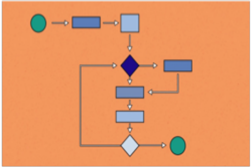
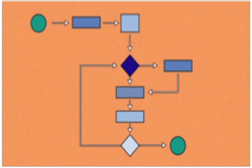
Self-Play Compatibility in Multi-Agent Learning

Ethan Zell and Anthony DiGiovanni

Introduction: The Self-Play Problem



VS



- Multi-agent sequential decisions
- Standard RL: model (maybe implicit) $\rightarrow$ gather data $\rightarrow$ optimize
- What if multiple users adopt your algorithm?

Repeated Games

	1	**2**
1	10, 10	0, 9
2	9, 0	2, 2

Stag Hunt

- Each player $i = 1, \ldots, n$ has reward tensor R_i, action space $\mathcal{A}_i$
- Each round, simultaneously choose distributions π_i over $\mathcal{A}_i$
- **Nash equilibrium:** Tuple $(\pi_1^*, \ldots, \pi_n^*)$ such that for any i, π_i:

$$\mathbb{E}_{(\pi_1^*,\ldots,\pi_n^*)} R_i \geq \mathbb{E}_{(\pi_1^*,\ldots\pi_{i-1}^*,\pi_i,\pi_{i+1}^*,\ldots,\pi_n^*)} R_i$$

- **Security value:** $\max_{\pi_i} \min_{(\pi_1,\ldots,\pi_{i-1},\pi_{i+1},\ldots,\pi_n)} \mathbb{E}_{(\pi_1,\ldots,\pi_n)} R_i$
- Repetition $\Rightarrow$ players adapt π_i to past history

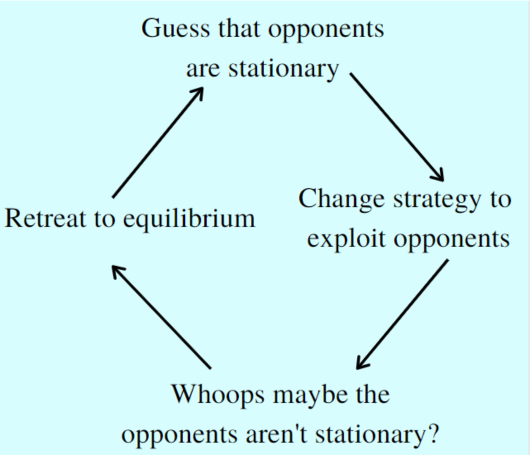
Balancing Several Goals

- Powers and Shoham [2004] criteria:
  1. Targeted Optimality: optimal policy for target class (e.g. stationary)
  2. Safety: achieve no worse than security value
  3. Compatibility: achieve value of an NE in self-play
- Challenge: Tradeoff betw *adaptation* (1) and *stability* (2, 3)

State of the Art

- Barely any theory so far!
- Mostly stateless games, asymptotics rather than regret...
- ...Or only self-play at expense of other goals [Tossou et al., 2020]

The AWESOME Algorithm



AWESOME: Key Attributes[1]

1. Learn to play optimally against eventually stationary opponents.
2. Convergence to Nash Equilibrium in self-play.

[1] See [Conitzer and Sandholm, 2006]

Detecting Non-stationarity

Suppose that your opponent is playing a stationary strategy if and only if:

$$\max_{a_i \in A_i} \left| p_{h_i}^{a_i} - p_{h_i^{prev}}^{a_i} \right| < \varepsilon_s$$

Detecting Non-stationarity

Suppose that your opponent is playing a stationary strategy if and only if:

$$\max_{a_i \in A_i} \left| p_{h_i}^{a_i} - p_{h_i^{prev}}^{a_i} \right| < \varepsilon_s$$

where A_i is the set of actions for the i^{th} player, h_i^{prev} gives the previous distribution from the last "epoch" of the game.

Detecting Non-stationarity

Suppose that your opponent is playing a stationary strategy if and only if:

$$\max_{a_i \in A_i} \left| p_{h_i}^{a_i} - p_{h_i^{prev}}^{a_i} \right| < \varepsilon_s$$

where A_i is the set of actions for the i^{th} player, h_i^{prev} gives the previous distribution from the last "epoch" of the game.

In a similar way, we detect if someone is playing the equilibrium.

Convergence

How do we get theoretical convergence results and not get stuck in a loop?

Convergence

How do we get theoretical convergence results and not get stuck in a loop?

Definition

A schedule $\{\varepsilon_e^t, \varepsilon_s^t, N^t\}_{t\in\mathbb{N}}$ is a called valid if

1. $\varepsilon_e^t, \varepsilon_s^t$ decrease monotonically to zero,
2. $N^t \nearrow \infty$,
3. $\Pi_{t\in\mathbb{N}}(1 - |A|_\Sigma \left[N^t(\varepsilon_s^{t+1})^2\right]^{-1}) > 0$,
4. $\Pi_{t\in\mathbb{N}}(1 - |A|_\Sigma \left[N^t(\varepsilon_e^{t})^2\right]^{-1}) > 0$.

Convergence

How do we get theoretical convergence results and not get stuck in a loop?

Definition

A schedule $\{\varepsilon_e^t, \varepsilon_s^t, N^t\}_{t\in\mathbb{N}}$ is a called valid if

1. $\varepsilon_e^t, \varepsilon_s^t$ decrease monotonically to zero,
2. $N^t \nearrow \infty$,
3. $\prod_{t\in\mathbb{N}}(1 - |A|_\Sigma \left[N^t(\varepsilon_s^{t+1})^2\right]^{-1}) > 0$,
4. $\prod_{t\in\mathbb{N}}(1 - |A|_\Sigma \left[N^t(\varepsilon_e^{t})^2\right]^{-1}) > 0$.

Theorem

A valid schedule exists.

AWESOME: Upshot of Valid Schedule

Theorem

Under a valid schedule, AWESOME converges to a Nash Equilibrium in self-play with probability 1.

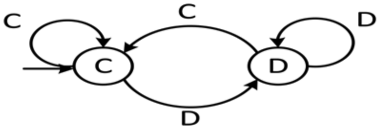
AWESOME: Upshot of Valid Schedule

Theorem

Under a valid schedule, AWESOME converges to a Nash Equilibrium in self-play with probability 1.
Instead if opponents are eventually stationary, then AWESOME converges to a best response with probability 1.

CMLeS: Adaptive Opponents and Safety Guarantee

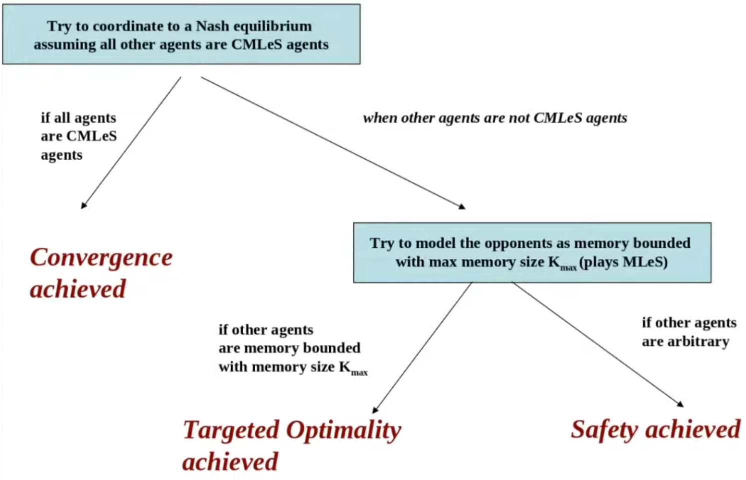
	C	**D**
C	3, 3	0, 4
D	4, 0	1, 1



Prisoner's Dilemma and "Tit-for-Tat" strategy

- Problem with AWESOME: non-stationary agents
  - Condition on "state" given by past K joint actions
  - $\Rightarrow$ Opponents are an "Adversary-Induced MDP"
  - Stage game NE not necessarily "optimal"
- Chakraborty and Stone [2010]: **C**onvergence with **M**odel **Le**arning and **S**afety

High Level

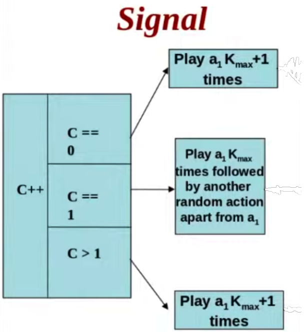


Credit: Peter Stone

CMLeS Details

1. Play NE for an epoch
2. If action frequencies suggest **not** playing the same NE:
   - Signal (with counter C): guaranteed to detect memory-bounded
   - If all players signaled, recompute an NE and return to (1)
3. Solve Adversary-Induced MDP with R-max

- At any step, if rewards less than security: $\arg\max_{\pi_i} \min_{(\pi_1,\dots,\pi_{i-1},\pi_{i+1},\dots,\pi_n)} \mathbb{E}_{(\pi_1,\dots,\pi_n)} R_i$



Credit: Peter Stone

A Hidden Question

The set up of the two previous algorithms begs the question: what is a "good" way to compute a Nash equilibrium?

Optimistic Nash Value Iteration (Nash-VI)

The overall strategy is:

1. Value iteration with “double” optimism to obtain a greedy policy π.
2. Execute π, collect samples, and reassess.

[2] See [Liu et al., 2021].

Optimistic Nash Value Iteration (Nash-VI)

The overall strategy is:

1. Value iteration with "double" optimism to obtain a greedy policy π.
2. Execute π, collect samples, and reassess.

Even this sharp-ish algorithm gets complexity:

$$\mathcal{O}\left(\Pi_{i\in I} A_i \cdot \frac{H^3 S}{\varepsilon^2}\right)$$

where H is the number of steps in each epsiode, S is the number of states, A_i is the number of actions for player i, and ε is a parameter of closeness to estimate the equilibrium.[2]

[2] See [Liu et al., 2021].

Complexity Issue

What are some known ways to solve complexity issues?

Complexity Issue

What are some known ways to solve complexity issues?

1. Make additional assumptions.
2. Mean field games (self-play adapts nicely here).

References I

Doran Chakraborty and Peter Stone. Convergence, targeted optimality and safety in multiagent learning. In *Proceedings of the Twenty-seventh International Conference on Machine Learning (ICML)*, 2010.

Vincent Conitzer and Tuomas Sandholm. Awesome: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents. *Machine Learning*, 2006.

Qinghua Liu, Tiancheng Yu, Yu Bai, and Chi Jin. A sharp analysis of model based reinforcement learning with self-play. 2021.

Rob Powers and Yoav Shoham. New criteria and a new algorithm for learning in multi-agent systems. *Neural Information Processing Systems*, 2004.

References II

Aristide C.Y. Tossou, Christos Dimitrakakis, Jaroslaw Rzepecki, and Katja Hofmann. A novel individually rational objective in multi-agent multi-armed bandits: Algorithms and regret bounds. *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems*, 2020.