LIMITS AND LIMITATIONS OF LEARNING IN GAMES

Panayotis Mertikopoulos

French National Center for Scientific Research (CNRS)

Laboratoire d'Informatique de Grenoble (LIG)

⟨ Politecnico di Torino | November 11, 2022 ⟩

About

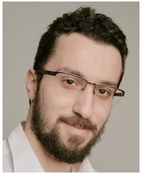

V. Boone


A. Giannou


C. Papadimitriou


G. Piliouras


W. H. Sandholm


M. Vlatakis

- Boone & M, *From equilibrium to resilience: Universal guarantees for the long-run behavior of learning in games,* working paper, 2022
- Giannou, Vlatakis & M, *Survival of the strictest: Stable and unstable equilibria under regularized learning with partial information,* COLT 2021
- Giannou, Vlatakis & M, *The convergence rate of regularized learning in games: From bandits and uncertainty to optimism and beyond,* NeurIPS 2021
- M, Papadimitriou & Piliouras, *Cycles in adversarial regularized learning,* SODA 2018
- M & Sandholm, *Learning in games via reinforcement and regularization,* Mathematics of Operations Research, vol. 41, no. 4, pp. 1297-1324, Nov. 2016.

Outline

Game of roads

A beautiful morning commute in the Bay Area

Learning in games

- **Multiple agents**, individual objectives

  # Select a route from home to work

- Payoffs determined by actions of **all** agents

  # Encounter other commuters on the road

- Agents receive payoffs, **adjust actions**, and the process repeats

  # Update road choice tomorrow

Learning in games

- **Multiple agents**, individual objectives

  # Select a route from home to work

- Payoffs determined by actions of **all** agents

  # Encounter other commuters on the road

- Agents receive payoffs, **adjust actions**, and the process repeats

  # Update road choice tomorrow

What does the agents' long-run behavior look like?

Learning in games, cont'd

Sequence of events – generic

for each ***epoch*** and every ***player*** **do**	# continuous / discrete
Choose ***action***	# continuous / finite
Receive ***reward***	# endogenous / exogenous
Get ***feedback*** (maybe)	# full info / oracle / payoff-based
end for	

Defining elements

- **Time:** continuous or discrete?
- **Players:** continuous or finite?
- **Actions:** continuous or finite?
- **Reward mechanism:** endogenous or endogenous (determined by other players or by “Nature”)?
- **Feedback:** observe other actions / other rewards / only received?

Learning in games, cont'd

Sequence of events – generic

for each ***epoch*** and every ***player*** **do** — # continuous / discrete

  Choose ***action*** — # continuous / finite

  Receive ***reward*** — # endogenous / exogenous

  Get ***feedback*** (maybe) — # full info / oracle / payoff-based

end for

Defining elements

- **Time:** continuous or discrete?
- **Players:** ~~continuous or~~ ***finite***
- **Actions:** ~~continuous or~~ ***finite***
- **Reward mechanism:** ***endogenous*** ~~or exogenous~~ (determined by other players ~~or by “Nature”~~)
- **Feedback:** observe other actions / other rewards / only received?

Finite games in normal form

Finite games

A ***finite game in normal form*** is a collection $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$ of the following primitives:

- A finite set of ***players*** $\mathcal{N} = \{1, \dots, N\}$
- A finite set of ***actions*** (or ***pure strategies***) $\mathcal{A}_i = \{1, \dots, A_i\}$ per player $i \in \mathcal{N}$
- An ensemble of ***payoff functions*** $u_i \colon \mathcal{A} \equiv \prod_i \mathcal{A}_i \to \mathbb{R}$, $i \in \mathcal{N}$

Notation:

- ***Action profile:*** $a = (a_1, \dots, a_N) \in \mathcal{A} := \prod_i \mathcal{A}_i$
- ***Realized payoff*** of player i:

$$u_i(a) \equiv u_i(a_1, \dots, a_N) \equiv u_i(a_i; a_{-i})$$

- ***Payoff vector*** of player i:

$$v_i(a) \equiv v_i(a_1, \dots, a_N) := (u_i(a_i'; a_{-i}))_{a_i' \in \mathcal{A}_i}$$

Mixed extensions

Mixed extension of a finite game:

- Given: finite game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$
- ***Mixed strategy*** of player i:

$$x_i = (x_{ia_i})_{a_i \in \mathcal{A}_i} \in \Delta(\mathcal{A}_i) =: \mathcal{X}_i$$

x_{ia_i} = prob. that player i plays $a_i \in \mathcal{A}_i$

- ***Mixed payoff*** of player i

$$u_i(x) = \mathbb{E}_{a \sim x}\, u_i(a) = \sum_{a_1 \in \mathcal{A}_1} \dots \sum_{a_N \in \mathcal{A}_N} x_{1,a_1} \cdots x_{N,a_N}\; u_i(a_1, \dots, a_N)$$

- ***Mixed payoff vector*** of player i:

$$v_i(x) \equiv v_i(x_1, \dots, x_N) \coloneqq (u_i(a_i; x_{-i}))_{a_i \in \mathcal{A}_i}$$

☞ vector of *expected* rewards

☞ $v_i(x)$ *only* depends on x_{-i}

- ***Notation:*** $\bar{\Gamma} \equiv \Delta(\Gamma)$

Nash equilibrium

Nash equilibrium (Nash, 1950)

"No player has an incentive to deviate from their chosen strategy if other players don't"

Nash equilibrium

Nash equilibrium (Nash, 1950)

"No player has an incentive to deviate from their chosen strategy if other players don't"

- For player payoffs:

$$u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*) \quad \text{for all } x_i \in \mathcal{X}_i,\, i \in \mathcal{N}$$

- For pure strategy payoffs:

$$u_i(a_i^*; x_{-i}^*) \geq u_i(a_i; x_{-i}^*) \quad \text{for all } a_i^* \in \operatorname{supp}(x_i^*),\, a_i \in \mathcal{A}_i,\, i \in \mathcal{N}$$

Nash equilibrium

Nash equilibrium (Nash, 1950)

"No player has an incentive to deviate from their chosen strategy if other players don't"

- **For player payoffs:**

$$u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*) \quad \text{for all } x_i \in \mathcal{X}_i,\, i \in \mathcal{N}$$

- **For pure strategy payoffs:**

$$v_{ia_i^*}(x^*) \geq v_{ia_i}(x^*) \qquad \text{for all } a_i^* \in \operatorname{supp}(x_i^*),\, a_i \in \mathcal{A}_i,\, i \in \mathcal{N}$$

Nash equilibrium

Nash equilibrium (Nash, 1950)

"No player has an incentive to deviate from their chosen strategy if other players don't"

- **For player payoffs:**

$$u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*) \quad \text{for all } x_i \in \mathcal{X}_i, i \in \mathcal{N}$$

- **For pure strategy payoffs:**

$$v_{ia_i^*}(x^*) \geq v_{ia_i}(x^*) \qquad \text{for all } a_i^* \in \operatorname{supp}(x_i^*), a_i \in \mathcal{A}_i, i \in \mathcal{N}$$

- **Pure equilibrium:** $\operatorname{supp}(x^*)$ = singleton # $x^* = a^* \in \mathcal{A}$
- **Strict equilibrium:** ">" instead of "≥" where appropriate # unique best response; necessarily pure

Nash equilibrium

Nash equilibrium (Nash, 1950)

"No player has an incentive to deviate from their chosen strategy if other players don't"

- **For player payoffs:**

$$u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*) \quad \text{for all } x_i \in \mathcal{X}_i,\, i \in \mathcal{N}$$

- **For pure strategy payoffs:**

$$v_{ia_i^*}(x^*) \geq v_{ia_i}(x^*) \qquad \text{for all } a_i^* \in \operatorname{supp}(x_i^*),\, a_i \in \mathcal{A}_i,\, i \in \mathcal{N}$$

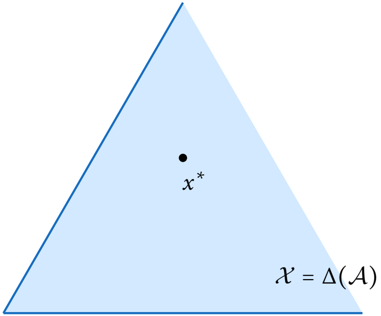
- **Pure equilibrium:** $\operatorname{supp}(x^*) = \text{singleton}$ # $x^* = a^* \in \mathcal{A}$
- **Strict equilibrium:** ">" instead of "≥" where appropriate # unique best response; necessarily pure

Variational formulation (Stampacchia, 1964)

$$\langle v(x^*), x - x^* \rangle \leq 0 \quad \text{for all } x \in \mathcal{X}$$

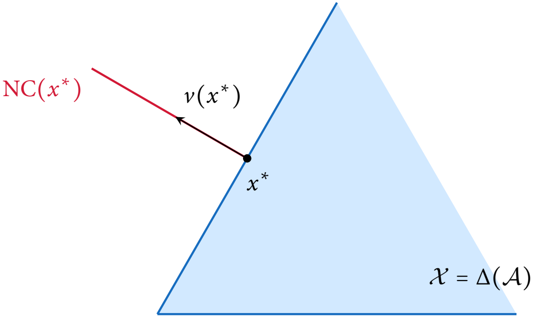
Equilibrium configurations

Figure. Different equilibrium configurations: ***fully mixed***



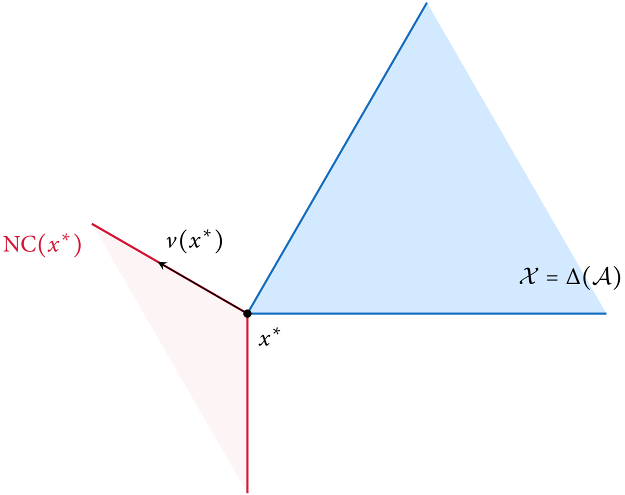
Equilibrium configurations

Figure. Different equilibrium configurations: fully mixed vs. ***mixed***



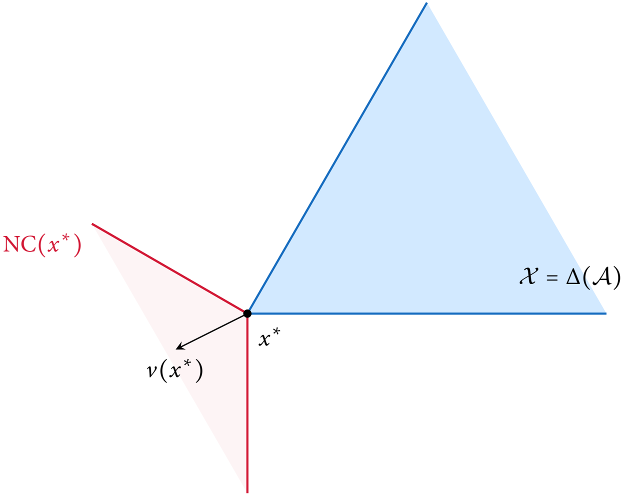
Equilibrium configurations

Figure. Different equilibrium configurations: fully mixed vs. mixed vs. ***pure***



Equilibrium configurations

Figure. Different equilibrium configurations: fully mixed vs. mixed vs. pure vs. ***strict***



Outline

Learning in continuous time

Sequence of events – continuous time

Require: finite game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$

repeat

At each epoch $t \geq 0$ **do simultaneously** for all players $i \in \mathcal{N}$ — # continuous time

Choose ***mixed strategy*** $x_i(t) \in \mathcal{X}_i := \Delta(\mathcal{A}_i)$ — # mixed extension

Observe mixed payoff vector $v_i(x(t))$ — # feedback phase

until end

Defining elements

- **Time:** $t \geq 0$
- **Players:** finite
- **Actions:** finite
- **Mixing:** yes
- **Feedback:** mixed payoff vectors

Learning with exponential weights

- Agents record cumulative payoff of each strategy

$$y_a(t) = \int_0^t v_a(x(\tau))\, d\tau$$

$\Longrightarrow$ ***propensity*** of choosing a strategy

Learning with exponential weights

- Agents record cumulative payoff of each strategy

$$y_a(t) = \int_0^t v_a(x(\tau))\, d\tau$$

$\implies$ ***propensity*** of choosing a strategy

- Choice probabilities $\rightsquigarrow$ exponentially proportional to propensity scores

$$x_a(t) \propto \exp(y_a(t))$$

➻ Littlestone & Warmuth (1994), Auer et al. (1995), Rustichini (1999), Sorin (2009)

Learning with exponential weights

- Agents record cumulative payoff of each strategy

$$y_a(t) = \int_0^t v_a(x(\tau))\, d\tau$$

$\Longrightarrow$ ***propensity*** of choosing a strategy

- Choice probabilities $\rightsquigarrow$ exponentially proportional to propensity scores

$$x_a(t) = \frac{\exp(y_a(t))}{\sum_{a'} \exp(y_{a'}(t))}$$

➠ Littlestone & Warmuth (1994), Auer et al. (1995), Rustichini (1999), Sorin (2009)

Learning with exponential weights

- Agents record cumulative payoff of each strategy

$$y_a(t) = \int_0^t v_a(x(\tau))\, d\tau$$

$\Longrightarrow$ ***propensity*** of choosing a strategy

- Choice probabilities $\rightsquigarrow$ exponentially proportional to propensity scores

$$x_a(t) = \frac{\exp(y_a(t))}{\sum_{a'} \exp(y_{a'}(t))}$$

➻ Littlestone & Warmuth (1994), Auer et al. (1995), Rustichini (1999), Sorin (2009)

- Evolution of mixed strategies

$$\dot{x}_a = \cdots = x_a[v_a(x) - u(x)]$$

Learning with exponential weights

- Agents record cumulative payoff of each strategy

$$y_a(t) = \int_0^t v_a(x(\tau))\,d\tau$$

$\Longrightarrow$ ***propensity*** of choosing a strategy

- Choice probabilities $\rightsquigarrow$ exponentially proportional to propensity scores

$$x_a(t) = \frac{\exp(y_a(t))}{\sum_{a'} \exp(y_{a'}(t))}$$

Littlestone & Warmuth (1994), Auer et al. (1995), Rustichini (1999), Sorin (2009)

- Evolution of mixed strategies

$$\dot{x}_a = \cdots = x_a[v_a(x) - u(x)]$$

Replicator dynamics (Taylor & Jonker, 1978)

$$\dot{x}_{ia_i} = x_{a_i}[v_{ia_i}(x) - u_i(x)] \tag{RD}$$

General case: regularized learning

- The logit map $\Lambda(y) = (\exp(y_a))_{a\in\mathcal{A}} / \sum_a \exp(y_a)$ approximates the "***leader***" (best response map)

$$y \mapsto \arg\max_{x\in\mathcal{X}} \langle y, x\rangle$$

General case: regularized learning

- The logit map $\Lambda(y) = (\exp(y_a))_{a\in\mathcal{A}} / \sum_a \exp(y_a)$ approximates the "***leader***" (best response map)

$$y \mapsto \arg\max_{x\in\mathcal{X}} \{\langle y, x\rangle - h(x)\}$$

where $h(x) = \sum_{a\in\mathcal{A}} x_a \log x_a$ is the (negative) entropy of $x \in \mathcal{X}$

General case: regularized learning

- The logit map $\Lambda(y) = (\exp(y_a))_{a\in\mathcal{A}} / \sum_a \exp(y_a)$ approximates the "***leader***" (best response map)

$$y \mapsto \arg\max_{x\in\mathcal{X}}\{\langle y, x\rangle - h(x)\}$$

  where $h(x) = \sum_{a\in\mathcal{A}} x_a \log x_a$ is the (negative) entropy of $x \in \mathcal{X}$

- **Regularized best responses**

$$Q(y) = \arg\max_{x\in\mathcal{X}}\{\langle y, x\rangle - h(x)\}$$

  where $h\colon \mathcal{X} \to \mathbb{R}$ is a (strictly) convex **regularizer function**

General case: regularized learning

- The logit map $\Lambda(y) = (\exp(y_a))_{a\in\mathcal{A}} / \sum_a \exp(y_a)$ approximates the "***leader***" (best response map)

$$y \mapsto \arg\max_{x\in\mathcal{X}} \{\langle y, x\rangle - h(x)\}$$

  where $h(x) = \sum_{a\in\mathcal{A}} x_a \log x_a$ is the (negative) entropy of $x \in \mathcal{X}$

- **Regularized best responses**

$$Q(y) = \arg\max_{x\in\mathcal{X}} \{\langle y, x\rangle - h(x)\}$$

  where $h\colon \mathcal{X} \to \mathbb{R}$ is a (strictly) convex **regularizer function**

Follow the regularized leader (FTRL) in continuous time

$$\dot{y}(t) = v(x(t))$$
$$x(t) = Q(y(t)) \tag{FTRL-C}$$

General case: regularized learning

- The logit map $\Lambda(y) = (\exp(y_a))_{a\in\mathcal{A}} / \sum_a \exp(y_a)$ approximates the "***leader***" (best response map)

$$y \mapsto \arg\max_{x\in\mathcal{X}} \{\langle y, x\rangle - h(x)\}$$

where $h(x) = \sum_{a\in\mathcal{A}} x_a \log x_a$ is the (negative) entropy of $x \in \mathcal{X}$

- **Regularized best responses**

$$Q(y) = \arg\max_{x\in\mathcal{X}} \{\langle y, x\rangle - h(x)\}$$

where $h\colon \mathcal{X} \to \mathbb{R}$ is a (strictly) convex **regularizer function**

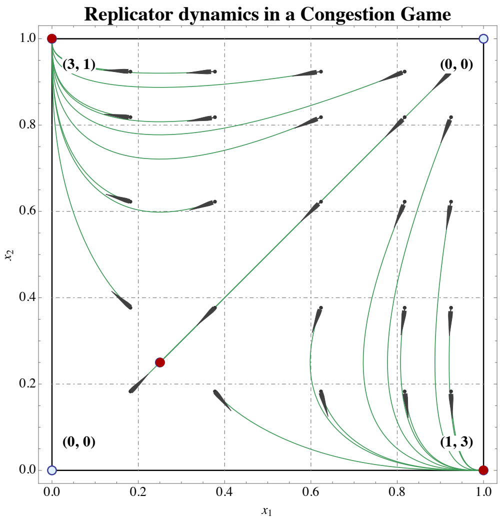
Follow the regularized leader (FTRL) in continuous time

$$\dot{y}(t) = v(x(t))$$
$$x(t) = Q(y(t)) \tag{FTRL-C}$$

Focus on entropy/replicator for simplicity

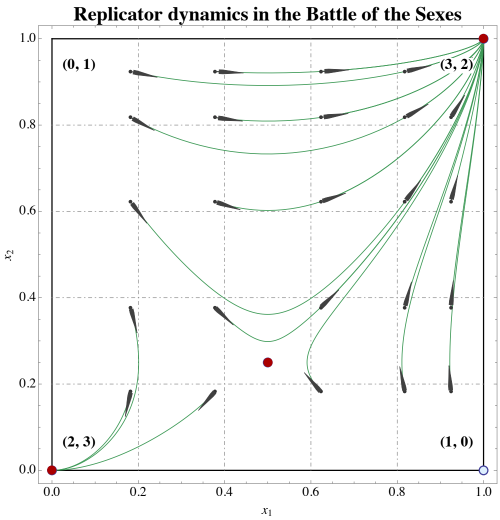
Evolution of mixed strategies: Examples

What do the dynamics look like?



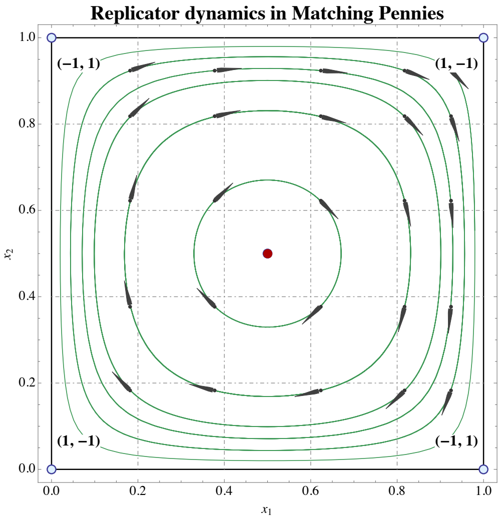
Evolution of mixed strategies: Examples

What do the dynamics look like?



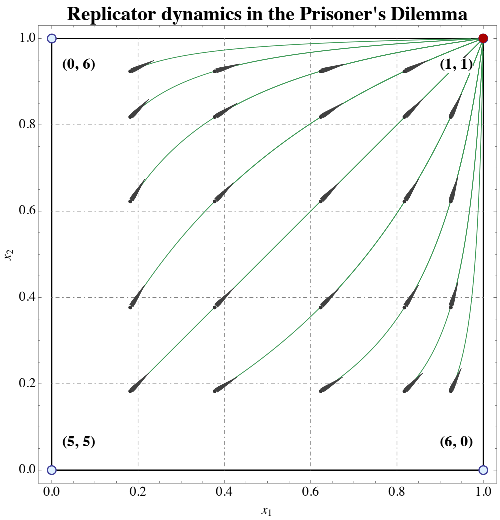
Evolution of mixed strategies: Examples

What do the dynamics look like?



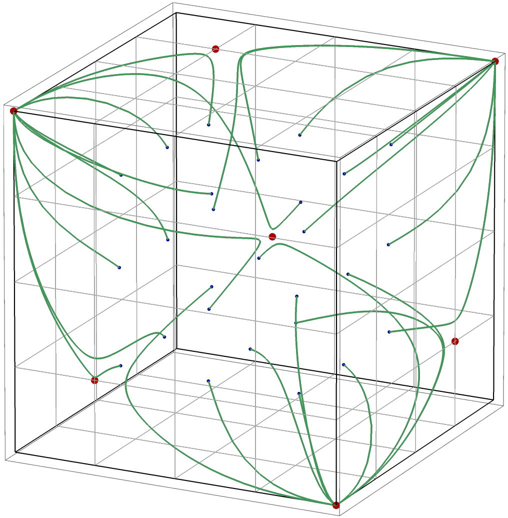
Evolution of mixed strategies: Examples

What do the dynamics look like?



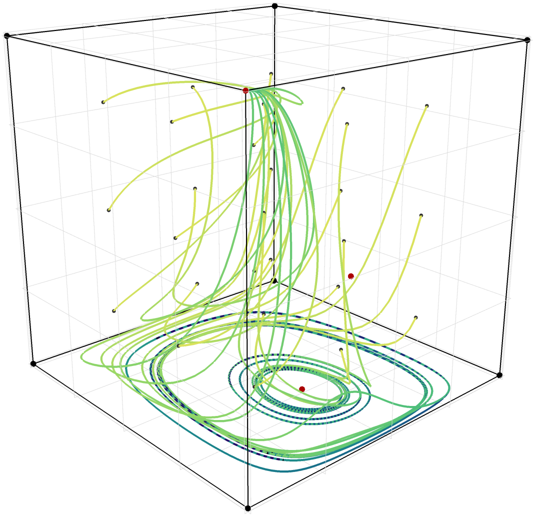
Evolution of mixed strategies: Examples

What do the dynamics look like?



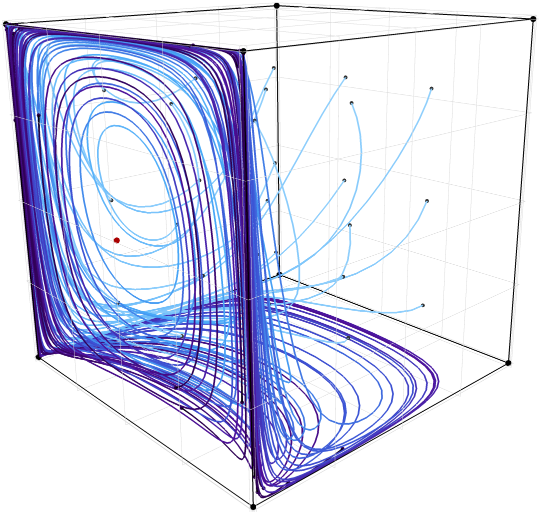
Evolution of mixed strategies: Examples

What do the dynamics look like?



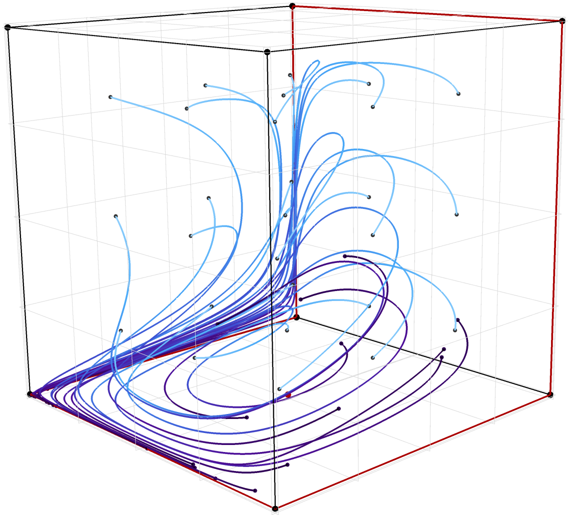
Evolution of mixed strategies: Examples

What do the dynamics look like?



Evolution of mixed strategies: Examples

What do the dynamics look like?



Stationarity versus stability

Stationarity of Nash equilibria

Let $x(t) = Q(y(t))$ be a trajectory of (FTRL-C). Then:

$$x(0) \text{ is a Nash equilibrium} \implies x(t) = x(0) \text{ for all } t \geq 0$$

⚠ The converse does not hold!

⚠ Are all stationary points created equal?

Stationarity versus stability

Stationarity of Nash equilibria

Let $x(t) = Q(y(t))$ be a trajectory of (FTRL-C). Then:

$$x(0) \text{ is a Nash equilibrium } \implies x(t) = x(0) \text{ for all } t \geq 0$$

⚠ The converse does not hold!

⚠ Are all stationary points created equal?

Definition (Notions of stability)

- x^* is *(Lyapunov) stable* if, for every neighborhood $\mathcal{U}$ of x^* in $\mathcal{X}$, there exists a neighborhood $\mathcal{U}'$ of x^* such that

$$x(0) \in \mathcal{U}' \implies x(t) \in \mathcal{U} \quad \text{for all } t \geq 0$$

Trajectories that start close to x^* remain close for all time

- x^* is *attracting* if $\lim_{t\to\infty} x(t) = x^*$ whenever $x(0)$ is close enough to x^*

Trajectories that start close to x^* eventually converge to x^*

- x^* is *asymptotically stable* if it is stable and attracting

A "folk theorem" for learning

Are all equilibria created equal?

Theorem (M & Sandholm, 2016; Flokas et al., 2020)

Let $x(t) = Q(y(t))$ *be a trajectory of* (FTRL-C). *Then:*

1. x^* *is Nash* $\implies$ x^* *is stationary*
2. $\lim_{t\to\infty} x(t) = x^*$ $\implies$ x^* *is Nash*
3. x^* *is stable* $\implies$ x^* *is Nash*
4. x^* *is stable and attracting* $\iff$ x^* *is strict Nash*

Some remarks:

- ***Only strict equilibria can be stable and attracting***
- For replicator dynamics $\rightsquigarrow$ folk theorem of evolutionary game theory ➥ Hofbauer & Sigmund, 2003

Non-convergence in min-max games

The min-max case is quite different (and special):

Non-convergence in min-max games

The min-max case is quite different (and special):

x^* is a fully mixed equilibrium $\implies$ (RD) admits a **constant of motion**

KL divergence: $D_{\mathrm{KL}}(x^*, x) = \sum_i \sum_{a_i} x^*_{ia_i} \log \frac{x^*_{ia_i}}{x_{ia_i}}$

Non-convergence in min-max games

The min-max case is quite different (and special):

x^* is a fully mixed equilibrium $\implies$ (RD) admits a **constant of motion**

KL divergence: $D_{\mathrm{KL}}(x^*, x) = \sum_i \sum_{a_i} x^*_{ia_i} \log \frac{x^*_{ia_i}}{x_{ia_i}}$

Theorem (Hofbauer et al., 2009)

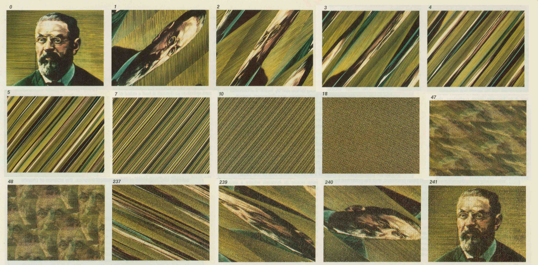
Assume a min-max game admits an interior equilibrium. Then:

- *Interior trajectories of* (RD) ***do not converge*** *(unless stationary)*
- *Time-averages* $\bar{x}(t) = t^{-1} \int_0^t x(\tau)\, d\tau$ ***converge to Nash equilibrium***

Poincaré recurrence in min-max games

Definition (Poincaré, 1890's)

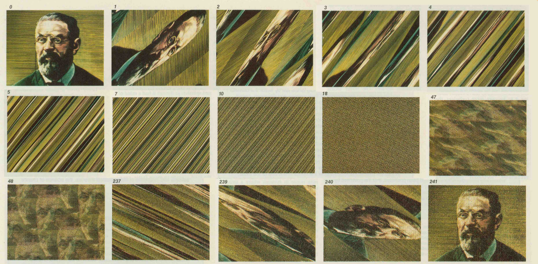
A dynamical system is **Poincaré recurrent** if almost all solution trajectories return ***arbitrarily close*** to their starting point ***infinitely many times***



Poincaré recurrence in min-max games

Definition (Poincaré, 1890's)

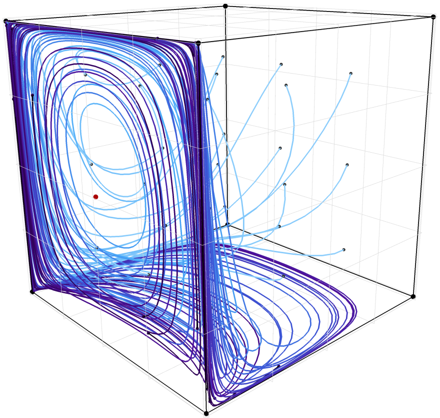
A dynamical system is **Poincaré recurrent** if almost all solution trajectories return ***arbitrarily close*** to their starting point ***infinitely many times***



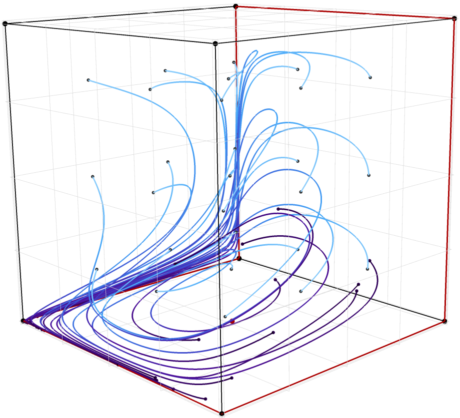
Theorem (M Papadimitriou & Piliouras, 2018)

The dynamics of FTRL are Poincaré recurrent in all min-max games with a fully mixed equilibrium

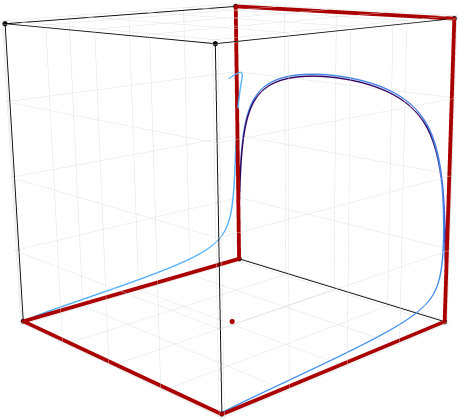
Is that all?



Is that all?



Is that all?



In many games, the dynamics are neither recurrent, nor pointwise convergent

Universal convergence guarantees

Can we characterize the limiting behavior of the FTRL dynamics?

Limit sets

The ***limit set*** of a trajectory $X(t)$ is the set of all its limit points, i.e.,

$$\mathcal{L}(X) := \bigcap_{t\geq 0} \mathrm{cl}\{X(s) : s \geq t\} = \{x \in \mathcal{X} : X(t_n) \to x \text{ for some sequence } t_n \to \infty\}$$

Equivalently, $\mathcal{L}(X)$ is the smallest subset of $\mathcal{X}$ such that $\mathrm{dist}(\mathcal{L}, X(t)) \to 0$ as $t \to \infty$

Universal convergence guarantees

Can we characterize the limiting behavior of the FTRL dynamics?

Limit sets

The ***limit set*** of a trajectory $X(t)$ is the set of all its limit points, i.e.,

$$\mathcal{L}(X) := \bigcap_{t\geq 0} \operatorname{cl}\{X(s) : s \geq t\} = \{x \in \mathcal{X} : X(t_n) \to x \text{ for some sequence } t_n \to \infty\}$$

Equivalently, $\mathcal{L}(X)$ is the smallest subset of $\mathcal{X}$ such that $\operatorname{dist}(\mathcal{L}, X(t)) \to 0$ as $t \to \infty$

Examples

- Nash equilibria — # Battle of the Sexes, coordination/anti-coordination games, ...
- Periodic orbits — # Matching Pennies, min-max games, ...
- Heteroclinic cycles — # chair game, ...
- …

Universal convergence guarantees

Can we characterize the limiting behavior of the FTRL dynamics?

Limit sets

The ***limit set*** of a trajectory $X(t)$ is the set of all its limit points, i.e.,

$$\mathcal{L}(X) := \bigcap_{t\geq 0} \operatorname{cl}\{X(s) : s \geq t\} = \{x \in \mathcal{X} : X(t_n) \to x \text{ for some sequence } t_n \to \infty\}$$

Equivalently, $\mathcal{L}(X)$ is the smallest subset of $\mathcal{X}$ such that $\operatorname{dist}(\mathcal{L}, X(t)) \to 0$ as $t \to \infty$

Theorem (Boone & M, 2022)

The limit set $\mathcal{L}$ of any solution trajectory $x(t) = Q(y(t))$ of (FTRL-C) *is characterized by the following properties:*

- **Minimality:** *$\mathcal{L}$ does not contain any proper attractors*
- **Resilience:** *every deviation x from $\mathcal{L}$ is unilaterally nullified by some element x^* of $\mathcal{L}$, i.e.,*

$$u_i(x^*) \geq u_i(x_i; x^*_{-i}) \quad \textit{for all } i \in \mathcal{N}$$

Outline

Learning in discrete time

Sequence of events – discrete time

Require: finite game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$

repeat

At each epoch $n = 1, 2, \dots$ **do simultaneously** for all players $i \in \mathcal{N}$ # discrete time

Choose ***mixed strategy*** $X_{i,n} \in \mathcal{X}_i := \Delta(\mathcal{A}_i)$ # mixed extension

Choose ***action*** $a_{i,n} \sim X_{i,n}$ # random action selection

Observe mixed payoff vector $v_i(X_n)$ # feedback phase

until end

Defining elements

- Time: $n = 1, 2, \dots$
- Players: finite
- Actions: finite
- Mixing: yes
- Feedback: mixed payoff vectors

Learning in discrete time

Sequence of events – discrete time

Require: finite game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$

repeat

At each epoch $n = 1, 2, \dots$ **do simultaneously** for all players $i \in \mathcal{N}$ — # discrete time

Choose ***mixed strategy*** $X_{i,n} \in \mathcal{X}_i := \Delta(\mathcal{A}_i)$ — # mixed extension

Choose ***action*** $a_{i,n} \sim X_{i,n}$ — # random action selection

Observe pure payoff vector $v_i(a_n)$ — # feedback phase

until end

Defining elements

- **Time:** $n = 1, 2, \dots$
- **Players:** finite
- **Actions:** finite
- **Mixing:** yes
- **Feedback:** pure payoff vectors

Learning in discrete time

Sequence of events – discrete time

Require: finite game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$

repeat

At each epoch $n = 1, 2, \dots$ **do simultaneously** for all players $i \in \mathcal{N}$ # discrete time

Choose ***mixed strategy*** $X_{i,n} \in \mathcal{X}_i := \Delta(\mathcal{A}_i)$ # mixed extension

Choose ***action*** $a_{i,n} \sim X_{i,n}$ # random action selection

Observe realized payoff $u_i(a_n)$ # feedback phase

until end

Defining elements

- **Time:** $n = 1, 2, \dots$
- **Players:** finite
- **Actions:** finite
- **Mixing:** yes
- **Feedback:** realized payoffs

The feedback process

Types of feedback

From best to worst (more to less info):

- ***Mixed payoff vectors***: $v_i(X_n)$ # deterministic vector feedback
- ***Pure payoff vectors***: $v_i(a_n)$ # stochastic vector feedback
- ***Bandit / Payoff-based:*** $u_i(a_n)$ # stochastic scalar feedback

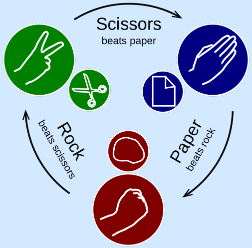
The feedback process

Types of feedback

From best to worst (more to less info):

- ***Mixed payoff vectors*:** $v_i(X_n)$ — # deterministic vector feedback
- ***Pure payoff vectors*:** $v_i(a_n)$ — # stochastic vector feedback
- ***Bandit / Payoff-based*:** $u_i(a_n)$ — # stochastic scalar feedback

Example (RPS)



- Player i: play $x_i = (1/2, 1/3, 1/6)$ $\rightsquigarrow$ draw R
- Player $-i$: play $x_i = (1/3, 1/3, 1/3)$ $\rightsquigarrow$ draw P

Full feedback (mixed payoff vectors)

$v_i(x_i; x_{-i})$ 0 0 0

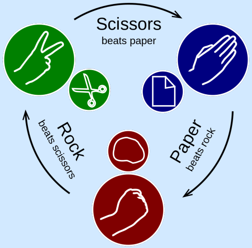
The feedback process

Types of feedback

From best to worst (more to less info):

- ***Mixed payoff vectors***: $v_i(X_n)$ — # deterministic vector feedback
- ***Pure payoff vectors***: $v_i(a_n)$ — # stochastic vector feedback
- ***Bandit / Payoff-based:*** $u_i(a_n)$ — # stochastic scalar feedback

Example (RPS)



- Player i: play $x_i = (1/2, 1/3, 1/6)$ $\rightsquigarrow$ draw R
- Player $-i$: play $x_i = (1/3, 1/3, 1/3)$ $\rightsquigarrow$ draw P

Realization-based feedback (pure payoff vectors)

$v_i(\mathrm{R};\mathrm{P})$ -1 0 1

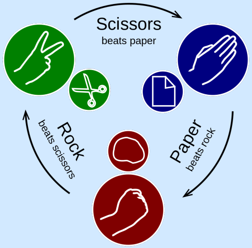
The feedback process

Types of feedback

From best to worst (more to less info):

- ***Mixed payoff vectors***: $v_i(X_n)$ — # deterministic vector feedback
- ***Pure payoff vectors***: $v_i(a_n)$ — # stochastic vector feedback
- ***Bandit / Payoff-based:*** $u_i(a_n)$ — # stochastic scalar feedback

Example (RPS)



- Player i: play $x_i = (1/2, 1/3, 1/6)$ $\rightsquigarrow$ draw R
- Player $-i$: play $x_i = (1/3, 1/3, 1/3)$ $\rightsquigarrow$ draw P

Bandit feedback (payoff-based)

$u_i(\mathrm{R};\mathrm{P})$ -1 ✗   ✗

The feedback process

Types of feedback

From best to worst (more to less info):

- ***Mixed payoff vectors***: $v_i(X_n)$ # deterministic vector feedback
- ***Pure payoff vectors***: $v_i(a_n)$ # stochastic vector feedback
- ***Bandit / Payoff-based:*** $u_i(a_n)$ # stochastic scalar feedback

Features:

- ***Vector*** (mixed / pure payoff vecs) vs. ***Scalar*** (bandit)
- ***Deterministic*** (mixed payoff vecs) vs. ***Stochastic*** (pure payoff vecs, bandit)

☞ Randomness defined relative to ***history of play*** $\mathcal{F}_n := \mathcal{F}(X_1, \dots, X_n)$

☞ Other feedback models also possible (noisy / delayed observations,...)

From payoffs to payoff vectors

How to estimate the payoff $u_i(a_i; a_{-i,n})$ of an unplayed action $a_i \neq a_{i,n}$?

From payoffs to payoff vectors

How to estimate the payoff $u_i(a_i; a_{-i,n})$ of an unplayed action $a_i \neq a_{i,n}$?

Definition (Importance weighted estimators)

The ***importance weighted estimator*** of a vector $v \in \mathbb{R}^{\mathcal{A}}$ relative to a mixed strategy $x \in \Delta(\mathcal{A})$ is defined as

$$\hat{v}_a = \frac{\mathbb{1}_a}{x_a} v_a = \begin{cases} v_a / x_a & \text{if } a \text{ is drawn} \\ 0 & \text{otherwise} \end{cases} \tag{IWE}$$

From payoffs to payoff vectors

How to estimate the payoff $u_i(a_i; a_{-i,n})$ of an unplayed action $a_i \neq a_{i,n}$?

Definition (Importance weighted estimators)

The ***importance weighted estimator*** of a vector $v \in \mathbb{R}^{\mathcal{A}}$ relative to a mixed strategy $x \in \Delta(\mathcal{A})$ is defined as

$$\hat{v}_a = \frac{\mathbb{1}_a}{x_a} v_a = \begin{cases} v_a / x_a & \text{if } a \text{ is drawn} \\ 0 & \text{otherwise} \end{cases} \tag{IWE}$$

Statistical properties of (IWE)

- ***Unbiased:***

$$\mathbb{E}_x[\hat{v}_a] = v_a$$

- ***Second moment:***

$$\mathbb{E}_x[\hat{v}_a^2] = \frac{v_a^2}{x_a}$$

The oracle model

Definition (Black-box oracle)

A ***stochastic first-order oracle*** of $v(X_n)$ is a random (or deterministic) vector of the form

$$\hat{v}_n = v(X_n) + U_n + b_n \tag{SFO}$$

where U_n is **zero-mean** and $b_n = \mathbb{E}[\hat{v}_n \mid \mathcal{F}_n] - v(X_n)$ is the **bias** of $\hat{v}_n$.

The oracle model

Definition (Black-box oracle)

A ***stochastic first-order oracle*** of $v(X_n)$ is a random (or deterministic) vector of the form

$$\hat{v}_n = v(X_n) + U_n + b_n \tag{SFO}$$

where U_n is **zero-mean** and $b_n = \mathbb{E}[\hat{v}_n \mid \mathcal{F}_n] - v(X_n)$ is the **bias** of $\hat{v}_n$.

Examples

- Mixed payoff vectors: $\hat{v}_{i,n} = v_i(X_n)$ — # noise $U_n = 0$; bias $b_n = 0$
- Pure payoff vectors: $\hat{v}_{i,n} = v_i(a_n)$ — # noise $U_n = \mathcal{O}(1)$; bias $b_n = 0$
- Payoff-based: $\hat{v}_{i,n} = \dfrac{u_i(a_n)}{\mathbb{P}(a_{i,n} = a_i)} e_{a_{i,n}}$ — # noise $U_n = \mathcal{O}(1/\min_{a_i} x_{ia_i,n})$; bias $b_n = 0$

Exponential weights redux

Algorithm Exponential weights in discrete time (ExpWeight)

Require: finite game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$; stochastic first-order oracle $\hat{v}$

Initialize: $Y_i \in \mathbb{R}^{\mathcal{A}_i}$, $i = 1, \dots, N$

for all $n = 1, 2, \dots$ all players $i \in \mathcal{N}$ **do simultaneously**

set $X_{i,n} \propto \exp(Y_{i,n})$ # mixed strategy

play $a_{i,n} \sim X_{i,n}$ # choose action

get $\hat{v}_{i,n} \in \mathbb{R}^{\mathcal{A}_i}$ # receive feedback

set $Y_{i,n+1} \leftarrow Y_{i,n} + \gamma_n \hat{v}_{i,n}$ # update scores

end for

Basic idea:

- Score actions by aggregating payoff vector estimates provided by oracle
- Choose actions with probability exponentially proportional to their scores
- Rinse / repeat

Example 1: ExpWeight with mixed payoff vector observations

If players observe ***mixed payoff vectors:***

$$\hat{v}_{i,n} = v_i(X_{i,n}; X_{-i,n})$$

Oracle features:

- ***Deterministic:*** no randomness!
- ***Bias:*** $B_n = 0$
- ***Variance:*** $\sigma_n = 0$
- ***Second moment:*** $M_n = \mathcal{O}(1)$

☞ Also known as **Multiplicative Weights Update** ➹ Arora et al. (2012)

Example 2: ExpWeight with pure payoff vector observations

If players observe ***pure payoff vectors:***

$$\hat{v}_{i,n} = v_i(a_{i,n}; a_{-i,n})$$

Oracle features:

- ***Stochastic:*** random action selection
- ***Bias:*** $B_n = 0$
- ***Variance:*** $\sigma_n = \mathcal{O}(1)$
- ***Second moment:*** $M_n = \mathcal{O}(1)$

☞ Also known as **Hedge**

➻ Auer et al. (1995), Auer et al. (2002)

Example 3: ExpWeight with bandit feedback

If players observe ***realized payoffs:***

$$\hat{v}_{i,n} = \frac{u_i(a_{i,n}; a_{-i,n})}{\mathbb{P}(a_{i,n} = a_i)} e_{a_{i,n}}$$

Oracle features:

- ***Stochastic:*** random action selection
- ***Bias:*** $B_n = 0$
- ***Variance:*** $\sigma_n = \mathcal{O}(1/X_{ia_{i,n}})$
- ***Second moment:*** $M_n = \mathcal{O}(1/X_{ia_{i,n}})$

☞ Also known as **EXP3**

➻ Auer et al. (1995), Auer et al. (2002)

Example 4: ExpWeight with bandit feedback

If players observe ***realized payoffs:***

$$\hat{v}_{i,n} = \frac{u_i(a_{i,n}; a_{-i,n})}{\mathbb{P}(a_{i,n} = a_i)} e_{a_{i,n}}$$

Oracle features:

- ***Stochastic:*** random action selection
- ***Explicit exploration:*** draw $a_{i,n} \sim X_{i,n}$ with prob. $1 - \delta_n$, otherwise uniformly
- ***Bias:*** $B_n = \mathcal{O}(\delta_n)$
- ***Variance:*** $\sigma_n = \mathcal{O}(1/\delta_n^2)$
- ***Second moment:*** $M_n = \mathcal{O}(1/\delta_n^2)$

☞ Also known as **EXP3 WITH EXPLICIT EXPLORATION**

➻ Shalev-Shwartz (2011), Lattimore & Szepesvári (2020)

Example 5: Optimistic ExpWeight

If players are ***optimistic***:[1]

Rakhlin & Sridharan (2013)

$$\hat{v}_{i,n} = v_i(X_{i,n+1/2}; X_{-i,n+1/2})$$

Oracle features:

- ***Deterministic:*** no randomness
- ***Bias:*** $B_n = v(X_{n+1/2}) - v(X_n) = \mathcal{O}(\gamma_n)$
- ***Variance:*** $\sigma_n = 0$
- ***Second moment:*** $M_n = \mathcal{O}(1)$

[1] Feedback obtained via the sequence

$$Y_{n+1/2} = Y_n + \gamma_n v_n(X_{n-1/2}) \qquad X_{i,n+1/2} \propto \exp(Y_{i,n+1/2}) \qquad Y_{n+1} = Y_n + \gamma_n v(X_{n+1/2})$$

Follow the regularized leader

Follow the regularized leader

$$\begin{aligned} Y_{i,n+1} &= Y_{i,n} + \gamma_n \hat{v}_{i,n} \\ X_{i,n+1} &= Q_i(Y_{i,n+1}) \equiv \arg\max_{x_i \in \mathcal{X}_i} \{ \langle Y_{i,n+1}, x_i \rangle - h_i(x_i) \} \end{aligned} \tag{FTRL}$$

➻ Shalev-Shwartz & Singer (2006), Nesterov (2009)

- Generalized version of "follow the regularized leader"
- $\gamma_n > 0$ is the method's ***step-size*** # To be specialized later
- $\hat{v}_{i,n}$ is an stochastic first-order oracle (SFO) model for $v_i(X_n)$ # To be specialized later
- Every player's ***regularizer*** $h_i \colon \mathcal{X}_i \to \mathbb{R}$ is continuous on $\mathcal{X}_i$, differentiable on ri $\mathcal{X}_i$, and strongly convex on $\mathcal{X}_i$

$$h_i(x_i') \geq h_i(x_i) + \langle \nabla h_i(x_i), x_i' - x_i \rangle + (K_i/2)\|x_i' - x_i\|^2$$

Evolution of mixed strategies: Examples

What does the sequence of play look like?

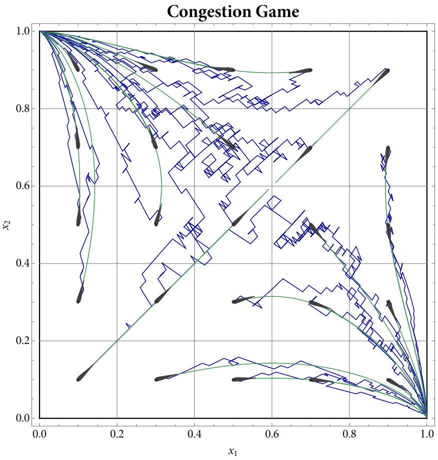


Figure. ExpWeight with constant step-size

Evolution of mixed strategies: Examples

What does the sequence of play look like?

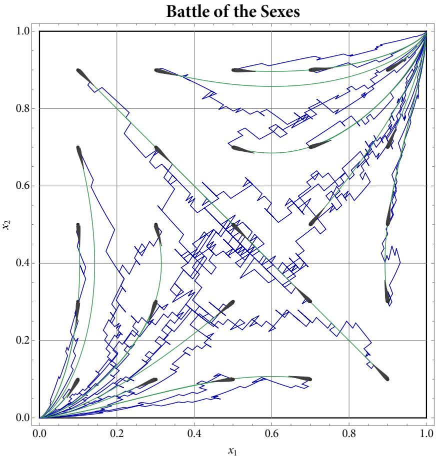


Figure. ExpWeight with constant step-size

Evolution of mixed strategies: Examples

What does the sequence of play look like?

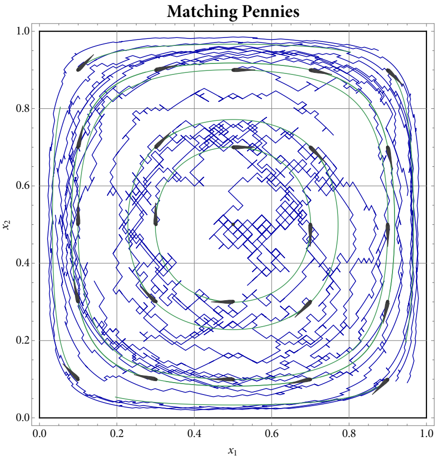


Figure. ExpWeight with constant step-size

Evolution of mixed strategies: Examples

What does the sequence of play look like?

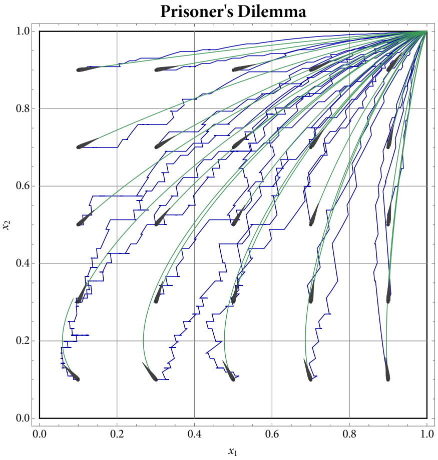


Figure. ExpWeight with constant step-size

Evolution of mixed strategies: Examples

What does the sequence of play look like?

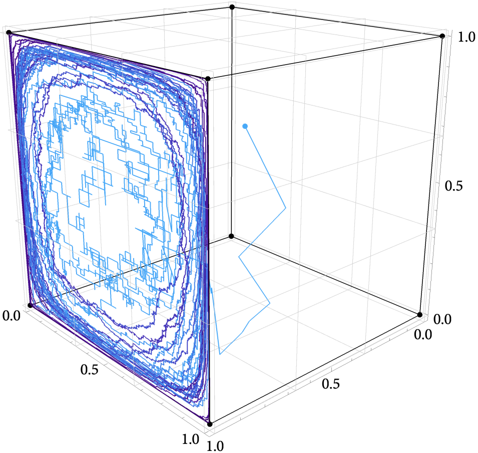


Figure. ExpWeight with constant step-size

Evolution of mixed strategies: Examples

What does the sequence of play look like?

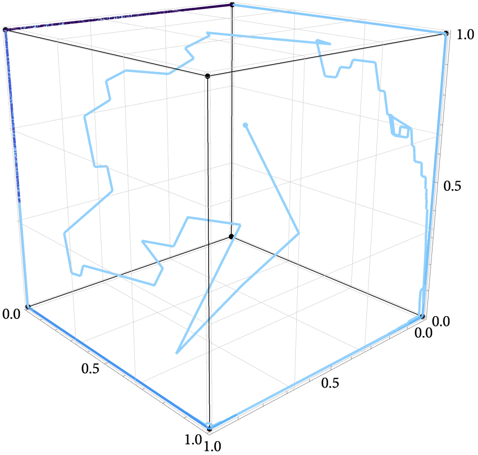


Figure. ExpWeight with constant step-size

Notions of stability

Definition (Stochastic stability)

$x^* \in \mathcal{X}$ is ***stochastically stable*** under X_n if, for every confidence level $\delta > 0$ and every neighborhood $\mathcal{U}$ of x^*, there exists a neighborhood $\mathcal{U}_1$ of x^* such that

$$\mathbb{P}(X_n \in \mathcal{U} \text{ for all } n = 1, 2, \dots \mid X_1 \in \mathcal{U}_1) \geq 1 - \delta$$

Intuition: with high probability, if X_n starts near x^*, it remains nearby

Notions of stability

Definition (Stochastic stability)

$x^* \in \mathcal{X}$ is ***stochastically stable*** under X_n if, for every confidence level $\delta > 0$ and every neighborhood $\mathcal{U}$ of x^*, there exists a neighborhood $\mathcal{U}_1$ of x^* such that

$$\mathbb{P}(X_n \in \mathcal{U} \text{ for all } n = 1, 2, \dots \mid X_1 \in \mathcal{U}_1) \geq 1 - \delta$$

Intuition: with high probability, if X_n starts near x^*, it remains nearby

Definition (Stochastic asymptotic stability)

- $x^* \in \mathcal{X}$ is ***attracting*** if, for every confidence level $\delta > 0$, there exists a neighborhood $\mathcal{U}_1$ of x^* such that

$$\mathbb{P}(X_n \to x^* \; as \; n \to \infty \mid X_1 \in \mathcal{U}_1) \geq 1 - \delta$$

- $x^* \in \mathcal{X}$ is ***stochastically asymptotically stable*** if it is stochastically stable and attracting.

Intuition: with high probability, if X_n starts near x^* then, it remains nearby and eventually converges to x^*

The long-run behavior of regularized learning

Theorem

⚠ **Assume:** *All players run (FTRL) with step-size γ_n and oracle parameters b_n (bias) and U_n (noise) such that:*

(A1) $\gamma_n = \gamma/n^p$ for some $p \in (0,1]$ ✓ *ok for all models*

(A2) $b_n = \mathcal{O}(1/n^b)$ for some $b > 0$ ✓ *ok for all models*

(A3) $\mathbb{E}[\|U_n\|^q] = \mathcal{O}(1/n^r)$ for some $q > 2$, $r < 1/2$ ✓ *ok for all models*

The long-run behavior of regularized learning

Theorem

⚠ **Assume:** *All players run* (FTRL) *with step-size* γ_n *and oracle parameters* b_n *(bias) and* U_n *(noise) such that:*

(A1) $\gamma_n = \gamma/n^p$ *for some* $p \in (0,1]$ ✓ *ok for all models*

(A2) $b_n = \mathcal{O}(1/n^b)$ *for some* $b > 0$ ✓ *ok for all models*

(A3) $\mathbb{E}[\|U_n\|^q] = \mathcal{O}(1/n^r)$ *for some* $q > 2$, $r < 1/2$ ✓ *ok for all models*

☞ **Then:** *the sequence* X_n *generated by* (FTRL) *enjoys the following properties*

(P1) *If* X_n *converges, its limit is a Nash equilibrium* ➺ *M & Zhou (2019)*

(P2) *If* x^* *is stochastically stable, it is a Nash equilibrium* ➺ *Giannou et al. (2021)*

(P3) x^* *is stochastically asymptotically stable if and only if it is a strict Nash equilibrium* ➺ *Giannou et al. (2021)*

(P4) *If* $p > 1/2$ *and* $\mathcal{G}$ *is a congestion game, then* X_n *converges to a Nash equilibrium (a.s.)* ➺ *Héliou et al. (2017)*

Rate of convergence

Theorem (Giannou et al., 2021)

⚠ **Assume:** *All players run* ExpWeight *with step-size γ_n and oracle parameters b_n and U_n as before*

☞ **Then:** *if x^* is a strict Nash equilibrium and X_n converges to x^*, we have*

$$\|X_n - x^*\|_1 \leq \sum_{a \notin \operatorname{supp}(x^*)} \exp\left(A - B \sum_{k=1}^{n} \gamma_k\right)$$

where $A, B > 0$ are positive constants.

Universal convergence guarantees

Can we characterize the limiting behavior of (FTRL)?

Limit sets

The ***limit set*** of a sequence X_n, $n = 1, 2, \dots$, is the set of all its limit points, i.e.,

$$\mathcal{L}(X) := \bigcap_{n=1}^{\infty} \operatorname{cl}\{X_k : k \geq n\} = \{x \in \mathcal{X} : X_{n_k} \to x \text{ for some sequence } n_k \to \infty\}$$

Equivalently, $\mathcal{L}(X)$ is the smallest subset of $\mathcal{X}$ such that $\operatorname{dist}(\mathcal{L}, X_n) \to 0$ as $n \to \infty$

Universal convergence guarantees

Can we characterize the limiting behavior of (FTRL)?

Limit sets

The ***limit set*** of a sequence X_n, $n = 1, 2, \dots$, is the set of all its limit points, i.e.,

$$\mathcal{L}(X) := \bigcap_{n=1}^{\infty} \operatorname{cl}\{X_k : k \geq n\} = \{x \in \mathcal{X} : X_{n_k} \to x \text{ for some sequence } n_k \to \infty\}$$

Equivalently, $\mathcal{L}(X)$ is the smallest subset of $\mathcal{X}$ such that $\operatorname{dist}(\mathcal{L}, X_n) \to 0$ as $n \to \infty$

Theorem (Boone & M, 2022)

⚠ ***Assume:*** *All players run* ExpWeight *with step-size γ_n and oracle parameters b_n and U_n as before.*

☞ ***Then:*** *With probability 1, the limit set $\mathcal{L}$ of* (FTRL) *is characterized by the following properties:*

- **Minimality:** *$\mathcal{L}$ does not contain any proper attractors*
- **Resilience:** *every deviation x from $\mathcal{L}$ is unilaterally nullified by some element x^* of $\mathcal{L}$, i.e.,*

$$u_i(x^*) \geq u_i(x_i; x^*_{-i}) \quad \textit{for all } i \in \mathcal{N}$$

Overview

I. Learning in continuous time

- Nash equilibrium $\Longrightarrow$ stationarity
- Lyapunov stability $\Longrightarrow$ equilibrium
- Asymptotic stability $\Longleftrightarrow$ strict equilibrium
- Min-max games $\Longrightarrow$ Poincaré recurrence
- Limit sets $\Longleftrightarrow$ minimally resilient

II. Learning in discrete time

- ✗ Depends on feedback, step-size, … # stochastic ≠ deterministic
- ✗ Nash equilibrium $\not\Longrightarrow$ stationarity
- ✓ Lyapunov stability $\Longrightarrow$ equilibrium
- ✓ Asymptotic stability $\Longleftrightarrow$ strict equilibrium # mixed equilibria are ***unstable***
- ✗ Min-max games $\not\Longrightarrow$ Poincaré recurrence # convergence to the boundary
- ✓ Limit sets $\Longrightarrow$ minimally resilient # converse does not hold

Open issues

- Adaptive step-size / learning rate? # challenging analysis
- Robustness to delays / corruptions / …
- Learning in continuous games?

References I

[1] Arora, S., Hazan, E., and Kale, S. The multiplicative weights update method: A meta-algorithm and applications. *Theory of Computing*, 8(1): 121–164, 2012.

[2] Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, 1995.

[3] Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1): 48–77, 2002.

[4] Boone, V. and Mertikopoulos, P. From equilibrium to resilience: Universal guarantees for the long-run behavior of learning in games, 2022.

[5] Flokas, L., Vlatakis-Gkaragkounis, E. V., Lianeas, T., Mertikopoulos, P., and Piliouras, G. No-regret learning and mixed Nash equilibria: They do not mix. In *NeurIPS '20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020.

[6] Giannou, A., Vlatakis-Gkaragkounis, E. V., and Mertikopoulos, P. Survival of the strictest: Stable and unstable equilibria under regularized learning with partial information. In *COLT '21: Proceedings of the 34th Annual Conference on Learning Theory*, 2021.

[7] Giannou, A., Vlatakis-Gkaragkounis, E. V., and Mertikopoulos, P. The convergence rate of regularized learning in games: From bandits and uncertainty to optimism and beyond. In *NeurIPS '21: Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2021.

[8] Héliou, A., Cohen, J., and Mertikopoulos, P. Learning with bandit feedback in potential games. In *NIPS '17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.

[9] Hofbauer, J. and Sigmund, K. Evolutionary game dynamics. *Bulletin of the American Mathematical Society*, 40(4):479–519, July 2003.

[10] Hofbauer, J., Sorin, S., and Viossat, Y. Time average replicator and best reply dynamics. *Mathematics of Operations Research*, 34(2):263–269, May 2009.

References II

[11] Lattimore, T. and Szepesvári, C. *Bandit Algorithms*. Cambridge University Press, Cambridge, UK, 2020.

[12] Littlestone, N. and Warmuth, M. K. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

[13] Mertikopoulos, P. and Zhou, Z. Learning in games with continuous action sets and unknown payoff functions. *Mathematical Programming*, 173 (1-2):465–507, January 2019.

[14] Mertikopoulos, P., Papadimitriou, C. H., and Piliouras, G. Cycles in adversarial regularized learning. In *SODA '18: Proceedings of the 29th annual ACM-SIAM Symposium on Discrete Algorithms*, 2018.

[15] Nash, J. F. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences of the USA*, 36:48–49, 1950.

[16] Nesterov, Y. Primal-dual subgradient methods for convex problems. *Mathematical Programming*, 120(1):221–259, 2009.

[17] Rakhlin, A. and Sridharan, K. Optimization, learning, and games with predictable sequences. In *NIPS '13: Proceedings of the 27th International Conference on Neural Information Processing Systems*, 2013.

[18] Rustichini, A. Optimal properties of stimulus-response learning models. *Games and Economic Behavior*, 29(1-2):244–273, 1999.

[19] Shalev-Shwartz, S. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

[20] Shalev-Shwartz, S. and Singer, Y. Convex repeated games and Fenchel duality. In *NIPS' 06: Proceedings of the 19th Annual Conference on Neural Information Processing Systems*, pp. 1265–1272. MIT Press, 2006.

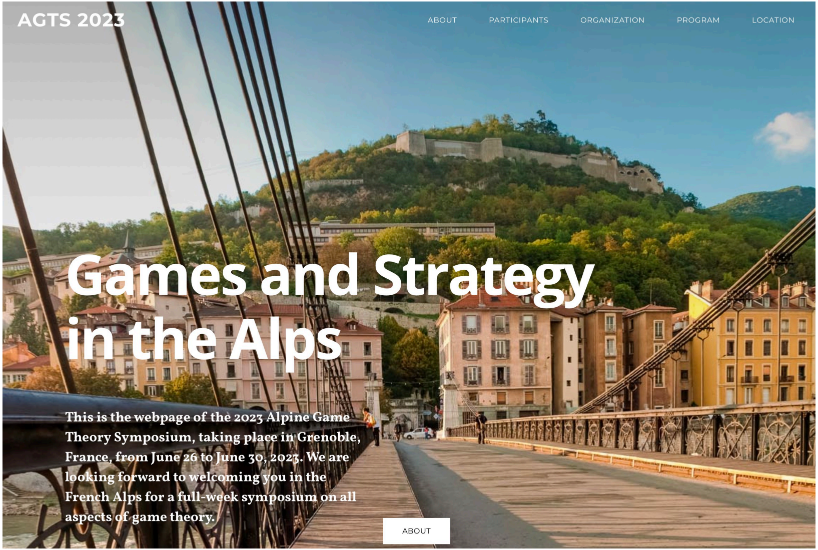
[21] Sorin, S. Exponential weight algorithm in continuous time. *Mathematical Programming*, 116(1):513–528, 2009.

[22] Stampacchia, G. Formes bilineaires coercitives sur les ensembles convexes. *Comptes Rendus Hebdomadaires des Séances de l'Académie des Sciences*, 1964.

[23] Taylor, P. D. and Jonker, L. B. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 40(1-2):145–156, 1978.

Outline

Alpine Game Theory Symposium



 # If you like mountains and/or games $\rightsquigarrow$ `https://agts-2023.weebly.com`

Games, Learning, and Networks

Games, Learning, and Networks

(03 Apr 2023–21 Apr 2023)

Organizing Committee

Co-chairs

- Georgios Piliouras (Singapore University of Technology and Design)
- Marco Scarsini (Libera Università Internazionale degli Studi Sociali Guido Carli)

In the tropics $\rightsquigarrow$ `https://ims.nus.edu.sg/events/games-learning-and-networks/`