Counterfactual Regret Minimization for Decentralized Planning

Bikramjit Banerjee
The University of Southern Mississippi
118 College Dr. #5106
Hattiesburg, MS 39402
Bikramjit.Banerjee@usm.edu

Landon Kraemer
The University of Southern Mississippi
118 College Dr. #5106
Hattiesburg, MS 39402
Landon.Kraemer@eagles.usm.edu

ABSTRACT

Regret minimization is an effective technique for almost surely producing Nash equilibrium policies in coordination games in the strategic form. Decentralized POMDPs offer a realistic model for sequential coordination problems, but they yield doubly exponential sized games in the strategic form. Recently, counterfactual regret has offered a way to decompose total regret along a (extensive form) game tree into components that can be individually controlled, such that minimizing all of them minimizes the total regret as well. However, a straightforward extension of this decomposition in decentralized POMDPs leads to a complexity exponential in both the joint action and joint observation spaces. We present a more tractable approach to regret minimization where the regret is decomposed along the nodes of agents' policy trees that yields a complexity exponential only in the joint observation space. We present an algorithm, REMIT, to minimize regret by this decomposition and prove that it converges to a Nash equilibrium policy in the limit. We also use a stronger convergence criterion with REMIT, such that if this criterion is met then the algorithm must output a Nash equilibrium policy in finite time. We found empirically that in every benchmark problems that we tested, this criterion was indeed met and (near) optimal Nash equilibrium policies were achieved.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems*; I.2.8 [**Problem Solving, Control Methods, and Search**]:

General Terms

Algorithms, Experimentation, Performance

Keywords

Game and decision theory, Decentralized partially observable Markov decision processes

INTRODUCTION

Decentralized partially observable Markov decision processes (Dec-POMDPs) offer a powerful and realistic model for multi-agent coordination problems. On the one hand many exact solvers have been developed for Dec-POMDPs, which yield the optimal Nash equilibrium solution, but are highly inefficient due to the inherent complexity of the problem [18, 17]. On the other hand, many approximate solvers are known that produce high quality solutions more efficiently, but generally do not guarantee that the returned solution will be a Nash equilibrium [15, 9]. There also exist *local search* techniques, which aim for a locally optimal Nash equilibrium solution [11, 13]. Stability of the solution is not important for the exact or approximate solvers, since they are evaluated by the quality of the solution produced, and any agent's incentive for deviation at policy execution time can only improve the group utility. However, stability is a key question for local search techniques. Since a local optimum is their logical objective, it is useful to know whether they reach such a solution. Our work falls in the category of local search techniques. In this paper we examine the application of *regret minimization* [19] – an effective (and often efficient) class of techniques from the literature on learning in games – to Dec-POMDPs.

Recently, regret minimization has been shown to almost surely achieve Nash equilibrium solution in strategic form coordination games [10]. We argue that while Dec-POMDPs can be represented as strategic form coordination games, the resulting strategy space would be hopelessly intractable for the application of regret-based techniques. On the other hand, a recent approach called counterfactual regret which has been very successful in poker variants, has shown a more compact way to minimize regret, albeit in extensive form competitive game trees. The main idea is to decompose overall regret into components that can be minimized independently. A straightforward extension of this method to Dec-POMDPs would decompose total regret along the histories of action-observations for each agent, which would yield a complexity exponential in both joint action and joint observation spaces. We present a more compact way to decompose total regret in Dec-POMDPs – one where the individually controllable components are defined at each node of each agent's policy tree. We show that this decomposition is indeed valid, i.e., minimizing all component regrets will indeed minimize the total regret. We present an algorithm, REMIT, for minimizing regret components, with a complexity that is exponential only in the joint observation space, and show that it converges to a Nash equilibrium policy in the limit. We also present a stronger convergence criterion, that, if satisfied, will yield a Nash equilibrium in finite time. We show experimentally that this criterion is indeed satisfied in a range of benchmark problems, with REMIT producing

Appears in *The Eighth Annual Workshop on Multiagent Sequential Decision-Making Under Uncertainty* (*MSDM-2013*), held in conjunction with *AAMAS,* May 2013, St. Paul, Minnesota, USA.

(near) optimal Nash equilibrium policies.

A major motivation for investigating regret based techniques is the fact that they are often precursors to effective sample based reinforcement learning (RL) algorithms. A most recent RL algorithm for Dec-POMDPs has agents learning alternately, and has a complexity exponential in both joint action and joint observation spaces for each agent's learning phase [2]. Therefore, a sampling based variant of REMIT, where agents can also update simultaneously, holds the promise of being relatively more scalable. This paper establishes some of the theoretical substrate on which such a distributed learning approach could be built.

BACKGROUND

Dec-POMDPs

A Decentralized POMDP (Dec-POMDP) is defined as a tuple $\langle k, S, b_0, A, P, R, \Omega, O \rangle$, where:

- k is the number of agents in the system.
- S is a finite set of (unobservable) environment states.
- $b_0 \in \Delta(S)$ is the initial belief state.
- $A = \times_i A_i$ is a set of joint actions, where A_i is the set of individual actions that agent i can execute.
- $P(s'|s, a)$ gives the probability of transitioning to state $s' \in S$ when joint action $a \in A$ is taken in state $s \in S$.
- $R : S \times A \rightarrow \Re$, where $R(s, a)$ gives the immediate reward the agents receive upon executing joint action $a \in A$ in state $s \in S$.
- $\Omega = \times_i \Omega_i$ is the set of joint observations, where Ω_i is the finite set of individual observations that agent i can receive from the environment.
- $O(\omega|s', a)$ gives the probability of the agents jointly observing $\omega \in \Omega$ if the current state is $s' \in S$ and the previous joint action was $a \in A$.

Additionally, a horizon T is also specified for finite horizon problems, and the transition and observation probabilities are often jointly represented as $P(s', \omega|s, a)$. The reward function R, transition model P, and observation model O are defined over joint actions and/or observations, which forces the agents to coordinate. The goal of the Dec-POMDP problem is to find a set of policies – one for each agent, π_i – that maximizes the total expected reward over T steps of interaction, given that the agents cannot communicate their observations and actions to each other. For finite horizon problems, π_i is a mapping from the histories of action-observation pairs of agent i to actions in A_i. A t-step history is represented as $h_t = (a^0, \omega^0, \ldots, a^{t-1}, \omega^{t-1})$. Figure 1 shows the example of an agent's policy in the DEC-TIGER domain for horizon $T = 3$, with the policy represented as a mapping from histories to actions on the left, and equivalently as a policy tree on the right. In this paper we will refer to a policy π_i in the tree form. The problem of finding an optimal joint policy (i.e., set of trees, one for each agent) has been proven to be NEXP-complete [3].

$$
\begin{aligned}
() &\rightarrow \text{L} \\
(L, HL) &\rightarrow \text{L} \\
(L, HL, L, HL) &\rightarrow \text{OR} \\
(L, HL, L, HR) &\rightarrow \text{L} \\
(L, HR) &\rightarrow \text{L} \\
(L, HR, L, HL) &\rightarrow \text{L} \\
(L, HR, L, HR) &\rightarrow \text{OL}
\end{aligned}
$$

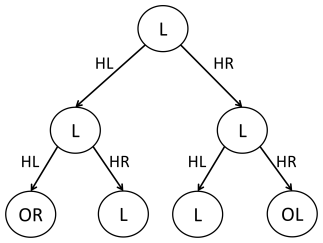

Figure 1: Two equivalent representations of an example policy for the single-agent tiger problem. The goal is to open a door that conceals treasure, instead of one that hides a tiger. The agent can execute actions listen (L), open right door (OR) or open left door (OL). It can hear the tiger's growl behind the left door (HL) or the right door (HR).

Strategic Games and Regret

A game in strategic form has three components: a set of players $\{1, 2, \ldots k\}$, a *pure strategy* space S_i for each player i, and payoff functions u_i that specifies the ith player's Von Neumann-Morgenstern utility $u_i(s)$ for each strategy profile $s = (s_1, s_2, \ldots s_k)$. In this paper, we are particularly interested in cooperative games of identical payoffs, i.e., where $u_i(s) = u(s)$ for all i, s. Henceforth we will only use this common utility function. We follow the game theoretic convention of representing all players except i as $-i$. Also, a variable will indicate a joint over all agents, unless subscripted by an index i or $-i$.

A *mixed* strategy of player i, σ_i, is a probability distribution over i's pure strategies, i.e., $\sigma_i \in \Delta(S_i)$. It is well-known that every finite strategic form game has at least one solution in the form of a mixed strategy Nash equilibrium [12], given by $\sigma_i^* \in \Delta(S_i)$ for all i, such that

$$u(\sigma_i^*, \sigma_{-i}^*) \geq u(s_i, \sigma_{-i}^*),\ \forall s_i \in S_i.$$

It is known that in games of identical payoffs, at least one pure strategy Nash equilibrium always exists. However, there may be multiple equilibria, so the goal is generally to find the (Pareto) optimal one.

The notion of *regret* is one of the key concepts to learning in games [19]. A player's regret for not playing a certain strategy in the past is determined by the ex-post amount of utility improvement that would have obtained had the player played that fixed strategy in every past round instead of the actual strategies that he did. Formally, given the sequence of mixed strategy profiles actually played for τ rounds, $\sigma^1, \ldots, \sigma^\tau$, average overall regret of player i is

$$R_i^\tau = \frac{1}{\tau} \max_{\sigma_i^*} \sum_{t=1}^{t=\tau} (u(\sigma_i^*, \sigma_{-i}^t) - u(\sigma^t)) \tag{1}$$

Obviously there is no way to know whether the other agents, $-i$, would have played the same sequence $\{\sigma_{-i}^t\}$ had i played σ_i^* all along, or would have responded differently to the hypothetical change. It turns out, however, that it is possible to drive the average regret R_i^τ to zero, *irrespective of* $\{\sigma_{-i}^t\}$. A simple yet elegant way to do this is *regret matching* [8] which updates

$$\sigma_i^{\tau+1}(s_i) = \frac{[R_i^\tau(s_i)]_+}{\sum_{s_j \in S_i} [R_i^\tau(s_j)]_+} \tag{2}$$

where

$$R_i^\tau(s_i) = \frac{1}{\tau}\sum_{t=1}^{t=\tau}(u(s_i, \sigma_{-i}^t) - u(\sigma^t))$$

and $[x]_+ = \max(x, 0)$. If the denominator of equation 2 is zero, the probabilities are set arbitrarily.

[8] has proven that in a finite general game, regret matching by a given player almost surely yields no regret against every possible sequence $\{\sigma_{-i}^t\}$. A direct consequence of this is that if all players use regret matching, then the empirical distribution of joint strategies converges almost surely to the set of *coarse correlated equilibria* (CCE) [8].

This result in general games is rather weak in two respects: first the CCE is a (potentially vast) superset of Nash equilibria, and secondly the almost sure convergence is in terms of empirical play, not the σ^t directly. Recently, [10] has shown that in the special class of *weakly acyclic games*, a regret minimizing sequence $\{\sigma^t\}$ almost surely converges to a pure strategy Nash equilibrium. The set of weakly acyclic games contains games of identical payoffs as a special case, hence this stronger convergence result carries over to these games.

STRATEGIC GAME REPRESENTATION OF DEC-POMDPS

When a horizon T is specified, a Dec-POMDP can be converted to a strategic form game with identical payoffs, where each possible complete conditional T-step plan of agent i (i.e., policy tree π_i as shown in Figure 1) in the former is a separate strategy, s_i, for that agent in the latter [7], and the common values (for all agents) of each joint policy tree gives the identical payoffs of all agents for that joint strategy. Then by [10], regret minimization in such a strategic form game should lead to a pure strategy Nash equilibrium, which corresponds to a set of policy trees – exactly one for each agent – as we would expect a Dec-POMDP solver to yield.

Unfortunately, the strategy space of each agent in such a strategic form game is of size $O(|A_*|^{|\Omega_*|^T})$, where A_*, Ω_* are the largest individual action and observation spaces. Even if dominated strategies are eliminated, there is no known result that ensures significant compaction. Therefore, direct regret minimization in the strategic form is infeasible. The main contribution of this paper is a more tractable alternative approach to regret minimization in a Dec-POMDP, that automatically accomplishes regret minimization in its strategic form.

REGRET DECOMPOSITION IN A TREE

The main idea is to decompose the overall regret into independently controllable components, such that minimizing the component-regrets independently also minimizes the overall regret.

Recently, regret decomposition in an extensive form game tree in competitive alternate move games with imperfect information – called *counterfactual regret* (CFR) – has found successful application in variants of Poker [20]. In CFR, the overall regret is decomposed along the *information sets* of the game, which roughly translates to individual histories in a Dec-POMDP. [1] also uses history based regrets in their mixed integer linear programming (MILP) formulation of Dec-POMDPs. However, there are $O(|A_*|^T|\Omega_*|^T)$ histories in a Dec-POMDP of horizon T. We propose a more compact decomposition of overall regret in a Dec-POMDP along the *nodes of the policy tree*, of which there are $O(|\Omega_*|^T)$, leading to more efficient regret minimization. In order to anchor the notion of regret on the nodes of a tree and in the context of Dec-POMDPs, we will introduce a set of notations below.

We consider a stochastic policy tree π_i (of agent i), where each node (contrast with Figure 1 right) n_i is labeled by a mixed strategy, $\sigma_i(n_i)$, instead of the special case of a pure strategy. A history leading to node n_i in a policy tree π_i is a sequence of pairs of mixed strategies and observations, notated as $h^{\pi_i}(n_i)$. Note that a node in π_i can be identified simply by traversing the sequence of observations in $h^{\pi_i}(n_i)$.

We define the utility function $v(\pi, n, s)$ to represent the expected payoff given that the joint policy π is being played, that joint nodes n in the respective policy trees have been reached, and that the system state is s. If $\pi(n)$ gives the (joint) policy subtrees rooted at the joint nodes $n = (n_1, n_2, \ldots, n_k)$, then v evaluates the expected payoff of the joint subpolicy $\pi(n)$. Note that individual nodes n_i must be at the same level in all agents' policy trees, and that $\pi(root)$ is simply notated as π. Specifically,

$$v(\pi, n, s) = \sum_a \sigma(n, a)[R(s, a) + \sum_{s',\omega} P(s', \omega|s, a)v(\pi, n', s')],$$

where n' is the joint successor node from n following ω, say $n' = succ(n, \omega)$.

Let $D(n_i)$ be the set of nodes reachable from n_i in π_i, i.e., the set of nodes in the subtree rooted at (and including) n_i. Let $\pi_i|_{D(n_i)\to\pi_i'}$ represent the policy which is identical to π_i except that the subtree rooted at n_i is the same as in another policy π_i'. Then, for agent i, the full regret for playing a sequence of subpolicies rooted at node n_i, $\{\pi_i^t(n_i)\}$, is

$$\begin{aligned} R_{i,\text{full}}^\tau(n_i) &= \frac{1}{\tau}\max_{\pi_i'}\sum_{t=1}^{t=\tau}\sum_{s,n_{-i}} P(s, h^{\pi_{-i}^t}(n_{-i})|h^{\pi_i^t}(n_i)) \cdot \\ &\quad [v(\pi_i^t|_{D(n_i)\to\pi_i'}, \pi_{-i}^t, n, s) - v(\pi^t, n, s)] \quad (3) \end{aligned}$$

Consistent with the counterfactuality of CFR, equation 3 represents the average regret of i for not playing the subpolicy of π_i' rooted at n_i while the others played π_{-i}^t, but given that i *played to reach* n_i. Note that the path of observations leading to n_i, while determined by nature, is indirectly controlled by all agents via their actions. However, the contribution of agent i to reaching n_i is discounted by conditioning on $h^{\pi_i^t}(n_i)$ to reflect this counterfactuality.

For the sake of notational consistency, verify that

$$\begin{aligned} R_{i,\text{full}}^\tau(\text{root}) &= \frac{1}{\tau}\max_{\pi_i'}\sum_{t,s} P(s, \emptyset|\emptyset) \cdot \\ &\quad [v(\pi_i^t|_{D(\text{root})\to\pi_i'}, \pi_{-i}^t, \text{root}, s) - v(\pi^t, \text{root}, s)] \\ &= \frac{1}{\tau}\max_{\pi_i'}\sum_{t,s} b_0(s) \cdot \\ &\quad [v(\pi_i', \pi_{-i}^t, \text{root}, s) - v(\pi^t, \text{root}, s)] \\ &= \frac{1}{\tau}\max_{\pi_i'}\sum_t (u(\pi_i', \pi_{-i}^t) - u(\pi^t)) \\ &= R_i^\tau \text{ from equation 1} \end{aligned}$$

Given that a pure policy equilibrium always exists in Dec-POMDPs, i.e., where the node distributions σ_is are pure, we

Algorithm 1 REMIT

1: Initialize policy trees π_i^0, $\forall i$
2: Initialize $R_i^0(n_i, a_i) = 0$, $\forall i, n_i, a_i$
3: $t \leftarrow 0$
4: **repeat**
5: **for** each agent i **do**
6: **for** each node n_i in π_i^t (breadth first order) **do**
7: $\forall a_i \in A_i$, $R_i^{t+1}(n_i, a_i) \leftarrow \frac{t}{t+1} R_i^t(n_i, a_i) + \frac{1}{t+1}[\sum_{s,n_{-i}} P(s, h^{\pi_{-i}^t}(n_{-i})|h^{\pi_i^t}(n_i)) \cdot [v(\pi_i^t|_{\sigma_i(n_i)\to a_i}, \pi_{-i}^t, n, s) - v(\pi^t, n, s)]]$
8: $\forall a_i \in A_i$, update $\pi_i^{t+1}(n_i, a_i)$ to the following:
$$\begin{cases} \frac{[R_i^{t+1}(n_i,a_i)]_+}{\sum_{a_i}[R_i^{t+1}(n_i,a_i)]_+} & \text{if denominator} > 0 \\ \pi_i^t(n_i, a_i) & \text{otherwise} \end{cases}$$
9: **end for**
10: **end for**
11: $t \leftarrow t + 1$
12: **until** termination_condition

can define *node regret* as

$$R_i^\tau(n_i) = \frac{1}{\tau} \max_{a_i} \sum_{t=1}^{t=\tau} \sum_{s,n_{-i}} P(s, h^{\pi_{-i}^t}(n_{-i})|h^{\pi_i^t}(n_i)) \cdot [v(\pi_i^t|_{\sigma_i(n_i)\to a_i}, \pi_{-i}^t, n, s) - v(\pi^t, n, s)] \quad (4)$$

This gives the regret for not executing action a_i at node n_i while keeping the rest of the (possibly stochastic) subpolicy rooted at n_i unchanged.

As with CFR, the decomposition of regret along the nodes of a policy tree is also useful according to the following theorem (proof in appendix):

Theorem 1. *For each agent, the overall average regret is upper bounded by the sum of the positive node regrets, i.e.,*

$$R_i^\tau \leq \sum_{n_i} [R_i^\tau(n_i)]_+, \ \forall i$$

Therefore, as the individual node regrets approach the nonpositive orthant by regret matching (i.e., individually become ≤ 0, by Blackwell's approachability [4]), the overall average regret must also be minimized, i.e., become nonpositive. The key benefit of Theorem 1 is that the individual node regrets can be controlled independently as defined in equation 4. The algorithm, called REMIT (REgret MInimization on Trees), is shown as Algorithm 1. It initializes all agents' stochastic policy trees with the uniform distribution at each node. Then it traverses each agent's tree in a breadth first manner and computes the node regrets and the new node distribution based on regret matching. The complexity of the loop in lines 6–9 is $O(|A||S|^2|\Omega|^T)$. This is also effectively the complexity of the **for** loop in lines 5–10 over the set of agents, since this loop can be parallelized because i's updates only depend on the previous joint policies. This loop is repeated until a termination condition is met. In this paper we study the following strong convergence criterion:

Termination Condition. *Set the termination condition in Algorithm 1 line 12, to*

$$R_i^t(n_i, a_i) = R_i^{t-1}(n_i, a_i) \leq 0, \ \forall i, n_i, a_i.$$

This gives a strong convergence criterion that $\exists t'$ such that $\forall t \geq t'$, $\pi^t = \pi^{t'}$.

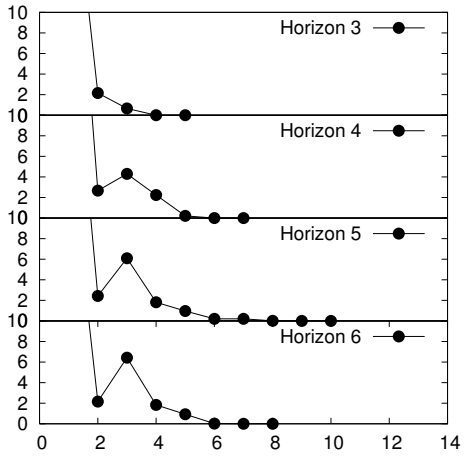

Figure 2: Relative errors in policy values from known optimals in Dec-Tiger.

This convergence criterion is stronger than the almost sure convergence of [10]. However, neither of these criteria can be theoretically guaranteed for REMIT. This is because Marden et al.'s results apply to the strategic form, and rely on the policy trajectory following a better response path in weakly acyclic games [10]. In contrast, we deliberately avoid working in the strategic form for efficiency, as a result of which there is no guarantee that the sequence of policies produced by tree-regret minimization treads the better response path. In fact, our empirical results show that the policies do not have strictly improving payoffs, i.e., they are indeed not following the better response path. Note however, that even though the means differ, the end is common. That is, by minimizing the component regrets, $R_i^\tau(n_i)$, we indirectly minimize the overall regret by Theorem 1. Therefore it is not surprising that we can, in fact, guarantee that the policies π^t approach a Nash equilibrium. We show this with the help of the above termination condition as follows.

While the satisfaction of the above termination condition is not guaranteed, it can be shown that if it is indeed achieved, then the terminal policies do indeed form a Nash equilibrium (proof in appendix).

Theorem 2. *If the* **Termination Condition** *is satisfied at time τ, then π^τ is a Nash equilibrium.*

Note that REMIT performs regret matching at each node n_i, therefore by [6] we know that

$$R_i^\tau(n_i) \leq \frac{Z\sqrt{|A_i|}}{\sqrt{\tau}}$$

where $Z = \max_{\pi,\pi'}(u(\pi) - u(\pi'))$. In other words, the **Termination Condition** is indeed satisfied in the limit. Hence we have the following theorem:

Theorem 3. *Even if we set termination_condition to 'false', REMIT converges to a Nash equilibrium in the limit.*

Our approach – which is singly exponential in T instead of doubly exponential – does not contradict the NEXP-hardness of Dec-POMDPs. Regret minimization converges to *some* Nash equilibrium joint policy in the Dec-POMDP, not necessarily the Pareto dominant one. On the other hand

the complexity of Dec-POMDPs reflect the cost of finding the Pareto optimal Nash equilibrium, which is clearly harder than finding just any Nash equilibrium. However, as our experiments show in the next section, REMIT does find (near) optimal equilibrium policies in a range of benchmark Dec-POMDP problems.

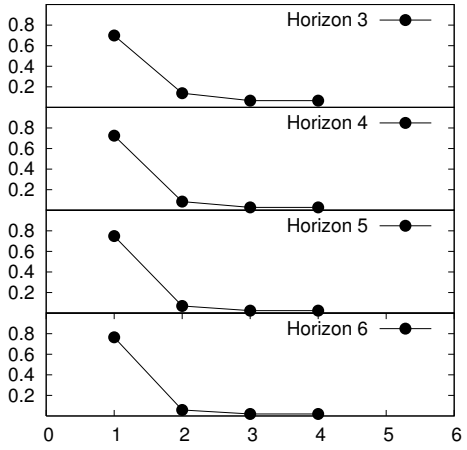

Figure 3: Relative errors in policy values from known optimals in Recycling-Robots.

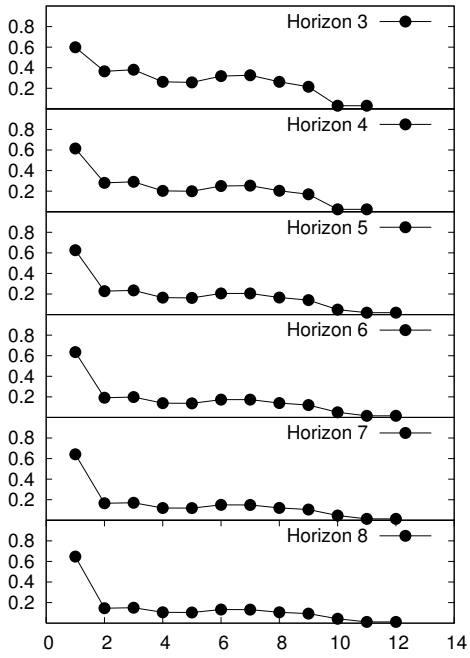

Figure 4: Relative errors in policy values from known optimals in BroadCast-Channel.

EXPERIMENTAL RESULTS

For experiments, we make an enhancement to line 7 in Algorithm 1, where we replace the weights $t/t+1$ and $1/t+1$ by $(1-\alpha)$ and α respectively, setting $\alpha = 0.7$. This updates the

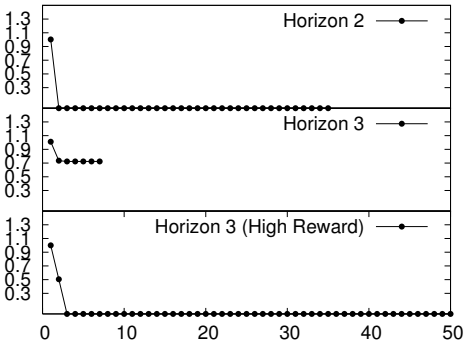

Figure 5: Relative errors in policy values from known optimals in Cooperative-Box-Pushing.

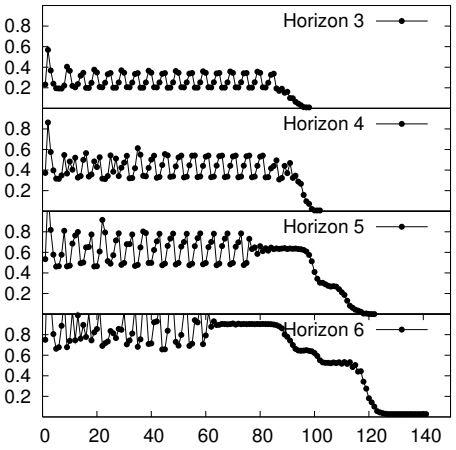

Figure 6: Relative errors in policy values from known optimals in Fire-Fighting.

regrets with *fading memory*. As used in [10] [1], fading memory is particularly useful for coordination problems when we seek convergence of policies, instead of empirical play. We also use the **Termination Condition** as given before in our experiments. Although there is no guarantee that such a strong convergence condition will be met, we found it to be satisfied in every benchmark problem and horizon that we attempted.

We have experimented in 5 benchmark Dec-POMDP domains – Dec-Tiger, Recycling-Robots, BroadCast-Channel, Cooperative-Box-Pushing, and Fire-Fighting (see [16] for details of these problems) – and found REMIT to converge to near-optimal equilibria in all except Cooperative-Box-Pushing at $T = 3$. Even in this case, it turns out that the reason that REMIT finds a highly suboptimal equilibrium is that the optimal equilibrium is not risk-dominant [5]. As observed in [5], "One might expect that in a learning setting it would be unlikely for play to converge to a very risky equilibrium, even if the equilibrium is Pareto efficient." To remedy this, we simply raised the payoff of both agents pushing the large box cooperatively from 100 to 10000, which turns out to be sufficient

[1]They also use inertia, which causes slower changes in policy. While this has a higher chance of reaching the optimal equilibrium, it also slows down convergence and has not been tried in the current experiments.

to make the optimal equilibrium risk dominant. REMIT converges to the optimal equilibrium in this case as well.

Figures 2– 6 show the results of REMIT on the above 5 domains. The x-axis in each plot is the number of iterations of the **repeat** loop in Algorithm 1. The y-axis shows the relative error in policy value at the end of each iteration, compared to the known optimal for each domain and horizon [16]. The right-most plot point in each (sub)figure corresponds to the iteration number when **Termination Condition** became true. We see that in some cases, especially in Cooperative-Box-Pushing, the policy values have converged long before this condition is met. If fading memory was not used, then the weight of the older regrets in the accumulated regrets would have been larger, meaning the initial regrets would have left a larger imprint in the accumulated regrets, exacerbating this effect. It is because the older regrets are weighed down sharply that the accumulated regrets can approach nonpositive values (and hence the termination condition) faster. We also see that REMIT attains optimal equilibria in Dec-Tiger (all horizons) and Cooperative-Box-Pushing (horizon 3 in the modified version only, as described above). In Recycling-Robots, BroadCast-Channel, and Fire-Fighting, it attains nearly optimal equilibria. Cooperative-Box-Pushing (horizon 3 original version) is the only problem (that we tried) where it converges to a highly suboptimal equilibrium. In Figure 6 we see an initial oscillatory pattern, which usually indicates the existence of counteracting attractors in the policy space, from which the REMIT trajectory eventually frees itself to achieve monotonically improving policy values. These attractors appear to be of progressively worse value for increasing horizons.

The main bottleneck for REMIT is the exponentially increasing size of policy trees with T. Currently, it is unable to exploit ideas such as history clustering [14] to compactly represent policies, and hence cannot solve problems for large T. However, it does not suffer from the memory and other time bottlenecks that many exact solvers face, and as a result can solve some problems beyond the known maximal horizons quite easily. E.g., in Dec-Tiger, REMIT can solve horizons 7 and 8 in matter of seconds, producing values 9.99357 and 12.2173. Since Dec-Tiger has only been solved up to horizon 6, we do not know if these are optimal, but these are very likely to be so (within rounding errors). This is because the policies produced by REMIT are basically concatenations of horizon 3 policy with horizon 4(5) policy which are very likely to be optimal for $T = 7(8)$, given the periodic structure of the problem. For reference, the optimal horizon 6 policy is the concatenation of two horizon 3 policies, leading to the value $10.3816 = 2 \times 5.1908$. We do not show plots for T beyond the known max in any problem, due to uncertainty about the optimal policy values.

The consistent satisfaction of the strong convergence criterion in a range of benchmark problems with different characteristics is surprising. It raises the intriguing possibility that it can be rigorously proven for general regret matching techniques (including REMIT) in identical payoff games. This would be a new result in game theory as well. We leave this for future investigation.

CONCLUSIONS

We have presented a compact way to apply counterfactual regret minimization to Dec-POMDPs, with a per-node decomposition of regret as opposed to per-history decomposition. We have proven that our algorithm, REMIT, converges to a Nash equilibrium policy. We have also shown empirically that it actually yields (near) optimal Nash equilibria in a finite number of iterations in a range of benchmark problems.

An immediate future direction is to test a sampling version of REMIT, where the node regrets are *estimated* on the basis of unbiased samples of the subpolicy values. We believe this will yield a more efficient reinforcement learning algorithm than what currently exists.

ACKNOWLEDGMENT

We thank the reviewers for helful comments and suggestions. This work was supported in part by the U.S. Army under grant #W911NF-11-1-0124.

REFERENCES

[1] R. Aras and A. Dutech. An investigation into mathematical programming for finite horizon decentralized POMDPs. *JAIR*, 37:329–396, 2010.

[2] B. Banerjee, J. Lyle, L. Kraemer, and R. Yellamraju. Sample bounded distributed reinforcement learning for decentralized POMDPs. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12)*, pages 1256–1262, Toronto, Canada, July 2012.

[3] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research*, 27:819–840, 2002.

[4] D. Blackwell. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6:1–8, 1956.

[5] D. Fudenberg and K. Levine. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1998.

[6] G. J. Gordon. No-regret algorithms for online convex programs. In *In Neural Information Processing Systems 19*, 2007.

[7] E. A. Hansen, D. S. Bernstein, and S. Zilberstein. Dynamic programming for partially observable stochastic games. In *Proceedings of the 19th National Conference on Artifical Intelligence*, pages 709–715, San Jose, CA, 2004.

[8] S. Hart and A. Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68:1127–1150, 2000.

[9] A. Kumar and S. Zilberstein. Dynamic programming approximations for partially observable stochastic games. In *Proceedings of The 22nd International FLAIRS Conference (FLAIRS-09)*, pages 547–552, 2009.

[10] J. Marden, G. Arslan, and J. Shamma. Regret based dynamics: Convergence in weakly acyclic games. In *Proceedings of 6th Intl. Joint Conf. on Autonomous Agents and Multi-agent Systems*, pages 194–201, 2007.

[11] R. Nair, M. Tambe, M. Yokoo, D. Pynadath, and S. Marsella. Taming decentralized POMDPs: Towards efficient policy computation for multiagent settings. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03)*, pages 705–711, Acapulco, Mexico, 2003.

[12] J. F. Nash. Non-cooperative games. *Annals of Mathematics*, 54:286 – 295, 1951.

[13] F. A. Oliehoek, J. F. Kooij, and N. Vlassis. The cross-entropy method for policy search in decentralized POMDPs. *Informatica*, 32:341–357, 2008.

[14] F. A. Oliehoek, S. Whiteson, and M. T. J. Spaan. Lossless clustering of histories in decentralized POMDPs. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-09)*, pages 577–584, Budapest, Hungary, 2009.

[15] S. Seuken and S. Zilberstein. Memory-bounded dynamic programming for DEC-POMDPs. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 2009–2015, Hyderabad, India, 2007.

[16] M. Spaan. Dec-POMDP problem domains and format. `http://users.isr.ist.utl.pt/~mtjspaan/decpomdp/`.

[17] M. T. J. Spaan, F. A. Oliehoek, and C. Amato. Scaling up optimal heuristic search in Dec-POMDPs via incremental expansion. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI-11)*, pages 2027–2032, Barcelona, Spain, 2011.

[18] D. Szer, F. Charpillet, and S. Zilberstein. MAA*: A heuristic search algorithm for solving decentralized POMDPs. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pages 576–590, 2005.

[19] H. P. Young. *Strategic Learning and its Limits*. Oxford University Press, 2004.

[20] M. Zinkevich, M. Johanson, M. Bowling, and C. Piccione. Regret minimization in games with incomplete information. In *NIPS*, 2007.

APPENDIX

PROOF OF THEOREM 1

Suppose the state transitions from s to s' when the agents execute joint action a at joint history h, producing joint observation ω. Let n_{h_i} represent the node at the end of history h_i, and $h' = (h.a.\omega)$, i.e., the next joint history. For brevity, we write the transition and observation probabilities jointly as $P(s', \omega|s, a)$. Although our proof of Theorem 1 is somewhat similar to that of Theorem 3 in [20], a key lemma that distinguishes our proof concerns the propagation of i's conditional belief $P(s, h_{-i}|h_i)$, given next. There are also some original steps embedded in the proof of Lemma 5 which is the counterpart of Lemma 5 in [20].

Lemma 4. *The conditional belief propagation is given as*

$$\begin{aligned} P(s', h'_{-i}|h'_i) &= \sum_s P(s', \omega|s, a)\pi_{-i}(n_{h_{-i}}, a_{-i}) \cdot \\ &\quad P(s, h_{-i}|h_i)/P(\omega_i|a_i, h_i) \end{aligned}$$

Proof: We observe

$$\begin{aligned} P(s', h'_{-i}|h'_i) &= \sum_s P(s, s', h, a, \omega)/P(h_i, a_i, \omega_i) \\ &= \sum_s P(s', \omega|s, a)P(s, h, a)/P(h_i, a_i, \omega_i) \\ &= \sum_s P(s', \omega|s, a)P(a_{-i}|h_{-i}) \cdot \\ &\quad P(s, h, a_i)/P(h_i, a_i, \omega_i) \\ &= \sum_s P(s', \omega|s, a)\pi_{-i}(n_{h_{-i}}, a_{-i}) \cdot \\ &\quad P(s, h_{-i}|h_i)P(h_i, a_i)/P(h_i, a_i, \omega_i) \\ &= \sum_s P(s', \omega|s, a)\pi_{-i}(n_{h_{-i}}, a_{-i}) \cdot \\ &\quad P(s, h_{-i}|h_i)/P(\omega_i|a_i, h_i) \end{aligned}$$

Lemma 5. *Suppose the successor joint node after making the joint observation ω at joint node n be n'. Then,*

$$R^\tau_{i,full}(n_i) \leq R^\tau_i(n_i) + \sum_{n'_i} [R^\tau_{i,full}(n'_i)]_+, \ \forall i$$

Proof: We break $R^\tau_{i,\text{full}}(n_i)$ into two components: the node regret at n_i and the full regret of the subtrees under n_i, starting at its definition in equation 3.

$$\begin{aligned} R^\tau_{i,\text{full}}(n_i) &= \frac{1}{\tau} \max_{\pi'_i} \sum_{t=1}^{t=\tau} \sum_{s, n_{-i}} P(s, h^{\pi^t_{-i}}(n_{-i})|h^{\pi^t_i}(n_i)) \cdot \\ &\quad [v(\pi^t_i|_{D(n_i)\to\pi'_i}, \pi^t_{-i}, n, s) - v(\pi^t, n, s)] \\ &= \frac{1}{\tau} \max_{a_i, \pi'_i} \sum_{t=1}^{t=\tau} \sum_{s, n_{-i}} P(s, h^{\pi^t_{-i}}(n_{-i})|h^{\pi^t_i}(n_i)) \cdot \\ &\quad \Big[v(\pi^t_i|_{\sigma_i(n_i)\to a_i}, \pi^t_{-i}, n, s) - v(\pi^t, n, s) + \\ &\quad \sum_{n', s', a_{-i}} P(n', s', a_{-i}|n, s, a_i) \cdot \\ &\quad \{v(\pi^t_i|_{D(n'_i)\to\pi'_i}, \pi^t_{-i}, n', s') - v(\pi^t, n', s')\}\Big] \\ &\leq R^\tau_i(n_i) + \max_{a_i, \pi'_i} \frac{1}{\tau} \sum_{t, s, n_{-i}} P(s, h^{\pi^t_{-i}}(n_{-i})|h^{\pi^t_i}(n_i)) \cdot \\ &\quad \sum_{n', s', a_{-i}} P(n', s', a_{-i}|n, s, a_i) \cdot \\ &\quad \Big[v(\pi^t_i|_{D(n'_i)\to\pi'_i}, \pi^t_{-i}, n', s') - v(\pi^t, n', s')\Big] \\ &= R^\tau_i(n_i) + \max_{a_i, \pi'_i} \frac{1}{\tau} \sum_{t, s, n_{-i}} P(s, h^{\pi^t_{-i}}(n_{-i})|h^{\pi^t_i}(n_i)) \cdot \\ &\quad \sum_{\omega, s', a_{-i}} P(s', a_{-i}, \omega|s, h^{\pi^t}(n), a_i) \cdot \\ &\quad \Big[v(\pi^t_i|_{D(n'_i)\to\pi'_i}, \pi^t_{-i}, n', s') - v(\pi^t, n', s')\Big] \\ &= R^\tau_i(n_i) + \max_{a_i, \pi'_i} \frac{1}{\tau} \sum_{t, \omega_i} \Big\{ \sum_{s', n_{-i}, a_{-i}, \omega_{-i}} \sum_s \\ &\quad P(s, h^{\pi^t_{-i}}(n_{-i})|h^{\pi^t_i}(n_i)) P(s', a_{-i}, \omega|s, h^{\pi^t}(n), a_i) \cdot \\ &\quad [v(\pi^t_i|_{D(n'_i)\to\pi'_i}, \pi^t_{-i}, n', s') - v(\pi^t, n', s')]\Big\} \end{aligned}$$

Now,

$$\begin{aligned} P(s', a_{-i}, \omega | s, h^{\pi^t}(n), a_i) &= P(s', \omega | s, a) P(a_{-i} | s, h^{\pi^t}(n), a_i) \\ &= P(s', \omega | s, a) P(a_{-i} | h^{\pi^t_{-i}}(n_{-i})) \\ &= P(s', \omega | s, a) \pi^t_{-i}(n_{h_{-i}}, a_{-i}) \end{aligned}$$

Therefore, we can replace

$$\sum_s P(s, h^{\pi^t_{-i}}(n_{-i}) | h^{\pi^t_i}(n_i)) P(s', a_{-i}, \omega | s, h^{\pi^t}(n), a_i)$$

in the last step of $R^\tau_{i,\text{full}}(n_i)$ by

$$\sum_s P(s, h^{\pi^t_{-i}}(n_{-i}) | h^{\pi^t_i}(n_i)) P(s', \omega | s, a) \pi^t_{-i}(n_{h_{-i}}, a_{-i}).$$

Then using Lemma 4, we can replace this by

$$P(s', h^{\pi^t_{-i}}(n'_{-i}) | h^{\pi^t_i}(n'_i)) P(\omega_i | h^{\pi^t_i}(n_i), a_i)$$

Then continuing from the last step of $R^\tau_{i,\text{full}}(n_i)$,

$$\begin{aligned} R^\tau_{i,\text{full}}(n_i) &\leq R^\tau_i(n_i) + \max_{a_i, \pi'_i} \frac{1}{\tau} \sum_{t, \omega_i} \Big\{ \sum_{s', n_{-i}, a_{-i}, \omega_{-i}} \\ &\quad P(s', h^{\pi^t_{-i}}(n'_{-i}) | h^{\pi^t_i}(n'_i)) P(\omega_i | h^{\pi^t_i}(n_i), a_i) \cdot \\ &\quad [v(\pi^t_i|_{D(n'_i) \to \pi'_i}, \pi^t_{-i}, n', s') - v(\pi^t, n', s')] \Big\} \\ &= R^\tau_i(n_i) + \max_{a_i, \pi'_i} \frac{1}{\tau} \sum_{t, \omega_i} \Big\{ P(\omega_i | h^{\pi^t_i}(n_i), a_i) \cdot \\ &\quad \sum_{s', n'_{-i}} P(s', h^{\pi^t_{-i}}(n'_{-i}) | h^{\pi^t_i}(n'_i)) \cdot \\ &\quad [v(\pi^t_i|_{D(n'_i) \to \pi'_i}, \pi^t_{-i}, n', s') - v(\pi^t, n', s')] \Big\} \\ &= R^\tau_i(n_i) + \max_{a_i} \sum_{\omega_i} \Big\{ P(\omega_i | h^{\pi^t_i}(n_i), a_i) R^\tau_{i,\text{full}}(n'_i) \Big\} \\ &\leq R^\tau_i(n_i) + \sum_{\omega_i} R^\tau_{i,\text{full}}(n'_i) \\ &= R^\tau_i(n_i) + \sum_{n'_i} R^\tau_{i,\text{full}}(n'_i) \\ &\leq R^\tau_i(n_i) + \sum_{n'_i} [R^\tau_{i,\text{full}}(n'_i)]_+ \end{aligned}$$

This concludes the proof of lemma 5. □

The rest of the proof of Theorem 1 is by inductive application of Lemma 5 from the root to the leaves of i's policy, as in [20].

PROOF OF THEOREM 2

We make the standard assumption that all payoffs are bounded, so that there exists a well defined minimum value of joint policies:

$$m \triangleq \min_\pi u(\pi)$$

Now if the convergence criterion is satisfied, π^τ remains unchanged for all $t > \tau$ if REMIT keeps running beyond τ. Suppose, π^τ is not a Nash equilibrium; then there exists a policy π^*_i for some agent i such that the joint payoff strictly improves, i.e., improves by at least $\epsilon > 0$. That is

$$u(\pi^*_i, \pi^\tau_{-i}) \geq u(\pi^\tau) + \epsilon.$$

Let us fix some τ' beyond τ such that

$$\tau' > \tau(1 + |m|/\epsilon)$$

Now the overall regret at the policy root for agent i over τ' steps is

$$\begin{aligned} R^{\tau'}_i &= \frac{1}{\tau'} \max_{\pi'_i} \sum_{t=1}^{\tau'} (u(\pi'_i, \pi^t_{-i}) - u(\pi^t)) \\ &\geq \frac{1}{\tau'} \sum_{t=1}^{\tau'} (u(\pi^*_i, \pi^t_{-i}) - u(\pi^t)) \\ &= \frac{1}{\tau'} [\sum_{t=1}^{\tau} (u(\pi^*_i, \pi^t_{-i}) - u(\pi^t)) + \\ &\quad \sum_{t=\tau+1}^{\tau'} (u(\pi^*_i, \pi^t_{-i}) - u(\pi^t))] \\ &= \frac{1}{\tau'} [\sum_{t=1}^{\tau} (u(\pi^*_i, \pi^t_{-i}) - u(\pi^t)) + \\ &\quad (\tau' - \tau)(u(\pi^*_i, \pi^\tau_{-i}) - u(\pi^\tau))] \\ &\geq \frac{1}{\tau'} [\sum_{t=1}^{\tau} (u(\pi^*_i, \pi^t_{-i}) - u(\pi^t)) + (\tau' - \tau)\epsilon] \\ &\geq \frac{1}{\tau'} [-|m|\tau + (\tau' - \tau)\epsilon] \\ &> 0, \text{ by the choice of } \tau' \end{aligned}$$

However, since all node regrets of all agents are ≤ 0 at τ', in particular for agent i, by Theorem 1 we know that $R^{\tau'}_i \leq 0$ – a contradiction. Therefore π^τ must be a Nash equilibrium.