Computational Complexity and Game Theory

Mahya Jamshidian

Outline

- Preliminary
- Function Complexity classes
- **MNE** is **PPAD**-Complete
- Conclusion

Preliminary

- Decision Problems
- P and NP
- Completeness
- NP-Complete Problems
- Co-NP

Decision Complexity

A **decision problem** is a problem that can be posed as a yes-no question of the input values.

- Is *m*, a number, prime?
- Does graph *G* has a Hamiltonian path?
- Does the TM *M* halt on the input *w*?

Decision Complexity

A **decision problem** is a problem that can be posed as a yes-no question of the input values.

P	NP

Decision Complexity

A language ***L*** is in **P** if and only if there exists a deterministic Turing machine ***M***, such that

- ***M*** runs for polynomial time on all inputs
- For all ***x*** in ***L***, ***M*** outputs **1**
- For all ***x*** not in ***L***, ***M*** outputs **0**

A language ***L*** is in **NP** if and only if there exist polynomials ***p*** and ***q***, and a deterministic TM ***M***, such that

- For all ***x*** and ***y***, the ***M*** runs in time ***p*(|*x*|)** on input
- For all ***x*** in ***L***, there exists a string ***y*** of length ***q*(|*x*|)** such that **M(x,y)=1**
- For all ***x*** not in ***L*** and all strings ***y*** of length ***q*(|*x*|)**, **M(x,y) = 0**

Decision Complexity

P

$\stackrel{?}{=}$

NP

Completeness

A problem ***p*** is called **hard** for a complexity class ***C*** under a given type of reduction if there exists a reduction (of the given type) from any problem in ***C*** to ***p***. If a problem is both **hard** for the class and a member of the class, it is **complete** for that class (for that type of reduction).

NP-Complete problems

- SAT
- Clique
- TSP
- Hamiltonian Path
- Independent Set

Co-NP

- **NP** is not known to be closed under complement
  - Known as **NP vs Co-NP** problem
  - **TAUT**

Function Complexity

- Functions and Computation
- FP and FNP
- TFNP
- PPA, PPAD, PPP, PLS

Function Complexity

A function problem is a problem that given input x, has to return y such that some relation R(x, y) holds, and reject O.W.

- The actual solution to SAT: FSAT
- Return the IS of graph G, if exists.
- Factorize integer I

Function Complexity

FP	FNP

Decision Complexity

A binary relation **P(*x*,*y*)** is in **FP** if and only if there is a deterministic polynomial time algorithm that, given ***x***, can find some ***y*** such that **P(*x*,*y*)** holds.

A binary relation **P(*x*,*y*)**, where ***y*** is at most polynomially longer than ***x***, is in FNP if and only if there is a deterministic polynomial time algorithm that can determine whether **P(*x*,*y*)** holds given both ***x*** and ***y***.

Function Complexity

FP	$\stackrel{?}{=}$	FNP

Some remarks

- **FSAT** is **FNP-Complete**
- **SAT** is self-reducible
- Any **NP-C** problem is self-reducible
- Conjecture: IntFact is not self-reducible
- ***NP* = *P*** if and only if ***FP* = *FNP***
- *If **A** is in **NPC** and in **TFNP*** Hence ***NP*** = ***CoNP***

Total functions

A function defined for all possible input values.

TFNP

Total functions

A function defined for all possible input values.

A binary relation **P(*x*,*y*)** is in **TFNP** if and only if there is a deterministic polynomial time algorithm that can determine whether **P(*x*,*y*)** holds given both ***x*** and ***y***, and for every ***x***, there exists a ***y*** which is at most polynomially longer than ***x*** such that **P(*x*,*y*)** holds.

The Class PLS

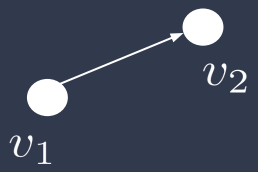
"Every DAG has a sink."

Suppose that a DAG with vertex set $\{0,1\}^n$ is defined by two circuits:

node id $\longrightarrow$ $\boxed{C}$ $\longrightarrow$ {node id_1, …, node id_k}

node id $\longrightarrow$ $\boxed{F}$ $\longrightarrow$ $\mathbb{R}$ [redacted]

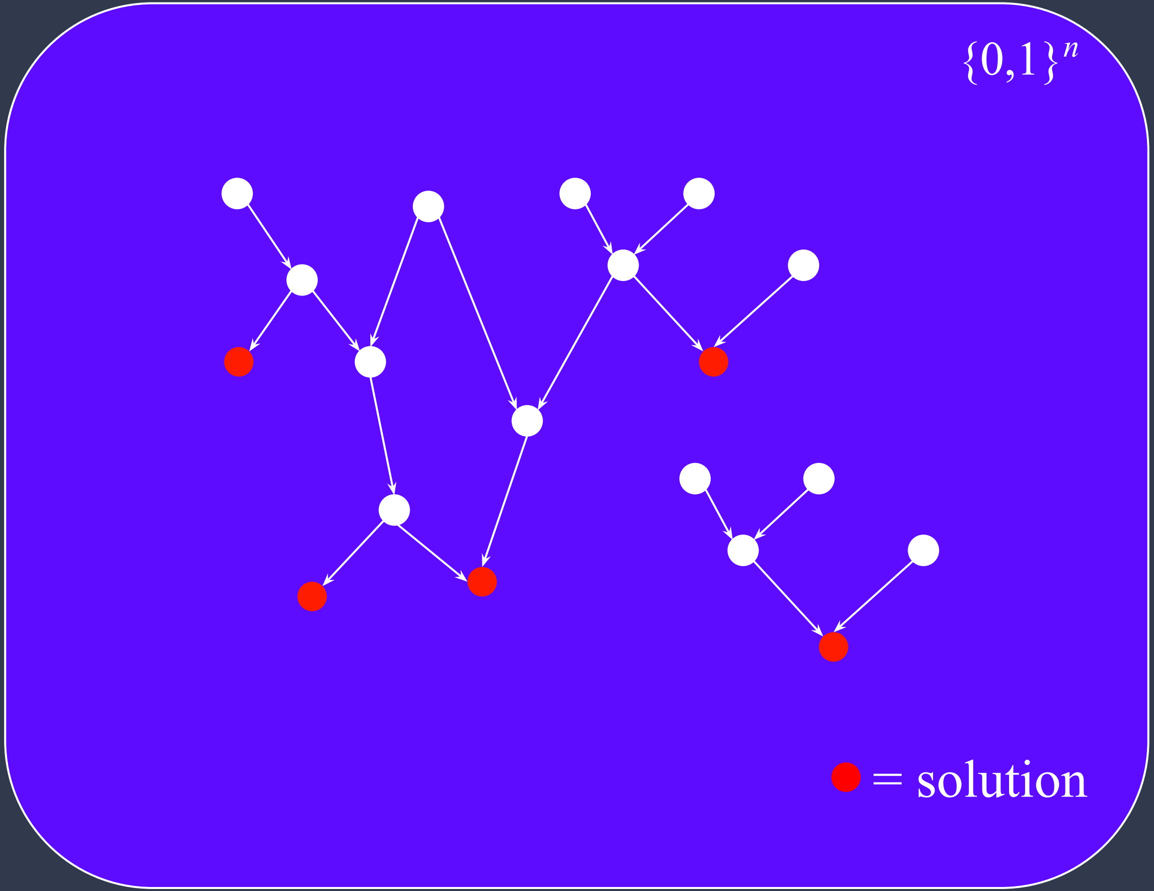
$$v_2 \in C(v_1) \quad \wedge \quad F(v_2) > F_{(}v_1)$$



FIND SINK: Given C, F: Find x s.t. F(x) ≥ F(y), for all $y \in C(x)$.

PLS = { *Function problems in FNP reducible to* FIND SINK }

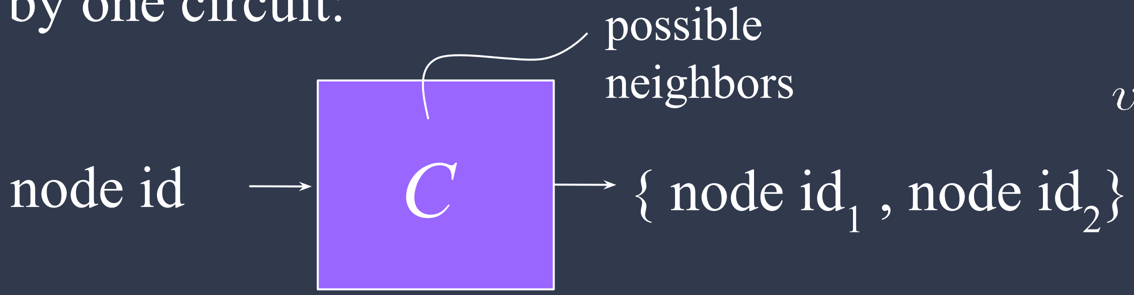
The DAG



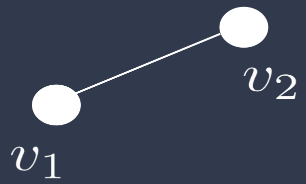
The Class PPA

“If a graph has a node of odd degree, then it must have another.”

Suppose that an exponentially large graph with vertex set $\{0,1\}^n$ is defined by one circuit:



$$v_1 \in C(v_2) \quad \wedge \quad v_2 \in C(v_1)$$



ODD DEGREE NODE: Given C: If 0^n has odd degree, find another node with odd degree. Otherwise say “yes”.

PPA = $\{$ *Search problems in FNP reducible to* ODD DEGREE NODE$\}$

The Class PPP

"If a function maps n elements to n-1 elements, then there is a collision."

Suppose that an exponentially large graph with vertex set $\{0,1\}^n$ is defined by one circuit:

$$\text{node id} \longrightarrow \boxed{C} \longrightarrow \text{node id}$$

COLLISION: Given C: Find x s.t. $C(x) = 0^n$; or find $x \neq y$ s.t. $C(x)=C(y)$.

PPP = { *Search problems in FNP reducible to* COLLISION }

The PPAD Class

Suppose that an exponentially large graph with vertex set $\{0,1\}^n$ is defined by two circuits:

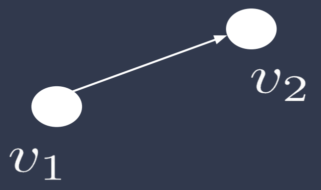
possible previous

node id $\longrightarrow$ $\boxed{P}$ $\longrightarrow$ node id

node id $\longrightarrow$ $\boxed{N}$ $\longrightarrow$ node id

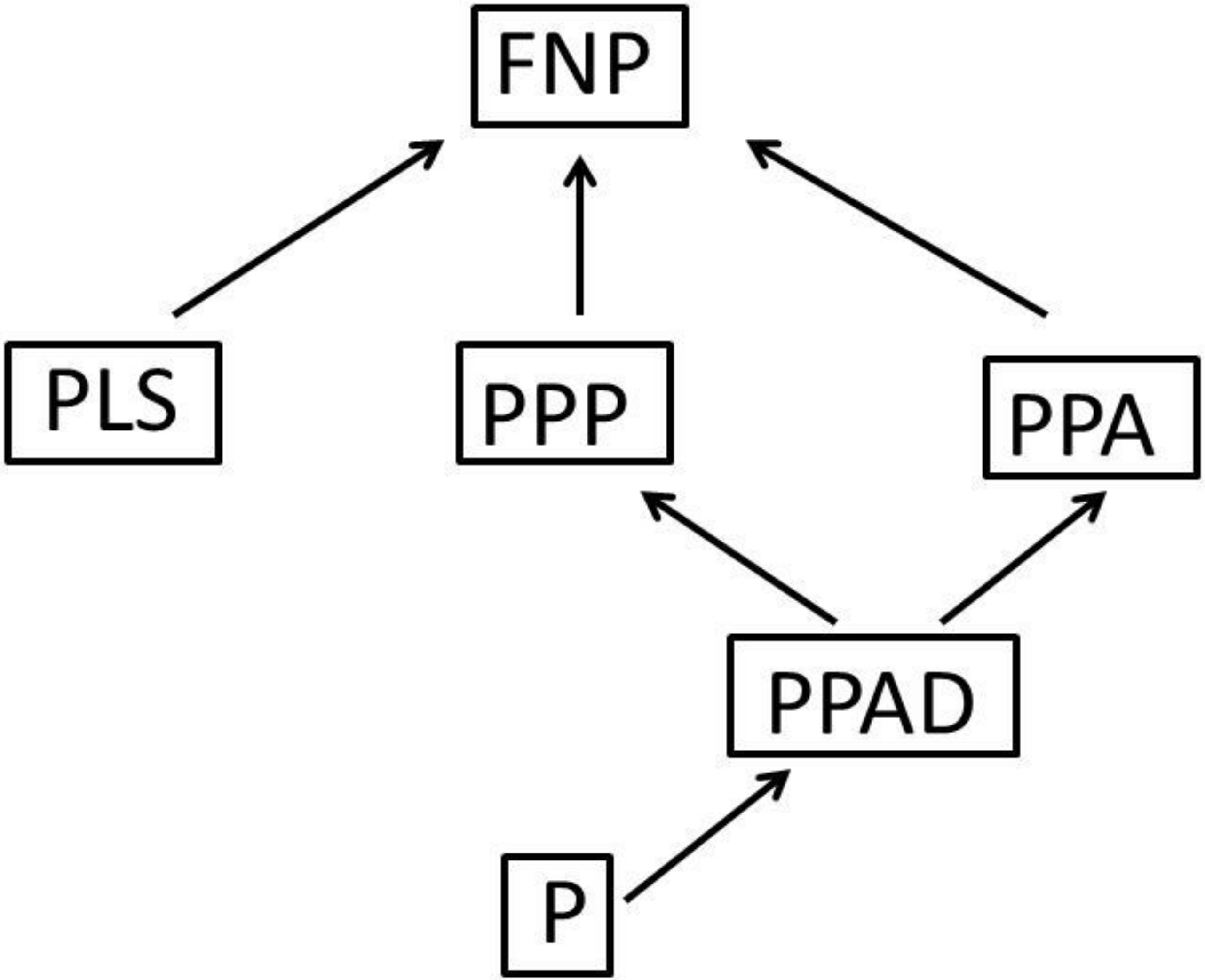
possible next

$$P(v_2) = v_1 \wedge N(v_1) = v_2$$



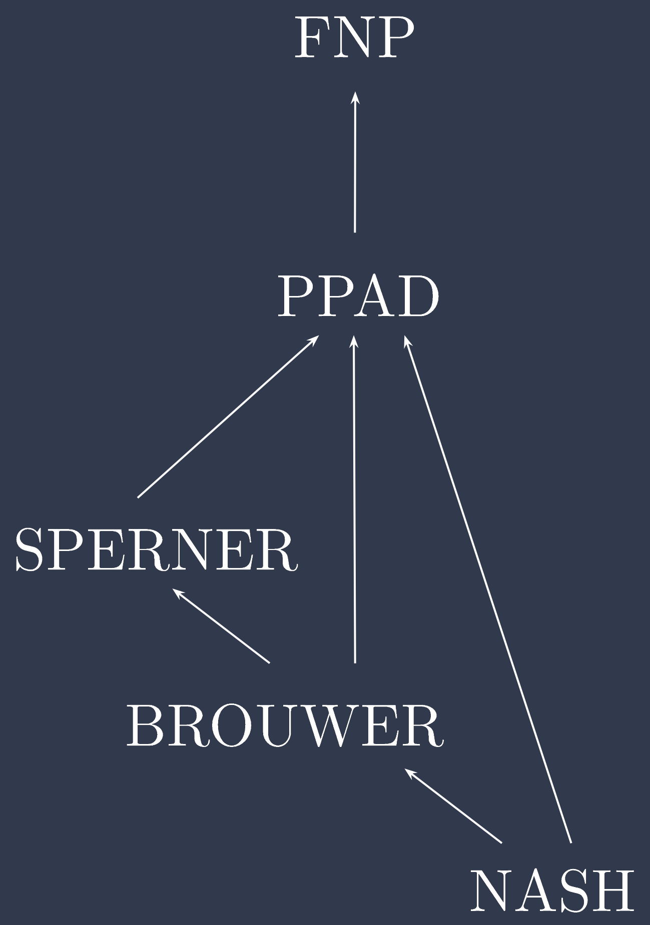
END OF THE LINE: {Given P and N: If 0^n is an unbalanced node, find another unbalanced node. Otherwise say "yes".}

PPAD = *{ Search problems in FNP reducible to END OF THE LINE}*


FNP
PLS
PPP
PPA
PPAD
P

NASH is PPAD-Complete

- Nash to Brouwer Theorem
- Brouwer Theorem to Sperner
- Sperner to END-OF-THE-LINE
- The other direction


FNP
PPAD
SPERNER
BROUWER
NASH

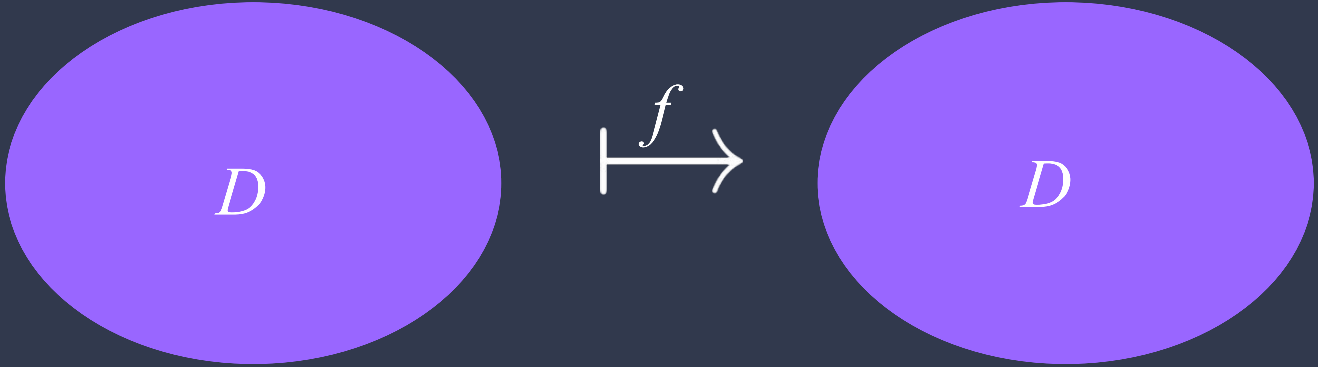
Brouwer's fixed point theorem

Theorem: Let $f : D \longrightarrow D$ be a continuous function from a convex and compact subset D of the Euclidean space to itself.

Then there exists an $x \in D$ s.t. $x = f(x)$.

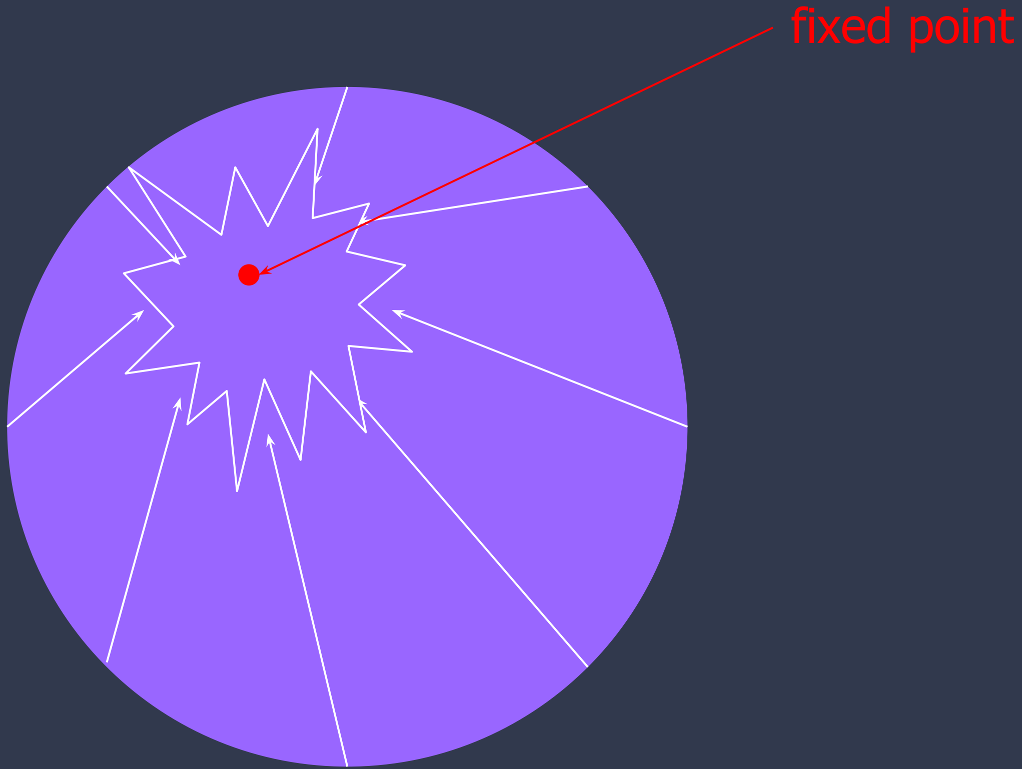
closed and bounded

Below we show a few examples, when D is the 2-dimensional disk.

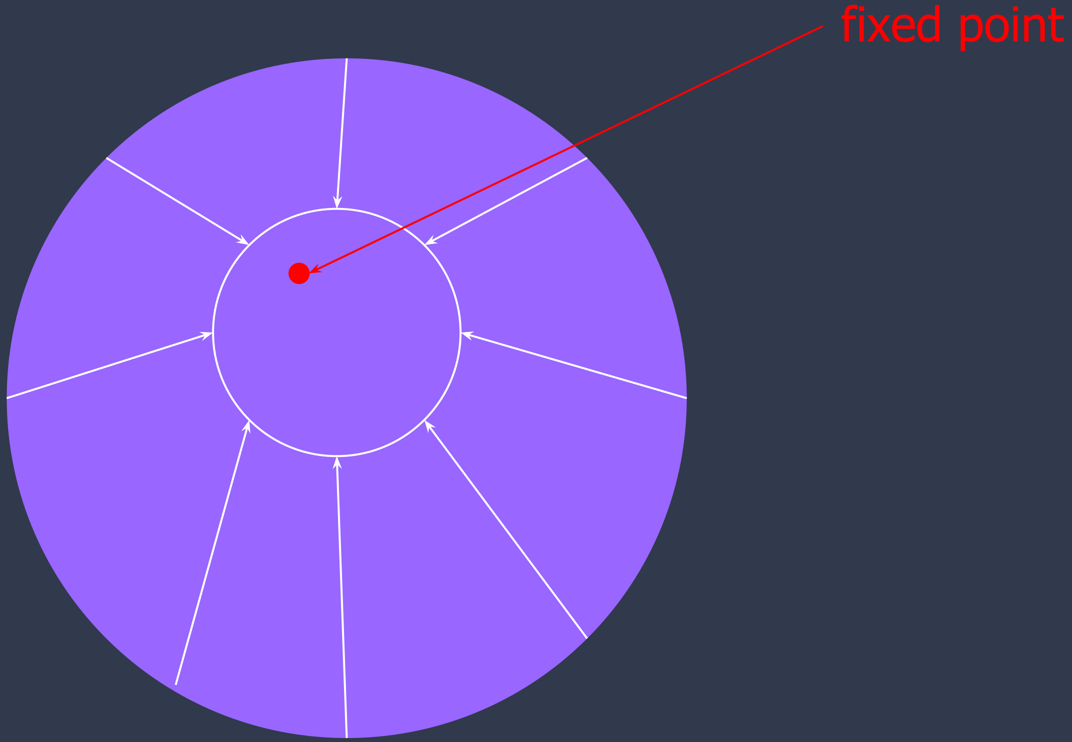


N.B. All conditions in the statement of the theorem are necessary.

Brouwer's fixed point theorem



Brouwer's fixed point theorem



Nash's Function

$$\Delta \ni x \overset{f}{\mapsto} y \in \Delta :$$

$$y_p(s_p) := \frac{x_p(s_p) + \mathrm{Gain}_{p;s_p}(x)}{1 + \sum_{s'_p \in S_p} \mathrm{Gain}_{p;s'_p} x)}$$

where: $\mathrm{Gain}_{p;s_p}(x) = \max\{u_p(s_p; x_{-p}) - u_p(x), 0\}$

Visualizing Nash's Construction

Kick / Dive	Left	Right
Left	1 , -1	-1 , 1
Right	-1 , 1	1, -1

Penalty Shot Game

$\mapsto$

$f: [0,1]^2 \rightarrow [0,1]^2$, continuous
such that
fixed points $\equiv$ Nash eq.

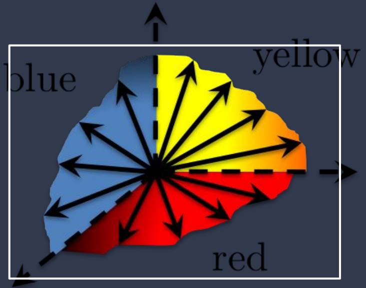
Visualizing Nash's Construction

Kick / Dive	Left	Right
Left	1 , -1	-1 , 1
Right	-1 , 1	1, -1

Penalty Shot Game

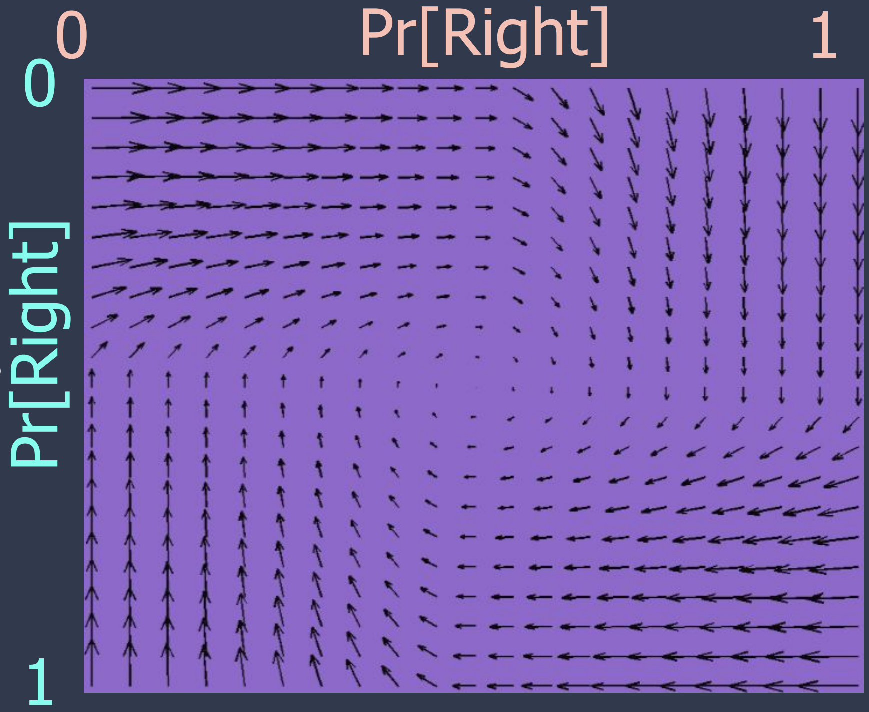
$\mapsto$

Visualizing Nash's Construction



Kick / Dive	Left	Right
Left	1 , -1	-1 , 1
Right	-1 , 1	1, -1

Penalty Shot Game

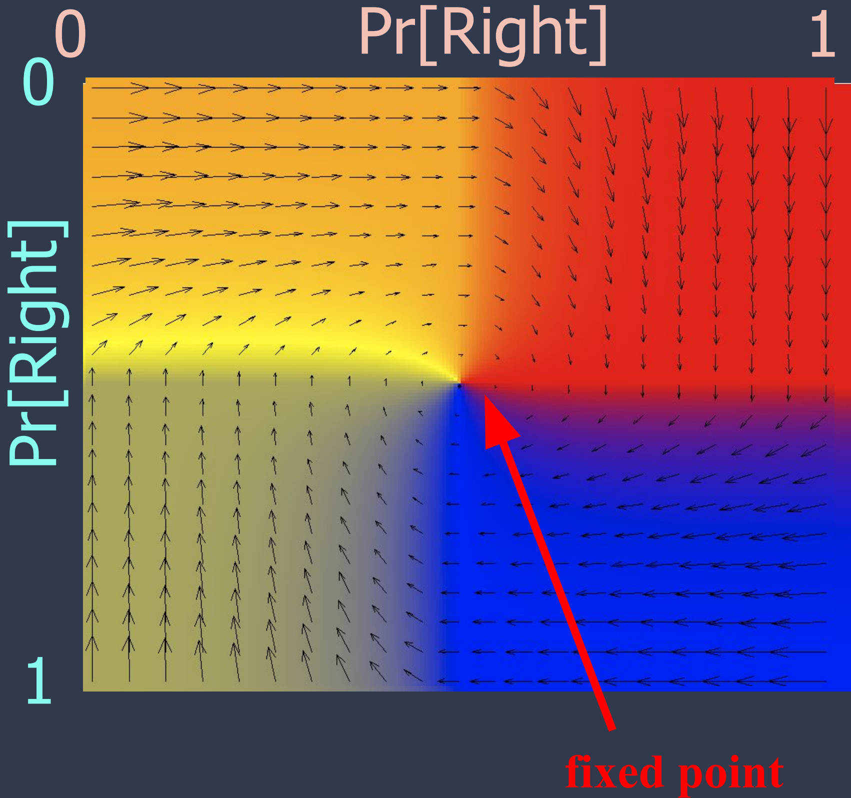


Visualizing Nash's Construction

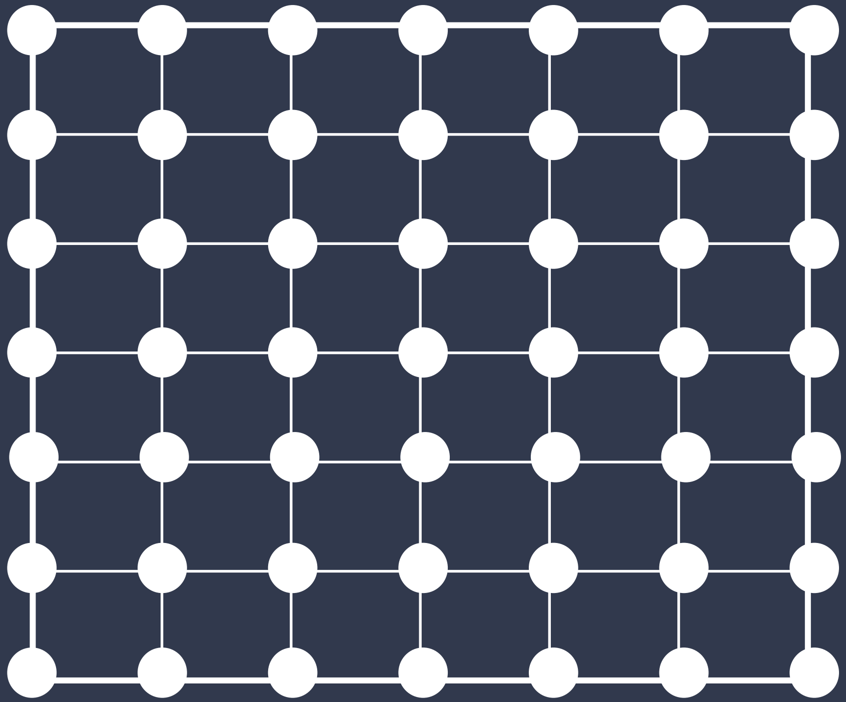
		½	½
	Kick / Dive	Left	Right
½	Left	1 , -1	-1 , 1
½	Right	-1 , 1	1, -1

Penalty Shot Game

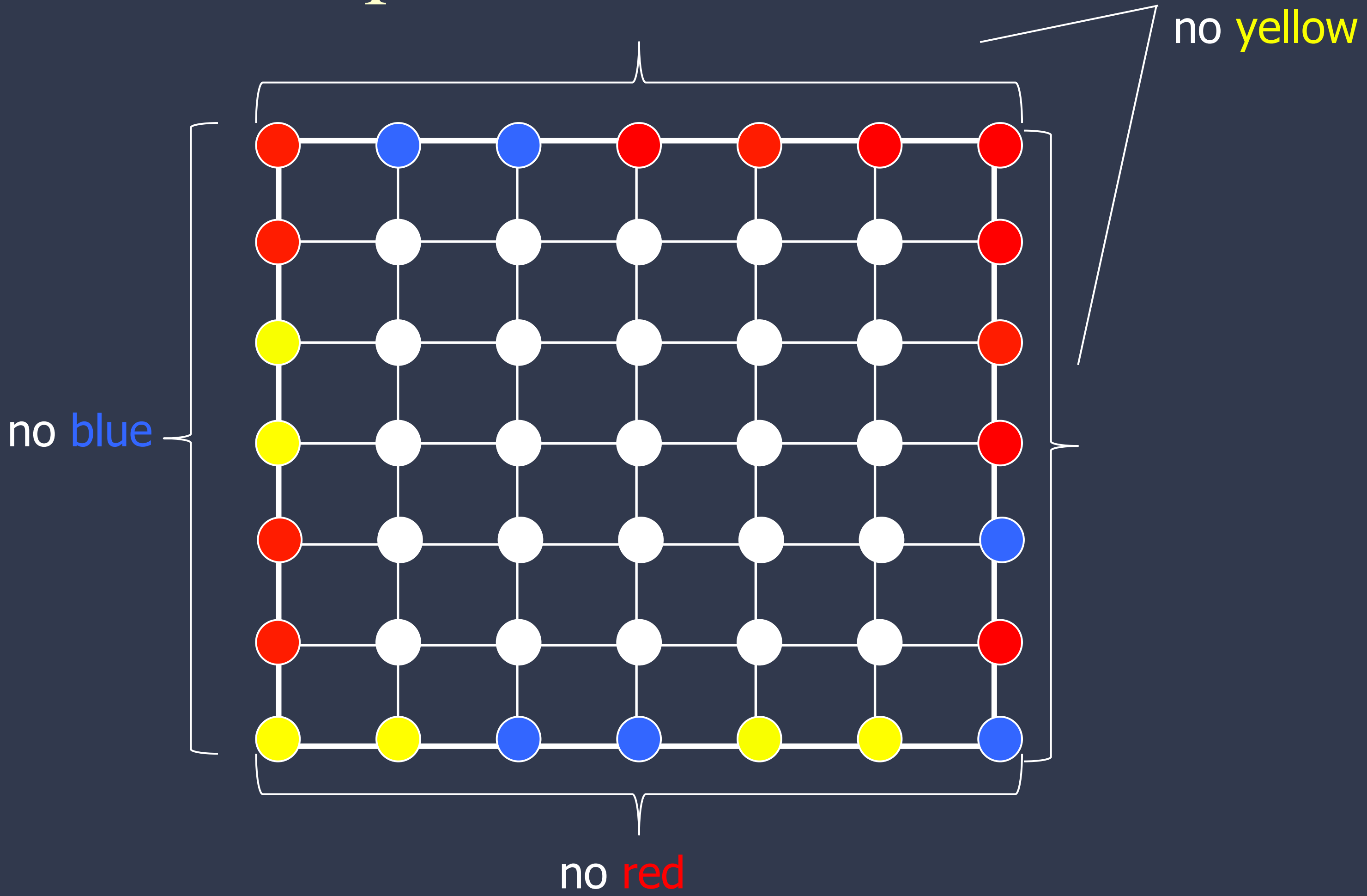
$\mapsto$



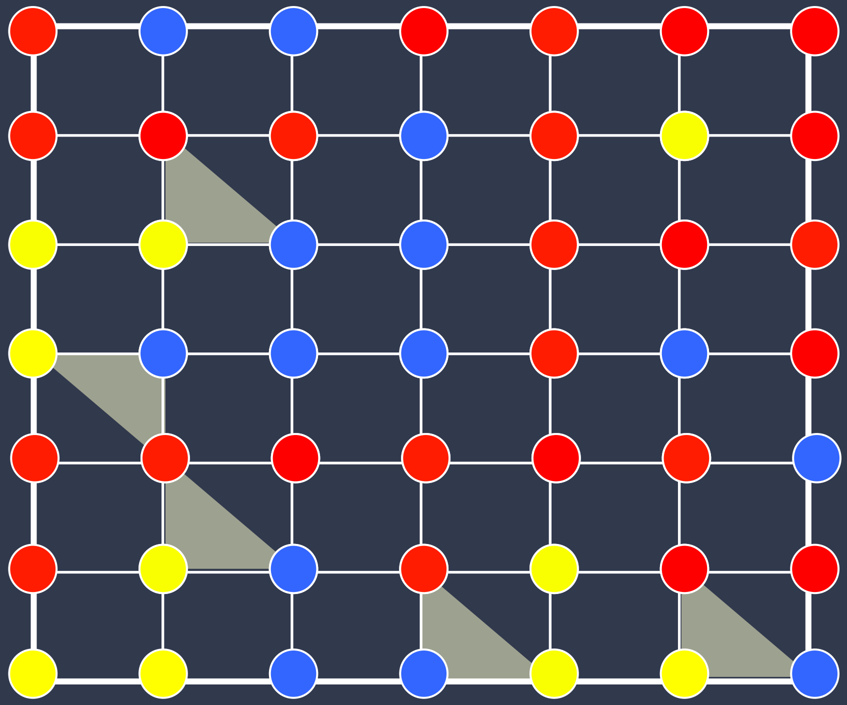
Sperner's Lemma



Sperner's Lemma



Sperner's Lemma



2D-Brouwer on the Square

Suppose f: $[0,1]^2 \rightarrow [0,1]^2$, continuous

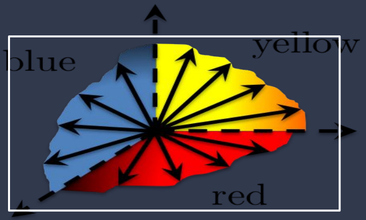
$$\forall \epsilon > 0,\ \exists \delta(\epsilon) > 0, s.t.$$
$$d(z,w) < \delta(\epsilon) \implies d(f(z), f(w)) < \epsilon$$

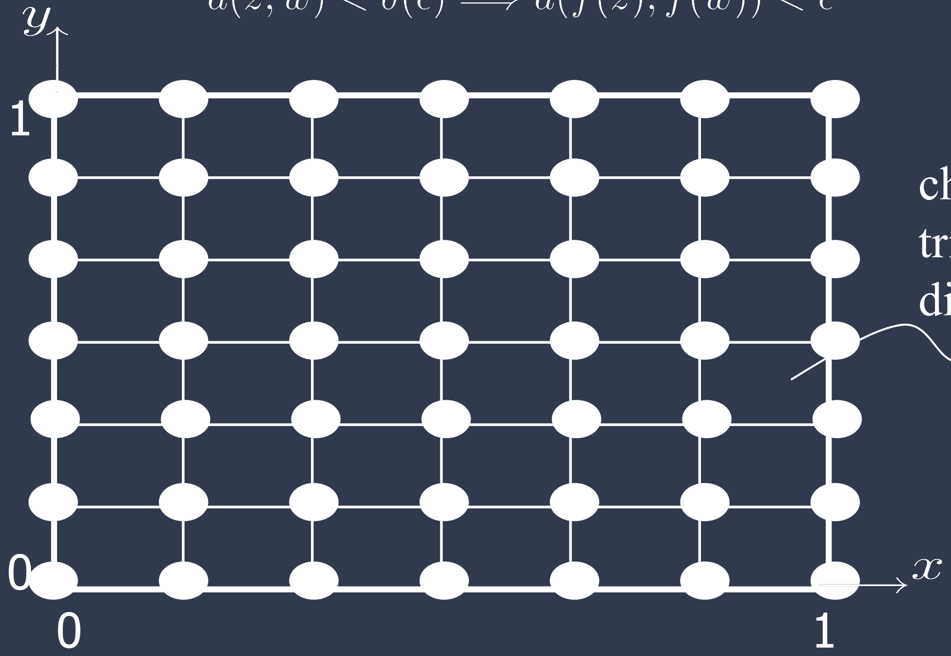
2D-Brouwer on the Square

Suppose f: $[0,1]^2 \rightarrow [0,1]^2$, continuous

$$\forall \epsilon > 0, \ \exists \delta(\epsilon) > 0, s.t.$$
$$d(z,w) < \delta(\epsilon) \implies d(f(z), f(w)) < \epsilon$$

color the nodes of the triangulation according to the direction of
$f(x) - x$





choose some ϵ and triangulate so that the diameter of cells is $\delta < \delta(\epsilon)$

2D-Brouwer on the Square

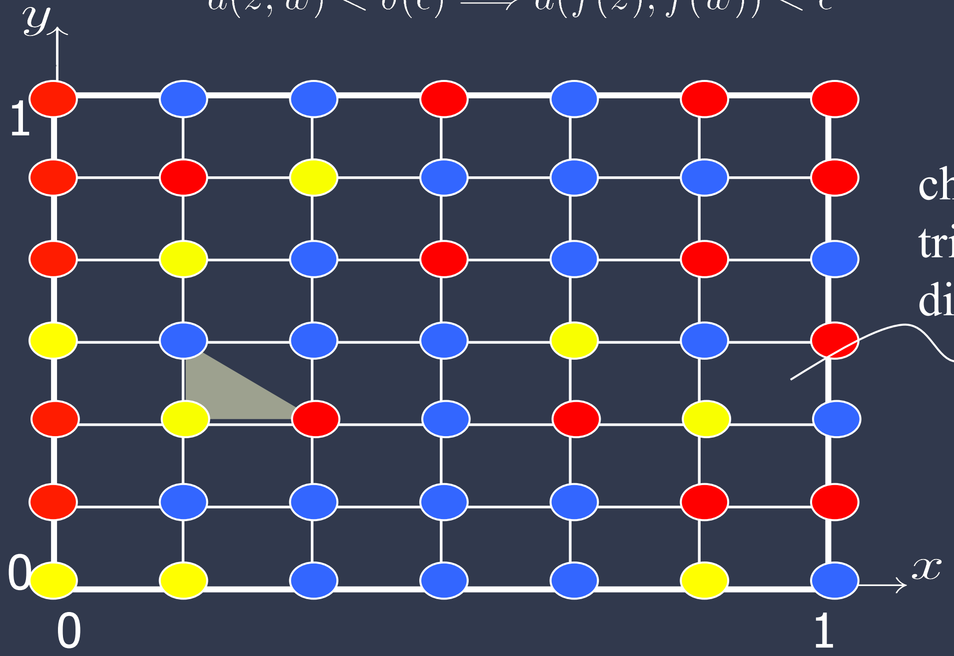
say d is the ℓ_∞ norm

Suppose f: $[0,1]^2 \rightarrow [0,1]^2$, continuous

$$\forall \epsilon > 0, \;\; \exists \delta(\epsilon) > 0, s.t.$$

$$d(z, w) < \delta(\epsilon) \implies d(f(z), f(w)) < \epsilon$$

color the nodes of the triangulation according to the direction of $f(x) - x$



choose some ϵ and triangulate so that the diameter of cells is $\delta < \delta(\epsilon)$

find a trichromatic triangle, guaranteed by Sperner

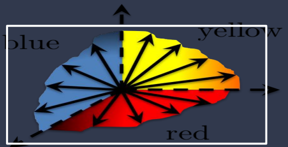
2D-Brouwer on the Square

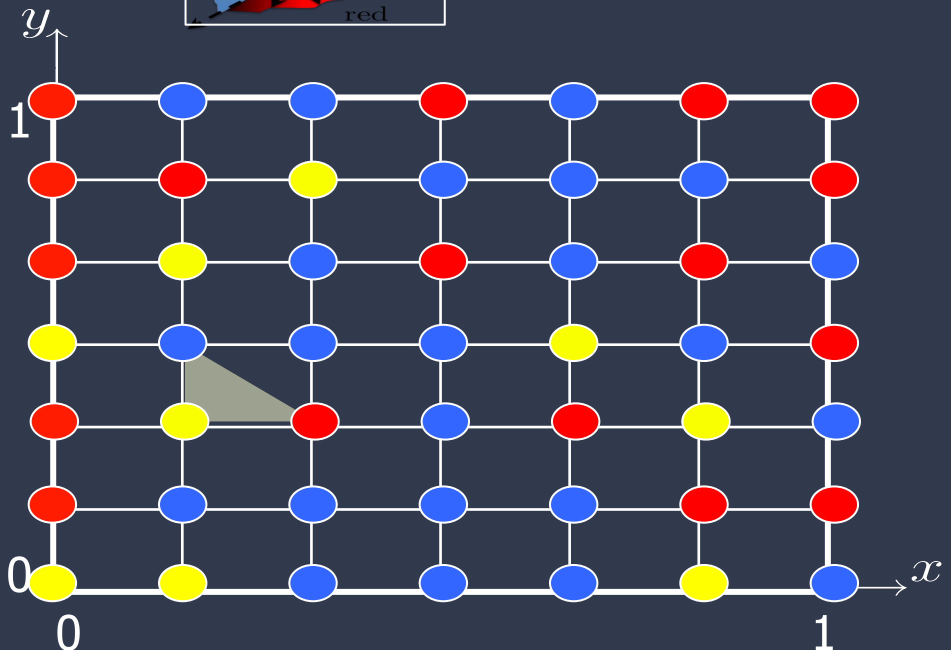
Suppose f: $[0,1]^2 \rightarrow [0,1]^2$, continuous

must be uniformly continuous (by the Heine-Cantor theorem)

$$\forall \epsilon > 0, \ \exists \delta(\epsilon) > 0, s.t.$$

$$d(z,w) < \delta(\epsilon) \implies d(f(z), f(w)) < \epsilon$$





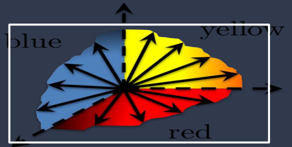
Claim: If z^{Y} is the yellow corner of a trichromatic triangle, then

$$|f(z^{\mathrm{Y}}) - z^{\mathrm{Y}}|_\infty < \epsilon + \delta.$$

Proof of Claim

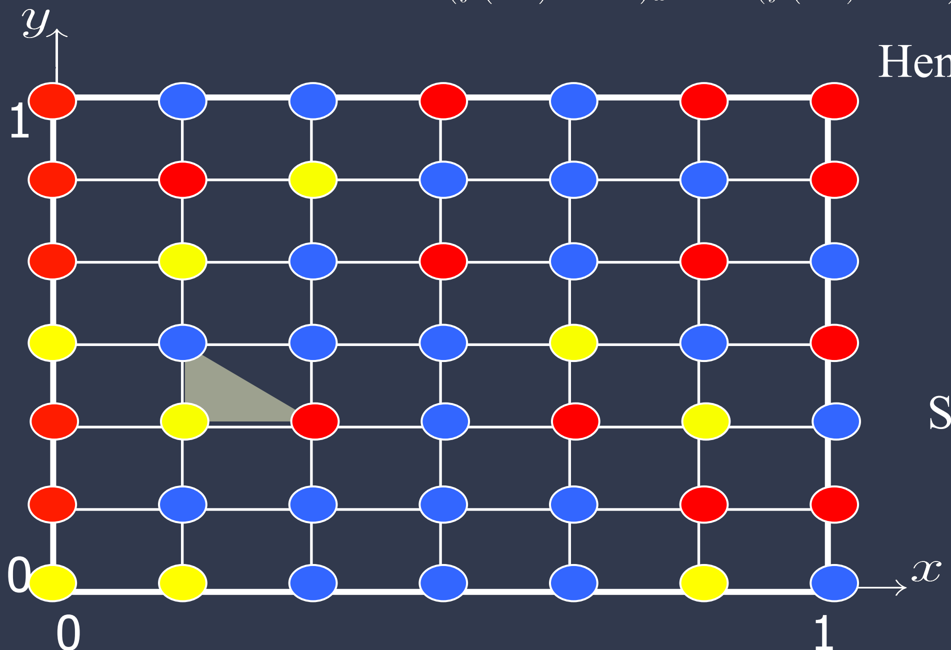
Claim: If z^{Y} is the yellow corner of a trichromatic triangle, then $|f(z^{\mathrm{Y}}) - z^{\mathrm{Y}}|_\infty < \epsilon + \delta.$

Proof: Let $z^{\mathrm{Y}}, z^{\mathrm{R}}, z^{\mathrm{B}}$ be the yellow/red/blue corners of a trichromatic triangle.



By the definition of the coloring, observe that the product of

$$(f(z^Y) - z^Y)_x \text{ and } (f(z^B) - z^B)_x \text{ is } \leq 0.$$



Hence:

$$\begin{aligned}
&|(f(z^Y) - z^Y)_x| \\
&\quad \leq |(f(z^Y) - z^Y)_x - (f(z^B) - z^B)_x| \\
&\quad \leq |(f(z^Y) - f(z^B))_x| + |(z^Y - z^B)_x| \\
&\quad \leq d(f(z^Y), f(z^B)) + d(z^Y, z^B) \\
&\quad \leq \epsilon + \delta.
\end{aligned}$$

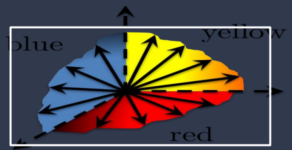
Similarly, we can show:

$$|(f(z^Y) - z^Y)_y| \leq \epsilon + \delta.$$

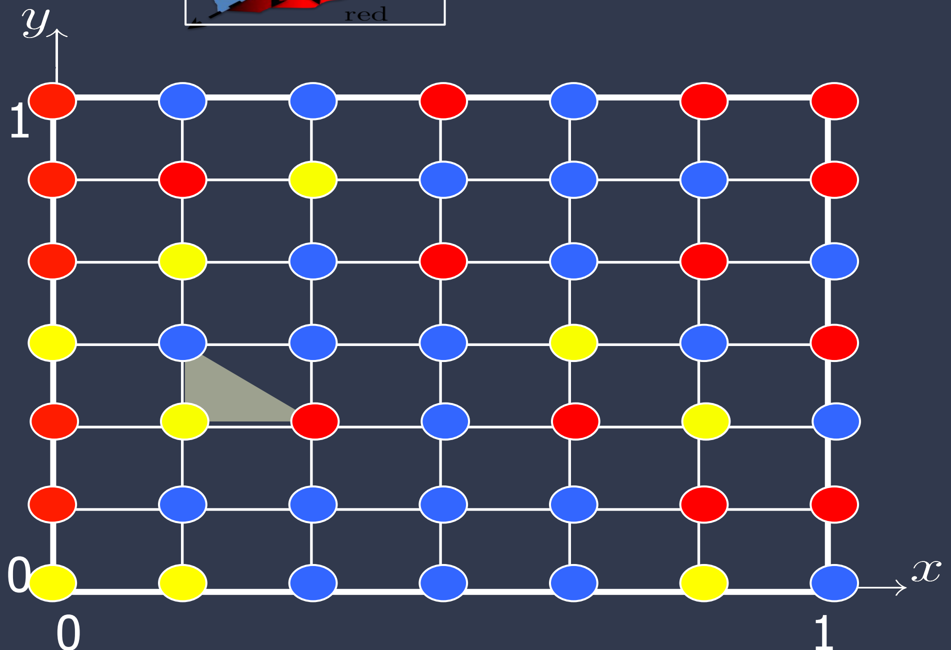
$\square$

2D-Brouwer on the Square

Suppose f: $[0,1]^2 \rightarrow [0,1]^2$, continuous



$$\forall \epsilon > 0, \ \exists \delta(\epsilon) > 0, s.t.$$
$$d(z, w) < \delta(\epsilon) \implies d(f(z), f(w)) < \epsilon$$



Claim: If z^{Y} is the yellow corner of a trichromatic triangle, then

$$|f(z^{\mathrm{Y}}) - z^{\mathrm{Y}}|_\infty < \epsilon + \delta.$$

Choosing $\delta = \min(\delta(\epsilon), \epsilon)$

$$|f(z^Y) - z^Y|_\infty < 2\epsilon.$$

Conclusion

- NASH is hard then, what to do?
- Alternatives, tractability of alternatives
- Open problems

NASH is important

- Universality
- But is NASH credible now?

If finding this Nash equilibrium is intractable (i.e. there exist some specialized instances when one cannot find it in any reasonable amount of time), then it loses universality, and therefore loses credibility as a predictor of behavior.

Approximate Nash equilibria

$$\sum_{s \in S_{-p}} u_p(js)x_s > \varepsilon + \sum_{s \in S_{-p}} u_p(j' s)x_s \Longrightarrow x^p_{j'} = 0.$$

The players have low incentive to change their strategy!

Epsilon-equilibria is PPAD-Complete on games with any number of players, yet more! tractable on games with constant number of player.

Remarks

- There is a FPTAS scheme for computing NASH!

(Fully Polynomial Time Approximation Scheme)

- There is sub-exponential algorithm that computes epsilon-equilibria on games with constant number of players.

$$O\left(n^{\log n/\varepsilon^2}\right)$$

Open Problem

- Whether or not there exists a PTAS for approximate nash equilibria?

- Better than $O\left(n^{\log n/\varepsilon^2}\right)$ algorithm for constant number of players ep-NASH?

Thank you

Questions?

References

X. Chen and X. Deng. Settling the Complexity of 2-Player Nash Equilibrium. In Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science. IEEE Computer Society, 2006.

X. Chen, X. Deng, and S.H. Teng. Computing Nash Equilibria: Approximation and Smoothed Complexity. In Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science. IEEE Computer Society, 2006.

V. Conitzer and T. Sandholm. Complexity Results about Nash Equilibria. In Proceedings of the 18th International Joint Conference on Artificial Intelligence, pages 765–771, 2003.

C. Daskalakis, P.W. Goldberg, and C.H. Papadimitriou. The Complexity of Computing a Nash Equilibrium. Communications of the ACM, 52(2):89–97, 2009.

References

V. Bubelis. “On Equilibria in Finite Games,” International Journal on Game Theory, 8(2):65–79 (1979).

X. Chen and X. Deng. “3-NASH is PPAD-Complete,” Electronic Colloquium in Computational Complexity, TR05-134, 2005.

X. Chen, X. Deng and S. Teng. “Computing Nash Equilibria: Approximation and Smoothed Complexity,” Proceedings of FOCS, 2006.

V. Conitzer and T. Sandholm. “Complexity Results about Nash Equilibria,” Proceedings of IJCAI, 2003.

C. Daskalakis and C. H. Papadimitriou. “Three-Player Games Are Hard,” Electronic Colloquium in Computational Complexity, TR05-139, 2005.

K. Etessami and M. Yannakakis. “On the Complexity of Nash Equilibria and Other Fixed Points (Extended Abstract),” Proceedings of FOCS, 113-123, 2007

References

H. E. Scarf. “The Approximation of Fixed Points of a Continuous Mapping,” SIAM Journal of Applied Mathematics, 15: 1328–1343, 1967.

Y. Shoham. “Computer Science and Game Theory,” Communications of the ACM, 51(8): 75–79.

Constantinos Daskalakis, “Topics in Algorithmic Game Theory” Lecture Note 6, Slides session 7-12, MIT

XiChena, XiaotieDengb: On the complexity of 2D discrete fixed point problem

Atanassov, K. T. (1996), "On Sperner's lemma", Studia Scientiarum Mathematicarum Hungarica

Paul W. Goldberg, A Survey of PPAD-Completeness for Computing Nash Equilibria

References

Kellogg, R. Bruce; Li, Tien-Yien; Yorke, James A. (1976). "A constructive proof of the Brouwer fixed point theorem and computational results". SIAM Journal on Numerical Analysis.

M. Bellare and S. Goldwasser. The complexity of decision versus search. SIAM Journal on Computing, Vol. 23, No. 1, February 1994.

Goldberg and Papadimitriou. Towards a Unified Complexity Theory of Total Functions, 2018

Megiddo and Papadimitriou. A Note on Total Functions, Existence Theorems and Computational Complexity. Theoretical Computer Science 1989.

ohnson, David S; Papadimitriou, Christos H; Yannakakis, Mihalis (1988). "How easy is local search?". Journal of Computer and System Sciences. 37

References

Michelangelo Grigni (1995). "A Sperner Lemma Complete for PPA". Information Processing Letters.

Fortnow, Lance (2005). "What is PPAD?". Retrieved 2007-01-29.

Amin Saberi, “Computation of Equilibria” Lecture Notes, Lecture 1 Stanford