Reinforcement Learning in Extensive Form Games with Incomplete Information: the Bargaining Case Study

Alessandro Lazaric  
Politecnico di Milano, DEI  
piazza Leonardo da Vinci 32, I-20133 Milan, Italy  
lazaric@elet.polimi.it

Enrique Munoz de Cote  
Politecnico di Milano, DEI  
piazza Leonardo da Vinci 32, I-20133 Milan, Italy  
munoz@elet.polimi.it

Nicola Gatti  
Politecnico di Milano, DEI  
piazza Leonardo da Vinci 32, I-20133 Milan, Italy  
gatti@elet.polimi.it

ABSTRACT
We consider the problem of finding optimal strategies in infinite extensive form games with incomplete information that are repeatedly played. This problem is still open in literature. Game theory allows to compute equilibrium strategies exclusively in presence of perfectly rational agents and a common prior, but these assumptions are unrealistic in several practical settings. When these assumptions do not hold, the resort to learning techniques is common. Nevertheless, learning literature does not provide a mature solution for the problem above. In this paper we present a novel learning principle that aims at avoiding oscillations in the agents’ strategies induced by the presence of concurrent learners. We apply our algorithm in alternating-offers bargaining with deadlines, and we experimentally evaluate it showing that using this principle self-interested reinforcement learning algorithms can improve their convergence time.

1. INTRODUCTION
In several multi-agent applications (e-commerce, political negotiations, board and card games, just to mention a few) agents take their moves sequentially according to an order fixed by an interaction protocol. The “natural” analysis of such situations is provided by game theory. Specifically, game theory models such situations as extensive form games [3] and provides predictions about how the game should be played by perfectly rational agents in presence of a common prior. When at least one of the these assumptions on the basis of game theory, i.e. perfectly rationality and common prior, drops, alternative techniques to the game theory are needed. It is in these situations that learning approaches enter the picture.

Multi-agent learning emerges from artificial intelligence to reason on the interaction between multiple agents perceiving, reasoning, and acting in a common dynamical environment. Multi-agent learning approaches can be split into two main categories: those that build beliefs about the future behavior of other agents and then act accordingly, and those that exclusively consider their own expected utility and try to maximize it. The methods in the first category are commonly addressed as fictitious-play methods, while the others are Reinforcement Learning (RL) methods. In this work we address learning methods belonging to the latter category.

RL is an attractive learning technique that has been extensively studied in artificial intelligence. Through repeated interactions, RL techniques allow an agent to learn the optimal policy, i.e. that policy maximizes in any state the infinite-horizon discounted reward or the long-run average reward. These results hold only for single-agent learning problems, i.e. both single-agent problems and multi-agent problems where only one agent learns while the others follow stationary policies. On the other hand, multi-agent learning deals with problems where a collection of autonomous and self-interested agents concurrently learn. Multi-agent reinforcement learning comes as the common extension of single-agent RL, and the associated cost of such extension is that the guarantees do not transfer directly to the multi-agent context. Differently from single-agent RL, multi-agent learning is a young research field where several problems are still open.

In this paper we focus on the application of the RL approach to a subclass of multi-agent problems: extensive form games with incomplete information. It is worth noting that most work in multi-agent RL is centered on learning in repeated strategic form games, where the agents, at each step, simultaneously choose their actions and each of them takes a payoff according to the joint action performed. The application of RL techniques in extensive form games with incomplete information is still largely unexplored. To the best of our knowledge, the most relevant result in these settings is the one presented by Huang and Sycara [8]. They present two multi-agent learning algorithms (a learning automata and a reinforce based algorithm) that provably converge in self-play to the subgame perfect equilibrium in finite extensive form games with incomplete information assuming that there are no terminal nodes that give to an agent the same payoff. For what regards the reinforce based learning algorithm, the convergence theorem postulates that it is pos-

1 In [8] the expression “complete information” is used to characterize games in which agents know at any node of the game tree the actions previously executed by all the other agents. Following the classic literature [3], we use “observable action” when referring to such games, while the expression “complete information” is used for games where each agent knows exactly all the parameters of the game, e.g. the payoffs of all the other agents.
sible to approximate with arbitrary precision the subgame perfect equilibrium reward in any node of the game tree, by keeping the exploration probability below a threshold that is function of the required accuracy (higher accuracy asks for lower exploration probabilities). On the other hand, exploration strategies play a central role in RL algorithms, and low exploration probabilities typically lead to very long convergence times, that may be not affordable in practical applications. In fact, unless the game is played infinite times, we are not interested in the asymptotic result, rather we would prefer to maximize the payoff collected during the learning process.

In this paper we propose a general learning principle (CoLF: Change or Learn Fast) that, by opportunely modifying the learning rate, aims at reducing the non-stationary effects induced by explorative actions performed by the other learning agents. If all the learning agents adopt the CoLF principle exploration rates may be kept higher, thus significantly reducing convergence times.

The experimental activity has been focused on a specific case study: a slight variation of Rubinstein’s alternating-offers bargaining [15], where agents have deadlines. We experimentally show that when a best response learner is coupled with a fixed policy rational agent that holds complete information, the learner is able to converge to the equilibrium solution; when coupled with a suboptimal (irrational) agent, the learner exploits this vulnerability and outperforms her competitors; and when coupled with another learner of its same kind, the two learners are able to converge to the equilibrium strategies, in this case subgame perfect. In all these settings, experimental results show that the CoLF principle applied to the Q-Learning algorithm drastically improves the convergence speed.

The remainder of the paper is structured as follows: next section formally introduces extensive form games and discusses some issues that arise when learning is applied to these games. Section 3 introduces how an extensive form game can be translated into a stochastic game, thus allowing the straightforward application of Q-learning (a popular RL algorithm) and its modified version through the CoLF principle. Section 4 describes the formal model of the alternating-offers bargaining with deadlines and its translation as a stochastic game. In the experimental section we present comparisons between two different parameterizations of Q-Learning and a version of Q-Learning based on the CoLF principle on alternating-offers bargaining against stationary agents and in self-play. Finally, we discuss some relevant points raised by the experimental analysis when there are indifferent actions and two subgame perfect equilibria are present. Conclusions and future research directions conclude the paper.

2. EXTENSIVE FORM GAMES AND LEARNING

In the following we describe the formal framework used to model extensive form games and discuss some issues arising when learning techniques are applied to such games.

2.1 Extensive Form Games

The model of an extensive form game, by contrast to the strategic one, describes the sequential structure of decision making explicitly, allowing the study of situations in which each agent is free to change her mind as events unfold [3]. As a strategic model, the extensive model specifies the set of agents, their allowed actions, and their preferences. In addition, it specifies the order of the agents’ moves and the actions each agent may take at any time she can make a move. Commonly, the time structure of an extensive form game is described resorting to a game tree with the following properties. The first move of the game is identified with a distinguished node called the root of the tree. The nodes represent the possible moves in the game and, except the root, can be terminal or non-terminal. If a node is terminal, then the game ends at such a node with a specific outcome. If a node is non-terminal, then it is assigned to one or more actions that can act at that move. The edges leading away from a node represent the choices or actions available at that move. Formally, an extensive form game is defined by a tuple \( (N, G, \sigma(s), \{A_i(s)\}, \{\Delta_i\}) \), where \( N \) is the set of the agents, \( G \) is a tree, \( \sigma(s) \) is the player function that gives the agents that play at node \( s \in G \), \( A_i(s) \) is the set of actions (or edges in the tree) allowed to agent \( i \) at node \( s \), and \( \Delta_i \) is the preference relation of agent \( i \) over the outcomes.

An extensive form game can be at finite-horizon (or at infinite-horizon), finite (or infinite), with observable actions (or with unobservable actions), and with complete information (or with incomplete information) [3]. Specifically, if the length of the longest possible play is in fact finite, it is said to be at finite horizon. If a game has a finite horizon and finite many actions, then it is said to be finite. If in a game each agent knows all actions chosen previously and moves alone in each node, then the game is said to be with observable actions. Finally, if in a game each agent knows with certainty her opponents’ payoffs and allowed actions and such information is common, then the game is said to be with complete information.

Extensive form games are commonly solved by game theoretic tools finding agents’ strategies such that they are a Nash equilibrium. When the information is complete, the appropriate notion of equilibrium for an extensive form game is subgame perfect equilibrium [6], i.e. a Nash equilibrium in each possible subgame, where a subgame is defined as the part of the game that follows from a non-terminal node [3]. When the information is incomplete, it is customary the introduction of a probability distribution over the unknown parameters of the game and the consequent assumption that the knowledge of such a probability distribution is common among the agents. In this situation the appropriate notion of equilibrium is Kreps and Wilson’s sequential equilibrium [9] where the equilibrium is made up by the equilibrium strategies and a system of beliefs that gives the beliefs of the agents in each possible node of the game tree.

2.2 Learning Issues in Extensive Form Games

Reinforcement learning techniques are commonly employed in a multiagent context to learn a best-response strategy [7]. They play a crucial role in all those situations that cannot be tackled by game theory in which agents are not perfectly rational or there is not any common prior. Such situations are very common in real settings, for instance agents can be computationally bounded, not being able to scan all their alternatives and choice the best one. Or agents may not know opponents’ payoffs. Rigorous game theory can tackle games with every information structure, even with prior not common, but the complexity of solving these prob-
lems make them practically intractable [12]. In all these situations reinforcement learning provides techniques to allow self-interested agents to learn optimal strategies by repeatedly play against their opponents without needing any prior information and common knowledge assumptions.

Multiagent reinforcement learning literature centers its attention in translating zero-sum stochastic games in the context of reinforcement learning. His algorithm, Minimax-Q, is an extension of the traditional single agent Q-Learning algorithm and converges to the stochastic game’s equilibrium solution. Afterwards, a series of works [7, 11, 5] tried to extend this result to general-sum games at the cost of loss in convergence guarantees, harder constraints, and computational time. In practice, none of these extensions for general-sum games can be successfully applied. Bowling and Veloso showed in [16] a variable learning rate principle (WoLF) that is guaranteed to converge to a Nash equilibrium in two-person, two-action, iterated general-sum games; they showed also a practical algorithm that merges the WoLF principle and policy hill-climbing (WoLF-PHC) in a multi-robot application.

Previous multiagent reinforcement learning techniques, as the ones described above, could be also used to solve some extensive form games by transforming them into an equivalent normal form version of the game as prescribed in game theory [3]; each strategy in the normal form corresponds to a plan of actions in the extensive form. Nevertheless, there are several drawbacks performing such transformation. In most cases the resulting matrix from such transformation will frequently be exponential in the size of the original game, furthermore, it is impossible to perform a transformation from an infinite extensive form game both when the actions are infinite and when the horizon is infinite. Another side problem from performing a transformation is with regards to the appropriate notion of equilibrium. This techniques have been designed for convergence to Nash equilibria, not specifically to the appropriate notions of equilibrium for an extensive form game.

To avoid these drawbacks, it is necessary to find solutions directly from the original game tree. There is little work in multiagent reinforcement learning literature that falls into this category. One worth noticing work [8] identifies a special type of extensive form games suitable for making Q-Learning with some specific parameterization converge under this type of game structures. Their convergence guarantees pose that for any deviation $\varepsilon$, there exists $0 < \sigma < 1$, such that if the exploration threshold of every player is less than $\sigma$, the expected value of an action in any subgame of the original game converges to the $\varepsilon$-range of the subgame perfect equilibrium reward. Although this is an encouraging result, its constrains are a little less encouraging. They impose an strong restriction in the structure of the extensive form game, namely, the extensive form game must be finite and must not have terminal nodes that give to an agent the same payoff. Furthermore, as well-known in the learning community, there is a strong trade-off between exploration, optimality of the solution, and convergence time, and given that their work does not address this issue, we can argue that this trade-off becomes a crucial aspect in learning. In the sense that, in large state spaces, one may need to increases $\sigma$ to effectively explore the whole state space, which by their theorem causes $\varepsilon$ to increase, thus meaning the convergence to a larger region around the optimal value. As a result, due to the possible of overlapping of two of these regions, the learning process is likely to fail to achieve subgame perfect equilibrium. On the other side, keeping $\sigma$ low can drastically increase convergence time.

Given this results, which offers us nice theoretical groundings, we build a more comprehensive formulation from a learning perspective specifically dealing with the problems stated before. In particular we consider as case study a game more general than the game considered in [8] allowing infinite actions and terminal states with the same reward. The case study we consider is one of the most significant games in multiagent community: alternating-offers bargaining with deadlines [15].

### 3. Applying Reinforcement Learning in Extensive Form Games

Up to now, the application of multi-agent reinforcement learning to extensive form games has been quite limited. In the following we propose a way to translate extensive form games into the more general framework of stochastic games, thus allowing a straightforward application of traditional RL algorithms (like Q-Learning).

#### 3.1 From Extensive Form to Stochastic Games

In order to apply learning algorithms to extensive form we adopt the framework of general-sum stochastic games. In a stochastic game, each agent’s reward depends on the current state, while state transitions obey the Markov property. In its general form it is defined by the tuple $\langle N, S, \{A_i(s)\}, T, \{R_i(s, a)\} \rangle$ where $N$ is the set of $n$ agents, $S$ is a set of states, $A_i(s)$ is the agent $i$’s set of actions at state $s \in S$, $T$ is the probability transition $S \times A(s) \rightarrow PD(S)$, where $A(s)$ is the set of joint actions in state $s$, i.e. $A(s) = \times_{i=1}^n A_i(s)$, and $PD(S)$ is a probability distribution over $S$, and $R_i(s)$ is agent $i$’s reward function. Let $\pi_i : S \rightarrow A_i$ and $\pi(s)$ the functions that give agent $i$’s strategy in state $s$ the agents’ joint strategy, respectively.

In a stochastic game the individual goal of each agent is to maximize its expected sum of discounted rewards. During the learning process, each agent $i$ saves the estimation of the expected sum of her discounted rewards in each state $s$ for action $a \in A_i(s)$ in the action value function $Q_i(s, a) = E[\sum_{t=0}^{\infty} \gamma^t R_i(s_t)]$, where $\gamma \in [0, 1]$ is the discount factor.

The solution concept appropriate for stochastic games is the joint strategy $\pi^* = (\pi_1^*, \pi_2^*, \ldots, \pi_n^*)$ at the Nash equilibrium, i.e. $\forall i$, it holds

$$Q_i(s, \pi_1^*, \ldots, \pi_i^*, \ldots, \pi_n^*) \geq Q_i(s, \pi_1^*(s), \ldots, \pi_i(s), \ldots, \pi_n^*(s)).$$

When only one of the agents of $N$ is a learning agent and all the others play a fixed stochastic strategy, the stochastic game trivially reduces to a Markov Decision Process (MDP). In that case, many learning algorithms (e.g., Q-Learning) are guaranteed to converge to the optimal strategy by repeatedly interaction with the environment. On the other hand, when many agents simultaneously learn, each agent perceives a non-stationary environment and the learning process is likely to fail.

\footnote{In general the reward function is defined as $R_i(s, a, s')$ where $a$ is the joint action and $s'$ is the arrival state, but we restrict our attention to state reward functions without loss of generality.}
The translation of an extensive form game into a stochastic game is straightforward. The mapping of agents and actions is trivial. All the nodes in the tree $G$ of an extensive form game can be mapped to the states in the stochastic game. The main difference between stochastic and extensive games is about the transition model and the reward function. In fact, while in stochastic games all agents choose their actions at the same time, in extensive form games only a part of the agents takes an action at a time. In order to make the translation from extensive to stochastic consistent, it is necessary to introduce a pass action, that is the only action available to agents outside their turns. Formally, if $i(s) \neq i$ it holds $A_i(s) = \{\text{pass}\}$. As far as the reward function is concerned, in stochastic games it is defined as a real value for each state-action pair. On the other hand, in extensive games the payoff is defined only in the terminal nodes where the game concludes. Thus, it is necessary to force the reward function to be zero in all the states but the states corresponding to the terminal nodes. Formally, if $s$ is a non-terminal node it holds $R_i(s) = 0$. Furthermore, the reward is accumulated every time the agent performs a pass until it is her turn. It is worth noting that the translation from extensive to stochastic game does not constrain neither the state nor the action spaces to be finite. Thus, also games as the alternating-offers bargaining can be easily translated into a stochastic game. Furthermore, by the equivalent between the solution concept of a stochastic game and the notion of subgame perfection in an extensive form games, follows that the optimal solution for a stochastic game corresponds to the subgame perfect equilibrium of the original extensive form game.

### 3.2 The CoLF Learning Principle

As previously stated, in stochastic games the performance of the learning process of each agent is negatively affected by the concurrent changes in the agents strategies. In fact, every time an agent changes its policy, all the other self-interested RL agents perceive a non-stationarity in the environment dynamics. In this context no known learning algorithm is guaranteed to converge.

The CoLF (Change or Learn Fast) principle is inspired by the work of Bowling and Veloso [16], where a variable learning rate is considered. According to the CoLF principle, the learning rate of the algorithm is set as follows: if the achieved outcome is unexpectedly changing, then learn slowly, otherwise learn quickly. This principle gives less importance to “unexpected” outcomes (i.e. payoffs that are quite different from those achieved recently in the same state), probably generated by non-stationary causes like exploration activity or normal learning dynamics, while allowing to speed up learning when the agents are playing near-stationary strategies. This principle is very general and may be applied to a variety of different learning algorithms. In the following we show how it can be implemented in Q-Learning.

At each step of Q-Learning algorithm, when agent $i$ takes action $a$ in state $s$, the action value function is updated with the following update formula:

$$Q_i(s, a) = (1 - \alpha)Q_i(s, a) + \alpha(R_i(s) + \gamma \max_{a'} Q_i(s', a'))$$

where $\alpha$ is the learning rate. In order to explore all the states and action space, at each time step agent $i$ plays an $\epsilon$-greedy strategy that takes her best action $a = \arg \max_a Q_i(s, a)$

with probability $1 - \epsilon$ and one random action with probability $\epsilon$.

As stated before, while in general stochastic games Q-Learning is not guaranteed to converge, in a subclass of extensive form games it is possible to prove that Q-Learning succeeds to approximate the exact values with an error $\xi$ when the exploration factor $\epsilon$ is under a given threshold. Although relevant, this theoretical result does not provide any information about the convergence speed of the algorithms and, as we discuss in the experimental section, could be a severe obstacle to the application of learning algorithms to significant problems. In fact, RL algorithms benefit from a long and exhaustive exploration in order to achieve optimal strategies. At the same time, in stochastic game, high exploration factors may contribute to the non-stationarity perceived by the agents. Therefore, when the outcome is significantly different from what expected, the CoLF principle assumes that this is due to a non-stationarity of the environment and states that a low learning rate should be used. As a result, the agent does not immediately modify her strategy because of the unexpected outcome, so that she can verify whether it is caused either by exploration or by an actual change in the other agents’ strategies.

In Algorithm 1 we report how the Q-learning algorithm changes with the introduction of the CoLF principle. For each pair $(s, a)$, besides the Q-value, the algorithm needs to store and update also the S-values. The S-values are exponential averages with weight factor $\lambda$ of the absolute differences between the current outcome $T$ and the corresponding Q-value. The algorithm requires two learning rates $\alpha_{NS}$ and $\alpha_S$, with $\alpha_{NS} > \alpha_S$. The choice of which learning rate should be used to update the Q-value associated to the pair $(s, a)$ depends on whether the absolute difference between the current outcome and the Q-value is greater than $k$ times the S-value.

### 4. THE CASE STUDY: ALTERNATING-OFFERS BARGAINING

One of the main applicative scenario in which extensive form games can be used to model multi-agent interactions is e-commerce, where autonomous agents are engaged in negotiations. In particular, we address the bilateral bargaining

---

**Algorithm 1 CoLF — Change or Learn Fast**

Let $\alpha_{NS} > \alpha_S$, and $\lambda$ be learning rates, $k > 1$

$$S(s, a) = 0, \forall s \in S, a_i \in A_i$$

$$Q(s, a_t) \rightarrow \min_{s \in S, a_i \in A_i}$$

for all steps do

choose action $a_i^t$ according to exploration strategy

execute $a_i^t$ and get the payoff $r^t_i$

read the state $s^t$

compute $T^t_i = r^t_i + \gamma \max_{a} Q(s^t, a)$

$\Delta T^t_i = |T^t_i - Q(s^{t-1}, a^t)]$

if $\Delta T^t_i > kS(s^{t-1}, a^t)$ then

$\alpha_i = \alpha_{NS}$

else

$\alpha_i = \alpha_S$

end if

$$Q(s^{t-1}, a^t) \leftarrow (1 - \alpha_i)Q(s^{t-1}, a^t) + \alpha (r^t_i + \gamma \max_{a} Q(s^t, a))$$

$$S(s^{t-1}, a^t) \leftarrow (1 - \lambda)S(s^{t-1}, a^t) + \lambda \cdot |\Delta T^t_i|$$

$t \rightarrow t + 1$

end for
situations characterized by two agents negotiate to reach an agreement on conflicting issues. Besides the applicative relevance, the choice for this case study is motivated also by some peculiar aspects of this extensive form game (such as infinite horizon and infinite actions) that make this problem quite challenging when faced with incomplete information.

4.1 The Formal Model

Alternating-offers bargaining with deadlines is essentially a finite horizon extensive form game with infinite actions and perfect information, where two agents, a buyer $b$ and a seller $s$, try to agree on the value of one or more parameters. The agent that acts at time $t$ is given by the player function $\iota : \mathbb{N} \rightarrow \{b, s\}$ where $\iota$ is such that $\iota(t) \neq \iota(t+1)$. The set of allowed actions depends only on time $t$. Specifically, the allowed actions of agent $\iota(t)$ at time $t > 0$ are $(1)$ offer $(\mathfrak{T})$, where $\mathfrak{T} \in \mathbb{R}$, $(2)$ exit, and $(3)$ accept; whereas at $t = 0$ agents can make only $(1)$ and $(2)$. If at $t$ agent $\iota(t)$ makes accept the game stops and the outcome is $(\mathfrak{T}, t)$, where $\mathfrak{T}$ is the value offered by agent $\iota(t-1)$ at time $t-1$. If at time $t$ agent $\iota(t)$ makes exit the game stops and the outcome is NoAgreement. Otherwise the bargaining continues to the next time point.

The gain of an agent $i$ in reaching an agreement $(x, t)$ is given by an utility function $U_i : (\mathbb{R} \times \mathbb{N})_i \cup \{\text{NoAgreement}\} \rightarrow \mathbb{R}$ that depends on three parameters of agent $i$: the reservation price $R_P$, the temporal discount factor $\delta$, and the deadline $T_i$, $\delta \in (0, 1]$, and the deadline $T_i \in \mathbb{N}$, $T_i > 0$. Exactly, if the outcome of the bargaining is an agreement $(x, t)$, then the utility functions $U_b$ and $U_s$ are respectively:

$$U_b(x, t) = \begin{cases} (R_P - x) \cdot \delta^t & \text{if } t \leq T_b, \\ 0 & \text{otherwise} \end{cases}$$

$$U_s(x, t) = \begin{cases} (x - R_P) \cdot \delta^t & \text{if } t \leq T_s, \\ 0 & \text{otherwise} \end{cases}$$

if the outcome is NoAgreement, then $U_b(\text{NoAgreement}) = U_s(\text{NoAgreement}) = 0$.

It is easy to see that there in alternating-offers bargaining there are several terminal nodes that give to the agents the same payoff. For instance, at time $t$ the acceptance of $R_P(t)$ and the exit from the game gives to $\iota(t)$ the same payoff, exactly 0.

When actions are infinite, i.e., when an agent can make any offer $x \in \mathbb{R}$, and information is complete, alternating-offers has a unique subgame perfect equilibrium. The equilibrium path is made up by a sequence of offers from the deadline to the initial time such that, at the equilibrium, they would be accepted. Such a sequence of offers prescribes that at the earlier deadline $T_i$ agent $i$ accepts her $R_P$, at time $t = T_i - 1$ agent $i$'s opponent would offer $R_P$, and for any $t < T_i - 1$ the equilibrium offer at $t$ is the one that gives to the agent $\iota(t+1)$ the same utility given by offering at $t+1$ her equilibrium offer. When actions are finite and information is complete, alternating-offers has two subgame perfect equilibria; later we refer to these as SPE1 and SPE2, respectively. The first is exactly the one when actions are infinite. The second is similar: at the earlier deadline $T_i$ agent $i$ accepts the discretized offer in $(R_{P_a}, R_{P_b})$ closest to her $R_P$, at time $t = T_i - 1$ agent $i$'s opponent would make such an offer, and for any $t < T_i - 1$ the equilibrium offer at $t$ is the one that gives to the agent $\iota(t+1)$ the same utility given by offering at $t+1$ her equilibrium offer. Such equilibria can be easily found by backward induction [13].

4.2 Bargaining Translation

The translation of the bargaining model can benefit from some regularities characteristic of the problem in order to reduce the dimensionality of the corresponding stochastic game. In general, each state in the tree of the extensive form needs to be translated into a state of the stochastic game in order to preserve the structure of the game. In case of bargaining, a state is fully determined by the current time and the offer made by the opponent at the previous time point. The sequence of offers that led to that state does not influence the future strategies and outcomes of the game. This means that it is not necessary to map each state of the tree into a stochastic game state, but only the current time and opponent’s last offer could be used as state variables for the stochastic games. This results in a dramatic reduction of the dimensionality of the problem and makes the application of learning techniques more feasible.

More formally, this translation is admissible every time the dynamics of the turn-based game shows a Markov-like (i.e., memoryless) property, that is the information about the time point $t$ and the joint action $a(t)$ are enough to define the state $s$ at time $t+1$. In that case, the extensive form game can be compressed into a stochastic game that completely ignores the history of the system.

In the translation of alternating-offers bargaining to stochastic games, the reward function is non-zero for both the agents only for termination states (except for the terminal state NoAgreement in which it returns 0 to both agents) and does not return the utility of the entire negotiation but is equal to the accepted offer. Thus, for the buyer the reward function is:

$$R_b(x, t) = \begin{cases} (R_P - x) & \text{if } t \leq T_b, \\ -1 & \text{otherwise} \end{cases}$$

This definition is consistent with the individual goal of each learning agent that tries to maximize the sum of its expected reward. In fact, given a discount factor $\gamma$ equal to the temporal discount factor $\delta$ the expected payback becomes $E[\sum_{j=0}^{\infty} \gamma^j R_s(s_j)] = \gamma^T R_b(s_T)$. Thus, the action value function is exactly the utility function defined in Section 4.1.

5. EXPERIMENTAL RESULTS

IN THE BARGAINING GAME

In the following experiments, we are interested in investigating whether the application of learning algorithms is feasible when no information about the type (i.e., discount factor, deadline, and payoffs) of the opponent is provided. In particular, we want to verify whether the CoLF learning principle actually allows to reduce the number of repeated interactions required to converge to an optimal strategy in presence of concurrent learners.

Since in the bargaining problem each player has continuous actions (i.e., the offer plus the accept and exit actions), in order to apply traditional RL algorithms based on a tabular representation of the action-value function, we need to...
The discrete set discretize the offer interval (that we assume to be [0, 1], without any loss of generality) into a finite set of available actions with a step $\Delta$. The discretization of the set of actions requires to compute the new sub-game perfect equilibrium of the game. In fact, in the usual backward induction construction of the subgame perfect equilibrium, at each time step the backward propagated offer must be discretized to one of the two adjacent discrete offers according to the rules shown in Algorithm 2.

The results reported in this paper are obtained on the bargaining problem defined by the parameters (deadlines and discount factors for the two agents, and the step of discretization) displayed in Table 1 and $z(0) = s$.

In order to transfer the theoretical results described in [8] to the bargaining domain we need to grant that each agent will not perceive the same payoff reaching different terminal states. For this reason, the two reservation prices must be excluded from the set of available offers (i.e., 0 and 1), and we substitute them with two offers that follows inside the offer interval and are close to the reservation states (in the experiments we use 0.001 and 0.999). Under these settings the game has a unique SPE, that can be achieved by two Q-learners providing that their exploration probabilities are small enough. Given the values chosen for the bargaining problem we fixed the exploration probability for Q-Learning to 0.025 (we refer to this version as Q-Lv1). For comparison we made experiments with a second version of Q-Learning (Q-Lv2) characterized by an optimistic initialization of the Q-values and by a larger exploration probability, so that convergence to the SPE cannot be guaranteed. Finally, we apply the CoLF learning principle to Q-Learning under the same exploration settings of Q-Learning v2 to directly compare their performance. The parameters for the learning algorithms are reported in Table 1. All the experimental data presented in the following have been average over 100 independent runs. Since our main concern is the speed of convergence and the behavior in the first repeated negotiations, the learning processes are limited to 20000 episodes even if the full convergence is not achieved.

The performed experiments can be grouped into three different categories. In the first group we have placed experiments in which a learning agents faces an opponent that follows a stationary policy. The second group collects self-play experiments, i.e. experiment in which the two bargaining agents adopt the same learning algorithm. For the third group we show experimental results in which the agents are allowed to offer also their reservation prices, thus rising new equilibrium solutions.

### 5.1 Learning versus Stationary Agents

The first set of experiments is run with one learning agent versus a stationary agent. In particular, we compared the results of the learning algorithms against a game theoretic agent (GT in the graphs) and a stationary agent that plays a suboptimal strategy (HC in the graphs). In the first case, the learning agents are expected to achieve the optimal strategy and to achieve the unique SPE. In the latter, we show how a learning agent is able to exploit the weakness of the stationary agent and to obtain better performance with respect to that of a game theoretic agent. Finally we compare the learning speed between Q-Learning and CoLF.

In the following we consider only the configuration in which the seller is the learning agent. The other possible configurations do not show any significantly different result.

The graphs in Figure 1 show the performance of the three learning agents against the game theoretic agent in repeated negotiations that always start at time instant 1 and in negotiation whose starting point is randomly extracted. The reason for this second class of experiments is to perform a more complete analysis of the learned strategies wrt to the SPE solution.

In the first case, we report the number of successful negotiations (i.e., the first agent proposes the best offers and the second one immediately accepts) along 5000 learning episodes. Although the learning parameters for QLv1 are more suitable for the reduction of the approximation error, it results to be significantly slower than both QLv2 and CoLF. The same behavior can be noticed in the second graph that shows the number of subgames in which the agent achieved the optimal policy. Even if all the algorithms succeeded in finding the subgame perfect equilibrium in about 5000 learning episodes, both QLv2 and CoLF learn a nearly optimal solutions already in the very first negotiations, while QLv1 performs poorly up to 4000 episodes.

The graph in Figure 2 shows the average payoff for the learning agents against the HC agent with randomly restarted negotiations. This simple experiment shows how learning agents are able to exploit a suboptimal (not fully rational) agent through repeated interaction, while a game theoretic agent with complete information of the game would achieve a very poor performance.

### 5.2 Learning in Self-Play

While the convergence to the optimal solution in stochastic games with one learning agent and several stationary agents is well-rooted in single agent learning theory, when a learning agent faces a non-stationary environment only few theoretical guarantees are available [8]. Nevertheless, it is
often relevant to study the behavior of a learning algorithm when all the agents play the same algorithm. Therefore, the second set of experiments include a comparison of the learning performance of Q-Learning and CoLF when both the buyer and the seller are learning agents with the same algorithm. In this setting, convergence to the subgame perfect equilibrium means that none of the agents succeeds in exploiting the opponent and, as a consequence, they both learn to play an equilibrium strategy.

The graph in Figure 3 shows the percentage of successful negotiations along the learning process, while the graphs in Figure 4 show the subgame optimality for the seller and for the buyer respectively. Unlike the results in the previous section, in this setting it is much more difficult for the learning agents to converge at the SPE within few episodes because of the concurrent learning processes. The only algorithm that has theoretical convergence guarantees is QL1. Although QL2 and CoLF do not have any theoretical guarantee, they show a fast convergence to the SPE solution. In particular, CoLF shows a significant improvement with respect to the other two learning algorithms both in terms of learning speed and optimality of the solution. The experiment with randomly restarted negotiations highlights the intrinsic asymmetry between the seller and the buyer that plays at the deadline and is “forced” to accept profitless offers. This makes more difficult the learning process for the buyer whose performance is quite far from the optimality.

5.3 Learning with Indifferent Strategies

In the experiments reported above, agents were not allowed to offer their own reservation price; in this way the game has only one SPE. As discussed in Section 4.1, when this restriction is removed and agents have discrete actions the game has two SPEs and in the following we analyze how the learning process is affected by this condition.

At first we coupled a game theoretical agent that plays the SPE1 strategy with a learning agent using CoLF. In this context the learning agent is expected to learn the optimal strategy. The game theoretical agent starts the negotiation as the seller, while the learner is the buyer and plays at the deadline. The experiment has been carried out with random restart, and it aims at evaluating the percentage of states in which the learning agent has learned the SPE1 strategy. As shown by Figure 5 the optimal strategy learned does not perfectly overlap with the SPE1 strategy. This result is not surprising; in fact, when the buyer plays on the deadline and the other agent has offered buyer’s reservation price, independently from the action taken by the buyer its payoff will be 0. While in a game with complete information, by knowing the utility of the other agent, the equilibrium strategy is that the buyer at the deadline should accept her own reservation price, in a game with incomplete information the buyer faces several indifferent strategies.

This phenomenon is even more evident when we consider the negotiation between two learners. Figure 6 displays the percentages of optimality of the learned policies (the seller on the left and the buyer on the right) evaluated with respect to the two SPEs strategies of the game. These graphs show that the two learners in presence of these two SPEs converge to SPE2. This result can be explained as follows: since the agent that plays at the deadline (say the buyer), when receiving an offer equal to her reservation price, will randomize among its actions, the other agent (say the seller) will perceive a low outcome associated to that offer. For this reason the seller will switch to the first available offer below the reservation price of the buyer, and this offer will be unconditionally accepted by the seller.

Although we limit our analysis to the learning algorithm with the CoLF principle (due to its lower learning time), the same conclusions could be drawn for other RL algorithms.

6. RELATED WORKS
on the assumptions that, first, players' bounded rationality 
in [1] put forward an evolutionary argument which is based 
is captured by the representation of their strategies as fini te 
play in random pairs in incomplete information; they con-
ment learning in bargaining in combination of bounded ra-
ally more costly, e.g. because they are more difficult to learn 
repeatedly played. Nevertheless, two crucial aspects 
need prominent attention: the convergence to the optimal 
equilibrium outcome when \( \tau \) = 2. They develop an evolutionary setting that assumes 
information is partial and uncertain. In a pure game 
equilibrium is present.

Concerning future works, we intend to follows two routes. 
In a pure reinforcement learning perspective: we will develop 
and evaluate algorithms to address bargaining situations in 
which information is partial and uncertain. In a pure game 
theory perspective: we will analyze the most appropriate 
notion of equilibrium in an extensive form game in presence 
of no-perfectly rational agents such as learning agents.

8. ADDITIONAL AUTHORS

Additional authors: Marcello Restelli, email: restelli@elet.polimi.it.

9. REFERENCES