Congestion Games

Congestion games

Congestion games

- Restricted class of games that are useful for modeling some important real-world settings
- Have attractive theoretical properties
- Simplify the representation of a game by imposing constraints on the effects that a single agent’s action can have on other agents’ utilities

Congestion games

- In a congestion game each player chooses some subset from a set of resources
- The cost of each resource depends on the number of other agents who select it

Congestion games

- A ***congestion game*** is a tuple (N, R, A, c), where
  - N is a set of n agents
  - R is a set of r resources
  - $A = A_1 \times \cdots \times A_n$, where $A_i \subseteq 2^R \setminus \{\varnothing\}$ is the set of actions for agent i
  - $c = (c_1, \ldots, c_r)$, where $c_k : \mathbb{N} \to \mathbb{R}$ is a cost function for resource $k \in R$

Congestion games

- $\#: R \times A \rightarrow \mathbb{N}$
  - a function for the number of players who took any action that involves resource ***r*** under action profile ***a***
- $c_k : \mathbb{N} \rightarrow \mathbb{R}$
  - a cost function for each resource ***k***

- Given a pure-strategy profile $\boldsymbol{a = (a_i, a_{-i})}$

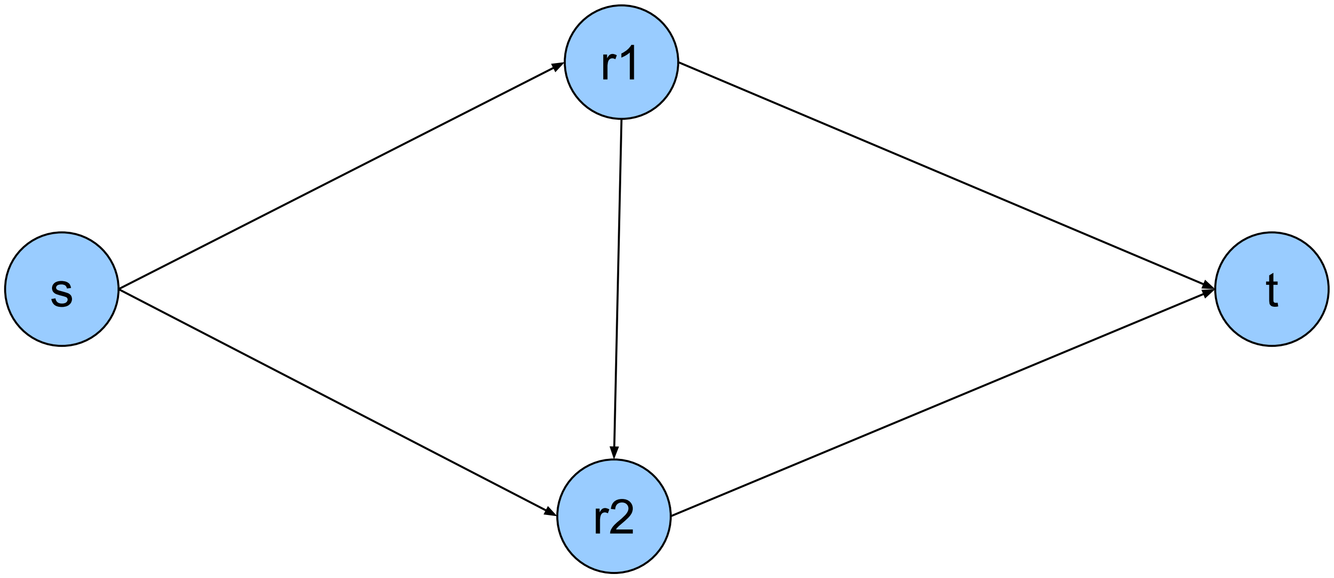
$$u_i(a) = - \sum_{r \in R | r \in a_i} c_r(\#(r, a))$$

Congestion games

- Agents can have different actions available to them
- But, they all have the same utility function
- Congestion games have the anonymity property
  - players care about how many others use a given resource but they do not care about which others do so

Congestion games

- Example:



R = {(s,r1), (s, r2), (r1, t), (r2, t), (r1, r2)}

A_i = {[(s,r1), (r1, t)], [(s,r1), (r1,r2), (r2,t)], [(s,r2), (r2,t)]}

c(i,j) = latency(#(r,a))

Congestion games

- Features functions $\boldsymbol{c_k(\cdot)}$ that are increasing in the number of people who choose that resource
- But can just as easily handle positive externalities
  - or even cost functions that oscillate

Congestion games

- Santa Fe Bar problem
  - each of a set of people independently selects whether or not to go to the bar
  - the utility of attending increases with the number of other people who select the same night, up to the capacity of the bar
  - Beyond this point, utility decreases because the bar gets too crowded
  - Deciding not to attend yields a baseline utility that does not depend on the actions of the participants
  - minority games: agents get the highest payoff for choosing a minority action

Computing equilibria

- **Theorem 6.4.2**
  - Every congestion game has a pure-strategy Nash equilibrium

Computing equilibria

- Mixed-strategy equilibria are open to criticisms that they are less likely than pure-strategy equilibria to arise in practice
- This theorem tells us that if we want to compute a sample Nash equilibrium of a congestion game, we can look for a pure-strategy equilibrium

Computing equilibria

function MYOPICBESTRESPONSE (game G, action profile a) **returns** a
while *there exists an agent* i *for whom* a_i *is not a best response to* a_{-i} **do**
  $a'_i \leftarrow$ some best response by i to a_{-i}
  $a \leftarrow (a'_i, a_{-i})$
return a

	L	C	R
U	$-1, 1$	$1, -1$	$-2, -2$
M	$1, -1$	$-1, 1$	$-2, -2$
D	$-2, -2$	$-2, -2$	$2, 2$

Computing equilibria

- MYOPICBESTRESPONSE may be too simplistic to be useful in practice
- Interestingly, it is useful for congestion games
- **Theorem 6.4.3**
  - The MYOPICBESTRESPONSE procedure is guaranteed to find a pure-strategy Nash equilibrium of a congestion game

Computing equilibria

- To prove the two previous theorems from the previous section, it is useful to introduce the concept of ***potential games***

Potential games

- Definition
  - A game $\boldsymbol{G = (N, A, u)}$ is a potential game if there exists a function $\boldsymbol{P : A} \rightarrow \mathbb{R}$ such that, for all $\boldsymbol{i \in N}$, all $\boldsymbol{a_{-i} \in A_{-i}}$ and $\boldsymbol{a_i}$, $\boldsymbol{a_i'} \in \boldsymbol{A_i}$
  - $\boldsymbol{u_i(a_i, a_{-i}) - u_i(a_i', a_{-i}) = P(a_i, a_{-i}) - P(a_i', a_{-i})}$
  - !the incentive of all players to change their strategy can be expressed using a single global function called the potential function!

Potential games

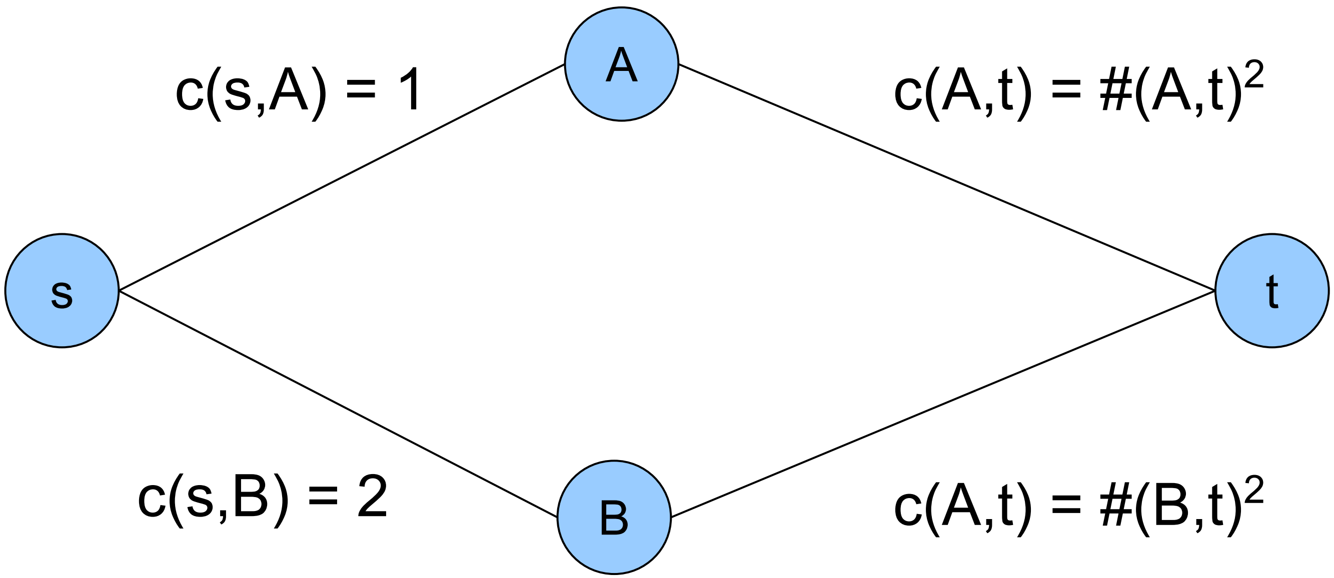
- **Theorem 6.4.5**
  - Every (finite) potential game has a pure-strategy Nash equilibrium
- **Proof**
  - Let $\boldsymbol{a* = argmax_{a \in A}\ P(a)}$
  - Clearly for any other action profile $\boldsymbol{a'}$, $\boldsymbol{P(a*) \geq P(a')}$
  - Thus by the definition of a potential function, for any agent $\boldsymbol{i}$ who can change the action profile from $\boldsymbol{a*}$ to $\boldsymbol{a'}$ by changing his own action,
  - $\boldsymbol{u_i(a*) \geq u_i(a')}$

Potential games

- **Theorem 6.4.6**
  - Every congestion game is a potential game
- **Proof**
  - if 
  - then

Potential games



$$u_i(a) = -\sum_{r \in R | r \in a_i} c_r(\#(r,a))$$

u(a)	A	B
A	(-5,-5)	(-2,-3)
B	(-3,-2)	(-6,-6)

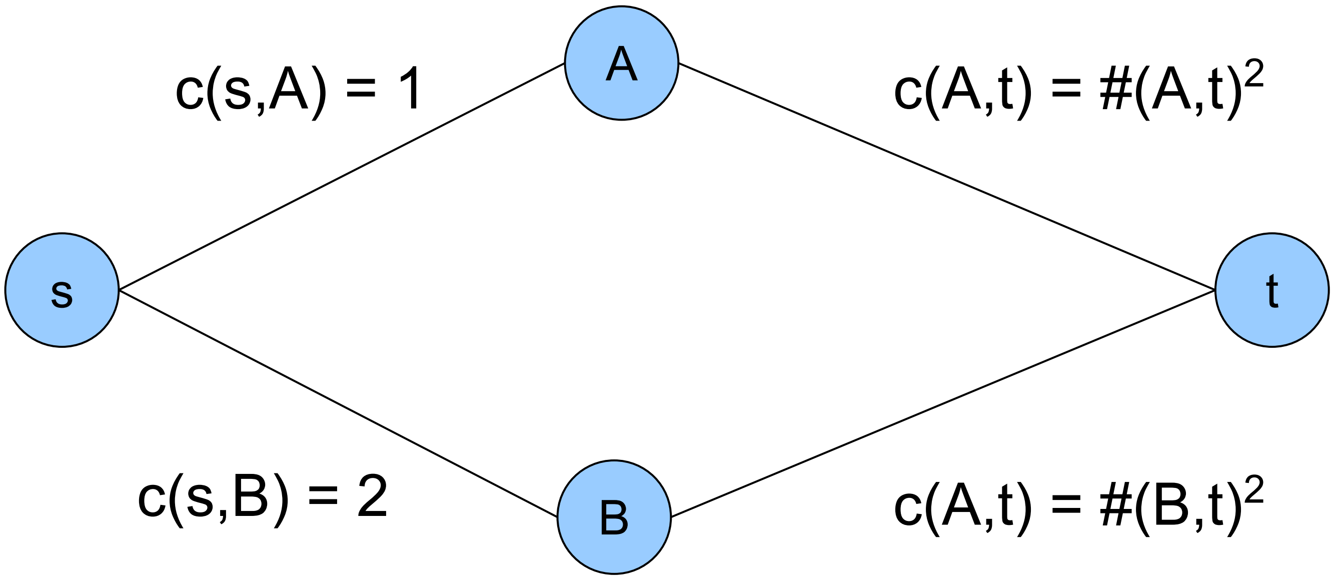
P(a)	A	B
A	7	5
B	5	9


Potential games

- **Theorem 6.4.6**
  - Every congestion game is a potential game
- **Proof**
  - if 
  - then

Potential games



$$u_i(a) = -\sum_{r \in R | r \in a_i} c_r(\#(r, a))$$

u(a)	A	B
A	(-5,-5)	(-2,-3)
B	(-3,-2)	(-6,-6)

P(a)	A	B
A	-7	-5
B	-5	-9

Error in the MAS book

- Thus, for the rest of the slides, consider that:

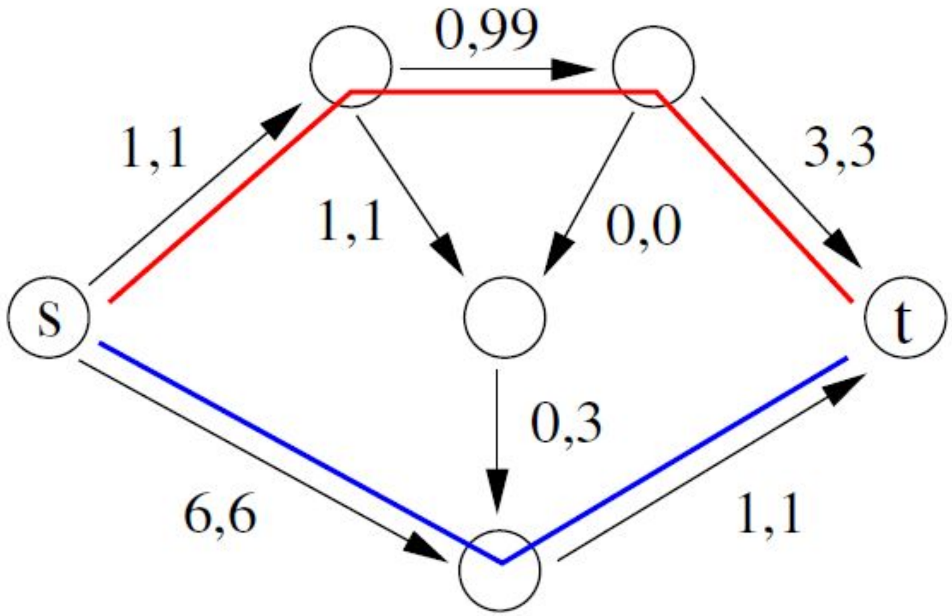
Potential games

- **Theorem 6.4.3**
  - The MYOPICBESTRESPONSE procedure is guaranteed to find a pure-strategy Nash equilibrium of a congestion game
- **Proof**
  - With every step of the while loop, $\boldsymbol{P(a)}$ strictly increases, because $\boldsymbol{u_i(a_i', a_{-i}) > u_i(a_i, a_{-i})}$ by construction
  - By the definition of a potential function $\boldsymbol{P(a_i', a_{-i}) > P(a_i, a_{-i})}$
  - Since there are only a finite number of action profiles, the algorithm must terminate

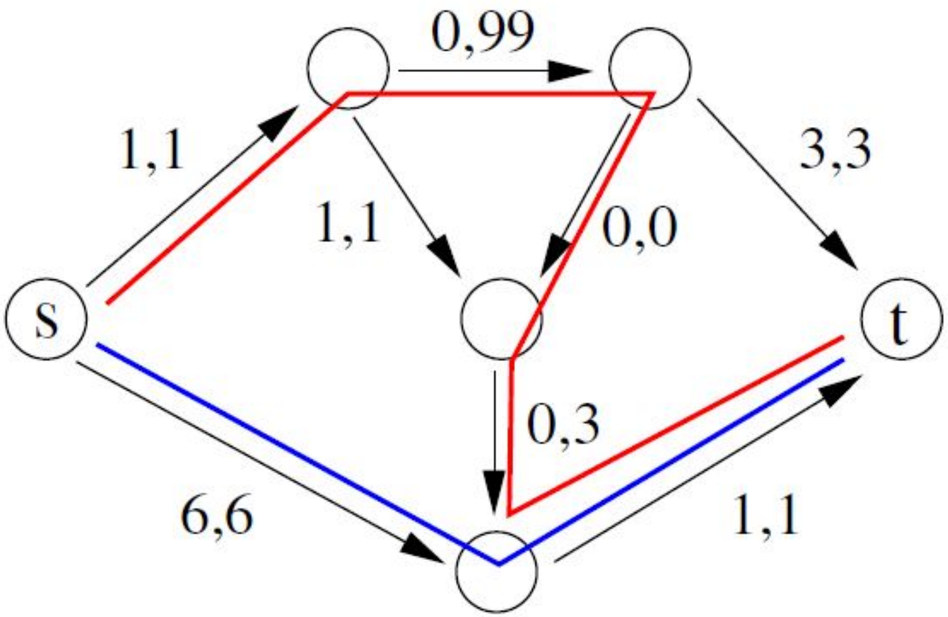
Potential games

- Given a congestion game, MYOPICBESTRESPONSE will converge regardless of
  - the cost functions (e.g., they do not need to be monotonic)
  - the action profile with which the algorithm is initialized
  - which agent we choose as agent ***i*** in the while loop
    - when there is more than one agent who is not playing a best response)
- Furthermore, it is not even necessary that agents best respond at every step

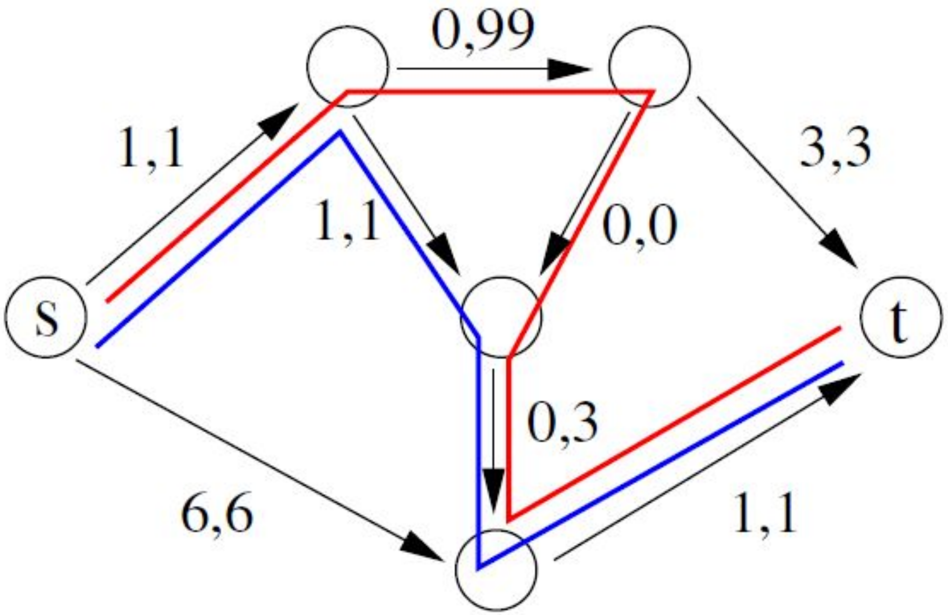
Example



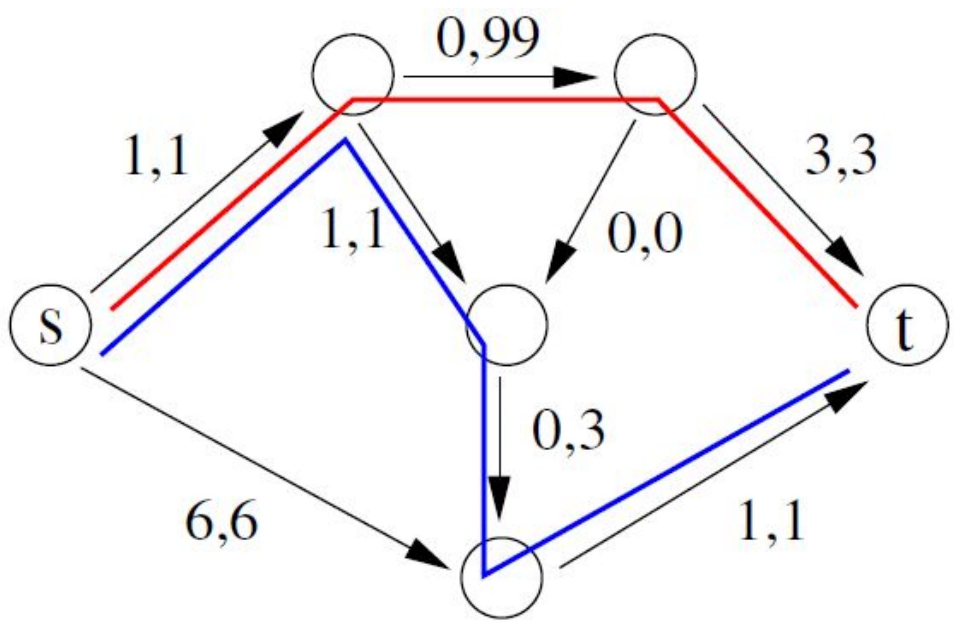
Example



Example



Example



Potential games

- Recently, it was shown that the problem of finding a pure NE in a congestion game is PLS-complete
  - as hard to find as any other object whose existence is guaranteed by a potential function argument
- Intuitively, this means that our problem is as hard as finding a local minimum in a traveling salesman problem using local search

Nonatomic congestion games

- A ***nonatomic congestion game*** is a congestion game that is played by an uncountably infinite number of players
- These games are used to model congestion scenarios in which
  - the number of agents is very large
  - each agent’s effect on the level of congestion is very small
- For example, consider modeling traffic congestion in a freeway system

Nonatomic congestion games

- A nonatomic congestion game is a tuple (N, μ, R, A, ρ, c), where:
  - $N = \{1, \ldots, n\}$ is a set of types of players
  - $\mu = (\mu_1, \ldots, \mu_n)$; for each $i \in N$ there is a continuum of players represented by the interval $[0, \mu_i]$
  - R is a set of k resources
  - $A = A_1 \times \cdots \times A_n$, where $A_i \subseteq 2^R \setminus \{\emptyset\}$ is the set of actions for agents of type i
  - $\rho = (\rho_1, \ldots, \rho_n)$, where for each $i \in N$, $\rho_i : A_i \times R \to \mathbb{R}_+$ denotes the amount of congestion contributed to a given resource $r \in R$ by players of type i selecting a given action $a_i \in A_i$
  - $c = (c_1, \ldots, c_k)$, where $c_r : \mathbb{R}_+ \to \mathbb{R}$ is a cost function for resource $r \in R$, and c_r is nonnegative, continuous and nondecreasing

Nonatomic congestion games

- To simplify notation, denote  as the union of $A_1, \ldots, A_n$
- Let 
- An action distribution $s \in S$ indicates how many players choose each action
- $s(a_i)$, denote the element of s that corresponds to the measure of the set of players of type i who select action $a_i \in A_i$
- An action distribution s must have the properties that all entries are nonnegative real numbers and that 
- Note that $\rho_i(a_i, r) = 0$ when $r \notin a_i$

Nonatomic congestion games

- Overloading notation, we write as $\boldsymbol{s_r}$ the amount of congestion induced on resource $\boldsymbol{r \in R}$ by action distribution $\boldsymbol{s}$:

$$s_r = \sum_{i \in N} \sum_{a_i \in A_i} \rho_i(a_i, r) s(a_i)$$

Nonatomic congestion games

- We can now express the utility function
- As in (atomic) congestion games, all agents have the same utility function
  - depends only on how many agents choose each action rather than on these agents' identities

Nonatomic congestion games

- The cost under an action distribution ***s*** to agents of type ***i*** who choose action $\boldsymbol{a_i}$ is given by

$$c_{a_i}(s) = \sum_{r \in a_i} \rho_i(a_i, r) c_r(s_r)$$

- and so we have

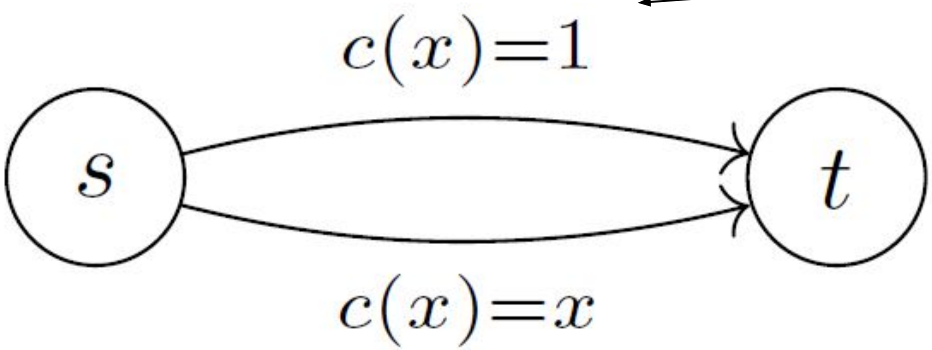
Nonatomic congestion games

- Finally, we can define the **social cost** of an action profile as the total cost born by all the agents

$$C(s) = \sum_{i \in N} \sum_{a_i \in A_i} s(a_i) c_{a_i}(s)$$

Nonatomic congestion games

input for cost functions



$$s_r = \sum_{i \in N} \sum_{a_i \in A_i} \rho_i(a_i, r) s(a_i)$$

$$c_{a_i}(s) = \sum_{r \in a_i} \rho_i(a_i, r) c_r(s_r)$$

$$C(s) = \sum_{i \in N} \sum_{a_i \in A_i} s(a_i) c_{a_i}(s)$$

$N = \{1\}$
$\mu = \{\mu_1\}$ and $\mu_1 = 1$
Interval: $[0\ 1]$
$R = \{e_{top}, e_{bottom}\}$
$A_1 = \{a_1, a_2\}$
$a_1=\{e_{top}\}, a_2=\{e_{bottom}\}$
$\rho_1(a, r)=1\ \forall\ a \in A_1, r \in R \cap a$
$\rho_1(a, r)=0$ *otherwise*

for $s(a_1) = 0, s(a_2) = 1$:
$s_{top} = \rho_1(a_1,e_{top})*0 + \rho_1(a_2,e_{top})*1$
$= 1*0 + 0*1 = 0$
$s_{bottom} = \rho_1(a_1,e_{bottom})*0 + \rho_1(a_2,e_{bottom})*1$
$= 0*0 + 1*1 = 1$
$c_{a1}(s) = \rho_1(a_1,e_{top})*c_{top}(0) = 1*1 = 1$
$c_{a2}(s) = \rho_1(a_2,e_{bottom})*c_{bottom}(1) = 1*1 = 1$
$C(s) = 0*1 + 1*1 = 1$

Nonatomic congestion games

- Even with an uncountably infinite number of agents, we can still define a Nash equilibrium in the usual way

Nonatomic congestion games

- Definition
  - An action distribution $\boldsymbol{s}$ arises in a pure-strategy equilibrium of a nonatomic congestion game if for each player type $\boldsymbol{i \in N}$ and each pair of actions $\boldsymbol{a_1, a_2 \in A_i}$ with $\boldsymbol{s(a_1) > 0}$, $\boldsymbol{u_i(a_1, s) \geq u_i(a_2, s)}$ (and hence $\boldsymbol{c_{a1}(s) \leq c_{a2}(s)}$).

Nonatomic congestion games

- Here we will only be concerned with pure-strategy equilibria
  - Not restrictive, since any mixed-strategy equilibrium corresponds to an “equivalent” pure-strategy equilibrium where the number of agents playing a given action is the expected number under the original equilibrium
- We say only that an action distribution arises in an equilibrium because an action distribution does not identify the action taken by every individual agent, and hence cannot constitute an equilibrium

Nonatomic congestion games

- **Theorem 6.4.9**
  - Every nonatomic congestion game has a pure-strategy Nash equilibrium.
- **Theorem 6.4.10**
  - All equilibria of a nonatomic congestion game have equal social cost

Selfish Routing

- Selfish routing is a model of how self-interested agents would route traffic through a congested network
- This model was studied as early as 1920—long before game theory developed as a field
- Today, we can understand these problems as nonatomic congestion games

Selfish Routing

- Let $G = (V,E)$ be a directed graph having n source–sink pairs $(s_1, t_1), \dots, (s_n, t_n)$
- Some volume of traffic must be routed from each source to each sink
- For a given source–sink pair (s_i, t_i) let P_i denote the set of simple paths from s_i to t_i
- We assume that $P \neq \varnothing$ for all i
- It is permitted for there to be multiple "parallel" edges between the same pair of nodes in V
- Also permitted for paths from P_i and P_j $(j \neq i)$ to share edges

Selfish Routing

- Let $\boldsymbol{\mu} \in \mathbb{R}_+^n$ denote a vector of traffic rates
  - $\boldsymbol{\mu}_i$ denotes the amount of traffic that must be routed from $\boldsymbol{s}_i$ to $\boldsymbol{t}_i$
- Every edge $\boldsymbol{e} \in \boldsymbol{E}$ is associated with a cost function $\boldsymbol{c}_e : \mathbb{R}_+ \rightarrow \mathbb{R}$ that can depend on the amount of traffic carried by the edge
  - e.g. amount of delay

Selfish Routing

- Problem:
  - Determine how the given traffic rates will lead traffic to flow along each edge, assuming that agents are selfish and will direct their traffic to minimize the sum of their own costs

Selfish Routing

- Selfish routing problems can be encoded as nonatomic congestion games as follows:
  - $\boldsymbol{N}$ is the set of source–sink pairs
  - $\boldsymbol{\mu}$ is the set of traffic rates
  - $\boldsymbol{R}$ is the set of edges $\boldsymbol{E}$
  - $\boldsymbol{A_i}$ is the set of paths $\boldsymbol{P_i}$ from $\boldsymbol{s_i}$ to $\boldsymbol{t_i}$
  - $\boldsymbol{\rho_i}$ is always $\boldsymbol{1}$
  - $\boldsymbol{c_r}$ is the edge cost function $\boldsymbol{c_e}$

The price of anarchy

- From the reduction to nonatomic congestion games and from Theorems **6.4.9** and **6.4.10**
  - every selfish routing problem has at least one pure strategy Nash equilibrium (or Wardrop equilibrium)
  - all of a selfish routing problem's equilibria have equal social cost. These properties allow us to ask an interesting question:

The price of anarchy

- Question: how similar is the optimal social cost to the social cost under an equilibrium action distribution?

The price of anarchy

- Definition
  - The price of anarchy of a nonatomic congestion game ***(N, μ, R, A, ρ, c)*** having equilibrium ***s*** and social cost minimizing action distribution ***s∗*** is defined as

$$\frac{C(s)}{C(s^*)}$$

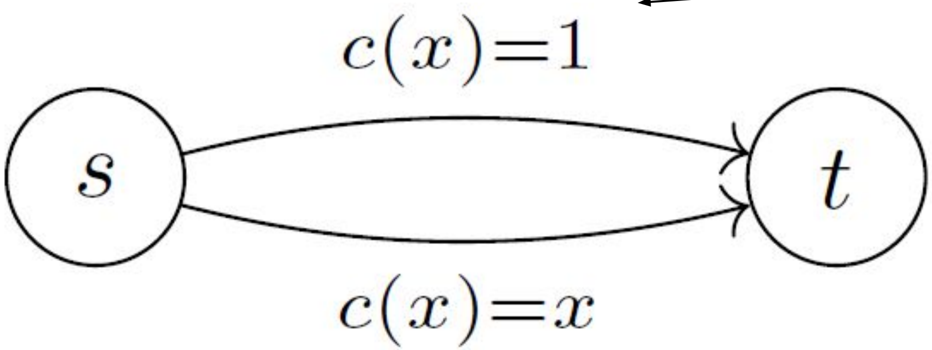
  - unless ***C(s∗) = 0***, in which case the price of anarchy is defined to be ***1***

The price of anarchy

- Intuition: the proportion of additional social cost that is incurred because of agents’ selfishness
- When this ratio is close to 1, agents are routing traffic about as well as possible
- When this ratio is large, agents’ selfish behavior is causing significantly suboptimal network performance
  - is it possible to change either the network or the agents’ behavior in order to reduce the social cost?

Nonatomic congestion games

input for cost functions



$$s_r = \sum_{i \in N} \sum_{a_i \in A_i} \rho_i(a_i, r) s(a_i)$$

$$c_{a_i}(s) = \sum_{r \in a_i} \rho_i(a_i, r) c_r(s_r)$$

$$C(s) = \sum_{i \in N} \sum_{a_i \in A_i} s(a_i) c_{a_i}(s)$$

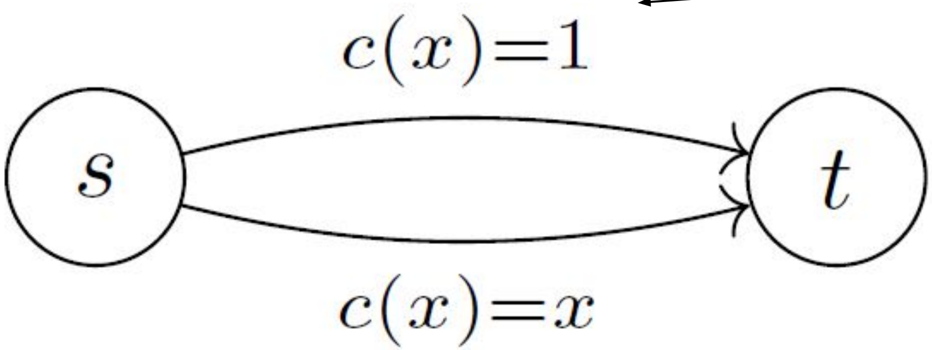
$N = \{1\}$
$\mu = \{\mu_1\}$ and $\mu_1 = 1$
Interval: $[0\ 1]$
$R = \{e_{top}, e_{bottom}\}$
$A_1 = \{a_1, a_2\}$
$a_1=\{e_{top}\}$, $a_2=\{e_{bottom}\}$
$\rho_1(a, r)=1\ \forall\ a \in A_1, r \in R \cap a$
$\rho_1(a, r)=0$ otherwise

solution in equilibrium:

$s(a_1) = 0,\ s(a_2) = 1$:
$s_{top} = \rho_1(a_1,e_{top})*0 + \rho_1(a_2,e_{top})*1$
$= 1*0 + 1*1 = 1$
$s_{bottom} = \rho_1(a_1,e_{bottom})*0 + \rho_1(a_2,e_{bottom})*1$
$= 1*0 + 1*1 = 1$
$c_{a1}(s) = \rho_1(a_1,e_{top})*c_{top}(1) = 1*1 = 1$
$c_{a2}(s) = \rho_1(a_1,e_{bottom})*c_{bottom}(1) = 1*1 = 1$
$C(s) = 0*1 + 1*1 = 1$

The price of anarchy

input for cost functions



$$s_r = \sum_{i \in N} \sum_{a_i \in A_i} \rho_i(a_i, r) s(a_i)$$

$$c_{a_i}(s) = \sum_{r \in a_i} \rho(a_i, r) c_r(s_r)$$

$$C(s) = \sum_{i \in N} \sum_{a_i \in A_i} s(a_i) c_{a_i}(s)$$

$N = \{1\}$
$\mu = \{\mu_1\}$ and $\mu_1 = 1$
Interval: $[0\ 1]$
$R = \{e_{top}, e_{bottom}\}$
$A_1 = \{a_1, a_2\}$
$a_1=\{e_{top}\}$, $a_2=\{e_{bottom}\}$
$\rho_1(a, r)=1\ \forall\ a \in A_1, r \in R \cap a$
$\rho_1(a, r)=0$ *otherwise*

optimal solution

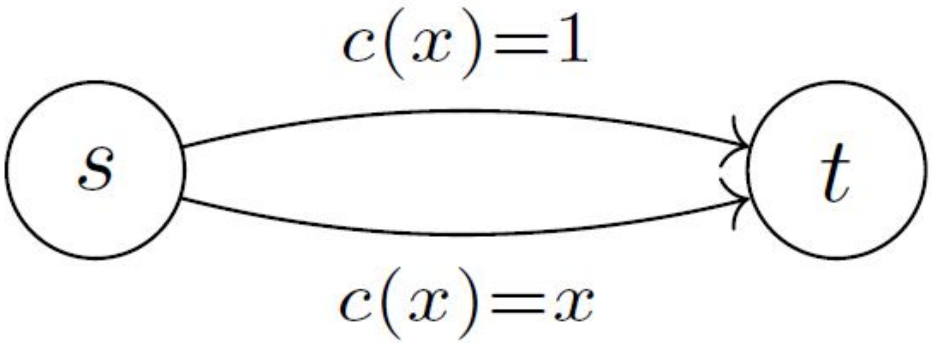
$s(a_1) = ½$, $s(a_2) = ½$:

$s_{top} = \rho_1(a_1,e_{top})*½ + \rho_1(a_2,e_{top})*½$
$= 1*½ + 0*½ = ½$

$s_{bottom} = \rho_1(a_1,e_{bottom})*½ +$
$\rho_1(a_2,e_{bottom})*½$
$= 0*½ + 1*½ = ½$

$c_{a1}(s) = \rho_1(a_1,e_{top})*c_{top}(½) = 1*1 = 1$
$c_{a2}(s) = \rho_1(a_1,e_{bottom})*c_{bottom}(½) = 1*½ =$
$½$

The price of anarchy



optimal solution

C(s*) = 0.75

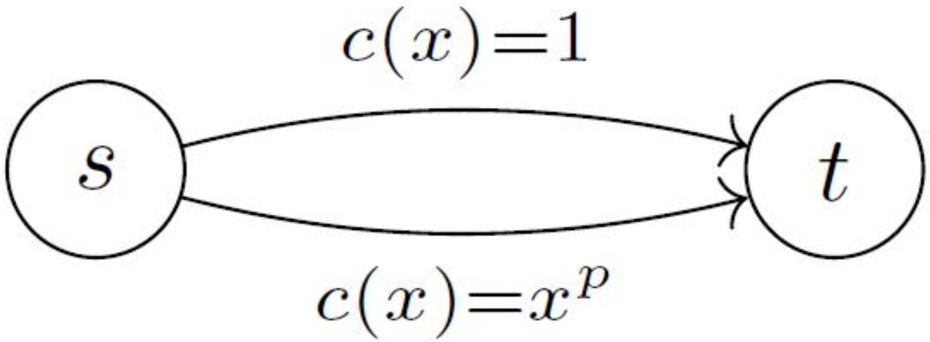
solution in equilibrium

C(s) = 1

price of anarchy = $\dfrac{C(s)}{C(s^*)}$

= 4/3

The price of anarchy



price of anarchy = $\dfrac{C(s)}{C(s^*)}$

$\longrightarrow \infty$

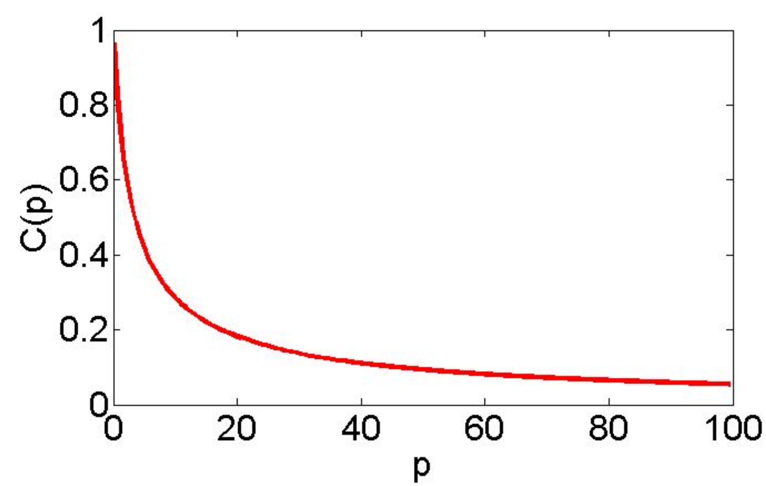
***N* = {1}**
***μ* = {*μ*$_1$}** and ***μ*$_1$ = 1**

solution in equilibrium

C(s) = 0*1 + 1*1 = 1

optimal solution

C(x) = (1-x) + x.x^p
argmin$_x$(C(x)) = (p+1)$^{-1/p}$
C(s*) = 1-p.(p+1)$^{-(p+1)/p}$
C(s*) $\rightarrow$ 0 as p $\rightarrow \infty$



The price of anarchy

- This example illustrates that the price of anarchy is unbounded for unrestricted cost functions
- On the other hand, it turns out to be possible to offer bounds in the case where cost functions are restricted to a particular set

The price of anarchy

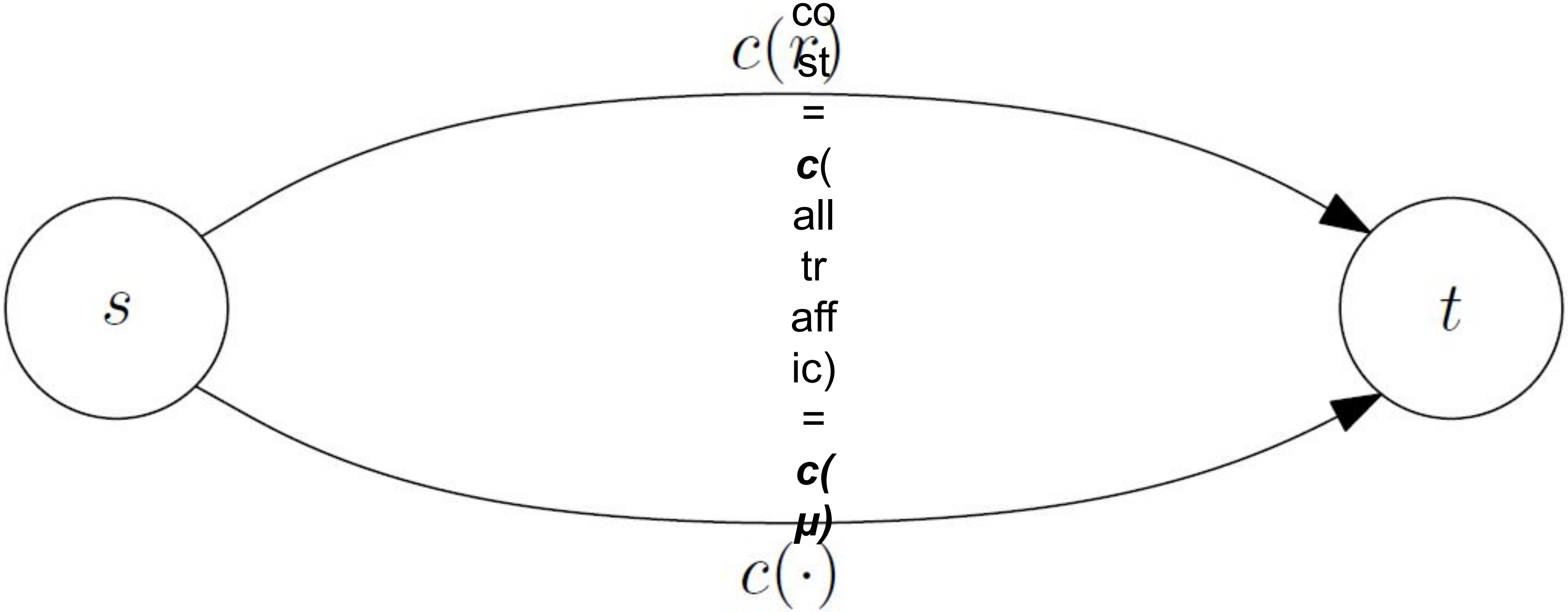
- Question: how inefficient are Nash flows in more realistic networks?
- Recommended reading: *Selfish Routing and the Price of Anarchy, Tim Roughgarden, 2006*

The price of anarchy

Arthur C. Pigou

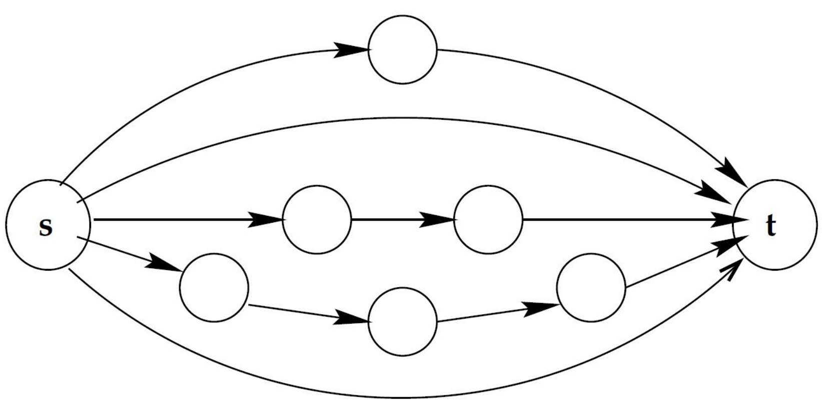
- Pigou network
  - The economics of Welfare, 1920

constant cost = ***c***(all traffic) = ***c(μ)***

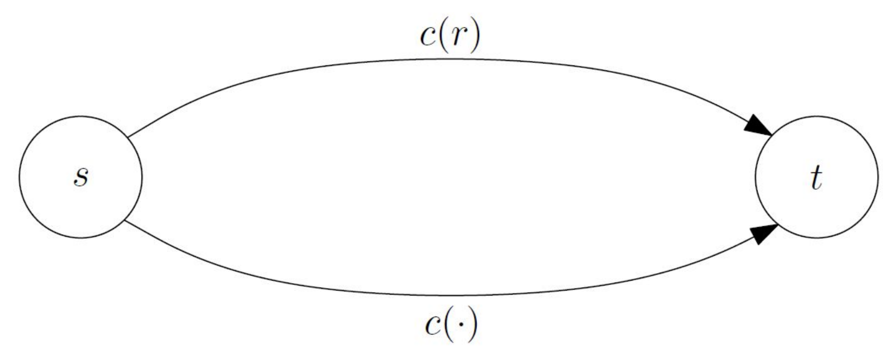


The price of anarchy

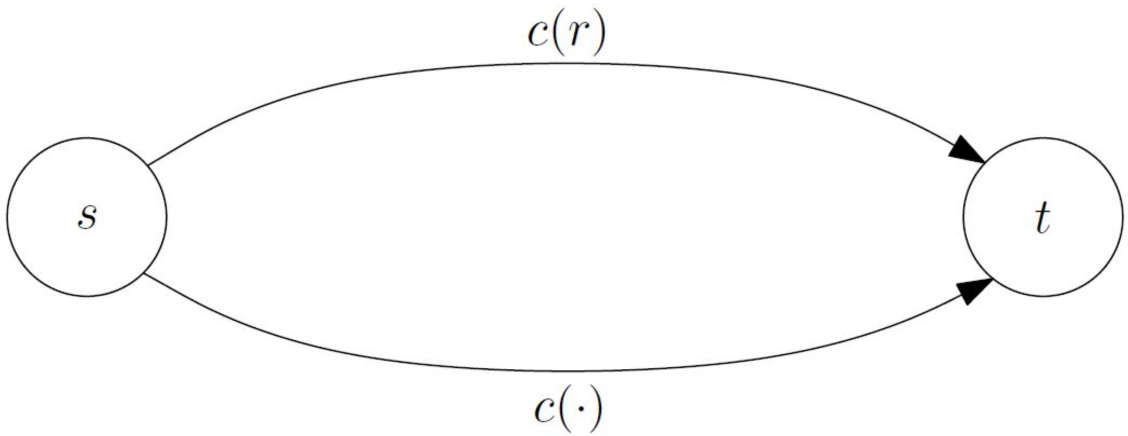
- Theorem [Roughgarden 02]:
  - Fix any class of latency functions, and the worst *price of anarchy* occurs in a Pigou network



$<$



The price of anarchy



- **Theorem 6.4.12**
  - The price of anarchy of a selfish routing problem whose cost functions are taken from the set **C** is never more than α(**C**), where

$$\alpha(\mathcal{C}) = \sup_{c \in \mathcal{C}} \sup_{x, r \geq 0} \frac{r \cdot c(r)}{x \cdot c(x) + (r - x)c(r)}$$

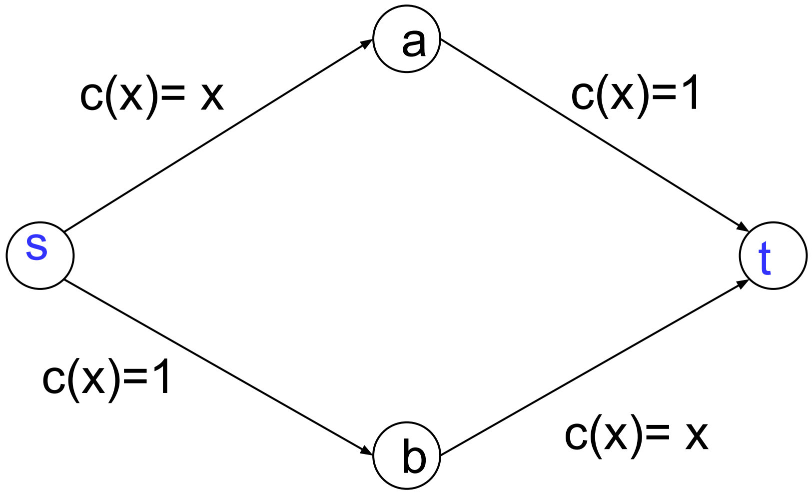
The price of anarchy

- **Theorem 6.4.12**
  - The price of anarchy of a selfish routing problem whose cost functions are taken from the set **C** is never more than ***α(C)***

Description	Typical Representative	Price of Anarchy
Linear	$ax + b$	$4/3$
Quadratic	$ax^2 + bx + c$	$\frac{3\sqrt{3}}{3\sqrt{3}-2} \approx 1.6$
Cubic	$ax^3 + bx^2 + cx + d$	$\frac{4\sqrt[3]{4}}{4\sqrt[3]{4}-3} \approx 1.9$
Quartic	$ax^4 + \cdots$	$\frac{5\sqrt[4]{5}}{5\sqrt[4]{5}-4} \approx 2.2$
Polynomials of degree $\leq d$	$\sum_{i=0}^{d} a_i x^i$	$\frac{(d+1)\sqrt[d]{d+1}}{(d+1)\sqrt[d]{d+1}-d} \approx \frac{d}{\ln d}$

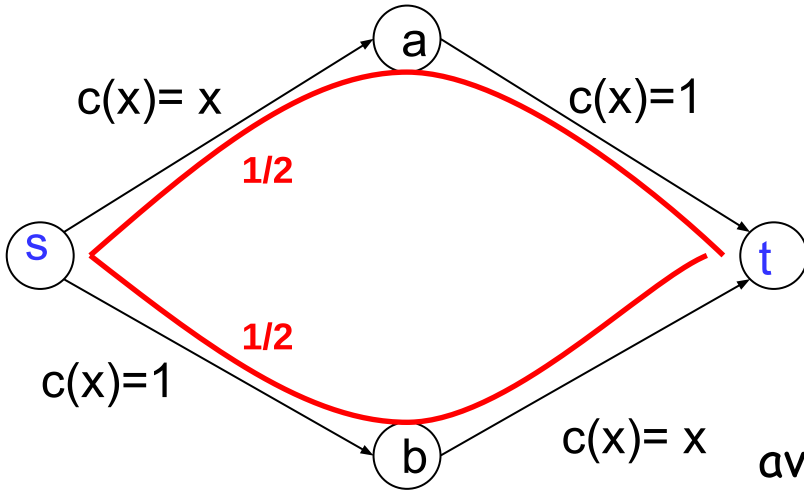
Reducing the Social Cost

- When the equilibrium social cost is undesirably high, a network operator might want to intervene in some way in order to reduce it



$N = \{1\}$
$\mu = \{\mu_1\}$ and $\mu_1 = 1$
$R = \{(s,a), (a,t), (s,b), (b,t)\}$
$A_1 = \{a_1, a_2\}$
$a_1=\{(s,a), (a,t)\}, a_2=\{(s,b), (b,t)\}$
$\rho_1 = 1$ (by definition)

Braess's Paradox



average latency = 1+0.5 =1.5

$\boldsymbol{N = \{1\}}$

$\boldsymbol{\mu = \{\mu_1\}}$ and $\boldsymbol{\mu_1 = 1}$

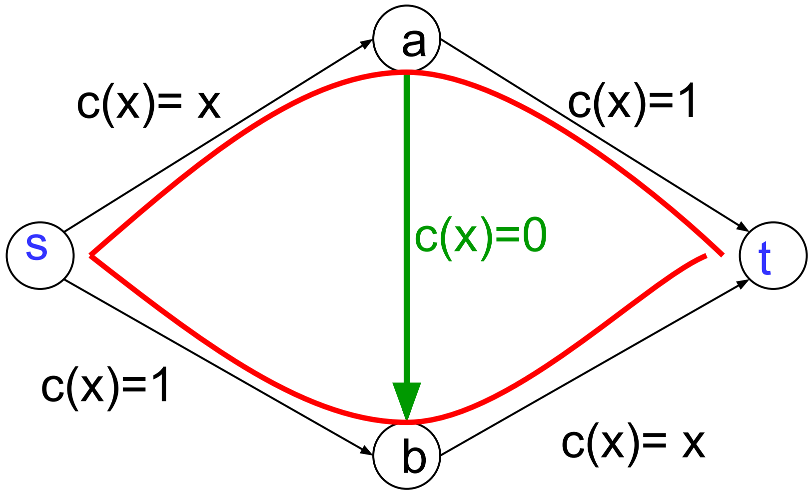
$\boldsymbol{R = \{(s,a), (a,t), (s,b), (b,t)\}}$

$\boldsymbol{A_1 = \{a_1, a_2\}}$

$\boldsymbol{a_1=\{(s,a), (a,t)\}, a_2=\{(s,b), (b,t)\}}$

$\boldsymbol{\rho_1 = 1}$ (by definition)

Braess's Paradox



$\boldsymbol{N = \{1\}}$

$\boldsymbol{\mu = \{\mu_1\}}$ and $\boldsymbol{\mu_1 = 1}$

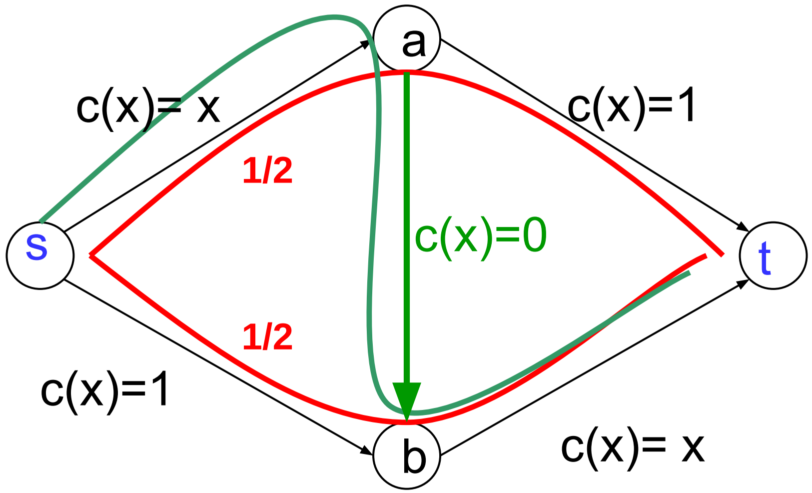
$\boldsymbol{R = \{(s,a), (a,t), (s,b), (b,t), (a,b)\}}$

$\boldsymbol{A_1 = \{a_1, a_2, a_3\}}$

$\boldsymbol{a_1=\{(s,a), (a,t)\}, a_2=\{(s,b), (b,t)\}, a_3=\{(s,a), (a,b), (b,t)\}}$

$\boldsymbol{\rho_1 = 1}$ (by definition)

Braess's Paradox



$N = \{1\}$

$\mu = \{\mu_1\}$ and $\mu_1 = 1$

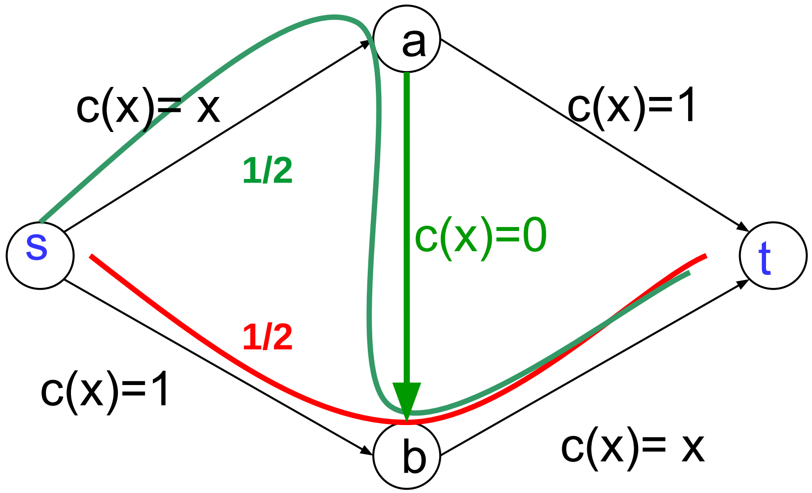
$R = \{(s,a), (a,t), (s,b), (b,t), (a,b)\}$

$A_1 = \{a_1, a_2, a_3\}$

$a_1=\{(s,a), (a,t)\}$, $a_2=\{(s,b), (b,t)\}$, $a_3=\{(s,a), (a,b), (b,t)\}$

$\rho_1 = 1$ (by definition)

Braess's Paradox



$N = \{1\}$

$\mu = \{\mu_1\}$ and $\mu_1 = 1$

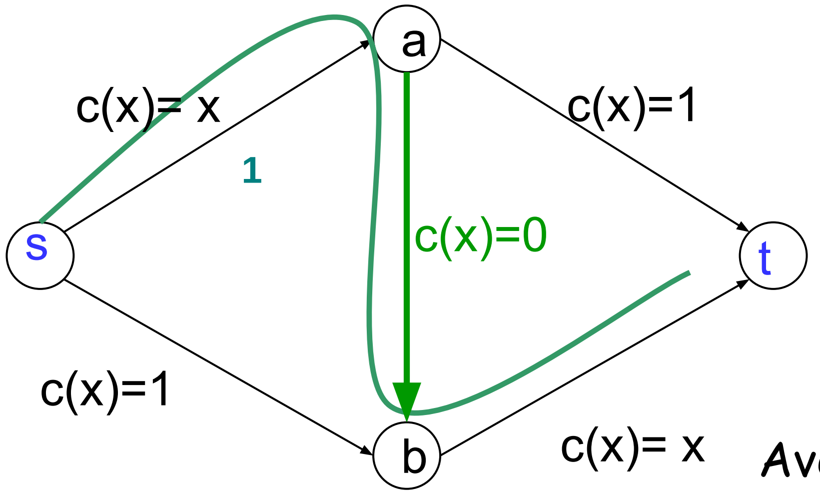
$R = \{(s,a), (a,t), (s,b), (b,t), (a,b)\}$

$A_1 = \{a_1, a_2, a_3\}$

$a_1=\{(s,a), (a,t)\}, a_2=\{(s,b), (b,t)\}, a_3=\{(s,a), (a,b), (b,t)\}$

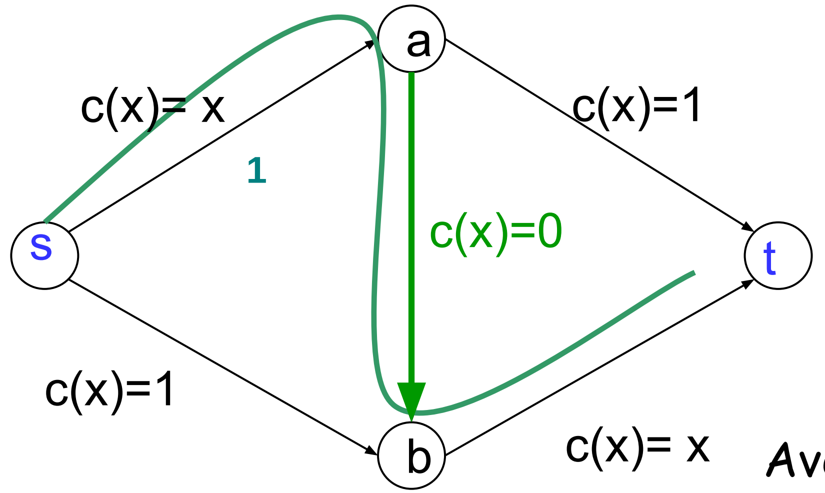
$\rho_1 = 1$ (by definition)

Braess's Paradox



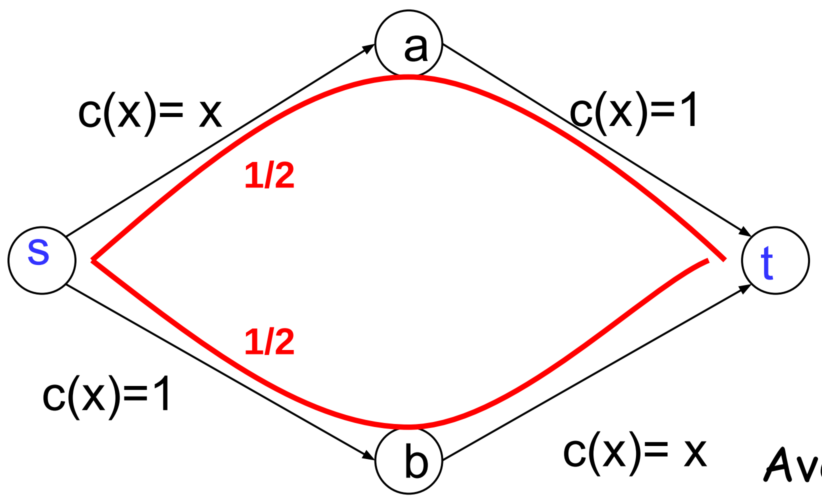
$\boldsymbol{N = \{1\}}$
$\boldsymbol{\mu = \{\mu_1\}}$ and $\boldsymbol{\mu_1 = 1}$
$\boldsymbol{R = \{(s,a), (a,t), (s,b), (b,t), (a,b)\}}$
$\boldsymbol{A_1 = \{a_1, a_2, a_3\}}$
$\boldsymbol{a_1=\{(s,a), (a,t)\}, a_2=\{(s,b), (b,t)\}, a_3=\{(s,a), (a,b), (b,t)\}}$
$\boldsymbol{\rho_1 = 1}$ (by definition)

Braess's Paradox



Average latency = 1+1 =2

PoA = 2/1.5 = 4/3

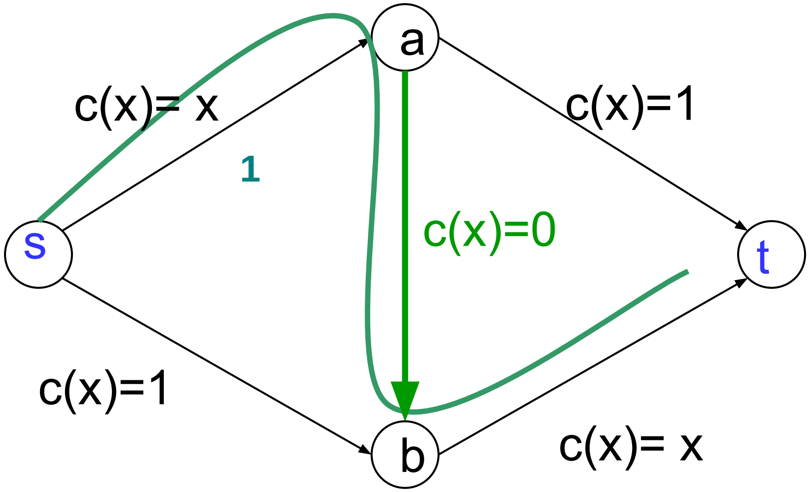


Average latency = 1+0.5 =1.5

PoA = 1.5/1.5 = 1

Reducing the Social Cost

- Given a selfish routing problem, is it possible to find edges to remove to reduce the price of anarchy?



Reducing the Social Cost

- ***<u>Theorem 6.4.13</u>***
  - It is NP-complete to determine whether there exists any set of edges whose removal from a selfish routing problem would reduce the social cost in equilibrium
- This result implies that identifying the optimal set of edges to remove from a selfish routing problem in order to minimize the social cost in equilibrium is also NP-complete

Real world examples

- In Seoul, South Korea, a speeding-up in traffic around the city was seen when a motorway was removed as part of the Cheonggyecheon restoration project
- In Stuttgart, Germany after investments into the road network in 1969, the traffic situation did not improve until a section of newly built road was closed for traffic again
- In 1990 the closing of 42nd street in New York City reduced the amount of congestion in the area
- In 2012, scientists at the Max Planck Institute for Dynamics and Self-Organization demonstrated through computational modeling the potential for this phenomenon to occur in power transmission networks where power generation is decentralized
- In 2012, a team of researchers published in Physical Review Letters a paper showing that Braess paradox may occur in mesoscopic electron systems. They showed that adding a path for electrons in a nanoscopic network paradoxically reduced its conductance

Reducing the Social Cost

- When it is relatively inexpensive to speed up a network, doing so can have more significant benefits than getting agents to change their behavior

Reducing the Social Cost

- Stackelberg routing
  - a small fraction of agents are routed centrally, and the remaining population of agents is free to choose their own actions
  - e.g. Pigou network
- Raising edge costs
  - Taxes can be imposed on certain edges in the graph in order to encourage agents to adopt more socially beneficial behavior
  - “marginal cost pricing”: each agent pays the amount his presence cost other agents who are using the same edge