Using Adaptive Consultation of Experts to Improve Convergence Rates in Multiagent Learning

(Short Paper)

Greg Hines
Cheriton School of Computer Science
University of Waterloo
Waterloo, Canada
ggdhines@cs.uwaterloo.ca

Kate Larson
Cheriton School of Computer Science
University of Waterloo
Waterloo, Canada
klarson@cs.uwaterloo.ca

ABSTRACT

We present a regret-based multiagent learning algorithm which is provably guaranteed to converge (during self-play) to the set of Nash equilibrium in a wide class of games. Our algorithm, FRAME, consults *experts* in order to obtain strategy suggestions for agents. If the experts provide effective advice for the agent, then the learning process will quickly reach a desired outcome. If, however, the experts do not provide good advice, then the agents using our algorithm are still protected. We further expand our algorithm so that agents learn, not only how to play against the other agents in the environment, but also which experts are providing the most effective advice for the situation at hand.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms

Algorithms, Theory

Keywords

Multiagent Learning, Game Theory

1. INTRODUCTION

How and what agents should learn in the presence of others is one of the important questions in multiagent systems. The problem has been studied from several different perspectives, and in particular has garnished a lot of interest from both the game-theory community (see, for example, [4]) and the AI community (see, for example, [2]).

In this paper we investigate the problem of whether identical agents, who repeatedly play against each other, can learn to play strategies which form a Nash equilibrium (see, for example [2]). In particular, we are interested in settings where there are potentially more than two agents, and where agents have potentially more than just two actions to choose from. We are also interested in ensuring that agents learn to play a best response against stationary opponents.

Our learning procedure, a *Framework for Regret Annealing Methods using Experts* or *FRAME*, is a regret-based learning algorithm for repeated games which combines a greedy random sampling method with consultation of *experts*, that return strategy profiles. More importantly, by consulting carefully chosen experts we can greatly improve the convergence rate to Nash equilibria in self-play, but in the case where the experts do not return useful advice, then we still have guarantees that our algorithm will lead agents to a Nash equilibrium.

Cite as: Using Adaptive Consultation of Experts to Improve Convergence Rates in Multiagent Learning (Short Paper), Greg Hines and Kate Larson, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008, Estoril, Portugal, pp.1337-1340.

2. BACKGROUND

An n-player *stage game* is defined as $G = \langle N, A_1, \ldots, A_n, u_1, \ldots, u_n \rangle$ where $N = \{1, 2, \ldots, n\}$ is the set of agents participating in the game, and A_i is the set of possible actions that agent i can take. During the stage game, agents simultaneously choose to play actions and each agent receives a reward based on the joint action $a = (a_1, \ldots, a_n)$. In particular $u_i : A_1 \times \ldots \times A_n \rightarrow \mathbb{R}$ is the utility function for agent i, and so $u_i(a)$ is the reward that agent i receives if the joint action is a. With out loss of generality, we assume $u_i \in [0, 1]$. Agents play *strategies*, where a strategy, σ_i, of agent i is a probability distribution over action space A_i and $\sigma_i(a_j)$ denotes the probability with which agent i chooses to play action $a_j \in A_i$. We let Σ_i denote the strategy space of agent i, and let $\sigma = (\sigma_1, \ldots, \sigma_n) \in \Sigma = \Sigma_1 \times \ldots \times \Sigma_n$ denote a joint strategy. If there exists an action a_j such that $\sigma_i(a_j) = 1$, then σ_i is called a pure strategy. We use the notation $\sigma = (\sigma_i, \sigma_{-i})$ to represent a joint strategy, where σ_{-i} is defined to be equal to $(\sigma_1, \ldots, \sigma_{i-1}, \sigma_{i+1}, \ldots, \sigma_n)$. By abuse of notation, we can define the utility of an agent in terms of a joint strategy $\sigma = (\sigma_i, \sigma_{-i})$ as

$$u_i(\sigma_i, \sigma_{-i}) = \sum_{a \in A} u_i(a) \Pi_{j=1}^{n} \sigma_j(a_j).$$

We assume that agents are self-interested and that they wish to play strategies that maximize their own utility. That is, if all agents but i are playing σ_{-i}, then agent i should play a strategy σ_i that maximizes its utility, i.e. σ_i should be a best response to σ_{-i}. We say that agents' strategies are in (Nash) equilibrium if no agent is willing to change their strategy, given that no other agents change.

DEFINITION 1. *A strategy profile $\sigma^* = (\sigma_1^*, \ldots, \sigma_n^*)$ is a*

Nash equilibrium if for every agent i

$$u_i(\sigma_i^*, \sigma_{-i}^*) \geq u_i(\sigma_i', \sigma_{-i}^*) \ \forall \sigma_i' \neq \sigma_i^*.$$

A strategy profile σ^* is an ϵ-Nash equilibrium if for every agent i, $u_i(\sigma_i^*, \sigma_{-i}^*) \geq u_i(\sigma_i', \sigma_{-i}^*) - \epsilon \ \forall \sigma_i' \neq \sigma_i^*$.

Agents are also able to evaluate their strategy choice by measuring the *regret* they experience from playing a particular strategy.

DEFINITION 2. *Given a joint strategy σ, agent i's regret is $r_i(\sigma) = \max_{\sigma_i' \in \Sigma_i} [u_i(\sigma_i', \sigma_{-i}) - u_i(\sigma_i, \sigma_{-i})]$.*

Given σ, we define the *regret of a game* to be the maximum regret among all agents, i.e. $r(\sigma) = \max_{i \in N}(r_i(\sigma))$.

A repeated game, $\mathcal{G}$, is a game where agents play a specific stage game over and over. At stage t we denote the strategy profile that the agents played by σ^t and the actual action profile that the agents played by a^t. Given σ^t, each agent i is able to compute its immediate regret, $r_i(\sigma^t)$.

As the stage game is repeated, agents gain experience and are able to adjust their strategies so that they fair better against their opponents. In this paper we are interested in learning approaches which use *regret*, and in particular regret-minimization, to guide the agents' strategy adaptations. Our goal is to develop a learning procedure which will converge to an interesting set of strategies for the agents. In particular, we would like to develop an approach such that $r(\sigma^t) \to 0$ as $t \to \infty$ (*i.e.* the process converges to the set of Nash equilibria for the stage game).

Regret-based learning is a broad type of learning that can achieve various degrees of convergence. However, the results for achieving convergence to the set of Nash equilibria are mostly negative. Some positive results have been achieved using *randomized* learning algorithms. One example of this approach is *Experimental Regret Technique (ERT)* [5]. The basic idea of ERT is to have all agents with high regret randomly choose a new strategy, to have all agents with medium regret to slightly modify their current strategy in some systematic way, and to have agents with low regret to keep playing their strategy. Germano and Lugosi further improved upon this technique with their algorithm Annealed Localized Experimental Regret Technique (ALERT) which provably converges to the set of Nash equilibria for almost all games and the set of ϵ-Nash equilibria for all games.

3. FRAME

Although ALERT is theoretically important, there are two main issues which limit its applicability in actual multiagent systems. First, since ALERT is an uncoupled algorithm, agents have almost no information from which they can determine whether they are playing an ϵ-equilibrium. Instead, ALERT's guarantees are in the form of bounds on the probability of *not* being in an ϵ-equilibrium. Second, ALERT uses a naive method for having agents find new strategies. In particular, ALERT has the agents choose new strategies uniformally at random and then checks whether these strategies meet a set of conditions. Our algorithm, a *Framework for Regret Annealing Methods using Experts*, or *FRAME*, is inspired by ALERT but explicitly addresses these two issues, while still maintaining the theoretical guarantees of ALERT.

To address the first issue, FRAME is not a fully uncoupled algorithm. Instead, we assume that the agents' strategies are publicly available to all agents, as is done by several other researchers [2]. We also assume that the maximum regret of all agents is publicly available. Our algorithm will still work without these two assumptions, as it is possible to experimentally determine regret (both for individual agents and overall), but this comes with a substantial increase in the number of iterations required by our algorithm.

Algorithm 1 $FRAME_i$

- σ_i^0 is a strategy picked uniformly at random
for $t = 0, 1, \ldots$ **do**
  - with probability p, β_i^{t+1} is the strategy returned by consulting the expert
  if β_i^{t+1} is not in the bounded region $B(\sigma_i^t, d(r(\sigma^t)))$ or the expert was not consulted **then**
    - β_i^{t+1} is the strategy picked uniformly from $B(\sigma_i^t, d(r(\sigma^t)))$
  end if
  if the regret of β is less than the regret of σ^t **then**
    - $\sigma^{t+1} = \beta^{t+1}$
  else
    - $\sigma^{t+1} = \sigma^t$
  end if
  - τ_i is strategy picked uniformly at random from Σ_i
  if the regret of τ is less than half the regret of σ^{t+1} **then**
    - with probability η, set $\sigma^{t+1} = \tau$.
  end if
end for

To deal with the second issue, FRAME allows an agent, with some probability, to consult an expert, which returns a possible new strategy. Any expert will work, even a malicious one that actively provides bad strategies. If the expert provides good strategies, then FRAME will be able to reduce an agent's regret quickly. If all agents are using FRAME and are consulting good experts, then the convergence rate to a Nash equilibrium greatly improves.

The FRAME algorithm for agent i is shown in Algorithm 1. The algorithm, with respect to agent i, works as follows. Agent i first chooses an initial strategy σ_i^0 uniformly at random from Σ_i. To obtain a new strategy for time $t+1$, FRAME then uses the provided expert, which agent i consults with a provided probability of p, independent of all other agents. If consulted, the expert returns a possible strategy β_i^{t+1}. To provide protection against poor experts, FRAME checks to see if β_i^{t+1} is inside the bounded region $B(\sigma_i^t, d(r^t))$, which is centered on σ_i^t and has a minimum width of $d()$.[1] If β_i^{t+1} is not, or the expert was not consulted, β_i^{t+1} is chosen uniformly at random from the bounded search region. Agent i then calculates $r_i(\beta^{t+1})$. If $r(\beta^{t+1}) < r(\sigma^t)$, then $\sigma^{t+1} = \beta^{t+1}$, otherwise, $\sigma^{t+1} = \sigma^t$. To avoid the off-chance of getting stuck at a locally optimal joint strategy, each agent chooses an alternative strategy τ_i uniformly at random from Σ_i. If the regret at τ is less than half the current regret, then with a given probability η, the game *resets* to τ. This process repeats until the regret is zero.

FRAME's correctness is provided by Proposition 1.

PROPOSITION 1. *If $\eta > 0$, then as t approaches infinity, σ^t approaches the set of Nash equilibria.*

The proof is omitted due to space limitations.

[1] $d()$ may be any function so long as $d(x) > 0$, for $x > 0$.

0, 0	1, 0	0, 1
0, 1	0, 0	1, 0
1, 0	0, 1	0, 0

Figure 1: Shapley's Game.

It should be noted that FRAME also works when some subset of the agents are playing stationary strategies. Specifically, agents using FRAME are able to achieve a best response against those agents playing stationary strategies.

3.1 Experimental Results

In this section we discuss our findings from a series of experiments.

3.1.1 Experimental Setup

While in theory any expert will work in FRAME, methods that make gradual adjustments to the strategies of agents are preferred. In our experiments we chose two such experts; Win or Learn Fast (WoLF) [2] and Logistic Fictitious Play (LFP) [4]. As a basis for comparison, we also used the *Naive Expert*, which always picks a strategy at random.

WoLF is a variable learning rate applied to a gradient-ascent learning approach. Each turn the strategy is moved towards a best response, however the strategy is moved more aggresively when the agent is doing worse than expected.

LFP is a form of learning where, at each iteration, the agent chooses a particular action with a probability that is in proportion to an exponential function of the utility that this action has yielded in the past.

We ran experiments on a wide range of games, including repeated Prisoner's Dilemma, Battle of the Sexes, 2-Player Matching Pennies and 3-Player Chicken. Due to space limitations we are unable to report our findings in these games in any detail, except to say that in self-play, agents using FRAME were able to quickly converge to Nash equilibria. We report, in detail, our findings from Shapley's game (Figure 1). Shapley's game is a classic but challenging one. In particular, WoLF does not converge in Shapley's game whereas LFP does.

For our experiments, LFP was run with $\lambda = 0.5$ and WoLF with $\delta_w = \frac{1}{100+t}$ and $\delta_l = 3\delta_w$. For FRAME, we let $p = 0.75$.

3.1.2 Results

A trial was said to have converged when the joint strategy was within three decimal places of any Nash equilibrium. Each of our experiments consisted of 1000 trials. We present our findings in a histogram format, which show the percentage of each experiment (grouped into 25 bins) that took a certain number of iterations to converge.

As shown in Figure 2, convergence in Shapley's Game is achieved using just a Naive Expert. However, by picking a better expert, we can do much better. Figure 2, shows the convergence when LFP is used as the expert and consulted 75% of the time. The convergence rate improves by three orders of magnitudes. We also conducted other experiments which showed that as LFP was consulted more and more often, the convergence continued to improve. On the other hand, Figure 2, shows the convergence rate when an expert poorly suited for Shapley's game, such as WoLF, is used as the expert, the convergence rate suffers but convergence is still achieved.

4. ADAPTIVE-FRAME

Despite the success of FRAME, it has one fundamental limitation. As our experiments showed, any specific expert is only useful for a limited set of games. Hence, once an agent picks its expert, it has limited the set of games for which it can achieve good convergence rates. Furthermore, even if an agent was allowed to pick a new expert for each game, it would not always be possible to know, before the game started, which expert was best to use.

To address this problem, we created a generalization of FRAME called adaptive-FRAME. Adaptive-FRAME allows an agent, at any point in a game, to choose from many possible experts. To help agents make the decision of which expert to actually consult, agents make use of an *experts algorithm*. An experts algorithm is any algorithm that, given a set of experts and their past performances, suggests which expert to consult. This allows adaptive-FRAME the flexibility to deal with new and unknown games.

Formally, the set of possible experts for agent i to consult is denoted by $E_i = \{e_{i,0}, \ldots, e_{i,|E_i|-1}\}$. The Naive Expert is always $e_{i,0}$. With slight abuse of notation, we define e_i to be some specific but undefined expert for agent i. At time t expert e_i is consulted with probability $p_i^t(e_i)$ and returns a suggested strategy β_{e_i}. Agent i's experts algorithm is denoted by æ$_i$ and p_i is called æ$_i$'s policy. We only require that for all t, $p^t_{e_{i,0}} > 0$ and $\sum_{t=0}^{\infty} p^t_{e_{i,0}} = \infty$; as long as this holds, the correctness for adaptive-FRAME follows directly from the proof of correctness for FRAME.

4.1 LERRM

To create a MAL experts algorithm, we first need a useful way of measuring performance of the experts. Since the goal of experts is to try and reduce an agent's regret, we created a metric, *Expected Regret Reduction* (ERR), defined as

$$ERR(e_i)_i^T = \frac{\sum_{t=0}^{T-1}(r(\beta^t)_i^t - r(\beta_{e_i}^{t+1}, \beta_{-i}^{t+1})_i^{t+1})}{T}.$$

ERR estimates expert e_i's ability to reduce an agent's regret over some time period $\{0, \ldots, T\}$ by assuming that all other agents' strategies are fixed but that e_i's suggested strategies were always followed. ERR then calculates the average reduction in regret e_i's strategies would have achieved.

Our experts algorithm, *Logistic Expected Regret Reduction Maximization* (LERRM), is based on the idea of LFP;

$$LERRM(e_i)_i^t = \frac{e^{\frac{1}{\lambda}ERR(e_i)_i^t}}{\sum_{e_i' \in E_i} e^{\frac{1}{\lambda}ERR(e_i')_i^t}}.$$

LERRM is designed as a general approach that can be used in other MAL settings.

4.2 Experimental Results

We tested adaptive-FRAME using Shapley's game. We tested three different experts algorithms. The Naive Experts Algorithm, which chooses each expert with equal probability, served as a benchmark by which to compare the others. Besides our experts algorithm, LERRM, we also used Hedge, a standard experts algorithm [3]. Hedge assigns "weights" to each expert and then consults an expert with a probability equal to that expert's weight proportional to all of the weights.

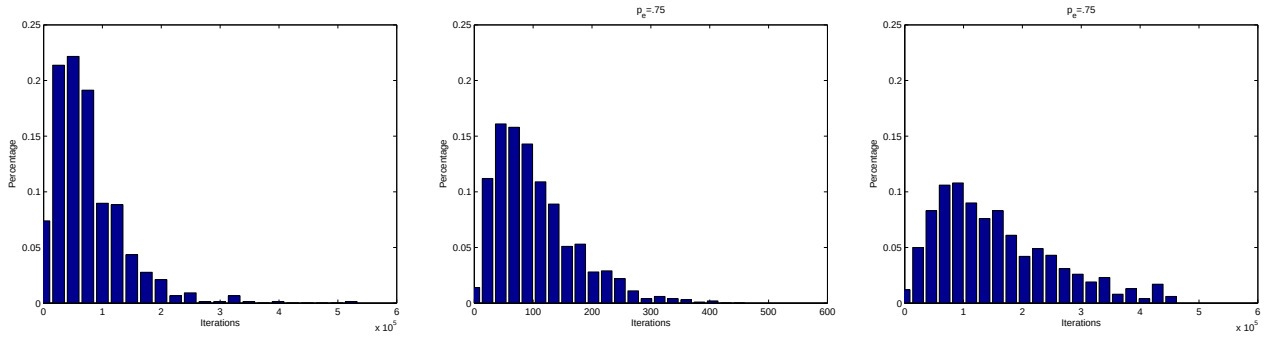

Figure 2: Convergence Rates for Shapley's Game using FRAME with the Naive Expert, LFP and WoLF, respectively. Note the difference in order of magnitude for the results for LFP.

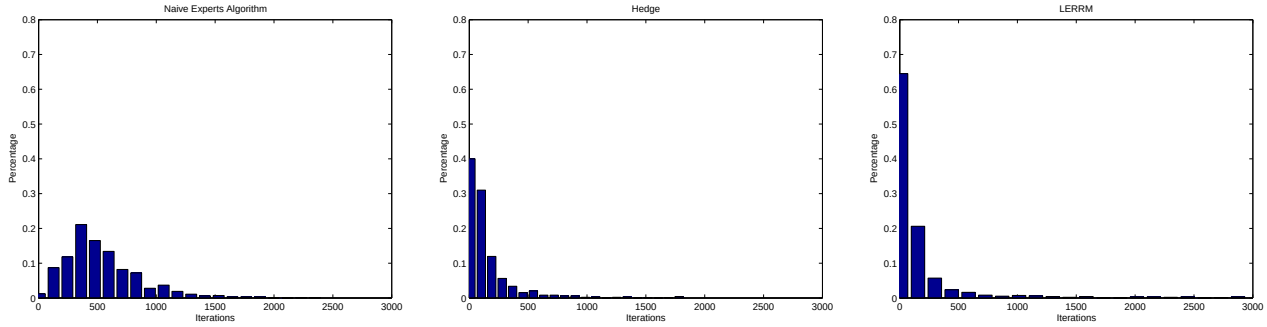

Figure 3: Convergence Rates for Shapley's Game using adaptive-FRAME with various experts algorithms.

4.2.1 Results

As seen by comparing Figure 3 to the results in Figure 2, all three of the experts algorithms do much better than the worst expert. Hedge and LERRM give, on average, much faster convergence rates compared to the Naive Experts Algorithm. In particular LERRM performs very well.

How are Hedge and LERRM able to achieve this performance? Since LFP is the best-suited expert for this game, Hedge and LERRM should consult LFP with high probability and WoLF with low probability. Our experimental results confirm this. At the point of convergence, Hedge was consulting LFP almost exclusively 20% of the time and LERRM consulted LFP almost exclusively 90% of the time. This difference helps explain why LERRM out performed Hedge.

5. CONCLUSION

In this paper we introduced two new multiagent learning algorithms, FRAME and adaptive-FRAME, and showed that, under certain assumptions, agents using either of these algorithms in self-play will converge to the set of Nash equilibria. The key idea of FRAME is that it will sometimes consult experts. If the expert is an effective learning procedure itself, then FRAME will also be effective. However, if the expert performs poorly, then FRAME's theoretical properties still hold, and in particular FRAME is still guaranteed to converge to a Nash equilibrium. The key idea of adaptive-FRAME is to allow agents the possibility of consulting different experts. Furthermore, agents can use experts algorithms to help them decide which expert to consult.

There are several research directions which we intend to pursue. First, there are several other experts, each specializing in their own class of games, that could be used [1]. By combining experts we might be able to create a powerful and highly effective general learning procedure.

6. REFERENCES

[1] B. Banerjee and J. Peng. $RV_{\sigma(t)}$: A unifying approach to performance and convergence in online multiagent learning. In *Proceedings of AAMAS-2006*, pages 2–7, Hakodate, Japan, 2006.

[2] M. Bowling and M. Veloso. Multiagent learning using a variable learning rate. *Artificial Intelligence*, 136:215–250, 2002.

[3] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[4] D. Fudenberg and D. Levine. *The Theory of Learning in Games*. MIT Press, 1998.

[5] F. Germano and G. Lugosi. Global Nash convergence of Foster and Young's regret testing. *Games and Economic Behavior*, 60(1):135–154, 2007.