Quantifying the Efficiency of Congestion Games

Martin Gairing

University of Liverpool, Computer Science Dept

WINE Tutorial, December 2014
Motivation: Systems with Selfish Agents

Our Focus

- Problems in which multiple agents interact

Motivation: the Internet

- Billions of users
- Tens of thousands of autonomous systems
- By design, centralized control is impossible
  - Technical constraints – resources
  - Political constraints – ISP, countries
- Decentralized operation and ownership
- Distributed control by competing entities
Motivation: Systems with Selfish Agents

Selfish Agents

- Have their own private objectives
- Are rational and selfish
  - Make choices to maximize their profit
  - Profit depends on choices of all agents

Goal

- Algorithms that account for strategic behavior by selfish agents

Natural Tool: Game Theory

- Theory of rational behavior in competitive, collaborative settings
  - [von Neumann/Morgenstern 1944]
Objectives

This Talk

- Understand consequences of non-cooperative behavior
- What is the “cost” of selfish behavior?
  - the price of anarchy [Koutsoupias/Papadimitriou 99]
  - the price of stability [Anshelevich et al. 04]

Our Scenario

- General model for non-cooperative sharing of resources
- Congestion games
Example
Motivating Example

Example

100 cars need to go from $s$ to $t$.

\begin{align*}
f(x) &= x \\
f(x) &= 100
\end{align*}

Question

What will selfish network users do?
Motivating Example

Example

100 cars need to go from s to t.

Question

What will selfish network users do?

Claim

In Nash equilibrium all traffic will take the top link.
Can we do better?

Example

100 cars need to go from $s$ to $t$. 

Consider instead

- ▶ 50 cars have delay 100 (same as before)
- ▶ 50 cars have delay 50 (big improvement!)
Can we do better?

Example

100 cars need to go from $s$ to $t$.

Consider instead

traffic split equally

- 50 cars have delay 100 (same as before)
- 50 cars have delay 50 (big improvement!)
Braess’s Paradox

Initial Network

delay=150
Braess’s Paradox

Initial Network

Augmented Network

What now?

delay=150
Braess’s Paradox [BRAESS, 68]

Initial Network

Augmented Network

delay=150
delay=200
(Weighted) congestion games

\[ \Gamma = (\mathcal{N}, (w_i)_{i \in \mathcal{N}}, E, (S_i)_{i \in \mathcal{N}}, (c_e)_{e \in E}) \]

- \( \mathcal{N} \) ... set of \( k \) players
- \( w_i \) ... weight of player \( i \in \mathcal{N} \)
- \( E \) ... set of resources e.g. edges in a graph
- \( S_i \subseteq 2^E \) ... set of strategies of player \( i \) e.g. set of paths from \( o_i \) to \( d_i \)
- \( c_e \) ... latency function of resource versus...
(Weighted) congestion games

\[ \Gamma = (\mathcal{N}, (w_i)_{i \in \mathcal{N}}, E, (S_i)_{i \in \mathcal{N}}, (c_e)_{e \in E}) \]

- \( \mathcal{N} \) ... set of \( k \) players
- \( w_i \) ... weight of player \( i \in \mathcal{N} \)
- \( E \) ... set of resources
e.g. edges in a graph
(Weighted) congestion games

\[ \Gamma = (\mathcal{N}, (w_i)_{i \in \mathcal{N}}, E, (S_i)_{i \in \mathcal{N}}, (c_e)_{e \in E}) \]

- \( \mathcal{N} \) . . . set of \( k \) players
- \( w_i \) . . . weight of player \( i \in \mathcal{N} \)
- \( E \) . . . set of resources
e.g. edges in a graph
- \( S_i \subseteq 2^E \) . . . set of strategies of player \( i \)
e.g. set of paths from \( o_i \) to \( d_i \)
(Weighted) congestion games

\[ \Gamma = (\mathcal{N}, (w_i)_{i \in \mathcal{N}}, E, (S_i)_{i \in \mathcal{N}}, (c_e)_{e \in E}) \]

- \( \mathcal{N} \) ... set of \( k \) players
- \( w_i \) ... weight of player \( i \in \mathcal{N} \)
- \( E \) ... set of resources
e.g. edges in a graph
- \( S_i \subseteq 2^E \) ... set of strategies of player \( i \)
e.g. set of paths from \( o_i \) to \( d_i \)
- \( c_e \) ... latency function of resource \( e \)
Subclasses of (weighted) congestion games

- unweighted congestion games (or simply congestion games):
  
  \[ w_i = 1 \text{ for all player } i \in \mathcal{N} \]
Subclasses of (weighted) congestion games

- unweighted congestion games (or simply congestion games):
  \[ w_i = 1 \text{ for all player } i \in \mathcal{N} \]

- symmetric games:
  \[ S_i = S_j \text{ for all player } i, j \in \mathcal{N} \]
Subclasses of (weighted) congestion games

- unweighted congestion games (or simply congestion games):
  \[ w_i = 1 \text{ for all player } i \in \mathcal{N} \]

- symmetric games:
  \[ S_i = S_j \text{ for all player } i, j \in \mathcal{N} \]

- network congestion games
Subclasses of (weighted) congestion games

- **unweighted congestion games (or simply congestion games):**
  \[ w_i = 1 \quad \text{for all player } i \in \mathcal{N} \]

- **symmetric games:**
  \[ S_i = S_j \quad \text{for all player } i, j \in \mathcal{N} \]

- **network congestion games**

- **singleton congestion games**
Load and Private Cost

Strategy profile
\[ s = (s_1, \ldots, s_n) \in S_1 \times \ldots \times S_n \]

Traffic on resource \( e \in E \)
\[ x_e(s) = \sum_{i \in N : e \in s_i} w_i \]

Private cost of player \( i \in N \)
\[ C_i(s) = w_i \cdot \sum_{e \in s_i} c_e(x_e(s)) \]

Quantifying the Efficiency of Congestion Games
Martin Gairing · 11
Nash Equilibrium

A strategy profile $s$ is a Nash equilibrium if and only if all players $i \in \mathcal{N}$ are satisfied, that is,

$$C_i(s) \leq C_i(s_{-i}, s'_i) \quad \text{for all } i \in \mathcal{N} \text{ and } s'_i \in S_i.$$
Nash Equilibrium

A strategy profile $s$ is a Nash equilibrium if and only if all players $i \in \mathcal{N}$ are satisfied, that is,

$$C_i(s) \leq C_i(s_{-i}, s'_i) \quad \text{for all } i \in \mathcal{N} \text{ and } s'_i \in S_i.$$ 

Remarks

- For simplicity we restrict to pure Nash equilibria.
- Many results hold also for mixed Nash equilibria.
  - Players randomize over their pure strategies
  - Guaranteed to exist [NASH, 1951]
### Theorem

[Rosenthal, 1973]

Every **unweighted** congestion game possesses a pure Nash equilibrium.

---

Define \( \Phi : (S_1 \times \ldots \times S_n) \rightarrow \mathbb{N} \) by

\[
\Phi(s) = \sum_{e \in E} x_e(s) - \sum_{j=1}^m c_e(j).
\]

Consider two strategy profiles \( s = (s_1, \ldots, s_k) \) and \( s' = (s'_i, s_{-i}) \):

\[
\Phi(s) - \Phi(s') = \sum_{e \in s_i - s'_i} c_e(x_e(s)) - \sum_{e \in s'_i - s_i} c_e(x_e(s')).
\]

Therefore:

\( \Phi(s) \) minimal \( \Rightarrow \) \( s \) is Nash equilibrium.
Existence of pure NE: positive result

**Theorem**

Every *unweighted* congestion game possesses a pure Nash equilibrium.

Define $\Phi : (S_1 \times ... \times S_n) \rightarrow \mathbb{N}$ by

$$
\Phi(s) = \sum_{e \in E} \sum_{j=1}^{x_e(s)} c_e(j).
$$

Therefore:

$$
\Phi(s) \text{ minimal} \Rightarrow s \text{ is Nash equilibrium}.
$$
Existence of pure NE: positive result

Theorem [Rosenthal, 1973]

Every unweighted congestion game possesses a pure Nash equilibrium.

Define $\Phi : (S_1 \times \ldots \times S_n) \to \mathbb{N}$ by

$$\Phi(s) = \sum_{e \in E} \sum_{j=1}^{c_e(s)} c_e(j).$$

Consider two strategy profiles $s = (s_1, \ldots, s_k)$ and $s' = (s'_i, s_{-i})$:

$$\Phi(s) - \Phi(s') = \sum_{e \in s_i - s'_i} c_e(x_e(s)) - \sum_{e \in s'_i - s_i} c_e(x_e(s'))$$

$$= C_i(s) - C_i(s').$$

Therefore: $\Phi(s)$ minimal $\Rightarrow$ $s$ is Nash equilibrium.
Existence of pure NE: negative result


**Theorem**

There is a *weighted* network congestion game that does *not admit* a pure Nash equilibrium.

Consider the following instance:

- 2 players
- $w_1 = 1$
- $w_2 = 2$
Existence of pure NE in weighted games

**Theorem**

Every **weighted** congestion game with **linear** latency functions possesses a pure Nash equilibrium.

Proof is based on the following potential function:

\[
\tilde{\Phi}(s) = \sum_{i \in \mathcal{N}} w_i \cdot \sum_{e \in s_i} (c_e(x_e(s)) + c_e(w_i))
\]

\[
= \sum_{e \in E} x_e(s) \cdot c_e(x_e(s)) + \sum_{i \in \mathcal{N}} w_i \cdot \sum_{e \in s_i} c_e(w_i).
\]

If \( s = (s_1, \ldots, s_k) \) and \( s' = (s'_j, s_{-j}) \) for some \( j \in \mathcal{N} \) and \( s'_j \in S_j \), then

\[
\tilde{\Phi}(s) - \tilde{\Phi}(s') = 2 \cdot (C_j(s) - C_j(s')).
\]
Do weighted congestion games always possess pure Nash Equilibria?
▶ Yes, for unweighted players. [Rosenthal, ’73]

**Rosenthal’s Potential Function**

\[
\Phi(s) = \sum_{e \in E} \sum_{i=1}^{x_e(s)} c_e(i)
\]

If a player decreases her cost by \( \Delta \) then also the potential decreases by \( \Delta \).
Existence and Complexity of Pure NE

- Do weighted congestion games always possess pure Nash Equilibria?
  - Yes, for unweighted players. [Rosenthal, ’73]
  - No. [Libman, Orda, ’01]
  - Characterisation. [Fotakis, Kontogiannis, Spirakis, ’04]
    [Goemans, Mirrokni, Vetta, ’05]
    [Harks, Klimm, ’12]
Existence and Complexity of Pure NE

- Do weighted congestion games always possess pure Nash Equilibria?
  - Yes, for unweighted players. [Rosenthal, ’73]
  - No. [Libman, Orda, ’01]
  - Characterisation. [Fotakis, Kontogiannis, Spirakis, ’04]
  - Characterisation. [Goemans, Mirrokni, Vetta, ’05]
  - Characterisation. [Harks, Klimm, ’12]

- Complexity of deciding for pure Nash equilibria?
  - NP-complete [Dunkel, Schulz, ’06]

- Complexity of computing pure Nash equilibria (unweighted)?
  - PLS-complete [Fabrikant, Papadimitriou, Talwar, ’04]
Price of Anarchy

http://thetyee.ca/News/2007/10/10/ChinaAutoMad/
Price of Anarchy

Social Cost

- Different definitions possible
- Here: **Total Latency**

\[
\text{SC}(s) = \sum_{i \in \mathcal{N}} C_i(s) = \sum_{e \in E} x_e(s) \cdot c_e(x_e(s))
\]

- Let \( \mathcal{G} \) be a class of games.

Price of Anarchy

[Koutsoupias, Papadimitriou, STACS’99]

\[
\text{PoA}(\mathcal{G}) = \sup_{\Gamma \in \mathcal{G}, \text{ s is NE in } \Gamma} \frac{\text{SC}(s)}{\text{OPT}}
\]
Price of Anarchy: Example

Network with 2 (unweighted) players

Symmetric
Price of Anarchy: Example

Nash Equilibrium

SC = 14 + 14 = 28
Price of Anarchy: Example

Nash Equilibrium

SC = 14 + 14 = 28

OPT

SC = 14 + 10 = 24
Price of Anarchy: Example

Nash Equilibrium

\[ SC = 14 + 14 = 28 \]

OPT

\[ SC = 14 + 10 = 24 \]

Price of Anarchy = \( \frac{28}{24} = \frac{7}{6} \)

If multiple equilibria, look at worst one
Price of Anarchy: State of the Art

(1) **Analytical simple classes** of cost functions
⇒ exact formula for PoA.

▶ linear

[C. CHRISTODOULOU, KOUTSOUPIAS, STOC'05]
[A. WERBUCH, A. ZAR, E. PEITSTEIN, STOC'05]
▶ bounded degree polynomials

[A. LAND ET AL, STACS'06]

▶ non-atomic (Wardrop model)

[R. ROUGHGARDEN, T. TARDOS, JACM'00]
▶ unweighted

[R. ROUGHGARDEN, STOC'09]
▶ weighted

[B. BHAWALKAR, G. G. AIRING, R. ROUGHGARDEN, ESA'10]

(3) Understanding of game complexity required for worst-case PoA to be realized.

▶ Ideally independent of cost functions.

▶ e.g. symmetric strategy sets, singleton strategy sets
Price of Anarchy: State of the Art

(1) **Analytical simple classes** of cost functions
   \[\Rightarrow\] **exact formula for PoA.**
   
   - linear
     
   - bounded degree polynomials

[Christodoulou, Koutsoupias, STOC’05]
[Awerbuch, Azar, Epstein, STOC’05]
[Aland et al., STACS’06]
Price of Anarchy: State of the Art

(1) Analytical simple classes of cost functions
⇒ exact formula for PoA.
  ▶ linear
    [Christodoulou, Koutsoupias, STOC’05]
    [Awerbuch, Azar, Epstein, STOC’05]
  ▶ bounded degree polynomials

(2) For every set of allowable cost functions
⇒ recipe for computing PoA.

[Aland et al., STACS’06]
Price of Anarchy: State of the Art

(1) **Analytical simple classes** of cost functions
    ⇒ **exact formula** for PoA.
    ▶ linear
      [Christodoulou, Koutsoupias, STOC’05]
      [Awerbuch, Azar, Epstein, STOC’05]
    ▶ bounded degree polynomials
      [Aland et al., STACS’06]

(2) For **every set** of allowable cost functions
    ⇒ **recipe** for computing PoA.
    ▶ non-atomic (Wardrop model)
      [Roughgarden, Tardos, JACM’00]
    ▶ unweighted
      [Roughgarden, STOC’09]
    ▶ weighted
      [Bhawalkar, Gairing, Roughgarden, ESA’10]
Price of Anarchy: State of the Art

(1) **Analytical simple classes** of cost functions ⇒ **exact formula** for PoA.
  ▶ linear [Christodoulou, Koutsoupias, STOC’05]
  ▶ bounded degree polynomials [Awerbuch, Azar, Epstein, STOC’05]

(2) For **every set** of allowable cost functions ⇒ **recipe** for computing PoA.
  ▶ non-atomic (Wardrop model) [Roughgarden, Tardos, JACM’00]
  ▶ unweighted [Roughgarden, STOC’09]
  ▶ weighted [Bhawalkar, Gairing, Roughgarden, ESA’10]

(3) Understanding of **game complexity** required for worst-case PoA to be realized.
  ▶ Ideally independent of cost functions.
  ▶ e.g. symmetric strategy sets, singleton strategy sets
Abstract Setup

- $n$ players, each picks a strategy $s_i$
- player $i$ incurs cost $C_i(s)$
Abstract Setup

- $n$ players, each picks a strategy $s_i$
- player $i$ incurs cost $C_i(s)$
- Important Assumption: objective function is $SC(s) = \sum_i C_i(s)$

Definition: [Roughgarden, STOC'09]

A game is $(\lambda, \mu) - \text{smooth}$ if for every pair $s, s^*$ of outcomes:

$$\sum_i C_i(s_{-i}, s^*_i) \leq \lambda \cdot SC(s^*) + \mu \cdot SC(s).$$

($\lambda > 0, \mu < 1$)
Theorem

If a game $G$ is $(\lambda, \mu) - \text{smooth}$, then

$$PoA(G) \leq \frac{\lambda}{1 - \mu}.$$
Smoothness $\Rightarrow$ PoA bound

**Theorem**

If a game $G$ is $(\lambda, \mu)$ – smooth, then

$$\text{PoA}(G) \leq \frac{\lambda}{1 - \mu}.$$ 

**Proof: s is a NE, $s^*$ is optimum**

$$\text{SC}(s) = \sum_i C_i(s) \leq \sum_i C_i(s_{-i}, s_i^*) \leq \lambda \cdot \text{SC}(s^*) + \mu \cdot \text{SC}(s)$$
Smoothness $\implies$ PoA bound

**Theorem**

If a game $G$ is $(\lambda, \mu)$ – **smooth**, then

$$\text{PoA}(G) \leq \frac{\lambda}{1 - \mu}.$$ 

**BUT**: smoothness is stronger

Diagram showing the hierarchy of solution concepts:

- no regret
- correlated eq
- mixed Nash
- pure Nash
Back to congestion games

- $\mathcal{C}$ ... arbitrary class of cost functions

Consider the set:

- $\mathcal{A}(\mathcal{C}) = \{ (\lambda, \mu) : x^* \cdot c(x + x^*) \leq \lambda \cdot x^* \cdot c(x^*) + \mu \cdot x \cdot c(x) \}$

where

- $0 \leq \mu < 1$ and $\lambda > 0$
- constraints range over all $c \in \mathcal{C}$ and $x \geq 0$ and $x^* > 0$. 
Back to congestion games

- $\mathcal{C}$ ... arbitrary class of cost functions

Consider the set:

- $A(\mathcal{C}) = \{ (\lambda, \mu) : x^* \cdot c(x + x^*) \leq \lambda \cdot x^* \cdot c(x^*) + \mu \cdot x \cdot c(x) \}$

where

- $0 \leq \mu < 1$ and $\lambda > 0$
- constraints range over all $c \in \mathcal{C}$ and $x \geq 0$ and $x^* > 0$.

Local smoothness implies global smoothness

For a class of functions $\mathcal{C}$, if $(\lambda, \mu) \in A(\mathcal{C})$ then every weighted congestion game with cost functions in $\mathcal{C}$ is $(\lambda, \mu)$-smooth.
Back to congestion games

- $\mathcal{C}$ ... arbitrary class of cost functions

Consider the set:

\[ \mathcal{A}(\mathcal{C}) = \{ (\lambda, \mu) : \mathbf{x}^* \cdot c(\mathbf{x} + \mathbf{x}^*) \leq \lambda \cdot \mathbf{x}^* \cdot c(\mathbf{x}^*) + \mu \cdot x \cdot c(x) \} \]

where

- $0 \leq \mu < 1$ and $\lambda > 0$
- constraints range over all $c \in \mathcal{C}$ and $x \geq 0$ and $\mathbf{x}^* > 0$.

Local smoothness implies global smoothness

For a class of functions $\mathcal{C}$, if $(\lambda, \mu) \in \mathcal{A}(\mathcal{C})$ then every weighted congestion game with cost functions in $\mathcal{C}$ is $(\lambda, \mu)$-smooth.

For unweighted congestion games: redefine $\mathcal{A}(\mathcal{C})$:

\[ \mathcal{A}(\mathcal{C}) = \{ (\lambda, \mu) : \mathbf{x}^* \cdot c(\mathbf{x} + 1) \leq \lambda \cdot \mathbf{x}^* \cdot c(\mathbf{x}^*) + \mu \cdot x \cdot c(x) \} \]

- and restrict $x, \mathbf{x}^*$ to be integer
Example

- Unweighted congestion games with $C = \{ c_1 \}$
  - $c_1(x) = x$
  - $\mathcal{A}(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$
Example

- Unweighted congestion games with $C = \{c_1\}$
  - $c_1(x) = x$
  - $A(C) = \{(\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)}\}$

$\lambda$

$\mu = 1$

$0 \leq \mu < 1$ and $\lambda > 0$
Example

- Unweighted congestion games with $C = \{ c_1 \}$
  - $c_1(x) = x$
  - $A(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

- $0 \leq \mu < 1$ and $\lambda > 0$
- Constraint for each $(c_1, x, x^*)$
Example

- Unweighted congestion games with $C = \{ c_1 \}$
  - $c_1(x) = x$
  - $A(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

Constraint for each $(c_1, x, x^*)$

- $0 \leq \mu < 1$ and $\lambda > 0$
Example

- Unweighted congestion games with $C = \{ c_1 \}$
  - $c_1(x) = x$
  - $A(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

- $0 \leq \mu < 1$ and $\lambda > 0$
- Constraint for each $(c_1, x, x^*)$
Example

- Unweighted congestion games with $C = \{c_1\}$
  - $c_1(x) = x$
  - $A(C) = \{(\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)}\}$

- 0 \leq \mu < 1 \text{ and } \lambda > 0
- Constraint for each $(c_1, x, x^*)$
Example

- Unweighted congestion games with \( C = \{ c_1 \} \)
  - \( c_1(x) = x \)
  - \( A(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \} \)

- 0 \( \leq \) \( \mu \) < 1 and \( \lambda > 0 \)
- Constraint for each \((c_1, x, x^*)\)
Example

- Unweighted congestion games with $\mathcal{C} = \{c_1\}$
  - $c_1(x) = x$
  - $\mathcal{A}(\mathcal{C}) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

- $0 \leq \mu < 1$ and $\lambda > 0$
- Constraint for each $(c_1, x, x^*)$
Example

- Unweighted congestion games with $C = \{ c_1 \}$
  - $c_1(x) = x$
  - $A(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

- $0 \leq \mu < 1$ and $\lambda > 0$
- Constraint for each $(c_1, x, x^*)$

Best possible upper bound on PoA:

- $\zeta(C) = \inf \left\{ \frac{\lambda}{1-\mu} : (\lambda, \mu) \in A(C) \right\}$

Quantifying the Efficiency of Congestion Games

Martin Gairing
Exercise

- Unweighted congestion games with \( C = \{ c_1, c_2 \} \)
  - \( c_1(x) = x \), \( c_2(x) = \min\{9, (x + 1)^2\} \)
  - \( A(C) = \{(\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^\ast)} - \mu \cdot \frac{x \cdot c(x)}{x^\ast \cdot c(x^\ast)}\} \)

\[ \mu = 1 \quad \text{and} \quad 0 \leq \mu < 1 \text{ and } \lambda > 0 \]
Exercise

Unweighted congestion games with \( C = \{ c_1, c_2 \} \)

- \( c_1(x) = x \), \( c_2(x) = \min\{9, (x + 1)^2 \} \)

- \( A(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \} \)

- 0 \leq \mu < 1 \) and \( \lambda > 0 \)

- Constraint for each \( (c_1, x, x^*) \)
Unweighted congestion games with $C = \{c_1, c_2\}$

- $c_1(x) = x$, $c_2(x) = \min\{9, (x + 1)^2\}$

- $A(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

- $0 \leq \mu < 1$ and $\lambda > 0$
- Constraint for each $(c_1, x, x^*)$
Exercise

- Unweighted congestion games with $C = \{c_1, c_2\}$
  - $c_1(x) = x$, $c_2(x) = \min\{9, (x + 1)^2\}$
  - $\mathcal{A}(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

Constraint for each $(c_1, x, x^*)$
Constraint for each $(c_2, x, x^*)$

- $0 \leq \mu < 1$ and $\lambda > 0$
Exercise

- Unweighted congestion games with $C = \{c_1, c_2\}$
  - $c_1(x) = x$, $c_2(x) = \min\{9, (x + 1)^2\}$
  - $\mathcal{A}(C) = \{ (\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)} \}$

- $0 \leq \mu < 1$ and $\lambda > 0$
- Constraint for each $(c_1, x, x^*)$
- Constraint for each $(c_2, x, x^*)$
Exercise

- Unweighted congestion games with $C = \{c_1, c_2\}$
  - $c_1(x) = x$, $c_2(x) = \min\{9, (x + 1)^2\}$
  - $A(C) = \{(\lambda, \mu) : \lambda \geq \frac{c(x+1)}{c(x^*)} - \mu \cdot \frac{x \cdot c(x)}{x^* \cdot c(x^*)}\}$

- $0 \leq \mu < 1$ and $\lambda > 0$
- Constraint for each $(c_1, x, x^*)$
- Constraint for each $(c_2, x, x^*)$

Best possible upper bound on PoA:

- $\zeta(C) = \inf \left\{ \frac{\lambda}{1 - \mu} : (\lambda, \mu) \in A(C) \right\}$
Questions:

▶ Are such upper bounds tight?
Questions:

- Are such upper bounds tight?
  - Yes, for unweighted. [ROUGHGARDEN, 2009]
  - Yes, for weighted with mild assumption on $C$. [BHAWALKAR, GAIRING, ROUGHGARDEN, 2010]

Closure under scaling and dilation:

- If $c(x) \in C$ and $r \in \mathbb{R}^+$ then
  - $r \cdot c(x) \in C$
  - $c(r \cdot x) \in C$
  - $\zeta(C)$ for linear/polynomial cost functions?
Questions

Questions:

- Are such upper bounds tight?
  - Yes, for unweighted. [ROUGHGARDEN, 2009]
  - Yes, for weighted with mild assumption on $C$. [BHAWALKAR, GAIRING, ROUGHGARDEN, 2010]

Closure under scaling and dilation:

If $c(x) \in C$ and $r \in \mathbb{R}^+$ then

- $r \cdot c(x) \in C$
- $c(r \cdot x) \in C$
Questions:

▶ Are such upper bounds tight?
  ▶ Yes, for unweighted. [ROUGHGARDEN, 2009]
  ▶ Yes, for weighted with mild assumption on $\mathcal{C}$.
    [BHAWALKAR, GAIRING, ROUGHGARDEN, 2010]

Closure under scaling and dilation:

If $c(x) \in \mathcal{C}$ and $r \in \mathbb{R}^+$ then

▶ $r \cdot c(x) \in \mathcal{C}$
▶ $c(r \cdot x) \in \mathcal{C}$

▶ $\zeta(\mathcal{C})$ for linear/polynomial cost functions?
PoA for linear/polynomial

- Polynomial latency functions: $C_d = \left\{ c \mid c(x) = \sum_{i=0}^{d} a_i \cdot x^i \right\}$
- $\Phi_d$ is solution to $(\Phi_d + 1)^d = \Phi_d^{d+1}$.
- $k = \lfloor \Phi_d \rfloor$
PoA for linear/polynomial

- Polynomial latency functions: \( C_d = \left\{ c \mid c(x) = \sum_{i=0}^{d} a_i \cdot x^i \right\} \)
- \( \Phi_d \) is solution to \((\Phi_d + 1)^d = \Phi_{d+1}^d\).
- \( k = \lfloor \Phi_d \rfloor \)

**Theorem**

If all latency functions are from \( C_d \), then for

(a) weighted congestion games: \( \text{PoA} = \Phi_{d+1}^d \)

(b) unweighted congestion games: \( \text{PoA} = \frac{(k+1)^{2d+1} - k^{d+1} \cdot (k+2)^d}{(k+1)^{d+1} - (k+2)^d + (k+1)^d - k^{d+1}} \)

**Corollary**

For the linear case (\( d = 1 \)) we have:

(a) weighted congestion games: \( \text{PoA} = \Phi_2^2 = 3 + \sqrt{5} \approx 2.618 \)

(b) unweighted congestion games: \( \text{PoA} = 2.5 \)
PoA for linear/polynomial

- Polynomial latency functions: $C_d = \{ c \mid c(x) = \sum_{i=0}^{d} a_i \cdot x^i \}$
- $\Phi_d$ is solution to $(\Phi_d + 1)^d = \Phi_{d+1}^d$.
- $k = \lceil \Phi_d \rceil$

**Theorem**

If all latency functions are from $C_d$, then for

(a) weighted congestion games: $\text{PoA} = \Phi_{d+1}^d$

(b) unweighted congestion games: $\text{PoA} = \frac{(k+1)^{2d+1} - k^{d+1}(k+2)^d}{(k+1)^{d+1} - (k+2)^d + (k+1)^d - k^{d+1}}$

**Corollary**

For the linear case ($d = 1$) we have:

(a) weighted congestion games: $\text{PoA} = \Phi^2 = \frac{3+\sqrt{5}}{2} \approx 2.618$

(b) unweighted congestion games: $\text{PoA} = 2.5$
Proof Sketch.

- \( n \geq \lfloor \Phi_d \rfloor + 2 \text{ player} \)
- \( E = \{ g_1, \ldots, g_n \} \cup \{ h_1, \ldots, h_n \} \)
- \( c_{g_i}(x) = a \cdot x^d, \quad c_{h_i}(x) = x^d \)
- \( S_i = \{ Q_i, P_i \} \) with
  - \( Q_i = \{ g_i, h_i \} \)
  - \( P_i = \{ g_{i+1}, \ldots, g_{i+k}, h_{i+1}, \ldots, h_{i+k+1} \} \)

\[ d = 2, \ n = 4, \]
\[ k = \lfloor \Phi_d \rfloor = \lfloor 2.148 \rfloor \]
Proof Sketch.

- $n \geq \lceil \Phi_d \rceil + 2$ player
- $E = \{g_1, \ldots, g_n\} \cup \{h_1, \ldots, h_n\}$
- $c_{g_i}(x) = a \cdot x^d$, $c_{h_i}(x) = x^d$
- $S_i = \{Q_i, P_i\}$ with
  - $Q_i = \{g_i, h_i\}$
  - $P_i = \{g_{i+1}, \ldots, g_{i+k}, h_{i+1}, \ldots, h_{i+k+1}\}$

$d = 2$, $n = 4$, $k = \lceil \Phi_d \rceil = \lceil 2.148 \rceil$
Proof Sketch.

- \( n \geq \lfloor \Phi_d \rfloor + 2 \) player
- \( E = \{g_1, \ldots, g_n\} \cup \{h_1, \ldots, h_n\} \)
- \( c_{g_i}(x) = a \cdot x^d, \quad c_{h_i}(x) = x^d \)
- \( S_i = \{Q_i, P_i\} \) with
  - \( Q_i = \{g_i, h_i\} \)
  - \( P_i = \{g_{i+1}, \ldots, g_{i+k}, h_{i+1}, \ldots, h_{i+k+1}\} \)

\[ d = 2, \quad n = 4, \quad k = \lfloor \Phi_d \rfloor = \lfloor 2.148 \rfloor \]
Lower bound for unweighted games

Proof Sketch.

- $n \geq \lfloor \Phi_d \rfloor + 2$ player
- $E = \{g_1, \ldots, g_n\} \cup \{h_1, \ldots, h_n\}$
- $c_{g_i}(x) = a \cdot x^d$, $c_{h_i}(x) = x^d$
- $S_i = \{Q_i, P_i\}$ with
  - $Q_i = \{g_i, h_i\}$
  - $P_i = \{g_{i+1}, \ldots, g_{i+k}, h_{i+1}, \ldots, h_{i+k+1}\}$

Choose $a > 0$ such that $P = (P_i)_{i \in [n]}$ NE with $C_i(P) = C_i(P_{-i}, Q_i)$.

$d = 2$, $n = 4$, $k = \lfloor \Phi_d \rfloor = \lfloor 2.148 \rfloor$
Price of Anarchy vs. Price of Stability

Price of Anarchy:

- assumes that \textit{worst-case} NE is reached
- we might be able to guide the players to a \textit{good} NE
Price of Anarchy vs. Price of Stability

Price of Anarchy:
- assumes that worst-case NE is reached
- we might be able to guide the players to a good NE

Price of Stability
- optimistic approach
- What is the best we can hope for in a NE?
- Much more accurate for instances with unique NE.
Price of Anarchy vs. Price of Stability

Price of Anarchy:
- assumes that worst-case NE is reached
- we might be able to guide the players to a good NE

Price of Stability
- optimistic approach
- What is the best we can hope for in a NE?
- Much more accurate for instances with unique NE.

Definition: Price of Stability

For a game $G$:
$$\text{PoS}(G) = \min_{s \text{ is NE}} \frac{\text{SC}(s)}{\text{OPT}}$$

For a class of games $\mathcal{G}$:
$$\text{PoS}(\mathcal{G}) = \sup_{G \in \mathcal{G}} \text{PoS}(G)$$
(1) **Analytical simple classes** of cost functions
⇒ **exact formula for PoS.**
(1) **Analytical simple classes** of cost functions
   ⇒ **exact formula for PoS.**
   - linear (PoS ≈ 1.577) [CHRISTODOULOU, KOUTSOUPIAS, ESA’05]
   - bounded degree polynomials [CHRISTODOULOU, GAIRING, ICALP’13]
(1) **Analytical simple classes** of cost functions
   ⇒ exact formula for PoS.
   ▶ linear (PoS ≈ 1.577)  
   [CHRISTODOULOU, KOUTSOUPIAS, ESA’05]
   [CARAGIANNIS ET AL., ICALP’06]
   ▶ bounded degree polynomials  
   [CHRISTODOULOU, GAIRING, ICALP’13]

(2) For **every set** of allowable cost functions
   ⇒ **recipe** for computing PoS.
Price of Stability: State of the Art

(1) **Analytical simple classes** of cost functions
⇒ **exact formula for PoS.**
  - linear (PoS ≈ 1.577)  
    - [Christodoulou, Koutsoupias, ESA’05]
    - [Caragiannis et al., ICALP’06]
  - bounded degree polynomials  
    - [Christodoulou, Gairing, ICALP’13]

(2) For every set of allowable cost functions
⇒ **recipe** for computing PoS.

???
Price of Stability: State of the Art

(1) **Analytical simple classes** of cost functions
   ⇒ **exact formula for PoS.**
   ▶ linear (PoS ≈ 1.577) [Christodoulou, Koutsoupias, ESA’05]
     [Caragiannis et al., ICALP’06]
   ▶ bounded degree polynomials [Christodoulou, Gairing, ICALP’13]

(2) For every set of allowable cost functions
   ⇒ **recipe** for computing PoS.

(3) Understanding of game complexity required for worst-case PoS to be realized.
   ▶ Ideally independent of cost functions.
   ▶ e.g. symmetric strategy sets, singleton strategy sets

???
Price of Stability: State of the Art

(1) **Analytical simple classes** of cost functions
⇒ **exact formula** for PoS.
  ▶ linear (PoS ≈ 1.577)  
    [Christodoulou, Koutsoupias, ESA’05]
    [Caragiannis et al., ICALP’06]
  ▶ bounded degree polynomials  
    [Christodoulou, Gairing, ICALP’13]

(2) **For every set** of allowable cost functions
⇒ **recipe** for computing PoS.

(3) **Understanding of game complexity** required for worst-case PoS to be realized.
  ▶ Ideally independent of cost functions.
  ▶ e.g. symmetric strategy sets, singleton strategy sets

This is still a very open field.
Price of Stability: Why is this harder?

- PoA methodology bounds cost of any NE
- PoS needs to capture the worst-case instance of the best NE
Price of Stability: Why is this harder?

- PoA methodology bounds cost of any NE
- PoS needs to capture the worst-case instance of the best NE
- useful characterisation of best-case NE is missing
- not straightforward to transfer techniques from PoA
Price of Stability: Why is this harder?

- PoA methodology bounds cost of any NE
- PoS needs to capture the worst-case instance of the best NE
- useful characterisation of best-case NE is missing
- not straightforward to transfer techniques from PoA

Approach to bound PoS

1) Define a restricted subset $R$ of NE
2) Find PoA w.r.t. NE that belong to $R$
Price of Stability: Why is this harder?

- PoA methodology bounds cost of any NE
- PoS needs to capture the worst-case instance of the best NE
- useful characterisation of best-case NE is missing
- not straightforward to transfer techniques from PoA

Approach to bound PoS

1) Define a restricted subset $R$ of NE
2) Find PoA w.r.t. NE that belong to $R$

New challenges

- What is a good choice for $R$?
- How can we incorporate the description of $R$ in the PoA methodology?
Potential Games and Price of Stability

Exact Potential Games:

- All games that admit a potential function $\Phi$, s.t. for all outcomes $s$, all player $i$, and all alternative strategies $s'_i$,

$$C_i(s'_i, s_{-i}) - C_i(s) = \Phi(s'_i, s_{-i}) - \Phi(s).$$

- Every congestion game is an exact potential game. [ROSENTHAL, 1973]
- For every exact potential game there exists a congestion game having the same potential function. [MÖNDER, S¸APLEY, 1996]
Potential Games and Price of Stability

Exact Potential Games:

- All games that admit a potential function \( \Phi \), s.t. for all outcomes \( s \), all player \( i \), and all alternative strategies \( s'_i \),

\[
C_i(s'_i, s_{-i}) - C_i(s) = \Phi(s'_i, s_{-i}) - \Phi(s).
\]

- Every congestion game is an exact potential game.  
  [ROSENTHAL, 1973]

- For every exact potential game there exists a congestion game having the same potential function.  
  [MONDERER, SHAPLEY, 1996]
Potential Games and Price of Stability

**Theorem**

Suppose that we have a potential game with potential function $\Phi$, and assume that for any outcome $s$, we have

$$\frac{SC(s)}{A} \leq \Phi(s) \leq B \cdot SC(s)$$

for some constants $A, B \geq 0$. Then the **price of stability** is at most $A \cdot B$. 
Potential Games and Price of Stability

**Theorem**
Suppose that we have a potential game with potential function $\Phi$, and assume that for any outcome $s$, we have

$$\frac{SC(s)}{A} \leq \Phi(s) \leq B \cdot SC(s)$$

for some constants $A, B \geq 0$. Then the price of stability is at most $A \cdot B$.

**Corollary**
Let $\mathcal{G}$ be the class of unweighted congestion games with polynomial cost functions of maximum degree $d$. Then,

$$\text{PoS}(\mathcal{G}) \leq d + 1.$$
Potential Games and Price of Stability

**Theorem**

Suppose that we have a potential game with potential function $\Phi$, and assume that for any outcome $s$, we have

$$\frac{SC(s)}{A} \leq \Phi(s) \leq B \cdot SC(s)$$

for some constants $A, B \geq 0$. Then the price of stability is at most $A \cdot B$.

**Corollary**

Let $\mathcal{G}$ be the class of unweighted congestion games with polynomial cost functions of maximum degree $d$. Then,

$$\text{PoS}(\mathcal{G}) \leq d + 1$$
PoS for polynomial (unweighted) congestion games

Theorem [CHRISTODOULOU, GAIRING, 2013]

For polynomial congestion games with cost functions from $C_d$ we have

$$\text{PoS} = \max_{r > 1} \frac{(2^d d + 2^d - 1) \cdot r^{d+1} - (d + 1) \cdot r^d + 1}{(2^d + d - 1) \cdot r^{d+1} - (d + 1) \cdot r^d + 2^d d - d + 1}.$$
Theorem [Christodoulou, Gairing, 2013]

For polynomial congestion games with cost functions from $C_d$ we have

$$\text{PoS} = \max_{r > 1} \frac{(2^d d + 2^d - 1) \cdot r^{d+1} - (d + 1) \cdot r^d + 1}{(2^d + d - 1) \cdot r^{d+1} - (d + 1) \cdot r^d + 2^d d - d + 1}.$$ 

- $d = 1$:
  $$\max_{r > 1} \frac{3 r^2 - 2 r + 1}{2 r^2 - 2 r + 2} = 1 + \frac{\sqrt{3}}{3} \approx 1.577$$
- $d = 2$:
  $$\max_{r > 1} \frac{11 r^3 - 3 r^2 + 1}{5 r^3 - 3 r^2 + 7} \approx 2.361$$
Theorem [Christodoulou, Gairing, 2013]

For polynomial congestion games with cost functions from $C_d$ we have

$$\text{PoS} = \max_{r > 1} \frac{(2^d d + 2^d - 1) \cdot r^{d+1} - (d + 1) \cdot r^d + 1}{(2^d + d - 1) \cdot r^{d+1} - (d + 1) \cdot r^d + 2^d d - d + 1}.$$  

- **$d = 1$:**
  $$\max_{r > 1} \frac{3 r^2 - 2 r + 1}{2 r^2 - 2 r + 2} = 1 + \frac{\sqrt{3}}{3} \approx 1.577$$

- **$d = 2$:**
  $$\max_{r > 1} \frac{11 r^3 - 3 r^2 + 1}{5 r^3 - 3 r^2 + 7} \approx 2.361$$

<table>
<thead>
<tr>
<th>$d$</th>
<th>PoS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.577</td>
</tr>
<tr>
<td>2</td>
<td>2.361</td>
</tr>
<tr>
<td>3</td>
<td>3.321</td>
</tr>
<tr>
<td>4</td>
<td>4.398</td>
</tr>
<tr>
<td>5</td>
<td>5.525</td>
</tr>
<tr>
<td>6</td>
<td>6.656</td>
</tr>
<tr>
<td>7</td>
<td>7.765</td>
</tr>
<tr>
<td>8</td>
<td>8.847</td>
</tr>
</tbody>
</table>
Upper bound high level proof idea:

- Consider NE $s$ with $\Phi(s) \leq \Phi(s^*)$

  \[ \Rightarrow \quad \text{SC}(s) \leq \text{SC}(s) + \Phi(s^*) - \Phi(s) \quad (1) \]
Upper bound high level proof idea:

- Consider NE $s$ with $\Phi(s) \leq \Phi(s^*)$

  \[ \Rightarrow \quad \text{SC}(s) \leq \text{SC}(s) + \Phi(s^*) - \Phi(s) \quad (1) \]

- Since $s$ is a NE

  \[ \text{SC}(s) = \sum_i C_i(s) \leq \sum_i C_i(s_{-i}, s_i^*) \quad (2) \]
Upper bound high level proof idea:

- Consider NE $s$ with $\Phi(s) \leq \Phi(s^*)$

  \[
  \Rightarrow \quad SC(s) \leq SC(s) + \Phi(s^*) - \Phi(s)
  \tag{1}
  \]

- Since $s$ is a NE

  \[
  SC(s) = \sum_i C_i(s) \leq \sum_i C_i(s_{-i}, s^*_i)
  \tag{2}
  \]

- Use linear combination $(1 - \nu) \cdot (1) + \nu \cdot (2)$ of the above and apply smoothness techniques.
Upper Bound: Key Insights

- Suffices to show local smoothness; i.e. \( \forall x, x^* \in \mathbb{N} \) and \( c \in C_d \):

\[
f(x, x^*, c, \nu) \leq \mu \cdot x \cdot c(x) + \lambda \cdot x^* \cdot c(x^*)
\]

- Sufficient to consider \( c(x) = x^d \).

- Tight constraints \((x, x^*)\) are (0, 1), (1, 1) and \((k \cdot r, k)\) for \( k \to \infty \).

- \( \lambda, \mu \) and \( \nu \) can be determined (in terms of \( r \)) as the “solution” of those 3 constraints.

- The hard part is to show that all other constraints are satisfied.
  - Without determining roots of high order polynomials.
Price of Stability: Structure of Lower Bound

Here: \( n = 5 \)

All cost functions of the form:

\[
c_e(x) = \alpha_e \cdot x^d
\]

\[
T_i = \frac{(k+i)^d - (k+i-1)^d}{2^{2d}-1}
\]

\( \leftarrow k \approx \frac{n}{r} \) additional players
Price of Stability: Structure of Lower Bound

Here: \( n = 5 \)

All cost functions of the form:

\[ c_e(x) = \alpha_e \cdot x^d \]

\[ T_i = \frac{(k+i)^d - (k+i-1)^d}{2^{2d-1}} \]

\( \leftarrow k \approx \frac{n}{i} \) additional players
Price of Stability: Structure of Lower Bound

Here: $n = 5$

All cost functions of the form:

$$c_e(x) = \alpha_e \cdot x^d$$

$$T_i = \frac{(k+i)^d - (k+i-1)^d}{2^{2d-1}}$$

$\leftarrow k \approx \frac{n}{i}$ additional players
Here: $n = 5$

All cost functions of the form:

$$c_e(x) = \alpha_e \cdot x^d$$

$$T_i = \frac{(k+i)^d - (k+i-1)^d}{2^{2^d-1}}$$
Price of Stability: Structure of Lower Bound

Here: $n = 5$

All cost functions of the form:

$$ c_e(x) = \alpha_e \cdot x^d $$

$$ T_i = \frac{(k+i)^d - (k+i-1)^d}{2^{2d-1}} $$

Unique NE

$k \approx \frac{n}{7}$ additional players
Here: $n = 5$

All cost functions of the form:

$$c_e(x) = \alpha_e \cdot x^d$$

$$T_i = \frac{(k+i)^d - (k+i-1)^d}{2^{2d} - 1}$$
PoS for *weighted* congestion games

- All PoS results presented are for unweighted players.
- How about the weighted case?

\[ \tilde{\Phi}(s) = \sum_{e \in E} x_e(s)c_e(x_e(s)) \leq SC(s) \]

\[ \Rightarrow SC(s) \leq \tilde{\Phi}(s) \leq 2 \cdot SC(s) \]

\[ \Rightarrow PoS(C_1) \leq 2 \]
PoS for *weighted* congestion games

- All PoS results presented are for unweighted players.
- How about the weighted case?
- Lower Bounds?

\[
\Phi(s) = \sum_{e \in E} x_e(s) \cdot c_e(x_e(s))
\]

\[
\Phi(s) \leq \sum_{i \in N} w_i \cdot x_i \cdot \sum_{e \in s_i} c_e(w_i) \leq SC(s) + \sum_{i \in N} w_i \cdot \sum_{e \in s_i} c_e(w_i) \leq 2 \cdot \Phi(s).
\]

\[\Rightarrow \text{PoS}(C_1) \leq 2\]
PoS for weighted congestion games

- All PoS results presented are for unweighted players.
- How about the weighted case?
- Lower Bounds?
  - Those from unweighted case hold also here.
  - No better lower bounds known.
PoS for **weighted** congestion games

- All PoS results presented are for unweighted players.

- **How about the weighted case?**

- **Lower Bounds?**
  - Those from unweighted case hold also here.
  - No better lower bounds known.

- **Upper Bounds?**

\[
\Phi^*(s) = \sum_{e \in E} x_e(s) \cdot c_e(x_e(s)) \leq SC(s) + \sum_{i \in N} w_i \cdot \sum_{e \in s_i} c_e(w_i) \leq 2 \cdot SC(s).
\]
PoS for *weighted* congestion games

- All PoS results presented are for unweighted players.
- How about the weighted case?
- Lower Bounds?
  - Those from unweighted case hold also here.
  - No better lower bounds known.
- Upper Bounds?
  - Except for linear case PoA upper bounds are the best known.
PoS for weighted congestion games

- All PoS results presented are for unweighted players.
- How about the weighted case?
- Lower Bounds?
  - Those from unweighted case hold also here.
  - No better lower bounds known.
- Upper Bounds?
  - Except for linear case PoA upper bounds are the best known.
  - for linear, recall potential
  \[
  \tilde{\Phi}(s) = \sum_{e \in E} x_e(s) \cdot c_e(x_e(s)) + \sum_{i \in N} w_i \cdot \sum_{e \in s_i} c_e(w_i).
  \]
  \[
  \tilde{\Phi}(s) = \text{SC}(s) + \sum_{i \in N} w_i \cdot \sum_{e \in s_i} c_e(w_i) \leq \text{SC}(s).
  \]
PoS for \textit{weighted} congestion games

- All PoS results presented are for unweighted players.

- How about the weighted case?

- Lower Bounds?
  - Those from unweighted case hold also here.
  - No better lower bounds known.

- Upper Bounds?
  - Except for linear case PoA upper bounds are the best known.
  - for linear, recall potential [Fotakis, Kontogiannis, Spirakis, 2004]

\[
\tilde{\Phi}(s) = \sum_{e \in E} x_e(s) \cdot c_e(x_e(s)) + \sum_{i \in N} w_i \cdot \sum_{e \in s_i} c_e(w_i) .
\]

\[= \text{SC}(s) \leq \text{SC}(s) \leq \text{PoS}(C_1) \leq 2\]
Conclusion and open problems

Take Home Points

- There is a strong theory on the PoA in congestion games
  - Exact values for polynomial cost functions.
  - Recipe for general functions.
- PoS has been studied to a much lesser extend.

Interesting open problems:

- PoS for general cost functions
- PoS for weighted players
  - main challenge: no potential function.
- PoA for instances with dominating strategy equilibrium
  - We showed separation. (For $d = 2$ smaller than PoS.)

Thanks. Any questions?
Conclusion and open problems

Take Home Points

▶ There is a strong theory on the PoA in congestion games
  ▶ Exact values for polynomial cost functions.
  ▶ Recipe for general functions.
▶ PoS has been studied to a much lesser extent.

Interesting open problems:
▶ PoS for general cost functions
▶ PoS for weighted players
  ▶ main challenge: no potential function.
▶ PoA for instances with dominating strategy equilibrium
  ▶ We showed separation. (For \( d = 2 \) smaller than PoS.)
## Conclusion and open problems

### Take Home Points

- There is a strong theory on the PoA in congestion games
  - Exact values for polynomial cost functions.
  - Recipe for general functions.
- PoS has been studied to a much lesser extent.

### Interesting open problems:

- PoS for general cost functions
- PoS for weighted players
  - main challenge: no potential function.
- PoA for instances with dominating strategy equilibrium
  - We showed separation. (For $d = 2$ smaller than PoS.)

Thanks. Any questions?