Last-iterate Convergence of Decentralized Optimistic Gradient Descent/Ascent in Infinite-horizon Competitive Markov Games

Chen-Yu Wei Chung-Wei Lee Mengxiao Zhang Haipeng Luo

University of Southern California

Two-Player Zero-Sum Markov Games

- Markov games are multi-player Markov decision processes where the transition and reward are jointly determined by the all players.
- We study **two-player zero-sum** Markov games: Player 1's loss is equal to Player 2's reward (e.g., board games, sports games).
- Recently there are extensive study on using **centralized algorithms** to find the Nash equilibrium of the game (Wei et al.'17, Bai and Jin'20, Xie et al.'20)

Centralized algorithms: a central controller collects samples from the players, and assign the policies or actions to be played by the players.

- However, how to solve Markov games using **decentralized algorithms** is much less clear.

Decentralized algorithms: Players independently optimize their payoff without coordination.

Two-Player Zero-Sum Markov Games

Motivation of studying decentralized algorithms:

- **Simple and scalable** – no need to model other players
- **Versatile and robust** – applicable to other types of games (e.g., general-sum, more than 2 players).

Since each player myopically optimizes their payoff

A player usually can converge to the best-response of other players if other players are stationary. (rational)

However, since all players simultaneously update, when all players use decentralized algorithms, the system may be chaotic (e.g., cycling).

How to design decentralized algorithms with which the system converges to Nash equilibria when both players use the same algorithm. (convergent)

We are seeking an algorithm that is both rational and convergent.

Two-Player Zero-Sum Markov Games

Bowling and Veloso (2001) “Rational and convergent learning in stochastic games”

Our contribution: developed a provably rational and convergent algorithm with a *finite-time* guarantee for the first time.

Besides, in our algorithm, the learner does not need to observe the opponent’s action (only need reward feedback).

Prior Works

Provably rational algorithms:

Standard algorithms for MDPs (e.g., Q-learning, Policy Gradient)
(not convergent / unclear whether they converge)

Provably convergent algorithms:

Minimax-Q (Szepesvari and Littman, 1999)
Algorithms with coordinated policies (Bai and Jin, 2020; Xie et al., 2020)
(not rational)

Provably rational and convergent algorithms:

(Perolat et al., 2018) and (Sayin et al., 2020)
(only asymptotic convergence is proven)

Two-Player Zero-Sum Markov Games

State space (finite): $\mathcal{S}$

Player 1’s action space (finite): $\mathcal{A}$

Player 2’s action space (finite): $\mathcal{B}$

Loss/payoff function:

$$\sigma : \mathcal{S} \times \mathcal{A} \times \mathcal{B} \to [-1, 1]$$

Transition kernel:

$$p : \mathcal{S} \times \mathcal{A} \times \mathcal{B} \to \Delta_{\mathcal{S}}$$

Discount factor: $\gamma \in (0, 1)$

Protocol

For $t = 1, 2, \ldots$

$$\left.\begin{array}{l}\text{Player 1 chooses } a_t \in \mathcal{A} \\ \text{Player 2 chooses } b_t \in \mathcal{B}\end{array}\right\}\text{simultaneously}$$

Player 1 **pays** $\sigma(s_t, a_t, b_t)$ to Player 2

$s_{t+1} \sim p(\cdot \mid s_t, a_t, b_t)$

Two-Player Zero-Sum Markov Games

Value function under stationary policies $x : \mathcal{S} \to \Delta_{\mathcal{A}}$ and $y : \mathcal{S} \to \Delta_{\mathcal{B}}$:

$$V_{x,y}(s) = \mathbf{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} \sigma(s_t, a_t, b_t) \;\middle|\; s_1 = s, \quad a_t \sim x(\cdot|s_t), \quad b_t \sim y(\cdot|s_t)\right]$$

Nash equilibrium (x_*, y_*):

$$\max_y V_{x_*,y}(s) = V_{x_*,y_*}(s) = \min_x V_{x,y_*}(s)$$ (simultaneously for all s)

Settings

1. Full-information Setting

   Players know the transition kernel

   Players share x_t, y_t after round t

```
For $t = 1, 2, \dots$
    Player 1 chooses $a_t \sim x_t(\cdot|s_t)$
    Player 2 chooses $b_t \sim y_t(\cdot|s_t)$
    Player 1 pays $\sigma(s_t, a_t, b_t)$ to Player 2
    $s_{t+1} \sim p(\cdot \mid s_t, a_t, b_t)$
    Player 1&2 share $x_t(\cdot|s), \; y_t(\cdot|s) \;\; \forall s$
```

2. Learning Setting

   Players do not know the transition kernel

   Players only observes $\sigma(s_t, a_t, b_t)$ and s_{t+1} after round t

Settings

1. Full-information Setting

   Players know the transition kernel

   Players share x_t, y_t after round t

2. Learning Setting

   Players do not know the transition kernel

   Players only observes $\sigma(s_t, a_t, b_t)$ and s_{t+1} after round t

For $t = 1, 2, \ldots$

Player 1 chooses $a_t \sim x_t(\cdot|s_t)$

Player 2 chooses $b_t \sim y_t(\cdot|s_t)$

Player 1 **pays** $\sigma(s_t, a_t, b_t)$ to Player 2

$s_{t+1} \sim p(\cdot \mid s_t, a_t, b_t)$

Player 1&2 share $x_t(\cdot|s), \; y_t(\cdot|s) \;\; \forall s$

Matrix Game ($|\mathcal{S}| = 1$)

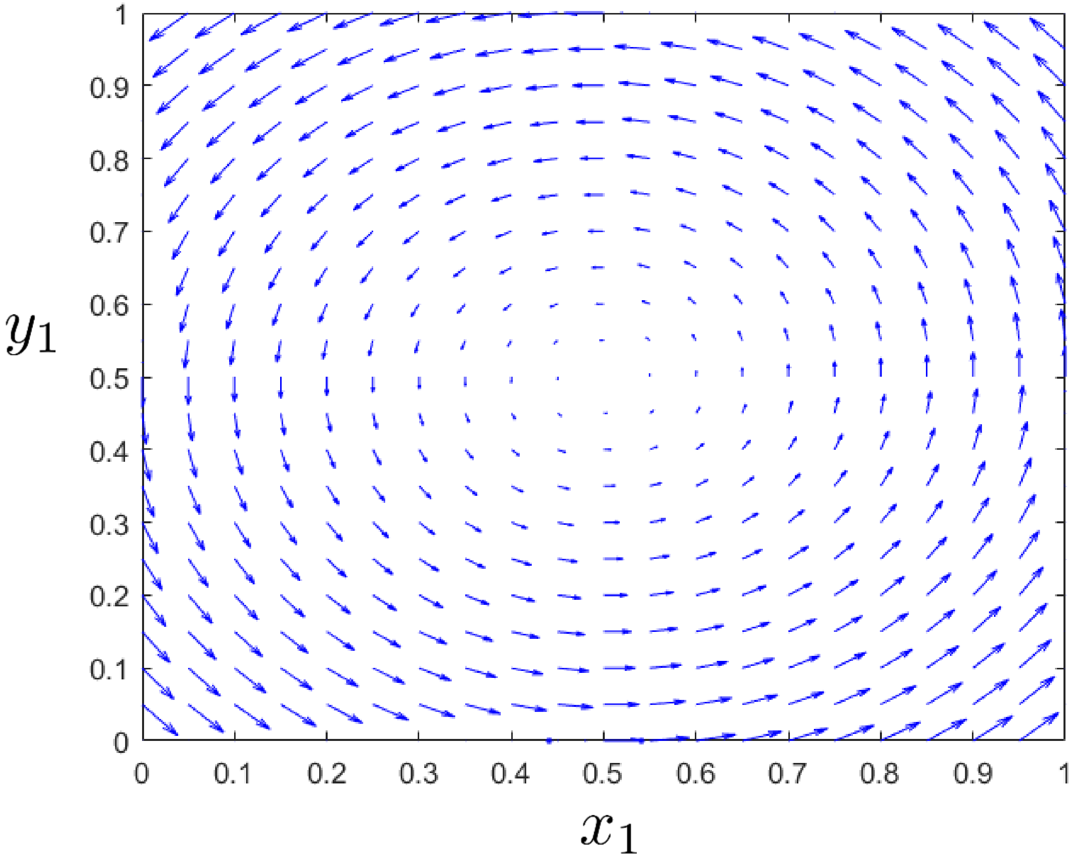
Projected Gradient Descent/Ascent (GDA)

$$x_{t+1} = \Pi_{\Delta_{\mathcal{A}}}\left[x_t - \eta \overset{\text{known}}{\downarrow}{G} \overset{\text{known}}{\downarrow}{y_t}\right]$$

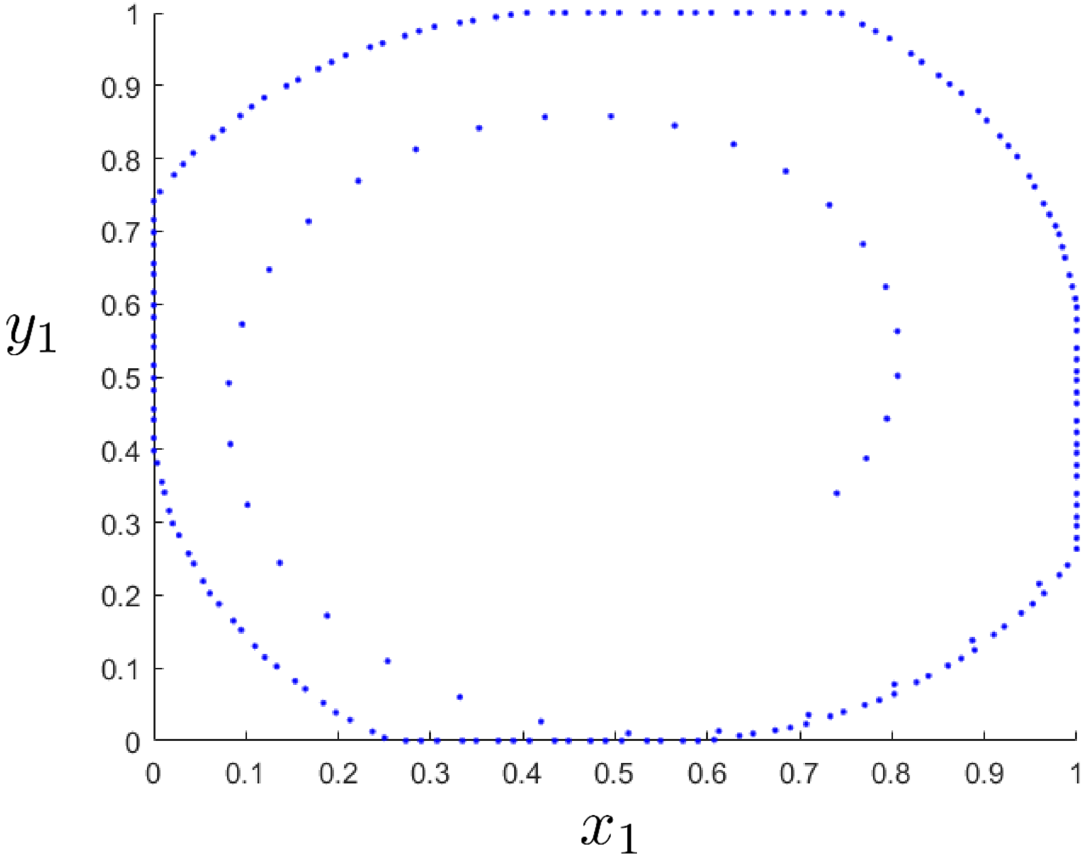
$$y_{t+1} = \Pi_{\Delta_{\mathcal{B}}}\left[y_t + \eta G^\top x_t\right]$$

The dynamics of GDA

$$G = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$



Update direction



Trajectory ($\eta = 0.1$)

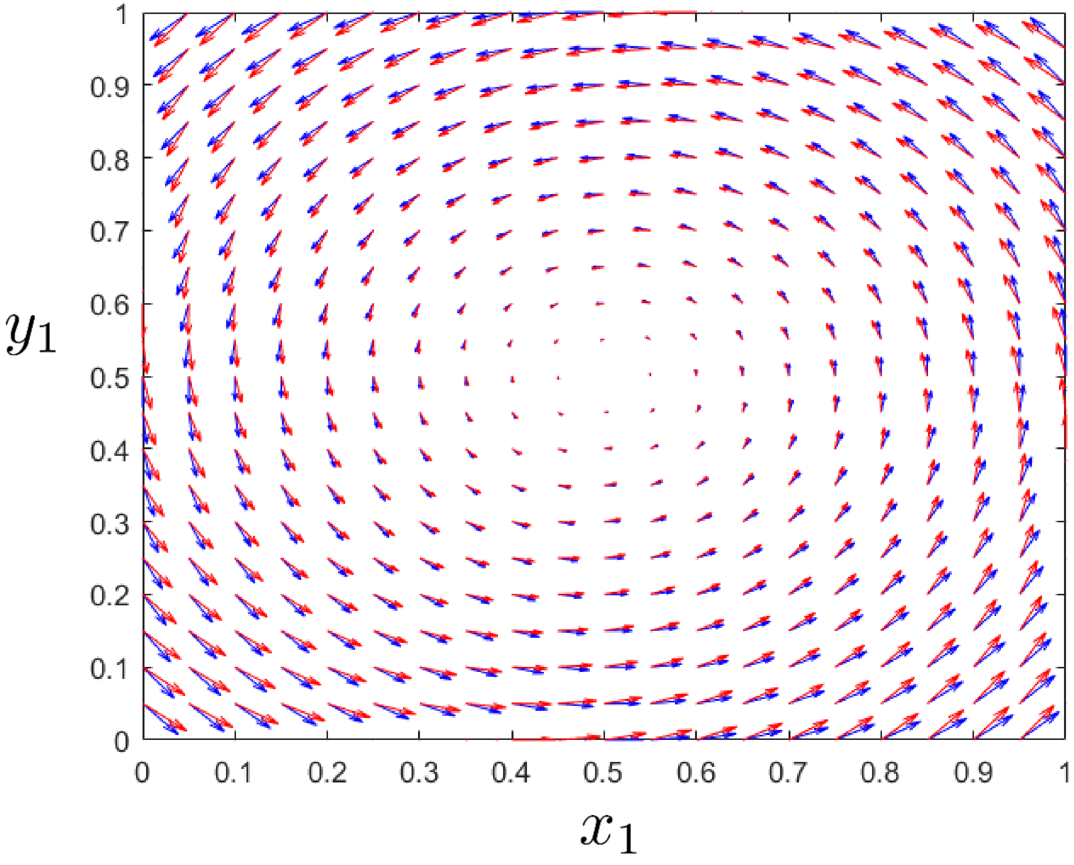
Optimistic Gradient Descent/Ascent (OGDA)

$$\widehat{x}_{t+1} = \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_t - \eta G y_t\right]$$
$$x_{t+1} = \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_{t+1} - \eta G y_t\right]$$

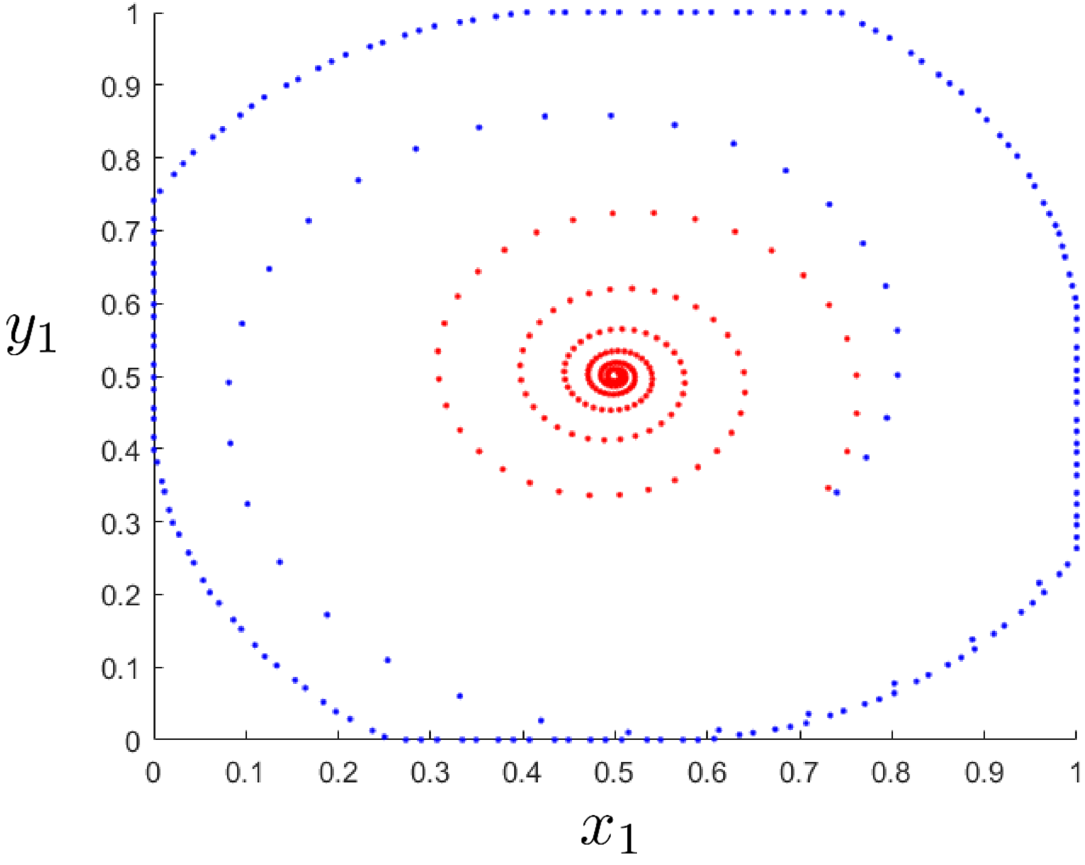
$$\widehat{y}_{t+1} = \Pi_{\Delta_{\mathcal{B}}}\left[\widehat{y}_t + \eta G^\top x_t\right]$$
$$y_{t+1} = \Pi_{\Delta_{\mathcal{B}}}\left[\widehat{y}_{t+1} + \eta G^\top x_t\right]$$

The dynamics of OGDA

Blue: GDA
Red: OGDA



Update direction



Trajectory ($\eta = 0.1$)

Convergence Analysis for Matrix Games under OGDA

(Wei et al. (2021))

$$z_t := (x_t, y_t) \qquad \widehat{z}_t := (\widehat{x}_t, \widehat{y}_t) \qquad z_* := (x_*, y_*)$$

$\delta_t := \|\widehat{z}_t - z_{t-1}\|^2 + \|z_{t-1} - \widehat{z}_{t-1}\|^2$ (the move between round $t-1$ and t)

Define potential $\boxed{\Phi_t := \|\widehat{z}_t - z_*\|^2 + \frac{1}{16}\delta_t}$

OGDA ensures $\boxed{\Phi_{t+1} \leq \left(1 - 0.1\eta^2 C^2\right)\Phi_t - 0.01\delta_t}$

$$\Rightarrow \|\widehat{z}_t - z_*\|^2 \leq \Phi_t \leq \left(1 - 0.1\eta^2 C^2\right)^t \Phi_0$$

where $C > 0$ is such that $\mathsf{Gap}(z) := \max_{x', y'} \left(x^\top G y' - x'^\top G y\right) \geq C\|z - z_*\|$

Extension to Markov Games (naively)

Perform OGDA on individual states: for all $s \in \mathcal{S}$,

$$\widehat{x}_{t+1}(\cdot|s) = \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_t(\cdot|s) - \eta Q_{x_t,y_t}(s,\cdot,\cdot)y_t(\cdot|s)\right]$$

$$x_{t+1}(\cdot|s) = \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_{t+1}(\cdot|s) - \eta \underbrace{Q_{x_t,y_t}(s,\cdot,\cdot)}_{\text{Game matrix}}y_t(\cdot|s)\right]$$

(y’s updates are similar)

$$Q_{x,y}(s,a,b) = \mathbf{E}\left[\sum_{t=1}^{\infty}\gamma^{t-1}\sigma(s_t,a_t,b_t) \,\middle|\, (s_1,a_1,b_1) = (s,a,b), \quad a_t \sim x(\cdot|s_t), \quad b_t \sim y(\cdot|s_t) \ \ \forall t \geq 2\right]$$

Extension to Markov Games (naively)

Similarly, define

$$\delta_t(s) := \|\widehat{z}_t(\cdot|s) - z_{t-1}(\cdot|s)\|^2 + \|z_{t-1}(\cdot|s) - \widehat{z}_{t-1}(\cdot|s)\|^2$$
$$\Phi_t(s) := \|\widehat{z}_t(\cdot|s) - z_*(\cdot|s)\|^2 + \tfrac{1}{16}\delta_t(s)$$

Then

$$\begin{aligned}\Phi_{t+1}(s) \leq \left(1 - 0.1\eta^2 C^2\right)\Phi_t(s) - 0.01\delta_t(s)\\ +10\eta\|Q_{x_t,y_t}(s,\cdot,\cdot) - Q_{x_*,y_*}(s,\cdot,\cdot)\|\\ +10\eta^2\|Q_{x_t,y_t}(s,\cdot,\cdot) - Q_{x_{t-1},y_{t-1}}(s,\cdot,\cdot)\|^2\end{aligned}$$

Because Q_{x_t,y_t} is unstable, the potential $\Phi_t(s)$ does not monotonically decrease.

Addressing the Issue

$$
\begin{aligned}
\Phi_{t+1}(s) \le \Big(1 - & \boxed{0.1\eta^2 C^2\Big)\,\Phi_t(s)}\ \boxed{-\,0.01\delta_t(s)} \\
& \boxed{\begin{aligned} &+10\eta\|Q_{x_t,y_t}(s,\cdot,\cdot) - Q_{x_*,y_*}(s,\cdot,\cdot)\| \\ &+10\eta^2\|Q_{x_t,y_t}(s,\cdot,\cdot) - Q_{x_{t-1},y_{t-1}}(s,\cdot,\cdot)\|^2 \end{aligned}}
\end{aligned}
$$

1. Making the “game matrix” on each state more stable (reducing the positive terms)
2. Using the extra negative term to cancel the positive terms

Addressing the Issue

$$\widehat{x}_{t+1}(\cdot|s) = \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_t(\cdot|s) - \eta Q_{x_t,y_t}(s,\cdot,\cdot)y_t(\cdot|s)\right]$$

$$x_{t+1}(\cdot|s) = \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_{t+1}(\cdot|s) - \eta Q_{x_t,y_t}(s,\cdot,\cdot)y_t(\cdot|s)\right]$$

$$\Downarrow$$

Actor (OGDA)

$$\left[\begin{aligned} \widehat{x}_{t+1}(\cdot|s) &= \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_t(\cdot|s) - \eta Q_t(s,\cdot,\cdot)y_t(\cdot|s)\right] \\ x_{t+1}(\cdot|s) &= \Pi_{\Delta_{\mathcal{A}}}\left[\widehat{x}_{t+1}(\cdot|s) - \eta Q_t(s,\cdot,\cdot)y_t(\cdot|s)\right] \end{aligned}\right.$$

Critic

$$\left[\begin{aligned} Q_t(s,a,b) &:= \sigma(s,a,b) + \gamma \mathbf{E}_{s'\sim p(\cdot|s,a,b)}\left[V_{t-1}(s')\right] \\ V_t(s) &= (1-\alpha_t)V_{t-1}(s) + \enclose{circle}{\alpha_t}\, x_t(\cdot|s)^\top Q_t(s,\cdot,\cdot)y_t(\cdot|s) \end{aligned}\right.$$

Making Q_t change slowly

Analysis

$$\Phi_{t+1}(s) \le \left(1 - 0.1\eta^2 C^2\right) \Phi_t(s) - 0.01\delta_t(s)$$

$$\boxed{\begin{aligned}&+10\eta\|Q_t(s,\cdot,\cdot) - Q_*(s,\cdot,\cdot)\| \\ &+10\eta^2\|Q_t(s,\cdot,\cdot) - Q_{t-1}(s,\cdot,\cdot)\|^2\end{aligned}} \quad (\star)$$

$$(\star) \le \mathcal{O}\left(\max_{s'} \sum_{\tau=1}^{t} \beta_t^\tau \delta_\tau(s') + \sum_{\tau=1}^{t} \beta_t^\tau \alpha_\tau \Phi_\tau(s')\right)$$

cancel

cancel

$-0.01\delta_t(s)$

$-0.05\eta^2 C^2 \Phi_t(s)$

The cancellation happens after we (weighted) sum the inequality over t and s

Theorem (Convergence in the Full-info Setting)

With $\alpha_t = \frac{2+(1-\gamma)}{2+t(1-\gamma)}$ and $\eta \le \frac{1}{10000}\sqrt{\frac{(1-\gamma)^5}{|\mathcal{S}|}}$, our algorithm ensures

1. $$\frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} \|\widehat{z}_t(\cdot|s) - z_*(\cdot|s)\|^2 = \mathcal{O}\left(\frac{|\mathcal{S}|^2}{\eta^4 C^4 (1-\gamma)^4 t}\right)$$

C: A quantity that depends on the Markov game; always positive

2. $$\frac{1}{t}\sum_{\tau=1}^{t} \mathsf{Gap}(x_\tau, y_\tau) := \frac{1}{t}\sum_{\tau=1}^{t} \max_{s,x',y'} \left(V_{\widehat{x}_\tau, y'}(s) - V_{x', \widehat{y}_\tau}(s)\right) = \mathcal{O}\left(\frac{|\mathcal{S}|}{\eta(1-\gamma)^2}\sqrt{\frac{\log t}{t}}\right)$$

Open Problems

- Removing the $|\mathcal{S}|$ dependency in the bound (necessary for the function approximation setting)
- (for the Learning Setting) removing the irreducible assumption
- (for the Learning Setting) removing the requirement of synchronization