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

Fast Distributed Strategic Learning for Global Optima in Queueing Access Games

Hamidou Tembine *

* *KAUST Strategic Research Initiative Center for Uncertainty Quantification in Computational Science and Engineering, CEMSE, KAUST*

Abstract: In this paper we examine combined fully distributed payoff and strategy learning (CODIPAS) in a queue-aware access game over a graph. The classical strategic learning analysis relies on vanishing or small learning rate and uses stochastic approximation tool to derive steady states and invariant sets of the underlying learning process. Here, the stochastic approximation framework does not apply due to non-vanishing learning rate. We propose a direct proof of convergence of the process. Interestingly, the convergence time to one of the global optima is almost surely finite and we explicitly characterize the convergence time. We show that pursuit-based CODIPAS learning is much faster than the classical learning algorithms in games. We extend the methodology to coalitional learning and proves a very fast formation of coalitions for queue-aware access games where the action space is dynamically changing depending on the location of the user over a graph.

Keywords: Coalitional learning, queue, access control

1. INTRODUCTION

Strategic learning is a framework for learning solutions of interactive decision-making problems. A strategic learning algorithm is said to be partially (or semi-) distributed if each player knows its own-action, its own-payoff function and the actions of the other players (or aggregate signal) at the previous steps. Semi-distributed learning algorithms are model-based in the sense that it requires the mathematical structure (model) of the payoff function. A strategic learning procedure is called fully distributed (or model-free or uncoupled) if each player knows only a numerical measurement of the realized payoff and its own-action. In 1952 Bellman described the optimality equation for model-based dynamic optimization Bellman [1952]. In 1953, Shapley proposed a model-based recursive equation for equilibrium payoffs in the context of stochastic games Shapley [1953]. One of the non-model learning algorithms have been proposed by Bush and Mosteller [1953]. The main idea behind these works is to combine both strategy and payoff learning schemes, which we have referred to as *combined learning* (see Tembine [2012]). In the model-based learning, it is assumed that the game is common knowledge. In particular, the transitions probabilities between the states are perfectly known. Then, each player updates her value function and uses a policy that maximizes the Hamiltonian, i.e., a balance between the payoff today and the continuation payoff. If the state is perfectly observed by the players, one can use state-based dynamic programming principle and a solution to such system provides an equilibrium strategy and equilibrium payoff. Moreover, the variance between the model-based learning and the equilibrium value is significantly reduced. On the other hand, in the non-model (or model-free) learning, the players do not have knowledge of the transition probabilities and do not have the mathematical structure of their payoff functions. Each player updates her tables after observing the signal (measured/observed payoff). Her policy should be designed by the strategy-learning pattern ($\max\max Q$, soft-max, imitation, best response estimates, etc). In the context of games, the convergence of such method remains an open issue. However, there are some special cases where the convergence can be proved as for a single player in a dynamic environment. This is what is done in a recent work by Martin and Tilak [2012]. Using stochastic approximation techniques, it have been shown in Kushner and Yin [2003] that for single player and vanishing learning rates, the non-model technique also converges to expected equilibrium value if all the states and actions have been sufficiently explored and exploited. Thus, the two approaches give similar expected payoffs in the long-run. An interesting remark is that if the performance criterion is limited to the expected equilibrium payoffs then, it is difficult to compare the model-based learning and the model-free learning. One has attempted to look at the convergence time and the speed of convergence of both classes of algorithms. However, the convergence time analysis remains a challenging task. Another alternative is to look at not only the mean payoff but also the variance of the payoff, leading to risk-sensitive approach.

Less is more: It is widely observed that model-free learning scheme is relatively simple and requires only few arithmetic operations. However, it was conjectured that it needs to run for lot of iterations in order to be close to the expected payoff. Thus, a natural question is to know if one can get a faster convergence time for such a learning scheme while preserving accuracy. In the case of single player in an i.i.d environment, the pursuit combined fully distributed payoff and strategy learning (pursuit

CODIPAS learning) algorithms are known to have very fast convergence time in practice. However, for more than one player the question is open as mentioned in Tembine [2012].

In this paper we construct a pursuit algorithm for queue-aware access games using coalitional combined fully distributed payoff and strategy learning (coalitional CODIPAS). The idea of coalitional CODIPAS is to estimate simultaneously the expected coalitional equilibrium payoffs for each action of the temporary coalition and the associated optimal intra and inter-coalition strategies. Based on it, we construct a pursuit strategy which consists to take the index of the maximum estimates with a certain probability and act with a distribution with full support in case of multiple maximizers. We show that the non-model algorithm can be accelerated by exploiting the experiences and by designing a learning rate appropriately. We specially focus on queue-aware access games, a class of anticoordination games for multiple items, i.e., games with negative externalities where choosing the same action creates a cost rather than a benefit. Anticoordination games are common problems in economics, engineering and transportation science: a number of players have to choose independently whether or not to undertake some activity, such as enter a market, go to a bar, drive on a road, choose a frequency/technology, choose a network or surf the web, the payoff from which is decreasing in the number of participants of the chosen decision. Those choosing not to undertake the activity can be thought of as staying at home, staying out of the market, choosing another technology or simply not participating. Such games typically admit a large number of Nash equilibria. Pure equilibria involve considerable coordination on asymmetric outcomes (decisions). Given this multiplicity of equilibria, an obvious question is: which type of equilibrium are agents likely to coordinate upon?

Review of access control interaction models: The authors in Liu and Liu [2013] considers a cognitive multiuser dynamic channel access where users can decide to stay or switch between different channels. The queue aspect is neglected in their analysis. In Nguyen and Baccelli [2012], a spatial modelling of the carrier-sensing-multiple-access (CSMA) network is proposed via poisson point processes. Their model captures some aspects of stochastic locations and mobility. However the independence and stationary assumption of user mobility seems questionable in the context of carrier sensing multiple access. Indeed, the successfulness of a transmission depends mainly on the powers, queues, locations, on interference created by the active users. Thus, their main assumptions fail when the correlation between the queues of users is taken into consideration. The work in MacKenzie and Wicker [2003] studies stability region for a slotted Aloha system with multipacket reception and selfish users for the case of perfect information. However the assumption of perfect information is not valid in many cases of interest. In particular in our setup a user may not be able observe the number active users present in the system. In this paper we deal with imperfect information and non-model learning algorithm.

Review of learning in access control interactions: The authors in Tekin and Liu [2011] studied online learning in opportunistic spectrum access based on a restless bandit approach. However, their reward model is not interactive and limited to single user case.

The work in Sarikaya et al. [2012] proposed a dynamic pricing and queue stability in wireless random access games. The authors proposed also a learning algorithm. However, their model has restriction of the probability of transmission to be bounded away from zero at any time. It means that each user transmits with non-zero probability all the time (even if the queue is empty!), which is clearly not a realistic model.

Why queue-aware access control model is more suitable in practice? Most prior works (see for example, Liu and Liu [2013]; Nguyen and Baccelli [2012]) assumed saturated queues, i.e. they assumed that each user has always a data to send. In practice however, it is observed that, some user may not a packet to send for a certain period of time. In order to capture such a scenario we shall introduce a queue-aware decision-making approach. By doing so, if the queue is empty, a player will not transmit and will save energy. It is important to notice that the different queue size evolutions are correlated since they depend on the success probability and the strategy adopted in each coalition of players.

The challenges: The action set may be changing in time due to changes in the queue or due to the position of the player over the connectivity graph. In some locations, there more available access points than some others. Most of learning procedures in the literature are limited to the case where the action set is static and fixed forever. The dynamic aspect of the choices and the constraints on queue size bring novel challenges to the learning procedure.

Thus, the coalitional CODIPAS that we have developed earlier Tembine [2012] needs to be adapted to graph-restricted coalition formation that changes in time. The challenge here is to get the possibility to incorporate this important aspect and evaluate the cost of making the coalition formation process while the group members are actively changing.

Another important challenge here is to construct a faster and more accurate algorithm for both coalition formation and distributed strategic learning for global optima.

Novel results from CODIPAS learning: To the best to our knowledge there has been no study of distributed strategic learning in the context of queue-aware access channel, either in terms of efficiency and accuracy or fast convergence to global optimum based on queue size. Here lies the contribution of the present paper. Our results can be summarized as follows. (i) Fast convergence to evolutionarily stable coalitional structure in queue-aware access games (ii)Characterization of the convergence time of pursuit CODIPAS in queue-aware access game in strategic form as well as in evolutionary coalitional form. See Propositions 1 and 2. (iii) Less is More: For queue-aware access game, we show that CODIPAS is much faster (and more accurate) than the classical learning in games such as fictitious play Brown [1951] and Bush-Mosteller-based CODIPAS Bush and Mosteller [1953]. This is counterintuitive because the pursuit CODIPAS uses less information than fictitious play (which requires perfect observations

of previous actions of the other players) but still fictitious play is slower than pursuit CODIPAS. Note that the fictitious play exhibits a cycling behavior between the states if it starts with initial strategy distribution in the form $(\epsilon, 1-\epsilon), \epsilon < 1/2$. Thus, the sequence of action profile does not converge under fictitious play and the cumulated payoff is worse than the worst Nash equilibrium payoffs. Under pursuit CODIPAS algorithm we show that the strategy profile and the long-run payoff converge to the best equilibrium which is also a global optimum, and this holds even for random and heterogeneous learning rates.

Structure: The remainder of the paper is organized as follows. In Section 2, we introduce the model and present the pursuit CODIPAS procedure. In Section 3 we present the main results. Section 4 concludes the paper.

We summarize some of the notations used in the paper in Table 1.

Table 1. Summary of Notations

Symbol	Meaning
$\mathcal{N}$	set of users (players)
n	cardinality of $\mathcal{N}$
$\mathcal{A}$	set of atomic actions (access points)
$a_j(t)$	action or set of simultaneously actions of j at time t
$q_j(t)$	queue length of user j at time t
DQ_j	departure process from queue of user j
AQ_j	arrival process to queue of user j
$r_j(a)$	payoff function
$r_j(t)$	measured payoff of user j at time t
$\hat{r}_j(t))$	estimated payoff (vector) of user j at time t
$x_j(t)$	mixed strategy of user j at time t
C	coalition (subset of $\mathcal{N}$)
$V(.,C)$	value of coalition C
ϕ_j	dynamic Shapley value
$\mathbb{E}$	expectation operator

2. MODEL

We consider n players (users), $\mathcal{N} = \{1, 2, \ldots, n\}: n \geq 2$. Every player j has its own queue. Let q_j be the queue length of player j. If $q_j = 0$ the queue of player j is empty, i.e., player j has no packet to send. Each player has m possible atomic-actions $\mathcal{A} = \{1, 2, \ldots, m\}$ if its queue is non-empty, and has only one action $\{0\}$ if its queue is empty. The action "0" is interpreted as "Not transmit" or "Wait". The actions, from 1 to m, correspond to different access points that can be chosen by the players. If its queue allows, a player can select several access points simultaneously. In that case, an action is a choice of a subset of $\mathcal{A}$.

In the one-shot queue-aware access game, each player j who has a packet can choose an action $a_j \in \mathcal{A}$ and receives a payoff, $r_j(a_1, \ldots, a_n)$, that depends on the actions picked by all players. Hence a pure strategy of a player in the one-shot queue-aware access game is a mapping from its own-queue length to a subset of $\mathcal{A}$ (union the singleton $\{0\}$).

Anticoordination games with saturated queues: We say that a player has a saturated queue if its queue is never empty, i.e., the player has always at least a packet to send. For $n = m = 2$ we represent the matrix game problem in a table. Player 1 chooses a row, player 2 chooses a column of the table. If player 1 chooses $a_1 \in \mathcal{A}$ and player 1 chooses $a_2 \in \mathcal{A}$ then player 1 receives $r_1(a_1, a_2)$, and player 2 will get $r_2(a_1, a_2)$. This gives a matrix game between the two players. The matrix game above is an anticoordination

Player 1 vs Player 2	a_1	a_2
a_1	$(r_1(1,1), r_2(1,1))$	$(r_1(1,2), r_2(1,2))$
a_2	$(r_1(2,1), r_2(2,1))$	$(r_1(2,2), r_2(2,2))$

Table 2. Matrix game

game if the table is strategically equivalent to the following one, where $\alpha_i \geq 0$ A Nash equilibrium is a strategy profile

Player 1 vs Player 2	action 1	action 2
action 1	$(0,0)$	$(\alpha_1, \alpha_2)^*$
action 2	$(\alpha_2, \alpha_1)^*$	$(0,0)$

such that no player can improve its (expected) payoff by unilateral deviation. In the randomized (mixed) strategies, the anticcordination game above has three Nash equilibria (2 pure equilibria and one mixed equilibrium - uniform). The fact that we have three equilibria is not surprising, this is part a more general result that says that, generically a finite game has an odd number of Nash equilibria in mixed strategies.

Remark 1. Most of the previous analysis (Liu and Liu [2013]; Nguyen and Baccelli [2012]) were conducted under saturated queues. As a consequence a player will be taking the suggested strategy even if its queue is empty (no data to send) which is not realistic. In the next subsection we propose and analyze a queue-length based game theoretic model.

Two players with non-saturated queues: Now we consider the case where the queue can be empty. In that case there are more possibilities and more strategic outcomes, represented in Fig. 1- 5. When there are two available access points operating at different frequencies f_1 and f_2, we distinguish six classes of states.

First class: $q_1 \geq 2, q_2 \geq 2$ In this first-class of states, each user has at least two packets to send and there are two available access points. A user has 4 four actions in this particular state: choose one of the two frequencies f_1, f_2, or to choose both frequencies simultaeously $\{f_1, f_2\}$, or to wait (no transmission). When frequencies are chosen simultaneously, the payoff is the sum-payoff (success condition) over the two channels. If a user has a successful transmission in both channels then its queue length is decreased by 2 for the corresponding slot and increased with the arrivals. the queue length is decreased by one if there exactly one successful transmission in one the two channels. If there is no suceesful transmission, the queue length will increase if there is an arrival and remains the same if not. Thus, from this first-class of states $(q_1 \geq 2, q_2 \geq 2)$, the stochastic game can move to the following classes of states: $(q_j = 1, q_{j'} \geq 2)$, $(q_j = 1, q_{j'} = 1)$, $(q_j = 0, q_{j'} \geq 2)$, or it can stay at the class of states $(q_j \geq 2, q_{j'} \geq 2)$.

A pure Nash equilibrium is an action profile such no user can improve its payoff by unilateral deviation. If the expectation of the random quantity α_i are non-zero, then the pure action profiles, (f_1, f_2), (f_2, f_1), $(Wait, both)$, $(both, Wait)$ are global optima and pure Nash equilibria of

the expected game in that particular state. These action profiles give the expected payoff $\mathbb{E}(\alpha_1 + \alpha_2) \leq 2$. We say that an action profile is Pareto optimal if one cannot improve of the payoff of one user cannot decreasing the payoff of the other. The (pure) Pareto optimal profiles are represented by star (*).

	f_1	f_2	$both$	$Wait$
f_1	$0,0$	$(\alpha_1,\alpha_2)^*$	$0,\alpha_2$	$\alpha_1,0$
f_2	$(\alpha_2,\alpha_1)^*$	$0,0$	$0,\alpha_1$	$\alpha_2,0$
$both$	$\alpha_2,0$	$\alpha_1,0$	$0,0$	$(sum,0)^*$
$Wait$	$0,\alpha_1$	$0,\alpha_2$	$(0,sum)^*$	$0,0$

$q_1 \geq 2, q_2 \geq 2$

Fig. 1. Each user has at least two packets to send. $\alpha_c = \mathbb{1}_{\{SNR_{j,c}\geq\beta|q_j\geq 2\}}$, $sum = \alpha_1 + \alpha_2$

	f_1	f_2	$Wait$
f_1	$0,0$	$(\alpha_1,\alpha_2)^*$	$\alpha_1,0$
f_2	$(\alpha_2,\alpha_1)^*$	$0,0$	$\alpha_2,0$
$both$	$\alpha_2,0$	$\alpha_1,0$	$(\alpha_1+\alpha_2,0)^*$
$Wait$	$0,\alpha_1$	$0,\alpha_2$	$0,0$

$q_1 \geq 2, q_2 = 1$

Fig. 2. One of the users has at least two packets to send and the other has exactly one packet.

Second class of states: $(q_j = 1, q_{j'} \geq 2)$ The fact that one of the users has only one packet will affect the system performance compared to the previous class of states. If the user who has at least two packets use both frequencies simultaneously and the other stay quiet then $(both, Wait)$ is an equilibrium. Interestingly these pure equilibria are also global optima.

Third class: $q_1 = 1 = q_2$ In this class, each player has exactly one packet to send. There are 3 equilibria (two of them are pure and one fully mixed). Under the two pure equilibrium strategies, the queue state will move $(0,0)$ if there is no arrival. In state $(0,0)$ the queues are all empty and therefore the action sets are reduced to the set $\{Wait\}$. As we can see in Fig. 3 the pure strategy $both$ (which consists to use both frequencies simultneously) is not feasible in this state. Similarly the action can be reduced to "0" or Wait if the energy level (battery-state) of the user does not allow.

	f_1	f_2	$Wait$
f_1	$0,0$	$(\alpha_1,\alpha_2)^*$	$\alpha_1,0$
f_2	$(\alpha_2,\alpha_1)^*$	$0,0$	$\alpha_2,0$
$Wait$	$0,\alpha_1$	$0,\alpha_2$	$0,0$

$q_1 = 1 = q_2$

Fig. 3. Each user has exactly one packet to send.

Fourth class: $q_j \geq 2, q_{j'} = 0$ In this class of states, one of the users has NO packet to send and the other has at least two packets. This is a dominant solvable game and is reduced to a single decision-maker problem. The equilibrium structure consists to use both channels by the user who has least two packets (the other user has no packet and hence will be waiting). The equilibrium payoff will have the form

	$Wait$
f_1	$\alpha_1,0$
f_2	$\alpha_2,0$
$both$	$(\alpha_1+\alpha_2,0)^*$
$Wait$	$0,0$

$q_1 \geq 2, q_2 = 0$

Fig. 4. One of the users has NO packet to send and the other has at least two packets.

of $(0, \alpha_1+\alpha_2)$ which is also a global optimum. The pursuit CODIPAS learning algorithm provided below converges to the global optimum in the corresponding state (Fig4).

Fifth class of states: $q_j = 1, q_{j'} = 0$ In this class, one of the users has NO packet to send and the other has exactly one packet. If $\alpha_1 \neq \alpha_2$, the game is dominant solvable game. Therefore the outcome is $(f_{c^*}, Wait)$ where c^* is the best channel among the two. The equilibrium payoff is in the form $(\max(\alpha_1,\alpha_2), 0)$. The pursuit CODIPAS learning algorithm provided below converges to the equilibrium which is also a global optimum in the corresponding state.

	$Wait$
f_1	$\alpha_1,0$
f_2	$\alpha_2,0$
$Wait$	$0,0$

$q_1 = 1, q_2 = 0$

Fig. 5. One of the users has NO packet to send and the other user has exactly one packet.

Sixth class: $q_j = q_{j'} = 0$ In this state, the queues are empty and both action sets are reduced to the singleton $\{Wait\}$ and the payoffs are 0. The queue state will move to the number of new arrivals.

As we can see from the above analysis, the equilibrium structures depend on the class of states. Therefore, there is no state-independent equilibrium. This means that one has to take into consideration the queue-length in our analysis. We consider an extension of this anticoordination game to n players and m atomic-actions.

Definition 1. We say that the general n-players m-atomic actions game is an anticoordination game if the payoff functions are defined by

$$r_j(a_1,\ldots,a_m) = \begin{cases} \alpha_{a_j} & \text{if } a_j \neq a_i,\ \forall i \neq j \\ 0 & \text{otherwise} \end{cases}$$

where α_{a_j} represents the indicator for success condition. When a_j is a subset we count the total number of success. The action outputs depend implicitly on the queue and battery (energy) level.

For channel c, the random variable α_c is the indicator condition $\mathbb{1}_{\{SNR_{j,c}\geq\beta|q_j>0\}}$ where

$$SNR_{j,c} = \frac{p_j|h_{jc}|^2}{N_0^2(\epsilon^2 + d_{j,c}^2)^{\frac{\xi}{2}}},$$

the allocated power

$$p_j = p_j(location, q_j, remaining\ energy\ level),$$

$N_0 > 0$ is a background noise, $d_{j,c}$ is the distance from the transmitter to the receiver (assimilated to as from user to

access point), $\xi \geq 2$ is the pathloss exponent and $\epsilon > 0$. In particular, it is important to notice that the optimal decision process and the payoff function depend on the queue size.

Evolution of queues: Let AQ_j be the arrival process of player $j's$ queue. $AQ_j(t)$ denotes the number of packets j receives at time t. Let $DQ_j(t)$ be the departure process, i.e., the number of packets successfully transmitted by player j over all the available channels/access points. The evolution of queue length can be written as $q_j(t+1) = \max\{0, q_j(t) - DQ_j(t)\} + AQ_j(t+1)$. The term max is used in order to capture the fact that $DQ_j(t) \leq q_j(t)$ at any time t. There is a departure from player j queue only if (i) its queue is non-empty, (ii) player j is the only one to transmit a packet to one of the access points (channels) and hence all the other players have no departure on that particular access point, (iii) its channel conditions on that access point are good. Thus, the queue dynamics of the users will be correlated through interactions via the departure processes $(DQ_j(t))_{j,t}$. An important issue is the boundedness of the queue length as time goes. A simple condition for the queue length to remain almost surely bounded is if the arrival rate is less than the success probability over both channels. This means that $\sum_j \mathbb{E}AQ_j < \mathbb{E}(\alpha_1 + \alpha_2) \leq 2$. Under the strategy π the condition yields

$$\mathbb{E}(AQ_1 + AQ_2) < \mathbb{E}(x_{1,1}(1-x_{2,1})\alpha_1 + x_{1,2}(1-x_{2,2})\alpha_2)$$

which achieves the bound $\mathbb{E}(\alpha_1+\alpha_2)$ under the equilibrium that is also global optimum.

Coalition formation over geographical area: Some of the players may be moving around a geophical area, thus, the list of available access points may be changing and the list of possible partners to form a coalition is also changing due to low connectivity. The geographical area is represented by a graph (Ve, Ed) where Ve is the set of the vertices (access points) and Ed is the list of some pairs of access points that are connected by links. This leads us to a game with action-constrained in time-varying environment. The action space is therefore function of queue-length and location of the player.

Let $V(t, C)$ be the value associated to the coalition C starting from t. This value is exactly the probability of success of the coalition C over the two channels and can be computed in function of the queue length under equilibrium strategies.

The dynamical Shapley value is a fair single-valued solution concept for dynamic coalitional games. Shapley's original goal was to answer the question "How much would a user be willing to get or pay for participating in a coalition formation game? Plainly, the answer to that question depends on how much the user expects to receive when he comes to play the coalition formation game. As we know that the core may be empty and grand coalition becomes unstable for higher cost of cooperation, we work directly with dynamical sustainable alllocation structure.

The optimal payoff per coalition satisfies the Bellman equation

$$v_j(t, l_j, q_j) = \max_{a_j \in \mathcal{A}(q_j, l_j)} \{r_j(t) + \mathbb{E}v_j(t+1, l'_j, \max\{0, q_j - DQ_j(t)\} + AQ_j(t+1))\}$$

where $\mathcal{A}(q_j, l_j)$ is the set of actions available from location l_j when its queue length is q_j.

Our goal is to design a fully distributed efficient, accurate and convergent learning algorithm to a global optimum of the game with time-varying and queue-aware action space. In order to define the learning pattern we introduce a long-run setup. By doing so, a player will have opportunity to revise her strategy during the game.

Pursuit CODIPAS learning: We introduce a particular learning algorithm, called pursuit CODIPAS learning. Let $x_{j,i}(t)$ be the probability for player $j \in \mathcal{N}$ to choose the action i, at time t, $r_j(t)$ its perceived/measured payoff and $\hat{r}_j(t) = (\hat{r}_{j,i}(t))_{i \in \mathcal{A}}$ be its estimated payoff per action. The action process of j is denoted by $a_j(t)$. Let $\mathrm{EBR}(\hat{r}_j(t))$ is the estimated best response strategy, i.e. a uniform distribution over the set

$$\left\{ i \in \mathcal{A} : \hat{r}_{j,i}(t) = \max_{k \in \mathcal{A}} \{\hat{r}_{j,k}(t)\} \right\},$$

and the learning rate $\lambda_{j,t} \in [\lambda_{j,\min}, \lambda_{j,\max}]$ depends on the private history $H_{j,t}$.

The pursuit CODIPAS learning algorithm is given by

$$\text{for each } j \in \mathcal{N}, \quad (1)$$

$$\text{Initialization: } x_j(0),$$

$$\text{Estimation of the optimal strategy:}$$

$$x_j(t+1) - x_j(t) = \lambda_{j,t} \left[\mathrm{EBR}_j(\hat{r}_j(t)) - x_j(t) \right], \quad (2)$$

$$\text{Estimation of the expected payoff:}$$

$$\text{for each } i \in \mathcal{A},$$

$$\hat{r}_{j,i}(t+1) - \hat{r}_{j,i}(t) = \mathbb{1}_{\{a_j(t)=i\}} \frac{1}{\theta_{j,i}(t)} \left(r_j(t) - \hat{r}_{j,i}(t) \right) \quad (3)$$

$$\theta_{j,i}(t+1) = \theta_{j,i}(t) + \mathbb{1}_{\{a_j(t)=i\}}, \quad (4)$$

For every player j, the algorithm requires the knowledge of a numerical measurement $r_j(t)$ and her local clock $\theta_{j,i}(t)$ which counts how many times the action i has been picked up to t.

3. MAIN RESULTS

In this section we present two main results (Propositions 1 and 2).

Proposition 1. (Convergence of pursuit CODIPAS). In a two users anticoordination game, for every $\lambda_j \in (0, 1]$, the pursuit CODIPAS learning algorithm converges almost surely to one of the global optima, i.e., $(x_{1,1}(t), x_{2,1}(t)) \to (1, 0)$ or $(0, 1)$ as time grows whenever it is feasible with the respect the queue and the remaining energy.

This is an important convergence result since it holds not only for small λ but also for big λ (around 1).

Proposition 2. (Convergence time). The following statements hold:

- The pursuit CODIPAS (2)-(4) is faster than the standard reinforcement learning of Bush-Mustoller (1955).
- In anticoordination matrix games, the pursuit CODIPAS converges almost surely in finite time if $\lambda_1 =$

$\lambda_2 = 1$ starting with estimations $\hat{r}_{1,1} = \hat{r}_{1,2} = \hat{r}_{2,1} = \hat{r}_{2,2} = 0$. Furthermore, the time of steps the system need to arrive to one of the global optimum has geometric distribution of parameter $\frac{1}{2}$.

- The pursuit CODIPAS (2)-(4) is faster than fictitious play.

Remark 2. Let GO be the set of global optima of the anti-coordination game, i.e., $GO = \arg\max_{\pi \in \Delta^n_{m-1}} \{\sum_{j \in \mathcal{N}} r_j(\pi)\}$ where we used by abuse notation $r_j(\pi)$ as the mixed extension of the payoff $r_j(a)$ i.e., $r_j(\pi) = \mathbb{E}_{a \sim \pi} r_j(a)$.

Following Martin and Tilak [2012], we say a learning algorithm is almost-optimal if, for any $\epsilon > 0$, and for any $\delta \in (0,1)$, there exists $T^* = T^*(\epsilon, \delta, m)$ and $\lambda^* = \lambda^*(\epsilon, \delta)$ such that

$$\mathbb{P}\left(\inf_{go \in GO} d(x_t, go) < \epsilon \right) > 1 - \delta$$

for all $t > T^*$ and $\lambda \in (0, \lambda^*)$. $d(x_t, go)$ denotes the distance between the vector x_t and the vector go. Our result above implies that the proposed learning is almost-optimal.

Effect of λ: The pursuit CODIAS converges for any learning rate $\lambda \in (0, 1]$. However, the convergence time may differ. It is interesting to notice that once the system is in a good state, the probability to stay there increases with λ. In other words, if the vector $(\hat{r}_{j,1}(t_0), \hat{r}_{j,2}(t_0))$ is such that $\hat{r}_{1,1}(t_0) > \hat{r}_{1,2}(t_0)$, and $\hat{r}_{2,1}(t_0) < \hat{r}_{2,2}(t_0)$ at a certain time t_0 then, $P((a_1(t), a_2(t)) = (1,2))$, for every time $t > t_0$ increases with λ.

Very fast coalition formation algorithm: The pursuit coalitional CODIPAS learning algorithm aims to form coalitions between the players in order to play a joint-effort maximizing action.

$$x_j(t+1) - x_j(t) = \lambda_{j,t} \left[\mathrm{EBR}_j(\hat{r}_j(t)) - x_j(t) \right], \quad (5)$$

$$\hat{r}_{j,C}(t+1) - \hat{r}_{j,C}(t) = \frac{\mathbb{1}_{\{a_j(t) = C\}}}{\theta_{j,C}(t)} \left(r_j(t) - \hat{r}_{j,C}(t) \right) \quad (6)$$

$$\theta_{j,C}(t+1) = \theta_{j,C}(t) + \mathbb{1}_{\{a_j(t) = C\}}, C \in 2^{\mathcal{N}} \backslash \{\}, \quad (7)$$

For every player j, the coalitional CODIPAS algorithm requires the knowledge of a numerical measurement $r_j(t)$ and her local clock $\theta_{j,C}(t)$ which counts how many times the coalition C has been formed up to t.

For two players the possible coalitional structures are $\{\{1\},\{2\}\}$, $\{\{12\}\}$. The coalitional structure $\{\{1\},\{2\}\}$ corresponds to fully non-cooperative case and the coalitional structure $\{\{12\}\}$ corresponds to full cooperation (grand coalition). The payoff inside a coalition is the Shapley value associated the success of the coalition minus the cost of making the corresponding coalition.

As a corollary of the above Propositions 1 and 2, if the cost of making coalition is small enough compared to α then CODIPAS converges to a grand coalition and the convergence time is faster than the one provided in Tembine [2012].

Thanks to big and non-vanishing learning rate our convergence time is better than most of the known learning algorithms. In addition, our coalition formation algorithm is accurate in the sense that it provides quickly a very small error to an optimal solution.

4. CONCLUDING REMARKS

In this paper, we have examined the convergence properties of pursuit CODIPAS to global optima in anticoordination games under queue dynamics. In contrast to the stochastic approximation framework that is widely used in classical learning algorithms in games, we have constructed a direct and simple proof of convergence and provided a convergence time bound by a geometric law. Our approach works for all range of learning rates. In addition, the convergence time of the proposed scheme is carefully studied and is bounded by a geometric law which is faster than the standard learning algorithms in games including fictitious play and relative entropy-driven CODIPAS.

REFERENCES

Bellman, R. (1952). On the theory of dynamic programming. *Proceedings of the National Academy of Sciences of the United States of America*, 38(8), 716.

Brown, G. (1951). Iterative solutions of games by fictitious play. *In Activity Analysis of Production and Allocation, T.C. Koopmans (Ed.), New York: Wiley.*

Bush, R.R. and Mosteller, F. (1953). A stochastic model with applications to learning. *The Annals of Mathematical Statistics*, 559–585.

Kushner, H.J. and Yin, G. (2003). *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer Verlag.

Liu, Y. and Liu, M. (2013). To stay or to switch: Multiuser dynamic channel access. *INFOCOM, Turin,Italy.*

MacKenzie, A.B. and Wicker, S.B. (2003). Stability of multi-packet slotted aloha with selfish users and perfect information. *in Proc. IEEE Int. Conf. Computer Communications (IEEE INFOCOM), USA*, 3, 1583–1590.

Martin, R. and Tilak, O. (2012). On ε-optimality of the pursuit learning algorithm. *Journal of Applied Probability*, 49(3), 795–805.

Nguyen, T.V. and Baccelli, F. (2012). On the spatial modeling of carrier sensing medium access wireless networks by random sequential packing models. *INFOCOM, Orlando, FL.*

Sarikaya, Y., Alpcan, T., and Ercetin, O. (2012). Dynamic pricing and queue stability in wireless random access games. *IEEE Journal of Selected Topics in Signal Processing*, 6(2).

Shapley, L.S. (1953). Stochastic games. *Proceedings of the National Academy of Sciences of the United States of America*, 39(10), 1095.

Tekin, C. and Liu, M. (2011). Online learning in opportunistic spectrum access: A restless bandit approach. *INFOCOM.*

Tembine, H. (2012). *Distributed strategic learning for wireless engineers.* CRC Press.