Pure Nash Equilibria
complexity versus succinctness

Maria Serna
Computer Science Department
Universitat Politècnica de Catalunya
Barcelona

September 9th, 2015

CS and Game Theory

Christos Papadimitriou (STOC 2001)

"The internet is unique among all the computer systems in that it is build, operated and used by multitude of diverse economic interests, in varing relationships of collaboration and competition with each other. This suggest that the mathematical tools and insights most appropriate for understanding the Internet may come from the fusion of algorithmic ideas with concepts and techniques from Mathematical Economics and Game Theory."

http://www.cs.berkeley.edu/~christos/games/cs294.html

Game Theory

- Game theory studies situations where players choose different actions in an attempt to maximize their returns.
- It provides a formal modeling approach to social situations in which decision makers interact.
- Extends the simpler optimization approach by adding other solution concepts.

Focus: computational complexity view

Focus: computational complexity view

- A huge set of computational problems to study.

Focus: computational complexity view

- A huge set of computational problems to study.
- Whose inputs are games.

Focus: computational complexity view

- A huge set of computational problems to study.
- Whose inputs are games.
- Aiming at formalizing the concept:
  games with polynomial utilities
- and studying the computational complexity of problems related to games.

Game types

- Non-cooperative games
  - strategic games
  - extensive games
  - repeated games
  - Bayesian games
- Coalitional games
  - simple games
  - weighted games
  - ...

Strategic game

A strategic game Γ (with ordinal preferences) consists of:

- A finite set $N = \{1, \ldots, n\}$ of players.
- For each player $i \in N$, a nonempty set of actions A_i.
- Each player chooses his action once. Players choose actions simultaneously.
  No player is informed, when he chooses his action, of the actions chosen by others.
- For each player $i \in N$, a preference relation (a complete, transitive, reflexive binary relation) $\preceq_i$ over the set $A = A_1 \times \cdots \times A_n$.

It is frequent to specify the players' preferences by giving utility functions $u_i(a_1, \ldots a_n)$. Also called pay-off functions.

Strategies: Notation

A strategy of player $i \in N$ in a strategic game Γ is an action $a_i \in A_i$.
A strategy profile $s = (s_1, \ldots, s_n)$ consists of a strategy for each player.

For each $s = (s_1, \ldots s_n)$ and $s_i' \in A_i$ we denote by

$$(s_{-i}, s_i') = (s_1, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n)$$

$s_{-i} = (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$
is not an strategy profile but can be seen as an strategy for the other players.

Example: Prisoner's Dilemma

The story

- Two suspects in a major crime are held in separate cells.
- Evidence to convict each of them of a minor crime.
- No evidence to convict either of them of a major crime unless one of them finks.

The penalties

- If both stay quiet, be convicted for a minor offense (one year prison).
- If only one finks, he will be freed (and used as a witness) and the other will be convicted for a major offense (four years in prison).
- If both fink, each one will be convicted for a major offense with a reward for cooperation (three years each).

Game representation

- Players $N = \{\text{Suspect } 1, \text{Suspect } 2\}$.
- Actions $A_1 = A_2 = \{\text{Quiet}, \text{Fink}\}$.
- Action profiles $A = A_1 \times A_2 = \{(\text{Quiet}, \text{Quiet}), (\text{Quiet}, \text{Fink}), (\text{Fink}, \text{Quiet}), (\text{Fink}, \text{Fink})\}$
- Preferences
  - Player 1

    $$(\text{Fink}, \text{Quiet}) \preceq_1 (\text{Quiet}, \text{Quiet}) \preceq_1 (\text{Fink}, \text{Fink}) \preceq_1 (\text{Quiet}, \text{Fink})$$

  - Player 2

    $$(\text{Quiet}, \text{Fink}), \preceq_2 (\text{Quiet}, \text{Quiet}) \preceq_2 (\text{Fink}, \text{Fink}) \preceq_2 (\text{Fink}, \text{Quiet})$$

Bi-matrix form

- Utilities

$$
\begin{aligned}
u_1(\mathrm{Fink}, \mathrm{Quiet}) = 3,\ & u_1(\mathrm{Quiet}, \mathrm{Quiet}) = 2 \\
u_1(\mathrm{Fink}, \mathrm{Fink}) = 1,\ & u_1(\mathrm{Quiet}, \mathrm{Fink}) = 0 \\
u_2(\mathrm{Quiet}, \mathrm{Fink}) = 3,\ & u_2(\mathrm{Quiet},\ \mathrm{Quiet}) = 2, \\
u_2(\mathrm{Fink}, \mathrm{Fink}) = 1,\ & u_2(\mathrm{Fink}, \mathrm{Quiet}) = 0
\end{aligned}
$$

Bi-matrix form

- Utilities

$$
\begin{aligned}
u_1(\text{Fink}, \text{Quiet}) = 3,\ & u_1(\text{Quiet}, \text{Quiet}) = 2 \\
u_1(\text{Fink}, \text{Fink}) = 1,\ & u_1(\text{Quiet}, \text{Fink}) = 0 \\
u_2(\text{Quiet}, \text{Fink}) = 3,\ & u_2(\text{Quiet},\ \text{Quiet}) = 2, \\
u_2(\text{Fink}, \text{Fink}) = 1,\ & u_2(\text{Fink}, \text{Quiet}) = 0
\end{aligned}
$$

We can represent pay-offs in a compact way on a bi-matrix:

	Quiet	Fink
Quiet	2,2	0,3
Fink	3,0	1,1

Example: Matching Pennies

- Two people choose, simultaneously, whether to show the head or tail of a coin.
- If they show same side, person 2 pays person 1, otherwise person 1 pays person 2.
- Payoff are equal to the amounts of money involved.

	Head	Tail
Head	1,-1	-1,1
Tail	-1,1	1,-1

This is an example of a zero-sum game

Example: Sending from s to t

Example: Sending from s to t

- We have a graph $G = (V, E)$ and two vertices $s, t \in V$.
- There is one player for each vertex $v \in V$, $v \neq t$.
- The set of actions for player u is $N_G(u)$.

Example: Sending from s to t

- We have a graph $G = (V, E)$ and two vertices $s, t \in V$.
- There is one player for each vertex $v \in V$, $v \neq t$.
- The set of actions for player u is $N_G(u)$.
- A strategy profile is formed by $n - 1$ vertices $(v_1, \ldots, v_{n-1})$.
- Pay-offs are defined as follows:

Example: Sending from s to t

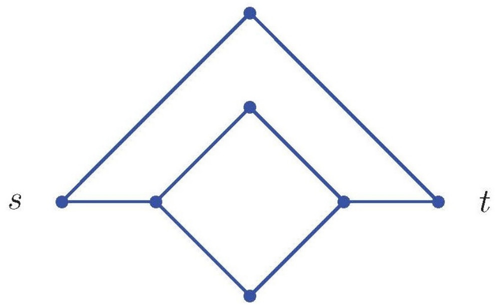
- We have a graph $G = (V, E)$ and two vertices $s, t \in V$.
- There is one player for each vertex $v \in V$, $v \neq t$.
- The set of actions for player u is $N_G(u)$.
- A strategy profile is formed by $n - 1$ vertices $(v_1, \ldots, v_{n-1})$.
- Pay-offs are defined as follows:
  player u gets 1 if the shortest path joining s to t in the digraph induced by $v_1, \ldots, v_{n-1}$ contains (u, v_u), otherwise gets 0.

Example: Sending from s to t

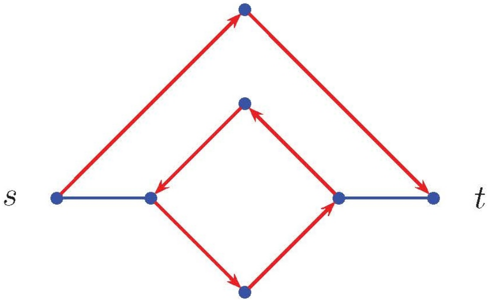
- We have a graph $G = (V, E)$ and two vertices $s, t \in V$.
- There is one player for each vertex $v \in V$, $v \neq t$.
- The set of actions for player u is $N_G(u)$.
- A strategy profile is formed by $n - 1$ vertices $(v_1, \dots, v_{n-1})$.
- Pay-offs are defined as follows:
  player u gets 1 if the shortest path joining s to t in the digraph induced by $v_1, \dots, v_{n-1}$ contains (u, v_u), otherwise gets 0.

  This is not the definition of one game but of a family of games.

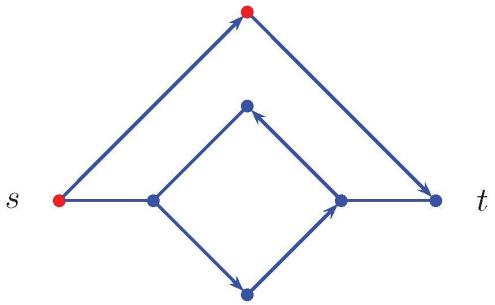
Sending from s to t: example of game



Sending from s to t: strategies



Sending from s to t: pay-offs



Red nodes get pay-off 1, blue nodes get pay-off 0.

Congestion games

Congestion games

A congestion game

- is defined on a finite set E of resources and
- has n players
- using a delay function d mapping $E \times \mathbb{N}$ to the integers.
- The actions for each player are subsets of E.
- The pay-off functions are the following:

$$u_i(a_1, \ldots, a_n) = -\left(\sum_{e \in a_i} d(e, f(a_1, \ldots, a_n, e))\right)$$

being $f(a_1, \ldots, a_n, e) = |\{i \mid e \in a_i\}|$.

Objective

Analyze from a computational complexity point of view problems on games with polynomial computable utilities.

Objective

Analyze from a computational complexity point of view problems on games with polynomial computable utilities.

Are you familiar with

Objective

Analyze from a computational complexity point of view problems on games with polynomial computable utilities.

Are you familiar with
Turing Machines?

Objective

Analyze from a computational complexity point of view problems on games with polynomial computable utilities.

Are you familiar with
Turing Machines?
Complexity classes P, NP, coNP?

Objective

Analyze from a computational complexity point of view problems on games with polynomial computable utilities.

Are you familiar with
Turing Machines?
Complexity classes P, NP, coNP?
Reductions?

Solution concepts

- Pure Nash equilibria
- Nash equilibria
- Strong Nash equilibria
- Correlated equilibria
- Dominant strategies
- ...

Pure Nash equilibrium

A pure Nash equilibrium is an strategy profile $a^* = (a_1^*, \ldots, a_n^*)$ no player i can do better choosing an action different from a_i^*, given that every other player j adheres to a_j^*:

Pure Nash equilibrium

A pure Nash equilibrium is an strategy profile $a^* = (a_1^*, \ldots, a_n^*)$ no player i can do better choosing an action different from a_i^*, given that every other player j adheres to a_j^*:

for every player i and for every action $a_i \in A_i$ it holds

$$u_i(a_{-i}^*, a_i^*) \geqslant u_i(a_{-i}^*, a_i).$$

Pure Nash equilibria, examples

	Quiet	Fink
Quiet	2,2	0,3
Fink	3,0	1,1

	Bach	Stravinsky
Bach	2,1	0,0
Stravinsky	0,0	1,2

	Stag	Hare
Stag	2,2	0,1
Hare	1,0	1,1

	Head	Tail
Head	1,-1	-1,1
Tail	-1,1	1,-1

Pure Nash equilibria, examples

	Quiet	Fink
Quiet	2,2	0,3
Fink	3,0	1,1

	Bach	Stravinsky
Bach	2,1	0,0
Stravinsky	0,0	1,2

	Stag	Hare
Stag	2,2	0,1
Hare	1,0	1,1

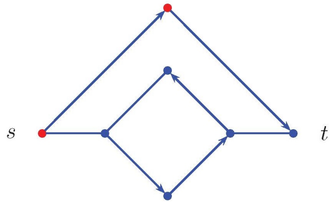
	Head	Tail
Head	1,-1	-1,1
Tail	-1,1	1,-1

- Prisoner's Dilemma, $(Fink, Fink)$.
- Bach or Stravinsky, $(Bach, Bach)$, $(Stravinsky, Stravinsky)$.
- Stag Hunt, $(Stag, Stag)$, $(Hunt, Hunt)$.
- Matching Pennies, none.

Example: Sending from s to t

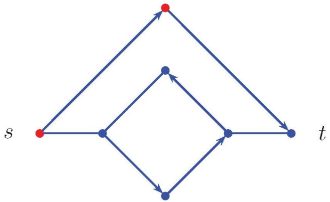
- We have a graph $G = (V, E)$ and two vertices $s, t \in V$.
- There is one player for each vertex $v \in V$, $v \neq t$.
- The set of actions for player u is $N_G(u)$.
- A strategy profile is a set of vertices $(v_1, \ldots, v_{n-1})$.
- Pay-offs are defined as follows:
  player u gets 1 if the shortest path joining s to t in the digraph induced by $v_1, \ldots, v_{n-1}$ contains (u, v_u), otherwise gets 0.

Sending from s to t: PNE?



Red nodes get pay-off 1, blue nodes get pay-off 0.

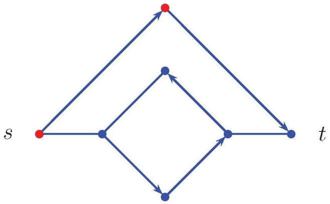
Sending from s to t: PNE?



Red nodes get pay-off 1, blue nodes get pay-off 0.

Is this strategy profile a PNE?

Sending from s to t: PNE?

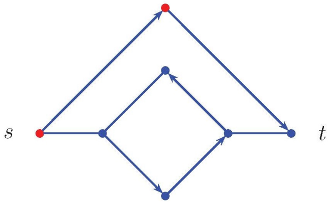


Red nodes get pay-off 1, blue nodes get pay-off 0.

Is this strategy profile a PNE?

Does the game have a PNE?

Sending from s to t: PNE?



Red nodes get pay-off 1, blue nodes get pay-off 0.

Is this strategy profile a PNE?

Does the game have a PNE?

Does the game have a PNE in which s gets 1?

Natural problems related to PNE

Is Nash (IsN)
Given a game Γ and a strategy profile a, decide whether a is a Nash equilibrium of Γ.

Exists Pure Nash (EPN)
Given a strategic game Γ, decide whether Γ has a Pure Nash equilibrium.

Pure Nash with Guarantees (PNGrant)
Given a strategic game Γ and a value v, decide whetherthere is a pure Nash equilibrium in which the first player gets payoff v or higher.

How to represent a game?

- We are interested in fixing the representation of a game as an input to a program.
- It is natural to consider different levels of succinctness.
- In the most generic model some components of the game have to be represented by a TM, for example the utility functions.

TMs in game representations

- All the TMs appearing in the description of games are deterministic.

TMs in game representations

- All the TMs appearing in the description of games are deterministic.
- The TMs will work for a limited number of timesteps (t). Which forms part of the input in unary ($\langle M, 1^t \rangle$).

TMs in game representations

- All the TMs appearing in the description of games are deterministic.
- The TMs will work for a limited number of timesteps (t). Which forms part of the input in unary ($\langle M, 1^t \rangle$).
- Convention: there is a pre-fixed interpretation of the contents of the output tape of a TM so that, both when the machine stops or when the machine is stopped, it always computes a rational value.

TMs in game representations

- All the TMs appearing in the description of games are deterministic.
- The TMs will work for a limited number of timesteps (t). Which forms part of the input in unary ($\langle M, 1^t \rangle$).
- Convention: there is a pre-fixed interpretation of the contents of the output tape of a TM so that, both when the machine stops or when the machine is stopped, it always computes a rational value.

We only consider rational valued utility functions
The convention guarantees a correct and unique game definition from its description

Explicit form

Strategic games in explicit form.

- *A game is given by a tuple*

$$\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle.$$

- *It has n players,*
- *For each player i, A_i is given explicitly by listing its elements.*
- *T is a table with an entry for each strategy profile a and player i.*
- *So, $u_i(a) = T(a, i)$.*

General form

Strategic games in general form.

- *A game is given by a tuple*

$$\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle.$$

- *It has n players,*
- *For each player i, A_i is given explicitly by listing its elements.*
- *The description of their pay-off is given by $\langle M, 1^t \rangle$.*
- *So, for each strategy profile a and player i, $u_i(a) = M(a, i)$ stopping after t steps.*

Implicit form

Strategic games in implicit form.

- *A game is given by a tuple*

$$\Gamma = \langle 1^n, 1^m, M, 1^t \rangle.$$

- *It has n players,*
- *For each player i,* $A_i = \Sigma^m$
- *The description of their pay-off is given by* $\langle M, 1^t \rangle$.
- *So, for each strategy profile a and player i,* $u_i(a) = M(a, i)$ *stopping after t steps.*

Forms of representation

Strategic games in explicit form. *A game is descrbed by a tuple* $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$.

Strategic games in general form. *A game is described by a tuple* $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$.

Strategic games in implicit form. *A game is described by a tuple* $\Gamma = \langle 1^n, 1^m, M, 1^t \rangle$.

What is the most suitable level of succinctness?

- Prisoners' dilemma?

What is the most suitable level of succinctness?

- Prisoners' dilemma?
  Explicit

What is the most suitable level of succinctness?

- Prisoners' dilemma?
  Explicit
- Sending from s to t?

What is the most suitable level of succinctness?

- Prisoners' dilemma?
  Explicit
- Sending from s to t?
  General

What is the most suitable level of succinctness?

- Prisoners' dilemma?
  Explicit
- Sending from s to t?
  General
- Congestion games?

What is the most suitable level of succinctness?

- Prisoners' dilemma?
  Explicit
- Sending from s to t?
  General
- Congestion games?
  Implicit

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \; \forall {a'}_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N\ \forall {a'}_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

Solving the IsPN

IsPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \; \forall a'_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is

Solving the IsPN

IsPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N\ \forall {a'}_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is polynomial.

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \ \forall a'_i \in A_i \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \ \forall a'_i \in A_i \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is polynomial.

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \; \forall a'_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t\rangle$ the cost is

Solving the IsPN

IsPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \; \forall {a'}_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t\rangle$ the cost is exponential.

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \ \forall a'_i \in A_i \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t\rangle$ the cost is exponential.
  A better classification?

Solving the IsPN

IsPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \ \forall a'_i \in A_i \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t \rangle$ the cost is exponential.
  A better classification?
  The condition $u_i(s) \geqslant u_i(s_{-i}, a_i)$ can be checked in polynomial time given Γ, s, and a_i.

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \ \forall a'_i \in A_i \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t \rangle$ the cost is exponential.
  A better classification?
  The condition $u_i(s) \geqslant u_i(s_{-i}, a_i)$ can be checked in polynomial time given Γ, s, and a_i.
  Thus the problem is in

Solving the IsPN

ISPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \; \forall a'_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t \rangle$ the cost is exponential.
  A better classification?
  The condition $u_i(s) \geqslant u_i(s_{-i}, a_i)$ can be checked in polynomial time given Γ, s, and a_i.
  Thus the problem is in coNP.

Solving the IsPN

IsPN Given a game Γ and a strategy profile s, is s is a PNE?.

$$\forall i \in N \ \forall {a'}_i \in A_i \ \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combinations

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t\rangle$ the cost is exponential.
  A better classification?
  The condition $u_i(s) \geqslant u_i(s_{-i}, a_i)$ can be checked in polynomial time given Γ, s, and a_i.
  Thus the problem is in coNP.
  Is this classification tight?

IsPN implicit form: Hardness

A coNP complete problem?

IsPN implicit form: Hardness

A coNP complete problem?

SAT: Given a boolean formula F in CNF form, determine whether F is satisfiable.

Is an NP complete problem.

IsPN implicit form: Hardness

A coNP complete problem?

SAT: Given a boolean formula F in CNF form, determine whether F is satisfiable.

Is an NP complete problem. So, its complement is coNP-complete.

IsPN implicit form: Hardness

A coNP complete problem?

SAT: Given a boolean formula F in CNF form, determine whether F is satisfiable.

Is an NP complete problem. So, its complement is coNP-complete.

We have to associate to F a game Γ and a strategy profile s so that:

- F is not satisfiable iff s is a PNE of Γ
- and show that a description of Γ in implicit form and of s can be obtained in time polynomial in $|F|$.

IsPN implicit form: Hardness

Given a CNF formula F on n variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{n+1}$
- $u_1(0x) = 0$, for any $x \in \{0,1\}^{n+1}$
- $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{n+1}$

IsPN implicit form: Hardness

Given a CNF formula F on n variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{n+1}$
- $u_1(0x) = 0$, for any $x \in \{0,1\}^{n+1}$
- $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{n+1}$

Consider the strategy $a_1 = 0^{n+1}$.

IsPN implicit form: Hardness

Given a CNF formula F on n variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{n+1}$
- $u_1(0x) = 0$, for any $x \in \{0,1\}^{n+1}$
- $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{n+1}$

Consider the strategy $a_1 = 0^{n+1}$.

a_1 is a PNE iff F is unsatisfiable

IsPN implicit form: Hardness

Given a CNF formula F on n variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{n+1}$
- $u_1(0x) = 0$, for any $x \in \{0,1\}^{n+1}$
- $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{n+1}$

Consider the strategy $a_1 = 0^{n+1}$.

a_1 is a PNE iff F is unsatisfiable

Thus $\Gamma(F), 0^{n+1}$ verify the first requirement.

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

An implicit form representation of $\Gamma(F)$ as $\langle 1^n, 1^m, M, 1^t \rangle$?

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

An implicit form representation of $\Gamma(F)$ as $\langle 1^n, 1^m, M, 1^t \rangle$?

- $n = 1$, $m = k + 1$

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

An implicit form representation of $\Gamma(F)$ as $\langle 1^n, 1^m, M, 1^t \rangle$?

- $n = 1$, $m = k + 1$
- M:

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

An implicit form representation of $\Gamma(F)$ as $\langle 1^n, 1^m, M, 1^t \rangle$?

- $n = 1$, $m = k + 1$
- M: There is a TM M' that given a CNF formula F and an truth assignment x computes $F(x)$ in linear time $O(|F|)$.

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

An implicit form representation of $\Gamma(F)$ as $\langle 1^n, 1^m, M, 1^t \rangle$?

- $n = 1$, $m = k + 1$
- M: There is a TM M' that given a CNF formula F and an truth assignment x computes $F(x)$ in linear time $O(|F|)$. M on input ax checks outputs 0 if $a = 0$ otherwise transfer the control to M' after writing in the input tape F and x.
- $t = |F|^2$.

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

An implicit form representation of $\Gamma(F)$ as $\langle 1^n, 1^m, M, 1^t \rangle$?

- $n = 1$, $m = k + 1$
- M: There is a TM M' that given a CNF formula F and an truth assignment x computes $F(x)$ in linear time $O(|F|)$. M on input ax checks outputs 0 if $a = 0$ otherwise transfer the control to M' after writing in the input tape F and x.
- $t = |F|^2$.

The time required to obtain $\langle 1^n, 1^m, M, 1^t \rangle$ given F is

IsPN implicit form: Hardness

Given a boolean formula F on k variables consider the game $\Gamma(F)$ which:

- Has one player and $A_1 = \{0,1\}^{k+1}$
- $u_1(0x) = 0$, $u_1(1x) = F(x)$, for any $x \in \{0,1\}^{k+1}$

An implicit form representation of $\Gamma(F)$ as $\langle 1^n, 1^m, M, 1^t \rangle$?

- $n = 1$, $m = k + 1$
- M: There is a TM M' that given a CNF formula F and an truth assignment x computes $F(x)$ in linear time $O(|F|)$. M on input ax checks outputs 0 if $a = 0$ otherwise transfer the control to M' after writing in the input tape F and x.
- $t = |F|^2$.

The time required to obtain $\langle 1^n, 1^m, M, 1^t \rangle$ given F is polynomial in $|F|$.

IsPN implicit form

Theorem

The IsPN problem for strategic games in implicit form is coNP-complete.

Solving the EPN

EPN Given a game Γ does it have a PNE?.

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s \; \forall i \in N \; \forall {a'}_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s\ \forall i \in N\ \forall a'_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s\ \forall i \in N\ \forall a'_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s \; \forall i \in N \; \forall a'_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s\ \forall i \in N\ \forall a'_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \dots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \dots, A_n, M, 1^t\rangle$ the cost is

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s\ \forall i \in N\ \forall a'_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t\rangle$ the cost is exponential.

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s \; \forall i \in N \; \forall a'_i \in A_i \;\; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is exponential. So, in NP.

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s\ \forall i \in N\ \forall a'_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t\rangle$ the cost is exponential.
  So, in NP.
  In the case that n is constant, in P.

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s \ \forall i \in N \ \forall a'_i \in A_i \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is exponential.
  So, in NP.
  In the case that n is constant, in P.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t \rangle$ the cost is

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s \ \forall i \in N \ \forall a'_i \in A_i \ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is exponential.
  So, in NP.
  In the case that n is constant, in P.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t \rangle$ the cost is exponential.

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s\ \forall i \in N\ \forall a'_i \in A_i\ u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \dots, A_m, T\rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \dots, A_n, M, 1^t\rangle$ the cost is exponential.
  So, in NP.
  In the case that n is constant, in P.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t\rangle$ the cost is exponential.
  A better classification?

Solving the EPN

EPN Given a game Γ does it have a PNE?.

$$\exists s \; \forall i \in N \; \forall a'_i \in A_i \; u_i(s) \geqslant u_i(s_{-i}, a_i)$$

Algorithm: Brute force, try all combination

- Given $\Gamma = \langle 1^n, A_1, \ldots, A_m, T \rangle$ the cost is polynomial.
- Given $\Gamma = \langle 1^n, A_1, \ldots, A_n, M, 1^t \rangle$ the cost is exponential.
  So, in NP.
  In the case that n is constant, in P.
- Given $\Gamma = \langle 1^n, 1^m, M, 1^t \rangle$ the cost is exponential.
  A better classification? in Σ_2^p.

EPN: general form

Theorem

The EPN problem for strategic games in general form is NP-complete.

We provide a reduction from SAT. Let F be a CNF formula.

- $F \to \Gamma(F) = \langle 1^n, \{0,1\} \dots \{0,1\}, M^F, 1^{(n+|F|)^2} \rangle$ where
- n is the number of variables in F and
- M^F is a $\textsc{TM}$ that on input (a, i), evaluates F on assignment a and afterwards it implements the utility function of the i-th player. According to the following definition:

EPN: general form

$$u_1(a) = \begin{cases} 5 & \text{if } F(a) = 1, \\ 4 & \text{if } F(a) = 0 \wedge a_1 = 0 \wedge a_2 = 1, \\ 3 & \text{if } F(a) = 0 \wedge a_1 = 1 \wedge a_2 = 1, \\ 2 & \text{if } F(a) = 0 \wedge a_1 = 1 \wedge a_2 = 0, \\ 1 & \text{if } F(a) = 0 \wedge a_1 = 0 \wedge a_2 = 0, \end{cases}$$

$$u_2(a) = \begin{cases} 5 & \text{if } F(a) = 1, \\ 4 & \text{if } F(a) = 0 \wedge a_1 = 0 \wedge a_2 = 0, \\ 3 & \text{if } F(a) = 0 \wedge a_1 = 0 \wedge a_2 = 1, \\ 2 & \text{if } F(a) = 0 \wedge a_1 = 1 \wedge a_2 = 1, \\ 1 & \text{if } F(a) = 0 \wedge a_1 = 1 \wedge a_2 = 0. \end{cases}$$

And, for any $j > 2$

$$u_j(a) = \begin{cases} 5 & \text{if } F(a) = 1, \\ 1 & \text{otherwise.} \end{cases}$$

Reduction correctness

We have that

- Given a description of F, $\Gamma(F)$ is computable in polynomial time.

Reduction correctness

We have that

- Given a description of F, $\Gamma(F)$ is computable in polynomial time.
  Similar arguments as before.

Reduction correctness

We have that

- Given a description of F, $\Gamma(F)$ is computable in polynomial time.
  Similar arguments as before.
- F is satisfiable iff $\Gamma(F)$ has a PNE?

Reduction trick

Look at the two player strategic game that can be played by the first and second players:

	0	1
0	1,4	4,3
1	2,1	3,2

PNE?

Reduction trick

Look at the two player strategic game that can be played by the first and second players:

	0	1
0	1,4	4,3
1	2,1	3,2

PNE?

None

Reduction correctness

- F is a yes instance of SAT.

Reduction correctness

- F is a yes instance of SAT.
  There is a satisfying assignmet x. So $u_i(x) = 5$, for any i.
  Such a strategy profile is a PNE.

Reduction correctness

- F is a yes instance of SAT.
  There is a satisfying assignmet x. So $u_i(x) = 5$, for any i.
  Such a strategy profile is a PNE.
- F is a no instance of SAT.

Reduction correctness

- F is a yes instance of SAT.
  There is a satisfying assignmet x. So $u_i(x) = 5$, for any i.
  Such a strategy profile is a PNE.
- F is a no instance of SAT.
  For any strategy profile the payoff of players $j > 2$ is always 1.
  So they cannot change strategy and improve payoff.
  However, players 1 and 2 are engaged in a game with no PNE so one of them can change strategy and increase its payoff.
  Therefore $\Gamma(F)$ has no PNE

Σ_2^p definition and a complete problem

Let $L \subseteq \Sigma^*$ be a language.
$L \in \Sigma_2^p$ if and only if there is a polynomially decidable relation R and a polynomial p such that

$$L = \{x \mid \exists z |z| \leqslant p(|x|) \forall y |y| \leqslant p(|x|) \langle x, y, z \rangle \in R\}.$$

Σ_2^p definition and a complete problem

Let $L \subseteq \Sigma^*$ be a language.
$L \in \Sigma_2^p$ if and only if there is a polynomially decidable relation R and a polynomial p such that

$$L = \{x \mid \exists z |z| \leqslant p(|x|) \forall y |y| \leqslant p(|x|) \langle x, y, z \rangle \in R\}.$$

Q2SAT
Given $\Phi = \exists \alpha_1, \ldots, \alpha_{n_1} \forall \beta_1, \ldots \beta_{n_2} F$ where F is a Boolean formula over the boolean variables $\alpha_1, \ldots, \alpha_{n_1}, \beta_1, \ldots, \beta_{n_2}$, decide whether Φ is valid.

Q2SAT is Σ_2^p-complete.

EPN: implicit form

Theorem

The EPN problem for strategic games in implicit form is Σ_2^p-complete.

Lets provide a reduction from $\mathrm{Q2SAT}$.

EPN implicit form:reduction

For each $\Phi = \exists \alpha_1, \ldots, \alpha_{n_1} \forall \beta_1, \ldots \beta_{n_2} F$
we define a game $\Gamma(\Phi)$ as follows.
There are four players:

EPN implicit form:reduction

For each $\Phi = \exists \alpha_1, \ldots, \alpha_{n_1} \forall \beta_1, \ldots \beta_{n_2} F$
we define a game $\Gamma(\Phi)$ as follows.
There are four players:

- Player 1, the *existential player*, assigns truth values to the boolean variables $\alpha_1, \ldots, \alpha_{n_1}$. Their set of actions is $A_1 = \{0,1\}^{n_1}$ and $a_1 = (\alpha_1, \ldots \alpha_{n_1}) \in A_1$.

EPN implicit form:reduction

For each $\Phi = \exists \alpha_1, \ldots, \alpha_{n_1} \forall \beta_1, \ldots \beta_{n_2} F$
we define a game $\Gamma(\Phi)$ as follows.
There are four players:

- Player 1, the *existential player*, assigns truth values to the boolean variables $\alpha_1, \ldots, \alpha_{n_1}$. Their set of actions is $A_1 = \{0, 1\}^{n_1}$ and $a_1 = (\alpha_1, \ldots \alpha_{n_1}) \in A_1$.
- Player 2, the *universal player*, assigns truth values to the boolean variables $\beta_1, \ldots, \beta_{n_2}$ and then their set of actions is $A_2 = \{0, 1\}^{n_2}$ and $a_2 = (\beta_1, \ldots, \beta_{n_2}) \in A_2$.

EPN implicit form:reduction

For each $\Phi = \exists \alpha_1, \ldots, \alpha_{n_1} \forall \beta_1, \ldots \beta_{n_2} F$
we define a game $\Gamma(\Phi)$ as follows.
There are four players:

- Player 1, the *existential player*, assigns truth values to the boolean variables $\alpha_1, \ldots, \alpha_{n_1}$. Their set of actions is $A_1 = \{0,1\}^{n_1}$ and $a_1 = (\alpha_1, \ldots \alpha_{n_1}) \in A_1$.
- Player 2, the *universal player*, assigns truth values to the boolean variables $\beta_1, \ldots, \beta_{n_2}$ and then their set of actions is $A_2 = \{0,1\}^{n_2}$ and $a_2 = (\beta_1, \ldots, \beta_{n_2}) \in A_2$.
- Players 3 and 4 avoid entering into a Nash equilibrium when the actions played by players 1 and 2 do not satisfy F. Their set of actions are $A_3 = A_4 = \{0,1\}$.

Let us denote by $F(a_1, a_2)$ the truth value of F under the assignment given by a_1 and a_2.

$$u_1(a_1, a_2, a_3, a_4) = \begin{cases} 1 & \text{if } F(a_1, a_2) = 1, \\ 0 & \text{otherwise.} \end{cases}$$

$$u_2(a_1, a_2, a_3, a_4) = \begin{cases} 1 & \text{if } F(a_1, a_2) = 0, \\ 0 & \text{otherwise.} \end{cases}$$

$$u_3(a_1, a_2, a_3, a_4) = \begin{cases} 5 & \text{if } F(a_1, a_2) = 1, \\ 4 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 0 \wedge a_4 = 1, \\ 3 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 1 \wedge a_4 = 1, \\ 2 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 1 \wedge a_4 = 0, \\ 1 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 0 \wedge a_4 = 0. \end{cases}$$

$$u_4(a_1, a_2, a_3, a_4) = \begin{cases} 5 & \text{if } F(a_1, a_2) = 1, \\ 3 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 0 \wedge a_4 = 1, \\ 2 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 1 \wedge a_4 = 1, \\ 1 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 1 \wedge a_4 = 0, \\ 4 & \text{if } F(a_1, a_2) = 0 \wedge a_3 = 0 \wedge a_4 = 0. \end{cases}$$

EPN implicit form:reduction correcteness

- Let us assume that $\Phi = \exists\alpha_1, \dots, \alpha_n \forall\beta_1, \dots, \beta_m F$, where F is a Boolean formula over the boolean variables $\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_m$, is true.
- Then there exists $\alpha \in \{0,1\}^n$ such that for all $\beta \in \{0,1\}^m$, $F(\alpha, \beta) = 1$.
- This means that if player 1 plays action α, for each $\beta \in \{0,1\}^m$, $a_3, a_4 \in \{0,1\}$, no player has incentive to change strategy.

EPN implicit form:reduction correcteness

- Let us assume that Φ is not valid.
- It means that for any $\alpha \in \{0,1\}^n$ there exists $\beta \in \{0,1\}^m$ such that $F(\alpha, \beta) = 0$.
- Let (α, β, a, b) be a strategy profile. We have two cases.

EPN implicit form:reduction correcteness

- Let us assume that Φ is not valid.
- It means that for any $\alpha \in \{0,1\}^n$ there exists $\beta \in \{0,1\}^m$ such that $F(\alpha, \beta) = 0$.
- Let (α, β, a, b) be a strategy profile. We have two cases.
- Case 1: $F(\alpha, \beta) = 0$, in this case players 3 an 4 engage in a no PNE game.

EPN implicit form:reduction correcteness

- Let us assume that Φ is not valid.
- It means that for any $\alpha \in \{0,1\}^n$ there exists $\beta \in \{0,1\}^m$ such that $F(\alpha, \beta) = 0$.
- Let (α, β, a, b) be a strategy profile. We have two cases.
- Case 1: $F(\alpha, \beta) = 0$, in this case players 3 an 4 engage in a no PNE game.
- Case 2: $F(\alpha, \beta) = 1$, since Φ is not valid, there exists $\beta' \in \{0,1\}^m$ such that $F(\alpha, \beta') = 0$. Therefore player 2 has an incentive to change strategy β by β'.

EPN implicit form:reduction correcteness

- Let us assume that Φ is not valid.
- It means that for any $\alpha \in \{0,1\}^n$ there exists $\beta \in \{0,1\}^m$ such that $F(\alpha, \beta) = 0$.
- Let (α, β, a, b) be a strategy profile. We have two cases.
- Case 1: $F(\alpha, \beta) = 0$, in this case players 3 an 4 engage in a no PNE game.
- Case 2: $F(\alpha, \beta) = 1$, since Φ is not valid, there exists $\beta' \in \{0,1\}^m$ such that $F(\alpha, \beta') = 0$. Therefore player 2 has an incentive to change strategy β by β'.
- Therefore, the strategy profile is not a PNE.

PNGrant problem

PNGrant Given a strategic game Γ and a value v, decide whether there is a PNE s so the $u_1(s) \geqslant v$.

Theorem

The PNGrant problem
can be solved in polynomial time for strategic games given in explicit form but it
is coNP-complete for strategic games given in general form
is Σ_2^p-complete for strategic games given in implicit form.

PNGrant problem

PNGrant Given a strategic game Γ and a value v, decide whether there is a PNE s so the $u_1(s) \geqslant v$.

Theorem

The PNGrant problem
can be solved in polynomial time for strategic games given in explicit form but it
is coNP-complete for strategic games given in general form
is Σ_2^p-complete for strategic games given in implicit form.

Membership follows from the same arguments.

PNGrant problem

PNGrant Given a strategic game Γ and a value v, decide whether there is a PNE s so the $u_1(s) \geqslant v$.

Theorem

The PNGrant problem
can be solved in polynomial time for strategic games given in explicit form but it
is coNP-complete for strategic games given in general form
is Σ_2^p-complete for strategic games given in implicit form.

Membership follows from the same arguments.
In all the reduction the utility for the first player in all PNE is constant, this provides the value of v in each reduction.

(Boolean) Circuit games

[Schoenebeck and Vadhan, EC 2006 - ACM TCT 2012]

- In a circuit game, players still control disjoint sets of variables, but each player's payoff is given by a single boolean circuit.
- The boolean circuit computes a rational value as the quotient of two integers
- Boolean circuit games are the special case of circuit games where each player controls a single boolean variable.

(Boolean) Circuit games

[Schoenebeck and Vadhan, EC 2006 - ACM TCT 2012]

- In a circuit game, players still control disjoint sets of variables, but each player's payoff is given by a single boolean circuit.
- The boolean circuit computes a rational value as the quotient of two integers
- Boolean circuit games are the special case of circuit games where each player controls a single boolean variable.

TMs can be simulated by circuits and viceversa

(Boolean) Circuit games

[Schoenebeck and Vadhan, EC 2006 - ACM TCT 2012]

- In a circuit game, players still control disjoint sets of variables, but each player's payoff is given by a single boolean circuit.
- The boolean circuit computes a rational value as the quotient of two integers
- Boolean circuit games are the special case of circuit games where each player controls a single boolean variable.

TMs can be simulated by circuits and viceversa

- Circuit games are equivalent to implicit form games
- Boolean circuit games are a subset of general form games.

(Boolean) weighted formula games

[Mavronicolas, Monien, Wagner, WINE 2007]

- In a formula game, players still control disjoint sets of variables, but each player's payoff is given by a weighted combination of boolean formulas.
- Boolean formula games are the special case of formula games where each player controls a single boolean variable.

(Boolean) weighted formula games

[Mavronicolas, Monien, Wagner, WINE 2007]

- In a formula game, players still control disjoint sets of variables, but each player's payoff is given by a weighted combination of boolean formulas.
- Boolean formula games are the special case of formula games where each player controls a single boolean variable.
- Formulas can be casted as circuits but not viceversa as the size might grow exponentially.

(Boolean) weighted formula games

[Mavronicolas, Monien, Wagner, WINE 2007]

- In a formula game, players still control disjoint sets of variables, but each player's payoff is given by a weighted combination of boolean formulas.
- Boolean formula games are the special case of formula games where each player controls a single boolean variable.

- Formulas can be casted as circuits but not viceversa as the size might grow exponentially.
- Nevertheless the utility functions of the provided reductions can be easily described in this way.
  So the problems are equivalent from the complexity point of view.

Graphical games

[Gottlob, Greco and Scarcello, JAIR 2005]

- Graphical games are a representation of multiplayer games meant to capture and exploit locality or sparsity of direct influences.
- They are most appropriate for large population games in which the payoffs of each player are determined by the actions of only a small subpopulation.
- Players' relationship is described by a graph and the payoff of a player depends only on the actions of its neighbors.

Graphical games

[Gottlob, Greco and Scarcello, JAIR 2005]

- Graphical games are a representation of multiplayer games meant to capture and exploit locality or sparsity of direct influences.
- They are most appropriate for large population games in which the payoffs of each player are determined by the actions of only a small subpopulation.
- Players' relationship is described by a graph and the payoff of a player depends only on the actions of its neighbors.
- Provide a complementary framework to analyze complexity based on the graph parameters:

Graphical games

[Gottlob, Greco and Scarcello, JAIR 2005]

- Graphical games are a representation of multiplayer games meant to capture and exploit locality or sparsity of direct influences.
- They are most appropriate for large population games in which the payoffs of each player are determined by the actions of only a small subpopulation.
- Players' relationship is described by a graph and the payoff of a player depends only on the actions of its neighbors.
- Provide a complementary framework to analyze complexity based on the graph parameters: bounded degree, bounded treewidth, . . .

Uniform families of games with polynomial time computable utilities

- We have analyzed the representations of the strategic games as potential inputs to a problem.
- However those representation forms do not capture completely the notion of games whose utility functions are computable in polynomial time as we expect to have a TM describing the game family and not a TM per game.

Uniform families of games with polynomial time computable utilities

- We have analyzed the representations of the strategic games as potential inputs to a problem.
- However those representation forms do not capture completely the notion of games whose utility functions are computable in polynomial time as we expect to have a TM describing the game family and not a TM per game.
- Even though in many papers studying the computational complexity of some specific games, it is assumed that the utilities are computable in polynomial time this assumption has different interpretations.

Uniform families of games with polynomial time computable utilities

- We adopt: games defined uniformly by polynomial time Turing machines.

Uniform families of games with polynomial time computable utilities

- We adopt: games defined uniformly by polynomial time Turing machines.
- Let M be a DTM and let us assume that an alphabet Σ is fixed.
- We define uniformly families of games associated to M:

Uniform families of games with polynomial time computable utilities

- We adopt: games defined uniformly by polynomial time Turing machines.

- Let M be a DTM and let us assume that an alphabet Σ is fixed.
- We define uniformly families of games associated to M:
- Observe that this approach do not make sense for the explicit forms.

Implicit form

M-implicit form strategic family *Each instance of the family specifies the number of players n and their set of actions in an succinct way.*

$$\{\langle 1^n, 1^{m_1}, \ldots, 1^{m_n}\rangle \mid n, m_1, \ldots, m_n \in \mathbb{N}\}.$$

In the game described by $\langle 1^n, 1^{m_1}, \ldots, 1^{m_n}\rangle$, $A_i = \Sigma^{\leqslant m_i}$ *and if a is a strategy profile of such a game, and* $1 \leqslant i \leqslant n$, *then the utility of the i-th player on a is defined as* $u_i(a) = M(a, i)$.

General form

M-general form strategic family. Each instance of the family describes a game by giving the number of players n and explicitly listing the set of actions of each player.

$$\{\langle 1^n, A_1, \dots, A_n\rangle \mid n, m \in \mathbb{N} \land \forall i \; A_i \subseteq \Sigma^*\}$$

In the game described by $\langle 1^n, A_1, \dots, A_n\rangle$*, if a is a strategy profile of such game, and* $1 \leqslant i \leqslant n$*, then the utility of the i-th player on a is defined as* $u_i(a) = M(a, i)$.

Hence, given a family of games defined from a polynomial time DTM M, we can also pose the question of determining whether a game of this family has a Nash equilibrium.

M-Exists Pure Nash (M-EPN)
Given a game Γ, defined uniformly by M, decide whether Γ has a Pure Nash equilibrium.

M-Pure Nash equilibrium with guarantee (M-PNGrant)
Given a game Γ, defined uniformly by M, a value u, and a player i, decide whether Γ has a Pure Nash equilibrium in which player i gets payoff u or higher.

Theorem

- *There exists a polynomial time* DTM *M for which the M-EPN problem for games in the M-implicit form strategic family is Σ_2^p-complete.*
- *There exists a polynomial time* DTM *M for which the M-EPN problem for games in the M-general form strategic family is NP-complete.*
- *There exists a polynomial time* DTM *M for which the M-*PNGRANT *problem for games in the M-implicit form strategic family is Σ_2^p-complete.*
- *There exists a polynomial time* DTM *M for which the M-*PNGRANT *problem for games in the M-general form strategic family is NP-complete.*

Uniformity vs non-uniformity

- We are considering uniform families of games.
- The main difference with respect to the proofs of the analogous results in the previous section is that the DTM can not be parameterized by the quantified boolean formula Φ or the CNF formula F.

Uniformity vs non-uniformity

- We are considering uniform families of games.
- The main difference with respect to the proofs of the analogous results in the previous section is that the DTM can not be parameterized by the quantified boolean formula Φ or the CNF formula F.
- Now these formulae should be part of the input of the machines.
- This requires an additional trick.

EPN problem: implicit uniform representation

- For any fixed polynomial time DTM M, the problem of deciding whether a game Γ in M-implicit form has a PNE can be solved by an Alternating TM, with 2 alternations, existential and universal, in polynomial time. Hence M-SPN $\in \Sigma_2^p$.

EPN problem: implicit uniform representation

- To prove hardness, we have to define first the polynomial time DTM M.
- Let M be the TM such that on input $(\Phi, a_1, a_2, a_3, a_4, i)$ being $\Phi = \exists \alpha_1, \dots, \alpha_{n_1} \forall \beta_1, \dots \beta_{n_2} F$ an instance of the Q2SAT problem, $a_1 \in A_1 = \{0,1\}^{n_1}$, $a_2 \in A_2 = \{0,1\}^{n_2}$ and $a_3, a_4 \in \{0,1\}$, computes the utilities defined as before.
- M works in polynomial time with respect to the input length.

EPN problem: implicit uniform representation

- For each Φ we define a game $\Gamma(\Phi)$ with five players. Players 1, 2, 3 and 4 are defined exactly equal to the four players of the game defined in the previous reduction.

EPN problem: implicit uniform representation

- For each Φ we define a game $\Gamma(\Phi)$ with five players. Players 1, 2, 3 and 4 are defined exactly equal to the four players of the game defined in the previous reduction.
- We have an additional player, player 0 who has a unique action that defines the rules of the game, i.e. $A_0 = \{\Phi\}$.

EPN problem: implicit uniform representation

- For each Φ we define a game $\Gamma(\Phi)$ with five players. Players 1, 2, 3 and 4 are defined exactly equal to the four players of the game defined in the previous reduction.
- We have an additional player, player 0 who has a unique action that defines the rules of the game, i.e. $A_0 = \{\Phi\}$.
- As we have shown, Φ is valid if and only if $\Gamma(\Phi)$ has a PNE, and the description of $\Gamma(\Phi)$ in implicit form can be obtained in polynomial time.

Conclusions

- We have analyzed some ways of describing games with polynomial time computable utilities: uniform and non-uniform models for strategic games.
- We have concentrated on the study of two computational problems.
- As expected complexity increases with succinctness.
- There are many other
  - game classes
  - and problems of interest

  not covered in this talk.

References

Contents taken from a subset of the results in

- C. Alvarez, J. Gabarr'o, M. Serna
  Equilibria problems on games: Complexity versus succinctness
  J. of Comp. and Sys. Sci. 77:1172-1197, 2011

References

Further suggested reading (among many others)

- G. Gottlob, G. Greco, F. Scarcello
  Pure Nash equilibria: Hard and easy games
  J. Artificial Intelligence Res. 24:357–406, 2005
- J. Gabarró, A. García, M. Serna
  The complexity of game isomorphism
  Theor. Comput. Sci. 412(48): 6675-6695, 2011.

References

- M. Mavronicolas, B. Monien, K. Wagner
  Weighted boolean formula games
  in: X. Deng, F. Graham (Eds.), WINE 2007,
  Lecture Notes in Comput. Sci., 4858:469–481, 2007.
- G.R. Schoenebeck, S. Vadhan
  The Computational Complexity of Nash Equilibria in Concisely Represented Games
  ACM Transactions on Computational Theory, 4(2) article 4, 2012

Thanks!

mjserna@cs.upc.edu