6.S890: Topics in Multiagent Learning

Lecture 18

Fall 2023

Recent AI Breakthroughs

images

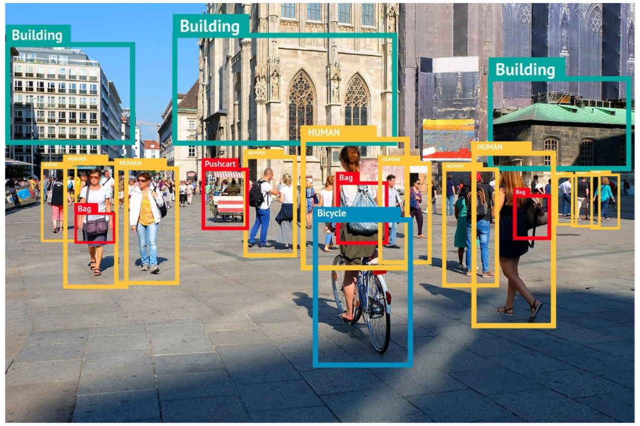
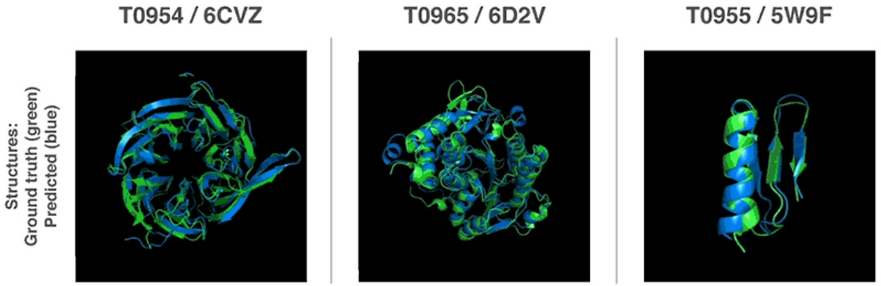


image recognition,
reconstruction, generation,
super-resolution,…

molecules



protein folding, molecule design,…

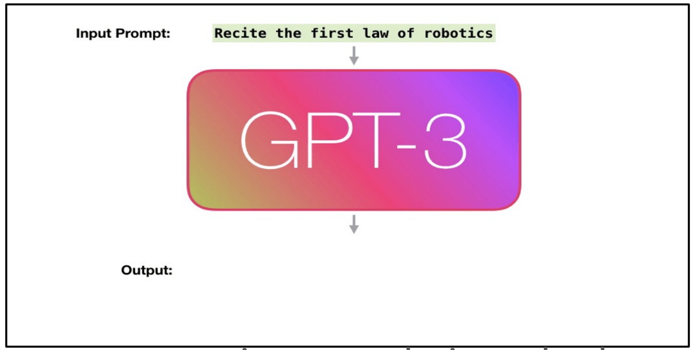
games

super-human play

time-series data

speech recognition, forecasting

natural language

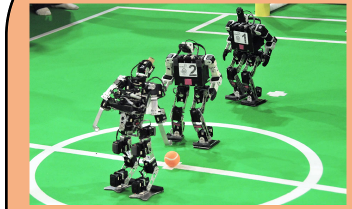


text generation, translation, chatbots,
text embeddings,…

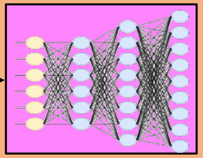
A Dawn of *Multi-Agent* Applications



Multi-player Game-Playing:
- Superhuman GO, Poker, Gran Turismo
- Human-level Starcraft, Diplomacy

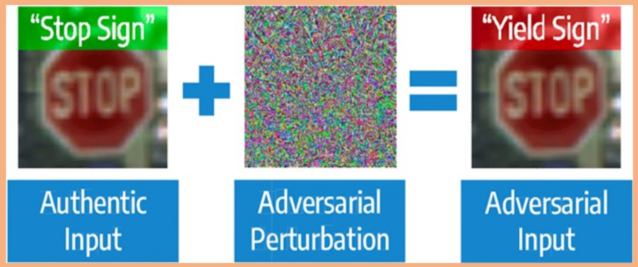


- Multi-robot interactions
- Autonomous driving
- Automated Economic policy design

$Z \sim \mathcal{N}(0, I) \longrightarrow$  $\longrightarrow$  $\sim P_{\text{target}}$

boring randomness — neural network — interesting randomness

Generative Adversarial Networks (GANs)
synthetic data generation



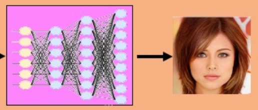
Adversarial Training
robustifying models against adversarial attacks

A Dawn of *Multi-Agent* Applications

Multi-player Game-Playing:
- Superhuman GO, Poker, Gran Turismo
- Human-level Starcraft, Diplomacy

- Multi-robot interactions
- Autonomous driving
- Automated Economic policy design

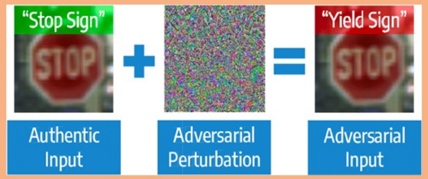
$Z \sim \mathcal{N}(0, I) \longrightarrow$



$\sim P_{\text{target}}$

boring randomness | neural network | interesting randomness

Generative Adversarial Networks (GANs)
synthetic data generation



Adversarial Training
robustifying models against adversarial attacks

Important notes and caveats…

(I) Strategic Behavior does not emerge from standard training

(I) Strategic Behavior does not emerge from standard training (cont’d)

ChatGPT

CO

I am the x player in a game of tic-tac-toe, the other player is o, I am supposed to play next, and the current board configuration looks as follows. Where should I put x?

```
x|  |x
o|o|
 | |
```

Based on the current board configuration, you should put an "x" in the middle-right cell to block your opponent from winning the game.

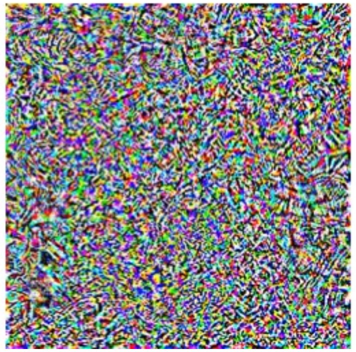
The updated board would look like this:

```
x| |x
o|o|x
| |
```

(II) Naively trained models can be manipulated

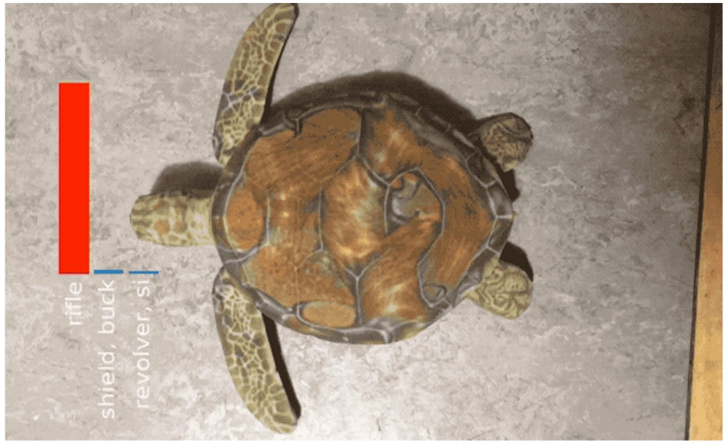
“pig”

+ 0.005 x



=

“airliner”



[Athalye, Engstrom, Ilyas, Kwok ICML’18]

“revolver” “mousetrap” “vulture” “orangutan”

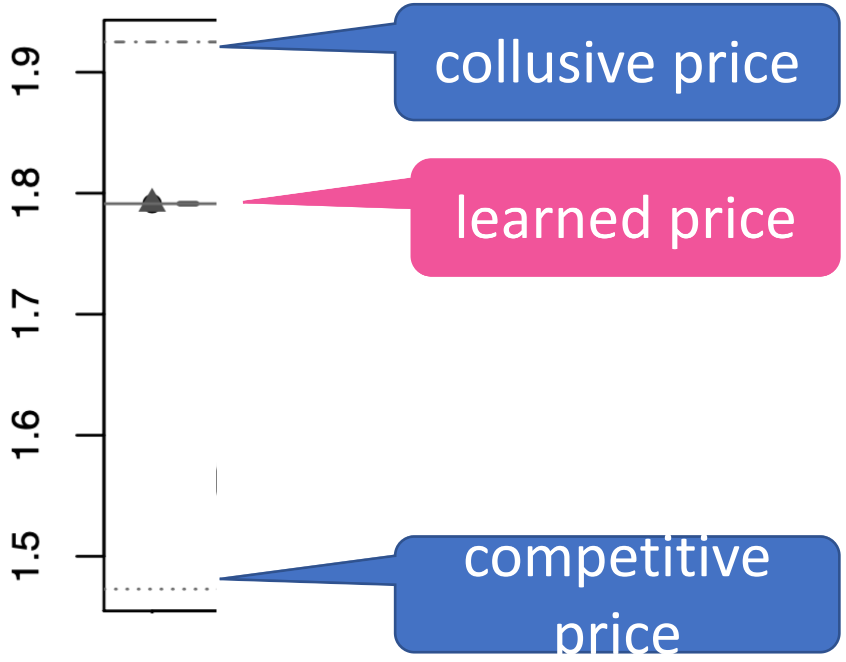
[Engstrom et al. 2019]

(III) Training without regard to the presence of other agents can lead to undesirable consequences

Example: AI for dynamic pricing

Setting: Duopoly w/ two symmetric firms

Independent Learning: firms cannot communicate other than setting prices, observing their profit and adjusting their price using some standard AI algorithm



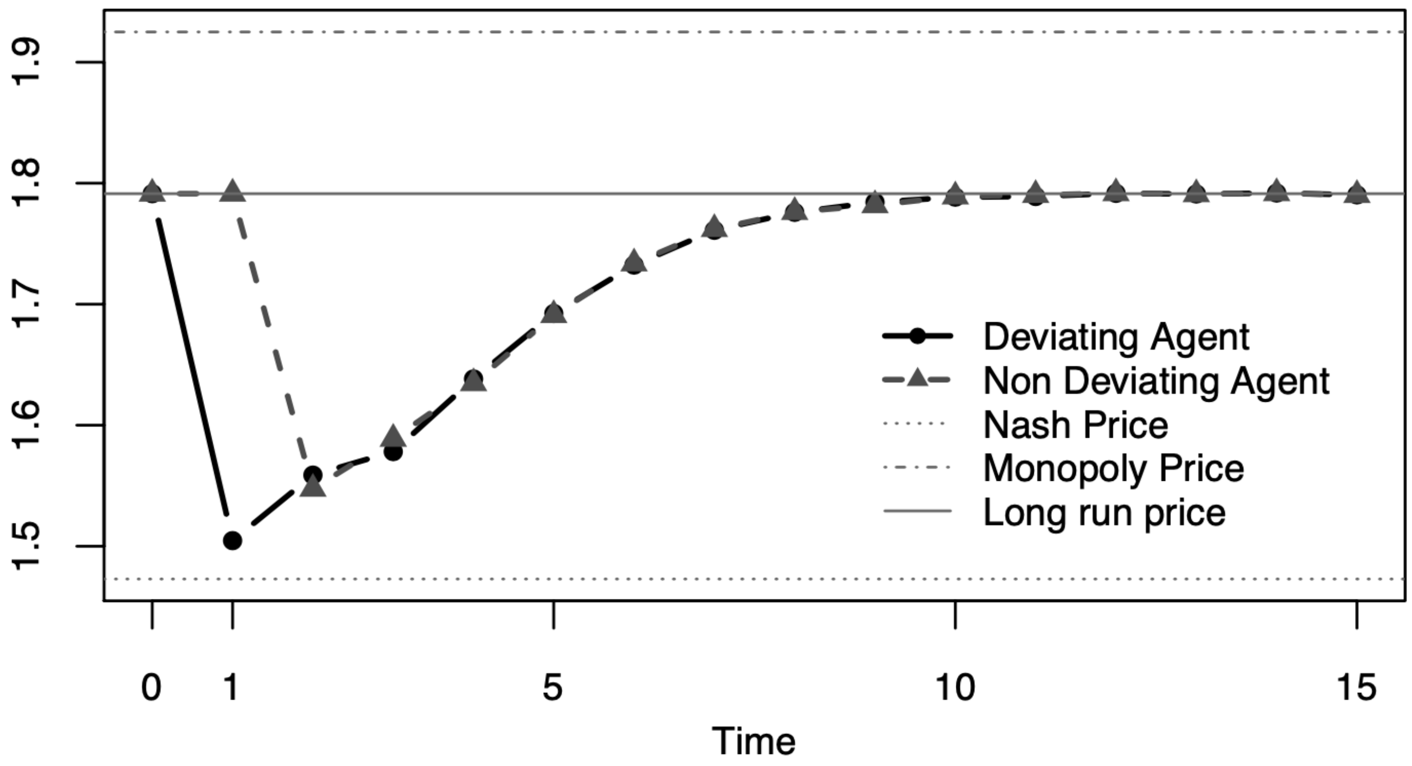
[Calvano, Calzolari, Denicolo, Pastorello: “Artificial Intelligence, Algorithmic Pricing, and Collusion,” American Economic Review, 2020]

(III) Training without regard to the presence of other agents can lead to undesirable consequences

Example: AI for dynamic pricing

Setting: Duopoly w/ two symmetric firms

Independent Learning: firms cannot communicate other than setting prices, observing their profit and adjusting their price using some standard AI algorithm



How deviations are punished by the learned price policies

[Calvano, Calzolari, Denicolo, Pastorello: "Artificial Intelligence, Algorithmic Pricing, and Collusion," American Economic Review, 2020]

(IV) The optimization workhorse of Deep Learning struggles in multi-agent settings

(IV) The optimization workhorse of Deep Learning struggles in multi-agent settings

STANDARD DEEP LEARNING OPTIMIZATION PROBLEM

$$\min_\theta \ell(\theta)$$

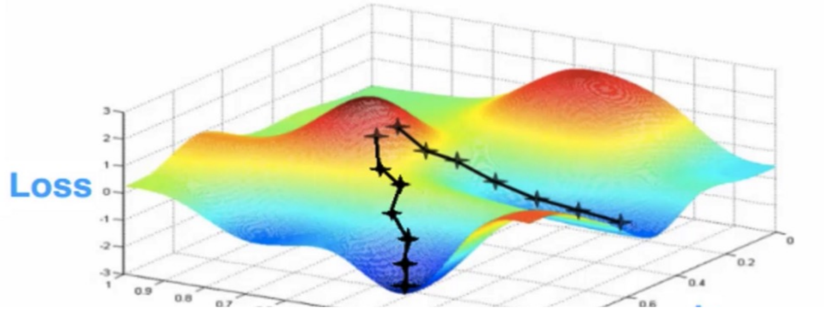
θ: high-dimensional

ℓ: nonconvex

essentially only accessible through $\ell(\theta)$ and $\nabla\ell(\theta)$ queries

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla\ell(\theta_t)$$

Gradient Descent

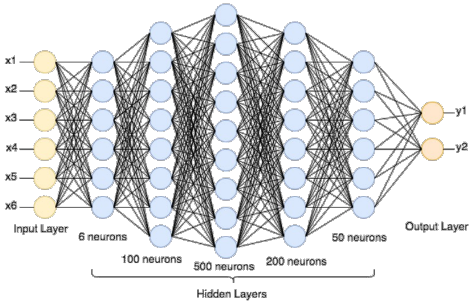


Theoretical Guarantee: Even if ℓ nonconvex, Gradient Descent efficiently computes *local minima* *[e.g. Ge et al '15, Lee et al'17]*

Empirical Finding: *Local minima* are good enough

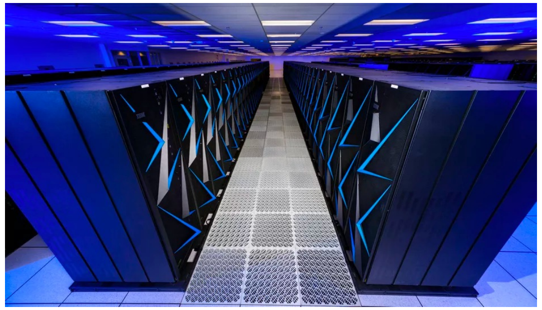
(IV) The optimization workhorse of Deep Learning struggles in multi-agent settings

Prominent Paradigm:



$+\ \theta_{t+1} \leftarrow \theta_t - \nabla_\theta \ell(\theta_t)\ +$

$+$



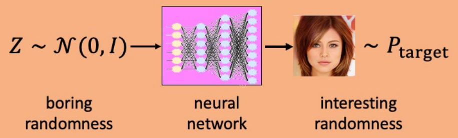
Caffe | Caffe2 | Chainer | Microsoft Cognitive Toolkit | MATLAB | mxnet

PaddlePaddle | PyTorch | TensorFlow | torch | Wolfram Language

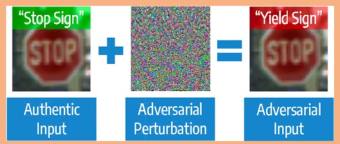
(IV) The optimization workhorse of Deep Learning (a.k.a. Gradient Descent) struggles in multi-agent settings

Multi-player Game-Playing:
- Superhuman GO, Poker, Gran Turismo
- Human-level Starcraft, Diplomacy

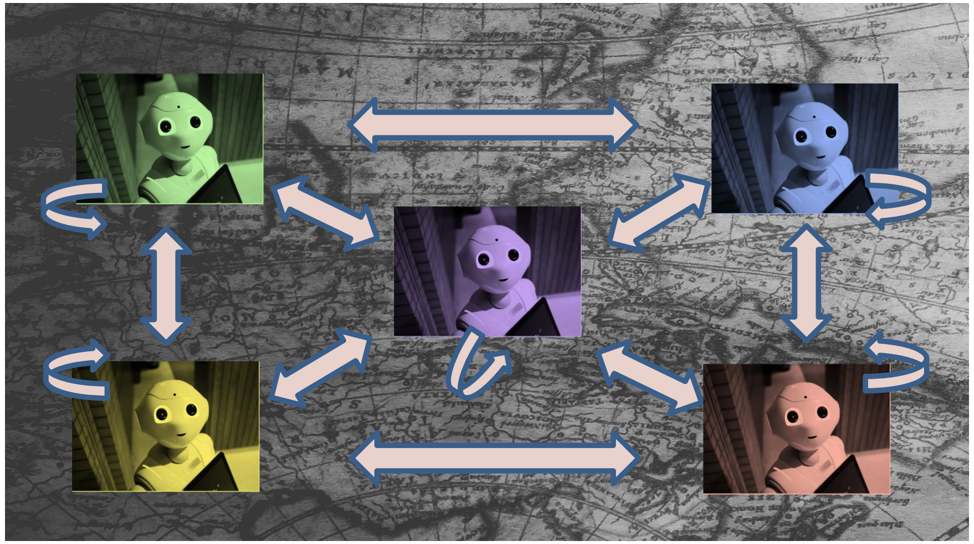
- Multi-robot interactions
- Autonomous driving
- Automated Economic policy design



Generative Adversarial Networks (GANs)
synthetic data generation



Adversarial Training
robustifying models against adversarial attacks



Practical Experience: While GD converges in single-agent learning settings, GD vs GD (vs GD…) is cyclic or chaotic in multi-agent settings, and it's an engineering challenge to make it identify a good solution

(IV) The optimization workhorse of Deep Learning struggles in multi-agent settings

GAN Training: solve two-player zero-sum game where generator player, θ, pays discriminator player, ω, depending on how well, $f(\theta, \omega)$, discriminator distinguishes real vs fake samples

Natural Algorithm: Simultaneous Gradient Descent/Ascent

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_\theta f(\theta_t, \omega_t)$$
$$\omega_{t+1} = \omega_t + \eta \cdot \nabla_\omega f(\theta_t, \omega_t)$$

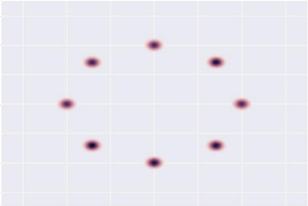
GAN training on MNIST Data:

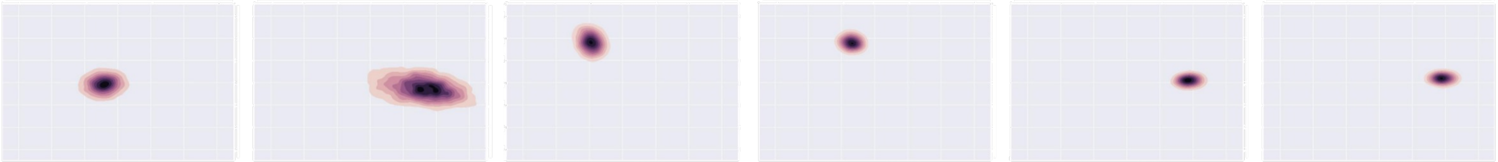
Target dist'n:

10k steps 20k steps 50K steps 100k steps

GAN training on Gaussian Mixture Data:

Target dist'n:



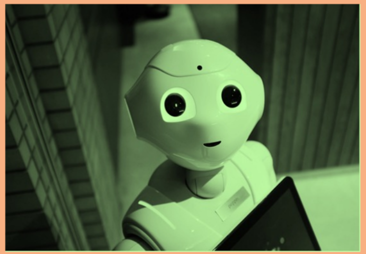


Step 0 Step 5k Step 10k Step 15k Step 20k Step 25k

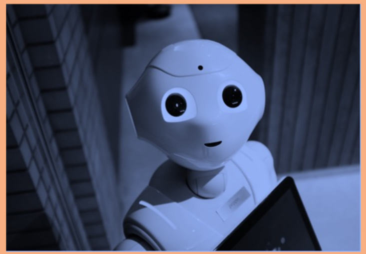
pictures from **[Metz et al ICLR'17]**

(V) Finally Game Theory Breaks

Setting:

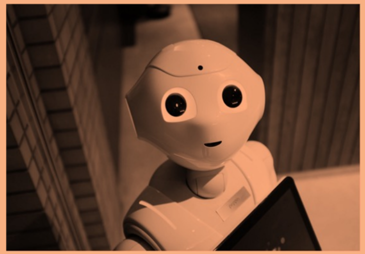


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \dots, x_n)$
(a.k.a. $\min \ell_1(x_1, \dots, x_n)$)



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \dots, x_n)$
(a.k.a. $\min \ell_2(x_1, \dots, x_n)$)

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \dots, x_n)$
(a.k.a. $\min \ell_n(x_1, \dots, x_n)$)

Emerging applications in ***Machine Learning*** involve multiple agents who:

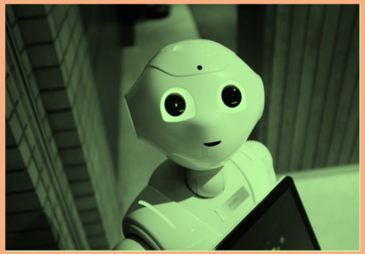
- choose high-dimensional strategies $x_i \in \mathcal{X}_i \subset \mathbb{R}^{d_i}$ (e.g. parameters in a DNN)
- maximize utility functions $u_i(x_i\,; x_{-i})$ that are ***nonconcave*** in their own strategy (a.k.a. minimize loss functions that are **nonconvex** in their own strategy)

Issue: Game Theory is fragile when utilities are nonconcave

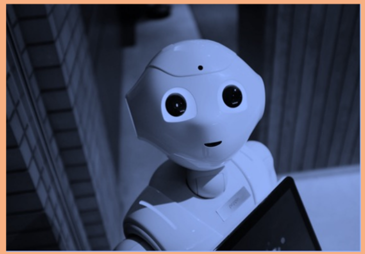
- in particular, Nash equilibrium (and other types of equilibrium) may not exist
- so what is even our recommendation about reasonable optimization targets?

(V) Finally Game Theory Breaks

Setting:

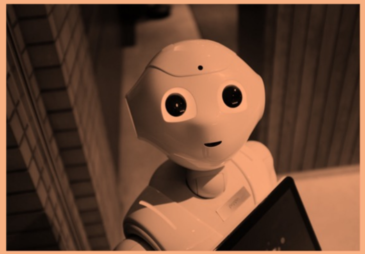


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \dots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \dots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \dots, x_n)$

Nash Eq: *A collection of* $x_1^*, \dots x_n^*$ s.t. for all i, x_i: $u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*)$

Randomized Nash Eq: *A collection of distributions* $p_1, \dots p_n$ s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i; x_{-i}^*)]$$

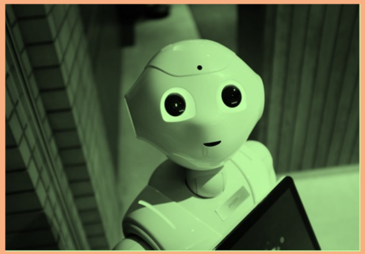
Coarse Correlated Eq: *A joint distribution of* p s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p}[u_i(x_i; x_{-i}^*)]$$

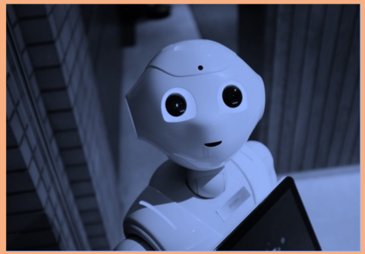
[Debreu'52, Rosen'65]: If each $u_i(x_i; x_{-i})$ is continuous and concave in x_i for all x_{-i} and each $\mathcal{X}_i$ is convex and compact, a Nash equilibrium exists.

(V) Finally Game Theory Breaks

Setting:

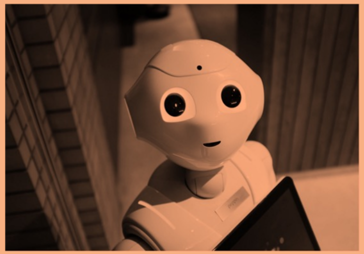


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \ldots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \ldots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \ldots, x_n)$

Nash Eq: *A collection of* $x_1^*, \ldots x_n^*$ s.t. for all i, x_i: $u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*)$

Randomized Nash Eq: *A collection of distributions* $p_1, \ldots p_n$ s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p_1 \times \cdots \times p_n}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p_1 \times \cdots \times p_n}[u_i(x_i; x_{-i}^*)]$$

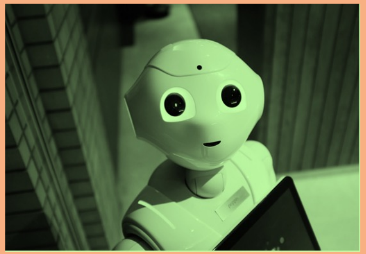
Coarse Correlated Eq: *A joint distribution of* p s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p}[u_i(x_i; x_{-i}^*)]$$

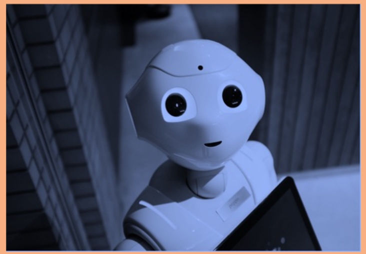
[Debreu'52, Rosen'65]: If each $u_i(x_i; x_{-i})$ is continuous and concave in x_i for all x_{-i} and each $\mathcal{X}_i$ is convex and compact, a Nash equilibrium exists.
e.g. Nash equilibrium in finite normal-form games **[Nash'50]**

(V) Finally Game Theory Breaks

Setting:

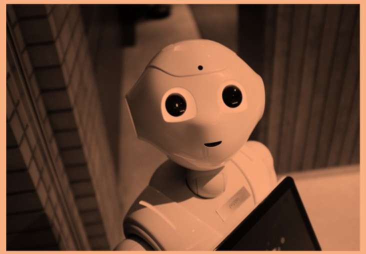


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \dots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \dots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \dots, x_n)$

Nash Eq: *A collection of* $x_1^*, \dots x_n^*$ s.t. for all i, x_i: $u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*)$

Randomized Nash Eq: *A collection of distributions* $p_1, \dots p_n$ s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i; x_{-i}^*)]$$

Coarse Correlated Eq: *A joint distribution of* p s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p}[u_i(x_i; x_{-i}^*)]$$

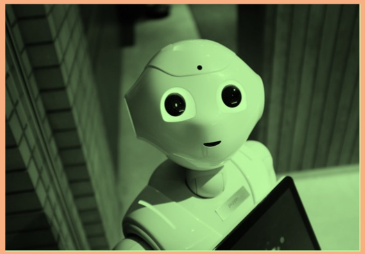
[Debreu'52, Rosen'65]: If each $u_i(x_i; x_{-i})$ is continuous and concave in x_i for all x_{-i} and each $\mathcal{X}_i$ is convex and compact, a Nash equilibrium exists.

e.g. Nash equilibrium in finite normal-form games **[Nash'50]**

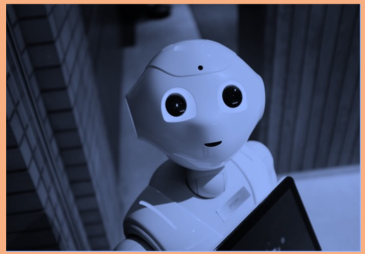
- in this case: $\mathcal{X}_i = \Delta(A_i)$ and $u_i(x_i; x_{-i}) = \sum_{a \in \times_j A_j} u_i(a) x_1(a_1) \cdots x_n(a_n)$

(V) Finally Game Theory Breaks

Setting:

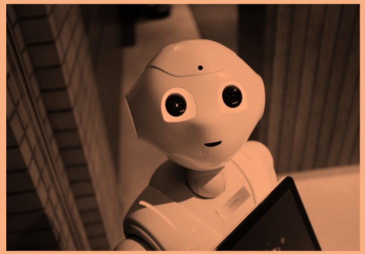


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \ldots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \ldots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \ldots, x_n)$

Nash Eq: *A collection of* $x_1^*, \ldots x_n^*$ s.t. for all i, x_i: $u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*)$

Randomized Nash Eq: *A collection of distributions* $p_1, \ldots p_n$ s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p_1 \times \cdots \times p_n}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p_1 \times \cdots \times p_n}[u_i(x_i; x_{-i}^*)]$$

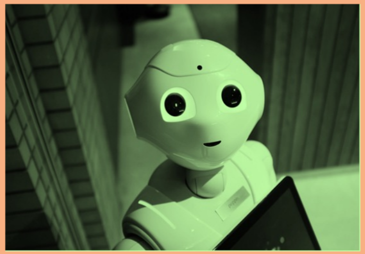
Coarse Correlated Eq: *A joint distribution of* p s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p}[u_i(x_i; x_{-i}^*)]$$

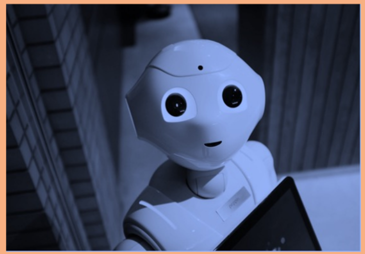
If some $u_i(x_i; x_{-i})$ is not concave in x_i for all x_{-i}, a Nash equilibrium does not necessarily exist
e.g. two-player zero-sum game: $u_1(x_1, x_2) = -u_2(x_1, x_2) = (x_1 - x_2)^2$ where $x_1, x_2 \in [-1,1]$

(V) Finally Game Theory Breaks

Setting:

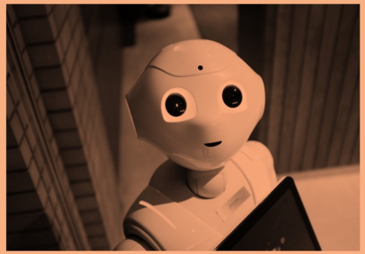


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \dots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \dots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \dots, x_n)$

Nash Eq: *A collection of* $x_1^*, \dots x_n^*$ s.t. for all i, x_i: $u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*)$

Randomized Nash Eq: *A collection of distributions* $p_1, \dots p_n$ s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i; x_{-i}^*)]$$

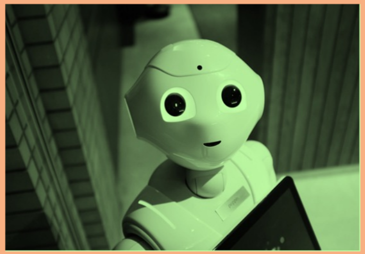
Coarse Correlated Eq: *A joint distribution of* p s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p}[u_i(x_i; x_{-i}^*)]$$

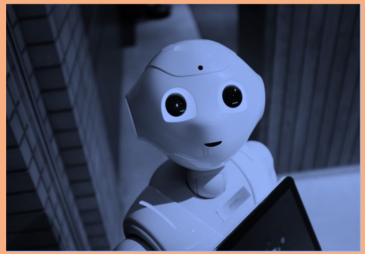
If some $u_i(x_i; x_{-i})$ is not concave in x_i for all x_{-i}, a Nash equilibrium does not necessarily exist
e.g.2 Generative adversarial networks

(V) Finally Game Theory Breaks

Setting:

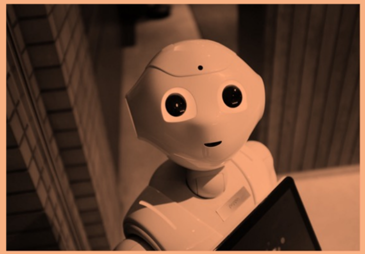


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \dots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \dots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \dots, x_n)$

Nash Eq: *A collection of* $x_1^*, \dots x_n^*$ s.t. for all i, x_i: $u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*)$

Randomized Nash Eq: *A collection of distributions* $p_1, \dots p_n$ s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p_1 \times \dots \times p_n}[u_i(x_i; x_{-i}^*)]$$

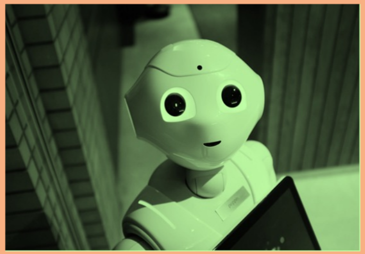
Coarse Correlated Eq: *A joint distribution of* p s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p}[u_i(x_i; x_{-i}^*)]$$

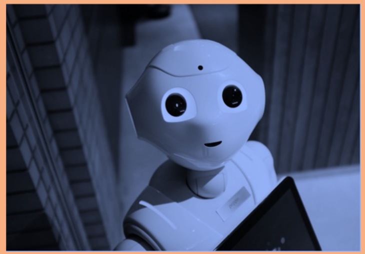
If some $u_i(x_i; x_{-i})$ is not concave in x_i for all x_{-i}, Nash equilibrium does not necessarily exist
[Glicksberg'52]: A *randomized* Nash equilibrium does exist if the $\mathcal{X}_i$'s are compact and the u_i's are continuous (and not necessarily concave), *but support could be uncountably infinite*.

(V) Finally Game Theory Breaks

Setting:

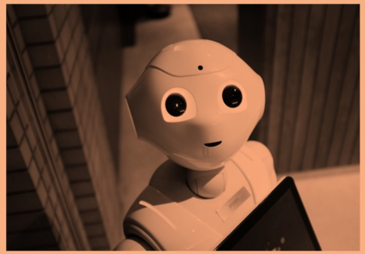


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \ldots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \ldots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \ldots, x_n)$

Nash Eq: *A collection of* $x_1^*, \ldots x_n^*$ s.t. for all i, x_i: $u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*)$

Randomized Nash Eq: *A collection of distributions* $p_1, \ldots p_n$ s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p_1 \times \cdots \times p_n}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p_1 \times \cdots \times p_n}[u_i(x_i; x_{-i}^*)]$$

Coarse Correlated Eq: *A joint distribution of* p s.t. for all i, x_i:

$$\mathrm{E}_{x^* \sim p}[u_i(x_i^*; x_{-i}^*)] \geq \mathrm{E}_{x^* \sim p}[u_i(x_i; x_{-i}^*)]$$

If some $u_i(x_i; x_{-i})$ is not concave in x_i for all x_{-i}, Nash equilibrium does not necessarily exist
If the $\mathcal{X}_i$'s are non-compact, even randomized Nash/correlated eq do not necessarily exist
e.g. "Guess-the-larger-number" game

- two players choose a real; whoever chooses the largest real receives one point from the other

Summary so far…

- (I) Strategic Behavior does not emerge from standard training
- (II) Naively trained models can be manipulated
- (III) Training without regard to the presence of other agents can lead to undesirable (e.g. collusive) consequences
- (IV) The optimization workhorse of Deep Learning (a.k.a. gradient descent) struggles in multi-agent settings
- (V) Finally Game Theory (namely the existence of Nash equilibrium and other types of equilibrium) breaks

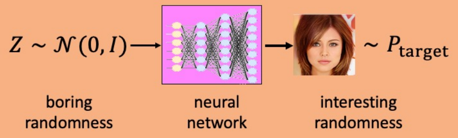
Motivating Questions



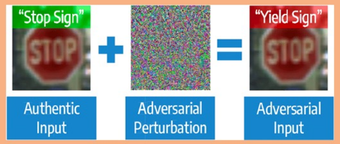
Multi-player Game-Playing:
- Superhuman GO, Poker, Gran Turismo
- Human-level Starcraft, Diplomacy



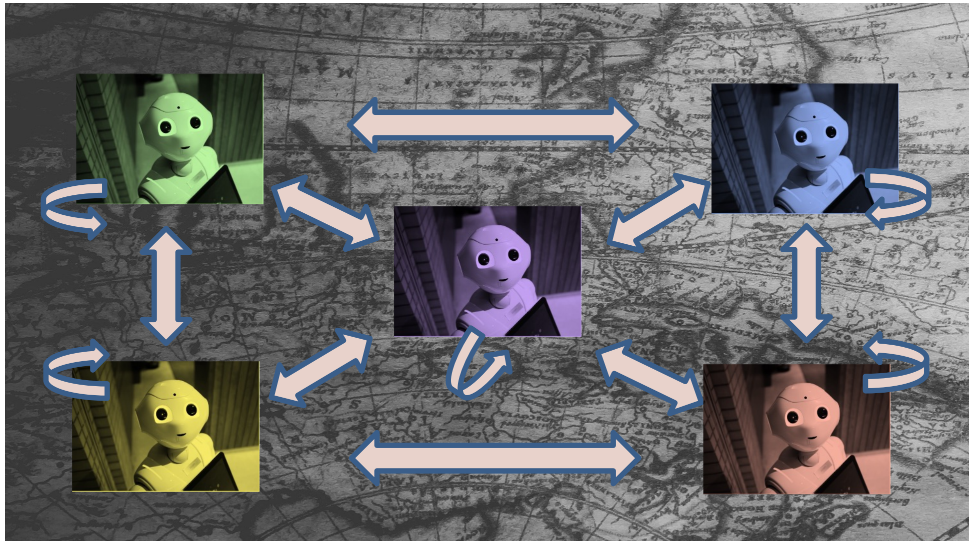
- Multi-robot interactions
- Autonomous driving
- Automated Economic policy design



Generative Adversarial Networks (GANs)
synthetic data generation



Adversarial Training
robustifying models against adversarial attacks



Practical Experience: GD vs GD (vs GD…) is cyclic or chaotic, and it is a hard engineering challenge to make it identify a good solution

What are meaningful and practically attainable optimization targets in this setting?

GENERALIZATIONS OF LOCAL OPTIMUM?

Why does GD vs GD struggle even in two-player zero-sum cases?

INTRACTABILITY? or WRONG METHOD?

Is there a generic optimization framework for Multi-Agent Deep Learning?

OR DO WE NEED STRUCTURE?

Intermission: Sign-up for project presentations!

PROJECT PRESENTATIONS

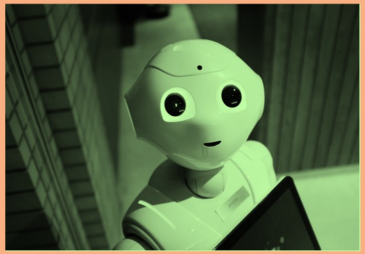
11/30	Projects
12/5	Projects
12/7	Projects

Presentation format: 15 mins + 5 mins Q & A

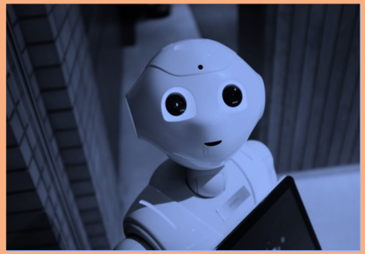
Write-up format: 10 pages + appendix (due 12/14)

Local Nash Equilibrium

Setting:

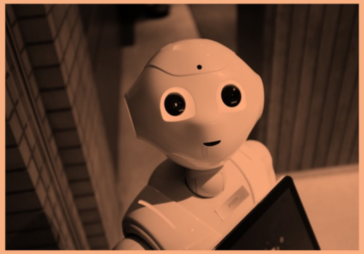


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \ldots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \ldots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \ldots, x_n)$

u_i is Lipschitz and smooth (i.e. has Lipschitz gradient) a.e.
[allow: global constraints $(x_1, x_2, \ldots, x_n) \in \mathcal{S} \subseteq \times_i \mathcal{X}_i$]

Overarching Q: What are meaningful and practically attainable optimization targets in this setting?

"*meaningful*:" at the very least universal, verifiable with the info that agents have about their loss functions
"*practically attainable*:" efficiently reachable via gradient descent-like (or similar light-weight) method

Q: Perhaps some generalization to this setting of local optimum?

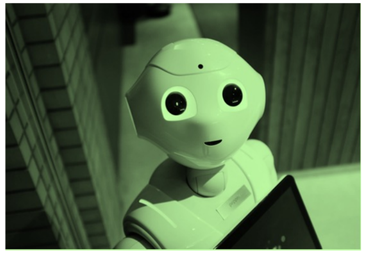
A weak optimization target: Local Nash Equilibrium [Ratliff-Burden-Sastry'16, Daskalakis-Panageas'18, Mazumdar-Ratliff'18, Jin-Netrapali-Jordan'20]
A point $x^* = (x_1^*, \ldots, x_n^*) \in \mathcal{S}$ such that, for each player i, x_i^* is local max of $u_i(x_i\,;\,x_{-i}^*)$ w.r.t. x_i

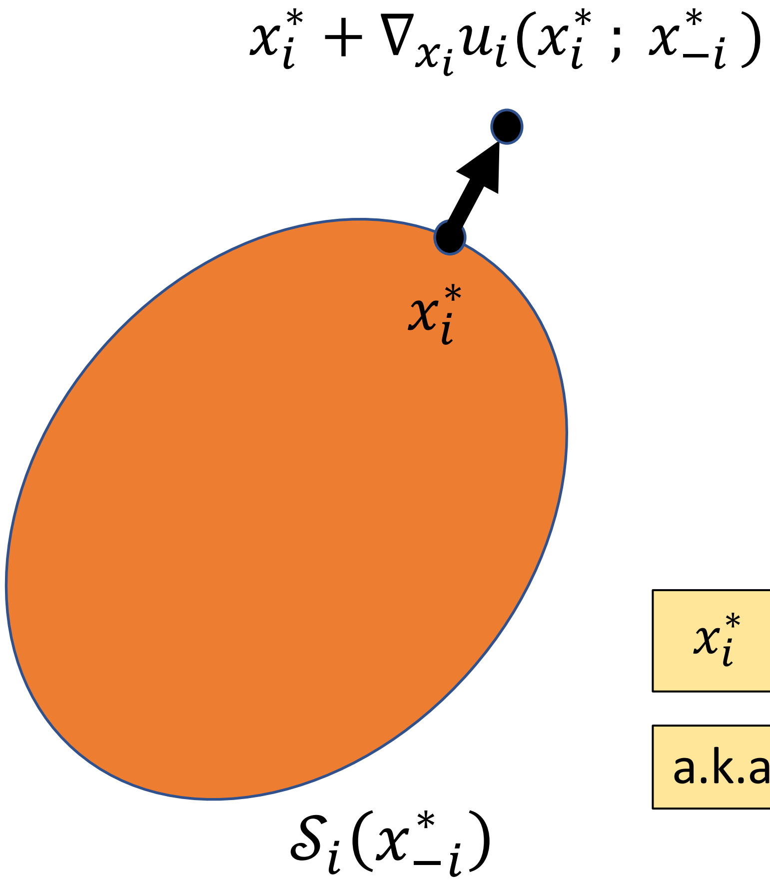
Weakest variant: First-Order Local Nash Equilibrium

Take "local max" to mean "First-order local max" i.e. max w.r.t. first-order Taylor appx

First-Order Local Nash Equilibrium: agent i's viewpoint

x_i^* best response to x_{-i}^* as far as the first-order Taylor approximation can tell

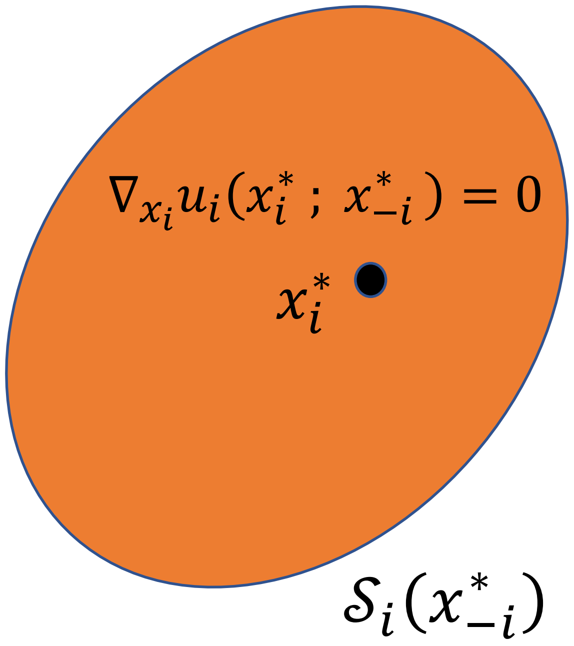




OR

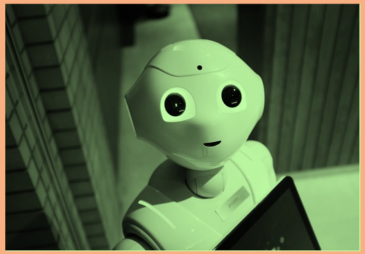
$x_i^* = \Pi_{\mathcal{S}_i(x_{-i}^*)}\left(x_i^* + \nabla_{x_i} u_i(x_i^* \,;\, x_{-i}^*)\right)$

a.k.a. fixed point of GD vs GD (vs GD…)

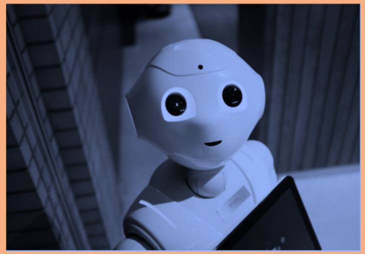


Local Nash Equilibrium: Existence

Setting:

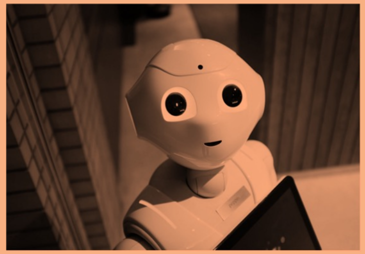


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \dots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \dots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \dots, x_n)$

u_i is Lipschitz and smooth (i.e. has Lipschitz gradient) a.e.
[allow: global constraints $(x_1, x_2, \dots, x_n) \in \mathcal{S} \subseteq \times_i \mathcal{X}_i$]

Def: A strategy profile $x^* = (x_1^*, \dots, x_n^*) \in \mathcal{S}$ is a *(first-order) local Nash equilibrium* iff for all i:

$$x_i^* = \Pi_{\mathcal{S}_i(x_{-i}^*)}\left(x_i^* + \nabla_{x_i} u_i(x_i^* \,;\, x_{-i}^*)\right)$$

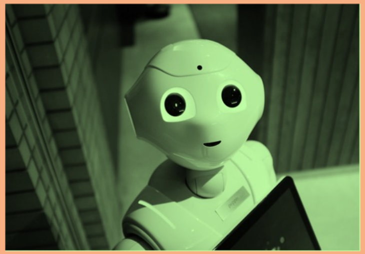
where $\mathcal{S}_i(x_{-i}^*) = \{x_i \,|\, (x_i \,;\, x_{-i}^*) \in \mathcal{S}\}$, and $\Pi_{\mathcal{S}_i(x_{-i}^*)}(\cdot)$ is the Euclidean projection onto the set $\mathcal{S}_i(x_{-i}^*)$

Proposition: If $\mathcal{S}$ is convex and compact, a *(first-order) local Nash equilibrium* exists.

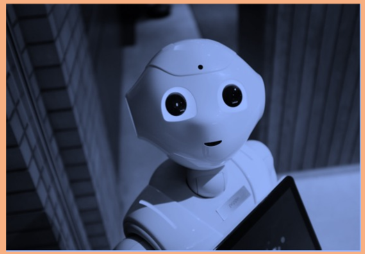
so both universal and verifiable with the info that players have about their utilities

Local Nash Equilibrium: Existence

Setting:

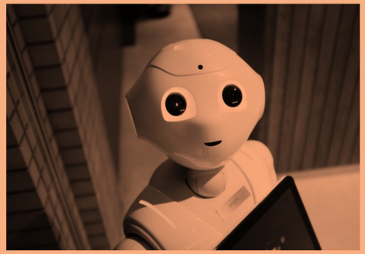


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \ldots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \ldots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \ldots, x_n)$

u_i is Lipschitz and smooth (i.e. has Lipschitz gradient) a.e.
[allow: global constraints $(x_1, x_2, \ldots, x_n) \in \mathcal{S} \subseteq \times_i \mathcal{X}_i$]

Def: A strategy profile $x^* = (x_1^*, \ldots, x_n^*) \in \mathcal{S}$ is a *(first-order) local Nash equilibrium* iff for all i:

$$x_i^* = \Pi_{\mathcal{S}_i(x_{-i}^*)}\left(x_i^* + \nabla_{x_i} u_i(x_i^* ; x_{-i}^*)\right)$$

where $\mathcal{S}_i(x_{-i}^*) = \{x_i \mid (x_i ; x_{-i}^*) \in \mathcal{S}\}$, and $\Pi_{\mathcal{S}_i(x_{-i}^*)}(\cdot)$ is the Euclidean projection onto the set $\mathcal{S}_i(x_{-i}^*)$

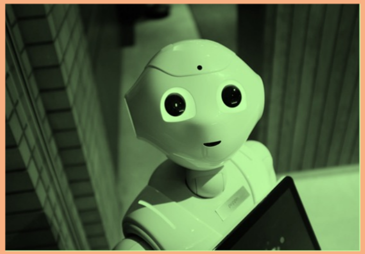
Proposition: If $\mathcal{S}$ is convex and compact, a *(first-order) local Nash equilibrium* exists.

so both universal and verifiable with the info that players have about their utilities

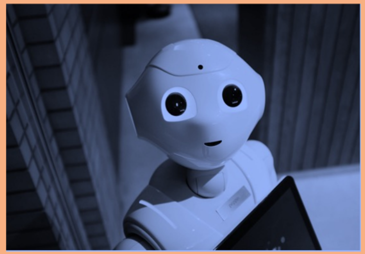
are they practically attainable?

Local Nash Equilibrium: Complexity

Setting:

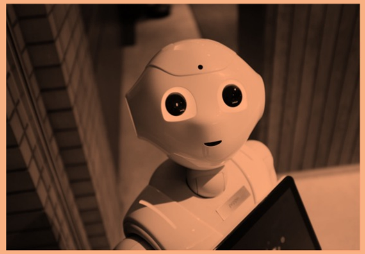


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \ldots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \ldots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \ldots, x_n)$

u_i is Lipschitz and smooth (i.e. has Lipschitz gradient) a.e.
[allow: global constraints $(x_1, x_2, \ldots, x_n) \in \mathcal{S} \subseteq \times_i \mathcal{X}_i$]

Def: A strategy profile $x^* = (x_1^*, \ldots, x_n^*) \in \mathcal{S}$ is a *(first-order) local Nash equilibrium* iff for all i:

$$x_i^* = \Pi_{\mathcal{S}_i(x_{-i}^*)}\left(x_i^* + \nabla_{x_i} u_i(x_i^* \,;\, x_{-i}^*)\right)$$

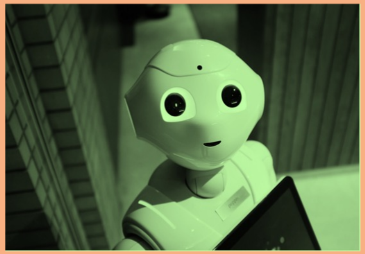
where $\mathcal{S}_i(x_{-i}^*) = \{x_i \mid (x_i \,;\, x_{-i}^*) \in \mathcal{S}\}$, and $\Pi_{\mathcal{S}_i(x_{-i}^*)}(\cdot)$ is the Euclidean projection onto the set $\mathcal{S}_i(x_{-i}^*)$

Proposition: If $\mathcal{S}$ is convex and compact, a *(first-order) local Nash equilibrium* exists.

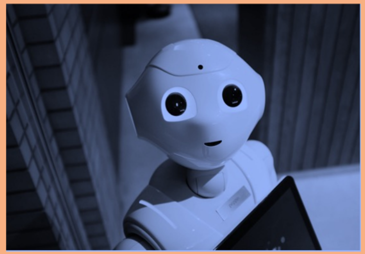
Theorem [w/ Skoulakis & Zampetakis STOC'21]: *Even in two-player zero-sum smooth non-concave games*, any method accessing the u_i's via value and gradient value queries needs exponentially many queries (in the dimension and/or $1/\varepsilon$) to compute even an ε-approximate local Nash equilibrium, i.e. some x^* such that for all i: $\left\| x_i^* - \Pi_{\mathcal{S}(x_{-i}^*)}\left(x_i^* + \nabla_{x_i} u_i(x_i^* \,;\, x_{-i}^*)\right)\right\| \le \varepsilon$.

Local Nash Equilibrium: Complexity

Setting:

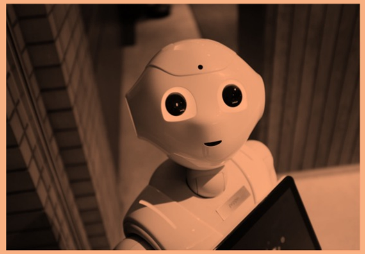


action: $x_1 \in \mathcal{X}_1 \subset \mathbb{R}^{d_1}$
goal: max $u_1(x_1, \dots, x_n)$



action: $x_2 \in \mathcal{X}_2 \subset \mathbb{R}^{d_2}$
goal: max $u_2(x_1, \dots, x_n)$

…



action: $x_n \in \mathcal{X}_n \subset \mathbb{R}^{d_n}$
goal: max $u_n(x_1, \dots, x_n)$

u_i is Lipschitz and smooth (i.e. has Lipschitz gradient) a.e.
[allow: global constraints $(x_1, x_2, \dots, x_n) \in \mathcal{S} \subseteq \times_i \mathcal{X}_i$]

Def: A strategy profile $x^* = (x_1^*, \dots, x_n^*) \in \mathcal{S}$ is a *(first-order) local Nash equilibrium* iff for all i:

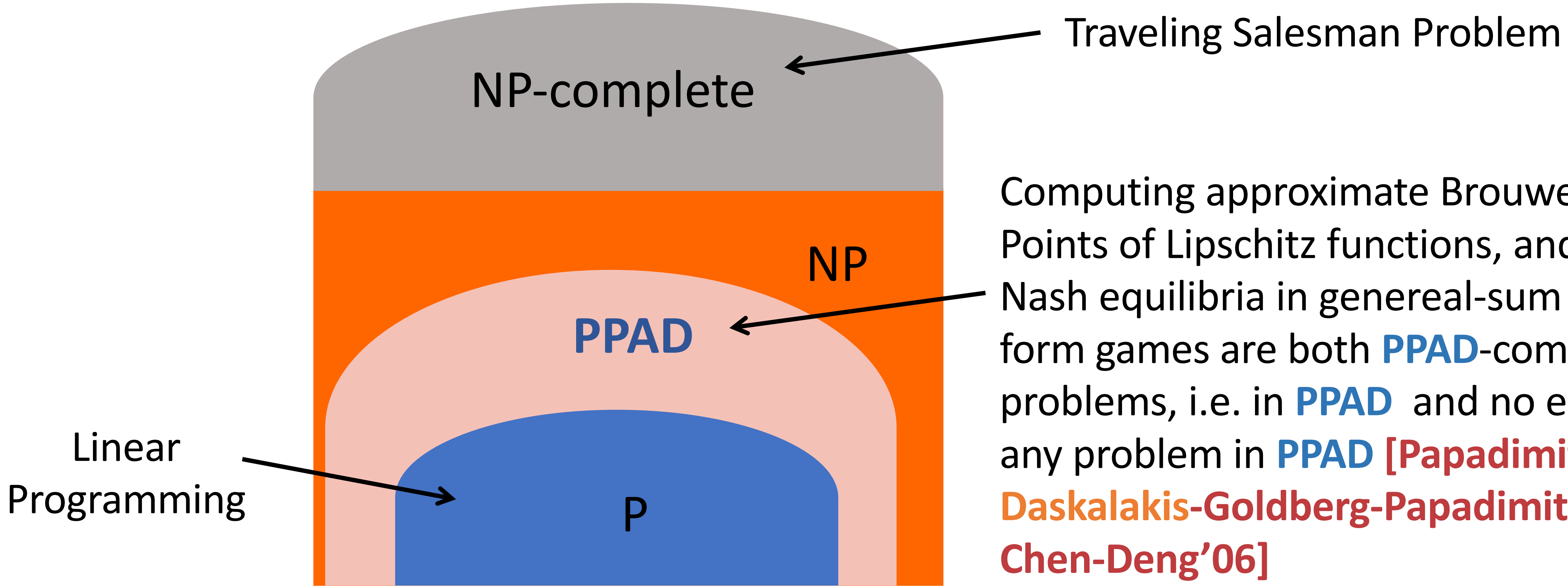
$$x_i^* = \Pi_{\mathcal{S}_i(x_{-i}^*)}\left(x_i^* + \nabla_{x_i} u_i(x_i^* ;\, x_{-i}^*)\right)$$

where $\mathcal{S}_i(x_{-i}^*) = \{x_i \mid (x_i ; x_{-i}^*) \in \mathcal{S}\}$, and $\Pi_{\mathcal{S}_i(x_{-i}^*)}(\cdot)$ is the Euclidean projection onto the set $\mathcal{S}_i(x_{-i}^*)$

Proposition: If $\mathcal{S}$ is convex and compact, a *(first-order) local Nash equilibrium* exists.

Theorem [w/ Skoulakis & Zampetakis STOC'21]: *Even in two-player zero-sum smooth non-concave games,* any method **at all** needs **super-polynomial-time** (in the dimension and/or $1/\varepsilon$) to compute even an ε-approximate local Nash equilibrium, **unless PPAD=P**.

The Complexity of Local Nash Equilibrium

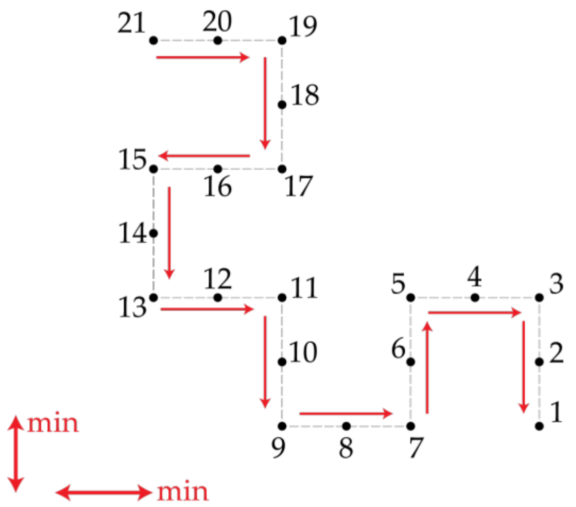


Computing approximate Brouwer Fixed Points of Lipschitz functions, and mixed Nash equilibria in genereal-sum normal-form games are both **PPAD**-complete problems, i.e. in **PPAD** and no easier than any problem in **PPAD** **[Papadimitriou'94, Daskalakis-Goldberg-Papadimitriou'06, Chen-Deng'06]**

[Daskalakis-Skoulakis-Zampetakis STOC'21]: Computing local Nash equilibria *(even in two-player zero-sum and smooth)* non-concave games is exactly as hard as (i) computing approximate Brouwer fixed points of Lipschitz functions; (ii) computing mixed Nash equilibria in general-sum normal-form games; and (iii) at least as hard as any other problem in **PPAD**.

Intuition: why are even two players too many?

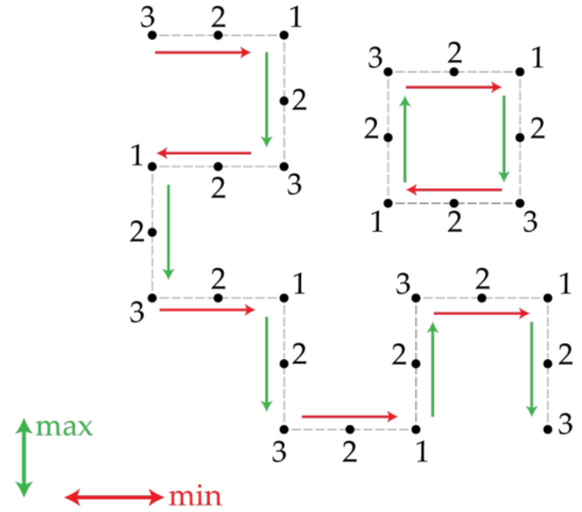
Compare properties of objective-improving moves in single-player optimization problems (where finding approximate local optima is known to be tractable) and better-response dynamics in two-player zero-sum games (where we show that finding approximate local Nash equilibria is intractable)



single-agent non-convex minimization /non-concave maximization

two-player zero-sum non-concave game

(showing player 1's value)



objective value decreases along objective-improving path, thus: (i) moving along path makes progress towards (local) optimum

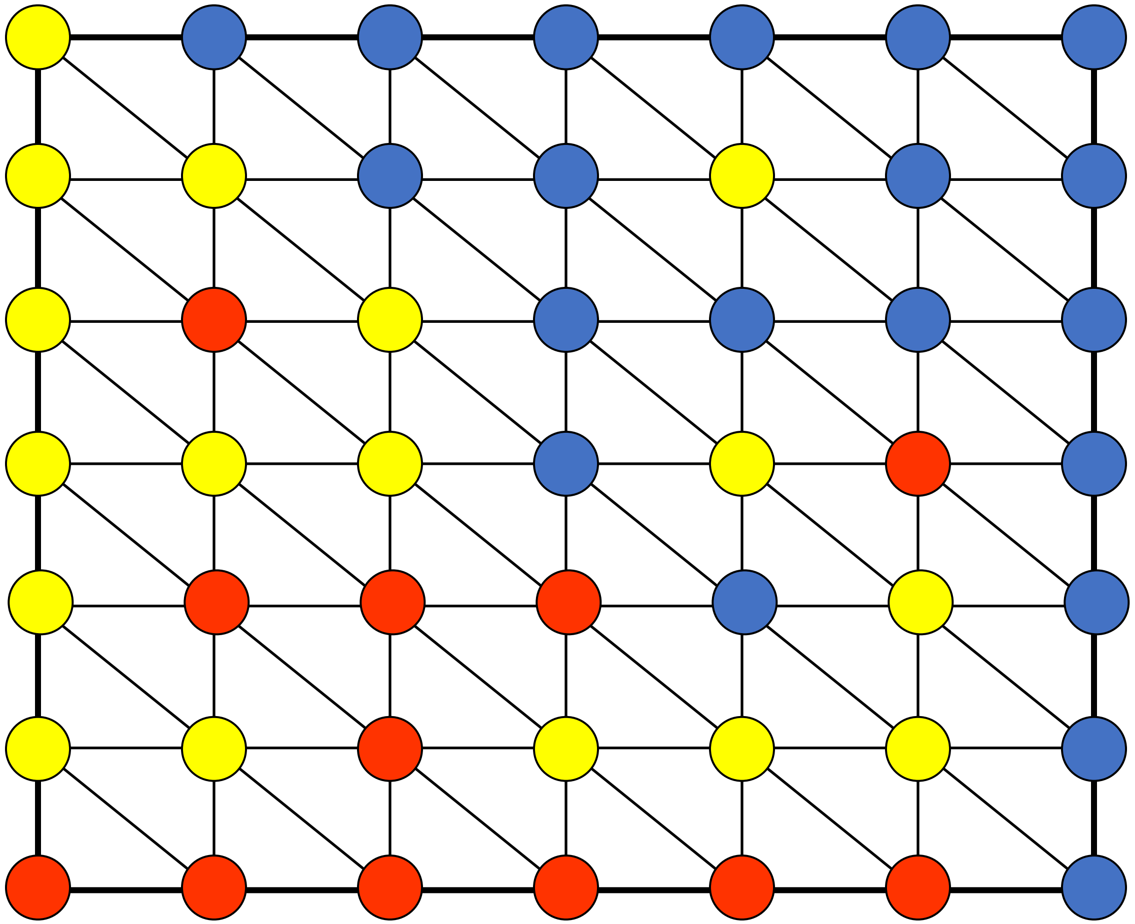
(ii) quantitative version: for bounded objectives (e.g. continuous objective over compact space), function value along ε-improving path bounds distance from the end of the path (memory/information gain)

better-response paths may be cyclic :S

objective value along non-cyclic ε-better-response path does not reveal information about distance to end of the path!

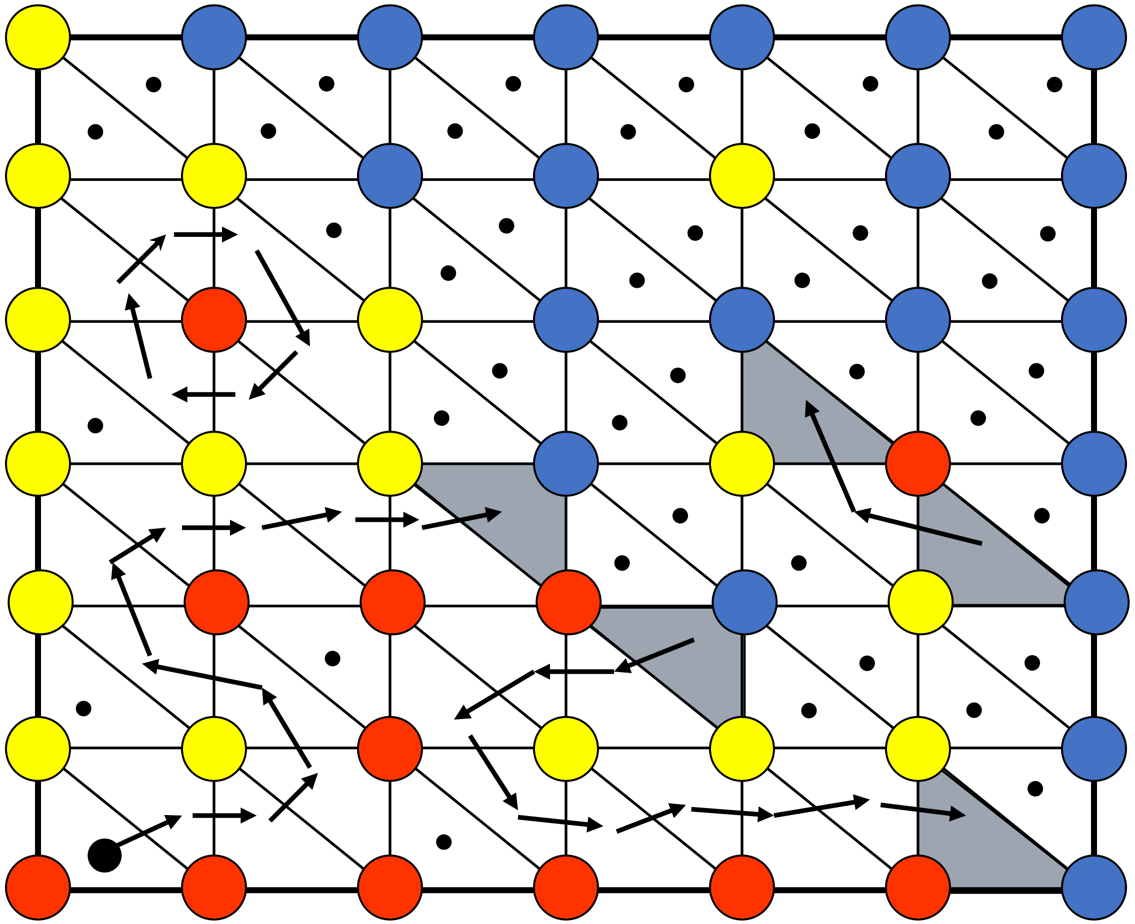
to turn this intuition into an intractability proof, need to hide exponentially long better-response path within ambient space s.t. *no matter where the function is queried* little information is revealed about location of local Nash equilibria

Rough Proof Idea: Reduce from Sperner



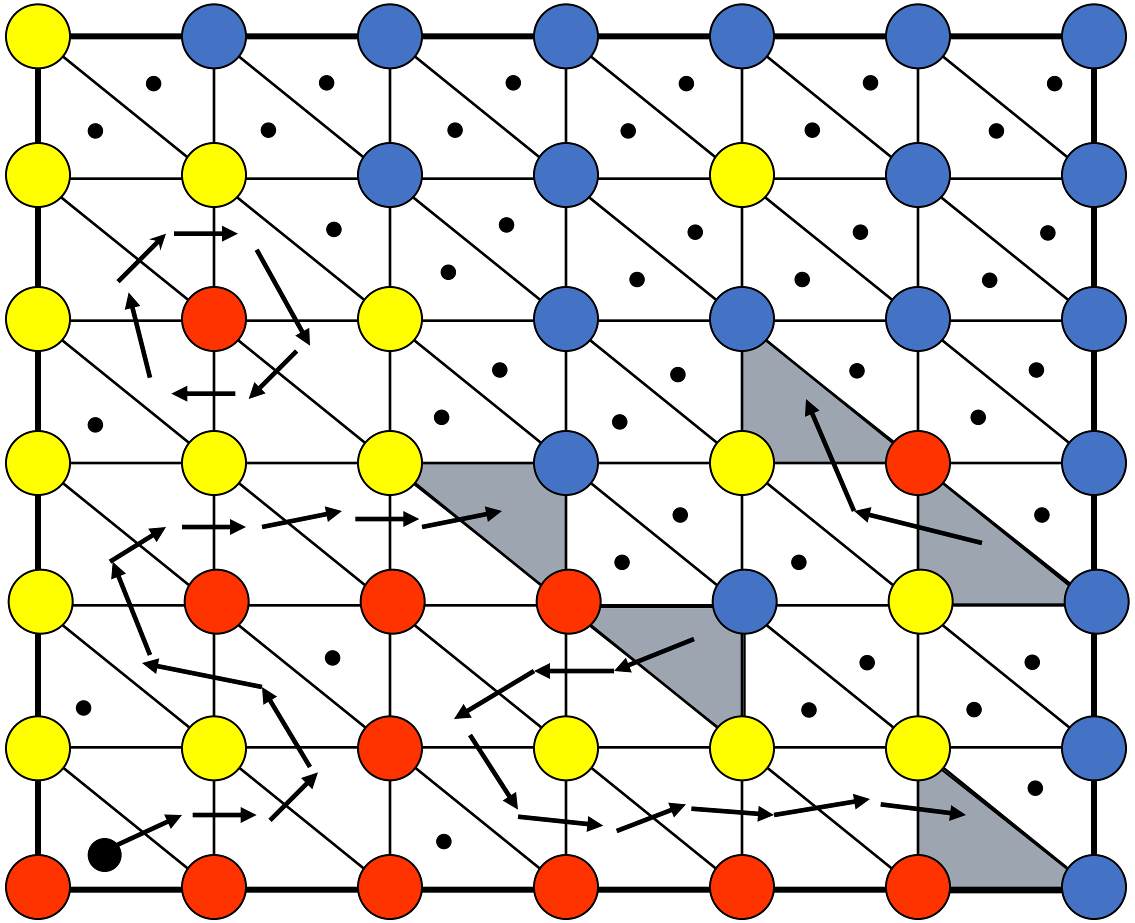
Lemma: If boundary coloring is valid, then no matter how the internal nodes are colored there exists a tri-chromatic triangle. In fact, an odd number of them.

Rough Proof Idea: Reduce from Sperner



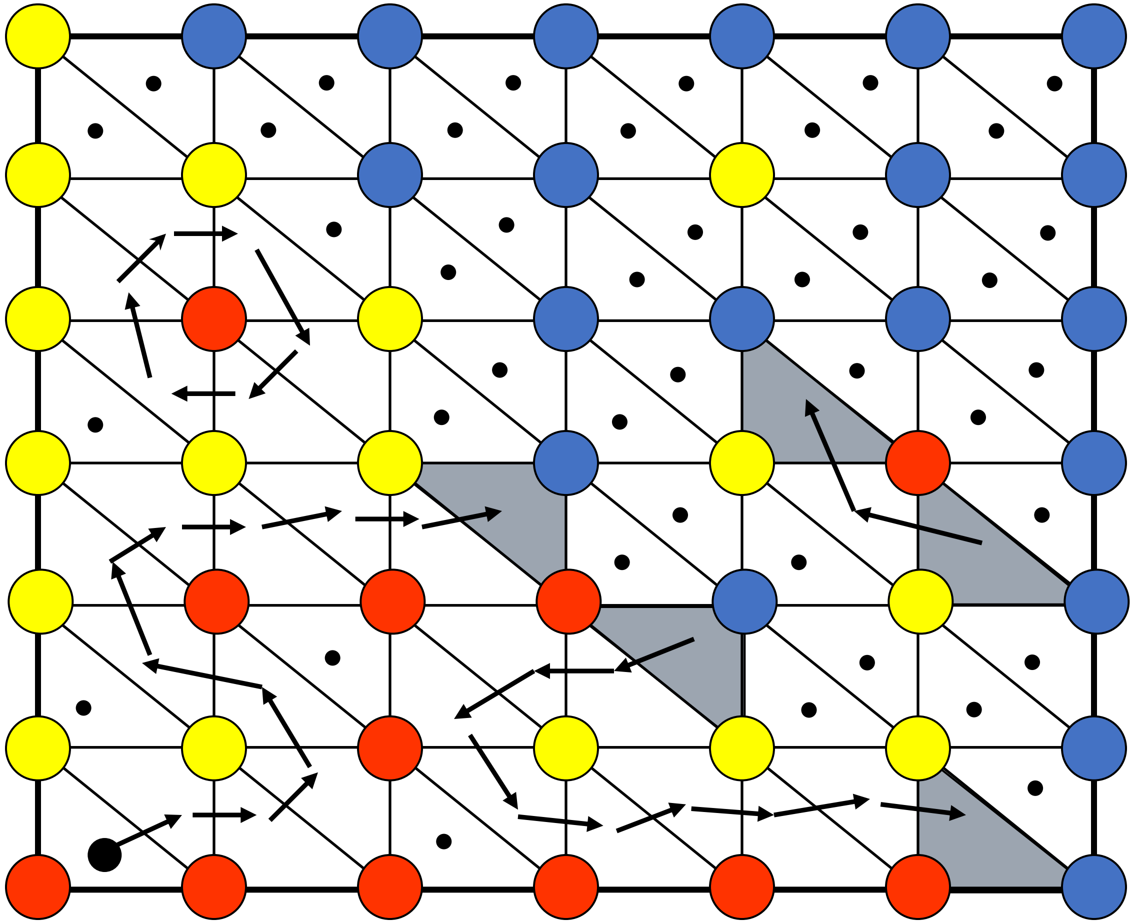
Lemma: If boundary coloring is valid, then no matter how the internal nodes are colored there exists a tri-chromatic triangle. In fact, an odd number of them.

Rough Proof Idea: Edge-Triangle Game



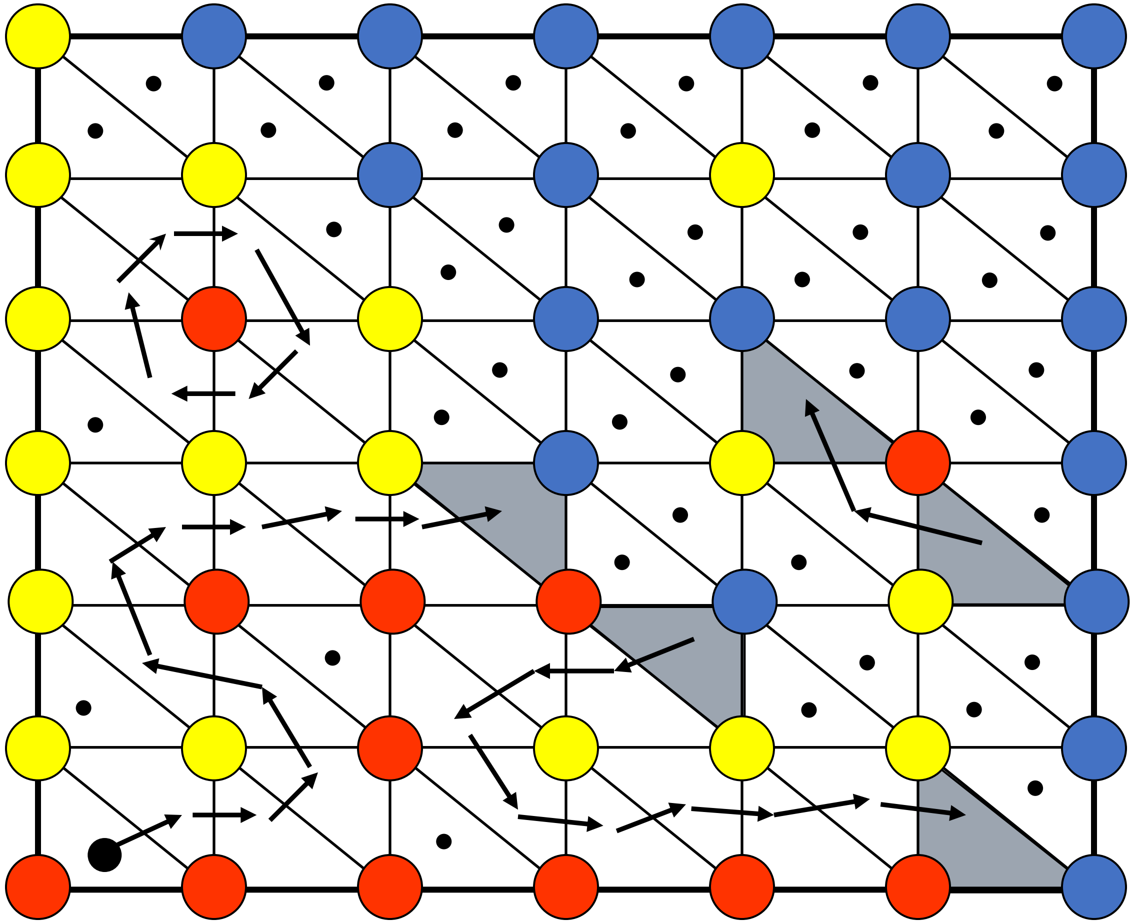
- Reduce an arbitrary instance of SPERNER (which is PPAD-complete, when colors are given by circuit) to local Nash in two-player zero-sum games

Rough Proof Idea: Edge-Triangle Game



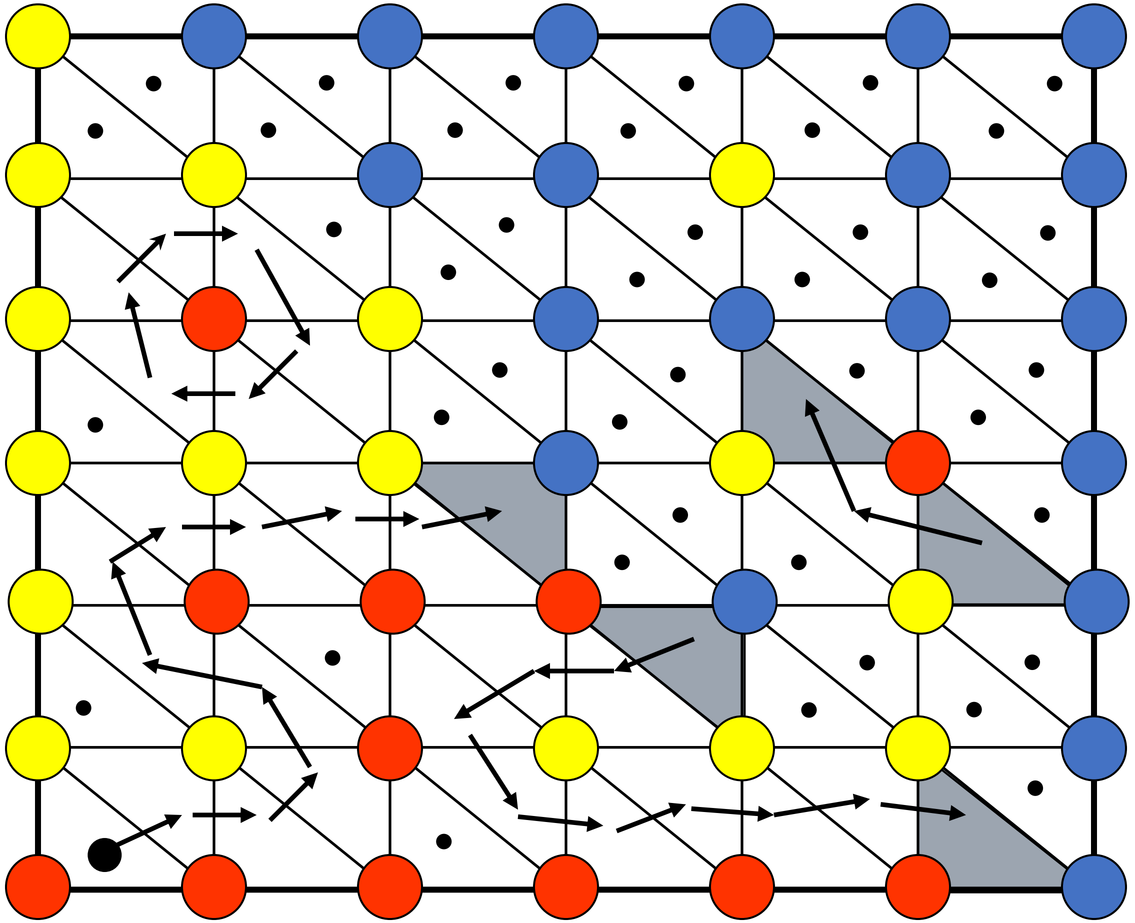
- Reduce an arbitrary instance of SPERNER (which is PPAD-complete, when colors are given by circuit) to local Nash in two-player zero-sum games by having the Max player choose a triangle, the Min player choose an edge of the triangle

Rough Proof Idea: Edge-Triangle Game



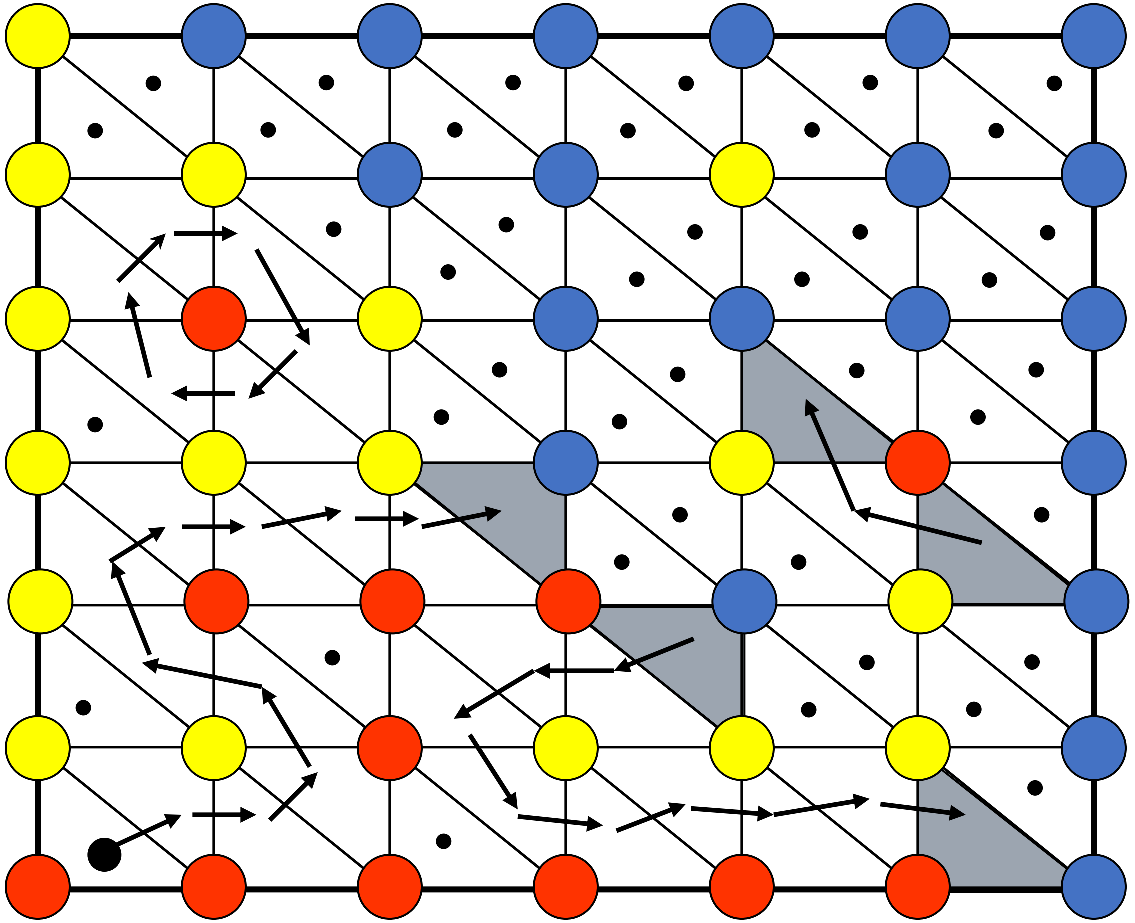
- Reduce an arbitrary instance of SPERNER (which is PPAD-complete, when colors are given by circuit) to local Nash in two-player zero-sum games by having the Max player choose a triangle, the Min player choose an edge of the triangle, and assigning payoffs depending on whether Max chose a triangle that has at least one red-yellow edge,

Rough Proof Idea: Edge-Triangle Game



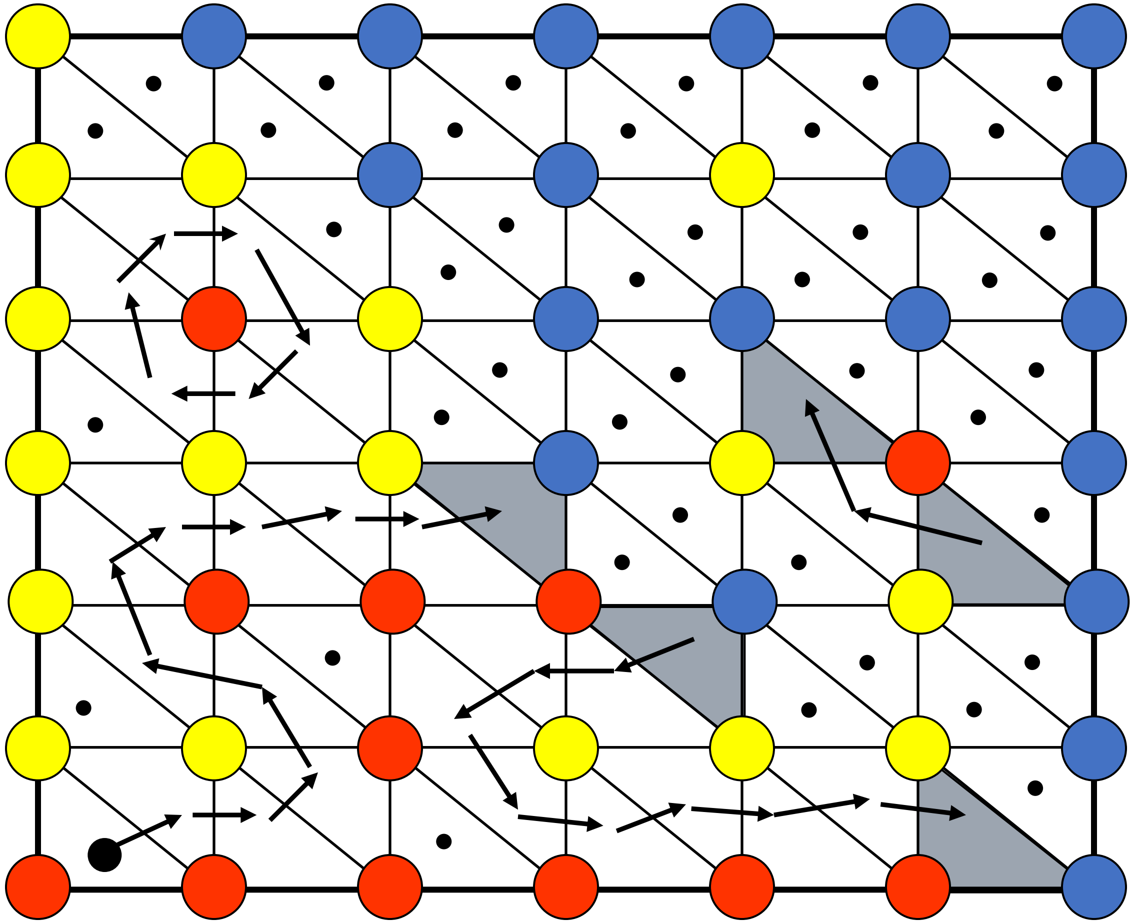
- Reduce an arbitrary instance of SPERNER (which is PPAD-complete, when colors are given by circuit) to local Nash in two-player zero-sum games by having the Max player choose a triangle, the Min player choose an edge of the triangle, and assigning payoffs depending on whether Max chose a triangle that has at least one red-yellow edge, whether Min chose a red-yellow edge in that triangle,

Rough Proof Idea: Edge-Triangle Game



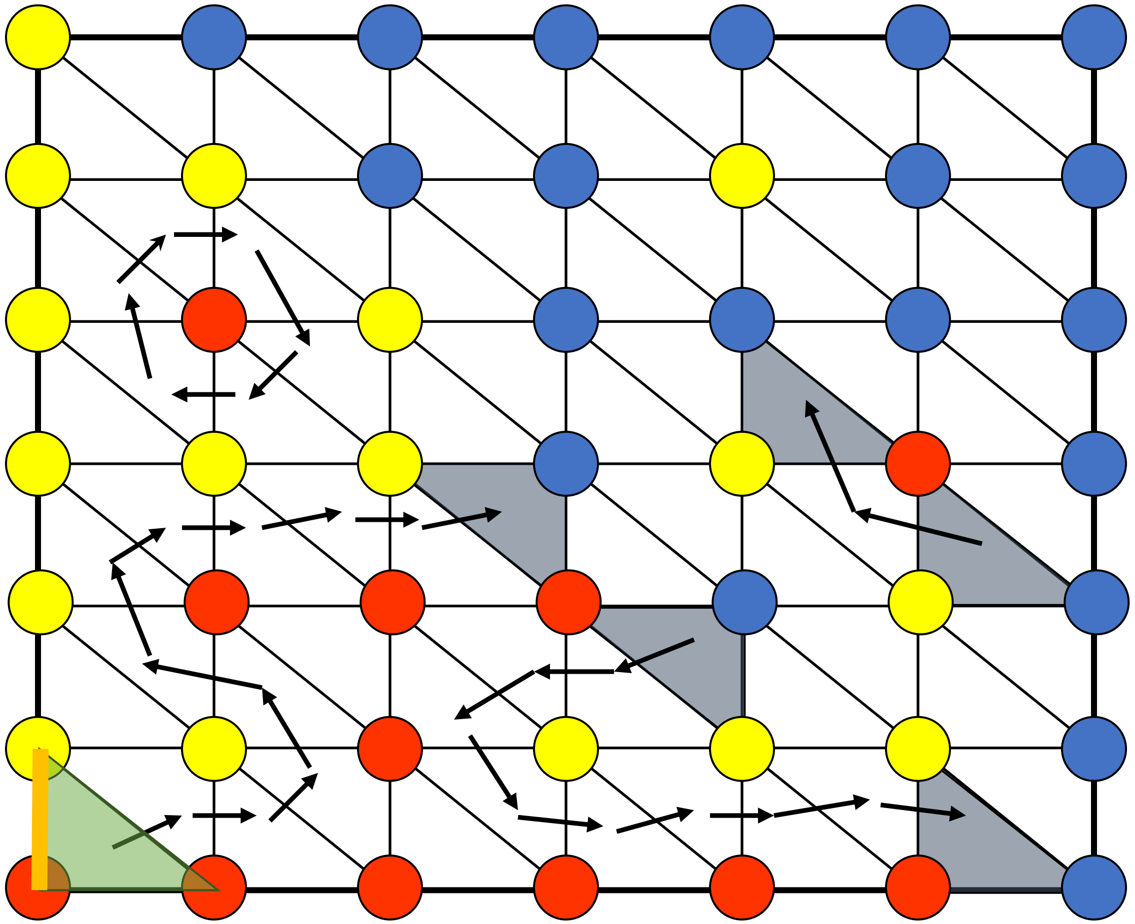
- Reduce an arbitrary instance of SPERNER (which is PPAD-complete, when colors are given by circuit) to local Nash in two-player zero-sum games by having the Max player choose a triangle, the Min player choose an edge of the triangle, and assigning payoffs depending on whether Max chose a triangle that has at least one red-yellow edge, whether Min chose a red-yellow edge in that triangle, as well as the orientation of the chosen edge in that triangle.

Rough Proof Idea: Edge-Triangle Game



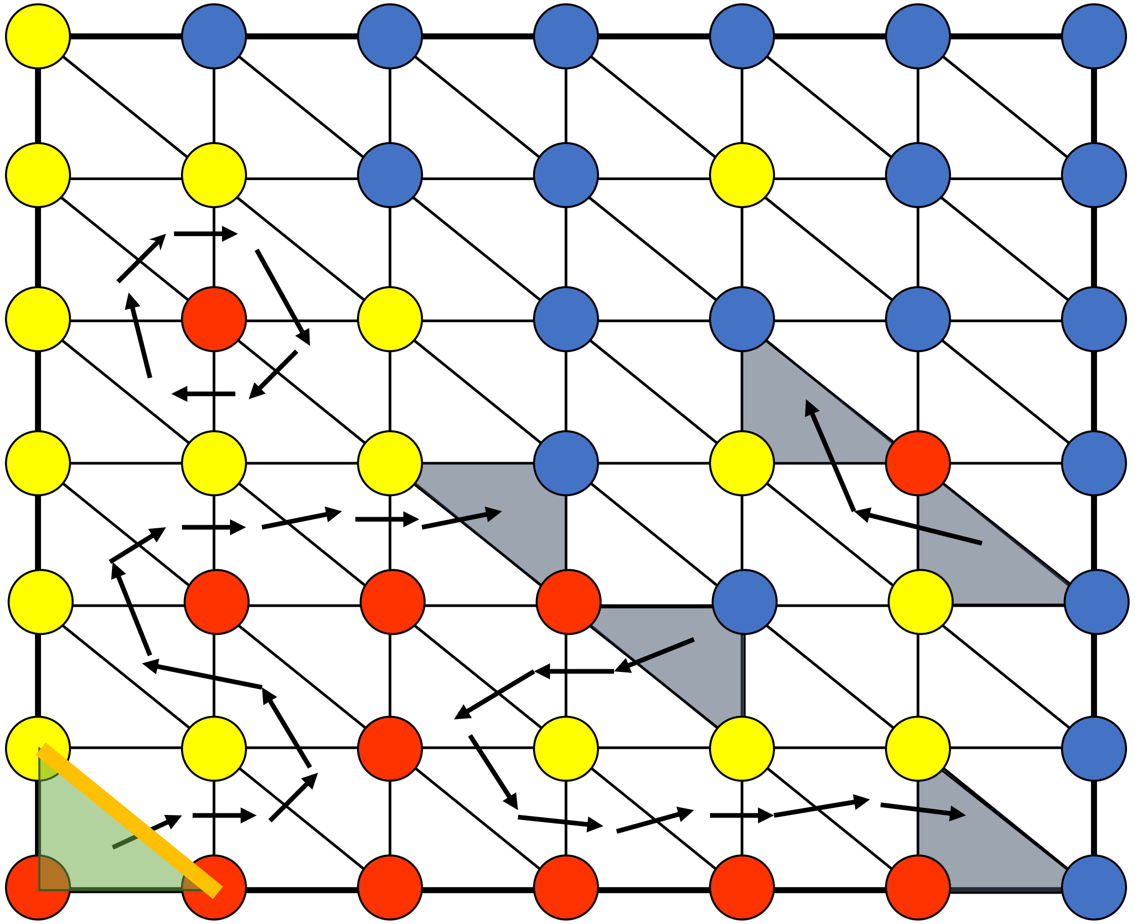
- Reduce an arbitrary instance of SPERNER (which is PPAD-complete, when colors are given by circuit) to local Nash in two-player zero-sum games by having the Max player choose a triangle, the Min player choose an edge of the triangle, and assigning payoffs depending on whether Max chose a triangle that has at least one red-yellow edge, whether Min chose a red-yellow edge in that triangle, as well as the orientation of the chosen edge in that triangle. GOAL: best-response dynamics simulate paths on SPERNER graph

Edge-Triangle Game (Min-Max)



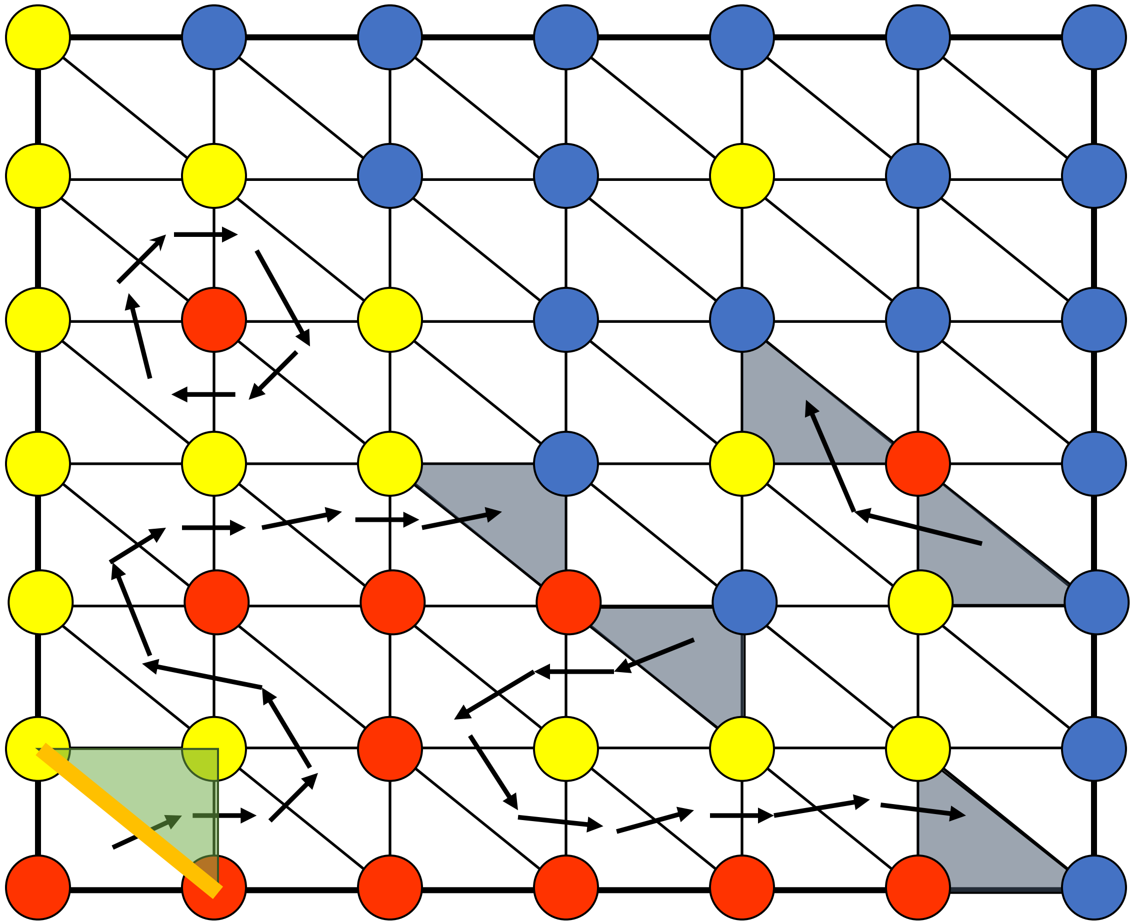
function value = + 1

Edge-Triangle Game (Min-Max)



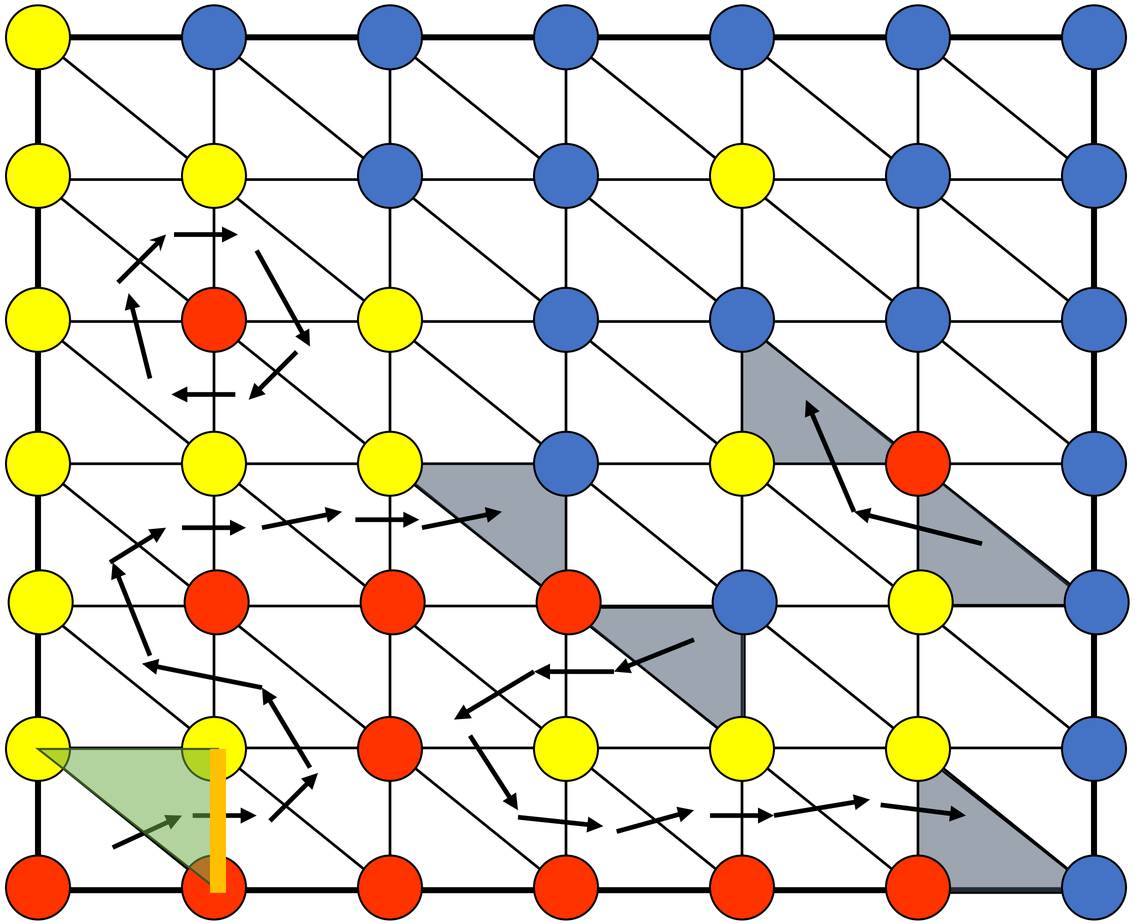
function value = - 1

Edge-Triangle Game (Min-Max)



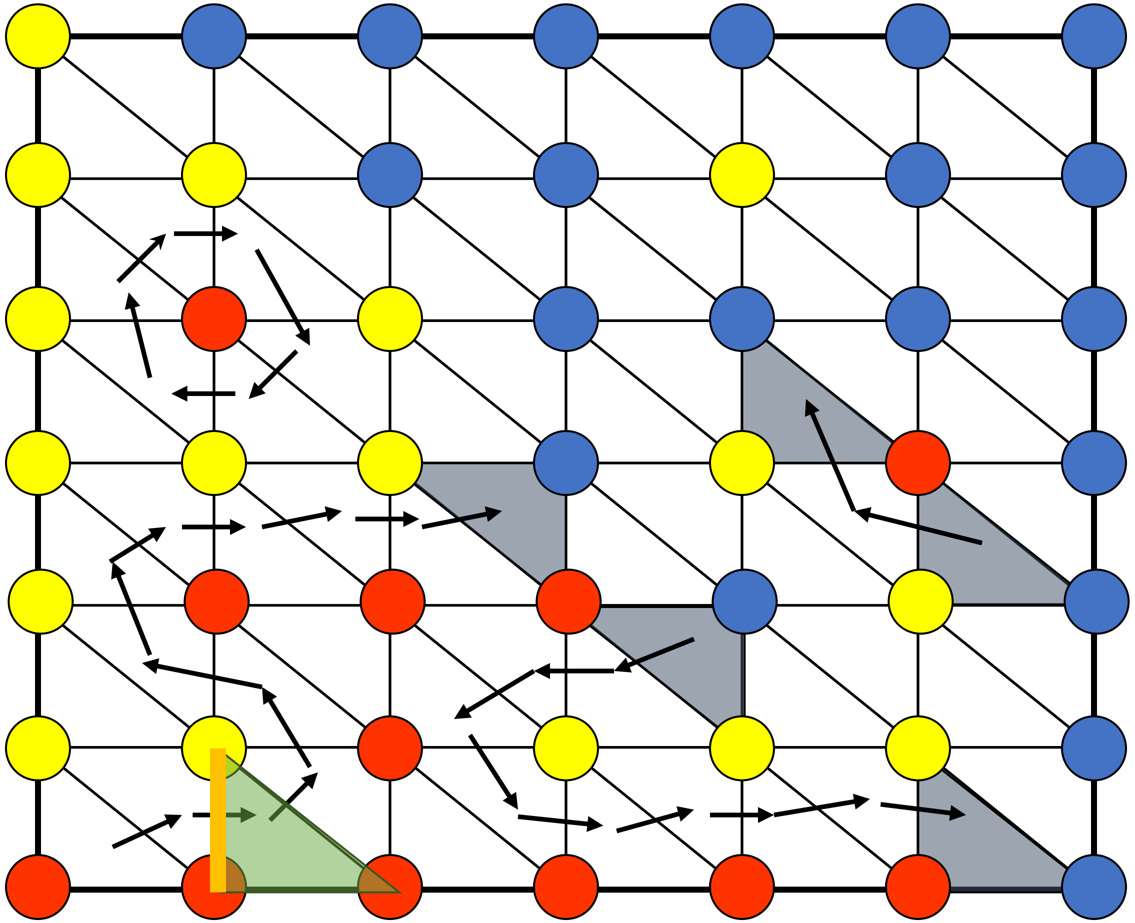
function value = + 1

Edge-Triangle Game (Min-Max)



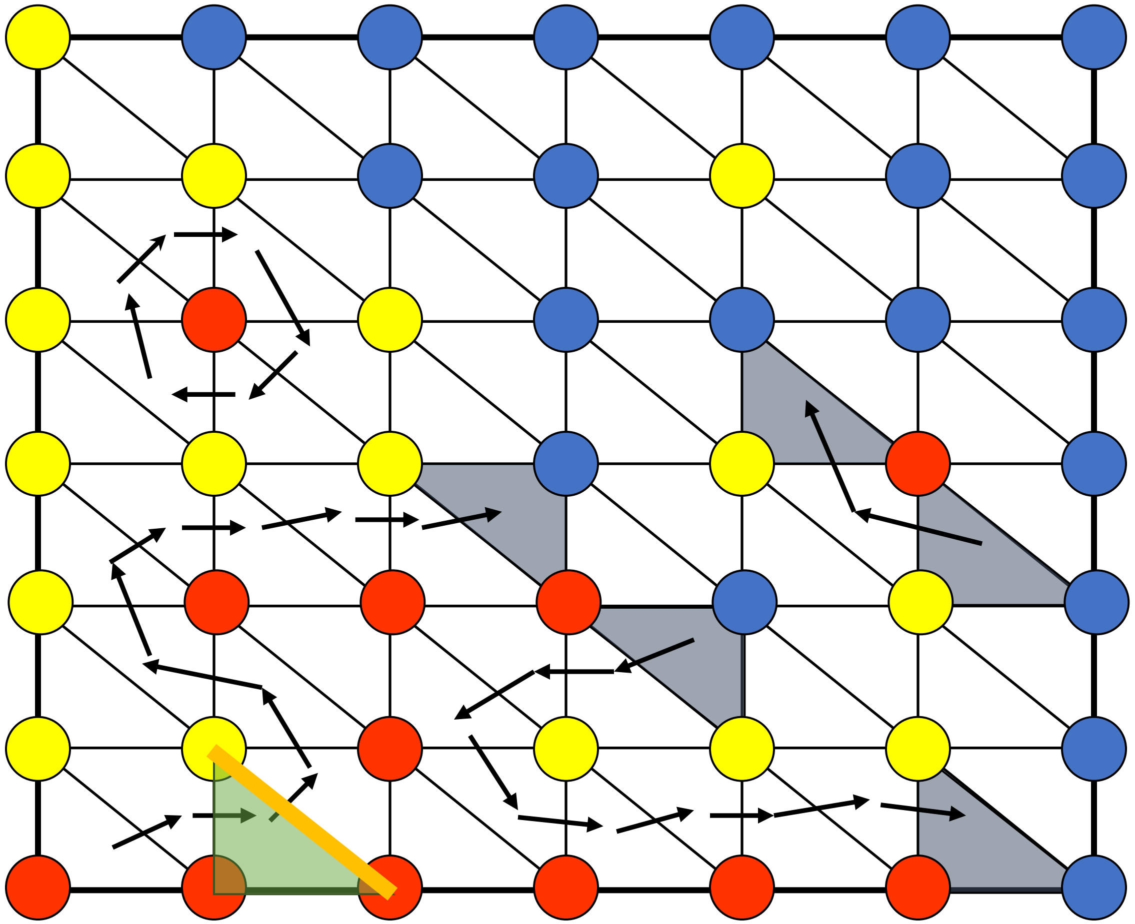
function value = - 1

Edge-Triangle Game (Min-Max)



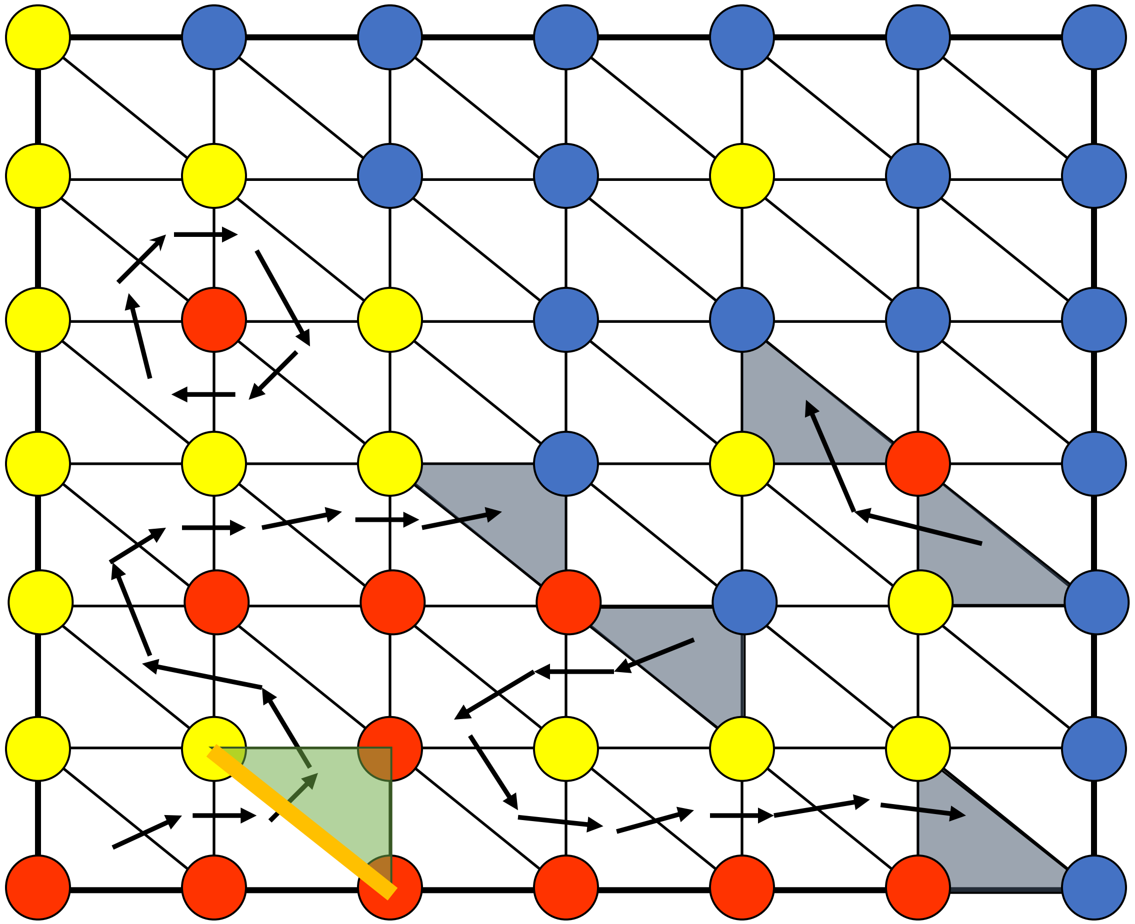
function value = + 1

Edge-Triangle Game (Min-Max)



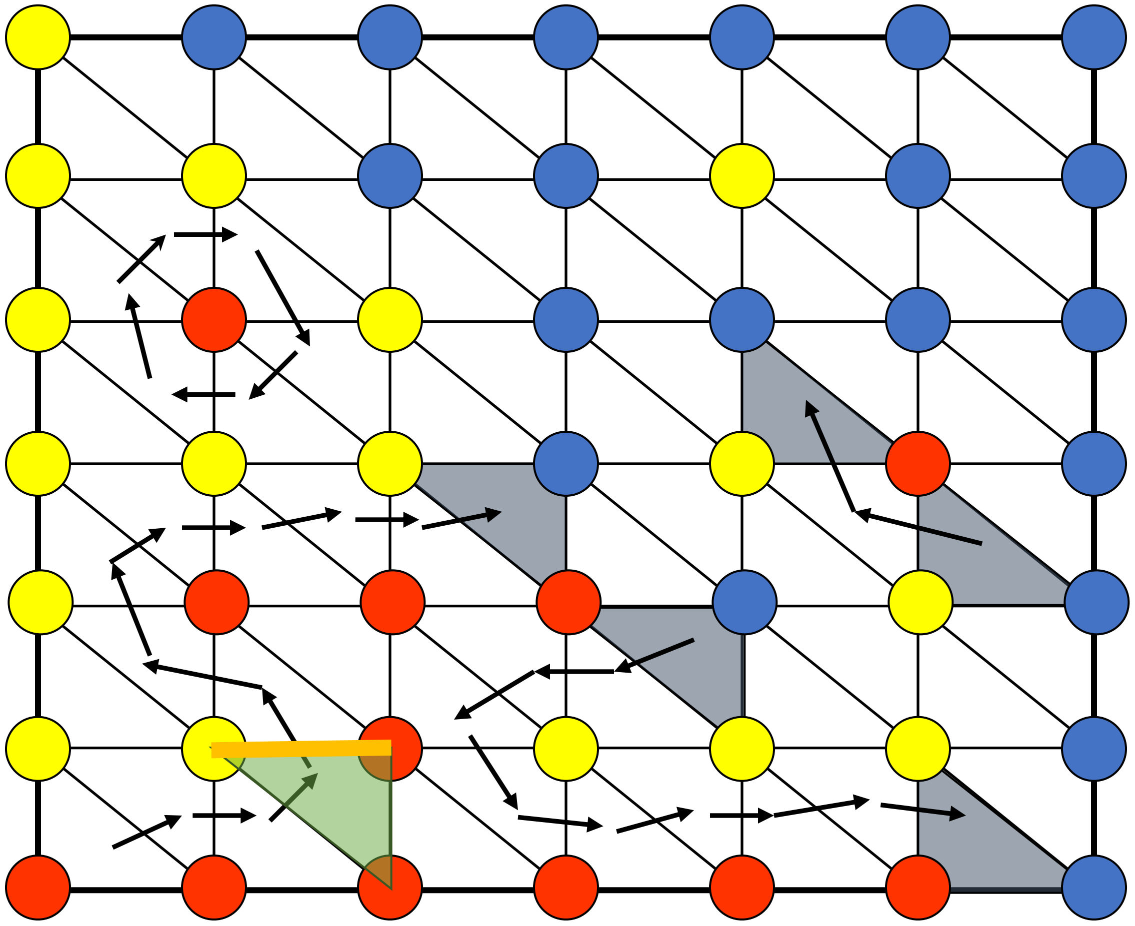
function value = - 1

Edge-Triangle Game (Min-Max)



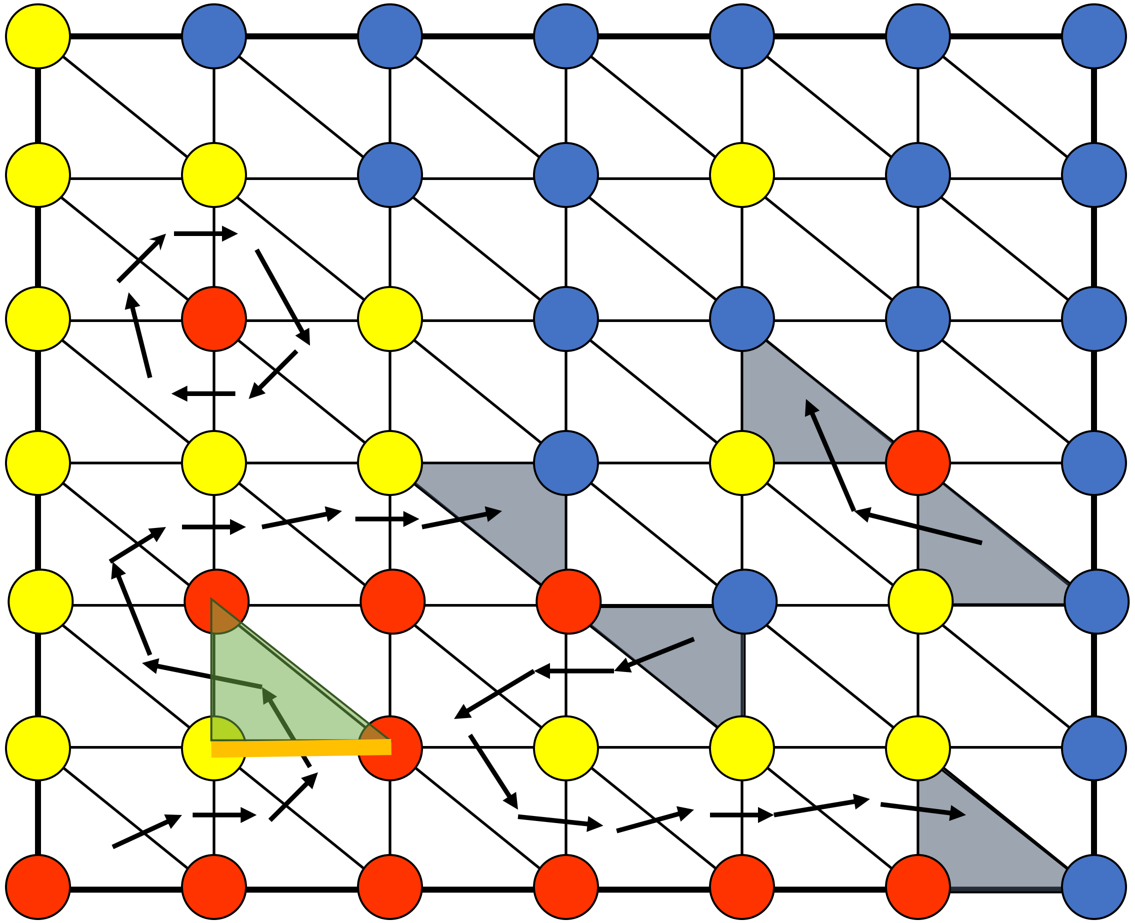
function value = + 1

Edge-Triangle Game (Min-Max)



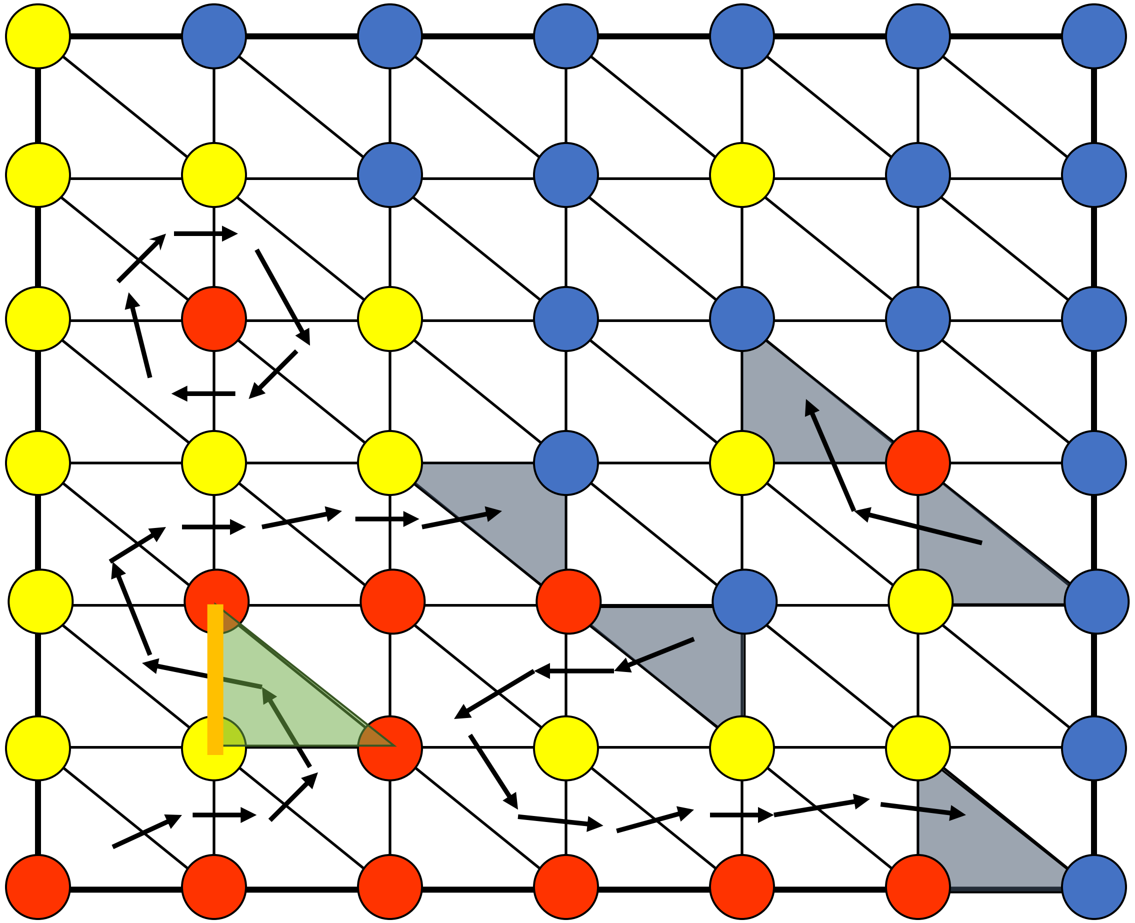
function value = - 1

Edge-Triangle Game (Min-Max)



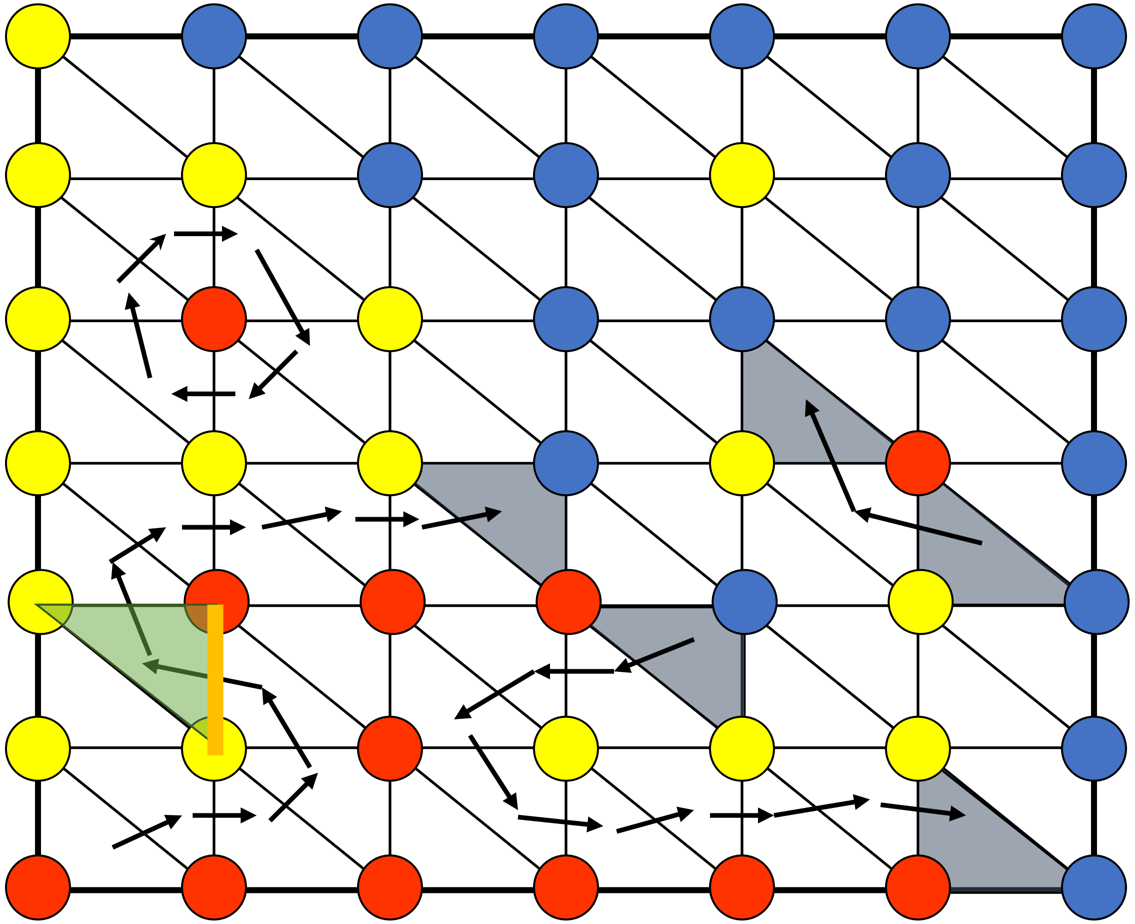
function value = + 1

Edge-Triangle Game (Min-Max)



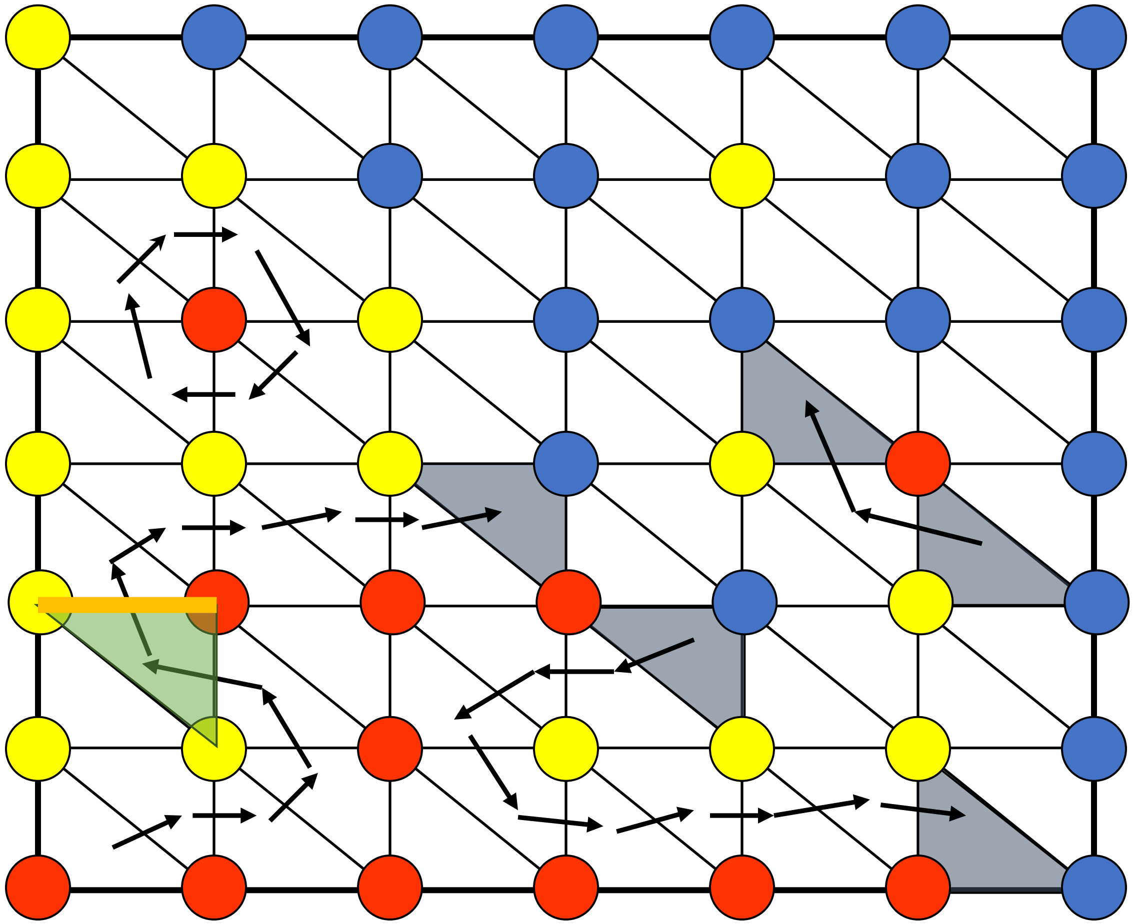
function value = - 1

Edge-Triangle Game (Min-Max)



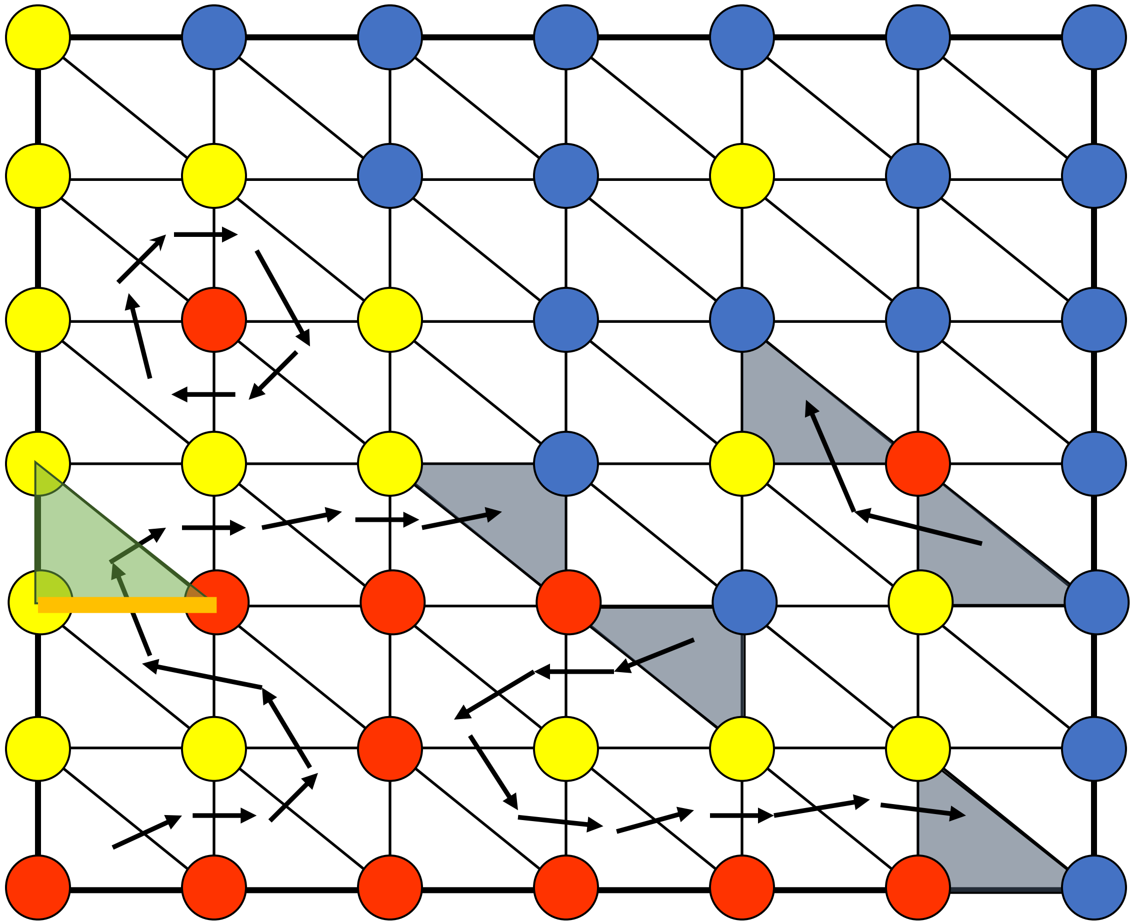
function value = + 1

Edge-Triangle Game (Min-Max)



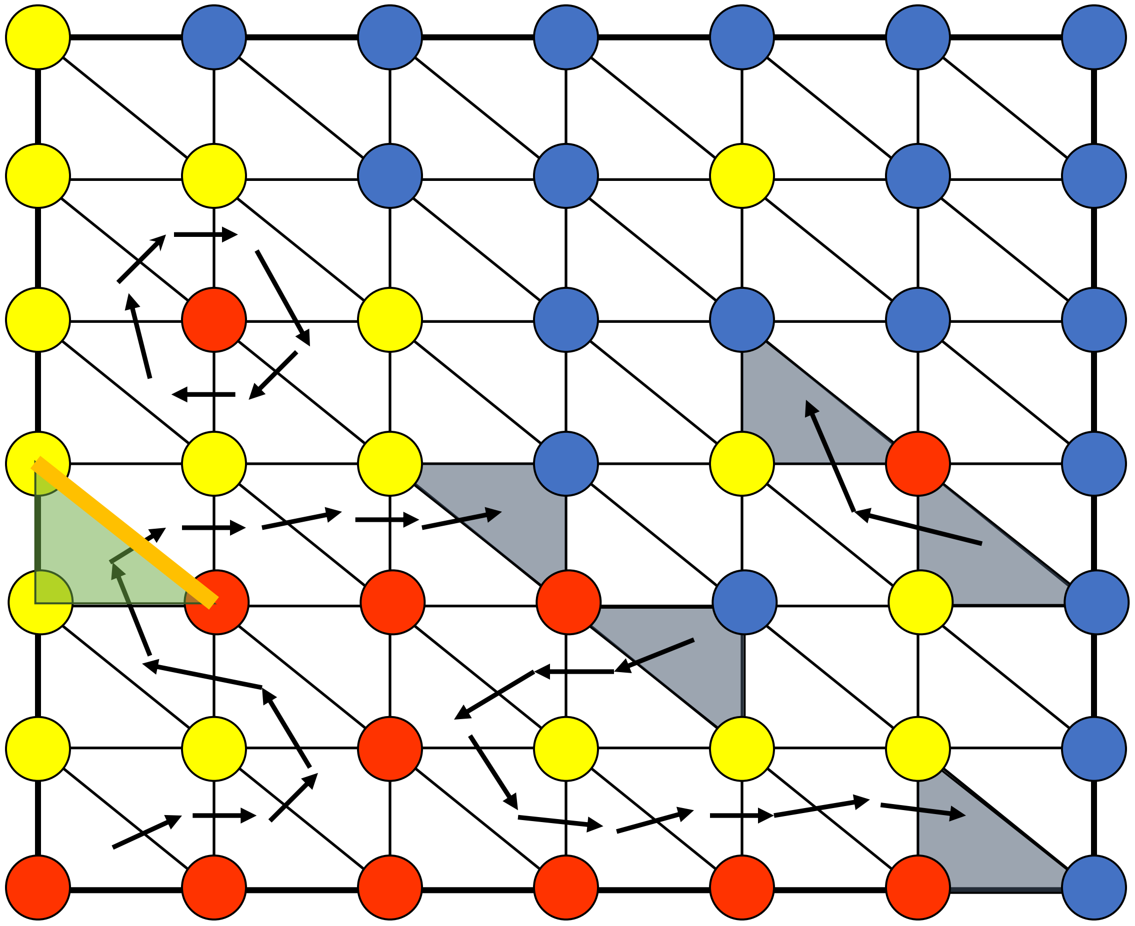
function value = - 1

Edge-Triangle Game (Min-Max)



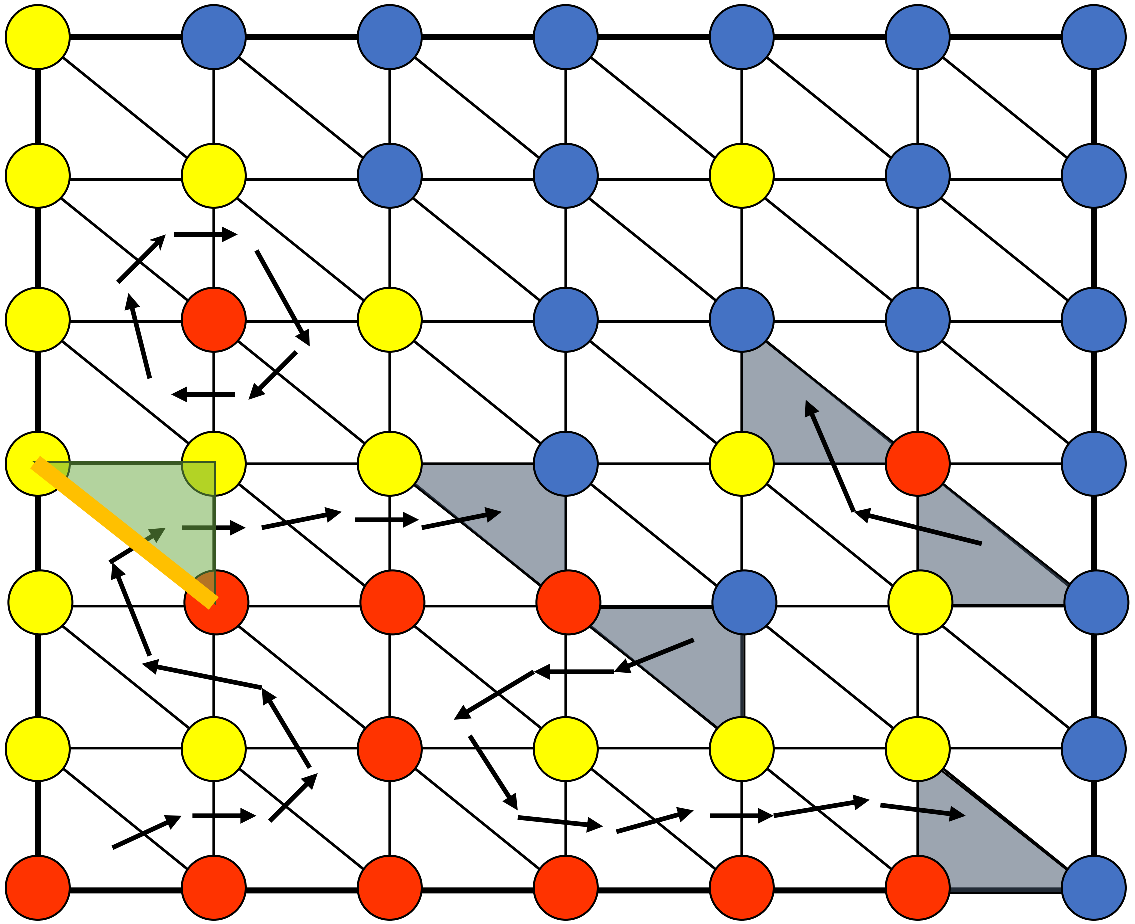
function value = + 1

Edge-Triangle Game (Min-Max)



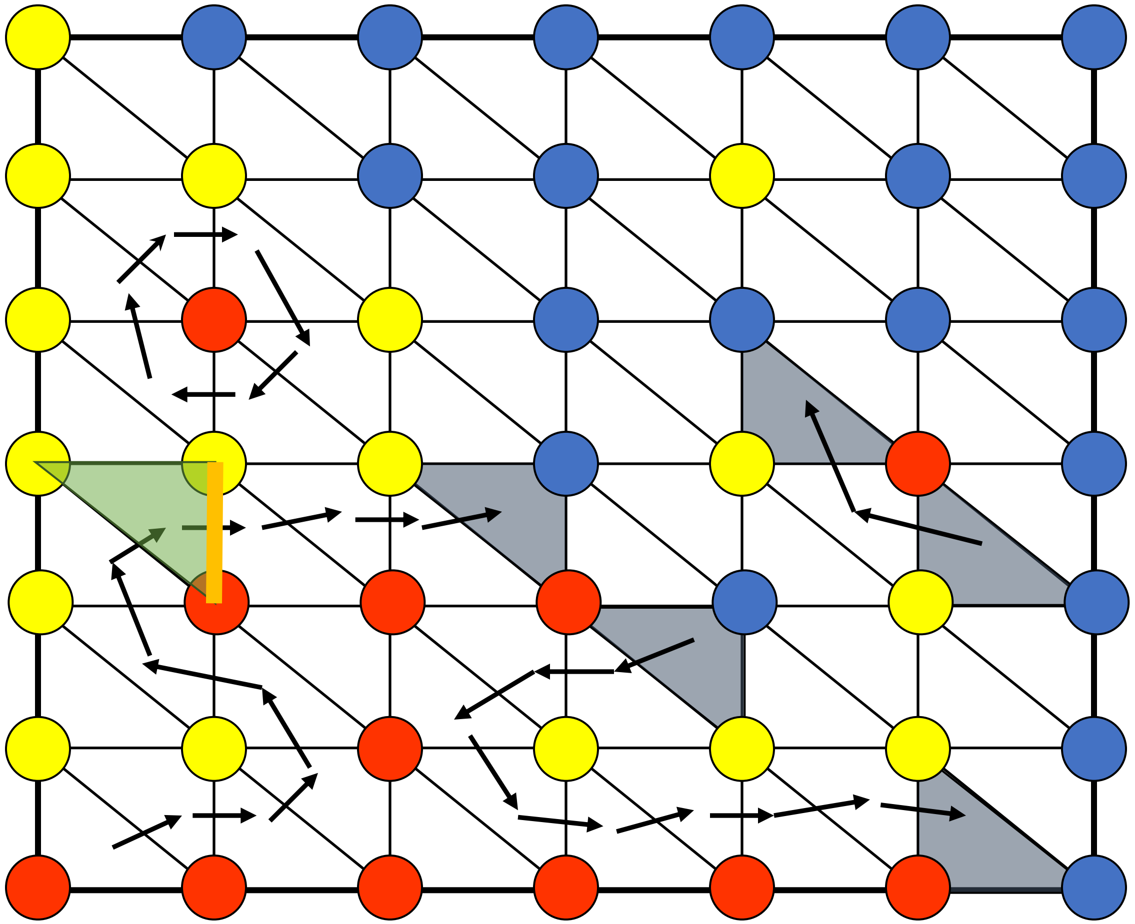
function value = - 1

Edge-Triangle Game (Min-Max)



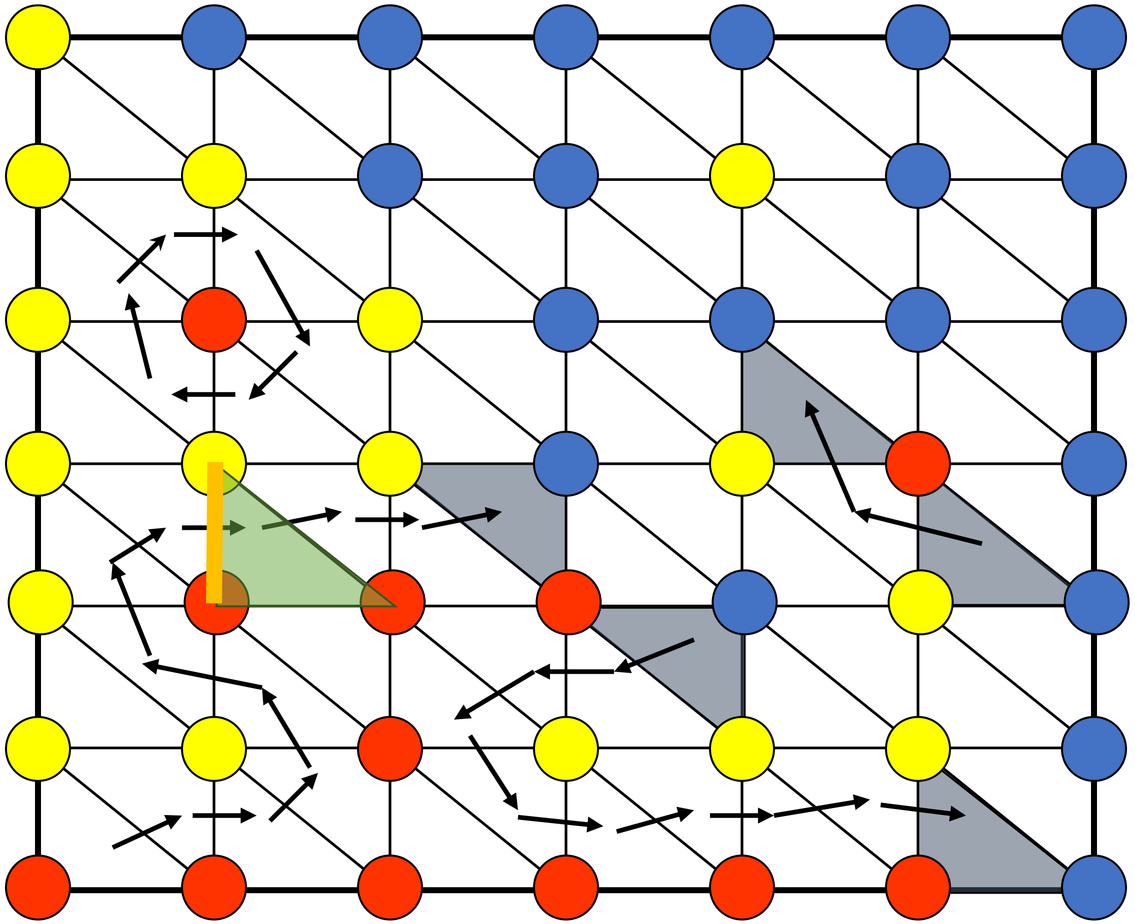
function value = + 1

Edge-Triangle Game (Min-Max)



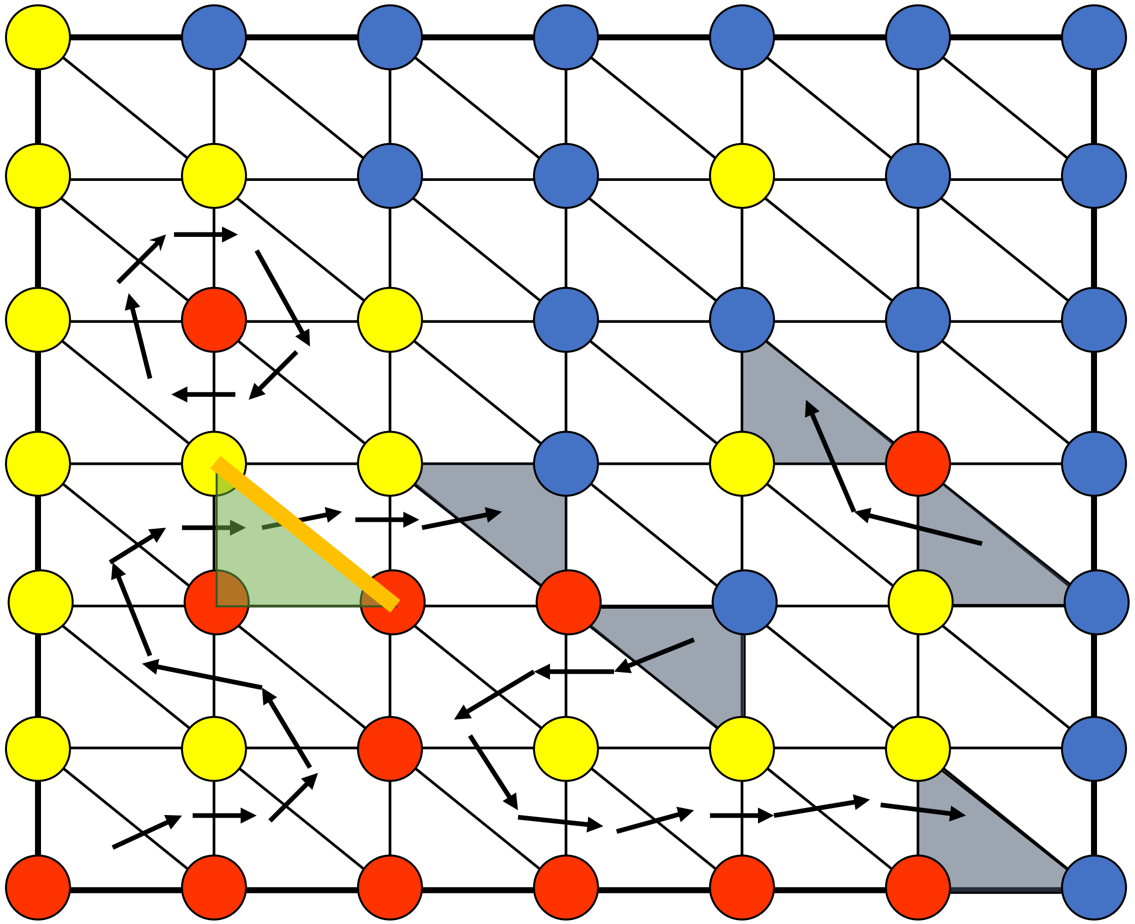
function value = - 1

Edge-Triangle Game (Min-Max)



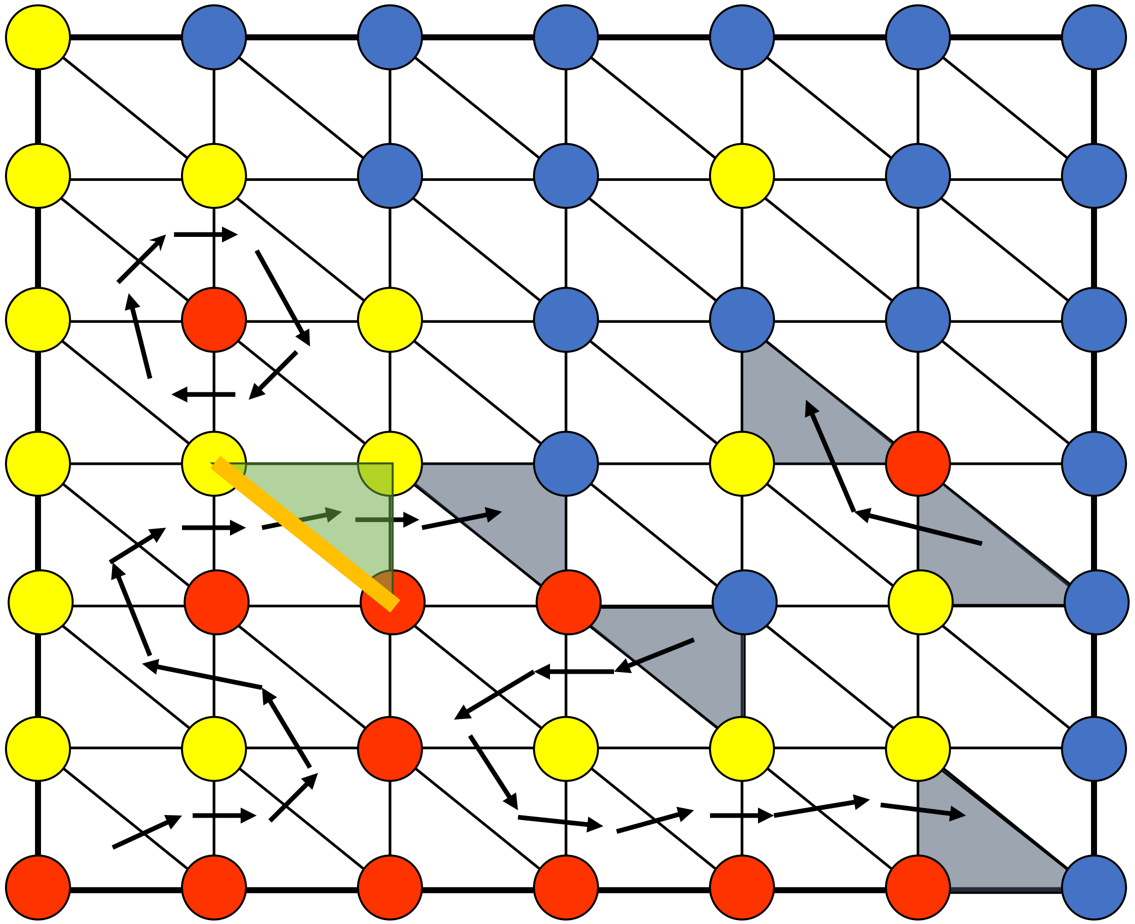
function value = + 1

Edge-Triangle Game (Min-Max)



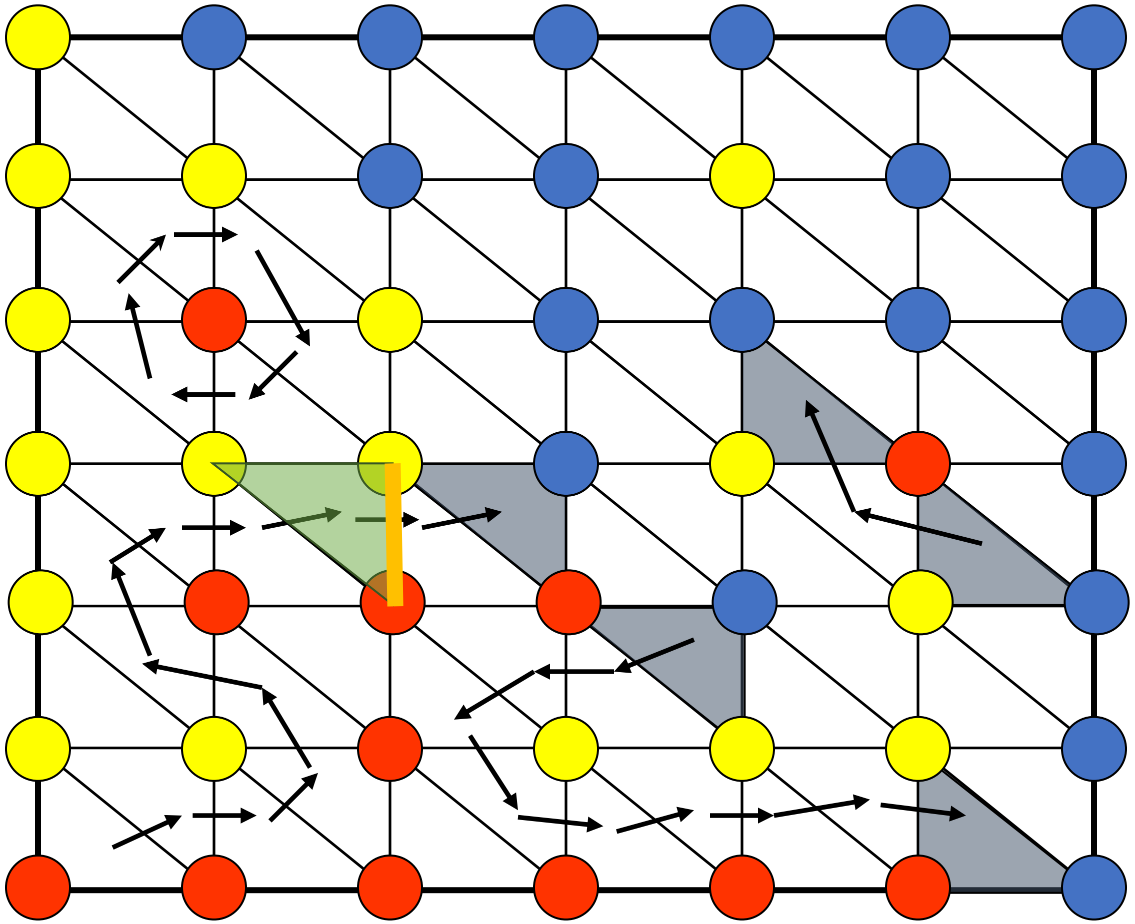
function value = - 1

Edge-Triangle Game (Min-Max)



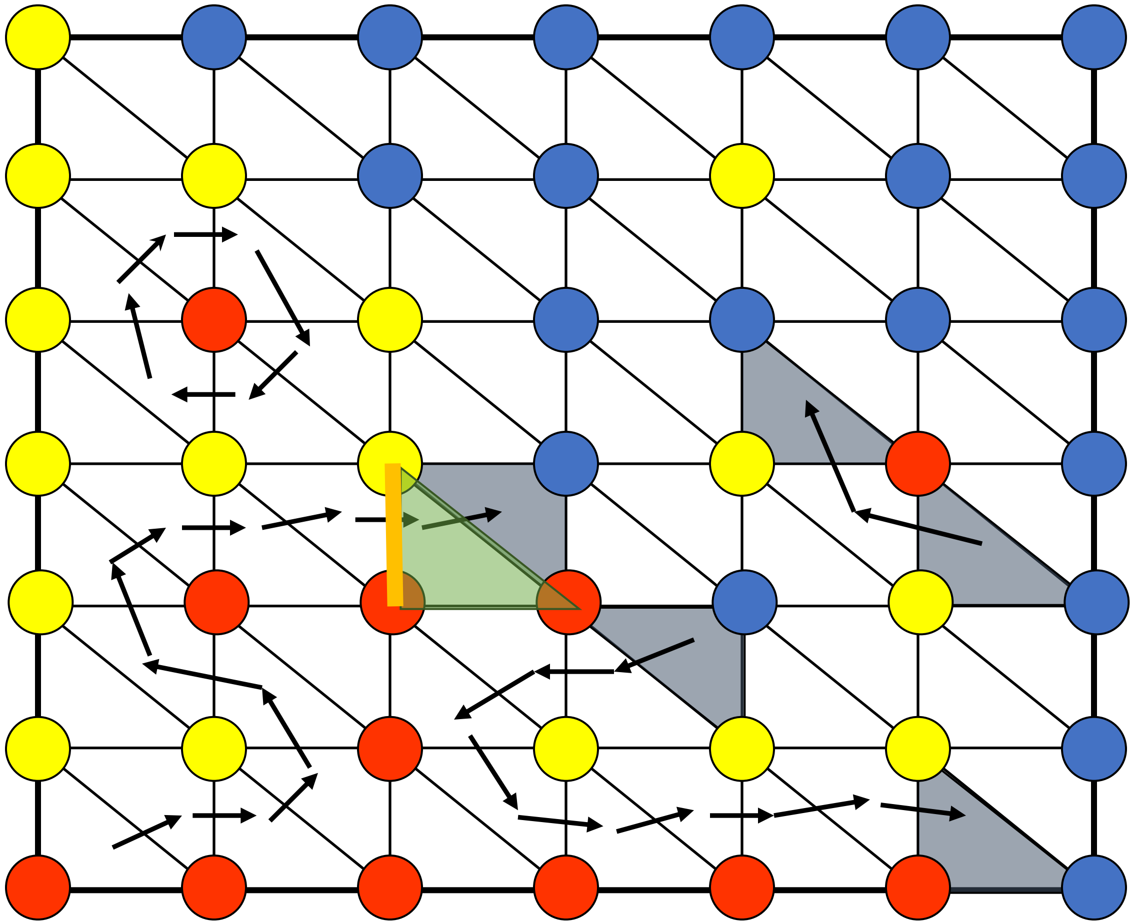
function value = + 1

Edge-Triangle Game (Min-Max)



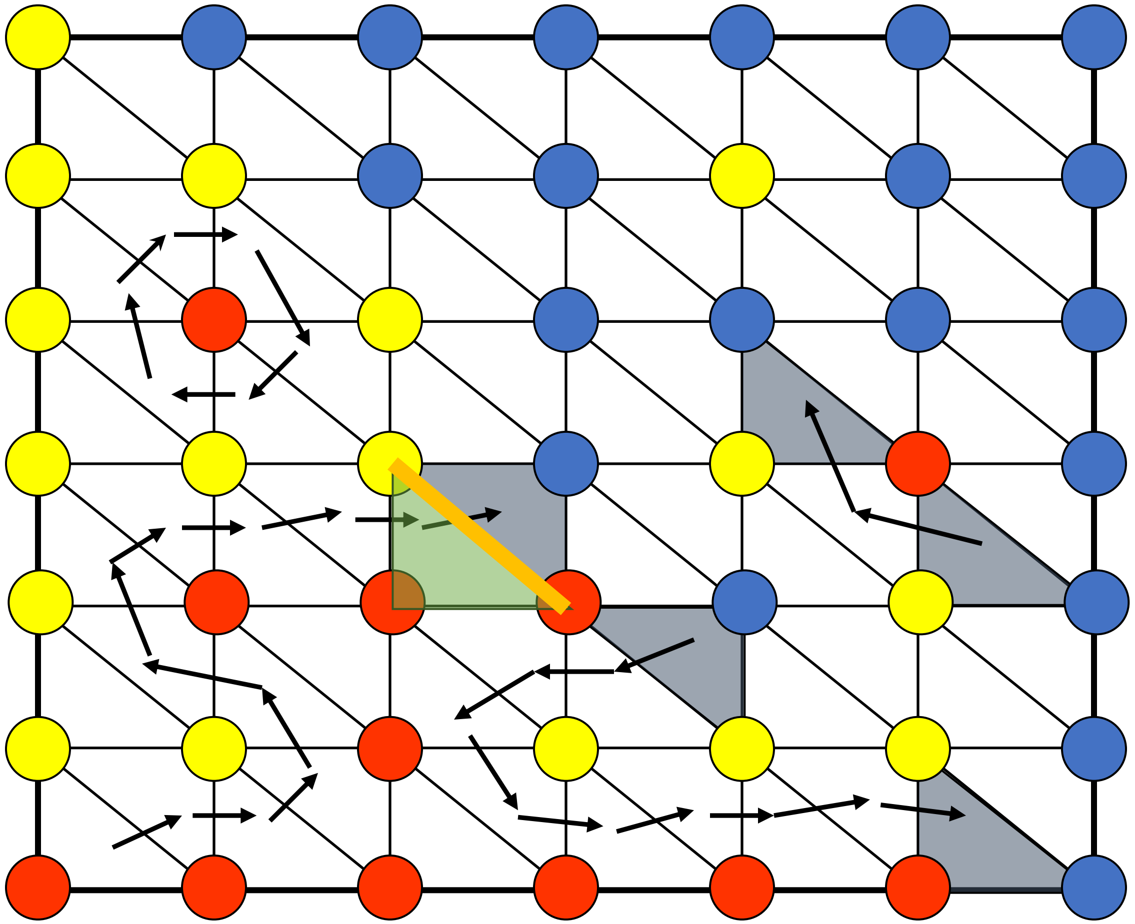
function value = - 1

Edge-Triangle Game (Min-Max)



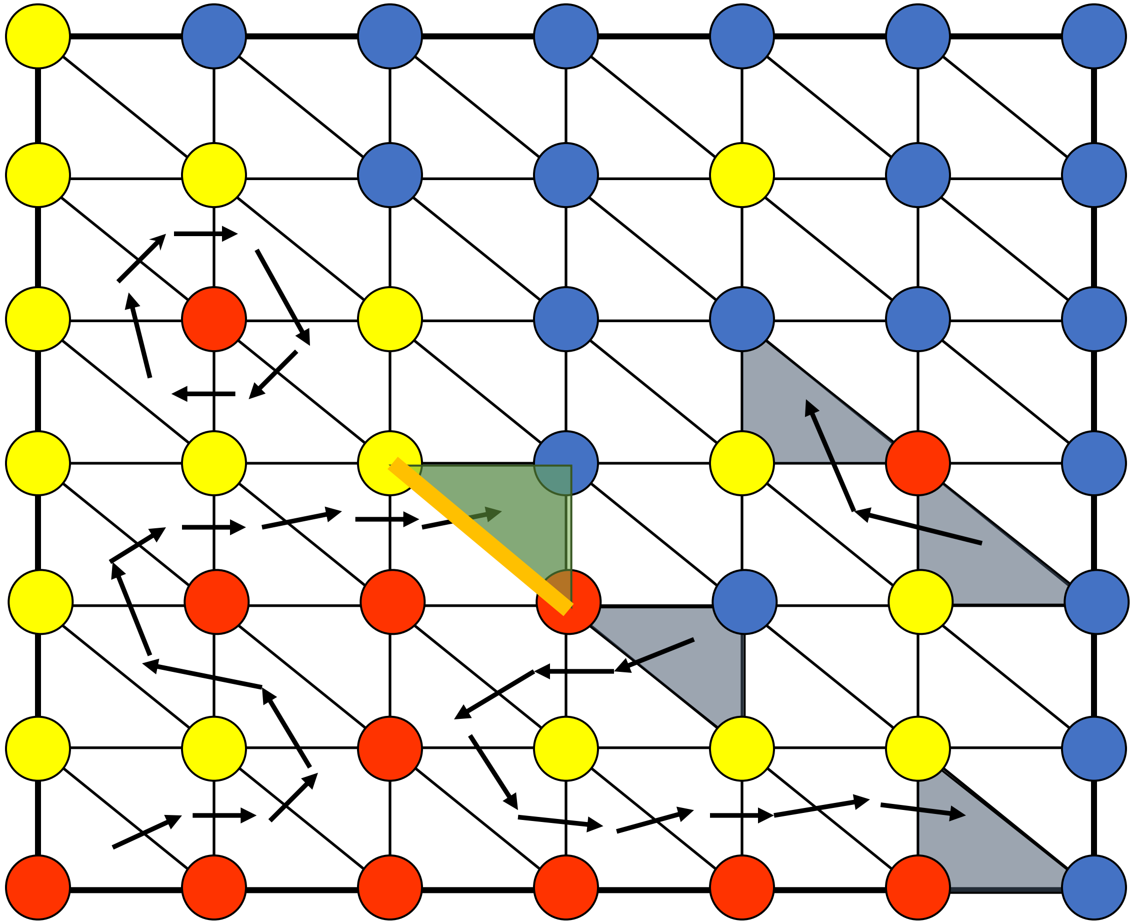
function value = + 1

Edge-Triangle Game (Min-Max)



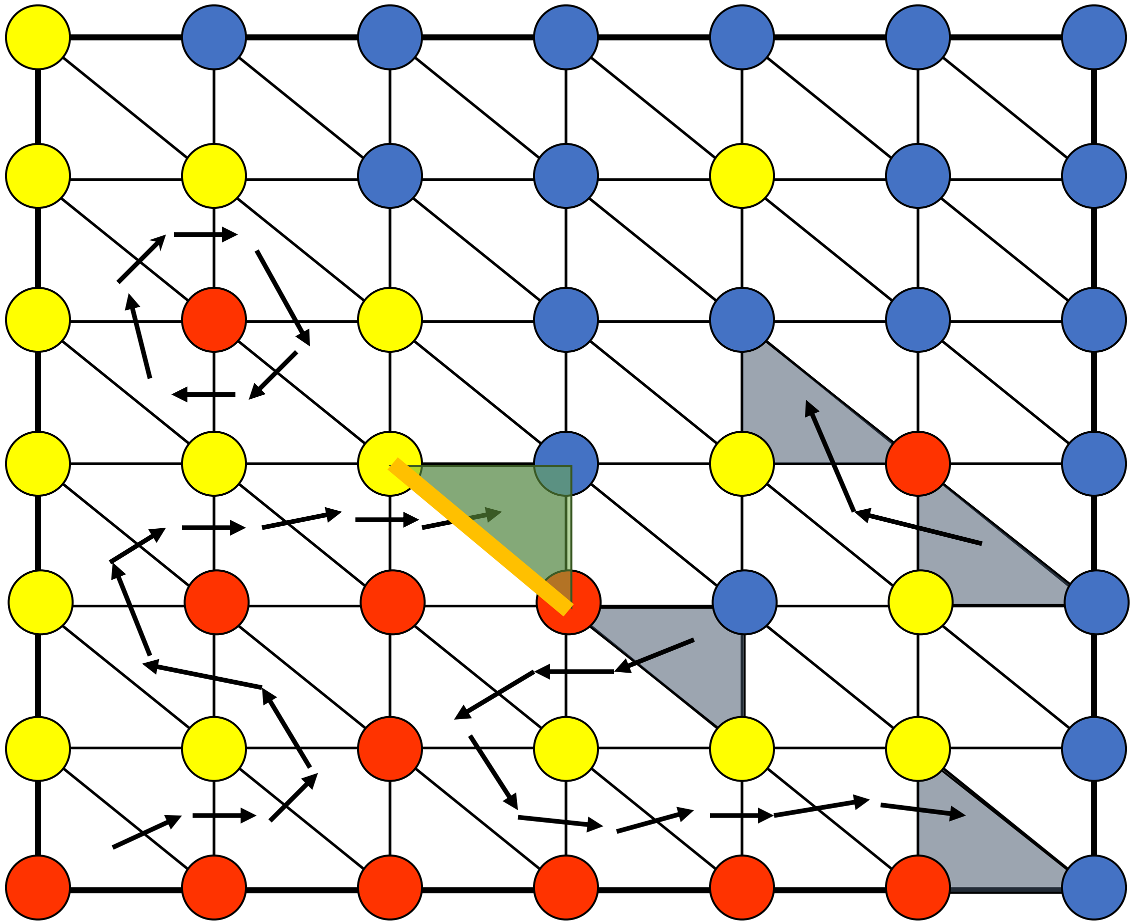
function value = - 1

Edge-Triangle Game (Min-Max)



function value = + 1

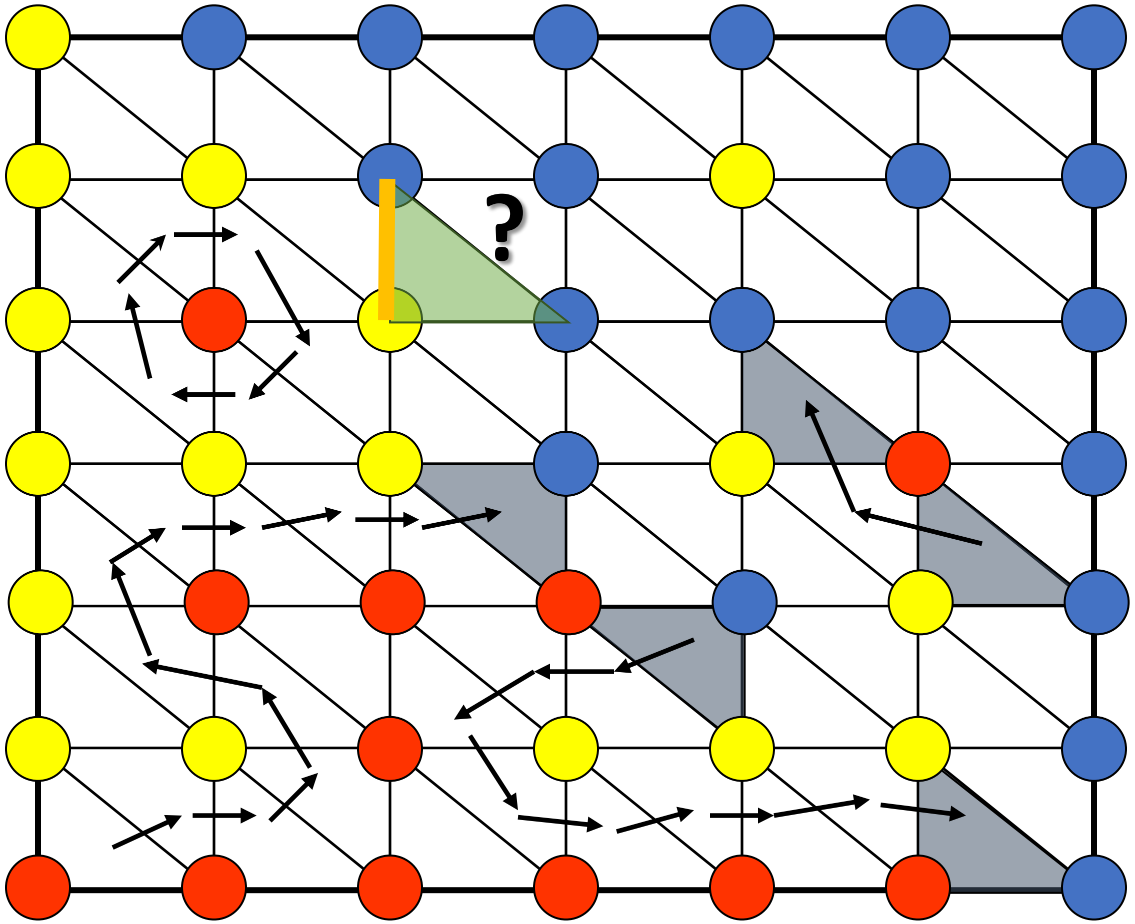
Edge-Triangle Game (Min-Max)



function value = + 1

no edge flip for edge player ⇒ local min-max equilibrium

Edge-Triangle Game (Min-Max)

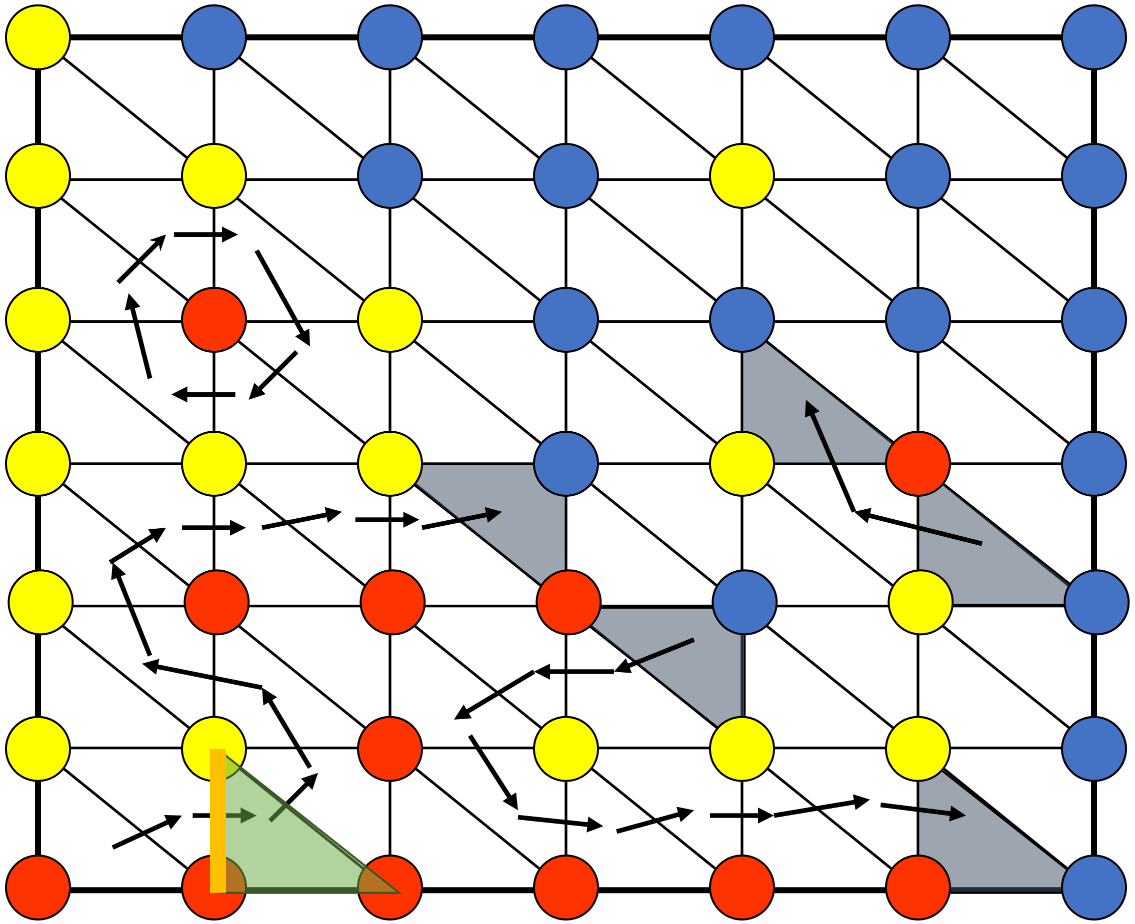


Challenges

1. function value outside the path?

we need to make sure that no spurious solutions are created

Edge-Triangle Game (Min-Max)

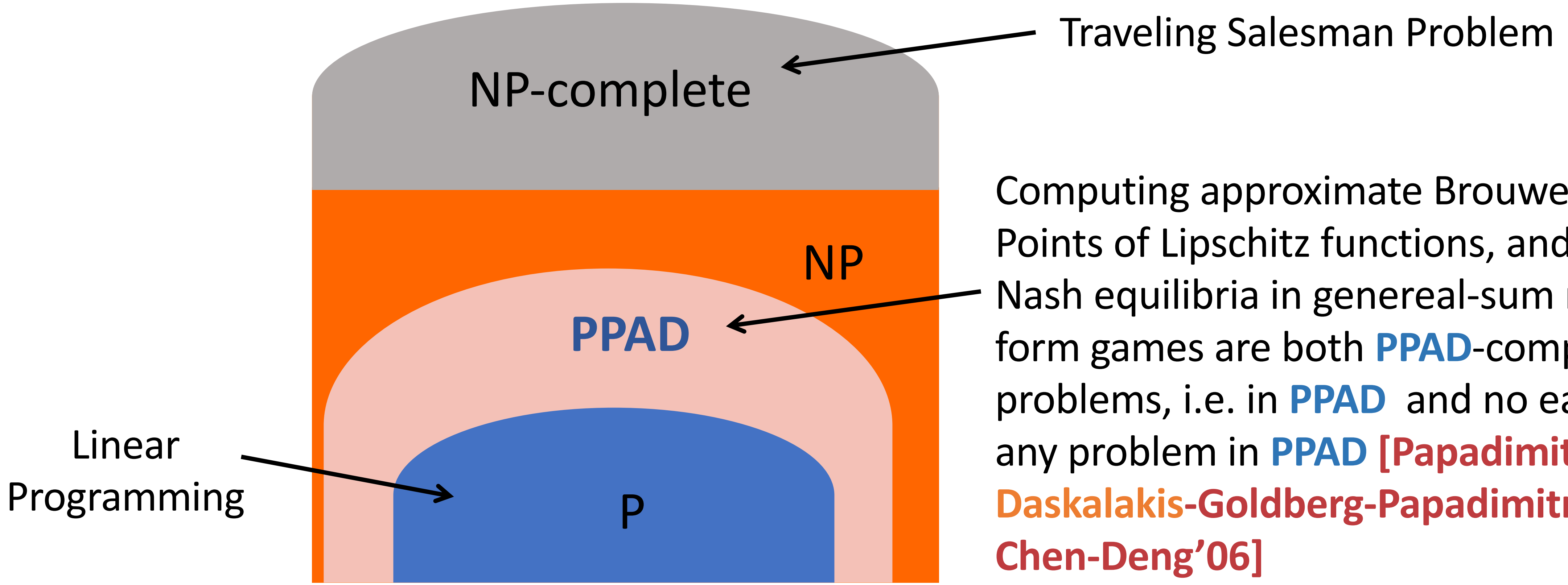


Challenges

2. function needs to be Lipschitz continuous and smooth

challenging problem in high-dimensions!

The Complexity of Local Nash Equilibrium

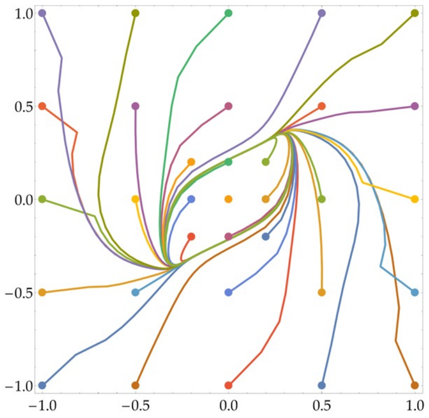


Computing approximate Brouwer Fixed Points of Lipschitz functions, and mixed Nash equilibria in genereal-sum normal-form games are both **PPAD**-complete problems, i.e. in **PPAD** and no easier than any problem in **PPAD** **[Papadimitriou'94, Daskalakis-Goldberg-Papadimitriou'06, Chen-Deng'06]**

[Daskalakis-Skoulakis-Zampetakis STOC'21]: Computing local Nash equilibria *(even in two-player zero-sum and smooth)* non-concave games is exactly as hard as (i) computing approximate Brouwer fixed points of Lipschitz functions; (ii) computing mixed Nash equilibria in general-sum normal-form games; and (iii) at least as hard as any other problem in **PPAD**.

Way Forward: Practical Local Nash Equilibrium

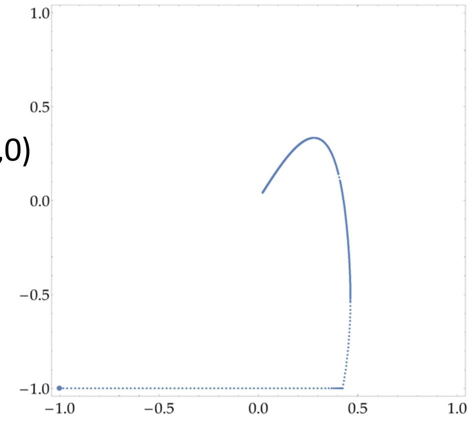
- *Practical Local Nash Equilibrium Computation*?
  - local Nash is intractable in general
  - …but can exploit connection to Brouwer fixed points to obtain 2^{nd}-order dynamics with guaranteed (albeit necessarily not poly-time) convergence **[Daskalakis-Golowich-Skoulakis-Zampetakis COLT'23]**
  - turn it into a 1^{st}-order method by cutting corners ?
  - identify structural properties of games under which it is efficient (beyond worst-case analysis of games)



(a) $f_1(\theta, \omega)$.

gradient descent

local Nash is at (0,0)

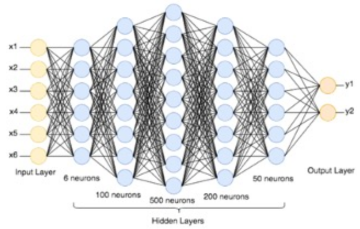


(a) $f_1(\theta, \omega)$.

our algorithm: Stay On the Ridge (or STON'R)

Way Forward: Consider Randomized Equilibria

- *Local* Correlated/Coarse Correlated equilibria?
  - what’s a reasonable way to define it in general non-concave games?
  - …so that it is also guaranteed to exist and is tractable?
  - proposal: $||\mathbb{E}_{x^*\sim p}[\nabla_{x_i} u_i(x_i^*; x_{-i}^*)]|| \leq \varepsilon$ (formally: project to the constraint set)
  - when p has support 1 this is a local Nash eq, so this exists but is intractable
  - is there some polynomial support, so that it is tractable?
  - **[Cai-Daskalakis-Luo-Wei-Zhang’23]:** If $\mathcal{S}$ is convex and compact and the u_i’s are Lipschitz and and smooth, a poly-size supported (in the dimension, in $1/\varepsilon$, in the Lipschitzness and the smoothness of the utilities) local CCE exists can be computed efficiently (using Gradient Descent) ☺



semi-agnostic

+ $x_{t+1} \leftarrow x_t - \nabla_x \ell(x_t)$ +

+

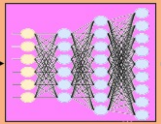
Next Time: Global Randomized Equilibria



Multi-player Game-Playing:
- Superhuman GO, Poker, Gran Turismo
- Human-level Starcraft, Diplomacy

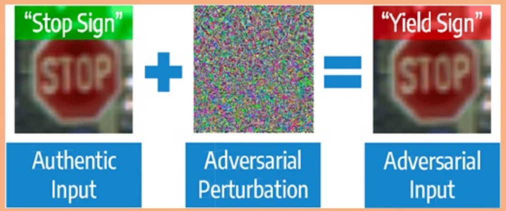


- Multi-robot interactions
- Autonomous driving
- Automated Economic policy design

$Z \sim \mathcal{N}(0, I) \longrightarrow$  $\longrightarrow$  $\sim P_{\text{target}}$

boring randomness — neural network — interesting randomness

Generative Adversarial Networks (GANs)
synthetic data generation



Adversarial Training
robustifying models against adversarial attacks