Learning in Games with Dynamic Population

Éva Tardos

Cornell, Computer Science

Based on joint work with Thodoris Lykouris, Vasilis Syrgkanis, Dylan Foster and Karthik Sridharan

Large population games: traffic routing

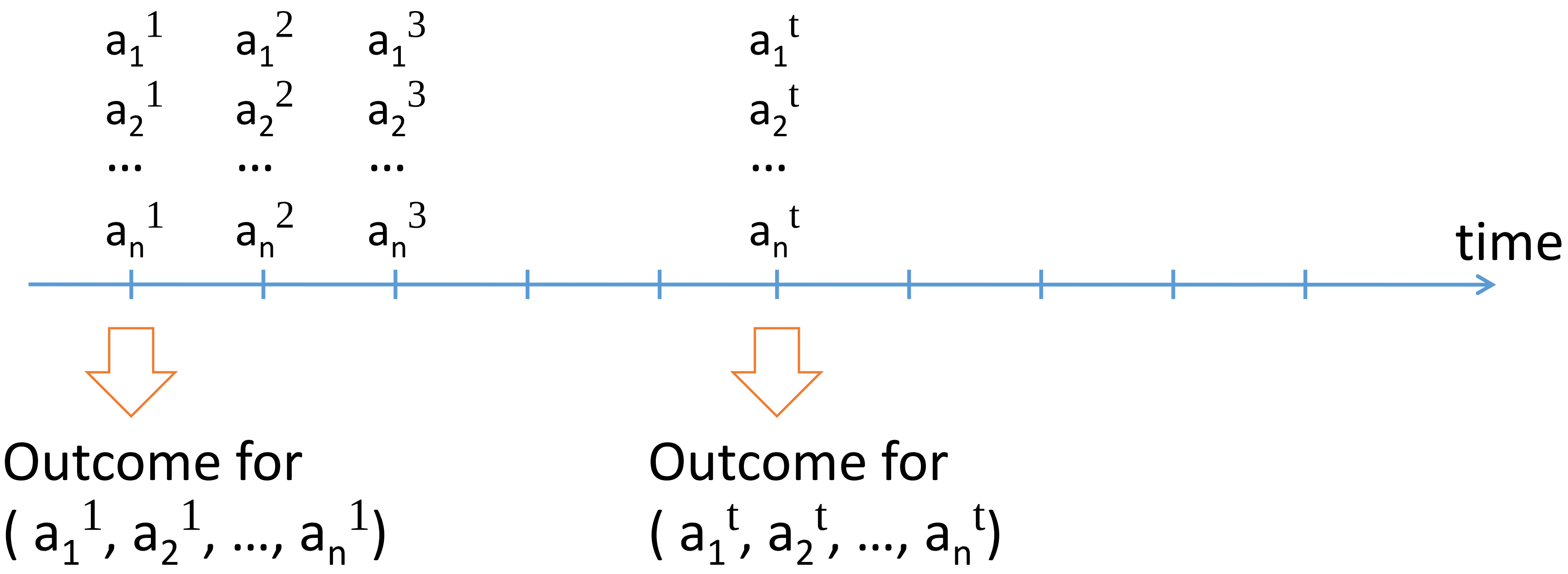
- Traffic subject to congestion delays
- cars and packets follow shortest path
- Congestion game =cost (delay) depends only on congestion on edges

Traffic streams change

e.g., popular sites may change

Changes in system setup

Repeated games



- Player’s value/cost additive over periods, while playing
- Players try to learn what is best from past data

What can we say about the outcome?

How long do they have to stay to ensure OK social welfare?

Result: routing, limit for very small users

Theorem (Roughgarden-T'02):

In any network with continuous, non-decreasing cost functions and small users

cost of Nash with rates r_i for all i	$\leq$	cost of opt with rates $2r_i$ for all i

Nash equilibrium: stable solution where no player had incentive to deviate.

$$\text{Price of Anarchy} = \frac{\text{cost of worst Nash equilibrium}}{\text{“socially optimum” cost}}$$

Examples of price of anarchy bounds

- Monotone increasing congestion costs

  Nash cost ≤ opt of double traffic rate (Roughgarden-T'02)

- affine congestion cost (Roughgarden-T'02) 4/3 price of anarchy

- Atomic game (players with >0 traffic) with linear delay (Awerbuch-Azar-Epstein & Christodoulou-Koutsoupias'05)

  $\Rightarrow$ 2.5 price of anarchy

These bounds are tight

Price of anatrchy in auctions

- First price is auction Hassidim, Kaplan, Mansour, Nisan EC'11)
  Price of anarchy 1.58…
- All pay auction price of anarchy 2
- First position auction (GFP) is price of anarchy 2
- Variants with second price (see also Christodoulou, Kovacs, Schapira ICALP'08) price of anarchy 2

Other applications include:

- public goods
- Fair sharing (Kelly, Johari-Tsitsiklis) price of anarchy 1.33
- Walrasian Mechanism (Babaioff, Lucier, Nisan, and Paes Leme EC'13)

Repeated game that is (slowly) changing [Lykouris, Syrgkanis, T.]

Dynamic population model:

At each step t each player i

is replaced with an arbitrary new player with probability p

In a population of N players, each step, Np players replaced in expectation

- Population changes all the time: need to adjust!
- players stay long enough to be able to learn (1/p steps)

Learning in Repeated Game

- What is learning?
- Does learning lead to finding Nash equilibrium?

Robinson'51:

- fictitious play = best respond to past history of other players

Goal: "pre-play" as a way to learn to play Nash.

Outcome of Fictitious Play in Repeated Game

- Does learning lead to finding Nash equilibrium?

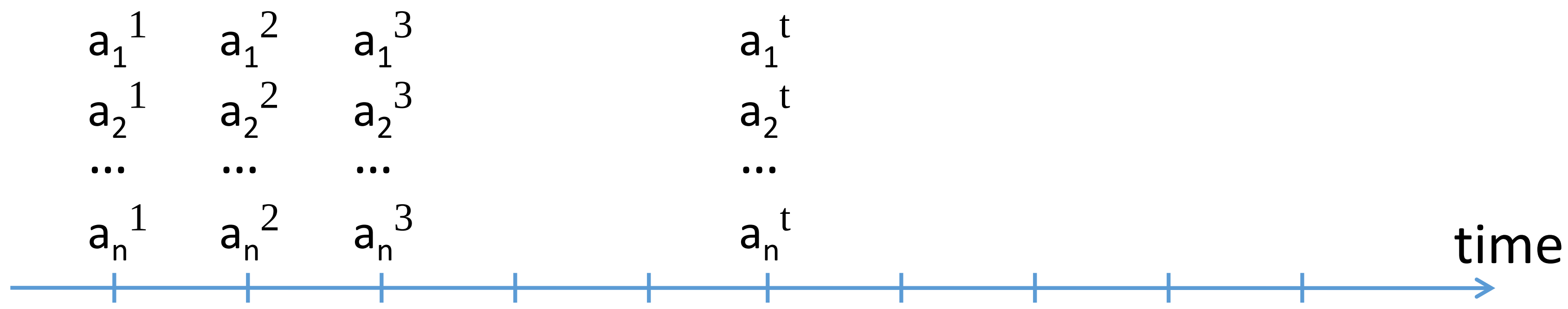
  mostly not

Theorem: Marginal distribution of each player actions converges to Nash in

Robinson’51: In two player 0-sum games

Miyasawa’61: In generic payoff 2 by 2 games

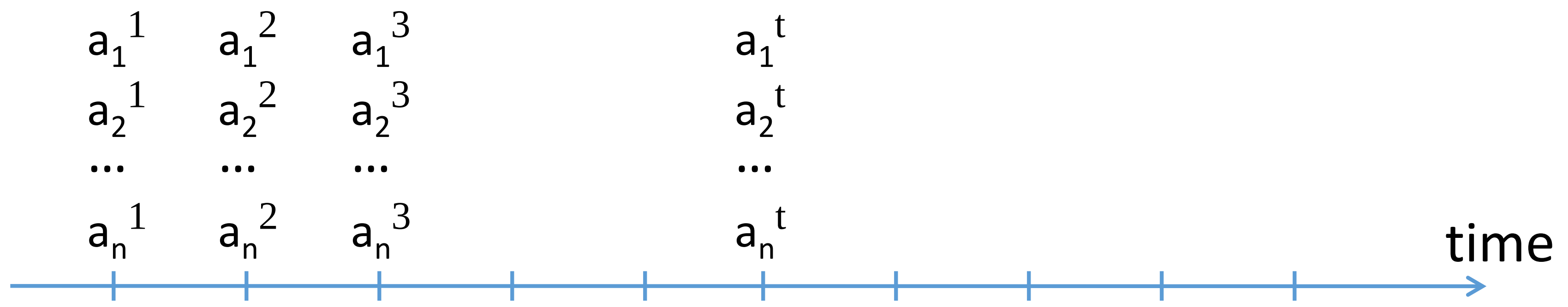
Learning outcome



Maybe here they don't know how to play, who are the other players, …

By here they have a better idea…

No-regret without stability: learning



No regret: for any fixed action x: (cost $\in [0,1]$):

$$\sum_t cost_i(a^t) \leq \sum_t cost_i(x, a_{-i}^t) \quad + error$$

regret

error $\leq \sqrt{T}$ (if $o(T)$ called no-regret)

Outcome of no-regret learning in a fixed game

Limit distribution σ of play (action vectors a=$(a_1, a_2, \ldots, a_n)$)

- all players i have no regret for all strategies x

$$E_{a\sim\sigma}\left(cost_i(a)\right) \geq E_{a\sim\sigma}(cost_i(x, a_{-i}))$$

Hart & Mas-Colell: Long term average play is (coarse) correlated equilibrium

Players update independently, but correlate on shared history

Today: approximate no-regret

For any fixed action x (with d options) :

$$\sum_t cost_i(a^t) \leq \sum_t cost_i(x, a_{-i}^t) + \sqrt{T \log d}$$

In fact, much better bound applies!

Foster, Li, Lykouris, Sridharan, T NIPS'16

$$\sum_t cost_i(a^t) \leq (1+\epsilon) \sum_t cost_i(x, a_{-i}^t) + \frac{\log d}{\epsilon}$$

Same algorithms! MWU (Hedge), Regret Matching, etc.

T=time, d=# strategies

No-regret learning as a behavioral model?

- Er'ev and Roth'96

  lab experiments with 2 person coordination game
- Fudenberg-Peysakhovich EC'14

  lab experiments with seller-buyer game

  recency biased learning
- Nekipelov-Syrgkanis-Tardos EC'15

  Bidding data on Bing-Ad-Auctions
- Nisan-Noti WWW'17

  Lab experiment with ad-auction games

Quality of Learning Outcome

Price of Anarchy [Koutsoupias-Papadimitriou'99]

$$PoA = \max_{a\ Nash} \frac{cost(a)}{Opt}$$

Assuming **no-regret learners** in fixed game: [Blum, Hajiaghayi, Ligett, Roth'08, Roughgarden'09]

$$PoA = \lim_{T\to\infty} \frac{\sum_{t=1}^{T} cost(a^t)}{T\ Opt}$$

[Lykouris, Syrgkanis, T. 2016] dynamic population

$$PoA = \lim_{T\to\infty} \frac{\sum_{t=1}^{T} cost(a^t, v^t)}{\sum_{t=1}^{T} Opt(v^t)}$$

where v^t is the vector of player types at time t

Proof Technique: Smoothness (Roughgarden'09)

Consider optimal solution: player i does action a_i^* in optimum

Nash: $\mathrm{cost_i(a)} \leq cost_i\ (\boldsymbol{a_i^*}, a_{-i})$ (doesn't need to know a_i^*)

A game is (λ,μ)-smooth (λ > 0; μ< 1): if for all strategy vectors a

$$\sum_i cost_i(a) \leq \sum_i cost_i(a_i^*, a_{-i}) \leq \lambda\, OPT + \mu\, cost(a)$$

Then: A Nash equilibrium a has cost(a) $\leq \frac{\lambda}{1-\mu}$Opt

If Opt much cheaper than a, some player will want to deviate to a_i^*

Learning and price of anarchy

Use approx no-regret learning:

$$\sum_t cost_i(a^t) \leq (1+\epsilon)\sum_t cost_i\left(a_i^*, a_{-i}^t\right) + AR$$

A cost minimization game is (λ,μ)-smooth (λ > 0; μ< 1):

$$\sum_t \sum_i cost_i\left(a_i^*, a_{-i}^t\right) \leq \lambda \sum_t Opt + \mu \sum_t cost(a^t)$$

A approx. no-regret sequence a^t has

$$\frac{1}{T}\sum_t cost(a^t) \leq \frac{(1+\epsilon)\lambda}{1-(1+\epsilon)\mu}\text{Opt} + \frac{\text{n}}{\text{T}(1-(1+\epsilon)\mu)}\text{AR}$$

Note the convergence speed! $AR = \frac{\log d}{\epsilon}$, so error

$$\frac{\text{n}}{\text{T}} \cdot \frac{\log d}{\epsilon(1-(1+\epsilon)\mu)}$$

Foster, Li, Lykouris, Sridharan, T, NIPS'16

Learning in Dynamic Game: [Lykouris, Syrgkanis, T. ‘16]


Dynamic population model:

At each step t each player i

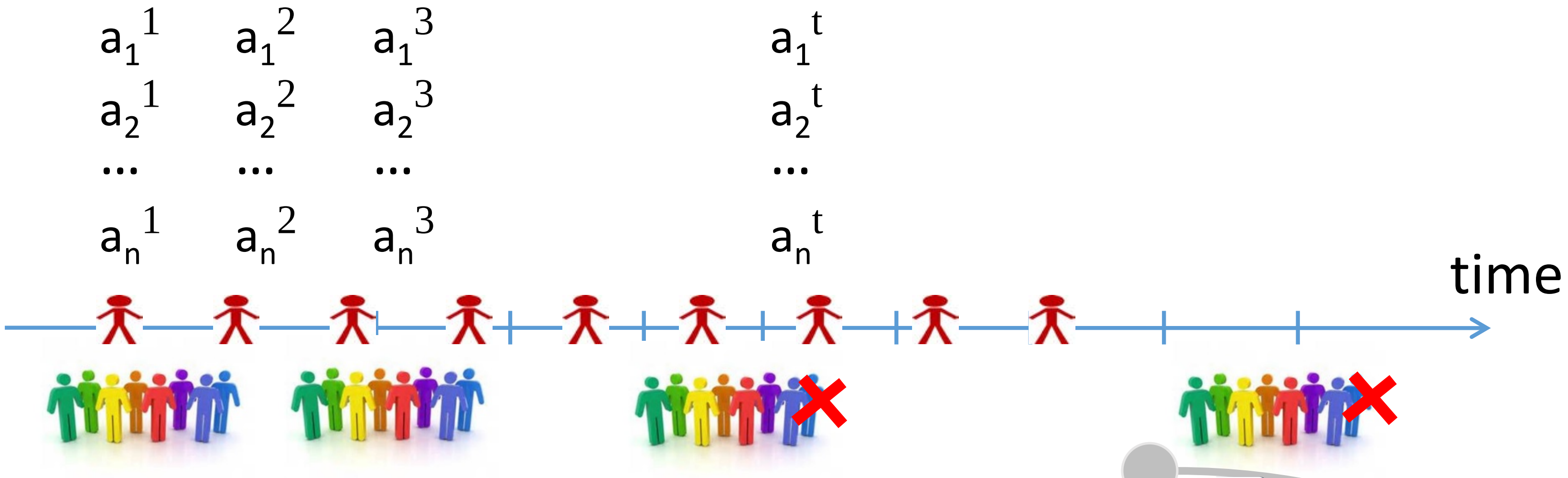
is replaced with an arbitrary new player with probability p

How should they learn from data?

No regret?

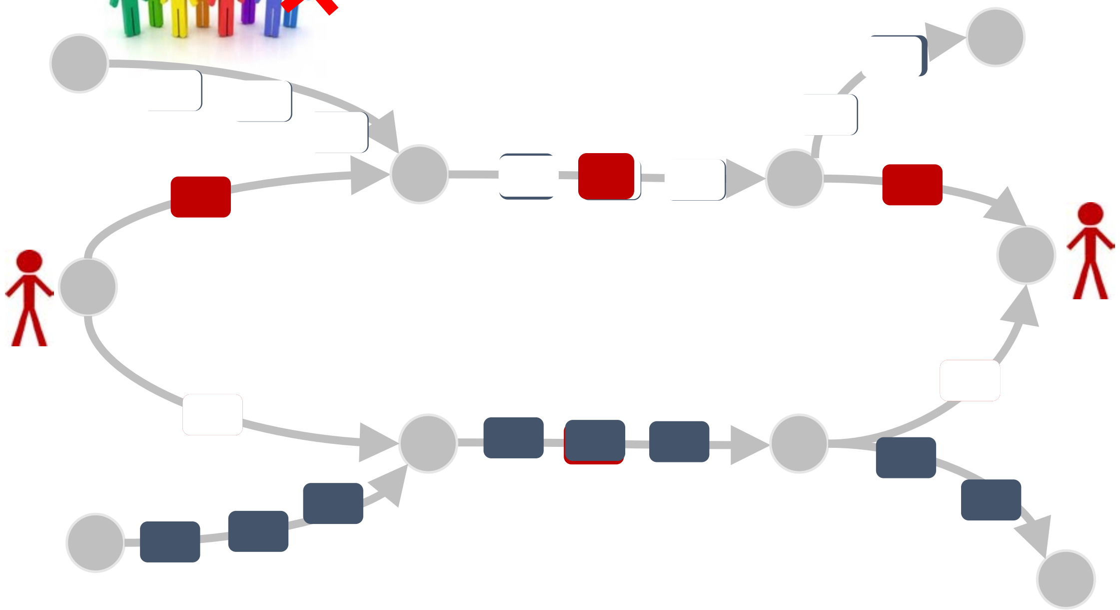
$$\sum_t cost_i(a^t) \leq (1+\epsilon) \sum_t cost_i(a_i^*, a_{-i}^t) + AR$$

Need for adaptive learning



Example routing

- Strategy = path
- Best "fixed" strategy in hindsight very weak in changing environment
- Learners should/can adapt to the changing environment



Adapting result to dynamic populations

Inequality we "wish to have"

$$\sum_t cost_i(a^t; v^t) \leq \sum_t cost_i(a_i^{*t}, a_{-i}^t; v^t)$$

where a_i^{*t} is the optimum strategy for the players at time t.

with stable population = no regret for a_i^*

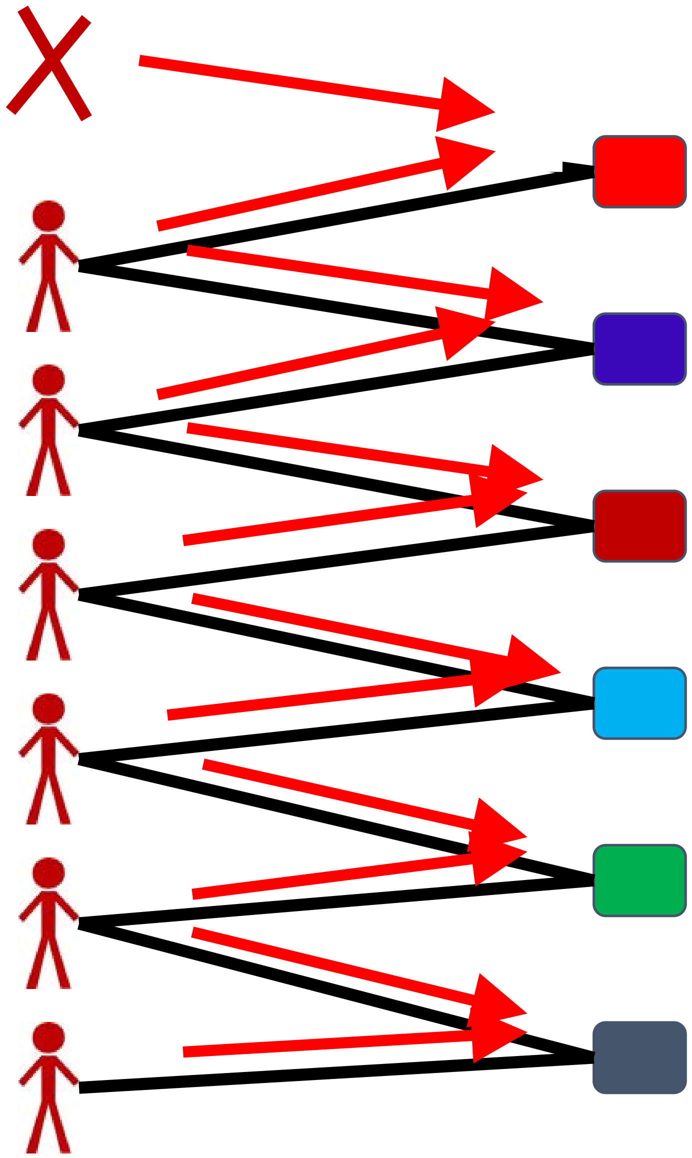
Too much to hope for in dynamic case:

- sequence a^{*t} of optimal solutions changes too much.
- No hope of learners not to learn this well!

Change in Optimum Solution

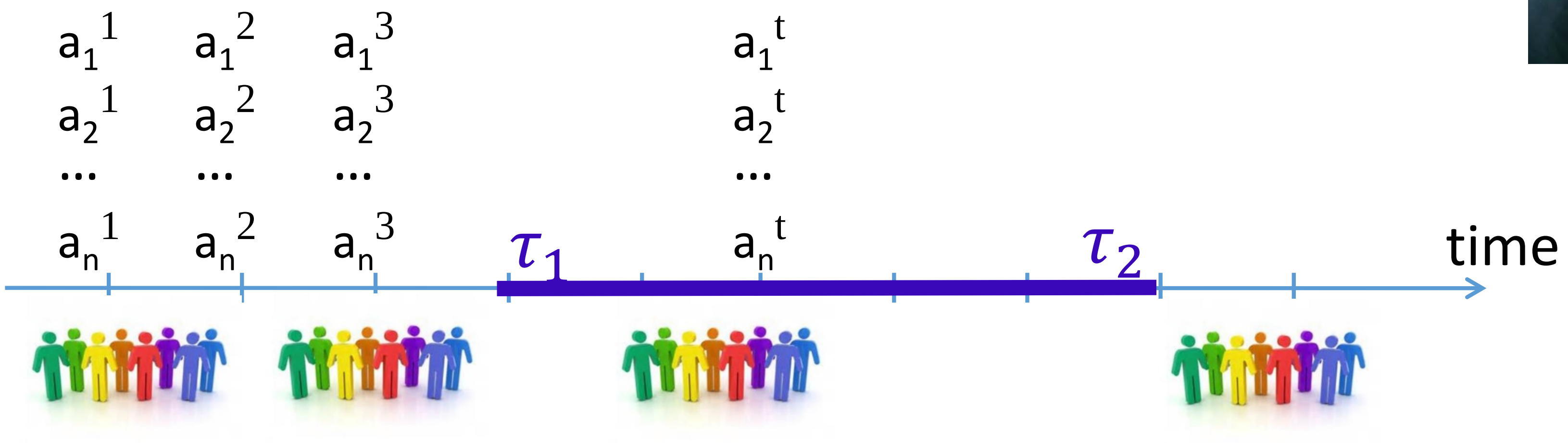
True optimum is too sensitive

- Example using matching
- The optimum solution
- One person leaving
- Can change the solution for everyone



- Np changes each step $\rightarrow$ No time to learn!! (we have p>>1/N)

Adaptive Learning



Theorem Approximate Regret [Foster,Li,Lykouris,Sridharan,T. NIPS'16]

for all player i, strategy x^t sequence that changes k times

$$\sum_t cost_i(a^t, v^t) \leq \sum_t (1+\epsilon)\, cost_i\left(x^t, a^t_{-i}; v^t\right) + O\left(\frac{k}{\epsilon} \log d\right)$$

Using any of MWU (Hedge), Regret Matching, etc. mixed with a bit of recency bias

Theorem (high level)

If a game satisfies a “smoothness property”

The welfare optimization problem admits an approximation algorithm whose outcome $\widetilde{a^*}$ is stable to changes in one player’s type

Then any adaptive learning outcome is approximately efficient

$$\text{PoA} = \lim_{T \to \infty} \frac{\sum_{t=1}^{T} cost(a^t, v^t)}{\sum_{t=1}^{T} Opt(v^t)} \text{ close to PoA}$$

Proof idea: use this approximate solution as $\widetilde{\boldsymbol{a}^*}$ in Price of Anarchy proof

With $\widetilde{\boldsymbol{a}^*}$ not changing much, learners have time to learn not to regret following $\widetilde{\boldsymbol{a}^*}$

Result (Lykouris, Syrgkanis, T'16) :

In many smooth games welfare close to Price of Anarchy even when the rate of change is high, $p \approx \frac{1}{\log n}$ with n players, assuming **adaptive** no-regret learners

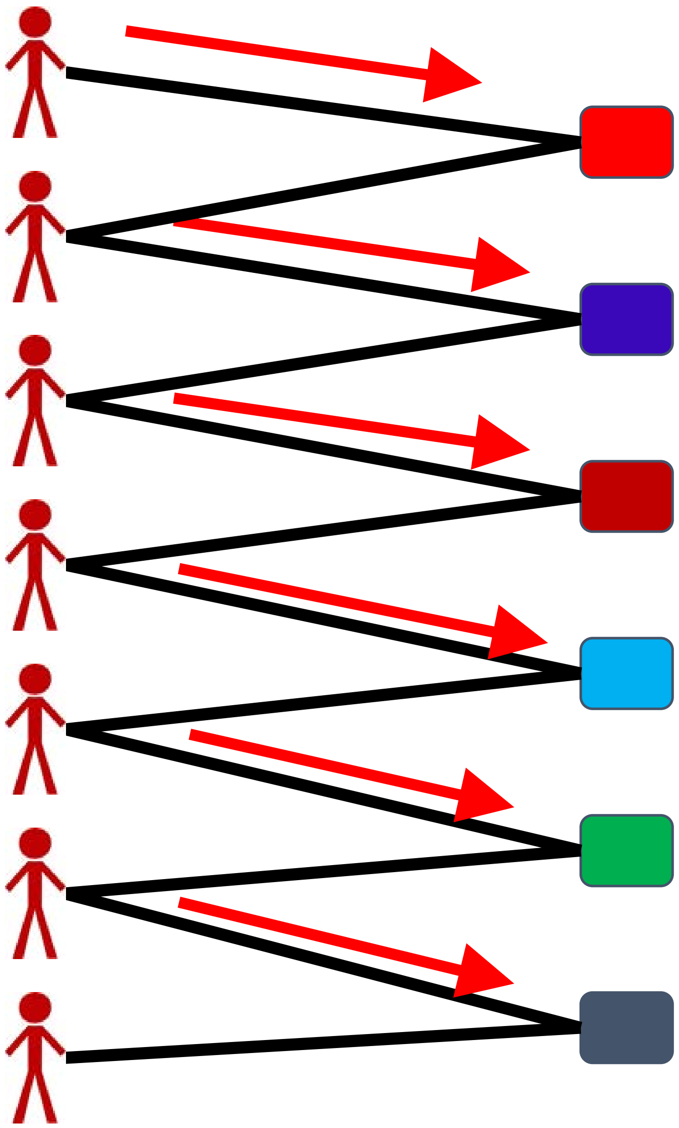
- Worst case change of player type $\Rightarrow$ need for learning players
- Bound $\alpha \cdot \beta \cdot \gamma$ depends on
  - α price of anarchy bound — as game gets large, goes to 1 in auctions, goes to 4/3 in linear congestion games
  - γ loss due to regret error — goes to 1 as $p \to 0$
  - β loss in opt for stable solutions — goes to 1 as $p \to 0$ & game is large

Stable ≈ Optimum in Matching

True optimum is too sensitive

- Use greedy allocation: assign large values first (loss of factor of 2)
- Use coarse approximation of value, e.g., power of 2 only
- Potential function argument:

  increase in log value of allocation only m $\log v_{max}$, decrease due to departures



Use Differential Privacy $\rightarrow$ Stable Solutions

Joint privacy [Kearns et al. ’14, Dwork et al. ‘06]

A randomized algorithm is jointly differentially private if

- when input from player **i** changes
- the probability of change in solution of players other than **i** is smaller than ϵ

- Turn a sequence of randomized solutions to a randomized sequence with small number of changes using Coupling Lemma
- and handling “failure probabilities” of private algorithms

Sample Application

Theorem 1. Matching markets (unit demand) [Lykouris, Syrgkanis,T'16]

if all values are $[\rho,1]$

$$\sum_t SW(a^t; v^t) \geq \frac{1}{4(1+\epsilon)} \sum_t \mathrm{OPT}(v^t)$$

with $p = O\left(\frac{\rho^2\epsilon^2}{polylog(m,1/\rho,1/\epsilon)}\right)$

assuming players use no regret learning with parameter $\epsilon > 0$

p = participant turnover probability

ρ= min item value

OPT(v^t) value of efficient outcome with player values v^t

SW(a^t, v^t) value of game outcome with player values v^t

Conclusions

Learning in games:

- Good way to adapt to opponents
- No need for common prior
- Takes advantage of opponent playing badly.

Learning players do well even in dynamic environments

- Stable approx. solution + good PoA bound $\Rightarrow$ good efficiency with dynamic population