Mathematical Programming Games

Andrea Lodi

ECCO XXXV - June 9, 2022

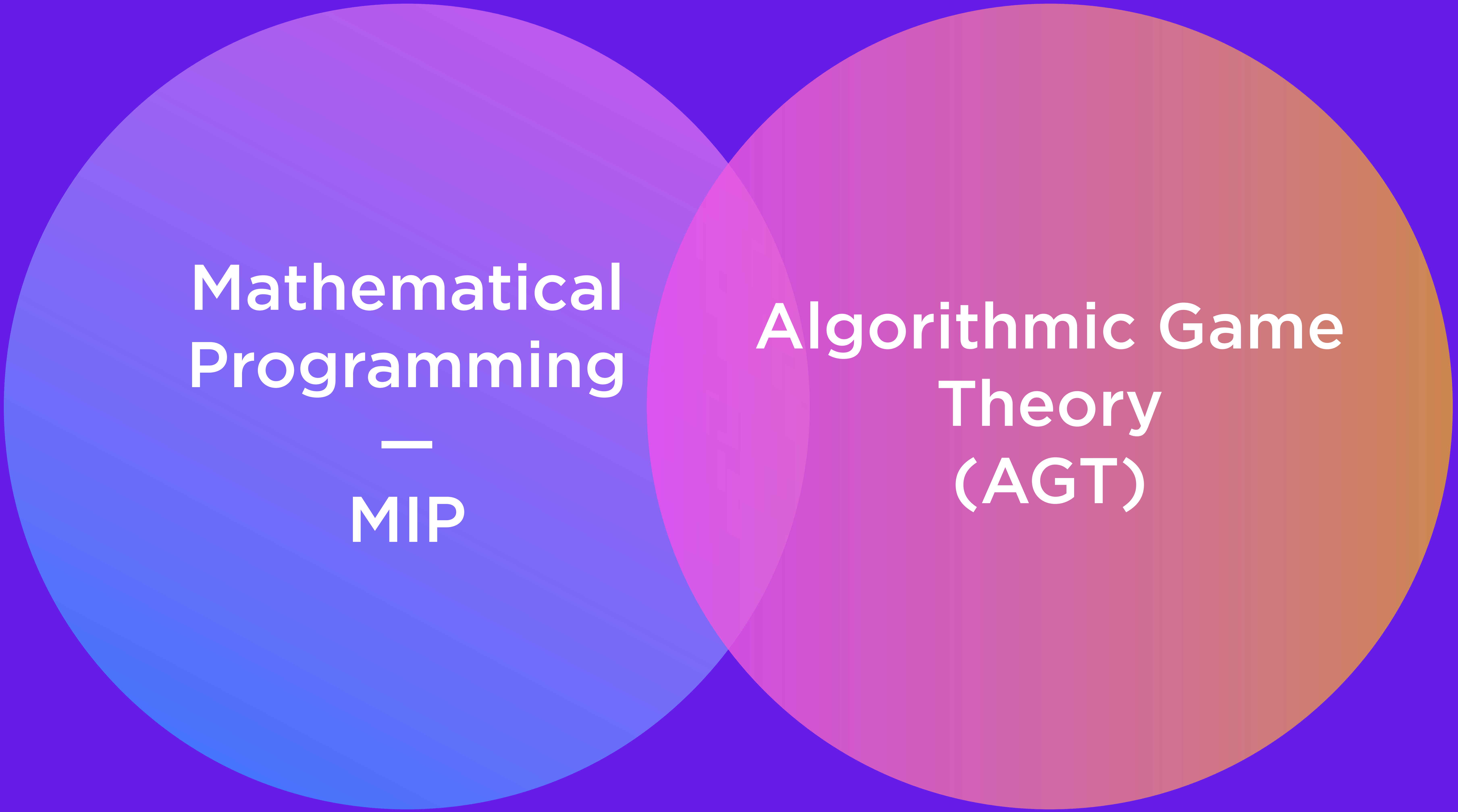
CANADA EXCELLENCE RESEARCH CHAIR
DATA SCIENCE FOR REAL-TIME DECISION-MAKING

EWG ECCO COMBINATORIAL OPTIMIZATION

Margarida Carvalho

Gabriele Dragotto

Sriram Sankaranarayanan


Mathematical Programming
—
MIP
Algorithmic Game Theory
(AGT)

Mathematical Programming Games (MPGs)

A Brief Overview of This Talk

What are **Mathematical Programming Games**

Why do we need them, some **applications**, and **core research questions**

How do we ***solve*** them in practice

What are MPGs?

What

Why

What are MPGs?

An ***MPG*** is a (static) **game** among n players where each **rational** player $i = 1,2,\ldots,n$ solves the optimization problem

$$\max_{x^i} \{ f^i(x^i, x^{-i}) : x^i \in \mathcal{X}^i \}$$

The payoff function for i

$$f^i(x^i, x^{-i}) : \prod_{j=1}^{n} \mathcal{X}^j \to \mathbb{R}$$

is **parametrized in** x^{-i}

The set of actions for i

$$\mathcal{X}^i$$

$$\max_{x^i} \{ f^i(x^i, x^{-i}) : x^i \in \mathscr{X}^i \}$$

The payoff function for i

$$f^i(x^i, x^{-i}) : \prod_{j=1}^{n} \mathscr{X}^j \to \mathbb{R}$$

is **parametrized in** x^{-i}

The set of actions for i

$$\mathscr{X}^i$$

The choices of i's opponents affect its payoff

However, they do not affect i's actions

$$\max_{x^i} \{ f^i(x^i, x^{-i}) : x^i \in \mathscr{X}^i \}$$

Action Representation

Each player's actions are represented with **an arbitrary** set $\mathscr{X}^i$

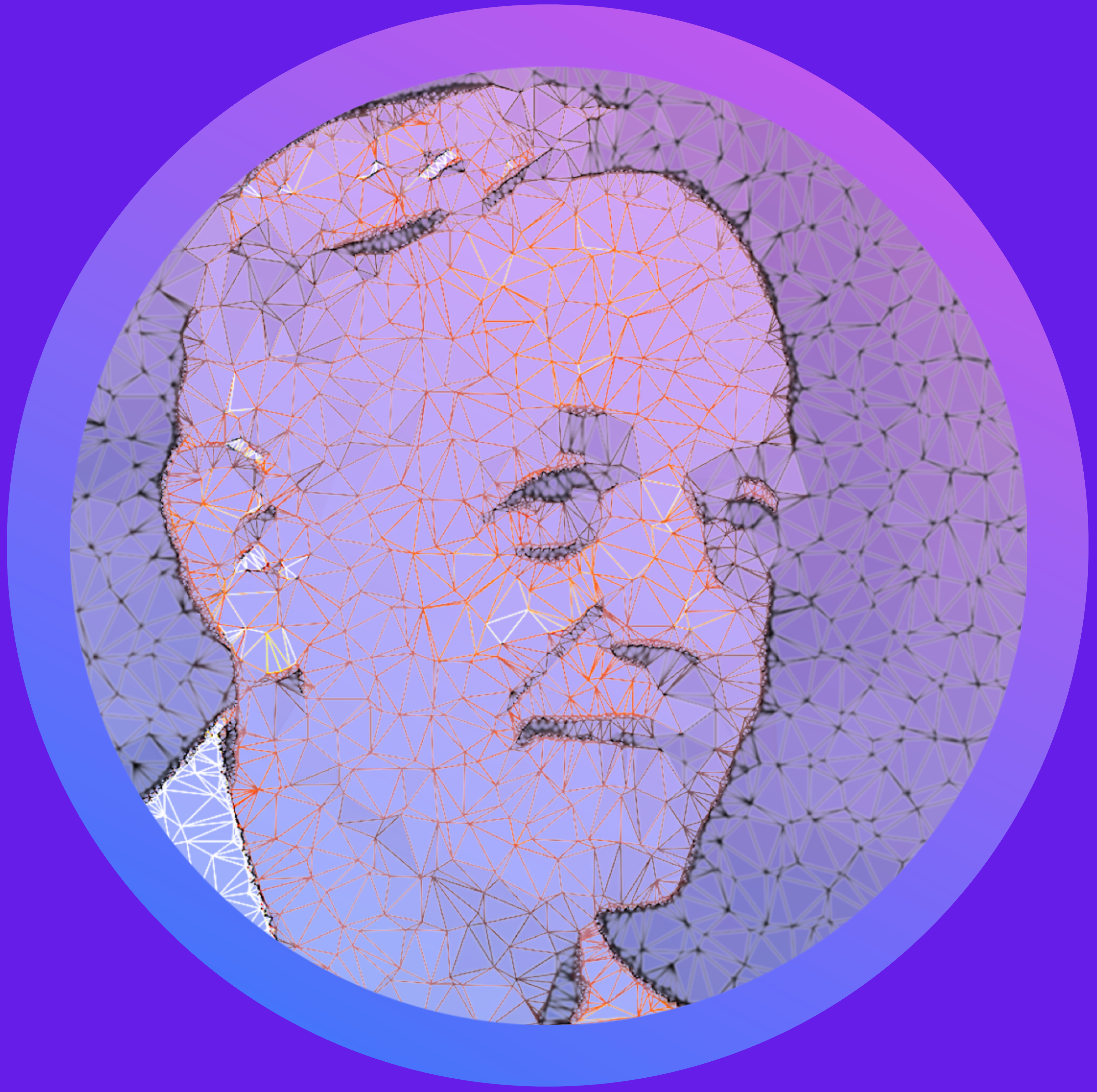
Modeling Requirements

In many applications, $\mathscr{X}^i$ may include a **complex set of operational requirements**

Language and Objectives

MPGs provide a **unified framework** to represent games from both AGT and Optimization

Equilibria as Solutions



A profile $\bar{x} = (\bar{x}^1, \ldots, \bar{x}^n)$ *— with* $\bar{x}^i \in \mathcal{X}^i$ *for any* i *— is a Pure Nash Equilibrium (***PNE***) if*

$$f^i(\bar{x}^i, \bar{x}^{-i}) \geq f^i(\hat{x}^i, \bar{x}^{-i}) \quad \forall \hat{x}^i \in \mathcal{X}^i$$

Does at least **one exist**? **How hard** is it to **compute** one?

How do we compute an NE, if any? And how do we **select one** when multiple equilibria exist?

How **efficient** is this NE?

A Few Examples

Integer Programming Games, or games among parametrized Integer Programs

Bilevel Programming and simultaneous games, specifically for energy

🧙🏾‍♂️ 🧚🏼‍♂️

Open 2 Convenience Stores

$$\max_{x^1} \quad 6x_1^1 + x_2^1$$

$$\text{s.t.} \quad 3x_1^1 + 2x_2^1 \leq 4$$

$$x^1 \in \{0,1\}^2$$

Their products **interact**!

$$\max_{x^1} \quad 6x_1^1 + x_2^1 - 4x_1^1x_1^2 + 6x_2^1x_2^2$$

$$\text{s.t.} \quad 3x_1^1 + 2x_2^1 \leq 4$$

$$x^1 \in \{0,1\}^2$$

$$\max_{x^2} \quad 4x_1^2 + 2x_2^2 - x_1^2x_1^1 - x_2^2x_2^1$$

$$\text{s.t.} \quad 2x_1^2 + 3x_2^2 \leq 4$$

$$x^2 \in \{0,1\}^2$$

Knapsack Games (Carvalho et al., 2022)

☀️⚡

Energy

Carvalho, Dragotto, Lodi, Feijoo, Sankaranarayanan (2020)

SolarCorp Inc.

Simultaneous
Game

Hydro Inc.

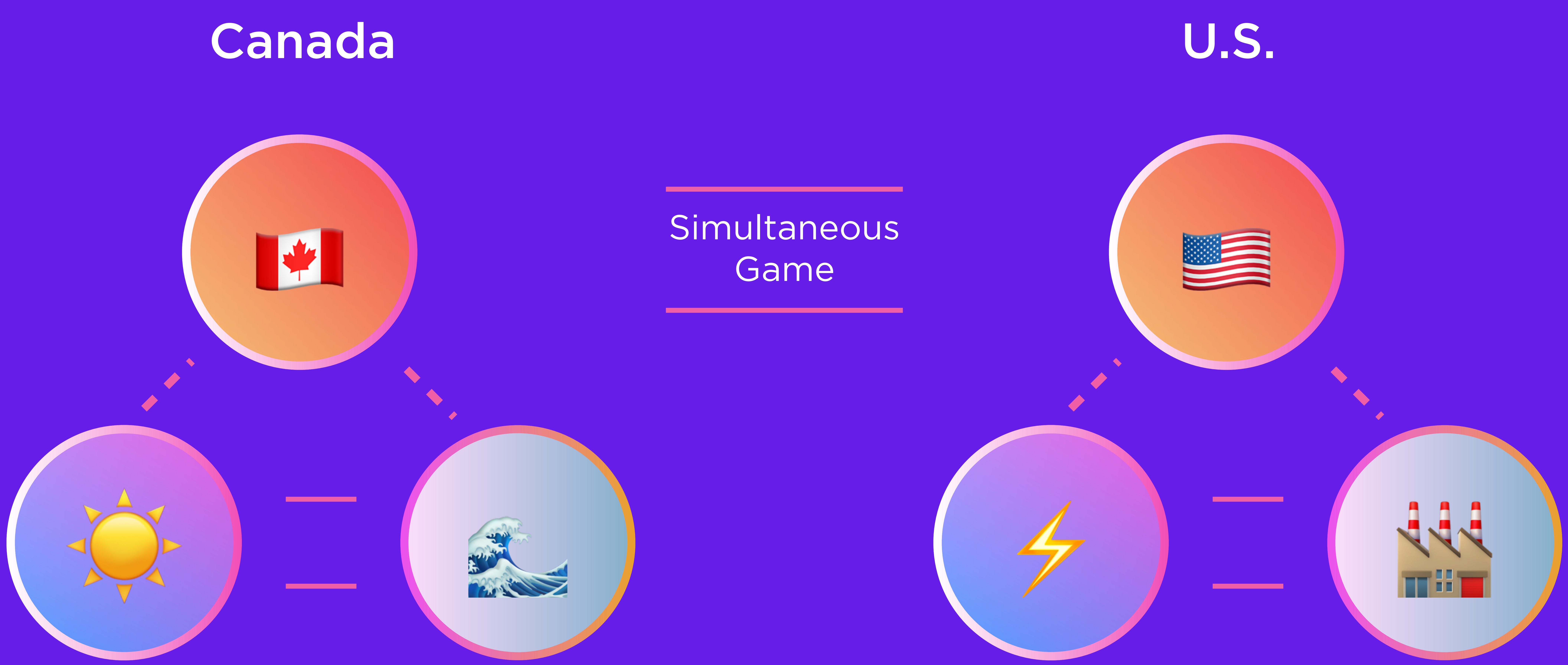
Canada taxes and regulates the production

Simultaneous Game

SolarCorp Inc.

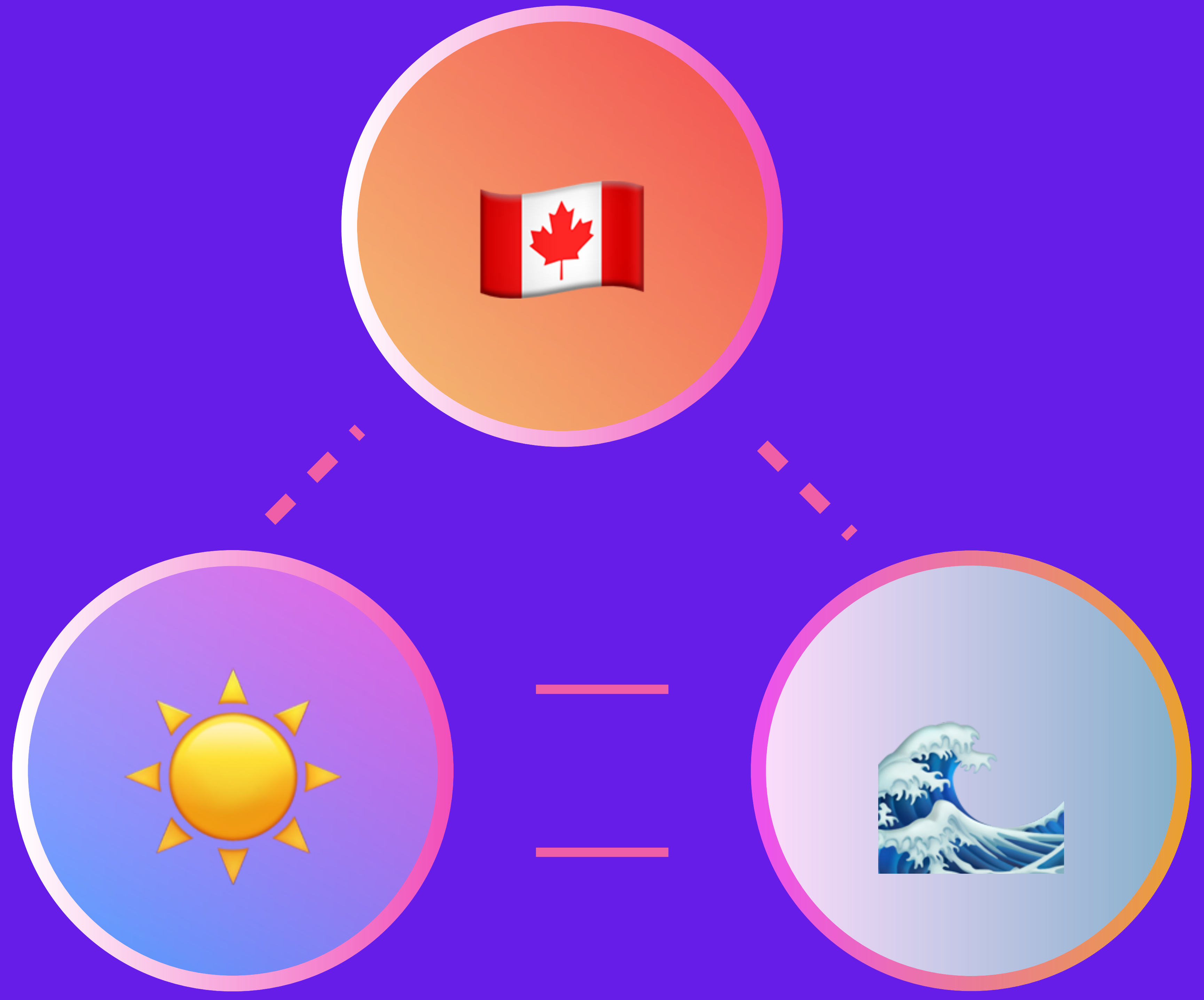
Hydro Inc.


Sequential
“Stackelberg” Game
Simultaneous
Game
SolarCorp Inc.
Hydro Inc.



This is a simultaneous game among bilevel (i.e., sequential) programs (**NASP**)

Canada



$$\max_{x^i} \{ (c^i)^\top x^i + (x^{-i})^\top C^i x^i : x^i \in \mathscr{F}^i \}$$

Each $\mathscr{X}^i$ includes the optimality conditions of each “follower” (i.e., producer)

$$\mathcal{F}^i = \left\{ \begin{array}{l} A^i x^i \leq b^i \\ z^i = M^i x^i + q^i \\ x^i \geq 0, z^i \geq 0 \end{array} \right\} \bigcap_{j \in \mathcal{C}^i} (\{z^i_j = 0\} \cup \{x^i_j = 0\}).$$

Modeling	Can MPGs model real-world problems?
Existence	When does at least an equilibrium exist?
Algorithms	How do we compute and select equilibria?
Efficiency	How do different equilibria (solutions) in MPGs differ?
Insights	Do equilibria promote socially-beneficial outcomes and provide insights?

How?

~~What~~ ~~are **Mathematical Programming Games**~~

~~Why~~ ~~do we need them, some **applications**, and **core research questions**~~

How do we use and ***solve*** them in practice

How?

How do we use and solve them in practice

Cut-And-Play

Computing Nash equilibria in some non-convex games

The Cut-and-Play Algorithm

Non-Convexities

How to compute equilibria in MPGs where players solve **non-convex optimization problems**?

Specifically, when $\mathscr{X}^i$ **is non convex?**

Integer Variables: indivisible quantities and logical conditions

Bilevel Constraints: hierarchical decision-making

Non-linear non-convex constraints: physical phenomena

The Problem

RBGs

We consider *Reciprocally-Bilinear Games* (***RBGs***), namely MPGs where each player solves

$$\max_{x^i}\{f^i(x^i, x^{-i}) = (c^i)^\top x^i + (x^{-i})^\top C^i x^i : x^i \in \mathcal{X}^i\}$$

- There is **common knowledge of rationality**, thus each player is **rational** and there is **complete information**,
- The game is **polyhedrally-representable** if cl conv$(\mathcal{X}^i)$ is a polyhedron for any i + blackbox to optimize a **linear function over** $\mathcal{X}^i$

Contributions

Algorithms

Cutting plane algorithm: computes **(Mixed) Nash equilibria** (MNEs)

The first algorithm to work with iteratively refined **outer approximations** of player's feasible sets (convex hulls) + **general non-convex games (polyhedrally representable)**

Integrates **integer programming machinery**

Practical

Extensive testing on Knapsack Games and games among bilevel leaders (NASPs)

How do we compute an NE, if any? And how do we **select one** when multiple equilibria exist?

Lemke-Howson Generalizations

Lemke and Howson, 1964;
Rosenmüller, 1971;
Wilson, 1971;
Avis et al., 2010;
Audet et al., 2006.

Homotopy-based

Scarf, 1967.

Equilibrium Programming

Facchinei and Pang, 2003;
Sagratella, 2016;
Pang and Scutari, 2011.

Support Enumeration

Sandholm et al., 2005;
Porter et al., 2008.

MIP

Sandholm et al., 2005;
Cronert and Minner, 2021;
Carvalho et al., 2022.

Equilibrium Programming

Facchinei and Pang, 2003;
Sagratella, 2016;
Pang and Scutari, 2011.

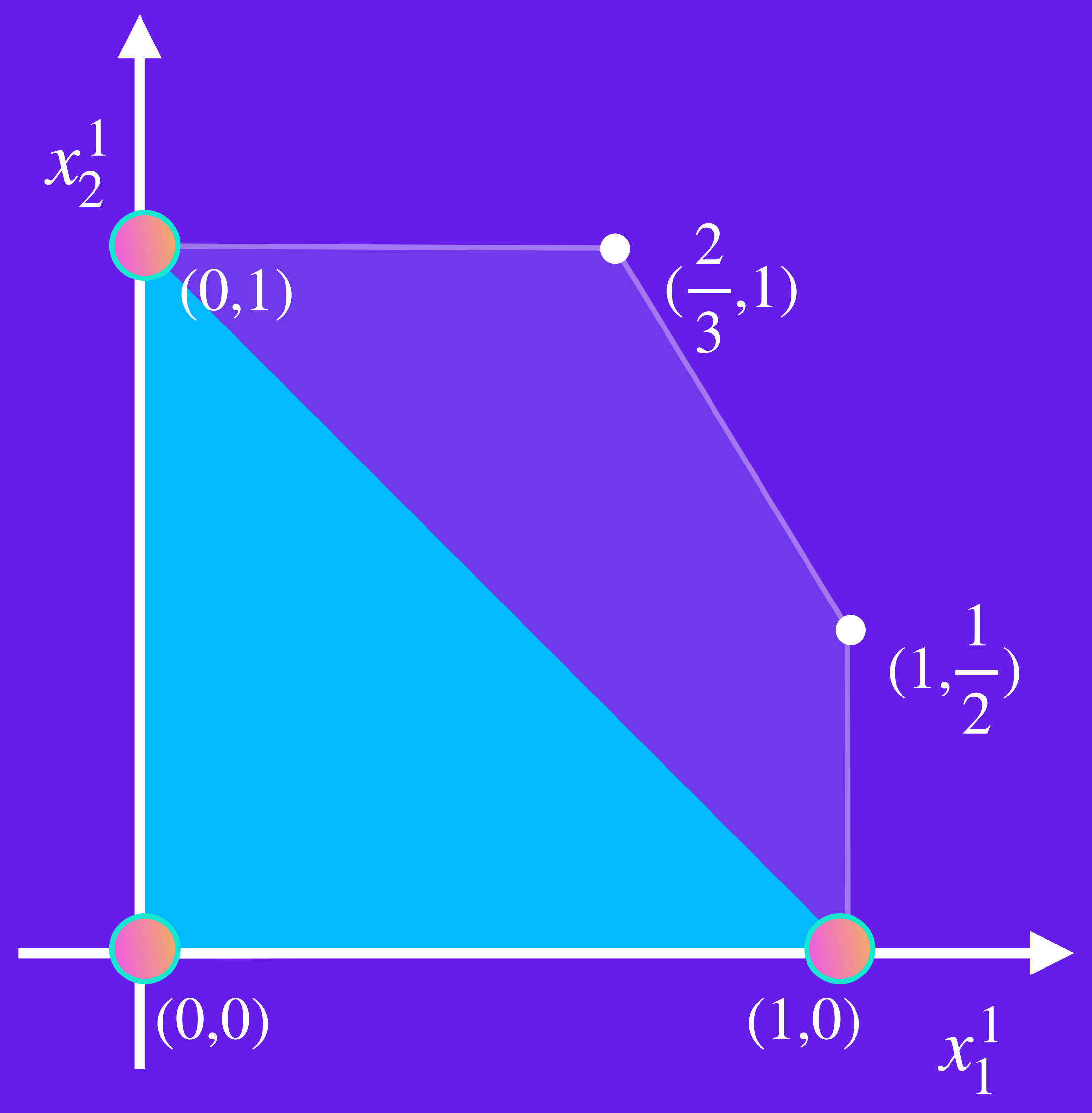
MIP

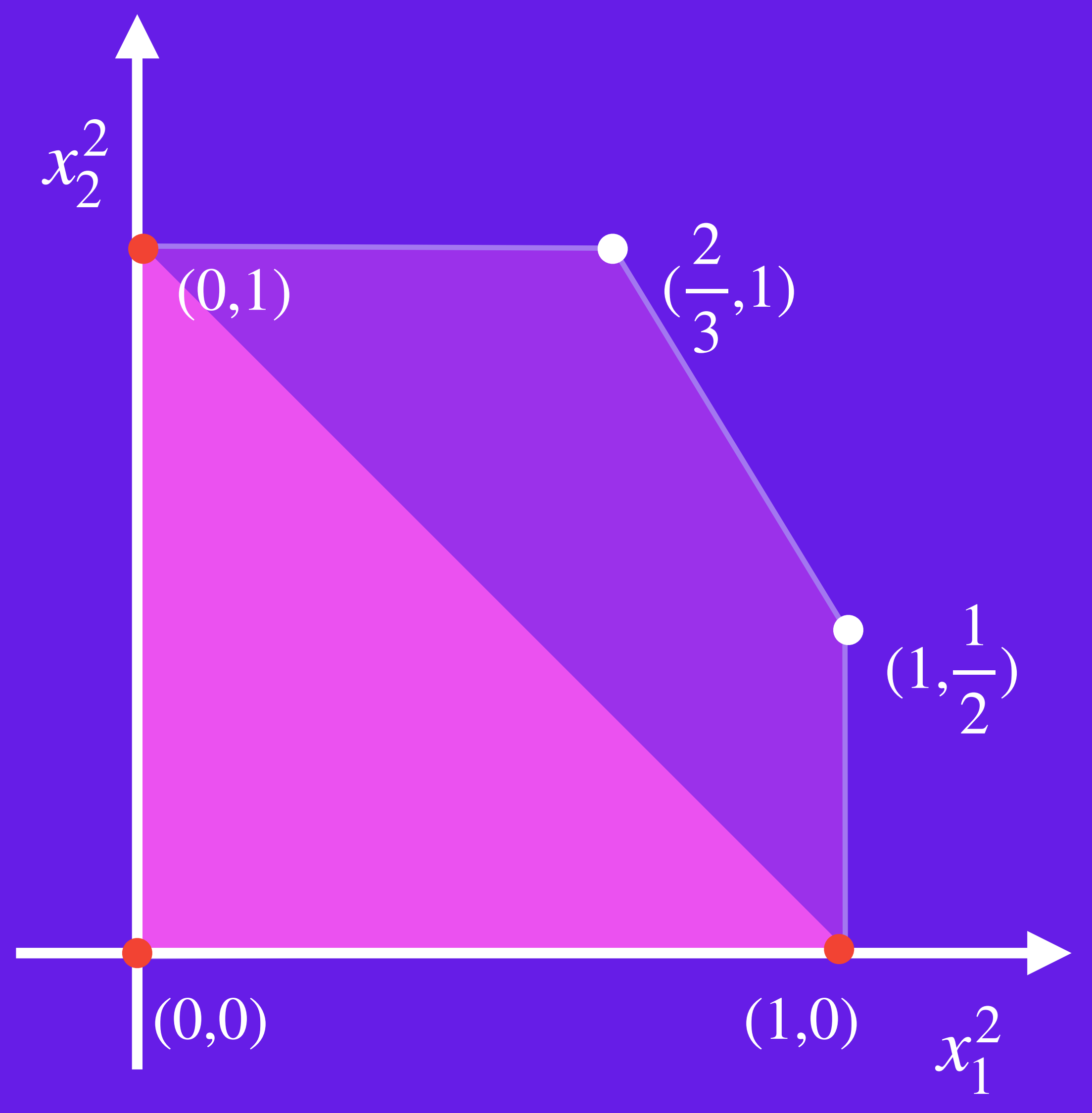
Sandholm et al., 2005;
Cronert and Minner, 2021;
Carvalho et al., 2022.

$$\max_{x^1} \quad 6x_1^1 + x_2^1 - 4x_1^1x_1^2 + 6x_2^1x_2^2$$
$$\text{s.t.} \quad 3x_1^1 + 2x_2^1 \leq 4$$
$$x^1 \in \{0,1\}^2$$

$$\max_{x^2} \quad 4x_1^2 + 2x_2^2 - x_1^2x_1^1 - x_2^2x_2^1$$
$$\text{s.t.} \quad 3x_1^2 + 2x_2^2 \leq 4$$
$$x^2 \in \{0,1\}^2$$

Best-responses





$$\max_{x^1} \quad 6x_1^1 + x_2^1 - 4x_1^1x_1^2 + 6x_2^1x_2^2$$
$$\text{s.t.} \quad 3x_1^1 + 2x_2^1 \leq 4$$
$$x^1 \in \{0,1\}^2$$

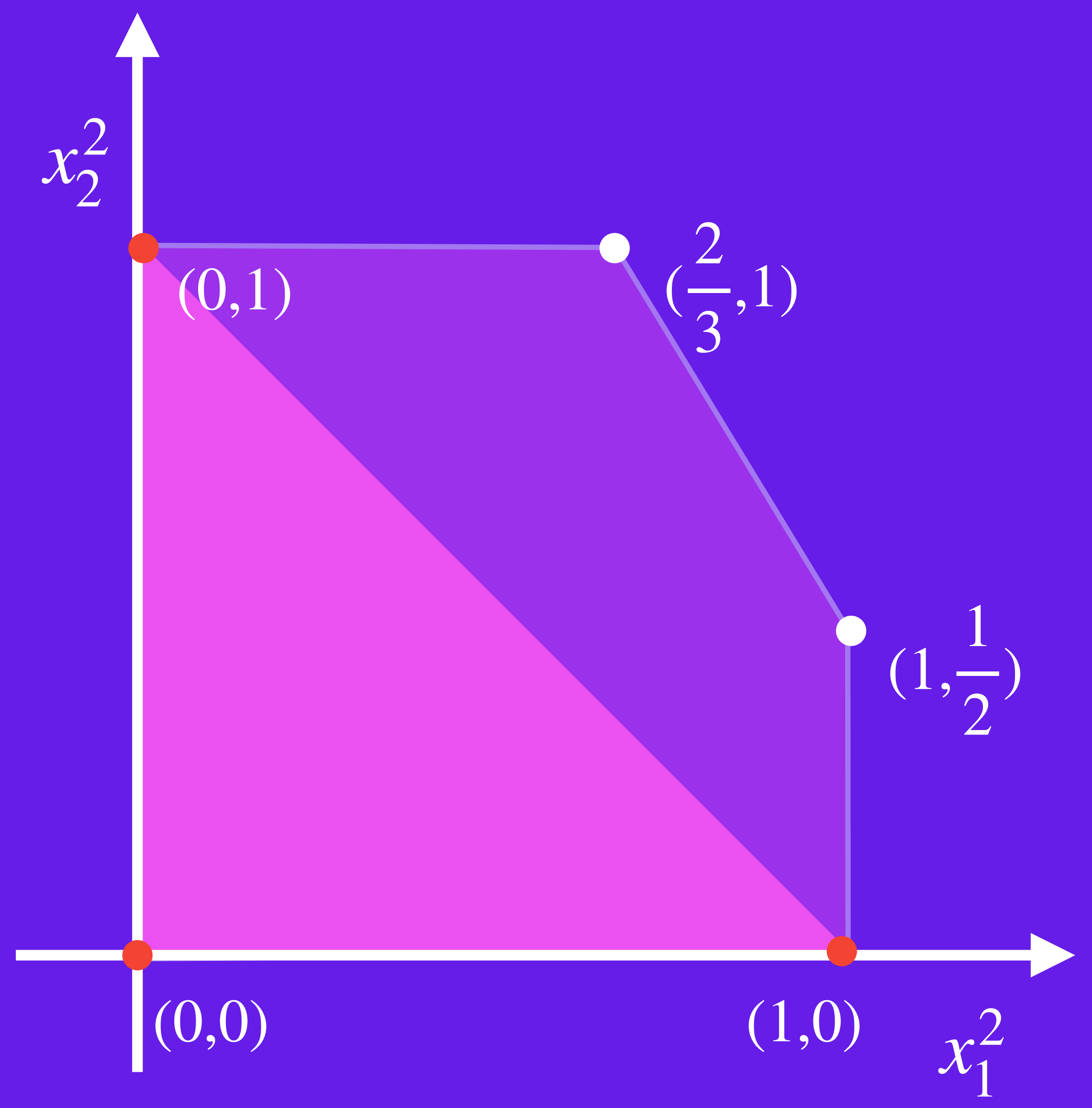
$$\max_{x^2} \quad 4x_1^2 + 2x_2^2 - x_1^2x_1^1 - x_2^2x_2^1$$
$$\text{s.t.} \quad 2x_1^2 + 3x_2^2 \leq 4$$
$$x^2 \in \{0,1\}^2$$

x^1 \ x^2	(0,0)	(1,0)	(0,1)
(0,0)	0 0	0 4	0 2
(1,0)	6 0	2 3	6 2
(0,1)	1 0	1 2	7 1

$$\max_{x^2} \quad 4x_1^2 + 2x_2^2 - x_1^2 x_1^1 - x_2^2 x_2^1$$

$$\text{s.t.} \quad 2x_1^2 + 3x_2^2 \leq 4$$

$$x^2 \in \{0,1\}^2$$

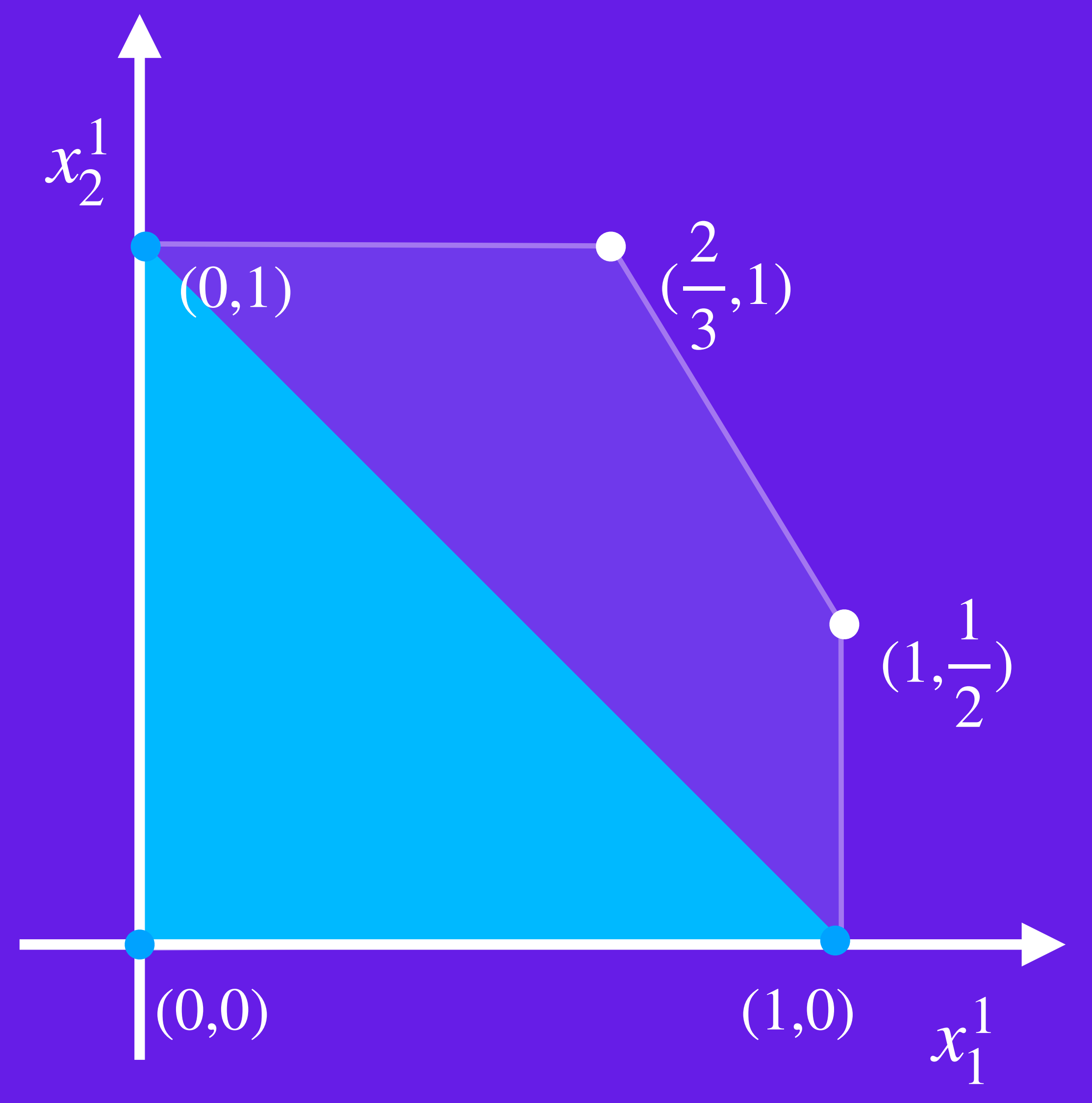


x^1 \ x^2	(0,0)	(1,0)	(0,1)
(0,0)	0 0	0 4	0 2
(1,0)	6 0	2 3	6 2
(0,1)	1 0	1 2	7 1

$$\max_{x^1} \quad 6x_1^1 + x_2^1 - 4x_1^1x_1^2 + 6x_2^1x_2^2$$

$$\text{s.t.} \quad 3x_1^1 + 2x_2^1 \leq 4$$

$$x^1 \in \{0,1\}^2$$



x^1 \ x^2	(0,0)	(1,0)	(0,1)
(0,0)	0 0	0 4	0 2
(1,0)	6 0	2 3	6 2
(0,1)	1 0	1 2	7 1

Caveat: this requires **an explicit enumeration** of the players' strategies…

		x^2		
		(0,0)	(1,0)	(0,1)
	(0,0)	0 0	0 4	0 2
x^1	(1,0)	6 0	2 3	6 2
	(0,1)	1 0	1 2	7 1

Caveat: this requires **an explicit enumeration** of the players' strategies...

What if we formulate a **Complementarity Problem** starting from the *linear relaxations of each player's problem*?

$$\max_{x^1} \quad (c^1)^\top x^1 + (x^{-1})^\top C^1 x^1$$

$$\text{s.t.} \quad A^1 x^1 \leq b^1$$

$$x^1 \in \{0,1\}^m$$

$$\max_{x^2} \quad (c^2)^\top x^2 + (x^{-2})^\top C^2 x^2$$

$$\text{s.t.} \quad A^2 x^2 \leq b^2$$

$$x^2 \in \{0,1\}^m$$

$$q = \begin{bmatrix} c^1 \\ b^1 \\ \vdots \\ c^n \\ b^n \end{bmatrix} \quad M = \begin{bmatrix} C^1 x^{-1} & A^{1\top} \\ -A^1 & 0 \\ \vdots & \\ C^n x^{-n} & A^{n\top} \\ -A^n & 0 \end{bmatrix}$$

$$z = M\sigma + q, \ \sigma^\top z = 0$$

$$\sigma \geq 0, \ z \geq 0$$

Provides all the MNEs for the game?

$$z = M\sigma + q,\ \sigma^\top z = 0$$

$$\sigma \geq 0,\ z \geq 0$$

Provides all the MNEs for the game?

Yes

If A^i, b^i **describe** cl conv$(\mathcal{X}^i)$, i.e., convex game

- Prohibitive in practice...

Maybe

If A^i, b^i **do not** describe cl conv$(\mathcal{X}^i)$

- Some MNEs may be excluded
- Some spurious MNEs may be introduced
- May not give bounds, as in Optimization

$$0 \leq \sigma \perp z = (M_t\sigma + q_t) \geq 0$$

Provides all the MNEs for the game?

Yes

If A^i, b^i **describe** cl conv$(\mathcal{X}^i)$

THEOREM *(the shortened version)*

Given an RBG $\mathbf{G}$ and a copy of it $\tilde{\mathbf{G}}$ where the feasible region of player $\boldsymbol{i}$ is cl conv $(\mathcal{X}^i)$ (instead of $\mathcal{X}^i$), then:

- *For any PNE $\tilde{\boldsymbol{\sigma}}$ of $\tilde{\mathbf{G}}$, there exists an MNE $\hat{\boldsymbol{\sigma}}$ of $\mathbf{G}$ so that each player get the same payoff in $\tilde{\mathbf{G}}$ and $\mathbf{G}$*
- *If $\tilde{\mathbf{G}}$ has no PNEs, then $\mathbf{G}$ has no MNEs.*

Yes

If A^i, b^i **describe** cl conv$(\mathscr{X}^i)$

Computing **MNEs**
in an RBG G

Computing **PNEs**
in a “convexified” RBG $\tilde{G}$

Yes

If A^i, b^i **describe** cl conv$(\mathscr{X}^i)$

Computing **MNEs**
in an RBG G

$\equiv$

Computing **PNEs**
in a "convexified" RBG $\tilde{G}$

The Idea

The Idea

Compute an MNE for an RBG G by computing (P)NEs for a series of "easier" convex games $\tilde{G}$

The Idea

Compute an MNE for an RBG G by computing (P)NEs for a series of "easier" convex games $\tilde{G}$

Maybe

If A^i, b^i **do not** describe cl conv$(\mathscr{X}^i)$

- Some MNEs may be excluded
- Some spurious MNEs may be introduced
- May not give bounds, as in Optimization

The Idea

Compute an MNE for an RBG G by computing (P)NEs for a series of "easier" convex games $\tilde{G}$

Maybe

If A^i, b^i do not describe cl conv$(\mathcal{X}^i)$

- Some MNEs may be excluded
- Some spurious MNEs may be introduced
- May not give bounds, as in Optimization

At each iteration, either we **find an MNE** for G or we **refine the approximation** in $\tilde{G}$

Approximation

PAG

Given the polyhedrally-representable *RBG* $\boldsymbol{G}$, we construct ***polyhedral approximate game*** $\tilde{G}$ where each $\boldsymbol{i}$ solves instead

$$\max_{x^i}\{f^i(x^i, x^{-i}) = (c^i)^\top x^i + (x^{-i})^\top C^i x^i : x^i \in \tilde{\mathcal{X}}^i\}$$

$$\tilde{\mathcal{X}}^i := \{\tilde{A}^i x^i \le \tilde{b}^i, x^i \ge 0\}, \mathcal{X}^i \subseteq \text{cl conv}(\mathcal{X}^i) \subseteq \tilde{\mathcal{X}}^i$$

Namely, $\tilde{\mathcal{X}}^i$ **(polyhedrally) outer approximates cl conv**$(\mathcal{X}^i)$

Finding MNEs

The LCP

$$\max_{x^i}\{f^i(x^i, x^{-i}) = (c^i)^\top x^i + (x^{-i})^\top C^i x^i : x^i \in \tilde{\mathscr{X}}^i\}$$

$$\tilde{\mathscr{X}}^i := \{\tilde{A}^i x^i \leq \tilde{b}^i, x^i \geq 0\}, \mathscr{X}^i \subseteq \text{cl conv}(\mathscr{X}^i) \subseteq \tilde{\mathscr{X}}^i$$

$$\tilde{q} = \begin{bmatrix} c^1 \\ \tilde{b}^1 \\ \vdots \\ c^n \\ \tilde{b}^n \end{bmatrix} \quad \tilde{M} = \begin{bmatrix} C^1 x^{-1} & \tilde{A}^{1\top} \\ -\tilde{A}^1 & 0 \\ \vdots & \\ C^n x^{-n} & \tilde{A}^{n\top} \\ -\tilde{A}^n & 0 \end{bmatrix}$$

$$\tilde{z} = \tilde{M}\tilde{\sigma} + \tilde{q},\ \tilde{\sigma}^\top \tilde{z} = 0$$

$$\tilde{\sigma} \geq 0,\ \tilde{z} \geq 0$$

Is $\tilde{\sigma}$ an MNE for G ?

Ask the Oracle

Oracle

Enhanced Sep. Oracle

Given a point $\bar{x}\ (=\tilde{\sigma}^i)$ *and* $\mathscr{X}\ (=\mathscr{X}^i)$*, the* **Enhanced Separation Oracle (ESO)** determines that either

$\bar{x} \in$ cl conv $(\mathscr{X})$ and an "extended proof"

$\bar{x} \notin$ cl conv $(\mathscr{X})$ + a cut for cl conv$(\mathscr{X})$ and $\bar{x}$

The ***extended proof*** is the support of $\bar{x}$, i.e. convex combination of elements in ext(cl conv$(\mathscr{X})$) and conic comb. of rays in rec(cl conv$(\mathscr{X})$).

In practice, the oracle builds a $\mathscr{V}$-polyhedral inner-approximation of cl conv $(\mathscr{X})$

Enhanced Separation Oracle

INPUT: A point $\bar{x}$ $(= \tilde{\sigma}^i)$ and $\mathcal{X}$ $(= \mathcal{X}^i)$ (a tolerance ε)

OUTPUT: yes and proof or no and a cut

$V = R = \emptyset$ or storage

```
Repeat:
  W ← conv(V) + cone(R)          👀 Inner approximation of cl conv(X)
  If x̄ ∈ W: return yes and proof of inclusion
  If x̄ ∉ W:
      π̄ᵀx ≤ π̄₀ separates x̄ and W
      G ← max_x{π̄ᵀx : x ∈ X} with ν maximizer
      If G = ∞: R ← R ∪ {r} with r extreme ray
      Else:
         If π̄ᵀν < π̄ᵀx̄: return no and π̄ᵀx ≤ π̄ᵀν
         Else: V ← V ∪ {ν}
```

In math notation:

- $\mathscr{W} \leftarrow \mathrm{conv}(V) + \mathrm{cone}(R)$ — 👀 *Inner approximation of cl conv$(\mathcal{X})$*
- If $\bar{x} \in \mathscr{W}$: return yes and proof of inclusion
- If $\bar{x} \notin \mathscr{W}$:
  - $\bar{\pi}^\top x \leq \bar{\pi}_0$ separates $\bar{x}$ and $\mathscr{W}$
  - $\mathscr{G} \leftarrow \max_x \{\bar{\pi}^\top x : x \in \mathcal{X}\}$ with ν maximizer
  - If $\mathscr{G} = \infty$: $R \leftarrow R \cup \{r\}$ with r extreme ray
  - Else:
    - If $\bar{\pi}^\top \nu < \bar{\pi}^\top \bar{x}$: return no and $\bar{\pi}^\top x \leq \bar{\pi}^\top \nu$
    - Else: $V \leftarrow V \cup \{\nu\}$

Enhanced Separation Oracle

INPUT: A point $\bar{x}\ (=\tilde{\sigma}^i)$ and $\mathcal{X}\ (=\mathcal{X}^i)$ (a tolerance ε)

OUTPUT: yes and proof or no and a cut

$V = R = \emptyset$ or storage

Repeat:

$\mathcal{W} \leftarrow \mathrm{conv}(V) + \mathrm{cone}(R)$

If $\bar{x} \in \mathcal{W}$: return yes and proof of inclusion

If $\bar{x} \notin \mathcal{W}$:

This is an LP

$\bar{\pi}^\top x \leq \bar{\pi}_0$ separates $\bar{x}$ and $\mathcal{W}$

$\mathcal{G} \leftarrow \max_x \{\bar{\pi}^\top x : x \in \mathcal{X}\}$ with ν maximizer

If $\mathcal{G} = \infty$: $R \leftarrow R \cup \{r\}$ with r extreme ray

Else:

If $\bar{\pi}^\top \nu < \bar{\pi}^\top \bar{x}$: return no and $\bar{\pi}^\top x \leq \bar{\pi}^\top \nu$

Else: $V \leftarrow V \cup \{\nu\}$

Enhanced Separation Oracle

$$
\begin{aligned}
\max_{\pi,\pi_0} \quad & \bar{x}^\top \pi - \pi_0 \\
& \pi v_k^\top - \pi_0 \leq 0 \quad \forall v_k \in V \\
& \pi r_j^\top \leq 0 \quad \forall r_j \in R \\
& e^\top (u + v) = 1 \\
& \pi + u - v = 0 \\
& u, v \geq 0
\end{aligned}
$$

YES

Objective is 0

Enhanced Separation Oracle

INPUT: A point $\bar{x}$ ($= \tilde{\sigma}^i$) and $\mathscr{X}$ ($= \mathscr{X}^i$) (a tolerance ε)

OUTPUT: yes and proof or no and a cut

$V = R = \emptyset$ or storage

Repeat:

$\mathscr{W} \leftarrow \texttt{conv}(V) + \texttt{cone}(R)$

If $\bar{x} \in \mathscr{W}$: return yes and proof of inclusion

If $\bar{x} \notin \mathscr{W}$:

$\bar{\pi}^\top x \leq \bar{\pi}_0$ separates $\bar{x}$ and $\mathscr{W}$

$\mathscr{G} \leftarrow \max_x \{\bar{\pi}^\top x : x \in \mathscr{X}\}$ with ν maximizer

If $\mathscr{G} = \infty$: $R \leftarrow R \cup \{r\}$ with r extreme ray

Else:

If $\bar{\pi}^\top \nu < \bar{\pi}^\top \bar{x}$: return no and $\bar{\pi}^\top x \leq \bar{\pi}^\top \nu$

Else: $V \leftarrow V \cup \{\nu\}$

Enhanced Separation Oracle

INPUT: A point $\bar{x}\ (=\tilde{\sigma}^i)$ and $\mathscr{X}\ (=\mathscr{X}^i)$ (a tolerance ε)

OUTPUT: yes and proof or no and a cut

$V = R = \varnothing$ or storage

Repeat:

$\mathscr{W} \leftarrow \mathrm{conv}(V) + \mathrm{cone}(R)$

If $\bar{x} \in \mathscr{W}$: return yes and proof of inclusion

If $\bar{x} \notin \mathscr{W}$:

$\bar{\pi}^\top x \leq \bar{\pi}_0$ separates $\bar{x}$ and $\mathscr{W}$

$\mathscr{G} \leftarrow \max_x \{\bar{\pi}^\top x : x \in \mathscr{X}\}$ with ν maximizer

If $\mathscr{G} = \infty$: $R \leftarrow R \cup \{r\}$ with r extreme ray

Else:

If $\bar{\pi}^\top \nu < \bar{\pi}^\top \bar{x}$: return no and $\bar{\pi}^\top x \leq \bar{\pi}^\top \nu$

Else: $V \leftarrow V \cup \{\nu\}$

Enhanced Separation Oracle

INPUT: A point $\bar{x}\ (=\tilde{\sigma}^i)$ and $\mathscr{X}\ (=\mathscr{X}^i)$ (a tolerance ε)

OUTPUT: yes and proof or no and a cut

$V = R = \emptyset$ or storage

```
Repeat:
  W ← conv(V) + cone(R)
  If x̄ ∈ W: return yes and proof of inclusion
  If x̄ ∉ W:
      π̄ᵀx ≤ π̄₀ separates x̄ and W
      G ← max_x{π̄ᵀx : x ∈ X} with ν maximizer
      If G = ∞: R ← R ∪ {r} with r extreme ray
      Else:
        If π̄ᵀν < π̄ᵀx̄: return no and π̄ᵀx ≤ π̄ᵀν
        Else: V ← V ∪ {ν}
```

Repeat:

$\mathscr{W} \leftarrow \mathrm{conv}(V) + \mathrm{cone}(R)$

If $\bar{x} \in \mathscr{W}$: return yes and proof of inclusion

If $\bar{x} \notin \mathscr{W}$:

$\bar{\pi}^\top x \leq \bar{\pi}_0$ separates $\bar{x}$ and $\mathscr{W}$

$\mathscr{G} \leftarrow \max_x\{\bar{\pi}^\top x : x \in \mathscr{X}\}$ with ν maximizer

If $\mathscr{G} = \infty$: $R \leftarrow R \cup \{r\}$ with r extreme ray

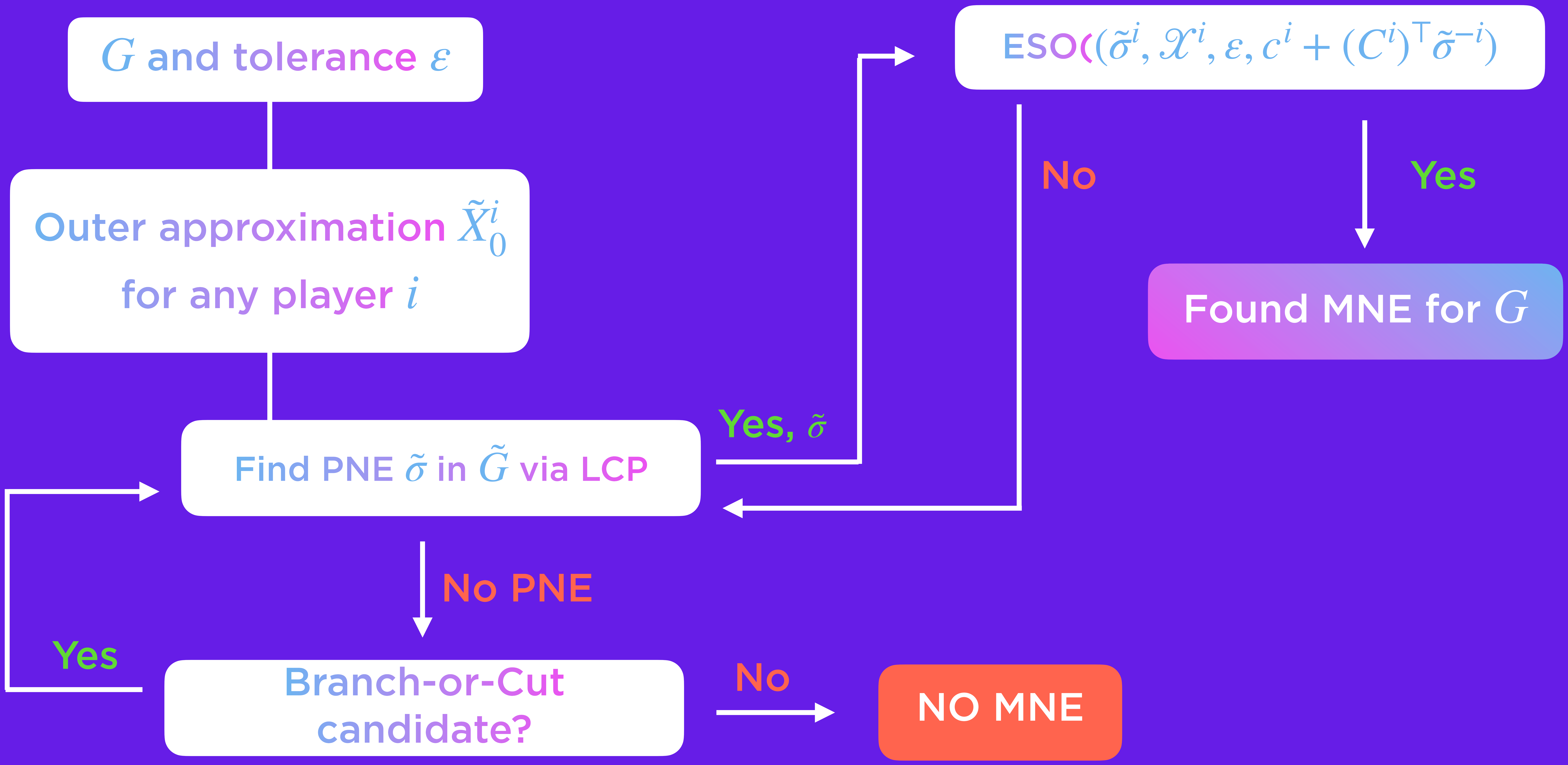
Else:

If $\bar{\pi}^\top \nu < \bar{\pi}^\top \bar{x}$: return no and $\bar{\pi}^\top x \leq \bar{\pi}^\top \nu$

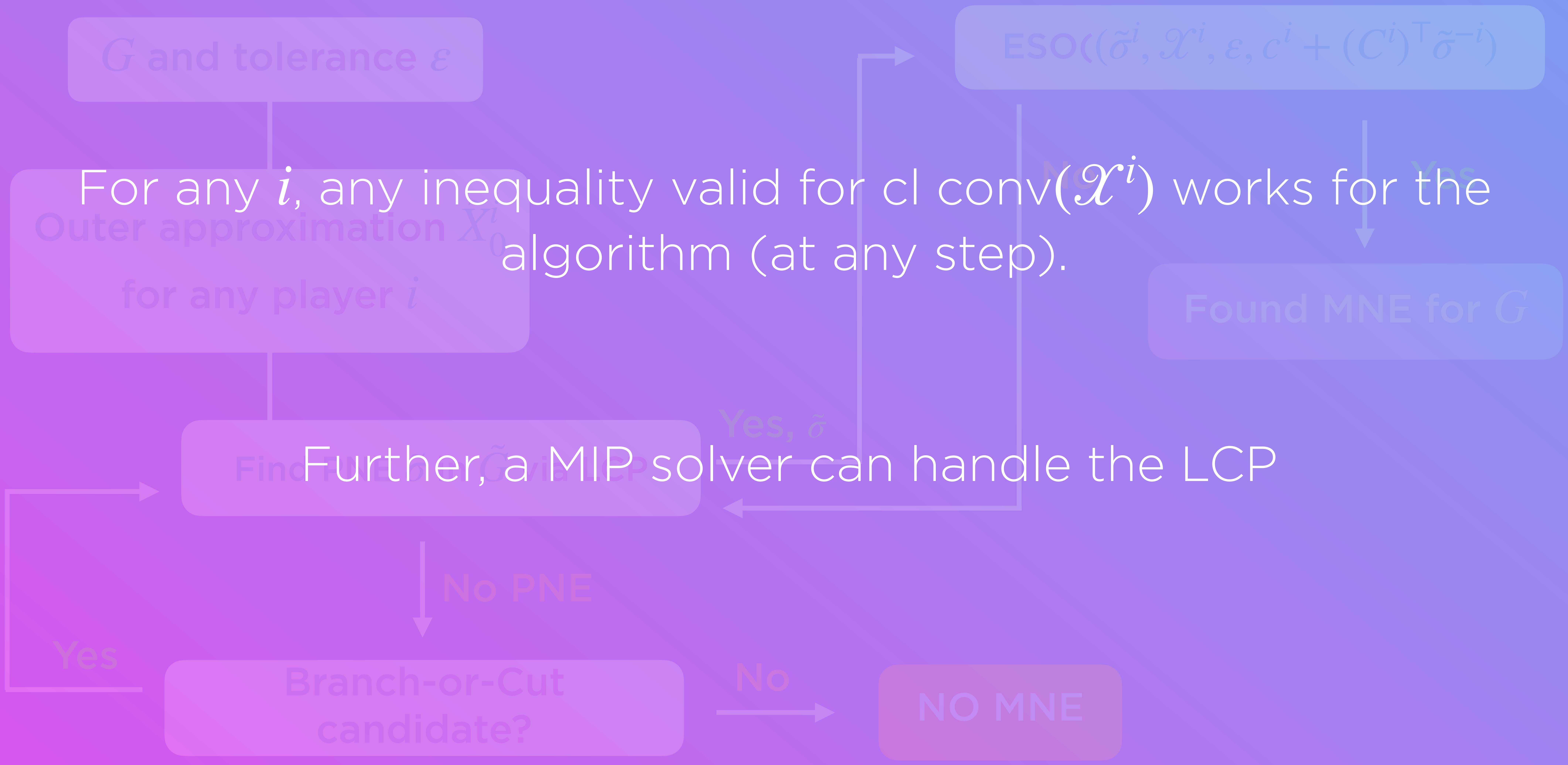
Else: $V \leftarrow V \cup \{\nu\}$

The Cut-and-Play

The Cut-And-Play



The Cut-And-Play



For any i, any inequality valid for cl conv$(\mathscr{X}^i)$ works for the algorithm (at any step).

Further, a MIP solver can handle the LCP

Experiments

Knapsack Game (*KPG*)

As for Wizard and Fairy, each player solves a binary Knapsack problem with some interaction terms in the objective

$$\max_{x^i} \left\{ \sum_{j=1}^{m} p_j^i x_j^i + \sum_{k=1, k \neq i}^{n} \sum_{j=1}^{m} C_{k,j}^i x_j^i x_j^k : \sum_{j=1}^{m} w_j^i x_j^i \leq b^i, \mathbf{x}^i \in \{0,1\}^m \right\}$$

W.l.o.g., each player controls m items

Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency ("~PoS")
SGM	-		0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
	SocialW	0	6.13	0	5.73	6.47	2.30	1.22
	SocialW	1	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
	-	0	0.05	0	5.27	5.9	2.07	1.35
	-	1	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-1	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
	SocialW	1	57.85	1	11.62	12.62	3.45	1.26
	SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
	-	1	4.48	0	9.62	10.25	2.42	1.30
	-	-1	4.32	0	8.22	14.35	8.43	1.30

Knapsack Game

	Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency (“~PoS”)
Small $nm \leq 80$	SGM	-		0.73	0	8.43	-	-	1.37
	CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
		SocialW	0	6.13	0	5.73	6.47	2.30	1.22
		SocialW	1	6.31	0	3.50	9.6	7.47	1.21
	CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
		-	0	0.05	0	5.27	5.9	2.07	1.35
		-	1	0.04	0	3.23	8.87	7.10	1.33
Large $nm > 80$	SGM	-	-1	20.86	6	18.58	-	-	1.50
	CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
		SocialW	1	57.85	1	11.62	12.62	3.45	1.26
		SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
	CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
		-	1	4.48	0	9.62	10.25	2.42	1.30
		-	-1	4.32	0	8.22	14.35	8.43	1.30

Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency ("~PoS")
SGM	-		0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
	SocialW	0	6.13	0	5.73	6.47	2.30	1.22
	SocialW	1	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
	-	0	0.05	0	5.27	5.9	2.07	1.35
	-	1	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-1	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
	SocialW	1	57.85	1	11.62	12.62	3.45	1.26
	SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
	-	1	4.48	0	9.62	10.25	2.42	1.30
	-	-1	4.32	0	8.22	14.35	8.43	1.30

Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency ("~PoS")
SGM	-		0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
	SocialW	0	6.13	0	5.73	6.47	2.30	1.22
	SocialW	1	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
	-	0	0.05	0	5.27	5.9	2.07	1.35
	-	1	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-1	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
	SocialW	1	57.85	1	11.62	12.62	3.45	1.26
	SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
	-	1	4.48	0	9.62	10.25	2.42	1.30
	-	-1	4.32	0	8.22	14.35	8.43	1.30

Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency ("~PoS")
SGM	-		0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
	SocialW	0	6.13	0	5.73	6.47	2.30	1.22
	SocialW	1	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
	-	0	0.05	0	5.27	5.9	2.07	1.35
	-	1	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-1	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
	SocialW	1	57.85	1	11.62	12.62	3.45	1.26
	SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
	-	1	4.48	0	9.62	10.25	2.42	1.30
	-	-1	4.32	0	8.22	14.35	8.43	1.30

Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency ("~PoS")
SGM	-	-	0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	❌	6.58	0	7.80	9.57	0.00	1.21
	SocialW	✅	6.13	0	5.73	6.47	2.30	1.22
	SocialW	✅✅	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	❌	0.36	0	7.60	10.2	0.00	1.21
	-	✅	0.05	0	5.27	5.9	2.07	1.35
	-	✅✅	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	❌	61.08	0	13.70	17.0	0.00	1.23
	SocialW	✅	57.85	1	11.62	12.62	3.45	1.26
	SocialW	✅✅	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	❌	6.68	0	13.55	16.35	0.00	1.24
	-	✅	4.48	0	9.62	10.25	2.42	1.30
	-	✅✅	4.32	0	8.22	14.35	8.43	1.30

Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency (“~PoS”)
SGM	-		0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
	SocialW	0	6.13	0	5.73	6.47	2.30	1.22
	SocialW	1	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
	-	0	0.05	0	5.27	5.9	2.07	1.35
	-	1	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-1	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
	SocialW	1	57.85	1	11.62	12.62	3.45	1.26
	SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
	-	1	4.48	0	9.62	10.25	2.42	1.30
	-	-1	4.32	0	8.22	14.35	8.43	1.30

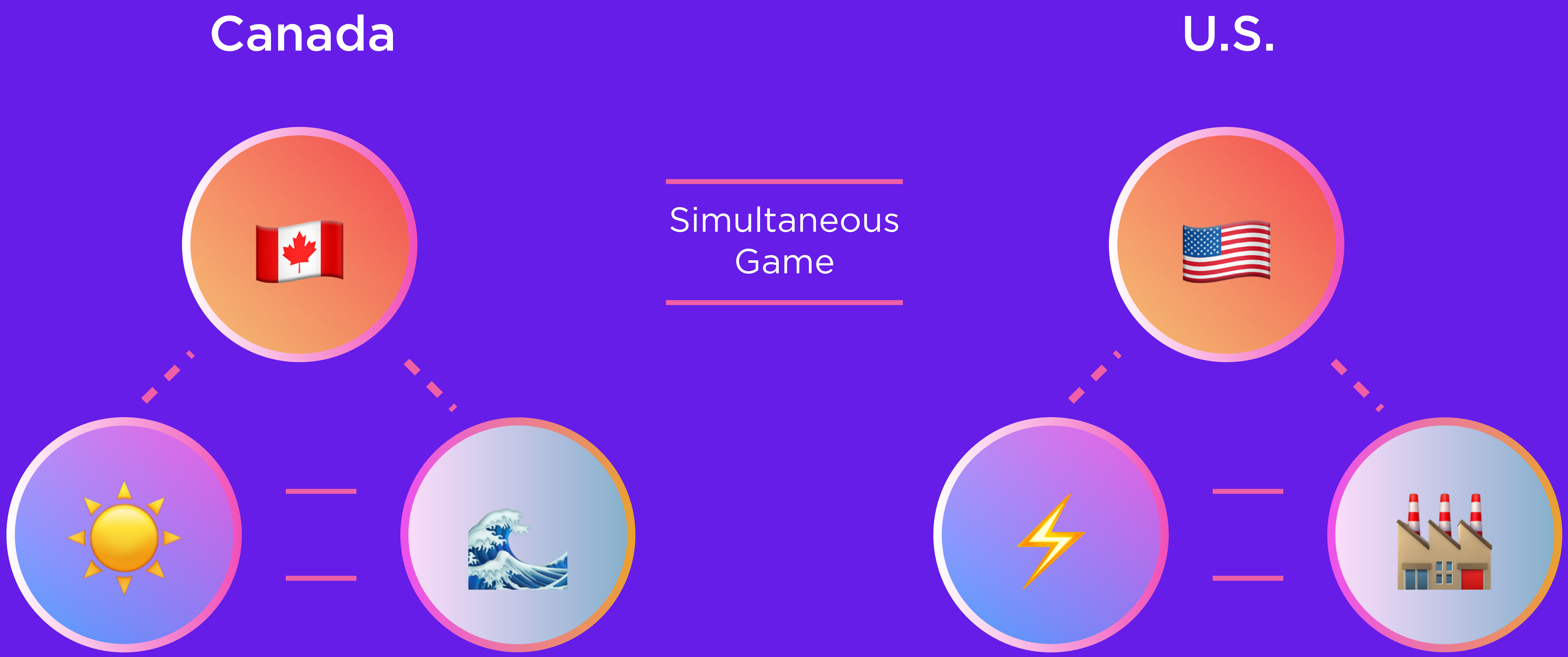
Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency (“~PoS”)
SGM	-		0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
	SocialW	0	6.13	0	5.73	6.47	2.30	1.22
	SocialW	1	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
	-	0	0.05	0	5.27	5.9	2.07	1.35
	-	1	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-1	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
	SocialW	1	57.85	1	11.62	12.62	3.45	1.26
	SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
	-	1	4.48	0	9.62	10.25	2.42	1.30
	-	-1	4.32	0	8.22	14.35	8.43	1.30

Knapsack Game

Algo	Obj	A	Geo t (s)	#TL	#It	Cuts	MIP	Efficiency ("~PoS")
SGM	-		0.73	0	8.43	-	-	1.37
CnP-MIP	SocialW	-1	6.58	0	7.80	9.57	0.00	1.21
	SocialW	0	6.13	0	5.73	6.47	2.30	1.22
	SocialW	1	6.31	0	3.50	9.6	7.47	1.21
CnP-PATH	-	-1	0.36	0	7.60	10.2	0.00	1.21
	-	0	0.05	0	5.27	5.9	2.07	1.35
	-	1	0.04	0	3.23	8.87	7.10	1.33
SGM	-	-1	20.86	6	18.58	-	-	1.50
CnP-MIP	SocialW	0	61.08	0	13.70	17.0	0.00	1.23
	SocialW	1	57.85	1	11.62	12.62	3.45	1.26
	SocialW	-1	68.20	0	9.48	16.8	10.32	1.23
CnP-PATH	-	0	6.68	0	13.55	16.35	0.00	1.24
	-	1	4.48	0	9.62	10.25	2.42	1.30
	-	-1	4.32	0	8.22	14.35	8.43	1.30

NASPs



This is a simultaneous game among bilevel (i.e., sequential) programs (**NASP**)

Are leaders (countries) further reducing their emission
if they optimize the income from a carbon-tax?

Does trade among countries under a carbon-tax reduce emissions?

Are leaders (countries) further reducing their emission if they optimize the income from a carbon-tax?

It depends on what source energy producers use (i.e., coal vs solar).
In general, **no.**

Does trade among countries under a carbon-tax reduce emissions?

Are leaders (countries) further reducing their emission if they optimize the income from a carbon-tax?

It depends on what source energy producers use (i.e., coal vs solar).
In general, **no.**

Does trade among countries under a carbon-tax reduce emissions?

Since trade is about money, the **intuitive answer is no.**
However, we found that countries with large quantities of clean energy can fulfil the need of countries with fossil fuel, **thus reducing the overall emissions.**

Remarks, Ideas, Directions

Some Remarks

In MPGs, the plausibility of the Nash equilibrium can only stem from the availability of ***efficient tools*** *to compute it.*

Optimization Framework

Scalable and flexible

Hybridization


MPGs
IPGs
RBGs
Finite Games

MPGs

If non-convexities are not necessarily integer:

$$\max_{x^i} \{ f^i(x^i, x^{-i}) = (c^i)^\top x^i + (x^{-i})^\top C^i x^i : x^i \in \mathcal{X}^i \}$$

So-called **Reciprocally-Bilinear Games**

Margarida Carvalho, Gabriele Dragotto, Andrea Lodi, Sriram Sankaranarayanan, *The Cut and Play Algorithm: Computing Nash Equilibria via Outer Approximations*, **arXiv:2111.05726**

An MPG library

ZERO

https://github.com/ds4dm/ZERO

Gabriele Dragotto, Sriram Sankaranarayanan, Margarida Carvalho, Andrea Lodi, *ZERO: Playing Mathematical Programming Games*, **arXiv:2111.07932**


```
Models::IPG::IPG IPG_Model(&GurobiEnv, IPG_Instance);
// Select the equilibrium to compute a Nash Equilibrium
IPG_Model.setAlgorithm(Data::IPG::Algorithms::CutAndPlay);
// Extra parameters
IPG_Model.setDeviationTolerance(3e-4);
IPG_Model.setNumThreads(8);
IPG_Model.setLCPAlgorithm(Data::LCP::Algorithms::PATH);

// Lock the model
IPG_Model.finalize();
// Run!
IPG_Model.findNashEq();
```

Directions

Methodology

Developments of efficient algorithms and theoretical frameworks to handle **complex non-convex problems**

Rational behavior through inequalities and Optimization, new solutions concepts

Practice

MPGs and applications

Fairness

Companies, governments, and in general, organizations are likely to solve optimization problems. Trade-off ***selfishness and social good***

Methodology

Practice

Fairness

Margarida Carvalho, Gabriele Dragotto, Andrea Lodi, Sriram Sankaranarayanan, *The Cut and Play Algorithm: Computing Nash Equilibria via Outer Approximations*, **arXiv:2111.05726**

Margarida Carvalho, Gabriele Dragotto, Felipe Feijoo, Andrea Lodi, Sriram Sankaranarayanan, *When Nash Meets Stackelberg*, **arXiv:1910.06452**

Carvalho M, Lodi A, Pedroso J (2022) Computing equilibria for integer programming games. **European Journal of Operational Research**

Carvalho M, Lodi A, Pedroso JP, Viana A (2017) Nash equilibria in the two-player kidney exchange game. **Mathematical Programming** *161(1-2):389–417*

David Fuller J, C̨elebi E (2017) Alternative models for markets with nonconvexities. **European Journal of Operational Research** *261(2):436–449,*

Facchinei F, Pang JS, eds. (2004) Finite-Dimensional Variational Inequalities and Complementarity Problems. **Springer Series in Operations Research and Financial Engineering**

Ferris, M.C. and Munson, T.S., 1999. Interfaces to PATH 3.0: Design, implementation and usage. **Computational Optimization and Applications**, *12(1), pp.207-227.*

Gabriel SA, Siddiqui SA, Conejo AJ, Ruiz C (2013) Solving Discretely-Constrained Nash–Cournot Games with an Application to Power Markets. **Networks and Spatial Economics** *13(3):307–326,*

Koeppe M, Ryan CT, Queyranne M (2011) Rational Generating Functions and Integer Programming Games. **Operations Research** *59(6):1445–1460*

Extra

Comparing MPGs

Equilibrium Programming

❌ Often $\mathscr{X}^i$ is continuous

❌ Algos: Complementarity or V.I.

❌ Global convergence?

❌ Non-convexities?

✅ Efficient in well-behaved cases

Normal/Extensive-form games

❌ No complex operational constraints

❌ Explicit (and *burdensome*) representation of action sets

✅ Popular in Game Theory literature

When Nash Meets Stackelberg

Joint work with **Margarida Carvalho, Felipe Feijoo, Andrea Lodi** and **Sriram Sankaranarayanan**

3

Contributions

Complexity

It is Σ_2^p-hard to determine a MNE/PNE, in general

Algorithms

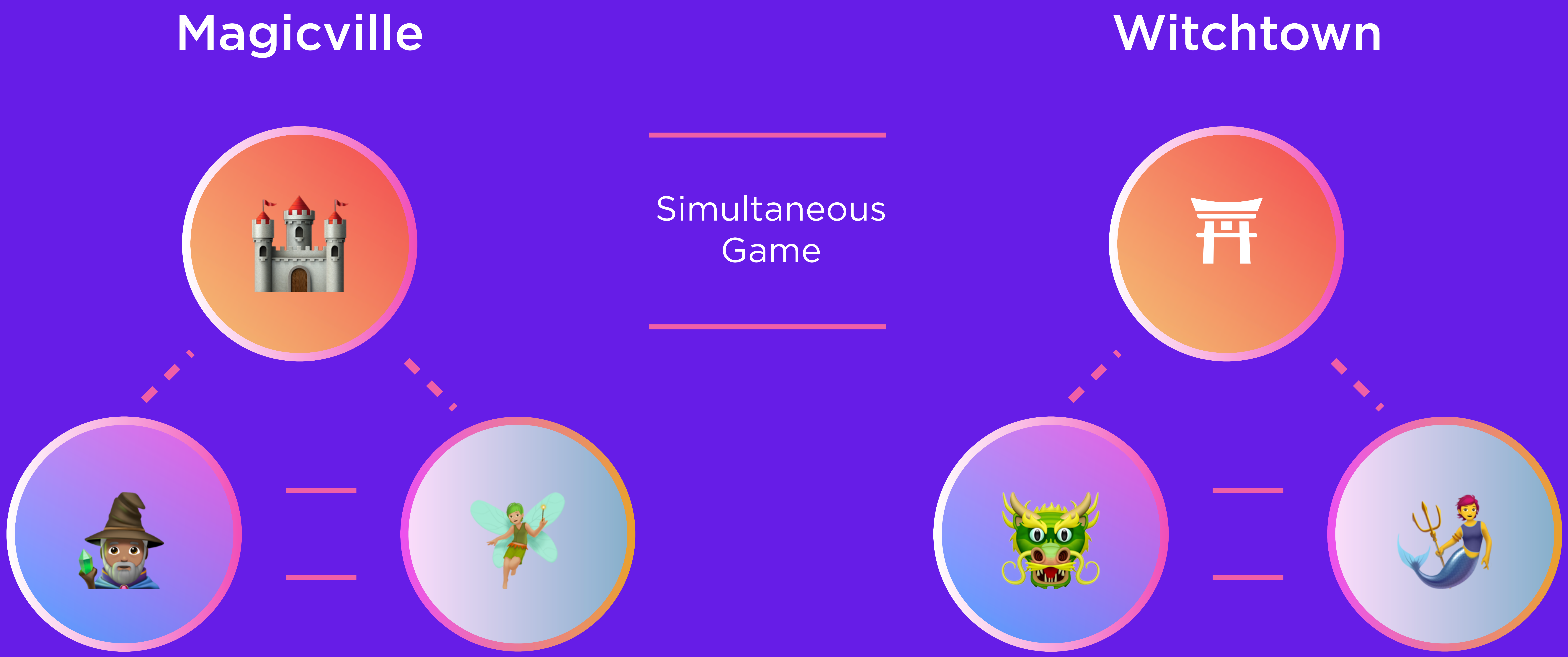
A **full enumeration scheme**, and an inner **approximation scheme**

Insights

Energy market tests, with **Chilean-Argentinean case study**

Magicville

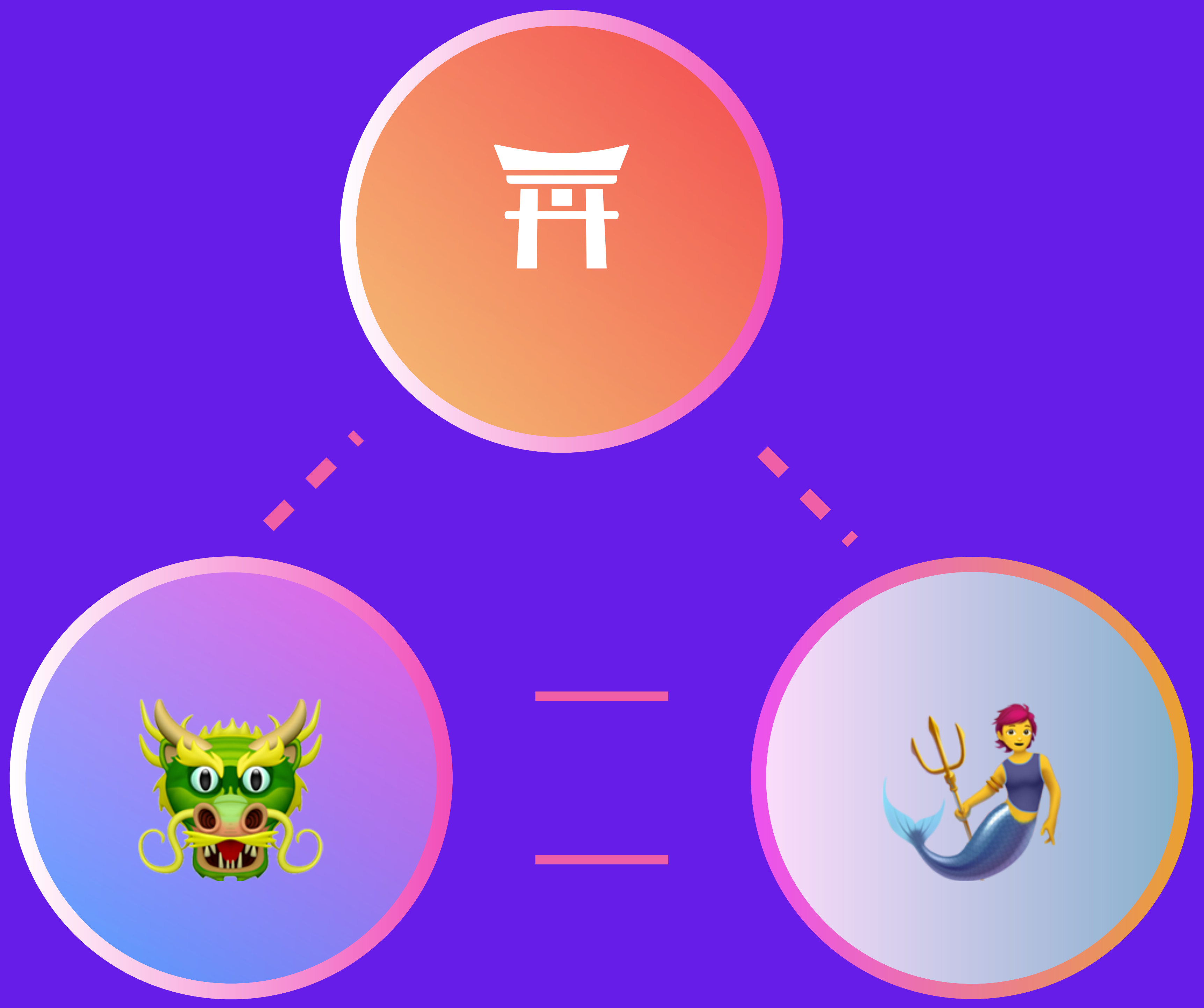
Reformulate each Stackelberg game as a single-level Optimization problem



Then, the game is an **RBG,** if objectives are compatible

Magicville

Witchtown



Among the reformulated bilevel programs, namely the ***real players***

Magicville

$$\max_{x^i} \{ (c^i)^\top x^i + (x^{-i})^\top C^i x^i : \underline{x^i \in \mathscr{F}^i} \}$$

The reformulated feasible region includes the KKT for the followers' problems

$$\mathcal{F}^i = \left\{ \begin{array}{l} A^i x^i \leq b^i \\ z^i = M^i x^i + q^i \\ x^i \geq 0, z^i \geq 0 \end{array} \right\} \bigcap_{j \in \mathcal{C}^i} (\{z^i_j = 0\} \cup \{x^i_j = 0\}).$$

Magicville

$$\max_{x^i} \{ (c^i)^\top x^i + (x^{-i})^\top C^i x^i : x^i \in \mathcal{F}^i \}$$

Algorithms

Fully enumerate cl conv$(\mathcal{F}^i)$

Inner approximate cl conv$(\mathcal{F}^i)$ (dual to CnP)

Magicville

Algorithms

	Time (s)	# TL
Fully enumerate cl conv($\mathscr{F}^i$)	120.2	9/149
Inner approximate cl conv($\mathscr{F}^i$)	3.73	0/149

The Problem(s)

Stackelberg Games

(Stackelberg, 1934;
Candler and Norto, 1977)

A Stackelberg game is a **sequential game** with **perfect information** where the players act in **rounds:**

- We consider games where there is an **unique** first-round player called ***the leader,*** who solves an optimization problem
- The second-round players are ***the followers*** solving optimization problems **depending on the leader's choices**

A solution is a vector of strategies that are optimal for both the leader and its followers

In the general case, determining a solution is $\mathcal{NP}$-hard

Could we reformulate the Stackelberg game as a single optimization problem?

Not always, yet...

The Problem(s)

Stackelberg Games

(Basu et al., 2020)

A Stackelberg game **can be reformulated into a single-level optimization problem** if:

1. The leader's objective function is linear in its variables and the ones of its followers
2. The leader's constraints are linear constraints
3. The followers solve convex quadratic optimization problems

Specifically, the feasible region of this program is a **union of polyhedra**

Complexity

Complexity

NASPs

THEOREM

Given a NASP with 2 leaders with 1 follower each, so that each follower solves a linear program and the leaders all have linear objectives in their variables:

1. It is $\Sigma_2^p -$ hard to determine if the game an MNE/PNE
2. If all reformulated problems have a bounded feasible region $\mathscr{F}^i$, **there exists an MNE**

Algorithmic Ideas

Full Enumeration

INPUT: A NASP N

OUTPUT: a NE or none exists

```
For every player i = 1,2,…,n
  Compute cl conv(𝓕^i) through Balas'

Solve an LCP with the convex hulls
If LCP has a solution: return yes and NE
Else: return no NE exists
```

Inner approximation

Inner-approximate cl conv$(\mathscr{X}^i)$

Inner approximation

Compute an MNE starting with a single polyhedron

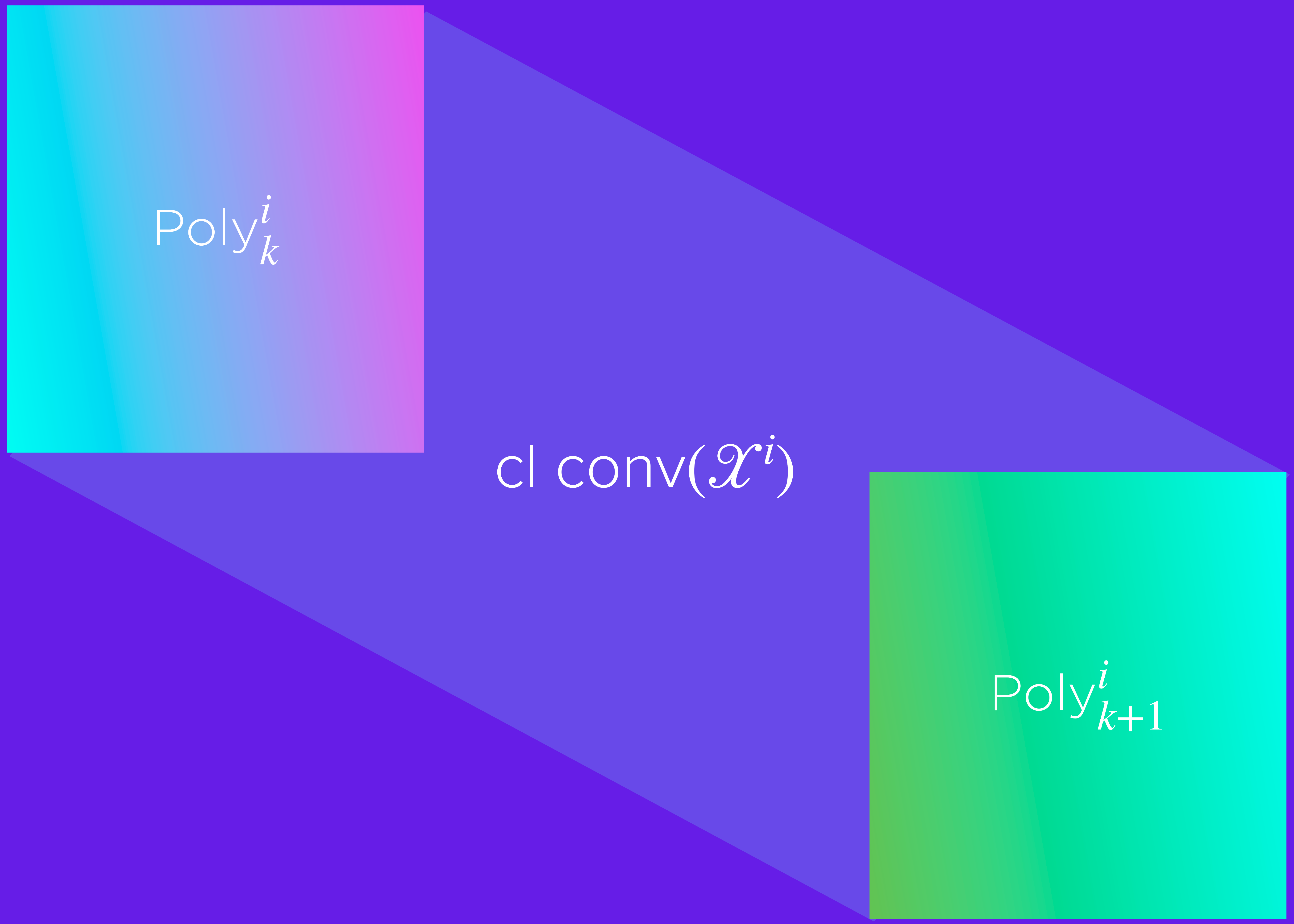
Inner approximation

Poly_k^i

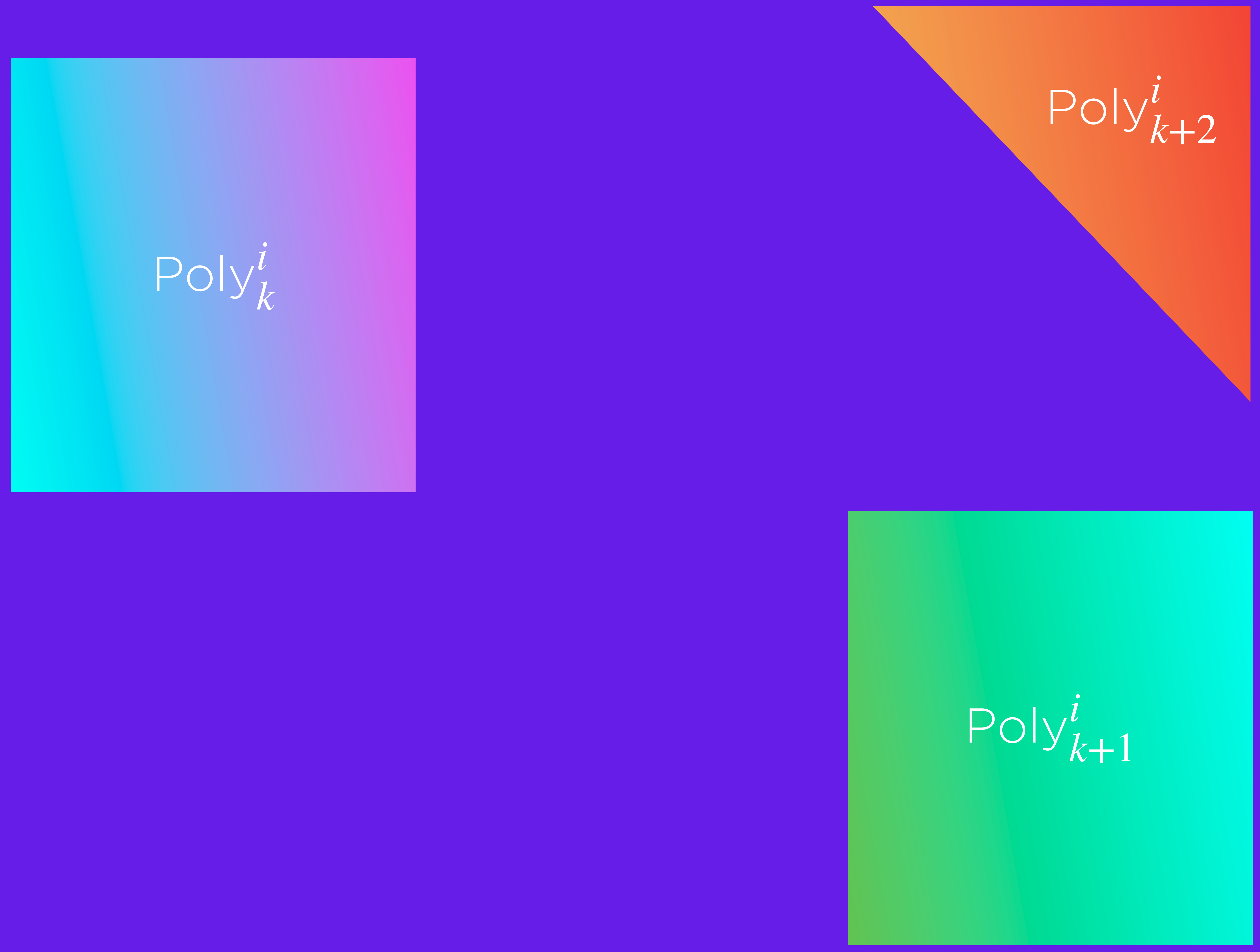
If there **exists a NE,** and also a **deviation**, add it to the next iteration

Poly_{k+1}^i

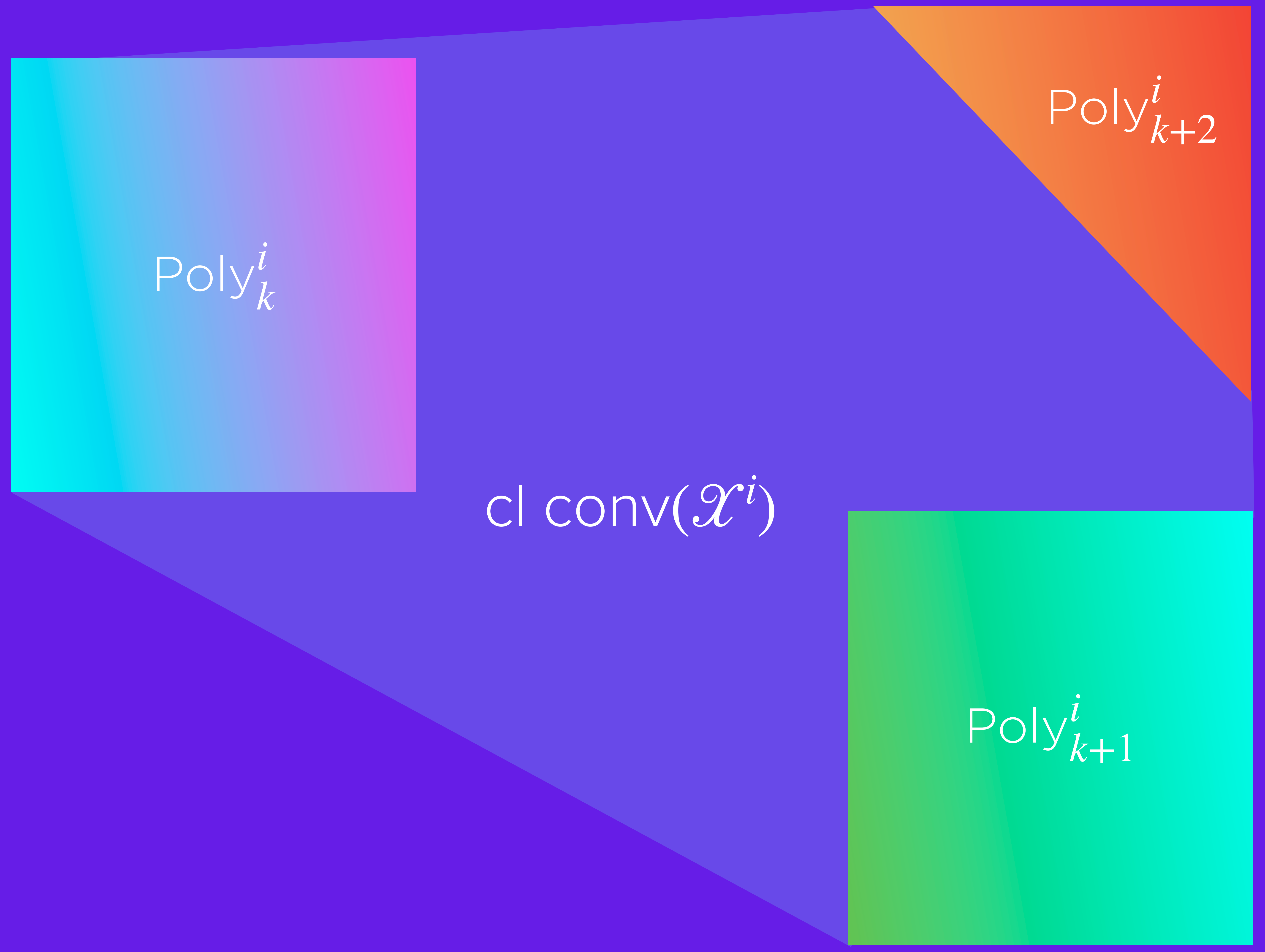
Inner approximation



Inner approximation



Inner approximation



Inner approximation

INPUT: A NASP N

OUTPUT: a NE or none exists

```
For every player i = 1,2,…,n
  Initialize 𝓕^i_* with one polyhedron from the union
While True:
  Solve an LCP to determine an NE
  If LCP has a solution:
      If no deviation: return yes and NE
      Else deviation for i: add the polyhedron to 𝓕^i_*
  If LCP has no solution:
      If no more polyhedra: return none exists
      Else: add random polyhedra to 𝓕^i_*
```

Clean Energy Experiments

Energy Game

	Algorithm	ES	k	Time (s) EQ	Time (s) NO	Time (s) All	Wins EQ	Wins NO	Solved
	FE	-	-	29.08	0.12	120.21	6	82	140/149
MNE	*InnerApp*	Seq	1	6.65	0.35	51.33	3	0	145/149
		Seq	3	17.76	0.18	55.82	5	0	145/149
		Seq	5	6.40	0.15	51.08	3	0	145/149
		Rev.Seq	1	7.97	0.36	3.73	26	0	149/149
		Rev.Seq	3	11.29	0.18	53.12	4	0	145/149
		Rev.Seq	5	9.53	0.15	76.41	5	0	143/149
		Random	1	5.22	0.36	26.60	8	0	147/149
		Random	3	32.42	0.18	85.65	5	0	143/149
		Random	5	23.67	0.15	58.26	2	0	145/149
PNE	*FE-P*	-	-	7.25	0.12	328.23	–	–	122/149

Energy Game

Small

	Algorithm	ES	k	Time (s) EQ	NO	All	Wins EQ	NO	Solved
	FE	-	-	29.08	0.12	120.21	6	82	140/149
MNE	*InnerApp*	Seq	1	6.65	0.35	51.33	3	0	145/149
		Seq	3	17.76	0.18	55.82	5	0	145/149
		Seq	5	6.40	0.15	51.08	3	0	145/149
		Rev.Seq	1	7.97	0.36	3.73	26	0	149/149
		Rev.Seq	3	11.29	0.18	53.12	4	0	145/149
		Rev.Seq	5	9.53	0.15	76.41	5	0	143/149
		Random	1	5.22	0.36	26.60	8	0	147/149
		Random	3	32.42	0.18	85.65	5	0	143/149
		Random	5	23.67	0.15	58.26	2	0	145/149
PNE	*FE-P*	-	-	7.25	0.12	328.23	–	–	122/149

Energy Game

	Algorithm	ES	k	Time (s) EQ	Time (s) NO	Time (s) All	Wins EQ	Wins NO	Solved
	FE	-	-	29.08	0.12	120.21	6	82	140/149
MNE	*InnerApp*	Seq	1	6.65	0.35	51.33	3	0	145/149
		Seq	3	17.76	0.18	55.82	5	0	145/149
		Seq	5	6.40	0.15	51.08	3	0	145/149
		Rev.Seq	1	7.97	0.36	3.73	26	0	149/149
		Rev.Seq	3	11.29	0.18	53.12	4	0	145/149
		Rev.Seq	5	9.53	0.15	76.41	5	0	143/149
		Random	1	5.22	0.36	26.60	8	0	147/149
		Random	3	32.42	0.18	85.65	5	0	143/149
		Random	5	23.67	0.15	58.26	2	0	145/149
PNE	*FE-P*	-	-	7.25	0.12	328.23	–	–	122/149

Energy Game

	Algorithm	ES	k	Time (s) EQ	NO	All	Wins EQ	NO	Solved
	FE	-	-	29.08	0.12	120.21	6	82	140/149
MNE	*InnerApp*	Seq	1	6.65	0.35	51.33	3	0	145/149
		Seq	3	17.76	0.18	55.82	5	0	145/149
		Seq	5	6.40	0.15	51.08	3	0	145/149
		Rev.Seq	1	7.97	0.36	3.73	26	0	149/149
		Rev.Seq	3	11.29	0.18	53.12	4	0	145/149
		Rev.Seq	5	9.53	0.15	76.41	5	0	143/149
		Random	1	5.22	0.36	26.60	8	0	147/149
		Random	3	32.42	0.18	85.65	5	0	143/149
		Random	5	23.67	0.15	58.26	2	0	145/149
PNE	*FE-P*	-	-	7.25	0.12	328.23	–	–	122/149

Energy Game

Large

	Algorithm	ES	k	Time (s) EQ	NO	All	Wins EQ	NO	Solved
	FE	-	-	260.29	1.12	1174.32	0	2	20/50
MNE	*InnerApp*	Seq	1	39.26	9.64	672.24	1	0	32/50
		Seq	3	62.66	3.88	616.25	1	0	34/50
		Seq	5	24.03	2.83	733.97	1	0	30/50
		Rev.Seq	1	171.47	9.66	262.74	27	0	47/50
		Rev.Seq	3	13.85	3.86	585.27	4	0	34/50
		Rev.Seq	5	78.57	2.83	798.90	6	0	29/50
		Random	1	34.65	9.65	497.06	0	0	37/50
		Random	3	123.02	3.87	588.03	2	0	36/50
		Random	5	39.18	2.86	711.77	4	0	41/50
PNE	*FE-P*	-	-	7.36	1.12	1441.95	–	–	10/50

Energy Game

	Algorithm	ES	k	Time (s) EQ	NO	All	Wins EQ	NO	Solved
	FE	-	-	260.29	1.12	1174.32	0	2	20/50
MNE	*InnerApp*	Seq	1	39.26	9.64	672.24	1	0	32/50
		Seq	3	62.66	3.88	616.25	1	0	34/50
		Seq	5	24.03	2.83	733.97	1	0	30/50
		Rev.Seq	1	171.47	9.66	262.74	27	0	47/50
		Rev.Seq	3	13.85	3.86	585.27	4	0	34/50
		Rev.Seq	5	78.57	2.83	798.90	6	0	29/50
		Random	1	34.65	9.65	497.06	0	0	37/50
		Random	3	123.02	3.87	588.03	2	0	36/50
		Random	5	39.18	2.86	711.77	4	0	41/50
PNE	*FE-P*	-	-	7.36	1.12	1441.95	–	–	10/50

NASPs

Algo	Inst	#	GT (s)	# NASH_EQ	GT (s)	# NO_EQ	GT (s)	#N ALL	#NI	#TL
Inn-S-1	**B**	50	6.22	49	69.76	1	6.56	50	0	0
Inn-S-3	**B**	50	4.94	49	23.96	1	5.12	50	0	0
Out-HB	**B**	50	7.47	46	29.37	1	7.71	47	3	0
Out-DB	**B**	50	9.45	46	11.81	1	9.50	47	3	0
Inn-S-1	**H7**	50	-	0	-	0	300.00	46	4	46
Inn-S-3	**H7**	50	-	0	-	0	-	0	50	0
Out-HB	**H7**	50	53.79	41	-	0	73.45	50	0	9
Out-DB	**H7**	50	52.58	35	-	0	88.92	50	0	15