})$$

The expected utility that is produced from the event $e$ is the product of its utility $V_e$ and the probability it has to be observed by the sensors, $i_{in}$ that are in sense mode when the event $e$ occurs and $e$ is in their sensing range. More formally we can express the utility that is produced from an event $e$ as:

$$U_e(s) = V_e(1 - \prod_{i \in i_{in}} (1 - p_{ie}))$$

The global utility is then the sum of the utilities that all events, $e \in E$, produce

$$U_{global}(s) = \sum_{e} U_e(s).$$

Each sensor after each iteration of the game receives some utility which is based on the sensors and the events that are inside his communication and sense range respectively. For a sensor $i$ we denote $\tilde{e}$ the events that are in its sensing range and $\tilde{s}^{-i}$ the joint action of the sensors that are inside his communication range. The utility that sensor $i$ will receive if his sense mode is $j$ will be

$$U_i(s^i = j, \tilde{s}^{-i}) = \sum_{\tilde{e}} U_{\tilde{e}}(s^i = j, \tilde{s}^{-i}).$$

We compared the performance of the two algorithms in 2 instances of the above scenario one with 20 and one with 50 sensors that are deployed in a unit square. In both instances sensors had to choose one time interval of the day that they will be in sense mode and use the rest time intervals to harvest energy. We consider cases where sensors had to choose their sense mode between 2, 3 and
4 available time intervals. Sensors are able to communicate with other sensors that are at most 0.6 distance units away, and can only observe events that are at most 0.3 distance units away. Moreover in both instances we assumed that 20 events took place in the unit square area. Those events were uniformly distributed in space and time, so an event could evenly appear in any point of the unit square area and it could occur at any time with the same probability. The duration of each event was uniformly chosen between (0-6] hours and each event had a value \( V_e \in (0 - 1] \). Figures 4 and 5 depict the average results of 50 replications of the game for the two algorithms. For each instance, both algorithms run for 50 iterations. To be able to average across the 50 replications we normalise the utility of a replication by the global utility that the sensors will gain if they were only in sense mode during the whole day.

(a) Results when sensors have to choose between two time intervals. (b) Results when sensors have to choose between three time intervals. (c) Results when sensors have to choose between four time intervals.

Figure 4: Results of the instance where 20 sensors should coordinate for both algorithms. The results of EKF fictitious play are the solid lines and the results of the classic fictitious play are the dash lines. The horizontal axis of the figures depict the iteration of the game and the vertical axis the global utility as a percentage of the global utility of the system in the case that sensors were always in sense mode.

As we observe in Figures 4 and 5, EKF fictitious play converges to a stable joint action faster than the fictitious play algorithm. In particular on average the EKF fictitious play algorithm needed 10 “negotiation” steps between the sensors in order to reach a stable joint action, when fictitious play needed more than 25. Moreover the classic fictitious play algorithm was always resulted in joint actions with smaller reward than the proposed algorithm.
(a) Results when sensors have to choose between two time intervals. (b) Results when sensors have to choose between three time intervals. (c) Results when sensors have to choose between four time intervals.

Figure 5: Results of the instance where 50 sensors should coordinate for both algorithms. The results of EKF fictitious play are the solid lines and the results of the classic fictitious play are the dash lines. The horizontal axis of the figures depict the iteration of the game and the vertical axis the global utility as a percentage of the global utility of the system in the case that sensors were always in sense mode.

7 Conclusion

We have introduced a variation of fictitious play that uses Extended Kalman filters to predict opponents’ strategies. This variation of fictitious play addresses the implicit assumption of the classic algorithm that opponents use the same strategy in every iteration of the game.

We showed that, for $2 \times 2$ games with at least one pure Nash equilibrium, EKF fictitious play converges in the pure Nash equilibrium of the game. More over the proposed algorithm converges in games with a better reply path, like potential games, and $n$ players that have 2 available actions.

EKF fictitious play performed better than the classic algorithm algorithm in the strategic form games and the ad-hoc sensor network surveillance problem we simulated. Our empirical observations indicate that EKF fictitious play converges to a solution that is better than the classic algorithm and needs only a few iterations to reach that solution. Hence by slightly increasing the computational intensity of fictitious play less communication is required between agents to quickly coordinate on a desired solution.
8 Acknowledgements

This work is supported by The Engineering and Physical Sciences Research Council EPSRC (grant number EP/I005765/1).

A Proof of Proposition 1

We will base the proof of Proposition 1 on the properties of EKF when they are used to estimate opponent’s strategy with two available actions. If player i’s opponent has two available actions 1 and 2, then we can assume that at time $t-1$ Player $i$ maintains beliefs about his opponent’s propensity, with mean $m_{t-1}$ and variance $P_{t-1}$. Moreover based on these estimations he chooses his strategy $\sigma_{t-1}$. At the prediction step of this process he uses the following equations to predict his opponent’s propensity and choose an action using best response.

\[ m_{t}^- = \begin{pmatrix} m_{t-1}[1] \\ m_{t-1}[2] \end{pmatrix} \]

\[ P_{t}^- = \begin{pmatrix} P_{t-1}[1,1] & P_{t-1}[1,2] \\ P_{t-1}[2,1] & P_{t-1}[2,2] \end{pmatrix} + qI \]

without loss of generality we can assume that his opponent in iteration $t$ chooses action 2. Then the update step will be:

\[ v_{t} = z_{t} - h(m_{t}^-) \]

since Players $i$’s opponent played action 2 and $h = \frac{e^{exp(Q_{t}[s]/\tau)}}{\sum_{t \in S} e^{exp(Q_{t}[s]/\tau)}}$ we can write $v_{t}$ and $H_{t}(m_{t}^-)$ as:

\[ v_{t} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} - \begin{pmatrix} \sigma_{t-1}(1) \\ 1 - \sigma_{t-1}(1) \end{pmatrix} \]

\[ = \begin{pmatrix} -\sigma_{t-1}(1) \\ \sigma_{t-1}(1) \end{pmatrix} \]

\[ H_{t}(m_{t}^-) = \begin{pmatrix} a_{t} & -a_{t} \\ -a_{t} & a_{t} \end{pmatrix} \]

where $a_{t}$ is defined $a_{t} = \sigma_{t-1}(1)\sigma_{t-1}(2)$. The estimation of $S_{t} = H(m_{t}^-)P_{t}^-HT(m_{t}^-) + Z$ will be:

\[ S_{t} = a_{t}^2 \begin{pmatrix} b & -b \\ -b & b \end{pmatrix} + Z \]

where $b = P_{t}^-[1,1]+P_{t}^-[2,2]-2P_{t}^-[1,2]$. The Kalaman gain, $K_{t} = P_{t}^-HT(m_{t}^-)S_{t}^{-1}$ can be written as

\[ K_{t} = \frac{1}{2rb + r^2} \begin{pmatrix} P_{t}^-[1,1] & k \\ k & P_{t}^-[2,2] \end{pmatrix} \begin{pmatrix} a_{t} & -a_{t} \\ -a_{t} & a_{t} \end{pmatrix} \begin{pmatrix} b + r & b \\ b & b + r \end{pmatrix} \]

up to a multiplicative constant we can write

\[ K_{t} \sim \begin{pmatrix} c & -c \\ -d & d \end{pmatrix} \]
where \( c = P_t^{-}[1, 1] - P_t^{-}[1, 2] \) and \( d = P_t^{-}[2, 2] - P_t^{-}[1, 2] \). The updates then for the mean and variance are:

\[
    m_t = m_t^{-} + K_t v_t \\
    P_t = P_t^{-} - K_t S_t K_t^T
\]

The mean of the Gaussian distribution that is used to estimate opponent’s propensities is:

\[
    m_t = \left( \begin{array}{c}
    m_t^{-}[1] \\
    m_t^{-}[2]
    \end{array} \right) = \left( \begin{array}{c}
    m_t^{-}[1] - 2 \sigma(1) \frac{\alpha (b-k)}{\alpha (b-k)+r+\epsilon} \\
    m_t^{-}[2] + 2 \sigma(1) \frac{\alpha (b-k)}{\alpha (b-k)+r+\epsilon}
    \end{array} \right)
\]

Based on the above we observe that \( m_t^{-}[1] < m_{t-1}^{-}[1] \) and \( m_t^{-}[2] > m_{t-1}^{-}[2] \) which completes the proof.

**B Proof of Proposition 2**

We consider 2 \( \times \) 2 games with at least one pure Nash equilibrium. In the case that only one Nash equilibrium exists, a dominant strategy exists and thus one of the players will not deviate from this action. Hence we are interested in in 2 \( \times \) 2 games with two pure Nash equilibria. Without loss of generality we consider a game with similar structure to the simple coordination game that is depicted in Table 2 with two equilibria, the joint actions in the diagonal of the payoff matrix, \((U, L)\) and \((D, R)\). We will present calculations for Player 1, but the same results hold also for Player 2. We define \( \lambda \) as the necessary confidence level that Player 1’s estimation of \( \sigma_t(L) \) should reach in order to choose action \( U \). Hence we Player 1 will choose \( D \) if:

\[
    \sigma_t(1) > \lambda \iff \frac{\exp(m_t^{-}[1])}{\exp(m_t^{-}[1]) + \exp(m_t^{-}[2])} > \lambda \iff
\]

\[
    m_t^{-}[1] > \ln\left( \frac{\lambda}{1-\lambda} \right) + m_t^{-}[2] \iff
\]

\[
    m_{t-1}^{-}[1] > \ln\left( \frac{\lambda}{1-\lambda} \right) + m_{t-1}^{-}[2]
\]

In order to prove Proposition 2 we need to show that when a player changes his action his opponent will change his action at the same iteration with probability less than 1. In the case where at time \( t-1 \) the joint action of the players is \( U, R \) then Player 1 believes that his opponent will play \( L \), while he observing him playing \( R \). Assume that Player 2’s beliefs about Player 1’s strategies has reached the necessary confident level about Players 1’s strategy and at iteration \( t \) he will change his action from \( R \) to \( L \). Player 1 will also change his action at the same time if

\[
    m_{t-1}^{-}[2] > \ln\left( \frac{1-\lambda}{\lambda} \right) + m_{t-1}^{-}[1]
\]

We want to show that players will not change actions simultaneously with probability 1. Hence it is enough to show that
\[ \text{Prob}(m_{t-1}[1] > \ln\left( \frac{\lambda}{1 - \lambda} \right) + m_{t-1}[2]) > 0 \quad (8) \]

We can replace \( m_{t-1}[1] \) and \( m_{t-1}[2] \) with their equivalent from (7) and write:

\[
\begin{align*}
&\quad m^{-}[1] - 2\sigma(1) \frac{a(b - k)}{4a^2(b - k) + (r + \epsilon)} > \ln\left( \frac{\lambda}{1 - \lambda} \right) + m^{-}[2] + 2\sigma(1) \frac{a(b - k)}{4a^2(b - k) + (r + \epsilon)} \Leftrightarrow \\
&\quad -4\sigma(1) \frac{a(b - k)}{4a^2(b - k) + (r + \epsilon)} > \ln\left( \frac{\lambda}{1 - \lambda} \right) + m^{-}[2] - m^{-}[1] \Leftrightarrow \\
&\quad \frac{a(b - k)}{4a^2(b - k) + (r + \epsilon)} < \frac{\ln\left( \frac{\lambda}{1 - \lambda} \right) + m^{-}[2] - m^{-}[1]}{-4\sigma(1)}
\end{align*}
\]

Solving this with respect to \( \epsilon \) we have

\[ \epsilon > \frac{a(b - k)\sigma(1)}{\ln\left( \frac{\lambda}{1 - \lambda} \right) + m^{-}[2] - m^{-}[1]} - a^2(b - k) - r \]

Thus we can write (8) as:

\[ \text{Prob}(\epsilon > \frac{a(b - k)\sigma(1)}{\ln\left( \frac{\lambda}{1 - \lambda} \right) + m^{-}[2] - m^{-}[1]} - a^2(b - k) - r) > 0 \quad (9) \]

Since \( \epsilon \) is a Gaussian white noise (9) is always true.

We also consider the case where at time \( t - 1 \) the joint action of the players is \( D, L \) then Player 1 believes that his opponent will play \( R \), while he observing him playing \( L \). Assume that Player 2’s beliefs about Player 1’s strategies has reached the necessary confident level and at \( t \) he will change his action from \( L \) to \( R \). Player 1 will also change his action at the same time if

\[ m_{t-1}[1] > \ln\left( \frac{\lambda}{1 - \lambda} \right) + m_{t-1}[2] \]

We want to show that Players will not change actions simultaneously with probability 1. Hence it is enough to show that

\[ \text{Prob}(m_{t-1}[2] > \ln\left( \frac{1 - \lambda}{\lambda} \right) + m_{t-1}[1]) > 0 \quad (10) \]

We can rewrite (10) using the results we obtained for \( m_{t-1}[1] \) and \( m_{t-1}[2] \) in (7) again as

\[ \text{Prob}(\epsilon > \frac{a(b - k)\sigma(1)}{\ln\left( \frac{\lambda}{1 - \lambda} \right) + m^{-}[2] - m^{-}[1]} - a^2(b - k) - r) > 0 \quad (11) \]

Since \( \epsilon \) is a Gaussian white noise (11) is always true.

If we define \( \xi_t \) 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 \( t_1, t_2, \ldots, t_t \), then the probability that they will also change their action simultaneously at time \( t_{T+1} \), \( P(\xi_{t_1}, \xi_{t_2}, \ldots, \xi_{t_T}, \xi_{t_{T+1}}) \) is almost zero for large but finite \( T \).
References

Arslan, G., Marden, J., Shamma, J.. Autonomous vehicle-target assignment: A game theoretical formulation. 2006.

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

Chapman, A.C., Leslie, D.S., Rogers, A., Jennings, N.R.. Convergent learning algorithms for potential games with unknown noisy rewards. Working Papers 05/2011; University of Sydney Business School, Discipline of Business Analytics; 2011.

Claus, C., Boutilier, C.. The dynamics of reinforcement learning in cooperative multiagent systems. In: Proceedings of the fifteenth nationalon Artificial intelligence. 1998.

Farinelli, A., Rogers, A., Jennings, N.. Maximising sensor network efficiency through agent-based coordination of sense/sleep schedules. In: Workshop on Energy in Wireless Sensor Networks in conjunction with DCOSS 2008. 2008. p. 43–56.

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

Grewal, M., Andrews, A.. Kalman filtering: theory and practice using MATLAB. Wiley-IEEE press, 2011.

Jazwinski, A.. Stochastic processes and filtering theory. volume 63. Academic press, 1970.

Kalman, R., et al. A new approach to linear filtering and prediction problems. Journal of basic Engineering 1960;82(1):35–45.

Kho, J., Rogers, A., Jennings, N.R.. Decentralized control of adaptive sampling in wireless sensor networks. ACM Trans Sen Netw 2009;5(3):1–35.

van Leeuwen, P., Hesselink, H., Rohlinga, J.. Scheduling aircraft using constraint satisfaction. Electronic Notes in Theoretical Computer Science 2002;76:252 – 268.

Monderer, D., Shapley, L.. Potential games. Games and Economic Behavior 1996;14:124–143.

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

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

Stranjak, A., Dutta, P.S., Ebden, M., Rogers, A., Vytelingum, P.. A multi-agent simulation system for prediction and scheduling of aero engine overhaul. In: AAMAS '08: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems. 2008. p. 81–88.
Wolpert, D., Turner, K.. An overview of collective intelligence. Handbook of Agent Technology 1999.;

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