Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

SEN

Software Engineering

Software ENgineering

Foresighted policy gradient reinforcement learning: solving large-scale social dilemmas with rational altruistic punishment

P.J. 't Hoen, S.M. Bohte, J.A. La Poutré

REPORT SEN-R0804 OCTOBER 2008

Centrum Wiskunde & Informatica (CWI) is the national research institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organisation for Scientific Research (NWO).
CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-369X

Foresighted policy gradient reinforcement learning: solving large-scale social dilemmas with rational altruistic punishment

ABSTRACT
Many important and difficult problems can be modeled as ``social dilemmas'', like Hardin's Tragedy of the Commons or the classic iterated Prisoner's Dilemma. It is well known that in these problems, it can be rational for self-interested agents to promote and sustain cooperation by altruistically dispensing costly punishment to other agents, thus maximizing their own long-term reward. However, self-interested agents using most current multi-agent reinforcement learning algorithms will not sustain cooperation in social dilemmas: the algorithms do not sufficiently capture the consequences on the agent's reward of the interactions that it has with other agents. Recent more foresighted algorithms specifically account for such expected consequences, and have been shown to work well for the small-scale Prisoner's Dilemma. However, this approach quickly becomes intractable for larger social dilemmas. Here, we advance on this work and develop a ``teach/learn'' stateless foresighted policy gradient reinforcement learning algorithm that applies to Social Dilemma's with negative, unilateral side-payments, in the from of costly punishment. In this setting, the algorithm allows agents to learn the most rewarding actions to take with respect to both the dilemma (Cooperate/Defect) and the ``teaching'' of other agent's behavior through the dispensing of punishment. Unlike other algorithms, we show that this approach scales well to large settings like the Tragedy of the Commons. We show for a variety of settings that large groups of self-interested agents using this algorithm will robustly find and sustain cooperation in social dilemmas where adaptive agents can punish the behavior of other similarly adaptive agents.

2000 Mathematics Subject Classification: 91A06
1998 ACM Computing Classification System: I.2.11
Keywords and Phrases: multi-agent games; iterated prisoner's dilemma; side payments
Note: This work has been carried out under theme SEN4 ``Computational Intelligence and Multi-Agent Games''. Part of this research has been performed within the framework of the project ``Distributed Engine for Advanced Logistics (DEAL)'' funded by the E.E.T.\ program in the Netherlands. Work of SB is supported by NWO VENI grant 639.021.203.

Foresighted Policy Gradient Reinforcement Learning: Solving Large-Scale Social Dilemmas with Rational Altruistic Punishment

Pieter Jan 't Hoen, Sander Bohte, and Han La Poutré

CWI, 1098SJ Amsterdam, The Netherlands,
{hoen,sbohte,hlp}@cwi.nl

Abstract. Many important and difficult problems can be modeled as "social dilemmas", like Hardin's Tragedy of the Commons or the classic iterated Prisoner's Dilemma. It is well known that in these problems, it can be rational for self-interested agents to promote and sustain cooperation by altruistically dispensing costly punishment to other agents, thus maximizing their own long-term reward. However, self-interested agents using most current multi-agent reinforcement learning algorithms will not sustain cooperation in social dilemmas: the algorithms do not sufficiently capture the consequences on the agent's reward of the interactions that it has with other agents. Recent more foresighted algorithms specifically account for such expected consequences, and have been shown to work well for the small-scale Prisoner's Dilemma. However, this approach quickly becomes intractable for larger social dilemmas. Here, we advance on this work and develop a "teach/learn" stateless foresighted policy gradient reinforcement learning algorithm that applies to Social Dilemma's with negative, unilateral side-payments, in the from of costly punishment. In this setting, the algorithm allows agents to learn the most rewarding actions to take with respect to both the dilemma (Cooperate/Defect) and the "teaching" of other agent's behavior through the dispensing of punishment. Unlike other algorithms, we show that this approach scales well to large settings like the Tragedy of the Commons. We show for a variety of settings that large groups of self-interested agents using this algorithm will robustly find and sustain cooperation in social dilemmas where adaptive agents can punish the behavior of other similarly adaptive agents.

1 Introduction

It is well known that in many cases, greedy and selfish behavior by individual agents may harm the value extracted by a collective of agents as a whole, and – importantly – ultimately decrease the value extracted by the individual agents. This class of problems is known as "social dilemmas", with the classic iterated Prisoner's Dilemma (IPD) being an example in the smallest – two player – setting, and Hardin's Tragedy of the Commons (Hardin, 1968) embodying the same problem instantiated with many players. In both settings it is the case that when

all agents take the immediately most rewarding action, the resulting collective joint action has an individual payoff that is worse for each individual agent than some other joint action. Thus, in repeated play of these games, myopic selfish behavior leads to poor outcomes for all individuals.

The problem of how cooperation can be sustained by self-interested agents in social dilemmas has been studied in such diverse fields as economics (Klein & Leffler, 1981; Shapiro, 1983; Anderson & Putterman, 2006), game theory (Rubinstein, 1979; Milgrom, North, & Weingast, 1990; Kandori, 1992), politics (Axelrod, 1984), evolutionary biology (Boyd, Gintis, Bowles, & Richerson, 2003) and the social sciences (Fehr & Fischbacher, 2003). For large groups of interacting agents facing social dilemmas, much research has focused on groups where many 2-player IPD games are played between agents (Boyd et al., 2003; Fehr & Fischbacher, 2003; Sen, 1996; Nowak & Sigmund, 2005). This corresponds closely with the classical example of trade. Many settings however more closely resemble Hardin's Tragedy of the Commons, where individuals can affect the reward for *all* involved.

It follows from the Folk Theorems (Fudenberg & Maskin, 1986) that Pareto-optimal cooperation can in principle be rationally enforced as a subgame perfect equilibrium in many social dilemmas: if all agents use a certain strategy, and all agents belief that the other agents also use this strategy, no individual agent has an incentive to deviate from that strategy. The proofs heavily rely on two conditions: first, the actions of individual agents are at least partially public information for other agents to act on. Second, observed deviation from desired behavior is actively punished by other agents, at personal cost to these agents: (seemingly) altruistic punishment (Kandori, 1992).

Here, we focus on the setting of the original Tragedy of the Commons, with the familiar properties of the n-player IPD (NIPD), with the additional option of agents making *side-payments* (Andreoni & Varian, 1999). Where positive side-payments may encourage certain actions, and (unilateral) negative side-payments - punishments - are discouraging. With negative unilateral side-payments, agents can dispense punishment to (some) other agents, at a personal cost. In (Milgrom et al., 1990), it is shown that from a game-theoretic perspective, dispensing seemingly altruistic punishment can be incentive compatible for a self-interested agent for both 2- and n-player IPD games.

In (Boyd et al., 2003), social dilemmas with possible altruistic punishment are modeled as a repeated game with two stages: in the first stage, the agents play an IPD game, and in the second stage, the agents may punish other agents. When agents can make unilateral side payments by dispensing punishment to (some) other agents – at a personal cost – defecting in the is no longer the dominant strategy in the first IPD stage of the one-shot game. However, as punishing itself is costly, there is still a second order "free-rider" problem in that agents prefer that other agents dispense the punishment, and when no one punishes, the original IPD problem returns.

To study large-scale social dilemmas of the NIPD variety, we use this two-stage formalization, where the first stage consists of a single round of the n-

player Prisoner's Dilemma involving all agents in the group. We denote this formalization as the NIPD-AP game: NIPD with altruistic punishment.

From a machine learning perspective, NIPD-AP type games are interesting, because they require agents to have an accurate "theory of mind" (Fehr & Fischbacher, 2003): the individual agents need to be able to predict the reactions and adaptations of other agents to their own actions. Only then can cooperation-supporting punishment strategies be potentially rational for the agent in terms of expected long term profits. Sufficiently foresighted and rational agents should thus be able to learn that in social dilemmas, given similarly smart opponents, it can be rational to *actively* sustain cooperation with costly altruistic punishment, even for large groups of agents in Multi-Agent Systems (MAS).

This issue of an agent's actions changing other agent's behaviors is considered in (Shoham, Powers, & Grenager, 2007): the authors remark that "In a multi-agent setting one cannot separate *learning* from *teaching*" (in the sense that adaptations may be reactive and can be influenced/taken into account). Although game theoretical results with full rationality show that cooperation can be enforced in NIPD games through disruptive so-called grim-trigger strategies, it is an open question how such results can be obtained in a gradual fashion by individual learning agents each using reinforcement learning algorithms.

Current state-of-the-art multi-agent reinforcement learning (MARL) algorithms are typically a mixture of Best Response[1] and fixed strategies (Crandall & Goodrich, 2005). Consequently, typical current MARL algorithms have great difficulty with the 2-player IPD (Crandall & Goodrich, 2005), and generally lack results for IPD games with more players.

We are not aware of any MARL work on NIPD games with side-payments. However, as we show in this paper, standard, myopic gradient-based reinforcement learning performs poorly in NIPD-AP games, and similar results are expected for any reinforcement learning algorithm that relies on myopic optimization: due to the second order dilemma of punishing, the Best Response action in the NIPD-AP game is to not punish, as the punishing agent incurs an immediate penalty without any immediate benefit. Additionally, for most current MARL algorithms, it is not obvious how punishing side-payments should be incorporated.

With these considerations in mind, we design a reinforcement learning algorithm that allows self-interested agents to be foresighted enough to learn to jointly sustain cooperation in large NIPD-AP games with many similarly foresighted agents participating in the MAS. We develop a foresighted policy gradient reinforcement learning algorithm, FPGRL, to learn to optimize an agent's reward in the NIPD-AP game. The FPGRL algorithm endows individual agents with an estimate of the impact of their actions on the policy adaptations of other agents, and the associated gradient of future reward. If the future expected reward conditional on dispensing punishment exceeds the immediate cost, it can become rational for an agent to do so. This conditional future reward would be derived from other agents cooperating more, under the expectation that if they defect,

[1] The best response is the strategy (or strategies) which produces the most favorable immediate outcome for the current agent, taking other agent's strategies as given.

their payoff is so diminished due to punishments, that cooperating is expected to be more rewarding (clearly, all agents have to be sufficiently foresighted for this scheme to work, so we focus on self-play here).

Our approach advances recent work on foresighted reinforcement learning in ('t Hoen, Bohte, & La Poutré, 2006a, 2006b). These algorithms enable self-interested agents to learn to maximize their reward by sustained cooperation in small-scale NIPD games. If effect, the algorithms find that by learning tit-for-tat like strategies (Axelrod, 1984), an agent can "threaten" to punish other agents by measurably reducing future cooperation in response to undesired behavior (just like game theoretic constructions like grim-trigger strategies to sustain subgame perfect equilibria). However, these algorithms only work for small groups of agents, as it rapidly becomes intractable for individual agents to track the effect their cooperation or defection has on the behavior of the other individual agents. For larger groups, the collective behavior then increasingly descends into the worst possible outcome of collective defection.

The FPGRL algorithm we develop here explicitly accounts for the joint learning/teaching intrinsic to multi-agent learning (Shoham et al., 2007), and the foresighted anticipation of expected reactions required for successful "teaching". "Teaching" is taken as the consideration whether or not it is rational, in the sense of a sufficient increase in expected future rewards, to punish other agents that are observed to exhibit undesired behavior. The "learning" accounts for both the expected immediate payoff of an action, as well as the expected amount of punishment that can be expected from other agents. We show that together these ingredients of the FPGRL algorithm allow agents to successfully learn both a suitable policy for cooperating and for dispensing altruistic punishments in large groups as a means to increase their own reward in the NIPD-AP game.

We thus show that FPGRL allows a large group of sufficiently smart competitive agents to learn to cooperate in the NIPD-AP game: they are each able to compute that, given the observed adaptive punishments of other agents, cooperating will increase the expected reward from the repeated interaction. The agents thus learn to cooperate through selfish maximization of their own perceived long-term rewards. The FPGRL algorithm plays a Best Response strategy when the opponents are estimated to have stationary strategies. We find that agents using FPGRL in the NIPD-AP game achieve and sustain cooperation in groups of more than 250 agents (further experiments were limited by computer memory), for a range of punishment models.

2 Model

In the classic NIPD game, each player has a choice of two actions: either **cooperate** (C) with the other players or **defect** (D). A game can be classified as an NIPD game if it has the following three properties: 1) Each player can choose between playing cooperation (C) and defection (D); 2) The D option is dominant for each player, i.e. each has a better payoff choosing D than C no matter how

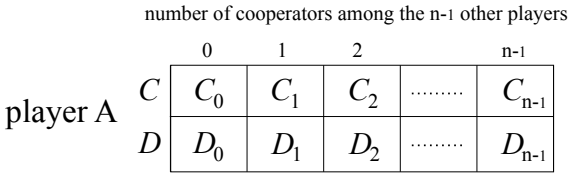

Fig. 1. Structured payoffs for the (symmetrical) NIPD

many of the other players choose C; 3) The dominant D strategies intersect in a deficient equilibrium.

The NIPD payoff matrix is shown in Figure 1; C_i and D_i respectively denote the reward for cooperating and defecting with i cooperators (and $n-i-1$ other defectors). The following conditions hold for the respective payoffs (Yao & Darwen, 1994): (1) $D_i > C_i$ for $0 \leq i \leq n-1$; (2) $D_{i+1} > D_i$ and $C_{i+1} > C_i$ for $0 \leq i < n-1$; (3) $C_i > \frac{(D_i+C_{i-1})}{2}$ for $0 \leq i \leq n-1$ (the payoff matrix is symmetric for each player).

When playing NIPD, the outcome where all (other) players choose their non-dominant C-strategies is preferable from every player's point of view to the one in which everyone chooses D, but no one is motivated to deviate unilaterally from playing Defect, regardless of what other agents are playing. The natural outcome of agents playing myopic Best-Response[1] policies in NIPD is Defection by all agents, which is stable (a single stage Nash-Equilibrium (NE)) but obviously Pareto-deficient[2](Gintis, 2000).

Here, we assume that an agent can observe its own payoffs, its own actions, as well as the actions taken by (a selection of) the other agents in each round of the game. Repeated games are modeled as a series of rounds with the same opponent(s). All agents concurrently choose their actions. A potential adaptation of the agents' policy, i.e. learning as a result of observed opponent behavior, takes effect in the next round of the game. Each agent aims to maximize its average reward from iterated play of the same game by adjusting its actions to follow the immediate reward gradient.

We formalize the NIPD-AP game – NIPD with altruistic punishment – as in (Boyd et al., 2003). In the NIPD-AP, each round consist of two stages: in the first stage, the agents play the NIPD, and in the second stage, the agents choose whether to punish certain other agents.

[2] A **Nash Equilibrium** is a joint strategy such that no agent may unilaterally change its strategy without lowering its expected payoff in the one shot play of the game. A **Pareto optimal** solution of the game is a joint strategy such that no agent may unilaterally increase its expected payoff without making another agent worse off. A (joint) strategy π_1 is said to **Pareto dominate** a strategy π_2 if the expected payoff for π_1 is at least as high as for π_2 and higher for at least one of the agents. A joint strategy is **Pareto deficient** if it is not Pareto optimal.

Formally, an agent K can in the first stage play cooperate, C, or defect, D, and the agents receive the payoffs as defined in the standard NIPD. In the second stage, an agent K can choose to punish or not punish certain other agents. When agent K chooses to punish another agent L, agent L receives a penalty $p_K > 0$ reducing its payoff by p_K, and agent K receives a penalty $pc_K > 0$, a "personal cost". Without such a personal cost there is no direct incentive for an individual agent to minimize the amount of dispensed punishing.

For altruistic punishment to be rational for an agent, the agent has to compute that dispensing some amount of punishment increases its expected future reward more than the immediate expense of punishing. In terms of the NIPD, this means that an agent will only dispense punishment if it computes that this sufficiently increases cooperation of the punished agent(s). Note that if punishment effectively promotes cooperation in the system, the required amount of actually dispensed punishment declines with increasing cooperation.

We define different degrees of public information on agent behavior, that other agents can act on. Let victims(K) be the set of agents whose actions agent K observes and can potentially punish. If agent K chooses to punish in the second stage, it punishes all agents $L \in$ victims(K) that played D in the first stage; the payoff of agents L is for each agent reduced by p_K, and the payoff of agent K by pc_K.

For punishment to encourage cooperation, it is required that the expected imposed penalty for an agent L exceeds the reward agent L can gain from defecting. Formally, let $\mathcal{A}_L$ denote the set of agents that can punish an agent L, and let $\langle p_{K_j} \rangle$ denote the expected immediate penalty from agent $K_j \in \mathcal{A}_L$ imposed onto agent L for defecting; further let C_i and D_i be the payoffs of the NIPD game. It is incentive compatible for agent L to cooperate when:

$$D_i - \sum_{K_j \in \mathcal{A}_L} \langle p_{K_j} \rangle < C_i. \tag{1}$$

We define three models for the victim(K) sets, corresponding to different degrees of public information that individual agents observe of other agents' actions in the NIPD-AP game. The different models let us gain insight into the degree to which rational altruistic punishment depends on the availability of public information. We investigate various sizes of the victim set, |victim(K)|, as well as a more general punishment scheme where an agent can punish all agents for an single observed defection. The latter is inspired by the observation of (Boyd et al., 2003) that in their evolutionary simulations, the ability of agents to also punish non-punishing agents significantly increases the cooperation in their system.

First, we define NIPD-AP(1:1). Each agent can potentially punish exactly one other agent. Agent K_i potentially punishes agent $K_{(i+1) mod\ n}$ to introduce a ring structure where every agent can punish exactly one other agent for defecting. For example, for three agents K_0, K_1, and K_2, agent K_0 can punish agent K_1, agent K_1 agent K_2, and agent K_2 can punish K_0.

We generalize NIPD-AP(1:1) for a larger victim set: NIPD-AP(1:K). In NIPD-AP(1:K) each agent can observe the defections of k other agents and then poten-

tially punish them. Agent K_i is then able to punish agents in the set victims(K_i) formed by the agents with indices ranging from $K_{(i+1)mod\ n}$ to $K_{(i+k)mod\ n}$, where $k < n - 1$. Note that a defector can now be punished by up to k agents at the same time for a single defection. For each agent punished, the punisher K_i pays the personal punishment cost pc_{K_i}; for k agents punished, a price of $k \times pc_{K_i}$ is paid.

Lastly, we introduce a "spite" model: NIPD-AP(1:∼N). Here, a punishment action by an agent dispenses a punishment to all agents in the system. Agents can then only punish defections by applying a broad penalty to the whole group of defectors and cooperators. For example, for agents K_0, K_1 and K_2, if agent K_1 defects then if agent K_0 punishes, agent K_2 also receives the punishment, as well as agent K_0 itself.

Policies and State space representation. The observed joint actions, and the value of these joint actions that can be learned, consists of four parts. First of all, each agent uses the fact whether they cooperated or defected in the last round. If they defected, they record whether they received a punishment. Furthermore, they record whether the agents they monitor for defection, i.e. the victim-set, cooperated or defected. Lastly, the agent records whether a defection by a victim was punished or not. Along with a cooperation and punishment policy, this represents how one agent models the (observable) joint actions of the MAS, its personal payoff for cooperation and punishment, and the expected value of its current policies.

An agent determines which action to play as a joint policy of two separate policies: the cooperation policy, and the punishment policy. Let the cooperation policy $\mu_k \in [0, 1]$ denote the likelihood that agent K cooperates in the first stage of a NIPD-AP round (and otherwise defects). Let $\eta_k \in [0, 1]$ denote the punishment policy, i.e. the likelihood that agent K will punish a defection by an agent in the victims(k) set.

We first consider the variant of NIPD-AP(1:1), i.e. each agent only potentially punishes exactly one other agent. For an agent K then, two other agents matter: the agent L that agent K monitors for defection: $L = \text{victims}(K)$, and agent M that monitors agent K: $K = \text{victims}(M)$. There are then z other agents playing the game. Let $\hat{\mu}_L$ and $\hat{\eta}_M$ be the estimates that agent K has for the cooperation and punishment policy of respectively agent L and agent M.

The expected payoff of agent K for potential joint actions is given in Table 1. For agents K, L, and M, the possible joint actions with *different* payoffs are given, numbered one through nine. For agent K and L, the choices of cooperation are given. Where relevant, the choices for *P*unishing or *N*ot *P*unishing are also given. The likelihood for the current cooperation and punishment policy along with the estimated opponent policies are sufficient for agent K to estimate the expected reward of its current policies. For the NIPD-AP (1:K) variants, with more than one agent in the VICTIMS set, Table 1 is also used as joint action values for an agent K, with the appropriate extensions to account for each individual agent in the victim set of K and the agent set monitoring agent K.

Table 1. Payoffs for Agent K playing the NIPD-AP for potential joint actions. The labels "got P/NP" and "to P/NP" respectively denote Agent K receiving/not receiving punishment, and dispensing/not dispensing punishment.

case	a_K	a_L	got P/NP	to P/NP	probability	reward
1	C	C	-	-	$\mu_K \times \hat{\mu}_L$	C_{z+2}
2	C	D	-	yes	$\mu_K \times (1-\hat{\mu}_L) \times \eta_K$	$C_{z+1} - pc_K$
3	C	D	-	no	$\mu_K \times (1-\hat{\mu}_L) \times (1-\eta_K)$	C_{z+1}
4	D	C	yes	-	$(1-\mu_K) \times \hat{\mu}_L \times \hat{\eta}_M$	$D_{z+1} - p_M$
5	D	C	no	-	$(1-\mu_K) \times \hat{\mu}_L \times (1-\hat{\eta}_M)$	D_{z+1}
6	D	D	yes	yes	$(1-\mu_K) \times (1-\hat{\mu}_L) \times \eta_K \times \hat{\eta}_M$	$D_z - p_M - pc_K$
7	D	D	yes	no	$(1-\mu_K) \times (1-\hat{\mu}_L) \times \eta_K \times \hat{\eta}_M$	$D_z - p_M$
8	D	D	no	yes	$(1-\mu_K) \times (1-\hat{\mu}_L) \times \eta_K \times (1-\hat{\eta}_M)$	$D_z - pc_K$
9	D	D	no	no	$(1-\mu_K) \times (1-\hat{\mu}_L) \times (1-\eta_K) \times (1-\hat{\eta}_M)$	D_z

3 The FPGRL Algorithm

Here, we present FPGRL, a stateless foresighted policy gradient reinforcement algorithm. FPGRL combines the idea of foresighted accounting of reactive-adaptive interactions with a policy gradient reinforcement learning approach to overcome the intractability of full state-space representations for large groups of agents. A foresighted algorithm is needed to sustain cooperation in NIPD-AP as an agent's actions directly influence the policy adaptations of other agents, and thus influence the agent's future payoff.

In ('t Hoen et al., 2006a) a foresighted reinforcement learning algorithm is developed that accounts for agents' reactive adaptation due to interactions. However, this approach works only for fine-grained interactions between a small number of agents, as states are encoded as the recent history of actions of all the other agents in the system. This becomes quickly intractable for large multi-agent systems (MAS).

In FPGRL, no states are encoded, but rather a cooperation policy μ_k and a punishment policy η_k are adapted with respect to the observed gradient in the reward, conditional on reactive adaptations of opponents. In our implementation of FPGRL, we only apply foresighted reinforcement learning to the punishment policy η_k, by attributing all opponent adaptation to changes in η_k. The cooperation policy μ_k is learned through standard Policy Gradient (Baird & Moore, 1999). The reasoning behind this choice is, that it is already known that focusing on the cooperation policy alone allows for only limited cooperation in large multi-agent systems ('t Hoen et al., 2006b).

In the FPGRL algorithm, an agent k updates its punishment policy η_k as follows. At each round of learning, the agent adapts η_k along the gradient of increasing reward. The gradient of reward is calculated including the expected adaptations of the observed agents due to the agent's own actions. The FPGRL algorithm tracks the changes in observed opponent policies over time. It is as-

Algorithm 1 FPGRL

For each agent k, initialize η_k, $\hat{\mu}_{-k}$ and V_k for all joint actions $\{a_k, a_{-k}\}$.
Do in each epoch
loop
  Calculate highest valued pua_j^* using (8)
  Update policy η_k using pua_j^*.
  Play $a_k \in A_k$ according to η_k.
  Receive reward $r_{k,t}$ for joint action $\{a_k, a_{-k}\}$.
  Update joint action reward estimate $V_k(\begin{bmatrix} a_k \\ a_{-k} \end{bmatrix})$, and update opponent policy $\hat{\mu}_{-k}$.
end loop

sumed that changes in the opponent behavior, at least in part, reflect a reaction to actions chosen by the FPGRL algorithm. The policy of the FPGRL is then optimized with expected future reactive adaptations of the opponents taken into account.

We describe the FPGRL algorithm from the perspectives of an agent k and its opponents, agents $-k$, we use this notation with subscripts to indicate policy, action, etc ... of the two types of agents. For example, a_k and a_{-k} are actions of agents k and $-k$ respectively.

Let $\hat{\mu}_{-k}$ be the estimate of an opponent's $-k$ cooperation policy. Here, we estimated this online using an Exponential Moving Average (EMA). At time t, after observing action a_{-k}, $\hat{\mu}_{-k,t}$ is adjusted according to:

$$\hat{\mu}_{-k,t+1}(a_{-k}) = (1 - \alpha_{EMA1})\hat{\mu}_{-k,t}(a_{-k}) + \alpha_{EMA1}. \tag{2}$$

with learning rate $0 \leq \alpha_{EMA1} \leq 1$. After this update, the policy $\hat{\mu}_{-k,t+1}$ is normalized to retain $\hat{\mu}_{-k}$ as a probability distribution.

The goal is to determine a policy adaptation that maximizes future expected reward. As computing forward all possible policy adaptations is intractable, we introduce a set of **policy update actions**. A policy update action pua_i determines whether the likelihood of playing an action a_i should be increased, including a *null* policy update action pua_{null} that does not change the current policy. From the set pua_i, the update expected to be most rewarding is selected as follows.

Let $\eta_k^{pua_i}$ be the policy achieved by applying the policy update action pua_i to the punishment policy η_k. The policy η_k of agent k given $pua_i \neq pua_{null}$ is updated according to:

$$\eta_{k,t+1}^{pua_i} = (1 - \alpha_L)\eta_{k,t}(a_i) + \alpha_L, \tag{3}$$

where α_L is the learning rate. The probabilities $\eta_k(\cdot)$ for actions $a_j \neq a_i$ are then normalized to maintain η_k as a probability distribution. Lastly, $\eta_{k,t+1}^{pua_{null}} = \eta_{k,t}$.

Let the function $\xi(\hat{\mu}_{-k}, a_i)$ estimate the changed cooperation policy $\hat{\mu}_{-k}$ of the opponent $-k$ due to an agent k playing an action a_i:

$$\hat{\mu}_{-k,t+1} = \xi(\hat{\mu}_{-k,t}, a_i) \tag{4}$$

As an implementation of ξ, we assume that the changes in the opponent policy are, at least in part, caused by the actions of agent k. We compute the change in opponent policy as a linear extrapolation of the change in the estimated opponent policy as observed N rounds in the past, i.e. the opponent policy at time $t - N$:

$$\xi(\hat{\mu}_{-k,t}, a_i) = \frac{\hat{\mu}_{-k,t} - \hat{\mu}_{-k,t-N}}{N} \frac{N_{a_i}}{N} + \hat{\mu}_{-k,t}, \tag{5}$$

where we limit ξ to $[0, 1]$ and N_{a_i} denotes the number of times action a_i has been played in the last N rounds. Here, we use $N = 10$.

Let the values $V_k : A_k \times A_{-k} \rightarrow \Re$ denote the estimates of the reward of a single joint action $\{a_k, a_{-k}\}$ to agent k (note: this is the part of the reward agent k receives, not the joint reward). As a simple implementation, the estimate for V_k is updated using the exponential moving average (EMA):

$$V_{k,t+1} = (1 - \alpha_{EMA2})V_{k,t} + \alpha_{EMA2} \times r_{k,t}, \tag{6}$$

where $r_{k,t}$ is the reward received by agent k in round t, and $0 < \alpha_{EMA2} \leq 1$ is the learning rate (round t denoting the last time the particular joint action was played).

The expected reward for a joint action $\{a_k, a_{-k}\}$ is computed using the agent's policy η_k, the opponent's estimated policy $\hat{\mu}_{-k}$, and the learned joint action values V:

$$val(\eta_k, \hat{\mu}_{-k}, V_k) = \sum_{a_i, a_{-j}} P(a_i|\eta_k) P(a_{-j}|\hat{\mu}_{-k}) V_k(\begin{bmatrix} a_i \\ a_{-j} \end{bmatrix}), \tag{7}$$

where for policy η_k, the probability of playing action a_i is denoted as $P(a_i|\eta_k)$.

With the opponent changes in policy estimated by $\xi(\hat{\mu}_{-k}, a_i)$, we compute the policy update with the highest expected payoff:

$$pua_j^* = \max_j val(\eta_k^{pua_j}, \xi(\hat{\mu}_{-k}, a_j), V) \tag{8}$$

where *val* calculates the expected payoff for the achieved joint policies after effecting policy update action pua_j and taking $\xi(\hat{\mu}_{-k}, a_j)$ as the estimated opponent cooperation adaptation, for learned reward V for joint actions.

The FPGRL algorithm thus obtained is outlined in Algorithm 1. Note that for static opponents that do not change their policy, ξ will be 0, and the policy-gradient will realize the best-response settings for μ_k and η_k.

4 Results

We investigate to what extend agents using the FPGRL algorithm in a multi-agent system (MAS) playing the NIPD-AP game learn to dispense altruistic punishment to maximize their own personal reward. Results are shown for learning rates α all equal to 0.01 for all agents. Rewards for full cooperation are scaled linearly from 1 for four agents up to a reward of 30 for 256 agents. We fixed the punishment

amount and the cost of punishment: punishment for defection is scaled so that according to Eq. (1) a 20% chance of punishment is sufficient to make cooperation more attractive in the first stage than defection for an individual agent; the cost of punishing was set at 0.1. Values in this range for these parameters were found to yield good results; setting the cost too high and the punishment too low will have the agents effectively ignore the punishment signals, and setting the cost too low and/or the punishment too high effectively forces the agents to playing a the cooperative joint move.

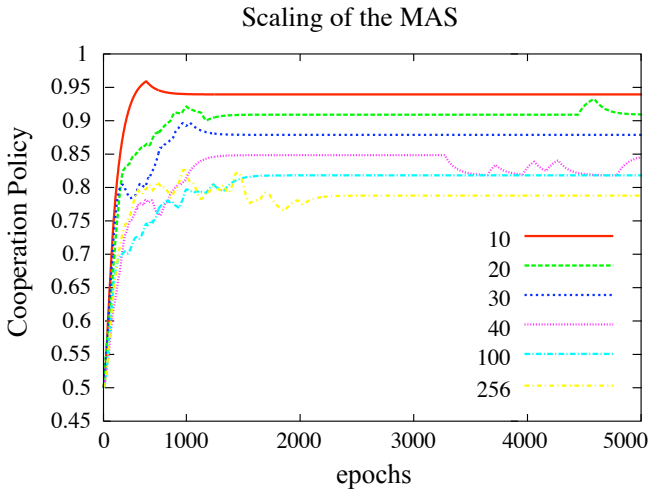

Fig. 2. Scaling of the FPGRL agents in the NIPD-AP game for MAS settings of different sizes, respectively 10, 20, 30, 40, 100 and 256 FPGRL agents. Plotted is the value for the cooperation policy μ_i, averaged over all agents i in the MAS, vs the number of epochs the system has run.

Figure 2 plot the amount of cooperation in a NIPD-AP game achieved in a MAS with respectively 10, 20, 30, 40, 100 and 256 FPGRL agents. It shows that substantial groups of agents each using FPGRL can learn to cooperate with very little public information, as in the NIPD-AP(1:1) game. Full cooperation is approximately achieved for up to 20 agents. Sufficiently many agents dispense altruistic punishment, in that they compute that the benefit from altering another agent's behavior outweighs the immediate cost of punishing it. When the number of agents in the MAS is increased, up to 256 FPGRL agents, the agents still find a high level of cooperation, though the amount of cooperation does slowly decreases with increasing agents in MAS as compared to cooperation in a MAS with fewer agents (a MAS with 256 FPGRL agents was the limit of available memory in the simulation). In line with the arguments of Wolpert & Tumer (D. Wolpert & Tumer, 2002), we believe the gradual decrease in performance as the number of agents increases is due to the fact that the noise in the signal of rewards increases with the number of agents.

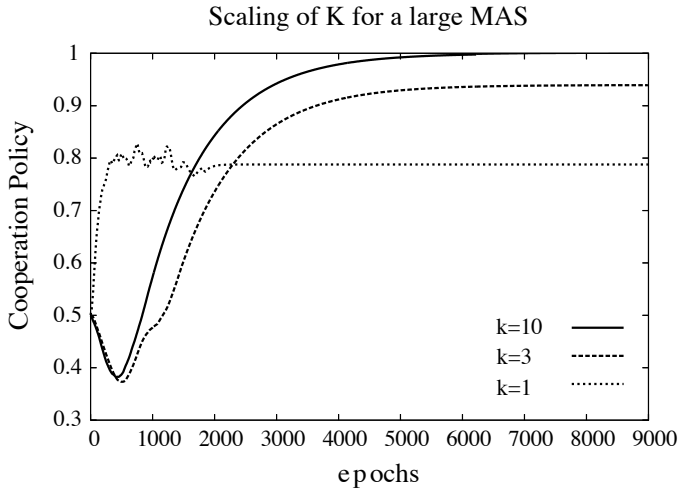

Fig. 3. Scaling of the FPGRL agents for 256 agents in the NIPD-AP(1:K) game: $K = 1$, $K = 5$, $K = 10$. Plotted is the value for the cooperation policy μ_i, averaged over all agents i in the MAS, vs the number of epochs the system has run.

Increasing the amounts of public information available enables increasingly large groups to sustain full cooperation, as is shown in the NIPD-AP(1:K) setting. In Figure 3, we show results for 256 agents playing the NIPD-AP(1:K) game for $K = 1, K = 3$ and $K = 10$. The high level op cooperation is however achieved at the expense of a longer learning period and a more pronounced drop in the initial level of cooperation. As a function of increasing values for K, individual agents will learn to punish less as the load of punishing a defector is spread over several agents (not shown). However, the FPGRL algorithm computes that at least some altruistic punishment is profitable for an individual agent, and a sufficient level of punishment is maintained to enforce cooperation.

In Figure 4(a), we compare the joint policy development in a MAS with agents using the FPGRL algorithm versus a MAS with agents employing a Best-Response type policy gradient reinforcement learning algorithm. Shown are the results for four agents collectively playing the NIPD-AP(1:1) game; plotted is the cooperation and punishment policy for an agent in the FPGRL MAS (FPGRL Coop and FPGRL punishment respectively), and the same policies for an agent in the MAS with agents using Best-Response policy gradient reinforcement learning (PGRL Coop and PGRL Punish)[3]. As before, in the MAS with agents using FPGRL, the agents converge to a sustained high level of cooperation, whereas in the Best-Response MAS cooperation is not sustained.

This result for the best-response PGRL approach was representative for all variants of the NIPD-AP, regardless of the number of agents, and, as we argued,

[3] The PGRL type agent is implemented by setting the ξ equation of Equation 4 with $\hat{\mu}_{-i,t+1} = \xi(\hat{\mu}_{-i,t}, a_i)$ to return $\hat{\mu}_{-i,t}$ as the estimate of the next, unchanged policy of the opponents in Algorithm 1, this corresponds to playing a Best Response policy.

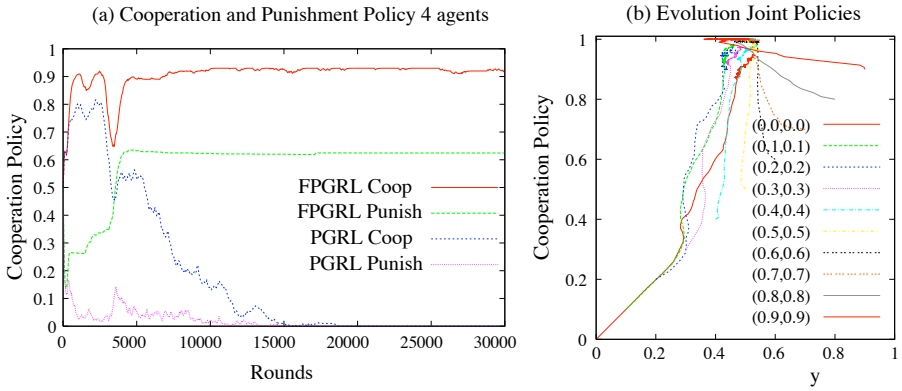

Fig. 4. (a)Learned Policies in a 4 agent NIPD-AP, FPGRL vs PGRL. Top to bottom legend order follows plotted lines top down. (b) A plot of the joint cooperation/punishment policy as learned over time, for a single participating agent, for various initial $(coop, punish)$ settings.

is typical for all types of myopic "follow the gradient" reinforcement learning type approaches. The personal cost that an agent incurs for punishing results in a negative gradient for the punishment policy, and punishment of defectors will disappear from the system. The myopic agents then accurately find the Pareto-deficient NE and all Defect.

In Figure 4(b), we show that the learned policies of the FPGRL agents are robust to the initialization of the cooperation and punishment policies. Plotted is the development over time of the joint punishment and cooperation policies for various initial settings of the cooperation and punishment policies. For all initial settings, the agents learn to cooperate and to punish. The agents typically overshoot in the required level of punishment, and then reduce this to a lower, less costly level. This reduction halts as the agents experience a drop in cooperation the cost of which exceed the cost of altruistic punishment. The point $(1, 0.5)$ acts as an equilibrium attractor of the system with converged agents shifting around this point due to exploratory moves and (inherent) inexact opponent modeling.

Lastly, in Figure 5 we show results for up to 256 agents punishing other agents according to the NIPD-AP$(1{:}\sim\text{N})$ model. Here, an agent's altruistic punishment hurts not only the defector, but also all agents in the MAS. A steep drop in initial cooperation is experienced before the FPGRL agents learn to cooperate. Compared to the significantly slower convergence in the NIPD-AP$(1{:}\text{K})$ game, the NIPD-AP$(1{:}\sim\text{N})$ model demonstrates the robust learning signal that "punishing" conveys to the group of agents as a whole. The scalability and practicality of this approach however is somewhat questionable, as the rapid initial decrease of cooperation to almost zero, as a function of the number participating agents, is a reason for concern.

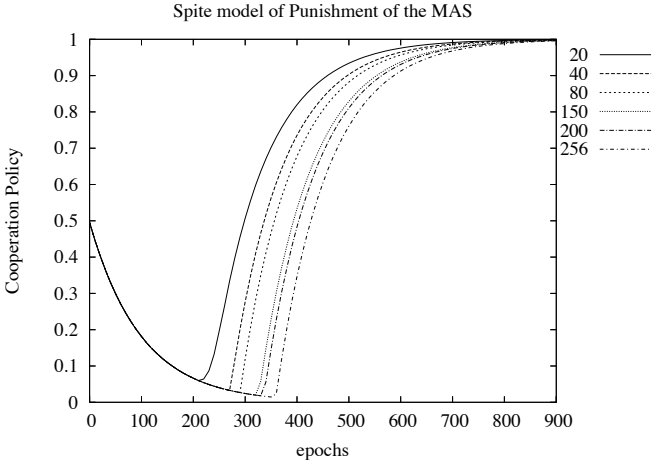

Fig. 5. Scaling of the FPGRL agents for the NIPD-AP(1:∼N) game.

5 Discussion

We have presented a foresighted policy gradient reinforcement learning algorithm, FPGRL, that enables individual self-interested agents in a multi-agent system to successfully sustain the long term Pareto-optimal strategy of full cooperation in the NIPD-AP game, with a large number of participating agents. The FPGRL algorithm advances on earlier foresighted reinforcement learning by developing a robust, policy gradient approach, thus enabling agents to learn with foresight in larger, more complex settings.

The FPGRL algorithm makes explicit the teach/learn dichotomy implicit in a system with many adapting agents (Shoham et al., 2007): an agent has to both learn how to behave profitably in the system, but can and will also modify (or: teach) the behavior of other agents due to its own actions. The FPGRL algorithm can be considered a first clear implementation of this realization.

In our setup, agents "teach" by possibly dispensing punishment. We assume explicitly that a single step punishment (or rather: the expectation thereof) may be sufficient to dissuade a targeted agent from defecting. The problem is thus reduced to the question whether or not to punish. This allows us to ignore the question of what particular punishment strategy should be followed – it is well known that formulating punishment strategies is itself a hard problem (Littman & Stone, 2005).

Considerable work has focused on "shaping" the reward that individual agents receive such that they are "nudged" in the direction of a desired joint action. Under certain conditions, it can be shown that reward shaping can be done without changing the optimal policy (Ng, Harada, & Russell, 1999). For example, in (Babes, Munoz de Cote, & Littman, 2008), a mutual beneficial subgame perfect equilibrium is used as the basis for a potential-based shaping function.

In a similar vein, the work by Wolpert *et al.* (D. H. Wolpert, Wheller, & Tumer, 1999; Lee & Wolpert, 2004) also focuses on shaping an agent's reward such as to include its effect on the collective system reward. Proper adjustment of this reward shaping aligns the learning of the individual agents with the maximization of the reward achieved by the collective of agents. Similarly, setting "aspiration levels" for individual agents as in (Macy & Flache, 2002) can be considered a variant of reward-shaping. However, these "reward shaping" approaches requires an externally imposed "rule".

We evaluated our FPGRL algorithm through MAS settings where agents all using FPGRL play against each other – *self-play*. One justification for this is that there are no robust multi-agent reinforcement learning algorithms available where the agents successfully solve the NIPD problem. The Joint-Action Learners in (Banerjee & Sen, 2007) are not very robust, and algorithms like M-Qubed (Crandall & Goodrich, 2005) do not readily scale beyond the 2-player IPD. Moreover, it is not obvious how the option of a "punishment" action should be incorporated in M-Qubed. As noted, there is considerable debate on how to evaluate different multi-agent reinforcement learning algorithms relative to each other (Shoham et al., 2007). Some criteria have been proposed, like robust results in self-play, best-response play against stationary opponents, and (ε-) no-regret (Bowling, 2005). The latter is obviously of limited use in NIPD type games; the approach presented here demonstrates learned cooperation in self-play, and converges to best-response play against stationary opponents.

Our findings support previous game theoretic considerations that seemingly altruistic punishment can be rational for an individual agent by enabling sustainable cooperation in large groups. This in contrast to the studies like (Boyd et al., 2003), where an additional disruptive mechanism of selection is required for successful scalable cooperation: in the evolutionary simulation, entire groups are occasionally continued or discontinued based on the joint group payoff. As noted, (Boyd et al., 2003) consider groups where many 2-player IPD interactions take place, whereas we consider the "Tragedy of the Commons" setting in the NIPD-AP game. We would argue that the latter setting is often of more importance in multi-agent systems.

Note that in groups with many simultaneous 2-player IPD interactions, the rational amount of altruistic punishment dispensed by an individual agent decreases with the group size (Kandori, 1992) (as the likelihood of an single agent interacting with a specific other agent decreases). This explains the limits to achievable cooperation observed in (Boyd et al., 2003).

The folk theorem implies that, in many games, there exists Nash-Equilibria (NE) for repeated games, repeated Nash-Equilibria (rNEs), that yield higher individual payoffs to all agents than do one-shot NEs, i.e. the rNE Pareto dominates the NE. Hence, in repeated games, a successful set of agents should learn to play profitable rNEs and the algorithm presented here is shown to allow agents to find the (single) Pareto Optimal rNE in the NIPD.

To be quite precise, by being "foresighted", we have shown that agents using FPGRL are able to find *a* rNE in the NIPD-AP game. Thus learning to *rationally*

sustain cooperation in the NIPD-AP is an important step forward. Since social dilemma's capture many hard-to-solve real-life problems, we believe this is highly significant contribution.

There are still however more complicated repeated games that have a large or even infinite number of rNEs, it is not obvious which rNE in such games would or *should* be learned. Apart from the difficulty in computing such rNEs (Littman & Stone, 2005), agents may have (private) preferences between different rNEs and prefer one above the other, if allowed by its opponents. Additionally, the more general question "when to teach?" can be asked. This is obviously dependent on the game structure, the internal actions available to agents, and the degree to which they anticipate future moves. We intend to pursue these directions in future work.

Another future challenge is to apply foresighted algorithms to larger and more complex state spaces. One direction may be to integrate our framework with new ideas about state space representations like Predictive State Representations (Wolfe, James, & Singh, 2005). As noted in (Milgrom et al., 1990), ultimately the generation and communication of information regarding other agents' actions are limiting factors in community enforcement, such as we consider here. Lessening this burden through institutionalization or interested/disinterested intermediation may enable further scaling up of successful cooperation among self-interested agents (e.g. (Biglaiser & Friedman, 1994)).

Acknowledgement

This work has been carried out under theme SEN4 "Computational Intelligence and Multi-Agent Games". Part of this research has been performed within the framework of the project "Distributed Engine for Advanced Logistics (DEAL)" funded by the E.E.T. program in the Netherlands. Work of SB is supported by NWO VENI grant 639.021.203.

References

Anderson, C. M., & Putterman, L. (2006). Do non-strategic sanctions obey the law of demand? *Games and Economic Behavior*, *54*, 1-24.

Andreoni, J., & Varian, H. (1999). Preplay contracting in the prisoners dilemma. *Proceedings National Academy of Sciences USA*, *96*(19), 10933 – 10938.

Axelrod, R. (1984). *The evolution of cooperation.* Basic Books, New York.

Babes, M., Munoz de Cote, E., & Littman, M. (2008, May). Social reward shaping in the prisoner's dilemma. In *Proc. of the seventh int. joint conference on autonomous agents and multi-agent systems (aamas'07), porto, portugal.*

Baird, L., & Moore, A. (1999). Gradient descent for general reinforcement learning. In *Proc. NIPS'98* (pp. 968–974).

Banerjee, D., & Sen, S. (2007). Reaching pareto-optimality in prisoner's dilemma using conditional joint action learning. *Autonomous Agents and Multi-Agent Systems*, *15*(1), 91-108.

Biglaiser, G., & Friedman, J. (1994). Middlemen as guarantors of quality. *Int. J. Industrial Organization*, *12*(4), 509–531.

Bowling, M. (2005). Convergence and no-regret in multiagent learning. In L. Saul, Y. Weiss, & L. Bottou (Eds.), *Advances in neural information processing systems 17* (p. 209-216). Cambridge, MA: MIT Press.

Boyd, R., Gintis, H., Bowles, S., & Richerson, P. J. (2003). The evolution of altruistic punishment. *Proc. Nat. Acad. Sci. USA*, *100*, 353-3535.

Crandall, J. W., & Goodrich, M. A. (2005). Learning to compete, compromise, and cooperate in repeated general-sum games. In *Proc. ICML'05* (pp. 161–168).

Fehr, E., & Fischbacher, U. (2003, October). The nature of human altruism. *Nature*, *425*, 785-791.

Fudenberg, D., & Maskin, E. (1986). The folk theorem in repeated games with discounting and with incomplete information. *Econometrica*, *54*, 533–554.

Gintis, H. (2000). *Game theory evolving: A problem centered introduction to modelling strategic behaviour.* Princeton University Press.

Hardin, G. (1968). The Tragedy of the Commons. *Science*, *162*, 1243–1248.

Kandori, M. (1992). Social norms and community enforcement. *Rev. Econ. Stud.*, *59*, 63–80.

Klein, B., & Leffler, K. (1981). The role of market forces in assuring contractual performance. *J. Political Econ.*, *89*, 615–641.

Lee, C. F., & Wolpert, D. H. (2004). Product distribution theory for control of multi-agent systems. In *Proceedings of the 3rd international joint conference on autonomous agents and multiagent systems (aamas 2004)* (p. 522-529). IEEE Computer Society.

Littman, M. L., & Stone, P. (2005). A polynomial-time nash equilibrium algorithm for repeated games. *Decision Support Systems*, *39*, 55–66.

Macy, M., & Flache, A. (2002). Learning dynamics in social dilemmas. *Proceedings of the National Academy of Sciences of the USA*, *99*, 7229–7236.

Milgrom, P., North, D., & Weingast, B. (1990). The role of institutions in the revival of trade: the law merchant, private judges, and the champagne fairs. *Economics and Politics*, *2*, 1–23.

Ng, A., Harada, D., & Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the sixteenth international conference on machine learning* (pp. 278 – 287).

Nowak, M., & Sigmund, K. (2005). Evolution of indirect reciprocity. *Nature*, *437*(27), 1291–98.

Rubinstein, A. (1979). Equilibrium in supergames with the overtaking criterion. *J. of Economic Theory*, *21*, 1–9.

Sen, S. (1996). Reciprocity: a foundational principle for promoting cooperative behavior among selfish agents. In *Proc ICMAS'96* (pp. 322–329).

Shapiro, C. (1983). Premiums for high quality products as returns to reputations. *Quart. J. Econ.*, *98*(4), 659–679.

Shoham, Y., Powers, R., & Grenager, T. (2007). If multi-agent learning is the answer, what is the question? *Artif. Intell.*, *171*(7), 365–377.

't Hoen, P., Bohte, S., & La Poutré, J. (2006a). Learning from induced changes in opponent (re)actions in multi-agent games. In *Proc. AAMAS'06* (pp. 728–735).

't Hoen, P., Bohte, S., & La Poutré, J. (2006b). Strategic foresighted learning in competitive multi-agent games. In *Proc. ECAI'06* (pp. 536–540).

Wolfe, B., James, M. R., & Singh, S. (2005). Learning predictive state representations in dynamical systems without reset. In *Proc. ICML'05* (pp. 980–987).

Wolpert, D., & Tumer, K. (2002). Collective intelligence, data routing, and braess' paradox. *Journal of Artificial Intelligence Research*, *16*, 359-387.

Wolpert, D. H., Wheller, K. R., & Tumer, K. (1999). General principles of learning-based multi-agent systems. In O. Etzioni, J. P. Müller, & J. M. Bradshaw (Eds.), *Proceedings of the third international conference on autonomous agents (agents'99)* (pp. 77–83). Seattle, WA, USA: ACM Press.

Yao, X., & Darwen, P. (1994). An experimental study of N-person iterated Prisoner's Dilemma games. *Informatica*, *18*(4), 435–450.