How Bad is Selfish Routing?

Tim Roughgarden and Éva Tardos

Presented by Brighten Godfrey

Game Theory

- Two or more players
- For each player, a set of strategies
- For each combination of played strategies, a payoff for each player

Blue player strategies (columns); Red player strategies (rows)

	Rock	Paper	Scissors
Rock	0, 0	0, 1	1, 0
Paper	1, 0	0, 0	0, 1
Scissors	0, 1	1, 0	0, 0

Nash Equilibrium

A choice of strategies such that
no player can improve by changing.

What's the N.E. in Rock-Paper-Scissors?

	Rock	Paper	Scissors
Rock	0, 0	0, 1	1, 0
Paper	1, 0	0, 0	0, 1
Scissors	0, 1	1, 0	0, 0

No (pure) N.E.!

What's the N.E. in Prisoner's Dilemma?

Blue prisoner (columns); Red prisoner (rows)

	Cooperate	Defect
Cooperate	-1, -1	-10, 0
Defect	0, -10	-5, -5

Price of Anarchy

[C. Papadimitriou, "Algorithms, games and the Internet", STOC 2001]

Assumes some global "cost" objective, e.g., social utility (sum of players' payoffs).

$$\text{Price of anarchy} = \frac{\text{worst Nash equilibria's cost}}{\text{optimal cost}}$$

Blue prisoner

Red prisoner	Cooperate	Defect
Cooperate	-1, -1	-10, 0
Defect	0, -10	-5, -5

Here, PoA = 10/2 = 5.

This week's papers in game theory terms

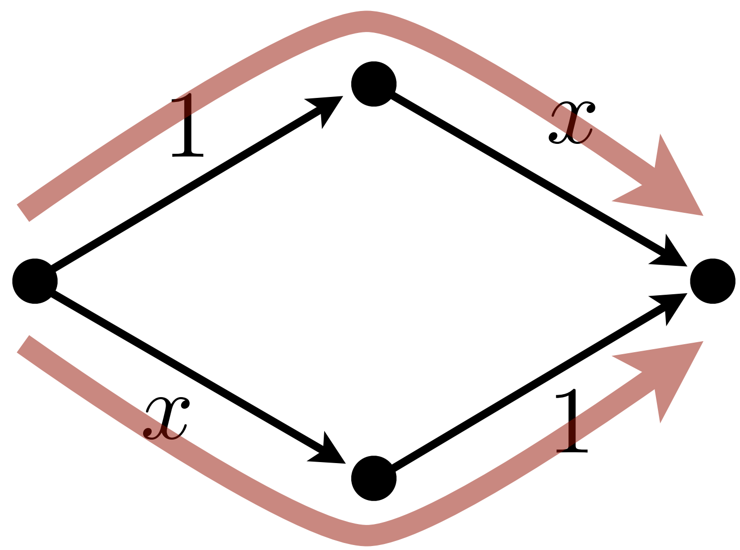
Stable Paths Problem (BGP convergence)	For the SPP game, in some instances no Nash equilibrium exists. NP-complete to decide whether one exists.
BGP-based mechanism for lowest-cost routing	Design a game so that the Nash equilibrium exists and is optimal (price of anarchy = 1).
How bad is selfish routing?	For the selfish routing game, the Nash equilibria are fairly close to optimal.
Selfish routing in Internet-like environments	In practice the Nash equilibria are extremely close to optimal.

How bad is selfish routing?

Demo!

The selfish routing game

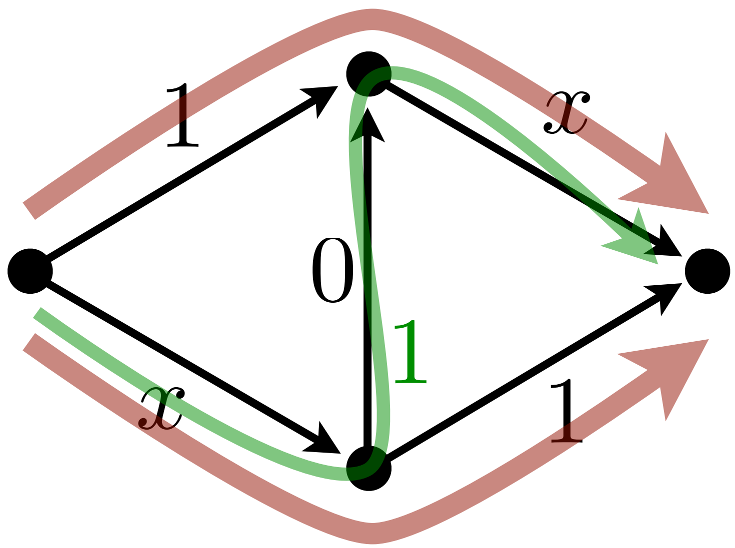
- Given graph, latency function on each edge specifying latency as function of total load on edge
- Path latency = sum of edge latencies
- Player strategy: pick a path on which to route
- Players selfishly pick paths with lowest latency
- For now assume many users, each with negligible load; total 1



Flow x = 0.5 on each path;
Total latency = 1.5

Example: Braess's paradox

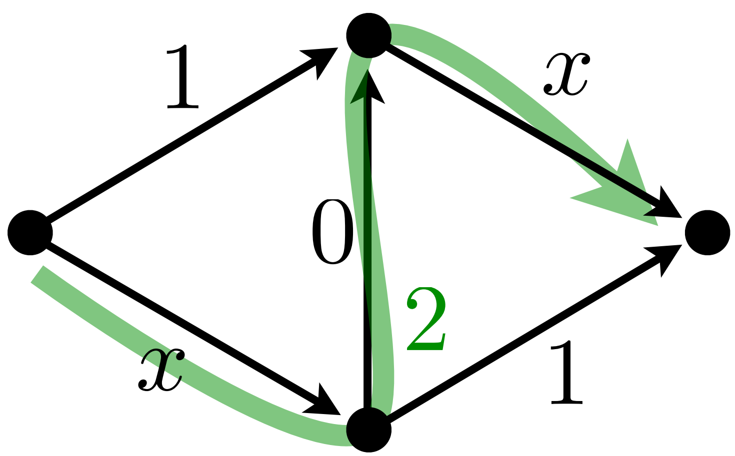
[D. Braess, 1968]



Green path is more attractive.
Everyone switches to it!

Example: Braess's paradox

[D. Braess, 1968]

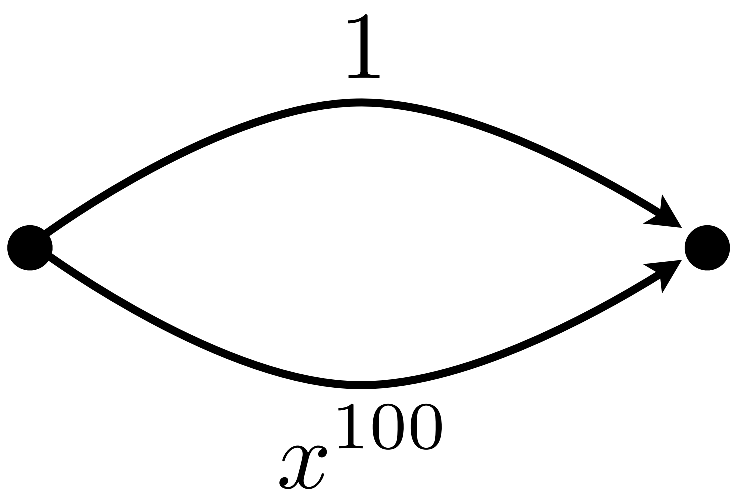


Nash equilibrium latency = 2

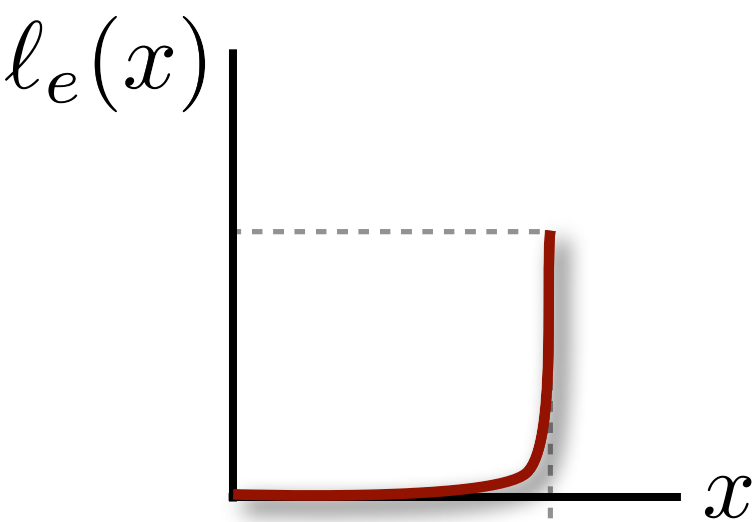
Optimal latency = 1.5

Price of Anarchy = 4/3

Example: arbitrarily bad



Optimal: nearly all flow on bottom; total latency near zero



Nash: all flow on bottom; total latency = 1

Results

- As we just saw, price of anarchy can be arbitrarily high
- But for linear latency functions: PoA <= 4/3
- For any latency function: Nash cost is at most optimal cost of 2x as much flow
- Extension to finitely many agents
    - Splittable flow: similar "2x" result
    - Unsplittable flow: can be very bad

Cost is at most that of routing 2x as much flow

$$f = \text{Nash equilibrium flow}$$

$$f^* = \text{Optimal flow with 2x as much traffic}$$

$$C(f) = \text{Total cost (latency) of flow}$$

Theorem: $C(f^*) \geq C(f).$

Proof outline

Define new "increased" latency functions, making f^* cost at most ( ) more.

$$C(f^*) \geq \tilde{C}(f^*) - C(f)$$

Increased latency is devised so that Nash equilibrium has minimum unit cost per flow...

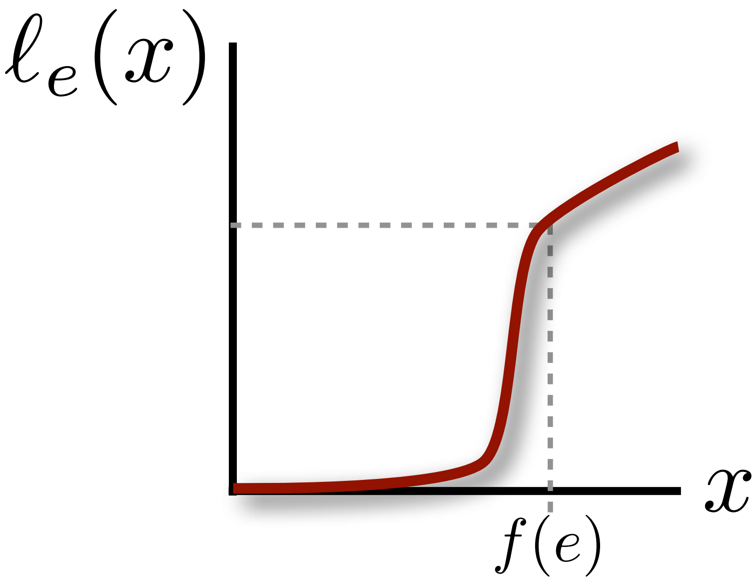
$$\geq 2\tilde{C}(f) - C(f)$$

...and that cost equals the Nash's cost under the original latency function.

$$= 2C(f) - C(f)$$

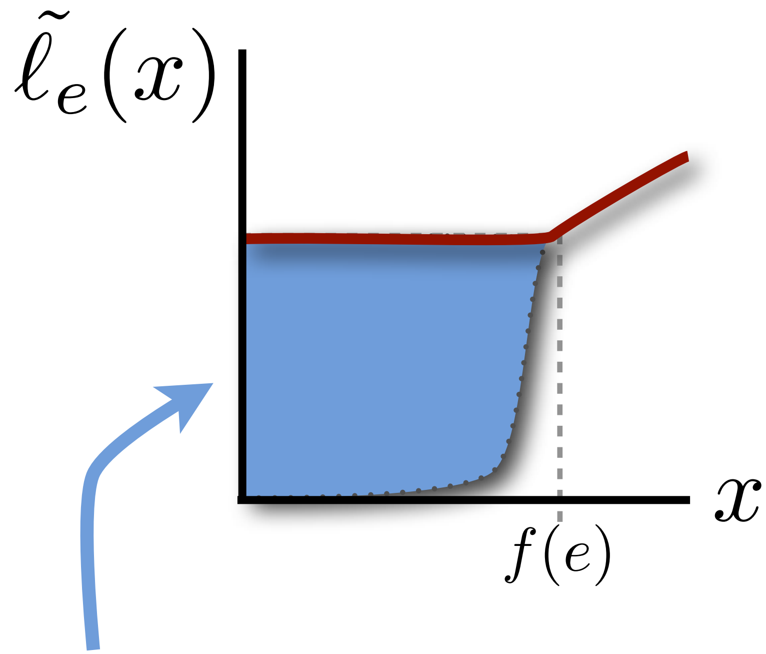
$$= C(f) \qquad \square$$

Original latency on edge e



Nash's cost for this link is area of grey rectangle.

Increased latency matches Nash's cost per unit flow



Increase in costs for any flow using this link is at most Nash cost.

Summing over all links, $\tilde{C}(f^*) \leq C(f^*) + C(f)$

i.e., $C(f^*) \geq \tilde{C}(f^*) - C(f)$

Proof outline

Define new "increased" latency functions, making f^* cost at most ( ) more.

$$C(f^*) \geq \tilde{C}(f^*) - C(f)$$ ✓

Increased latency is devised so that Nash equilibrium has minimum unit cost per flow...

$$\geq 2\tilde{C}(f) - C(f)$$

...and that cost equals the Nash's cost under the original latency function.

$$= 2C(f) - C(f)$$

$$= C(f) \qquad \square$$

Lemma: At Nash equilibrium, all paths being used have the minimum path latency *M*.

(Note this does not mean that the Nash is actually optimal. Why not?)

By construction, using the increased latencies, every edge has latency at least the Nash's.

Summing, every path has at least latency *M* regardless of how much flow it carries.

Thus, $$\tilde{C}(f^*) \geq M \cdot \mathrm{rate}(f^*)$$

$$= M \cdot 2 \cdot \mathrm{rate}(f)$$

$$= 2C(f).$$

Proof outline

Define new “increased” latency functions, making f^* cost at most ( ) more.

$$C(f^*) \geq \tilde{C}(f^*) - C(f)$$ ✓

Increased latency is devised so that Nash equilibrium has minimum unit cost per flow...

$$\geq 2\tilde{C}(f) - C(f)$$ ✓

...and that cost equals the Nash’s cost under the original latency function.

$$= 2C(f) - C(f)$$ ✓

$$= C(f) \qquad \square$$

Conclusion

- Selfishness hurts routing, but not too much.
- Is this a realistic model?
- Why are the results important or useful?
- Many more applications of game theory to CS (and CS to game theory). See: Nisan, Roughgarden, Tardos, Vazirani's book Algorithmic Game Theory, available free online!

Announcements

- Project midterm presentations one week from today
- 5 minute presentation (<= 3-5 slides)
  - What problem are you solving
  - Why doesn't the best past work solve it
  - Your solution approach
  - Demonstrate progress in your solution
- Meet with me!