Congestion Games with Shifted Latency Functions

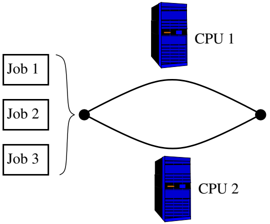
Igal Milchtaich[1], Burkhard Monien[2], **Karsten Tiemann**[2]

[1] Bar-Ilan University Ramat Gan, Israel. [2] University of Paderborn, Germany.

AEOLUS Workshop on Scheduling (March 9, 2007)

Outline

Self-Organizing Systems & Games with Selfish Agents



Situation in many dynamic self-organizing systems

- There is no central control in the system.
- Each autonomous agent tries to improve its private cost.
- The private cost of an agent depends on the behavior of all agents.

Example: Traffic System (1/2)

- There is no central control that steers all cars.
- Autonomous agent: Each car driver is an autonomous agent.
- Private cost: Each car driver wants to minimize its time of travel.
- The traveling time of a car driver also depends on the decision of other car drivers.

Example: Traffic System (2/2)

- Question: Outcome?
  What are the decisions of the car drivers?
  Which roads do they select for their trip?
- Answer: Stable state
  Each car driver decides for a route that minimizes his travel time.

Strategic Games and Strategies

Game $\mathcal{G} = (n, S_1, \dots S_n, \text{PC}_1, \dots, \text{PC}_n)$ in normal form

A game $\mathcal{G}$ is described by $\mathcal{G} = (n, S_1, \dots S_n, \text{PC}_1, \dots, \text{PC}_n)$ where

- n defines how many players 1,2,...,n are present,
- S_i is the set of strategies of player i,
- $\text{PC}_i : S_1 \times \dots \times S_n \to \mathbb{R}$ is the private cost function of player i.

Strategies and strategy profiles

- $s_i \in S_i$ is a (pure) strategy of player i
- $(s_1, \dots, s_n) \in S_1 \times \dots \times S_n$ is a (pure) strategy profile

Nash Equilibria

(Pure) Nash equilibrium

A (pure) strategy profile $(s_1, \ldots, s_n)$ is a (pure) Nash equilibrium if, for each player i and $\forall s_i' \in S_i$:

$$\mathrm{PC}_i(s_1, \ldots, s_i, \ldots, s_n) \leq \mathrm{PC}_i(s_1, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n).$$

Existence of pure Nash equilibria

In general games, pure Nash equilibria may not exist.

Open problem

Which games possess a pure Nash equilibrium?

Nash Equilibria

(Pure) Nash equilibrium

A (pure) strategy profile $(s_1, \ldots, s_n)$ is a (pure) Nash equilibrium if, for each player i and $\forall s_i' \in S_i$:

$$\mathrm{PC}_i(s_1, \ldots, s_i, \ldots, s_n) \leq \mathrm{PC}_i(s_1, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n).$$

Existence of pure Nash equilibria

In general games, pure Nash equilibria may not exist.

Open problem

Which games possess a pure Nash equilibrium?

Nash Equilibria

(Pure) Nash equilibrium

A (pure) strategy profile $(s_1, \ldots, s_n)$ is a (pure) Nash equilibrium if, for each player i and $\forall s_i' \in S_i$:

$$\mathrm{PC}_i(s_1, \ldots, s_i, \ldots, s_n) \leq \mathrm{PC}_i(s_1, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_n).$$

Existence of pure Nash equilibria

In general games, pure Nash equilibria may not exist.

Open problem

Which games possess a pure Nash equilibrium?

(Greedy) Selfish Steps

(Greedy) selfish steps

Player i unilaterally deviates to another strategy. This is

- a selfish step if i decreases its private cost,
- a greedy selfish step if the new strategy minimizes i's private cost.

Finite improvement and finite best-reply property

A game possesses the

- finite improvement property if any sequence of selfish steps is finite.
- finite best-reply property if any sequence of greedy selfish steps is finite.

(Greedy) Selfish Steps

(Greedy) selfish steps

Player i unilaterally deviates to another strategy. This is

- a selfish step if i decreases its private cost,
- a greedy selfish step if the new strategy minimizes i's private cost.

Finite improvement and finite best-reply property

A game possesses the

- finite improvement property if any sequence of selfish steps is finite.
- finite best-reply property if any sequence of greedy selfish steps is finite.

Selfish Steps and the Existence of Nash Equilibria

Helpful fact 1

Game $\mathcal{G}$ possesses the finite improvement property.

$\Rightarrow$

$\mathcal{G}$ possesses the finite best-reply property.

Helpful fact 2

Game $\mathcal{G}$ possesses the finite best-reply property.

$\Rightarrow$

$\mathcal{G}$ possesses at least one pure Nash equilibrium.

Selfish Steps and the Existence of Nash Equilibria

Helpful fact 1

Game $\mathcal{G}$ possesses the finite improvement property.

$$\Rightarrow$$

$\mathcal{G}$ possesses the finite best-reply property.

Helpful fact 2

Game $\mathcal{G}$ possesses the finite best-reply property.

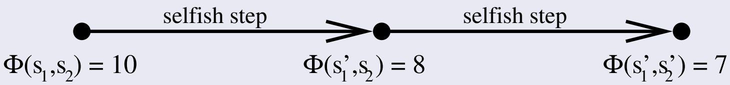
$$\Rightarrow$$

$\mathcal{G}$ possesses at least one pure Nash equilibrium.

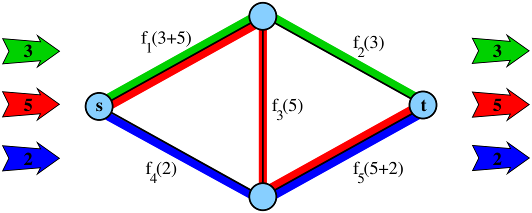
Finite Improvement Property

How to show the finite improvement property

- Define a function $\Phi : S_1 \times \ldots \times S_n \to \mathbb{R}$.
- Show that this function Φ decreases whenever a player does a selfish step.



Routing Games: Network Congestion Games



Example of a network congestion game

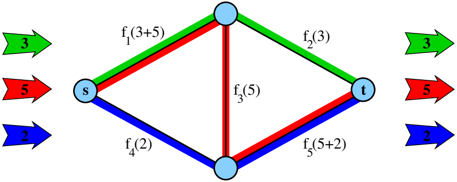
- There are 3 players of weight 3, 5, and 2.
- Each player selects as its strategy a path from *s* to *t*.
- A strategy profile defines a load on each edge.

Load and Private Cost

Load on resource e

$$\delta_e(s) = \sum_{i:e\in s_i} w_i$$

where w_i is player i's weight

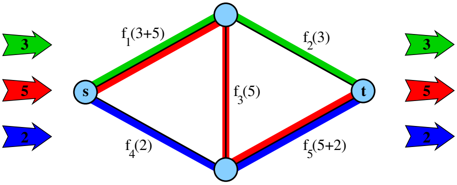


Load and Private Cost

Load on resource e

$$\delta_e(s) = \sum_{i: e \in s_i} w_i$$

where w_i is player i's weight



Private cost of player i

$$PC_i(s) = \sum_{e \in s_i} f_e(\delta_e(s))$$

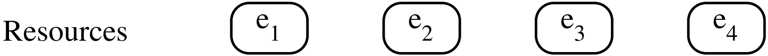
where f_e is the latency function of edge e

$PC_1 = f_1(8) + f_2(3)$

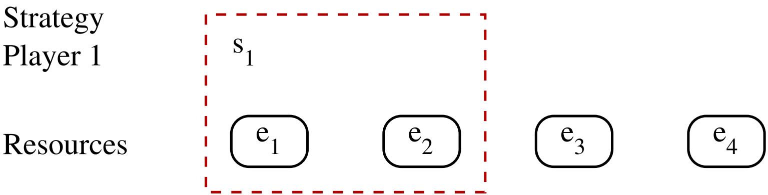
$PC_2 = f_1(8) + f_3(5) + f_5(7)$

$PC_3 = f_4(2) + f_5(7)$

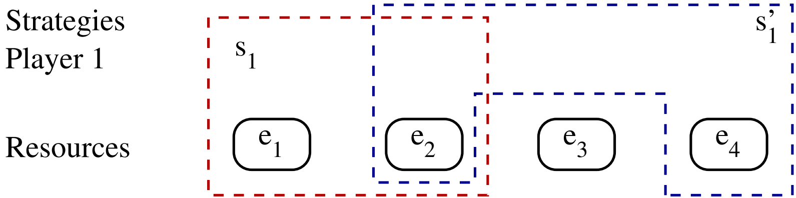
Congestion Games



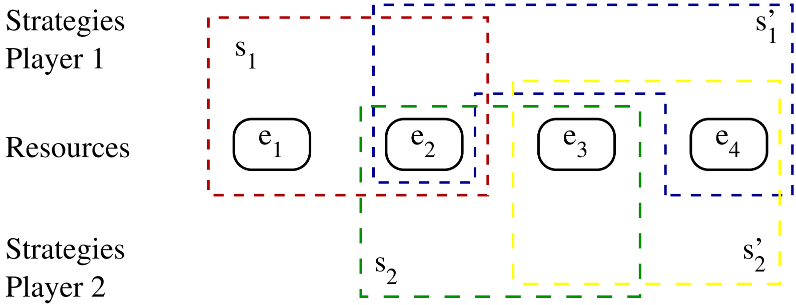
Congestion Games



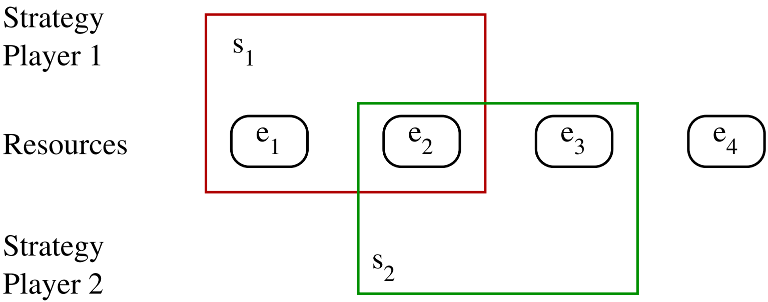
Congestion Games



Congestion Games



Congestion Games



$$PC_1 = f_{e_1}(w_1) + f_{e_2}(w_1 + w_2), \qquad PC_2 = f_{e_2}(w_1 + w_2) + f_{e_3}(w_2)$$

Player-specific Latency Functions

More latency functions per edge

- Congestion games:
  1 latency function per edge.
- Congestion games with player-specific latency functions:
  n latency functions per edge, one for each player.

Scenarios where player-specific functions are reasonable

- Players have different preferences or objectives.
- Players do not know the actual latency function.
  (Incomplete information.)

Player-specific Latency Functions

More latency functions per edge

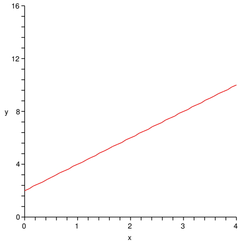
- Congestion games:
  1 latency function per edge.
- Congestion games with player-specific latency functions:
  n latency functions per edge, one for each player.

Scenarios where player-specific functions are reasonable

- Players have different preferences or objectives.
- Players do not know the actual latency function.
  (Incomplete information.)

Different Kinds of Latency Functions (1/3)

One latency function per edge



Private cost of player *i*

$$PC_i(s) = \sum_{e \in s_i} f_e(\delta_e(s))$$

where f_e is the latency function of edge e

[Rosenthal, 1973]

Player-specific latency functions

Private cost of player *i*

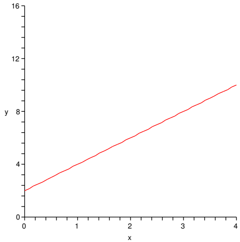
$$PC_i(s) = \sum_{e \in s_i} f_{ie}(\delta_e(s))$$

where f_{ie} is the latency function player i assigns to edge e

[Milchtaich, 1996]

Different Kinds of Latency Functions (1/3)

One latency function per edge



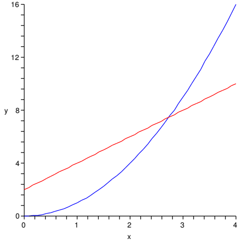
Private cost of player *i*

$$PC_i(s) = \sum_{e \in s_i} f_e(\delta_e(s))$$

where f_e is the latency function of edge e

[ROSENTHAL, 1973]

Player-specific latency functions



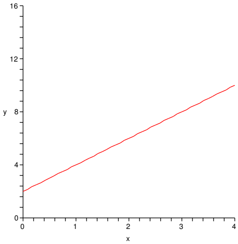
Private cost of player *i*

$$PC_i(s) = \sum_{e \in s_i} f_{ie}(\delta_e(s))$$

where f_{ie} is the latency function player i assigns to edge e

[MILCHTAICH, 1996]

Different Kinds of Latency Functions (2/3)



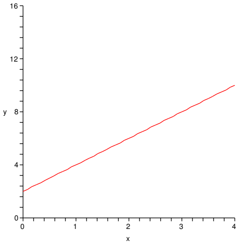
One latency function per edge:

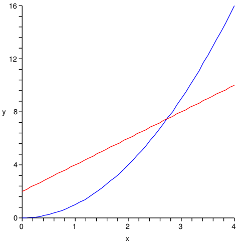
$2x + 2$

$2x + 2$

[Rosenthal, 1973]

Different Kinds of Latency Functions (2/3)





One latency function per edge:

$2x + 2$
$2x + 2$

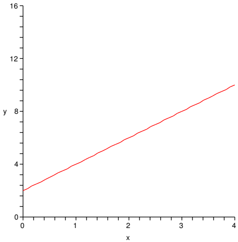
[Rosenthal, 1973]

Player-specific latency functions:

$2x + 2$
x^2

[Milchtaich, 1996]

Different Kinds of Latency Functions (2/3)

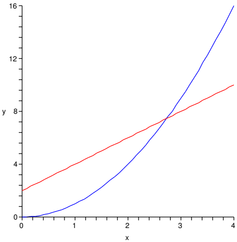


One latency function per edge:

$2x + 2$

$2x + 2$

[ROSENTHAL, 1973]

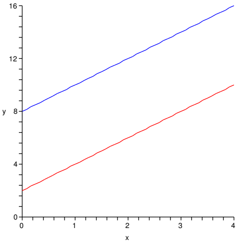


Player-specific latency functions:

$2x + 2$

x^2

[MILCHTAICH, 1996]



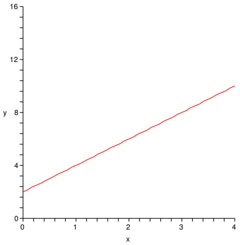
Shifted latency functions:

$2x + 2$

$2x + 8$

Considered here.

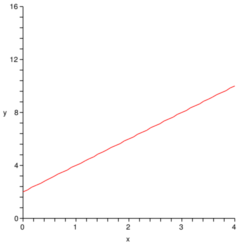
Different Kinds of Latency Functions (3/3)

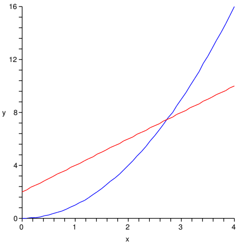


Congestion games

[ROSENTHAL, 1973]

Different Kinds of Latency Functions (3/3)





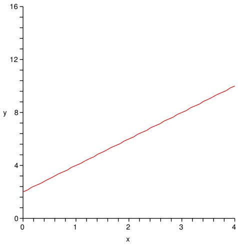
Congestion games

Congestion games with player-specific latency functions

[ROSENTHAL, 1973]

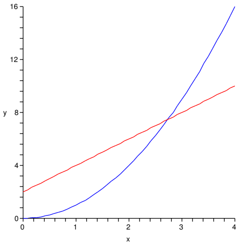
[MILCHTAICH, 1996]

Different Kinds of Latency Functions (3/3)



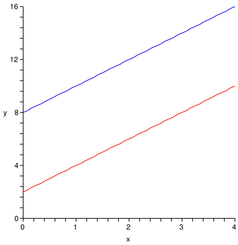
Congestion games

[ROSENTHAL, 1973]



Congestion games with player-specific latency functions

[MILCHTAICH, 1996]



Congestion games with shifted latency functions

Considered here.

Shifted Latency Functions

Shifted latency functions

- For an edge e we have
  - one common non-decreasing latency function f_e and
  - non-negative player-specific constants $c_{1e}, c_{2e}, \ldots, c_{ne}$.
- The latency that player i assigns to edge e is given by $c_{ie} + f_e(x)$.

Shifted Latency Functions

Shifted latency functions

- For an edge e we have
  - one common non-decreasing latency function f_e and
  - non-negative player-specific constants $c_{1e}, c_{2e}, \ldots, c_{ne}$.
- The latency that player i assigns to edge e is given by $c_{ie} + f_e(x)$.

Unweighted Players: Finite Improvement Property

Existence of Nash equilibria for unweighted games

- We start with unweighted players, i.e., $w_1 = w_2 = \ldots = w_n = 1$.
- It is well-known that all unweighted congestion games possess the finite improvement property. The proof by [Rosenthal, 1973] uses this potential function:

$$\Phi(s) = \sum_{e \in E} \sum_{i=1}^{\delta_e(s)} f_e(i)$$

Unweighted Players: Finite Improvement Property

Existence of Nash equilibria for unweighted games

- We start with unweighted players, i.e., $w_1 = w_2 = \ldots = w_n = 1$.
- It is well-known that all unweighted congestion games possess the finite improvement property. The proof by [Rosenthal, 1973] uses this potential function:

$$\Phi(s) = \sum_{e \in E} \sum_{i=1}^{\delta_e(s)} f_e(i)$$

Unweighted Players, Shifted Latency Functions: Finite Impr. Pr.

Theorem (Facchini, van Megen, Borm, Tijs, 1997)

All congestion games with

- *unweighted players and*
- *shifted latency functions*

possess the finite improvement property.

$$\Phi(s) = \sum_{e \in E} \sum_{i=1}^{\delta_e(s)} f_e(i) + \sum_{i=1}^{n} \sum_{e \in s_i} c_{ie}$$

Unweighted Players, Shifted Latency Functions: Finite Impr. Pr.

Theorem (Facchini, van Megen, Borm, Tijs, 1997)

All congestion games with

- *unweighted players and*
- *shifted latency functions*

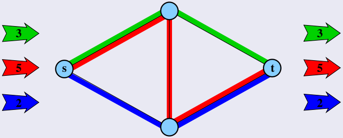
possess the finite improvement property.

$$\Phi(s) = \sum_{e \in E} \sum_{i=1}^{\delta_e(s)} f_e(i) + \sum_{i=1}^{n} \sum_{e \in s_i} c_{ie}$$

Unweighted Players: Nash Computation

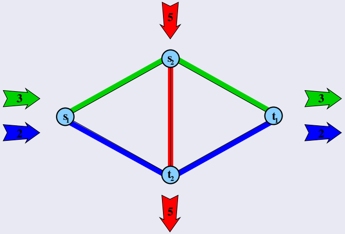
Computation of Nash equilibria for network congestion games

Symmetric games



Polynomial time

Asymmetric games



PLS-complete

[FABRIKANT, PAPADIMITRIOU, TALWAR, 2004]

Unw. Players, Shifted Latency Functions: Nash Computation

Theorem

It is PLS-complete to find a pure Nash equilibrium in a congestion game with

- unweighted players and
- shifted latency functions on a
- symmetric network.

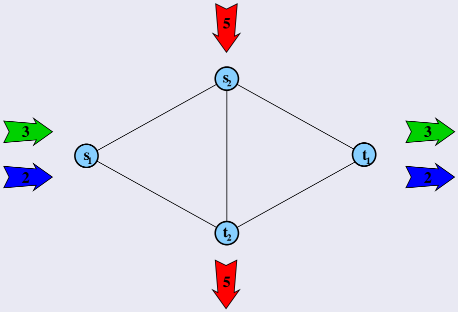
Unw. Players, Shifted Latency Functions: Nash Computation

Reduction: Computation of Nash equilibria

Nash computation unw. asymmetric network congestion game

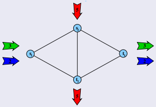
$\leq$

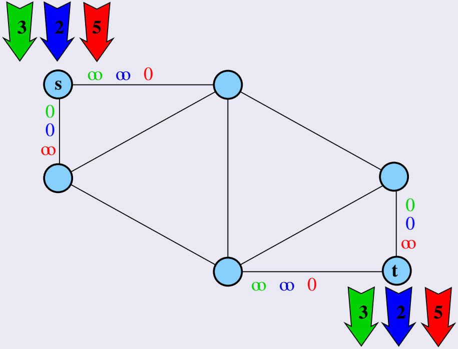
Nash computation unw. symmetric network congestion game with shifted latency functions



Unw. Players, Shifted Latency Functions: Nash Computation

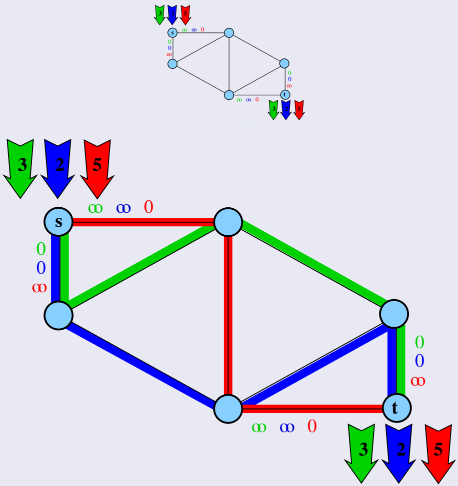
Reduction: Computation of Nash equilibria





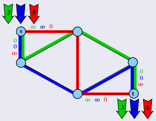
Unw. Players, Shifted Latency Functions: Nash Computation

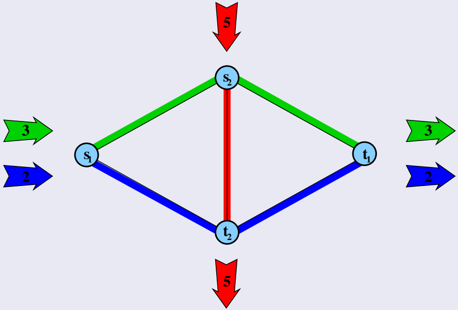
Reduction: Computation of Nash equilibria



Unw. Players, Shifted Latency Functions: Nash Computation

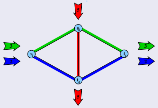
Reduction: Computation of Nash equilibria





Unw. Players, Shifted Latency Functions: Nash Computation

Reduction: Computation of Nash equilibria



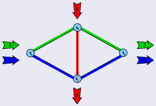
Nash computation for
unw. asymmetric network congestion game

$$\leq$$

Nash computation for
unw. symmetric network congestion game
with shifted latency functions

Unw. Players, Shifted Latency Functions: Nash Computation

Reduction: Computation of Nash equilibria



Nash computation for
unw. asymmetric network congestion game is PLS-complete

implies

Nash computation for
unw. symmetric network congestion game
with shifted latency functions is PLS-complete

Unw. Players, Shifted Latency Functions: Summary

Summary: Unweighted players

	Traditional Congestion Games	Congestion games with shifted latency functions
Finite improvement property	Yes [Rosenthal, 1973]	Yes
Nash polynomial time computation symmetric networks	Yes [Fabrikant et al., 2004]	No (unless all *PLS*-problems are solvable in polytime)

Unw. Players, Shifted Latency Functions: Summary

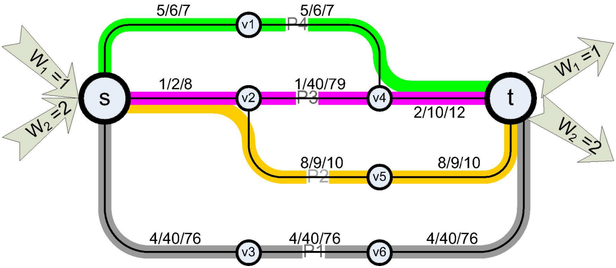
Summary: Unweighted players

	Traditional Congestion Games	Congestion games with shifted latency functions
Finite improvement property	Yes [Rosenthal, 1973]	Yes
Nash polynomial time computation symmetric networks	Yes [Fabrikant et al., 2004]	No (unless all *PLS*-problems are solvable in polytime)

Existence of Pure Nash Equilibria: Negative Result

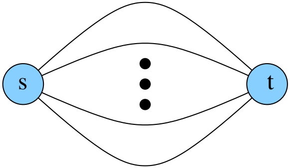
Theorem (LIBMAN & ORDA 2001, FOTAKIS, KONTOGIANNIS, SPIRAKIS 2004)

*There is a weighted network congestion game for that **no** pure Nash equilibrium exists.*



Cycle of selfish steps: (P3,P2), (P3,P4), (P1,P4), (P1,P2), (P3,P2)

Existence of Pure Nash Equilibria: Positive Result



Theorem (e.g. Fotakis et al. 2002)

Each congestion game with

- *weighted players and*
- *non-decreasing latency functions on*
- *parallel links*

possesses the finite improvement property.

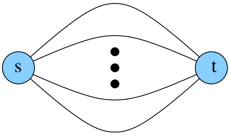
Weighted Players & Parallel Links

Question

Do all congestion games with

- weighted players and
- shifted latency functions
- on parallel links

possess the finite improvement property?



Answer

No, in general. Yes, if the latency functions are linear.

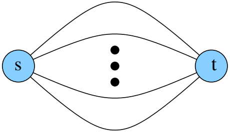
Weighted Players & Parallel Links

Question

Do all congestion games with

- weighted players and
- shifted latency functions
- on parallel links

possess the finite improvement property?



Answer

No, in general. Yes, if the latency functions are linear.

General Latency Functions: No Finite B.-rep. Property

Theorem

There is a congestion game with

- weighted players and
- shifted latency functions
- on parallel links

that does **not** possess the finite best-reply property.

General Latency Functions: No Finite B.-rep. Property

Instance

3 players $w_1 = 1, w_2 = 2$, $w_3 = 1$ on 3 parallel links.

Player i's latency for link j is $c_{ij} + f_j(x)$ where:

c_{ij}	Link 1	Link 2	Link 3
Player 1	15	2	99
Player 2	4	99	18
Player 3	99	3	20

Link j	$f_j(1)$	$f_j(2)$	$f_j(3)$
Link 1	13	27	31
Link 2	27	38	51
Link 3	11	12	20

Selfish step cycle

Cycle of selfish steps
$S_1 \to S_2 \to S_3 \to S_4 \to S_5 \to S_6 \to S_1$
where

	PC_1	PC_2	PC_3
S_1		35	31
S_2	46		30
S_3	40	31	
S_4		30	41
S_5	29		40
S_6	28	38	

General Latency Functions: No Finite B.-rep. Property

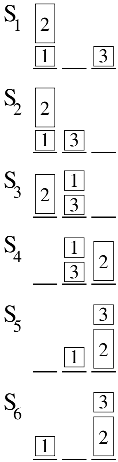
Instance

3 players $w_1 = 1$, $w_2 = 2$, $w_3 = 1$ on 3 parallel links.

Player i's latency for link j is $c_{ij} + f_j(x)$ where:

c_{ij}	Link 1	Link 2	Link 3
Player 1	15	2	99
Player 2	4	99	18
Player 3	99	3	20

Link j	$f_j(1)$	$f_j(2)$	$f_j(3)$
Link 1	13	27	31
Link 2	27	38	51
Link 3	11	12	20



Selfish step cycle

Cycle of selfish steps

$S_1 \to S_2 \to S_3 \to S_4 \to S_5 \to S_6 \to S_1$

where

	PC_1	PC_2	PC_3
S_1		35	31
S_2	46		30
S_3	40	31	
S_4		30	41
S_5	29		40
S_6	28	38	

General Latency Functions: No Finite B.-rep. Property

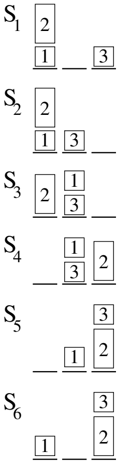
Instance

3 players $w_1 = 1$, $w_2 = 2$, $w_3 = 1$ on 3 parallel links.

Player i's latency for link j is $c_{ij} + f_j(x)$ where:

c_{ij}	Link 1	Link 2	Link 3
Player 1	15	2	99
Player 2	4	99	18
Player 3	99	3	20

Link j	$f_j(1)$	$f_j(2)$	$f_j(3)$
Link 1	13	27	31
Link 2	27	38	51
Link 3	11	12	20



Selfish step cycle

Cycle of selfish steps
$S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \rightarrow S_5 \rightarrow S_6 \rightarrow S_1$
where

	PC_1	PC_2	PC_3
S_1		35	31
S_2	46		30
S_3	40	31	
S_4		30	41
S_5	29		40
S_6	28	38	

Linear Latency Functions: Finite Impr. Property

Linear latency functions $f_j(x) = a_j \cdot x$

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

Theorem

All congestion games with

- weighted players and
- shifted linear latency functions

possess the finite improvement property.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot \sum_{e \in s_i} c_{ie} + \sum_{e \in E} a_e \cdot \left(\sum_{\substack{o: \\ s_o = e}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = e}} w_u \cdot w_v \right)$$

Linear Latency Functions: Finite Impr. Property

Linear latency functions $f_j(x) = a_j \cdot x$

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

Theorem

All congestion games with

- weighted players and
- shifted linear latency functions

possess the finite improvement property.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot \sum_{e \in s_i} c_{ie} + \sum_{e \in E} a_e \cdot \left(\sum_{\substack{o:\\ s_o = e}} {w_o}^2 + \sum_{\substack{\{u,v\}:\\ s_u = s_v = e}} w_u \cdot w_v \right)$$

Linear Latency Functions: Finite Impr. Property

Linear latency functions $f_j(x) = a_j \cdot x$

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

Theorem

All congestion games with

- weighted players and
- shifted linear latency functions

possess the finite improvement property.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot \sum_{e \in s_i} c_{ie} + \sum_{e \in E} a_e \cdot \left(\sum_{\substack{o: \\ s_o = e}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = e}} w_u \cdot w_v \right)$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o = j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.
$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned}
&\Psi(s) - \Psi(s') \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\
&= w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\
&> 0
\end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o = j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.

$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned} &\Psi(s) - \Psi(s') \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\ &= w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\ &> 0 \end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o:\\ s_o=j}} {w_o}^2 + \sum_{\substack{\{u,v\}:\\ s_u=s_v=j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.

$\text{PC}_k(s) \;=\; c_{kp} + a_p \cdot \delta_p(s) \;>\; c_{kq} + a_q \cdot \delta_q(s') \;=\; \text{PC}_k(s')$.

$$\begin{aligned} &\Psi(s) - \Psi(s') \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\ &= w_k \cdot (\text{PC}_k(s) - PC_k(s')) \\ &> 0 \end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o = j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.
$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned} & \Psi(s) - \Psi(s') \\ = \; & w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\ = \; & w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\ = \; & w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\ > \; & 0 \end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o = j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.
$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned} &\Psi(s) - \Psi(s') \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\ &= w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\ &> 0 \end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o:\\ s_o=j}} {w_o}^2 + \sum_{\substack{\{u,v\}:\\ s_u=s_v=j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.
$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned}
&\Psi(s) - \Psi(s') \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\
&= w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\
&> 0
\end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o=j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u=s_v=j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.

$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned} &\Psi(s) - \Psi(s') \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\ &= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\ &= w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\ &> 0 \end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o:\\ s_o=j}} {w_o}^2 + \sum_{\substack{\{u,v\}:\\ s_u=s_v=j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.
$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned}
&\Psi(s) - \Psi(s') \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\
&= w_k \cdot (PC_k(s) - PC_k(s')) \\
&> 0
\end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o = j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.

$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned}
&\Psi(s) - \Psi(s') \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\
&= w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\
&> 0
\end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o = j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.
$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned} & \Psi(s) - \Psi(s') \\ = \quad & w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\ = \quad & w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\ = \quad & w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\ > \quad & 0 \end{aligned}$$

Linear Latency Functions: Finite Impr. Property

Latency that player i assigns to link j is given by $c_{ij} + a_j \cdot x$.

$$\Psi(s) = \sum_{i=1}^{n} w_i \cdot c_{is_i} + \sum_{j=1}^{m} a_j \cdot \Big(\sum_{\substack{o: \\ s_o = j}} {w_o}^2 + \sum_{\substack{\{u,v\}: \\ s_u = s_v = j}} w_u \cdot w_v \Big)$$

Ψ decreases if player k does a selfish step $s \to s'$

Let $s = (\ldots, p, \ldots)$, $s' = (\ldots, q, \ldots)$.

$\mathrm{PC}_k(s) = c_{kp} + a_p \cdot \delta_p(s) > c_{kq} + a_q \cdot \delta_q(s') = \mathrm{PC}_k(s')$.

$$\begin{aligned}
&\Psi(s) - \Psi(s') \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot [w_k + \delta_p(s')] - a_q \cdot [w_k + \delta_q(s)]) \\
&= w_k \cdot (c_{kp} - c_{kq} + a_p \cdot \delta_p(s) - a_q \cdot \delta_q(s')) \\
&= w_k \cdot (\mathrm{PC}_k(s) - PC_k(s')) \\
&> 0
\end{aligned}$$

Weighted Players, Shifted Latency Functions: Summary

Summary: Weighted players

Finite improvement property	Traditional Congestion Games	Congestion games with shifted latency functions
non-decreasing latency functions, parallel links	Yes e.g. [Fotakis et al., 2002]	No
linear latency functions	Yes e.g. [Fotakis et al., 2004]	Yes

Weighted Players, Shifted Latency Functions: Summary

Summary: Weighted players

Finite improvement property	Traditional Congestion Games	Congestion games with shifted latency functions
non-decreasing latency functions, parallel links	Yes e.g. [Fotakis et al., 2002]	No
linear latency functions	Yes e.g. [Fotakis et al., 2004]	Yes

Thank you!

Your questions?