Fast Convergence of Regularized Learning in Games

Vasilis Syrgkanis
Microsoft Research NYC

Alekh Agarwal
Microsoft Research NYC

Haipeng Luo
Princeton University

Robert Schapire
Microsoft Research NYC
Strategic Interactions in Computer Systems

• Game theoretic scenarios in modern computer systems
Strategic Interactions in Computer Systems

• Game theoretic scenarios in modern computer systems

internet routing

advertising auctions
How Do Players Behave?

• Classical game theory: players play according to Nash Equilibrium
How Do Players Behave?

• Classical game theory: players play according to Nash Equilibrium
  • How do players converge to equilibrium?
  • Nash Equilibrium is computationally hard

Caveats!
How Do Players Behave?

• Classical game theory: players play according to Nash Equilibrium
  • How do players converge to equilibrium?
  • Nash Equilibrium is computationally hard

• Most scenarios: repeated strategic interactions
  • Simple adaptive game playing more natural
  • Learn to play well over time from past experience

Caveats!
How Do Players Behave?

• Classical game theory: players play according to Nash Equilibrium
  • How do players converge to equilibrium?
  • Nash Equilibrium is computationally hard

• Most scenarios: repeated strategic interactions
  • Simple adaptive game playing more natural
  • Learn to play well over time from past experience
  • e.g. Dynamic bid optimization tools in online ad auctions

Caveats!
No-Regret Learning in Games

• Each player uses no-regret learning algorithm with good regret against adaptive adversaries
No-Regret Learning in Games

• Each player uses no-regret learning algorithm with good regret against adaptive adversaries

• Regret of each player against any fixed strategy converges to zero
No-Regret Learning in Games

• Each player uses no-regret learning algorithm with good regret against adaptive adversaries

• Regret of each player against any fixed strategy converges to zero

• Many simple algorithms achieve no-regret
  
  • MWU, Regret Matching, Follow the Regularized/Perturbed Leader, Mirror Descent [Freund and Schapire 1995, Foster and Vohra 1997, Hart and Mas-Collel 2000, Cesa-Bianchi and Lugosi 2006,...]
Known Convergence Results

• Empirical distribution converges to generalization of Nash equilibrium: *Coarse Correlated Equilibrium* [Young2004]
Known Convergence Results

• Empirical distribution converges to generalization of Nash equilibrium: Coarse Correlated Equilibrium [Young2004]

• Sum of player utilities – Welfare - converges to approximate optimality (under structural conditions on the game) [Roughgarden2010]
Known Convergence Results

• Empirical distribution converges to generalization of Nash equilibrium: Coarse Correlated Equilibrium [Young2004]

• Sum of player utilities – Welfare - converges to approximate optimality (under structural conditions on the game) [Roughgarden2010]

• Convergence rate inherited from adversarial analysis: typically $O\left(\frac{1}{\sqrt{T}}\right)$
Known Convergence Results

- Empirical distribution converges to generalization of Nash equilibrium: Coarse Correlated Equilibrium [Young2004]

- Sum of player utilities – Welfare - converges to approximate optimality (under structural conditions on the game) [Roughgarden2010]

- Convergence rate inherited from adversarial analysis: typically $O\left(\frac{1}{\sqrt{T}}\right)$
- $O\left(\frac{1}{\sqrt{T}}\right)$ impossible to improve against adversaries
Main Question

When all players invoke no-regret learning:
Is convergence faster than adversarial rate?
High-Level Main Results

Yes! We prove that if each player’s algorithm:
1. Makes **stable predictions**
2. Regret bounded by **stability of the environment**
High-Level Main Results

Yes! We prove that if each player’s algorithm:

1. Makes **stable predictions**
2. Regret bounded by **stability of the environment**

Then convergence **faster** than $1/\sqrt{T}$
High-Level Main Results

Yes! We prove that if each player’s algorithm:

1. Makes stable predictions
2. Regret bounded by stability of the environment

Then convergence faster than $1/\sqrt{T}$

Can be achieved by regularization and recency bias.
Model and Technical Results
Repeated Game Model

• $n$ players play a normal form game for $T$ time steps

• Each player $i$ has strategy space $S_i$ and utility $U_i : S_1 \times \cdots \times S_n \to [0,1]$
Repeated Game Model

• $n$ players play a normal form game for $T$ time steps

• Each player $i$ has strategy space $S_i$ and utility $U_i : S_1 \times \cdots \times S_n \rightarrow [0,1]$
Repeated Game Model

- $n$ players play a normal form game for $T$ time steps
- Each player $i$ has bids in an auction and utility from winning an item at some price
Repeated Game Model

• $n$ players play a normal form game for $T$ time steps

• Each player $i$ has paths in a network and latency of chosen path
Repeated Game Model

• $n$ players play a normal form game for $T$ time steps

• Each player $i$ has strategy space $S_i$ and utility $U_i: S_1 \times \cdots \times S_n \rightarrow [0,1]$
Repeated Game Model

• $n$ players play a normal form game for $T$ time steps

• Each player $i$ has strategy space $S_i$ and utility $U_i : S_1 \times \cdots \times S_n \to [0,1]$

• At each step $t = 1 \ldots T$, each player $i$
  • Picks a distribution over strategies $p_i^t \in \Delta(S_i)$ and draws $s_i^t \sim p_i^t$
Repeated Game Model

- $n$ players play a normal form game for $T$ time steps
- Each player $i$ has strategy space $S_i$ and utility $U_i : S_1 \times \cdots \times S_n \to [0,1]$
- At each step $t = 1 \ldots T$, each player $i$
  - Picks a distribution over strategies $\mathbf{p}_i^t \in \Delta(S_i)$ and draws $s_i^t \sim \mathbf{p}_i^t$
  - Receives expected utility: $E[U_i(s_1^t, \ldots, s_n^t)]$
Repeated Game Model

• $n$ players play a normal form game for $T$ time steps

• Each player $i$ has strategy space $S_i$ and utility $U_i:S_1 \times \cdots \times S_n \to [0,1]$

• At each step $t = 1 \ldots T$, each player $i$:
  
  • Picks a distribution over strategies $p_i^t \in \Delta(S_i)$ and draws $s_i^t \sim p_i^t$
  
  • Receives expected utility: $E[U_i(s_1^t, \ldots, s_n^t)]$

  • Observes expected utility vector $u_i^t$, where coordinate $u_i^t[s_i]$ is expected utility had strategy $s_i$ been played

    $u_i^t[s_i] = E[U_i(s_1^t, \ldots, s_i, \ldots, s_n^t)]$
Regret of Learning Algorithm

- Regret: gain from switching to best fixed strategy

\[
\text{Regret}_i = \max_{s_i \in \mathcal{S}} \mathcal{E}_t = \frac{1}{T} \sum_{t=1}^{T} u_{i(s_1, \ldots, s_i, \ldots, s_n)} - \mathcal{E}_t = \frac{1}{T} \sum_{t=1}^{T} u_{i(s_1, \ldots, s_i, \ldots, s_n)}
\]
Regret of Learning Algorithm

- Regret: gain from switching to best fixed strategy

\[
\text{Regret}_i = \max_{s_i \in S_i} E \left[ \sum_{t=1}^{T} U_i(s_1^t, \ldots, s_i^t, \ldots, s_n^t) \right] - E \left[ \sum_{t=1}^{T} U_i(s_1^t, \ldots, s_i^t, \ldots, s_n^t) \right]
\]
Regret of Learning Algorithm

• Regret: gain from switching to best fixed strategy

\[
\text{Regret}_i = \max_{s_i \in S_i} E \left[ \sum_{t=1}^{T} U_i(s_{1}^{t}, \ldots, s_i^{t}, \ldots, s_{n}^{t}) \right] - E \left[ \sum_{t=1}^{T} U_i(s_{1}^{t}, \ldots, s_i^{t}, \ldots, s_{n}^{t}) \right]
\]

• Many algorithms achieve regret \(O(\sqrt{T})\)
Main Results

We give general sufficient conditions on learning algorithms (and concrete example families of algorithms) such that:

Thm 1. Each player's regret is $O(T^{1/4}) \Rightarrow$ Convergence to CCE at $O(T^{-3/4})$

Thm 2. Sum of player regrets is $O(1)$

Corollary. Convergence to approximately optimal welfare if the game is "smooth" at $O(T^{-1})$

Thm 3. Give generic way to modify an algorithm to achieve small regret against "nice" algorithms and $O(T)$ against any adversarial sequence
Main Results

We give general sufficient conditions on learning algorithms (and concrete example families of algorithms) such that:

**Thm1.** Each player’s regret is $O(T^{1/4}) \Rightarrow$ Convergence to CCE at $O(T^{-3/4})$
Main Results

We give **general sufficient conditions** on learning algorithms (and concrete example families of algorithms) such that:

**Thm1.** Each player’s regret is $O(T^{1/4}) \Rightarrow$ Convergence to CCE at $O(T^{-3/4})$

**Thm2.** Sum of player regrets is $O(1)$
Main Results

We give general sufficient conditions on learning algorithms (and concrete example families of algorithms) such that:

**Thm1.** Each player’s regret is $O(T^{1/4}) \Rightarrow$ Convergence to CCE at $O(T^{-3/4})$

**Thm2.** Sum of player regrets is $O(1)$

**Corollary.** Convergence to approximately optimal welfare if the game is “smooth” at $O(T^{-1})$
Main Results

We give general sufficient conditions on learning algorithms (and concrete example families of algorithms) such that:

**Thm1.** Each player’s regret is $O(T^{1/4}) \Rightarrow$ Convergence to CCE at $O(T^{-3/4})$

**Thm2.** Sum of player regrets is $O(1)$

**Corollary.** Convergence to approximately optimal welfare if the game is “smooth” at $O(T^{-1})$

- Defined by [Roughgarden 2010]
- Main tool for proving welfare bounds (price of anarchy)
- Applicable to e.g. congestion games, ad auctions
Main Results

We give general sufficient conditions on learning algorithms (and concrete example families of algorithms) such that:

**Thm1.** Each player’s regret is $O(T^{1/4}) \Rightarrow$ Convergence to CCE at $O(T^{-3/4})$

**Thm2.** Sum of player regrets is $O(1)$

**Corollary.** Convergence to approximately optimal welfare if the game is “smooth” at $O(T^{-1})$

- Defined by [Roughgarden 2010]
- Main tool for proving welfare bounds (price of anarchy)
- Applicable to e.g. congestion games, ad auctions

**Thm3.** Give generic way to modify an algorithm to achieve small regret against “nice” algorithms and $\tilde{O}(\sqrt{T})$ against any adversarial sequence
Related Work

• [Daskalakis, Deckelbaum, Kim 2011]
  • Two player zero sum games
  • Marginal empirical distributions converge to Nash at rate $O\left(\frac{1}{T}\right)$
  • Unnatural coordinated dynamics

• [Rakhlin, Sridharan 2013]
  • Two player zero sum games
  • Optimistic Mirror Descent and Optimistic Hedge
  • Sum of regrets $O(1) \Rightarrow$ Marginal empirical dist. converge to Nash at $O\left(\frac{1}{T}\right)$
Related Work

- [Daskalakis, Deckelbaum, Kim 2011]
  - Two player zero sum games
  - Marginal empirical distributions converge to Nash at rate $O(\frac{1}{T})$
  - Unnatural coordinated dynamics

- [Rakhlin, Sridharan 2013]
  - Two player zero sum games
  - Optimistic Mirror Descent and Optimistic Hedge
  - Sum of regrets $O(1) \Rightarrow$ Marginal empirical dist. converge to Nash at $O(\frac{1}{T})$

This work:
- General multi-player games vs. 2-player zero-sum
- Convergence to CCE and Welfare vs. Nash
- Sufficient conditions vs. specific algorithms
- Give new example algorithms (e.g. general recency bias, Optimistic FTRL)
Main Results

We give general sufficient conditions on learning algorithms (and concrete example families of algorithms) such that:

**Thm1.** Each player’s regret is $O(T^{1/4}) \Rightarrow$ Convergence to CCE at $O(T^{-3/4})$

**Thm2.** Sum of player regrets is $O(1)$

**Corollary.** Convergence to approximately optimal welfare if the game is “smooth” at $O(T^{-1})$

**Thm3.** Give generic way to modify an algorithm to achieve small regret against “nice” algorithms and $\tilde{O}(\sqrt{T})$ against any adversarial sequence
Why Expect Faster Rates?

• If your opponent doesn’t change mixed strategy a lot
Why Expect Faster Rates?

• If your opponent doesn’t change mixed strategy a lot

• Your expected utility from a strategy is approximately the same between iterations
Why Expect Faster Rates?

- If your opponent doesn’t change mixed strategy a lot
- Your expected utility from a strategy is approximately the same between iterations
- Last iteration’s utility good proxy for next iteration’s utility
Why Expect Faster Rates?

• If your opponent doesn’t change mixed strategy a lot
  \[ \| p_2^t - p_2^{t-1} \|_1 \leq small \]

• Your expected utility from a strategy is approximately the same between iterations

• Last iteration’s utility good proxy for next iteration’s utility
Why Expect Faster Rates?

• If your opponent doesn’t change mixed strategy a lot

\[ \| p_2^t - p_2^{t-1} \|_1 \leq small \]

• Your expected utility from a strategy is approximately the same between iterations

• Last iteration’s utility good proxy for next iteration’s utility

\[ \| u_1^t - u_1^{t-1} \|_\infty \leq small \]
Simplified Sufficient Conditions for Fast Convergence

1. Stability of Mixed Strategies

\[ \| p_i^t - p_i^{t-1} \| \leq \eta \cdot \gamma \]

2. Regret Bounded by Stability of Utility Sequence

\[ \text{Regret}_i \leq \frac{\alpha}{\eta} + \eta \cdot \beta \sum_{t=1}^{T} \| u_i^t - u_i^{t-1} \|_*^2 \]
Simplified Sufficient Conditions for Fast Convergence

1. Stability of Mixed Strategies

\[ \| p_i^t - p_i^{t-1} \| \leq \eta \cdot \gamma \]

2. Regret Bounded by Stability of Utility Sequence

\[ \text{Regret}_i \leq \frac{\alpha}{\eta} + \eta \cdot \beta \sum_{t=1}^{T} \| u_i^t - u_i^{t-1} \|_{*}^2 \]

Then, for \( \eta = O(T^{-1/4}) \), each player’s regret is \( O(T^{1/4}) \)
Simplified Sufficient Conditions for Fast Convergence

1. Stability of Mixed Strategies

\[ \|p_i^t - p_i^{t-1}\| \leq A \]

2. Regret Bounded by Stability of Utility Sequence

\[ \text{Regret}_i \leq B + C \sum_{t=1}^{T} \|u_i^t - u_i^{t-1}\|_\star^2 - D \sum_{t=1}^{T} \|p_i^t - p_i^{t-1}\|^2 \]

Plus conditions on constants \(A, B, C, D\)
Example Algorithm

Hedge [Littlestone-Warmuth’94, Freund-Schapire’97]

\[ p_i^{t+1}[s_i] \propto e^\eta \sum_{\tau=1}^{t} u_\tau^i[s_i] \]
Example Algorithm

Hedge \cite{Littlestone-Warmuth'94, Freund-Schapire'97}

\[ p_{i}^{t+1}[s_i] \propto e^{\eta \sum_{\tau=1}^{t} u_{i}[s_i]} \]

Past cumulative performance of action
Example Algorithm

Hedge \ [\text{Littlestone-Warmuth'94, Freund-Schapire'97}] \quad \text{Optimistic Hedge} \ [\text{Rakhlin-Sridharan'13}]

\[ p_{i}^{t+1}[s_i] \propto e^{\eta \sum_{\tau=1}^{t} u_{i}^{\tau}[s_i]} \quad p_{i}^{t+1}[s_i] \propto e^{\eta \left( \sum_{\tau=1}^{t} u_{i}^{\tau}[s_i] + u_{i}^{t}[s_i] \right)} \]

Past cumulative performance of action
Example Algorithm

Hedge [Littlestone-Warmuth’94, Freund-Schapire’97]

\[ p_{i}^{t+1}[s_i] \propto e^{\eta \sum_{t=1}^{t} u_i^t[s_i]} \]

Optimistic Hedge [Rakhlin-Sridharan’13]

\[ p_{i}^{t+1}[s_i] \propto e^{\eta \left(\sum_{t=1}^{t} u_i^t[s_i]+u_i^t[s_i]\right)} \]

Past cumulative performance of action

Past performance double counting last iteration
Example Algorithm

**Hedge** [Littlestone-Warmuth’94, Freund-Schapire’97]

\[ p_{i+1}^{t}[s_i] \propto e^{\eta \sum_{\tau=1}^{t} u_{\tau}^t[s_i]} \]

**Optimistic Hedge** [Rakhlin-Sridharan’13]

\[ p_{i+1}^{t}[s_i] \propto e^{\eta (\sum_{\tau=1}^{t} u_{\tau}^t[s_i] + u_{i}^t[s_i])} \]

*Past cumulative performance of action*  
*Past performance double counting last iteration*

**Lemma.** Optimistic Hedge satisfies both sufficient conditions

**Intuition.**

1. Uses last iteration as “predictor” for next iteration
2. Equivalent to Follow the Regularized Leader; Regularization ⇒ Stability
Example Algorithm

Hedge [Littlestone-Warmuth’94, Freund-Schapire’97]
\[ p_{i}^{t+1}[s_i] \propto e^{\eta \sum_{\tau=1}^{t} u_{i}^{\tau}[s_i]} \]

Optimistic Hedge [Rakhlin-Sridharan’13]
\[ p_{i}^{t+1}[s_i] \propto e^{\eta (\sum_{\tau=1}^{t} u_{i}^{\tau}[s_i] + u_{i}^{t}[s_i])} \]

Past cumulative performance of action
Past performance double counting last iteration

Corollary. If all players in a game use optimistic Hedge with step size \( O(T^{-1/4}) \) then each player’s regret is \( O(T^{1/4}) \)
Example Algorithm

**Hedge** [Littlestone-Warmuth’94, Freund-Schapire’97]

\[ p_{i}^{t+1}[s_i] \propto e^{\eta \sum_{\tau=1}^{t} u_{i}^{\tau}[s_i]} \]

**Optimistic Hedge** [Rakhlin-Sridharan’13]

\[ p_{i}^{t+1}[s_i] \propto e^{\eta \left( \sum_{\tau=1}^{t} u_{i}^{\tau}[s_i] + u_{i}^{t}[s_i] \right)} \]

Past cumulative performance of action

Past performance double counting last iteration

Prove it extends to **Optimistic Follow the Regularized Leader** Algorithms

\[ p_{i}^{t+1} = \text{argmax}_{p \in \Delta(S_i)} \sum_{\tau=1}^{t} \langle p, u_{i}^{\tau} \rangle + \langle p, u_{i}^{t} \rangle + \frac{R(p)}{\eta} \]
Simulations
Simulation Example: Auction Game

• 4 bidders bidding on 4 items
• At each iteration bidder picks one item and submits bid
• Highest bidder on each item wins and pays bid
Simulation Example: Auction Game

• 4 bidders bidding on 4 items
• At each iteration bidder picks one item and submits bid
• Highest bidder on each item wins and pays bid
Simulation Example: Auction Game

- 4 bidders bidding on 4 items
- At each iteration bidder picks one item and submits bid
- Highest bidder on each item wins and pays bid
Simulation Example: Auction Game

• 4 bidders bidding on 4 items
• At each iteration bidder picks one item and submits bid
• Highest bidder on each item wins and pays bid

Simulation Example: Auction Game

• 4 bidders bidding on 4 items
• At each iteration bidder picks one item and submits bid
• Highest bidder on each item wins and pays bid
Simulation Example: Auction Game

• 4 bidders bidding on 4 items
• At each iteration bidder picks one item and submits bid
• Highest bidder on each item wins and pays bid
Main Take-Away Points

• Learning algorithms can enjoy faster regret rates in game theoretic environments
• Extend previous work to general multi-player games
• Provide generic sufficient conditions
Main Take-Away Points

• Learning algorithms can enjoy faster regret rates in game theoretic environments
• Extend previous work to general multi-player games
• Provide generic sufficient conditions

• Recency bias and regularization are key components
Main Take-Away Points

• Learning algorithms can enjoy faster regret rates in game theoretic environments
• Extend previous work to general multi-player games
• Provide generic sufficient conditions
• Recency bias and regularization are key components

Thank you!
Appendix Slides
Black-Box Robustness to Any Sequence

• What if opponents do not use a stable algorithm?

• Solution: use adaptive step-size, tracking change of environment
  • Keep upper estimate $B$ on path length $I_t = \sum_{\tau=1}^{t} \|u_{i\tau} - u_{i\tau-1}\|_*$
  • Set parameter $\eta = \min \left\{ \frac{a}{\sqrt{B}}, T^{-\frac{1}{4}} \right\}$
  • Once path length becomes larger, double estimate $B$ and restart algorithm

**Theorem.** Same fast regret guarantees (up to log factors) when opponents are stable; $\tilde{O}(\sqrt{T})$ when opponents arbitrary.
Simulation Example: Zero-Sum Game

Matching Pennies Style Game:

\[
\begin{array}{c|c|c}
& H & T \\
H & 1 & 0 \\
T & 0 & -1 \\
\end{array}
\]

Hedge Dynamics

Optimistic Hedge Dynamics

Trajectory of mixed strategies
Nash equilibrium

---

Prob. of Player 1 Playing H
Prob. of Player 2 Playing H

Prob. of Player 1 Playing H
Prob. of Player 2 Playing H

---

Trajectory of mixed strategies
Nash equilibrium