Foolproof Cooperative Learning

Alexis Jacq[1], Julien Perolat[2], Matthieu Geist[1], and Olivier Pietquin[1]

[1]Google
[2]DeepMind

June 25, 2019

Abstract

This paper extends the notion of equilibrium in game theory to learning algorithms in repeated stochastic games. We define a learning equilibrium as an algorithm used by a population of players, such that no player can individually use an alternative algorithm and increase its asymptotic score. We introduce Foolproof Cooperative Learning (FCL), an algorithm that converges to a Tit-for-Tat behavior. It allows cooperative strategies when played against itself while being not exploitable by selfish players. We prove that in repeated symmetric games, this algorithm is a learning equilibrium. We illustrate the behavior of FCL on symmetric matrix and grid games, and its robustness to selfish learners.

1 Introduction

In William Golding's novel "Lord of the Flies", a group of children who survived an airplane crash try to establish rules on a desert island in order to avoid chaos. Unfortunately, they fail at forcing a cooperative solution and some of them start defecting, which results in a demented group behaviour. In this paper, we prevent such tragedies in learning algorithms by constructing a safe way to learn cooperation in unknown environments, without being exploitable by potentially selfish agents.

In multi-agent learning settings, environments are usually modeled by stochastic games [21]. Multi-agent reinforcement learning (MARL) brings a framework to construct algorithm that aim to solve stochastic games where players individually or jointly search for an optimal decision-making to maximize a reward function. Individualist approaches mostly aim at reaching equilibrium, taking the best actions whatever the opponents behaviors are [1, 11]. Joint approaches aim at optimizing a cooperative objective and can be viewed as a single agent problem in a larger dimension [2], but are easily exploited when one agent starts being individualist.

We focus on symmetric situations, making sure that no agent has an individual advantage. For example, this is the case on a desert island with a quantity of resources equally accessible to all agents. Moreover, we consider repeated games, modelling the recurrent possibility to start again the situation from the beginning. In the island resource example, repetitions could represent successive days or, at larger scale, 4-seasons cycles.

In this context, we introduce Foolproof Cooperative Learning (FCL), a model-free learning algorithm that, by construction, converges to a Tit-for-Tat behaviour, cooperative against itself and retaliatory against selfish algorithms. We propose a definition for learning equilibrium, describing a class of learning algorithms such that the best way to play against them is to adopt the same behaviour. We demonstrate that FCL is an example of learning equilibrium that forces a cooperative behaviour, and we empirically verify this claim with two-agents matrix and grid-world repeated symmetric games.

The proofs of all stated results are provided in the appendix.

2 Definitions and Notations

An N-player stochastic game can be written as a tuple $(\mathcal{S}, (\mathcal{A}_i)_{i=1\ldots N}, \mathcal{P}, \mu_0, (r_i)_{i=1\ldots N})$, where $\mathcal{S}$ is the set of states, $\mathcal{A}_i$ the set of actions for player i, $\mathcal{P}$ the transition probability $(\mathcal{P}(\cdot|s, a_1 \ldots a_N))$, μ_0 a distribution over states $(\mu(s^0))$, r_i the reward function for player i $(r_i(s, a_1 \ldots a_N))$.

We also assume bounded, deterministic reward functions and finite state and action spaces.

In a repeated stochastic game, a stochastic game (the stage game) is played and at each iteration, it continues with probability $\gamma \in [0,1[$ or terminates and starts again according to μ_0. This is repeated an infinite number of times, and players have to maximize their average return during a stage game [15]. Terminating with probability γ is equivalent to use a discount factor while playing a stage game.

A stationary strategy (or policy) for player i, $\pi_i(\cdot|s) \in \Pi_{\mathcal{A}_i}$, maps a state to a probability distribution over its set of possible actions. We note π_{-i} the product of all players strategies but player i and $\boldsymbol{\pi} = \pi_1 \times \cdots \times \pi_N = \pi_i \times \pi_{-i}$ the product of all player strategies, called the strategy profile. Given opponents strategies π_{-i}, the goal for a rational player i is to find a strategy π_i^* that maximizes its average return $\mathcal{R}_i$ during a stage game:

$$\pi_i^* = \underset{\pi_i}{\operatorname{argmax}}\, \mathcal{R}_i(\pi_i, \pi_{-i}) = \underset{\pi_i}{\operatorname{argmax}}\, \mathbb{E}_{\pi_i,\pi_{-i}}\left[\sum_l \gamma^l r(s^l, a_i^l, a_{-i}^l) \middle| s^{l+1} \sim \mathcal{P}(.|s^l, a_i^l, a_{-i}^l)\right].$$

The policy π_i^* depends on the opponents strategies and is called the best response for player i to π_{-i}. In general, we call strategy any process $\{\pi^t\}_t$ defining a stationary strategy for any stage t. The value of a player's non-stationary strategy $\{\pi^t\}_t$ is the average return over stage games, $\mathbb{E}_{t>0}[\mathcal{R}_i(\pi_i^t, \pi_{-i}^t)]$.

In order to allow rewarding or retaliation strategies, we only consider games where all players are aware of all opponents actions and rewards, and receive a signal each time the game is reset. We also admit players to share information with some opponents in order to organize joint retaliation actions or joint explorations. Moreover, we only consider *Repeated Symmetric Games* (RSG):

Definition 1 (Repeated Symmetric game (RSG))**.** *An N-player repeated stochastic game is symmetric if, for any stationary strategy profile $(\pi_1 \dots \pi_N)$ and for any permutation ψ over players:*

$$\forall 1 \leq i \leq N,\ \mathcal{R}_{\psi(i)}(\pi_i, \pi_{-i}) = \mathcal{R}_i(\pi_{\psi(i)}, \pi_{\psi(-i)}).$$

This generalizes the definition for symmetric N-player matrix games [3] to stochastic games where players utilities are replaced by average returns[1]. In this paper, we use the concept of N-cyclic permutations to construct specific strategies:

Definition 2 (N-cyclic permutation)**.** *A permutation σ is N-cyclic if for all $i, j \in \{1 \dots N\}$, there is k such that $\sigma^k(i) = j$.*

2.1 Nash equilibrium

A Nash equilibrium describes a stationary strategy profile $\boldsymbol{\pi}^* = \pi_1^* \times \cdots \times \pi_N^*$, such that no player can individually deviate and increase its payoff [16]:

$$\forall 1 \leq i \leq N,\ \forall \pi_i \in \Pi_{\mathcal{A}_i},\ \mathcal{R}_i(\pi_i, \pi_{-i}^*) \leq \mathcal{R}_i(\pi_i^*, \pi_{-i}^*).$$

Note that in a symmetric game, for any Nash equilibrium with returns $(\mathcal{R}_1 \dots \mathcal{R}_N)$ and for any permutation σ over players, there is another Nash equilibrium with returns $(\mathcal{R}_{\sigma(1)} \dots \mathcal{R}_{\sigma(N)})$. This definition can be extended to non-stationary policies using expected return over stage games: no players can individually deviate from an equilibrium non-stationary strategy and increase its average return over stage games:

$$\forall 1 \leq i \leq N,\ \forall \{\pi_i^t \in \Pi_{\mathcal{A}_i}\}_t,\ \mathbb{E}_{t>0}[\mathcal{R}_i(\pi_i^t, \pi_{-i}^{t,*})] \leq \mathbb{E}_{t>0}[\mathcal{R}_i(\pi_i^{t,*}, \pi_{-i}^{t,*})].$$

As $\mathbb{E}_{t>0}[\mathcal{R}_i(\pi_i^t, \pi_{-i}^t)] = \mathcal{R}_i(\pi_i, \pi_{-i})$ for stationary strategy profiles, any stationary strategy equilibrium is still an equilibrium among non-stationary processes.

[1] Actually, the definition initially given in [3], $\forall i, \mathcal{R}_i(\pi_i, \pi_{-i}) = \mathcal{R}_{\psi(i)}(\pi_{\psi(i)}, \pi_{\psi(-i)})$ is incorrect in the sens that symmetries are not independent of player identities, which is not the case if the right-hand return is indexed with the inverse permutation instead [22].

3 Cooperative strategies

We call cooperative any strategy (not necessary stationary) that maximizes a common quantity $\hat{\mathcal{R}} = f(\mathcal{R}_1 \dots \mathcal{R}_N)$. Usual examples are strategies that maximize the sum, the product or the minimum of players returns. In RSGs, the strategy that maximizes the minimum of player returns is particularly interesting as it coincides with Kalais [6] solutions to the Bargaining problem [17] and is easy to determine. In this paper, we refer to this strategy as the *bestmin* solution. An important property of RSGs is the fact that *bestmin* solutions can always be obtained by repeatedly applying a N-cyclic permutation on a stationary strategy that maximizes the sum of players returns.

Theorem 1. *Let π^{Σ} be a stationary strategy that maximizes the sum of players returns in an N-player RSG, σ an N-cyclic permutation over players, and t indexing the repeated stage games. Then, the strategy $\pi^t = (\pi^{\Sigma}_{\sigma^t(1)} \dots \pi^{\Sigma}_{\sigma^t(N)})$ (where $\sigma^t = \sigma \circ \dots \circ \sigma$ t times) is a bestmin strategy.*

3.1 Tit-for-Tat

Given a stochastic game, one player i can learn a strategy $\pi_i^{r,j}$ that retaliates when another player j deviates from a target strategy. If a retaliation is smaller than the reward obtained by the player while deviating, the strategy can be repeated until the retaliation is larger than this reward in total. In that case, the target strategy is said enforceable: if all player are accorded to retaliate when a player deviates from a strategy profile and if the retaliation is strong enough, no player can improve its payoff by individually deviating from the strategy profile. If opponents actions are part of the observable state and if the target strategy profile and the dynamics are deterministic, it becomes possible to construct a stationary strategy that retaliates when a player does not play according to the profile. If the retaliation lasts forever after the first deviation, the strategy is by construction a Nash equilibrium [18]. However, we are more interested in finished retaliations since it gives a chance to a selfish learning agent to learn the target strategy. Such processes are called Tit-for-Tat (TFT) and are known to induce cooperation in repeated social dilemma. In an RSG, if the target is a *bestmin* strategy, there is always a stationary way to retaliate and one can always construct a TFT strategy.

Theorem 2. *In an RSG, let $\boldsymbol{\pi}^{r,j} = \operatorname{argmin}_{\pi_{-j}} \operatorname{argmax}_{\pi_j} \mathcal{R}_j(\pi_j, \pi_{-j}) = \pi_j^{r,j} \times \pi_{-j}^{r,j}$, and $\boldsymbol{\pi}^*$ a bestmin strategy (not necessary stationary). Then, $\boldsymbol{\pi}^{r,j}$ is a retaliation strategy with respect to $\boldsymbol{\pi}^*$:*

$$\forall 1 \leq j \leq N, \forall \pi_j \in \Pi_{A_j}, \mathcal{R}_j(\pi_j, \pi_{-j}^{r,j}) \leq \mathbb{E}_{t \geq 0}[\mathcal{R}_j(\pi_j^{*,t}, \pi_{-j}^{*,t})].$$

For a player j, we note $V_j^c = \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi_j^{*,t}, \pi_{-j}^{*,t})]$ its average return when all player cooperate, $V_j^r = \mathcal{R}_j(\boldsymbol{\pi}^{r,j})$ its best average return when others retaliate and $V_j^d = \max_{\pi_j} \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi_j, \pi_{-j}^{*,t})]$ its best average return by defecting. When a single retaliation is too small so it still worth defecting for a selfish player, the retaliation must be repeated. The minimal number or retaliation repeats can be given by (see the proof of Thm. 3 in appendix):

$$K_j = \left\lceil \frac{V_j^d - V_j^c}{V_j^c - V_j^r} \right\rceil. \tag{1}$$

In the edge case where $V_j^c = V_j^r$, the retaliation strategy must be employed endless, but the cooperative objective is not affected (this is the case in Rock–paper–scissors). Let $\{\boldsymbol{\pi}^t_{\text{TFT}}\}_t$ be the (non-stationary) strategy that follows $\boldsymbol{\pi}^*$ if all players cooperate, or repeat $\boldsymbol{\pi}^{r,j}$ over K_j stage games if a player j deviates from $\boldsymbol{\pi}^*$. By construction, $\{\boldsymbol{\pi}^t_{\text{TFT}}\}_t$ is a Nash equilibrium.

Theorem 3. *$\{\boldsymbol{\pi}^t_{TFT}\}_t$ is a Nash equilibrium.*

4 Learning algorithm

In this work, we mean by *learning algorithm* played by i any random process $A_i = \{\pi_i^T\}_T$ conditioned, at any time $T > 0$, by the historic $\mathcal{H}^T = \{s^t, a_i^t, a_{-i}^t, r_i^t, r_{-i}^t\}_{t<T}$ of all states, actions and rewards up to time $T-1$. The algorithm profile $\boldsymbol{A} = (A_1 \dots A_N)$ is the set of all players algorithms. We will note $A_i(t) = \pi_i^t$.

4.1 Multi-agent learning

Reinforcement learning provides a class of algorithms that aim at maximizing an agent's return. Out of all of them, our interest concerns Q-learning approaches [23] for three reasons: they are model-free, off-policy and they are guaranteed to converge in finite state and action spaces. In a game $\mathcal{G}$, for a player i and given opponents policy π_{-i}, the basic idea is to learn a Q-function that approximates, for all states and actions, the average return starting from playing this action at this point while using the best strategy. Ideally, the Q-function Q_i associated with player i's policy that maximizes its return holds:

$$Q_i(s, a_i, a_{-i}) = r_i(s, a_i, a_{-i}) + \gamma \sum_{s'} \mathcal{P}(s'|s, a_i, a_{-i}) \max_{a'_i} \sum_{a'_{-i}} \pi_{-i}(a_{-i}|s') Q_i(s', a'_i, a'_{-i}).$$

Q-learning algorithms are constructed in order to progressively approximate the Q-function without approximating the problem dynamics $\mathcal{P}$ and reward functions r, and without knowing the decision process that generated the historic buffer (in contrast, for example, to policy gradient algorithms [24]). In finite states and actions spaces, the approximation is obtained by successively applying the updates:

$$\begin{aligned} Q_i^{t+1}(s^t, a_i^t, a_{-i}^t) &= Q_i^t(s^t, a_i^t, a_{-i}^t) \\ &+ \alpha_t \left(r_i^t + \gamma \max_{a_i} \sum_{a_{-i}} \pi_{-i}(a_{-i}|s^{t+1}) Q_i(s^{t+1}, a_i, a_{-i}) - Q_i(s^t, a_i^t, a_{-i}^t) \right), \end{aligned}$$

where α_t is the learning rate. However, when the opponent policy is not fixed, maximizing the Q-function with respect to actions is no longer an improvement of the policy (the response of the opponents to this deterministic policy can decrease the average player's return). MARL provides several alternative greedy improvements. For example, a defensive player can expect opponents to minimize its Q-function (*minimax* Q-learning). In that case, a greedy improvement of the policy to evaluate the value of a new state is obtained by solving the linear problem [10]:

$$\pi_i^{greedy}(.|s) = \operatorname*{argmax}_{\pi_i} \min_{a_{-i}} \sum_{a_i} \pi_i(a_i|s) Q_i(s, a_i, a_{-i}), \tag{2}$$

and the corresponding Q-learning update becomes:

$$\begin{aligned} Q_i^{t+1}(s^t, a_i^t, a_{-i}^t) &= Q_i^t(s^t, a_i^t, a_{-i}^t) \\ &+ \alpha_t \left(r_i^t + \gamma \max_{\pi_i} \min_{a_{-i}} \sum_{a_i} \pi_i(a_i|s^{t+1}) Q_i(s^{t+1}, a_i, a_{-i}) - Q_i(s^t, a_i^t, a_{-i}^t) \right). \end{aligned}$$

4.2 Learning equilibrium

We define a learning equilibrium as follows.

Definition 3 (Learning equilibrium)**.** *Let $\mathcal{G}$ be a set of stochastic games. An algorithm profile $\boldsymbol{A}^* = (A_1^* \dots A_N^*)$ is a learning equilibrium for $\mathcal{G}$ if, for any game $g \in \mathcal{G}$, there is a time T_g such that, for any player i and any learning algorithm A_i:*

$$\mathbb{E}_{t>T_g}\left[\mathcal{R}_i\Big(A_i(t), A_{-i}^*(t)\Big)\right] \leq \mathbb{E}_{t>T_g}\left[\mathcal{R}_i\Big(A_i^*(t), A_{-i}^*(t)\Big)\right]$$

Consequently, just like Nash equilibrium for the choice of a strategy, no player can individually follow an alternative algorithm and increase its asymptotic score. However, one important difference is the fact that a learning algorithm is not defined with respect to a particular game, but a set of games.

We may think that a process always playing a Nash equilibrium of the given game ($\pi_i^t = \pi_i^*$ for all t) is a learning equilibrium. However, such a process requires an initial knowledge about the dynamics and the reward functions of the game and can't be obtained from a process starting with an empty condition. Therefore, it can't be described as a learning algorithm. For the same reason, a TFT process is not a learning equilibrium. However, we may construct learning algorithms that asymptotically behave as a TFT or always play a Nash equilibrium. This is the key idea of FCL.

5 Foolproof cooperative learning

As we are interested in forced cooperation, we are looking for a learning algorithm profile that converges to a TFT process, retaliating if a player deviates from a cooperative strategy. Since the objective of a cooperative strategy is a common quantity and TFT processes are symmetric, such a convergence can be obtained if all players are using the same algorithm. FCL, as described in Alg. 1 (for a player i), has the property to converge to such a behavior when played by all players. In an N-player game, FCL approximates $2N+1$ Q-functions: one associated with the cooperative policy that maximizes the sum of all players (Q^c), N associated with retaliation policies preventing any defection from other players j (Q^r_j), and N associated with each opponent's best response to the cooperative strategy (Q^d_j). At each played stage game, FCL will play according to a *bestmin* cooperative strategy (learned through Q^c) unless one of the opponents deviated from that strategy. In case of an opponent's defection, all FCL agents will agree on a joint retaliation according to the *minimax* strategy (learned through Q^r_j with Eq.(2)) for K stages according to Eq.(1). In order to allow exploration, a deterministic process $\phi(t)$ is used to decide, at each time t, between exploration and exploitation. We design ϕ as a known realization of a random process such that explorations are endless ($\forall T\,,\, \exists t > T, \mathbb{P}[\phi(t) = \textbf{True}] > 0$), but becomes rare enough with time so the probability of explorations tends to zero ($\forall \epsilon > 0\,,\, \exists T_\epsilon\,,\, \forall t > T_\epsilon, \mathbb{P}[\phi(t) = \textbf{True}] < \epsilon$). This can be implemented using a pseudo-random process with a fixed seed, known by all FCL players. At exploration stages, all agents are allowed to perform any action without being accused of defection. In a way, this algorithm can be seen as a disentangled version of Friend-or-Foe Q-learning (FFQ) [11] which learns to play cooperatively if an opponent is cooperative, or defensively if the opponent is defective with a single Q-function. However, FFQ can't learn a TFT behavior as it is either always cooperative, or always defensive.

Theorem 4. *Assume $\mathcal{S}$ and $\mathcal{A}_i$ are finite spaces. Then, FCL converges to a TFT behavior forcing the bestmin cooperative strategy in RSGs.*

Theorem 5. *FCL is a learning equilibrium for RSGs.*

6 Experiments

Despite our theoretical claims are established for any number of agents, we restrict our experiments to games involving two players. We first explore the case of three well known repeated symmetric matrix games: Iterated Prisoners Dilemma (IPD), Iterated Chicken (ICH) and Rock-Paper-Scissors (RPC). Then, we investigate larger state spaces with grid games inducing coordination problems and social dilemma, as introduced in [15]. We added another grid game, closer to the concept of limited resource appropriation: the Temptation game. In Temptation, making a movement to the sides can be seen as taking immediately the resource, while making a movement to the bottom can be seen as waiting for the winter. All grid games are described in details in Table 1. In order to verify that FCL is a learning equilibrium, we compare the score obtained by FCL and by selfish learning algorithm, Q-learning and policy-gradient (PG), against FCL.

6.1 Implementation details

We implemented FCL using a state-dependent learning rate $\alpha_t = (\sum_{l<t} \delta\{s^l = s^t\})^{-1}$ that counts the number of state visits, and exploration $\phi(t) = \{X_t > \epsilon d^t\}$ where X_t is a pseudo-random uniform sample between 0 and 1 with a fixed seed, ϵ the initial treshold and d a decay parameter close to one. The closer is d to one, the longer lasts the exploration. For selfish Q-learning, we used a similar learning rate and exploration process, however with different seeds and decay parameters. The policy gradient was implemented with tabular parameters and Adam gradient descent with learning rate 0.1. Since matrix games are not sequential and since grid games were automatically reset after 30 steps, we could use a discount factor $\gamma = 1$ to estimate value functions. In practice, we found that adding 1 to the minimal number of retaliation repeats given in Eq. 1 significantly improves the robustness to selfish learners.

6.2 Results

An iterated matrix game can be seen as a repeated stochastic game with only one state. As it does not require large exploration, we used $\epsilon = 0.5$ and $d = 0.9$ for both selfish Q-learning and FCL.

Algorithm 1 FCL for player i.

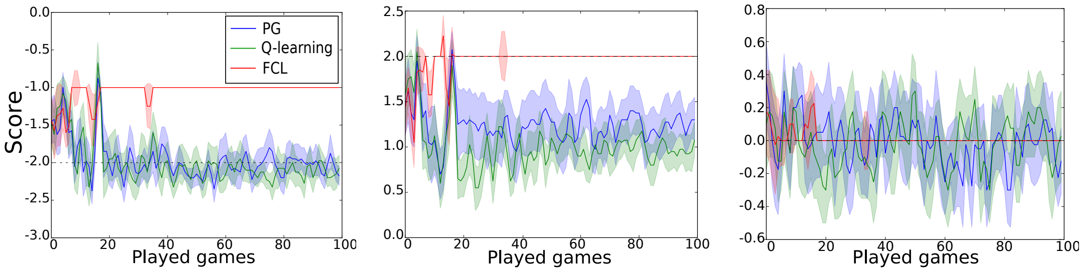
```
input List of counters $k_j = 0\ \forall j$ to repeat retaliations, exploration process $\phi(s,t)$, N-cyclic permutation $\sigma$, learning rate sequence $\{\alpha_t\}_t$, initial (arbitrary) functions $Q^c$, $\{Q_i^d\}_{i=1...N}$ and $\{Q_i^r\}_{i=1...N}$, initial state $s$.
for stages $t = 1$ to $+\infty$ do
  while stage continue do
    if $K_j = 0 \forall j$ then
      if $\phi(t)$ then
        Explore $a_i \sim \mathcal{U}(\mathcal{A}_i)$ with uniform probability
      else
        Take action $a_i = \text{argmax}_{a_{\sigma^t(i)}} \max_{a_{-\sigma^t(i)}} Q^c(s, a_i, a_{-i})$
      end if
    else
      Randomly select an agent $j$ such that $K_j > 0$
      Take action $a_i \sim \text{argmin}_{\pi_{-j}} \max_{a_j} \sum_{a_{-j}} \pi_{-j}(a'_{-j}|s) Q_j^r(s, a_j, a_{-j})$
      $k_j \leftarrow k_j - 1$
    end if
    Observe $a_{-i}$ and new state $s'$, receive reward $r_i = r_i(s, a_i, a_{-i})$ and observe $r_{-i}$
    $Q^{c\prime} \leftarrow \max_{a'_i} \max_{a'_{-i}} Q^c(s', a'_i, a'_{-i})$
    $Q^c(s, a_i, a_{-i}) \mathrel{+}= \alpha_t \left( \sum_{1 \le j \le N} r_j + \gamma Q^{c\prime} - Q^c(s, a_i, a_{-i}) \right)$
    for all other agents $j \neq i$ do
      $V_j^r(s') \leftarrow \min_{\pi_{-j}} \max_{a'_j} \sum_{a'_{-j}} \pi_{-j}(a'_{-j}|s') Q_j^r(s', a'_j, a'_{-j})$
      $V_j^d(s') \leftarrow \max_{a'_j} Q_j^r(s', a'_j, \text{argmax}_{a_{-j}} \max_{a'_j} Q^c(s', a_j, a_{-j}))$
      $Q_j^r(s, a_j, a_{-j}) \mathrel{+}= \alpha_t \left( r_j + \gamma V_j^r(s') - Q_j^r(s, a_j, a_{-j}) \right)$
      $Q_j^d(s, a_j, a_{-j}) \mathrel{+}= \alpha_t \left( r_j + \gamma V_j^d(s') - Q_j^d(s, a_j, a_{-j}) \right)$
      $K_j \leftarrow \left\lceil \frac{V_j^d(s') - V^c(s')}{V^c(s') - V_j^r(s')} \right\rceil$
      if not $\phi(t)$ and $a_j \neq \text{argmax}_{a_j} Q_j^c(s)$ then
        $k_j \leftarrow k_j + K_j$
      end if
    end for
    $s \leftarrow s'$
  end while
end for
```

Figure 1 displays our results with the three matrix games IPD, ICH and RPC. In grid games, we used $\epsilon = 1$ and $d = 0.995$. Figure 2 displays our results on grid games. As expected, FCL was never exploited by selfish learners, and successfully cooperated with another FCL. Except in RCP, defection conduced to less reward than cooperation because of retaliations. In RCP, FCL found the only way to retaliate by infinitely playing randomly against selfish learners, resulting in an average of 0 reward for all players, equivalent to the reward for cooperation.

7 Related work

Learning cooperative behaviours in a multi-agent setting is a vast field of research, and various approaches depend on assumptions about the type of games, the type and number of agents, the type of cooperation and the initial knowledge.

When the game's dynamic is initially known and in two-player settings, Kalais' bargaining solution can be obtained by mixing dynamic and linear programming. Therefore, a polynomial-time algorithm can be used to solve repeated matrix games [12], as well as repeated stochastic games [15]. Since a bargaining solution is always better than a *minimax* strategy (the disagreement point) [18], a cooperative equilibrium is immediately given. An alternative to our cooperate or



(a) Iterated prisoners dilemma (b) Iterated chicken (c) Rock paper scissors

Figure 1: Matrix games. Average scores over 20 runs obtained by two standard RL algorithms and FCL, playing against FCL. In IPD and ICH, after some iterations selfish behaviours, as induced by Q-learning and PG, start being sub-optimal because of FCL retaliations and accumulate less return than a cooperative behaviours, as induced by FCL against itself. In RPC, FCL learns to play with a uniform distribution against selfish algorithms so their average score is null. Black dotted line represents the average score after convergence of two selfish agent playing against themselves (the *minimax* solution).

Table 1: Grid games. A is the starting position of one player, B is the starting position of the other. At each turn, both players simultaneously select one action among going up, down, left, right or stay. When reward cells with \$ symbol are reached by one player, the player obtains the corresponding reward and the game is immediately reset. $\$_{A:X}$ means that only player A gets the reward X when reaching the cell, $\$_X$ means that any player gets reward X when reaching the cell, and $\$_{X,Y}$ means that the player who reach the cell gets X and the other gets Y (if the other player reach another rewarding cell, the rewards are summed). Two players can not be on the same cell at the same time and they can not cross each other. In case of conflict, one player reaches the cell and the other stays with probability 0.5.

(a) Grid prisoners dilemma

▨	▨	▨	▨	$\$_{100}$	▨	▨	▨	▨
$\$_{A:100}$			A		B			$\$_{B:100}$

(b) Compromise

	▨	$\$_{B:100}$	▨		▨	$\$_{A:100}$	▨		
		A				B			

(c) Coordination

$\$_{B:100}$		$\$_{A:100}$
A		B

(d) Temptation

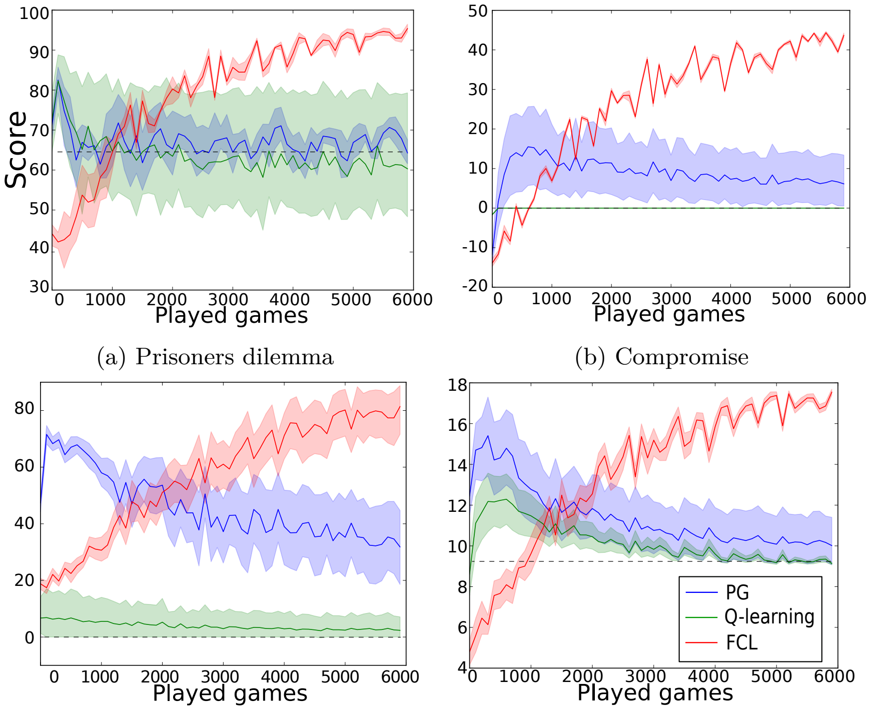
$\$_{20,-10}$	A	B	$\$_{20,-10}$
$\$_{40,-20}$			$\$_{40,-20}$

retaliate architecture consists in choosing between maximizing oneself reward (being competitive) or maximizing a cooperative reward, for example by inferring opponents intentions [7].

In games inducing social dilemmas and when the dynamic is accessible as an oracle, cooperative solutions can also be obtained by self-playing and then applied to define a TFT behaviour forcing cooperation [8], even when opponent actions are unknown, since in that case the reward function already brings sufficient information [20].

Closer to our setting, when the dynamic is unknown, online MARL can extract cooperative solution in some non-cooperative games, and particularly in restricted resource appropriation [19]. Using alternative objectives based on all players reward functions and their propensity to cooperate or defect improves and generalizes the emergence of cooperation in non-cooperative games and limits the risk of being exploited by purely selfish agents [5].

A similar approach, called Learning with Opponent Learning Awareness (LOLA), consists in modelling the strategies and the learning dynamics of opponents as part of the environment's dynamics and to derive the gradient of the average return's expectation [4]. If LOLA has no guaranty of convergence, a recent improvement of the gradient computation, which interpolates between first and second-order derivations, is proved to converge to local optimums [9]. Although such agents are purely selfish, empirical results show that they are able to shape each others learning trajectories and to cooperate in prisoners dilemma. A limitation of this approach toward building learning equilibrium is the strong assumption regarding the opponents learning algorithms, supposed to perform policy gradient. Also, this approach differs to our goal since LOLA is selfish



(a) Prisoners dilemma

(b) Compromise

(c) Coordination

(d) Temptation

Figure 2: Grid games. Average scores over 20 runs obtained by two standard RL algorithms and FCL, playing against FCL. After some iterations, selfish behaviours, as induced by Q-learning and PG, start being sub-optimal because of FCL retaliations and accumulate less return than a cooperative behaviour, as induced by FCL against itself. Black dotted line represents the average score after convergence of two selfish agents playing against themselves (the *minimax* solution).

and aims at shaping an opponent behavior (in 2-player settings) while FCL is cooperative but retaliates in response to selfish agents (in N-player settings).

8 Conclusion

We introduced FCL, a model-free learning algorithm that, by construction, converges to a TFT behaviour, cooperative against itself and retaliating against selfish algorithms. We proposed a definition for learning equilibrium, describing a class of learning algorithms such that the best way to play against it is to adopt the same behaviour. We demonstrated that FCL is an example of learning equilibrium that forces a cooperative behaviour, and we empirically verified this claim with two-agents matrix and grid-world repeated symmetric games.

Our approach could be improved by facilitating opponent's learning of the optimal cooperative response and by using faster learning approaches. It could also be adapted to larger dimensions such as continuous state spaces and partially observed settings with function approximation by replacing tabular Q-learning with deep Q-learning [14].

References

[1] Michael Bowling and Manuela Veloso. Rational and convergent learning in stochastic games. *Proceedings of the International joint conference on artificial intelligence*, 2001.

[2] Caroline Claus and Craig Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. *Proceedings of the Association for the Advancement of Artificial Intelligence*, 1998.

[3] Partha Dasgupta, Eric Maskin, et al. The existence of equilibrium in discontinuous economic games. *Review of Economic Studies*, 1986.

[4] Jakob Foerster, Richard Y Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. *Proceedings of the International Conference on Autonomous Agents and MultiAgent Systems*, 2018.

[5] Edward Hughes, Joel Z Leibo, Matthew Phillips, Karl Tuyls, Edgar Dueñez-Guzman, Antonio García Castañeda, Iain Dunning, Tina Zhu, Kevin McKee, Raphael Koster, et al. Inequity aversion improves cooperation in intertemporal social dilemmas. *Advances in Neural Information Processing Systems*, 2018.

[6] Ehud Kalai. Proportional solutions to bargaining situations: interpersonal utility comparisons. *Econometrica: Journal of the Econometric Society*, 1977.

[7] Max Kleiman-Weiner, Mark K Ho, Joseph L Austerweil, Michael L Littman, and Joshua B Tenenbaum. Coordinate to cooperate or compete: abstract goals and joint intentions in social interaction. *Proceedings of Annual Conference of the Cognitive Science Society*, 2016.

[8] Adam Lerer and Alexander Peysakhovich. Maintaining cooperation in complex social dilemmas using deep reinforcement learning. *arXiv preprint arXiv:1707.01068*, 2017.

[9] Alistair Letcher, Jakob Foerster, David Balduzzi, Tim Rocktäschel, and Shimon Whiteson. Stable opponent shaping in differentiable games. *Proceedings of the International Conference on Learning Representations*, 2018.

[10] Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. *Proceedings of the International Conference on Machine Learning*, 1994.

[11] Michael L Littman. Friend-or-foe q-learning in general-sum games. *Proceeding of the International Conference on Machine Learning*, 2001.

[12] Michael L Littman and Peter Stone. A polynomial-time nash equilibrium algorithm for repeated games. *Decision Support Systems*, 2005.

[13] Francisco S Melo. Convergence of q-learning: A simple proof. *Institute Of Systems and Robotics, Tech. Rep*, 2001.

[14] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 2015.

[15] Enrique Munoz de Cote and Michael L. Littman. A polynomial-time Nash equilibrium algorithm for repeated stochastic games. *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2008.

[16] John Nash. Non-cooperative games. *Annals of mathematics*, 1951.

[17] John F Nash Jr. The bargaining problem. *Econometrica: Journal of the Econometric Society*, 1950.

[18] Martin J Osborne and Ariel Rubinstein. *A course in game theory.* MIT press, 1994.

[19] Julien Pérolat, Joel Z Leibo, Vinicius Zambaldi, Charles Beattie, Karl Tuyls, and Thore Graepel. A multi-agent reinforcement learning model of common-pool resource appropriation. *Advances in Neural Information Processing Systems*, 2017.

[20] Alexander Peysakhovich and Adam Lerer. Consequentialist conditional cooperation in social dilemmas with imperfect information. *Proceedings of the International Conference on Learning Representations*, 2018.

[21] Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 1953.

[22] Steen Vester. *Symmetric Nash Equilibria.* PhD thesis, Master thesis from Ecole Normale Superieure de Cachan, 2012.

[23] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine Learning*, 1992.

[24] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992.

9 Appendix

9.1 Proof of Thm. 1

Proof. Since σ is N-cyclic, any player i receives the same average return every N stage games:

$$\begin{aligned}\mathbb{E}_{t\geq 0}\left[\mathcal{R}_i(\pi^t)\right] &= \frac{1}{N}\sum_{t=1}^{N}\mathcal{R}_i(\pi^{\Sigma}_{\sigma^t(i)}, \pi^{\Sigma}_{\sigma^t(-i)}) \textit{ (the strategy is stationary)} \\ &= \frac{1}{N}\sum_{t=1}^{N}\mathcal{R}_{\sigma^t(i)}(\pi^{\Sigma}_{i}, \pi^{\Sigma}_{-i}) \textit{ (the game is symmetric)} \\ &= \frac{1}{N}\sum_{t=1}^{N}\mathcal{R}_{t}(\pi^{\Sigma}_{i}, \pi^{\Sigma}_{-i}) \textit{ (changing the order).}\end{aligned}$$

Consequently, π_t maximizes the sum of returns at any t and the average return of the strategy is the same for all players. Now, imagine there is a strategy $\{\hat{\pi}^t\}_t$ such that:

$$\min_i \mathbb{E}_{t\geq 0}\left[\mathcal{R}_i(\hat{\pi}^t_i, \hat{\pi}^t_{-i})\right] > \min_i \mathbb{E}_{t\geq 0}\left[\mathcal{R}_i(\pi^t_i, \pi^t_{-i})\right].$$

In that case,

$$\sum_i \mathbb{E}_{t\geq 0}\left[\mathcal{R}_i(\hat{\pi}^t_i, \hat{\pi}^t_{-i})\right] > N\min_i \mathbb{E}_{t\geq 0}\left[\mathcal{R}_i(\pi^t_i, \pi^t_{-i})\right] = \sum_i \mathbb{E}_{t\geq 0}\left[\mathcal{R}_i(\pi^t_i, \pi^t_{-i})\right],$$

which is in contradiction with the fact that π_t maximizes the sum of returns at any t. □

9.2 Proof of Thm. 2

Proof. Assume that:

$$\exists \pi_j, \mathcal{R}_j(\pi_j, \pi^{r,j}_{-j}) > \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi^{*,t}_j, \pi^{*,t}_{-j})].$$

Then, the same is true in particular for j's best response to $\pi^{r,j}_{-j}$, that we note $\pi^{r,j}_j$:

$$\mathcal{R}_j(\pi^{r,j}_j, \pi^{r,j}_{-j}) > \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi^{*,t}_j, \pi^{*,t}_{-j})].$$

Since in that case, $\pi^{r,j}_{-j}$ minimizes j's return:

$$\forall \pi_{-j}, \mathcal{R}_j(\pi^{r,j}_j, \pi_{-j}) \geq \mathcal{R}_j(\pi^{r,j}_j, \pi^{r,j}_{-j}) > \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi^{*,t}_j, \pi^{*,t}_{-j})]$$

In particular, we have

$$\mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi^{r,j}_j, \pi^{*,t}_{-j})] > \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi^{*,t}_j, \pi^{*,t}_{-j})]. \tag{3}$$

Besides, we have:

$$\exists i, \mathbb{E}_{t\geq 0}[\mathcal{R}_i(\pi^{r,j}_j, \pi^{*,t}_{-j})] \leq \min_k \mathbb{E}_{t\geq 0}[\mathcal{R}_k(\pi^{*,t}_j, \pi^{*,t}_{-j})] \tag{4}$$

Indeed, if this was not the case, we would have

$$\forall i, \mathbb{E}_{t\geq 0}[\mathcal{R}_i(\pi^{r,j}_j, \pi^{*,t}_{-j})] > \min_k \mathbb{E}_{t\geq 0}[\mathcal{R}_k(\pi^{*,t}_j, \pi^{*,t}_{-j})]$$

and $\boldsymbol{\pi}^*$ is no longer a *bestmin*.

On the other hand, since the game is symmetric, one can apply the transposition that permutes players i and j strategies in Eq. (3):

$$\mathbb{E}_{t\geq 0}[\mathcal{R}_i(\pi^{r,j}_i, \pi^{*,t}_{-i})] > \mathbb{E}_{t\geq 0}[\mathcal{R}_k(\pi^{*,t}_j, \pi^{*,t}_{-j})] \geq \min_k \mathbb{E}_{t\geq 0}[\mathcal{R}_k(\pi^{*,t}_j, \pi^{*,t}_{-j})],$$

which is in contradiction with Eq.(4). □

9.3 Proof of Thm. 3

Proof. Since $K_j \geq \frac{V_j^d - V_j^c}{V_j^c - V_j^r}$ and $V_j^c \geq V_j^r$, we write:

$$K_j(V_j^c - V_j^r) \geq V_j^d - V_j^c k,$$

which gives:

$$V_j^c \geq \frac{1}{K_j + 1}(V_j^d + K_j V_j^r).$$

On the left, this is the average return over stages of an always cooperating player, on the right this is the average return over stages of any deviating player. Therefore, for any $\{\pi_j^t\}_t \neq \{\boldsymbol{\pi}^t_{\text{TFT}}\}_t$:

$$\mathbb{E}_{t\geq 0}[\mathcal{R}_j(\boldsymbol{\pi}^t_{\text{TFT}})] \geq \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi^t_j, \pi^t_{\text{TFT}_{-j}})].$$

□

9.4 Proof of Thm. 4

Proof. We assume that all agents are playing FCL. Given the fact that ϕ is deterministic, there are no defection. Let's focus on $\{\tilde{\pi}^t\}_t = \{\pi^t | \phi(t) = \textbf{True}\}$ the endless sub-process corresponding to exploration times. Let $\mathbb{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_N$. Clearly, for all t, $s \in \mathcal{S}$ and $a \in \mathbb{A}$, $\tilde{\pi}^t(a|s) > 0$. Consequently, the convergence of Q-functions is given by the convergence of the classic Q-learning using $\pi(a^t = a | s^t = s) = \tilde{\pi}^t(a|s)$ [13]. Let Q^{c*} and Q_j^{r*} be the corresponding points of convergence. By construction:

- $\boldsymbol{\pi^c}$, maximizing Q^{c*}, maximizes the sum of players returns.
- $\boldsymbol{\pi^*} = (\pi^c_{\sigma^t(1)} \dots \pi^c_{\sigma^t(N)})$ maximizes the min of players returns.
- $\boldsymbol{\pi^r}(j)$, minimizing Q_j^{r*}, retaliates when player j deviates from $\boldsymbol{\pi^*}$.

FCL decision rule at line 7 in Alg.1 corresponds to playing according to $\boldsymbol{\pi^*}$ when Q^c is close enough to Q^{c*} (when the difference between the max value and the second-max value is larger than twice an update size). Similarly, decision rule at line 11 corresponds to playing according to $\boldsymbol{\pi^r}(j)$ when Q_j^r is close enough to Q_j^{r*}. Let T_Q be the smallest time after which both Q^c and Q_j^r are close enough to Q^{c*} and Q_j^{r*} so lines 7 and 11 correspond to playing according to $\boldsymbol{\pi^*}$ and $\boldsymbol{\pi^r}(j)$. If explorations are stopped, FCL is a TFT strategy. Actually explorations never stop but the probability[2] of exploration times tends to zero, which translates to

$$\forall \epsilon > 0\,, \forall t > \max\{T_\epsilon, T_Q\}\,, \mathbb{P}\bigg[A_1(t) = \pi^t_{\text{TFT}}, \dots, A_N(t) = \pi^t_{\text{TFT}}\bigg] > 1 - \epsilon,$$

where $\{\pi^t_{\text{TFT}}\}_t$ is any player's TFT strategy induced by $\boldsymbol{\pi^*}$ and $\boldsymbol{\pi^r}$ as described in Thm. 3. □

9.5 Proof of Thm. 5

Proof. Let $A_j = \{\pi_j^t\}_t$ be a learning algorithm different that FCL and played by an agent j while all other players do FCL ($A_{-j} = \{\pi^t_{-j}\}_t$). We will distinguish two situations:

$$\begin{aligned}
&\text{(a) } \forall s \in \mathcal{S}\,, \forall a_j \in \mathcal{A}_j\,, \forall a_{-j} \in \mathcal{A}_{-j}\,, \forall T\,, \exists t > T, \pi_j^t(a_j|s) \times \pi^t_{-j}(a_{-j}|s) > 0,\\
&\text{(b) } \exists s \in \mathcal{S}\,, \exists a_j \in \mathcal{A}_j\,, \forall a_{-j} \in \mathcal{A}_{-j}\,, \exists T\,, \forall t > T, \pi_j^t(a_j|s) \times \pi^t_{-j}(a_{-j}|s) = 0.
\end{aligned}$$

In situation (a), conditions for the convergence of FCL agents are met and they converge to a TFT behavior:

$$\forall \epsilon > 0, \exists T_\epsilon, \forall t > T_\epsilon, \mathbb{P}\bigg[A_1(t) = \pi^t_{\text{TFT}_1}, \dots, A_N(t) = \pi^t_{\text{TFT}_N}\bigg] > 1 - \epsilon.$$

Let $\mathcal{R}_1 = \mathbb{E}_{t > T_\epsilon}[\mathcal{R}_j(A_j, A^*_{-j})]$ the deviating player's average return when other agents are doing TFT (with probability $1-\epsilon$), R_2 its average return when other agents are exploring during stages (whith probability ϵ and $\mathcal{R}_1^*$ and $\mathcal{R}_2^*$ the respective returns when no player deviates). Because of

[2] Actually, ϕ is deterministic but is given by the realization of a random process verifying this property. One can also considerate that all FCL players are observing the same random process telling them to explore or not.

Thm. 3, we know that $\mathcal{R}_1 \leq \mathcal{R}_1^*$. If $\mathcal{R}_2 \leq \mathcal{R}_2^*$, then the average return of a deviating player is always smaller than if it does not deviate. Otherwise, we have $\mathcal{R}_2 + \mathcal{R}_1^* > \mathcal{R}_2^* + \mathcal{R}_1$ and by taking:

$$\epsilon = \frac{\mathcal{R}_1^* - \mathcal{R}_1}{\mathcal{R}_2 + \mathcal{R}_1^* - (\mathcal{R}_2^* + \mathcal{R}_1)}$$

we obtain, for all $t > T_\epsilon$:

$$(1-\epsilon)\mathcal{R}_1 + \epsilon\mathcal{R}_2 \leq (1-\epsilon)\mathcal{R}_1^* + \epsilon\mathcal{R}_2^*.$$

Consequently:

$$\mathbb{E}_{t>T_\epsilon}\left[\mathcal{R}_j\left(A_j(t), A_{-j}^*(t)\right)\right] \leq \mathbb{E}_{t>T_\epsilon}\left[\mathcal{R}_j\left(A_j^*(t), A_{-j}^*(t)\right)\right].$$

In situation (b), there is still a subset of state-action couples Ω_∞ that will be explored an infinite number of times. If all other players restrict their states and actions to the same subset (using $\pi_i(a|s) > 0 \Leftrightarrow (a,s) \in \Omega_\infty$) the induced sub-game is still symmetric and player j is exploring the whole sub-game an infinite number of times. Consequently, FCL can at least learn a TFT strategy $\{\hat{\boldsymbol{\pi}}^t_{\mathrm{TFT}}\}_t$ based on a retaliation strategy $\boldsymbol{\pi}^{r,j}_{\Omega_\infty}$ and a cooperative strategy $\boldsymbol{\pi}^*_{\Omega_\infty}$ defined on Ω_∞ such that:

$$\forall \{\pi_j^t\}_t \neq \{\hat{\boldsymbol{\pi}}^t_{\mathrm{TFT}_t}\}, \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\hat{\boldsymbol{\pi}}^t_{\mathrm{TFT}})] = \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\boldsymbol{\pi}^*_{\Omega_\infty})] \geq \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi_j^t, \hat{\boldsymbol{\pi}}^t_{\mathrm{TFT}_{-j}})].$$

Since $\boldsymbol{\pi}^*_{\Omega_\infty}$ is necessarily sub-optimal to cooperate in the whole game, we have:

$$\forall \{\pi_j^t\}_t \neq \{\hat{\boldsymbol{\pi}}^t_{\mathrm{TFT}_t}\}, \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\boldsymbol{\pi}^*)] \geq \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\boldsymbol{\pi}^*_{\Omega_\infty})] \geq \mathbb{E}_{t\geq 0}[\mathcal{R}_j(\pi_j^t, \hat{\boldsymbol{\pi}}^t_{\mathrm{TFT}_{-j}})].$$

As a consequence, players can still retaliate and we can use the exact same argument than in (a) to obtain the desired statement. □