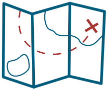
In Congestion Games, Taxes Achieve Optimal Approximation

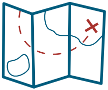
Dario Paccagnan, Martin Gairing

Imperial College London

UNIVERSITY OF LIVERPOOL

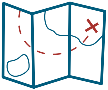


Problem: minimum social cost in atomic congestion games



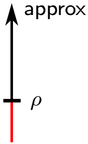
Problem: minimum social cost in atomic congestion games

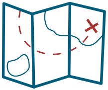
Main result I: tight NP-hardness of approximation



Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

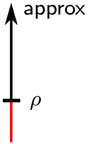


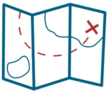


Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation

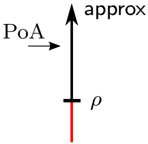


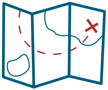


Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation

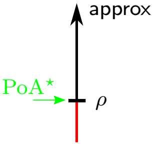


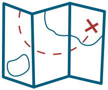


Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation



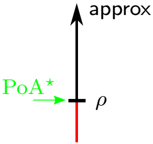


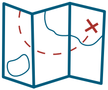
Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation

$\implies$ first poly algo optimal approx



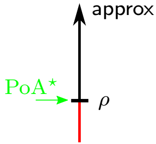


Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation

$\Longrightarrow$ first poly algo optimal approx

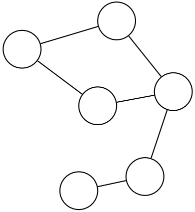


Judiciously designed taxes achieve optimal approx,
and no other tractable intervention can improve

Atomic congestion games

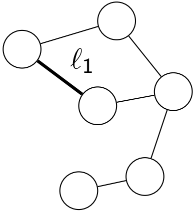
Atomic congestion games

- Set of resources $\mathcal{R}$



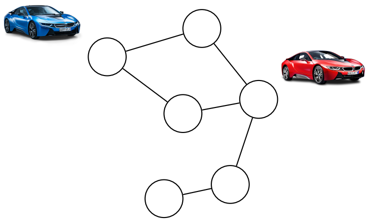
Atomic congestion games

- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$



Atomic congestion games

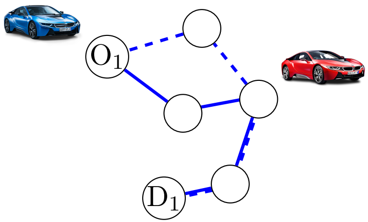
- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$
- Set of players $\{1, \ldots, N\}$



Atomic congestion games

- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$

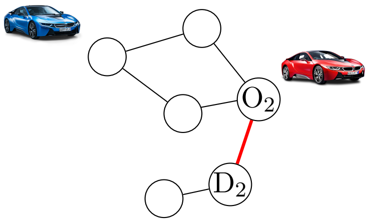
- Set of players $\{1, \ldots, N\}$
- Player i feasible set $\mathcal{A}_i \subseteq 2^{\mathcal{R}}$



Atomic congestion games

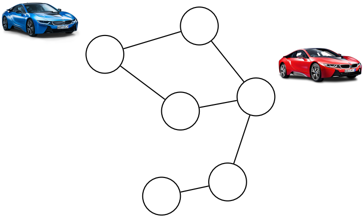
- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$

- Set of players $\{1, \dots, N\}$
- Player i feasible set $\mathcal{A}_i \subseteq 2^{\mathcal{R}}$

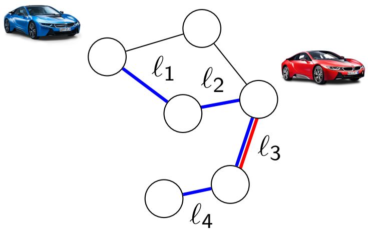


Atomic congestion games

- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$
- Set of players $\{1, \ldots, N\}$
- Player i feasible set $\mathcal{A}_i \subseteq 2^{\mathcal{R}}$
- Player i cost $C_i(a) = \sum_{r \in a_i} \ell_r(|a|_r)$



Atomic congestion games



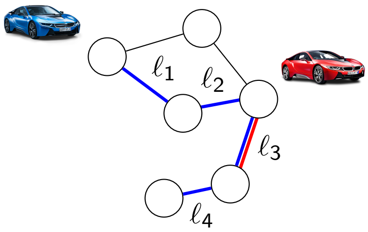
- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$

- Set of players $\{1, \dots, N\}$
- Player i feasible set $\mathcal{A}_i \subseteq 2^{\mathcal{R}}$
- Player i cost $C_i(a) = \sum_{r \in a_i} \ell_r(|a|_r)$

System cost: $\quad SC(a) = \sum_i C_i(a)$

Atomic congestion games

- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$



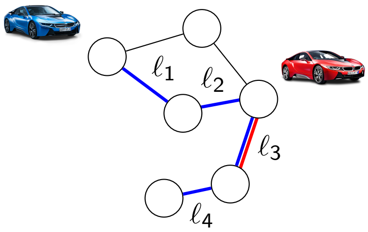
$SC = \ell_1(1) + \ell_2(1) + 2\ell_3(2) + \ell_4(1)$

- Set of players $\{1, \dots, N\}$
- Player i feasible set $\mathcal{A}_i \subseteq 2^{\mathcal{R}}$
- Player i cost $C_i(a) = \sum_{r \in a_i} \ell_r(|a|_r)$

System cost: $SC(a) = \sum_i C_i(a)$

Atomic congestion games

- Set of resources $\mathcal{R}$
- Resource costs $\ell_r(\cdot)$

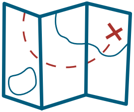


$SC = \ell_1(1) + \ell_2(1) + 2\ell_3(2) + \ell_4(1)$

- Set of players $\{1, \ldots, N\}$
- Player i feasible set $\mathcal{A}_i \subseteq 2^{\mathcal{R}}$
- Player i cost $C_i(a) = \sum_{r \in a_i} \ell_r(|a|_r)$

$$\text{System cost:} \quad SC(a) = \sum_i C_i(a)$$

Applications: routing, sensor allocation, scheduling, minimum power, ...



Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\implies$ first poly algo optimal approx

Hardness of approximation – related work

$$\texttt{MinSC} : \min_{a \in \mathcal{A}} SC(a)$$

Hardness of approximation – related work

$$\texttt{MinSC} : \min_{a \in \mathcal{A}} SC(a)$$

* $\texttt{MinSC}$ is NP-hard [Meyers/Schulz, Networks'12]

* $\texttt{MinSC}$ is NP-hard if latencies are linear [Castiglioni/Celli/Marchesi/Gatti, ArXiv'20]

Hardness of approximation – related work

$$\texttt{MinSC} : \min_{a \in \mathcal{A}} SC(a)$$

* $\texttt{MinSC}$ is NP-hard [Meyers/Schulz, Networks'12]

* $\texttt{MinSC}$ is NP-hard if latencies are linear [Castiglioni/Celli/Marchesi/Gatti, ArXiv'20]

* If latencies are polynomial of degree $\leq d$, then $\texttt{MinSC}$ is NP-hard to approx within a factor $(\beta d)^{\frac{d}{2}}$, for some $\beta > 0$ [Roughgarden, FOCS'12]

Hardness of approximation – related work

$$\texttt{MinSC} : \min_{a \in \mathcal{A}} SC(a)$$

* $\texttt{MinSC}$ is NP-hard [Meyers/Schulz, Networks'12]

* $\texttt{MinSC}$ is NP-hard if latencies are linear [Castiglioni/Celli/Marchesi/Gatti, ArXiv'20]

* If latencies are polynomial of degree $\leq d$, then $\texttt{MinSC}$ is NP-hard to approx within a factor $(\beta d)^{\frac{d}{2}}$, for some $\beta > 0$ [Roughgarden, FOCS'12]

Take-away: so far no tight computational lower bound

Hardness of approximation – main result

Hardness of approximation – main result

Theorem: In congestion games with resource costs identical to $b(\cdot)$

Hardness of approximation – main result

Theorem: In congestion games with resource costs identical to $b(\cdot)$, `MinSC` is NP-hard to approximate within any factor smaller than

$$\rho_b = \sup_{x \in \mathbb{N}} \frac{\mathbb{E}_{P \sim Poi(x)}[Pb(P)]}{xb(x)}$$

Hardness of approximation – main result

Theorem: In congestion games with resource costs identical to $b(\cdot)$, `MinSC` is NP-hard to approximate within any factor smaller than

$$\rho_b = \sup_{x \in \mathbb{N}} \frac{\mathbb{E}_{P \sim Poi(x)}[Pb(P)]}{xb(x)}$$

Extension to resource costs produced by non-negative combinations of functions $b_1, \ldots, b_m$ obtained replacing ρ_b with $\max_j \rho_{b_j}$

Hardness of approximation – main result

Theorem: In congestion games with resource costs identical to $b(\cdot)$, `MinSC` is NP-hard to approximate within any factor smaller than

$$\rho_b = \sup_{x \in \mathbb{N}} \frac{\mathbb{E}_{P \sim Poi(x)}[Pb(P)]}{xb(x)}$$

Extension to resource costs produced by non-negative combinations of functions $b_1, \ldots, b_m$ obtained replacing ρ_b with $\max_j \rho_{b_j}$

Corollary: In polynomial congestion games of max degree d

$$\max_j \rho_{b_j} = (d+1)\text{'st Bell number}$$

Hardness of approximation – main result

Theorem: In congestion games with resource costs identical to $b(\cdot)$, `MinSC` is NP-hard to approximate within any factor smaller than

$$\rho_b = \sup_{x \in \mathbb{N}} \frac{\mathbb{E}_{P \sim Poi(x)}[Pb(P)]}{xb(x)}$$

Extension to resource costs produced by non-negative combinations of functions $b_1, \ldots, b_m$ obtained replacing ρ_b with $\max_j \rho_{b_j}$

Corollary: In polynomial congestion games of max degree d

$$\max_j \rho_{b_j} = (d+1)\text{'st Bell number}$$

For example $d = 1$ corresponds to $\mathcal{B}(d+1) = 2$

Hardness of approximation – main result

Theorem: In congestion games with resource costs identical to $b(\cdot)$, `MinSC` is NP-hard to approximate within any factor smaller than

$$\rho_b = \sup_{x \in \mathbb{N}} \frac{\mathbb{E}_{P \sim Poi(x)}[Pb(P)]}{xb(x)}$$

Extension to resource costs produced by non-negative combinations of functions $b_1, \ldots, b_m$ obtained replacing ρ_b with $\max_j \rho_{b_j}$

Corollary: In polynomial congestion games of max degree d

$$\max_j \rho_{b_j} = (d+1)\text{'st Bell number}$$

For example $d = 1$ corresponds to $\mathcal{B}(d+1) = 2$
$d = 2$ corresponds to $\mathcal{B}(d+1) = 5$

$\vdots$

Proof Ideas

Proof Ideas

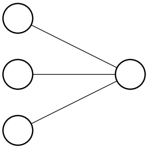
Reduction from **Gap-label-cover** to CG using **partitioning system**

Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph



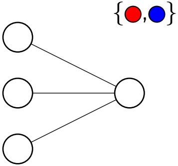
Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph

- palette of colors

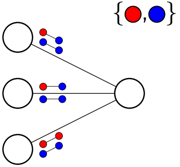


Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph
- palette of colors
- set of constraints

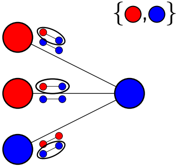


Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph
- palette of colors
- set of constraints

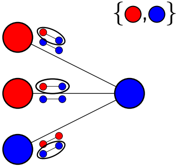


Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph
- palette of colors
- set of constraints



Partitioning system generalizes [Feige, JACM'98], used in [Barman/Fawzi/Fermé, STACS'21]

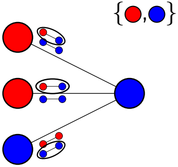
- resources $\mathcal{R}$, cost $b(\cdot)$

Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

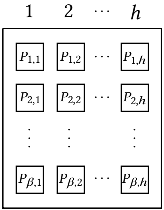
Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph
- palette of colors
- set of constraints



Partitioning system generalizes [Feige, JACM'98], used in [Barman/Fawzi/Fermé, STACS'21]

- resources $\mathcal{R}$, cost $b(\cdot)$
- subsets $P_{i,j} \subseteq \mathcal{R}$

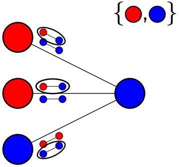


Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

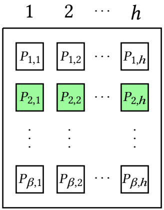
Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph
- palette of colors
- set of constraints



Partitioning system generalizes [Feige, JACM'98], used in [Barman/Fawzi/Fermé, STACS'21]

- resources $\mathcal{R}$, cost $b(\cdot)$
- subsets $P_{i,j} \subseteq \mathcal{R}$
- $SC(row)$

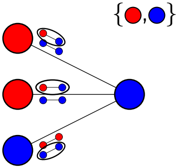


Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

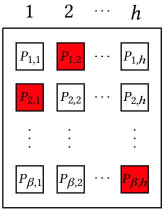
Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

- bi-partite graph
- palette of colors
- set of constraints



Partitioning system generalizes [Feige, JACM'98], used in [Barman/Fawzi/Fermé, STACS'21]

- resources $\mathcal{R}$, cost $b(\cdot)$
- subsets $P_{i,j} \subseteq \mathcal{R}$
- $SC(row)$, $SC(scr)$

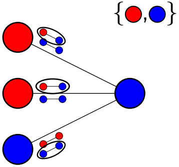


Proof Ideas

Reduction from **Gap-label-cover** to CG using **partitioning system**

Gap-label-cover [Feige JACM'98; Dudyciz/Manurangsi/Marcinkowski/Sornat IJCAI'20]

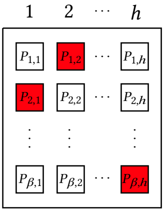
- bi-partite graph
- palette of colors
- set of constraints

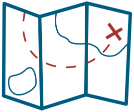


Partitioning system generalizes [Feige, JACM'98], used in [Barman/Fawzi/Fermé, STACS'21]

- resources $\mathcal{R}$, cost $b(\cdot)$
- subsets $P_{i,j} \subseteq \mathcal{R}$
- $SC(row)$, $SC(scr)$ satisfy

$$\frac{SC(scr)}{SC(row)} \approx \rho_b$$





Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\implies$ first poly algo optimal approx

Poly-time algorithm based on taxes

Poly-time algorithm based on taxes

Background:

– price of anarchy measures equilibrium quality, e.g., $SC(a^{\mathrm{NE}})/SC(a^{\mathrm{OPT}})$

[Koutsoupias/Papadimitriou STACS'99; Christodoulou/Koutsoupias STOC'05; Aland/Dumrauf/Gairing/Monien/Schoppmann, STACS'06; Roughgarden JACM'15]

Poly-time algorithm based on taxes

Background:

– price of anarchy measures equilibrium quality, e.g., $SC(a^{\mathrm{NE}})/SC(a^{\mathrm{OPT}})$

[Koutsoupias/Papadimitriou STACS'99; Christodoulou/Koutsoupias STOC'05; Aland/Dumrauf/Gairing/Monien/Schoppmann, STACS'06; Roughgarden JACM'15]

– efficient computation of CE/CCE

[Papadimitriou/Roughgarden, JACM'08; Xin Jiang/Leyton-Brown, GEB'15; Hart/Mas-Colell, Econometrica'00; Blum/Hajiaghayi/Ligett/Roth, STOC'08]

Poly-time algorithm based on taxes

Background:

– price of anarchy measures equilibrium quality, e.g., $SC(a^{\mathrm{NE}})/SC(a^{\mathrm{OPT}})$

[Koutsoupias/Papadimitriou STACS'99; Christodoulou/Koutsoupias STOC'05; Aland/Dumrauf/Gairing/Monien/Schoppmann, STACS'06; Roughgarden JACM'15]

– efficient computation of CE/CCE

[Papadimitriou/Roughgarden, JACM'08; Xin Jiang/Leyton-Brown, GEB'15; Hart/Mas-Colell, Econometrica'00; Blum/Hajiaghayi/Ligett/Roth, STOC'08]

Price of anarchy as approximation ratio

Poly-time algorithm based on taxes

Background:

– price of anarchy measures equilibrium quality, e.g., $SC(a^{\mathrm{NE}})/SC(a^{\mathrm{OPT}})$

[Koutsoupias/Papadimitriou STACS'99; Christodoulou/Koutsoupias STOC'05; Aland/Dumrauf/Gairing/Monien/Schoppmann, STACS'06; Roughgarden JACM'15]

– efficient computation of CE/CCE

[Papadimitriou/Roughgarden, JACM'08; Xin Jiang/Leyton-Brown, GEB'15; Hart/Mas-Colell, Econometrica'00; Blum/Hajiaghayi/Ligett/Roth, STOC'08]

Price of anarchy as approximation ratio

$\rightsquigarrow$ **Q:** How to improve PoA?

Poly-time algorithm based on taxes

Background:

– price of anarchy measures equilibrium quality, e.g., $SC(a^{\mathrm{NE}})/SC(a^{\mathrm{OPT}})$

[Koutsoupias/Papadimitriou STACS'99; Christodoulou/Koutsoupias STOC'05; Aland/Dumrauf/Gairing/Monien/Schoppmann, STACS'06; Roughgarden JACM'15]

– efficient computation of CE/CCE

[Papadimitriou/Roughgarden, JACM'08; Xin Jiang/Leyton-Brown, GEB'15; Hart/Mas-Colell, Econometrica'00; Blum/Hajiaghayi/Ligett/Roth, STOC'08]

Price of anarchy as approximation ratio

$\rightsquigarrow$ **Q:** How to improve PoA?

* coordination mechanisms: [Christodolou/Koutsoupias/Nanavati, ICALP'04 ...]
* Stackelberg strategies: [Fotakis, ESA'04; Swamy, SODA'07 ...]
* information provision: [Bhaskar/Cheng/Kun Ko/Swamy, EC'16; Nachbar/Xu ArXiv'20 ...]
* cost sharing: [Gkatzelis/Kollias/Roughgarden, WINE'14; Chen/Roughgarden/Valiant, J Comput'10 ...]
* taxes: [Caragiannis/Kaklamanis/Kannellopoulos, Trans Alg'10; Bilò/Vinci, EC'16 ...]

Poly-time algorithm based on taxes

Background:

– price of anarchy measures equilibrium quality, e.g., $SC(a^{\mathrm{NE}})/SC(a^{\mathrm{OPT}})$

[Koutsoupias/Papadimitriou STACS'99; Christodoulou/Koutsoupias STOC'05; Aland/Dumrauf/Gairing/Monien/Schoppmann, STACS'06; Roughgarden JACM'15]

– efficient computation of CE/CCE

[Papadimitriou/Roughgarden, JACM'08; Xin Jiang/Leyton-Brown, GEB'15; Hart/Mas-Colell, Econometrica'00; Blum/Hajiaghayi/Ligett/Roth, STOC'08]

Price of anarchy as approximation ratio

$\rightsquigarrow$ **Q:** How to improve PoA?

* coordination mechanisms: [Christodolou/Koutsoupias/Nanavati, ICALP'04 ...]
* Stackelberg strategies: [Fotakis, ESA'04; Swamy, SODA'07 ...]
* information provision: [Bhaskar/Cheng/Kun Ko/Swamy, EC'16; Nachbar/Xu ArXiv'20 ...]
* cost sharing: [Gkatzelis/Kollias/Roughgarden, WINE'14; Chen/Roughgarden/Valiant, J Comput'10 ...]
* taxes: [Caragiannis/Kaklamanis/Kannellopoulos, Trans Alg'10; Bilò/Vinci, EC'16 ...]

Polynomial time algorithms – related work

Polynomial time algorithms – related work

* Best-known approx (LP + rounding) [Makarychev/Sviridenko, JACM'18]

$$\sup_{x \in \mathbb{R}_{>0}} \frac{\mathbb{E}_{P \sim Poi(1)}[(xP)b(xP)]}{xb(x)}$$

Polynomial time algorithms – related work

* Best-known approx (LP + rounding) [Makarychev/Sviridenko, JACM'18]

$$\sup_{x \in \mathbb{R}_{>0}} \frac{\mathbb{E}_{P \sim Poi(1)}[(xP)b(xP)]}{xb(x)} \geq \text{NP-hardness factor}$$

Polynomial time algorithms – related work

* Best-known approx (LP + rounding) [Makarychev/Sviridenko, JACM'18]

$$\sup_{x \in \mathbb{R}_{>0}} \frac{\mathbb{E}_{P \sim Poi(1)}[(xP)b(xP)]}{xb(x)} \geq \text{NP-hardness factor}$$

* For polynomial costs, taxes achieve $\mathrm{PoA} = \mathcal{B}(\mathrm{d}+1)$
  [Caragiannis/Kaklamanis/Kannellopoulos, Trans Alg'10; Bilò/Vinci, EC'16]

Polynomial time algorithms – related work

* Best-known approx (LP + rounding) [Makarychev/Sviridenko, JACM'18]

$$\sup_{x \in \mathbb{R}_{>0}} \frac{\mathbb{E}_{P \sim Poi(1)}[(xP)b(xP)]}{xb(x)} \geq \text{NP-hardness factor}$$

* For polynomial costs, taxes achieve $\mathrm{PoA} = \mathcal{B}(\mathrm{d}+1)$
  [Caragiannis/Kaklamanis/Kannellopoulos, Trans Alg'10; Bilò/Vinci, EC'16]

Take-away: so far no matching approx in general

Matching polynomial time algorithm – main result

Matching polynomial time algorithm – main result

Theorem: Consider congestion games where all resource costs are equal to $b(\cdot)$, positive, non-decreasing, semi-convex.

Matching polynomial time algorithm – main result

Theorem: Consider congestion games where all resource costs are equal to $b(\cdot)$, positive, non-decreasing, semi-convex. For any $\varepsilon > 0$, it is possible to efficiently compute a taxation mechanism so that

$$\mathrm{PoA}_{\mathrm{CCE}} \leq \rho_{\mathrm{b}} + \varepsilon$$

Matching polynomial time algorithm – main result

Theorem: Consider congestion games where all resource costs are equal to $b(\cdot)$, positive, non-decreasing, semi-convex. For any $\varepsilon > 0$, it is possible to efficiently compute a taxation mechanism so that

$$\mathrm{PoA}_{\mathrm{CCE}} \leq \rho_{\mathrm{b}} + \varepsilon$$

Extends to resource costs obtained by non-negative combo of $b_1, \ldots, b_m$

Matching polynomial time algorithm – main result

Theorem: Consider congestion games where all resource costs are equal to $b(\cdot)$, positive, non-decreasing, semi-convex. For any $\varepsilon > 0$, it is possible to efficiently compute a taxation mechanism so that

$$\mathrm{PoA}_{\mathrm{CCE}} \leq \rho_{\mathrm{b}} + \varepsilon$$

Extends to resource costs obtained by non-negative combo of $b_1, \ldots, b_m$

Corollary: For any $\varepsilon > 0$, there exists a polynomial time algorithm producing an allocation a^* with cost

$$SC(a^*) \leq (\max_j \rho_{b_j} + \varepsilon) \cdot OPT$$

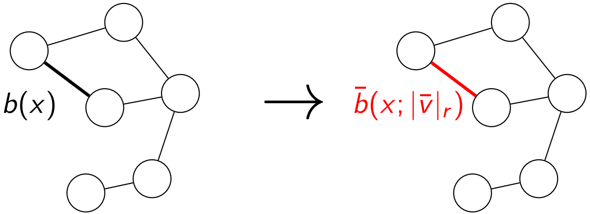
Matching polynomial time algorithm - Proof Ideas

Matching polynomial time algorithm - Proof Ideas

$$SC(a) = \sum_{r \in a} |a|_r b(|a|_r) \qquad SC_P(a) = \sum_{r \in a} \mathbb{E}_{P \sim \mathrm{Poi}(|a|_r)}[P b(P)]$$

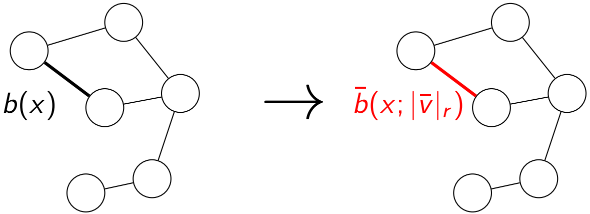
Matching polynomial time algorithm - Proof Ideas

$$SC(a) = \sum_{r \in a} |a|_r b(|a|_r) \qquad SC_P(a) = \sum_{r \in a} \mathbb{E}_{P \sim \mathrm{Poi}(|a|_r)}[P b(P)]$$



Matching polynomial time algorithm - Proof Ideas

$$SC(a) = \sum_{r \in a} |a|_r b(|a|_r) \qquad SC_P(a) = \sum_{r \in a} \mathbb{E}_{P \sim \mathrm{Poi}(|a|_r)}[Pb(P)]$$

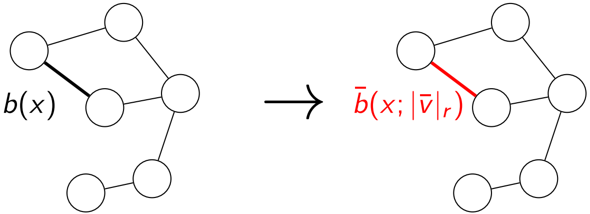


Key ingredients:

P1: $\bar{b}(x; v)$ solves crucial recursion

Matching polynomial time algorithm - Proof Ideas

$$SC(a) = \sum_{r \in a} |a|_r b(|a|_r) \qquad SC_P(a) = \sum_{r \in a} \mathbb{E}_{P \sim \mathrm{Poi}(|a|_r)}[Pb(P)]$$



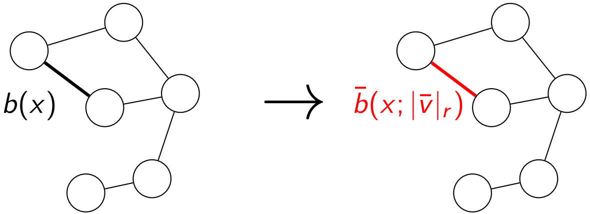
Key ingredients:

P1: $\bar{b}(x; v)$ solves crucial recursion

P2: $\bar{v}$ solves continuous relaxation of $\min SC_P(a)$

Matching polynomial time algorithm - Proof Ideas

$$SC(a) = \sum_{r \in a} |a|_r b(|a|_r) \qquad SC_P(a) = \sum_{r \in a} \mathbb{E}_{P \sim \mathrm{Poi}(|a|_r)}[Pb(P)]$$



Key ingredients:

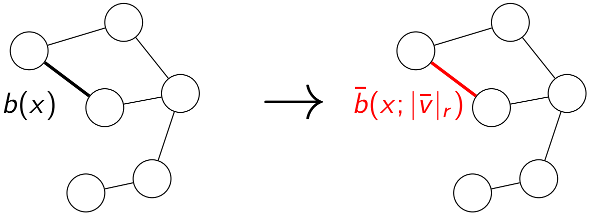
P1: $\bar{b}(x; v)$ solves crucial recursion

P2: $\bar{v}$ solves continuous relaxation of $\min SC_P(a)$

$$SC(a^{\mathrm{NE}}) \overset{P1}{\leq} SC_P(\bar{v})$$

Matching polynomial time algorithm - Proof Ideas

$$SC(a) = \sum_{r \in a} |a|_r b(|a|_r) \qquad SC_P(a) = \sum_{r \in a} \mathbb{E}_{P \sim \mathrm{Poi}(|a|_r)}[Pb(P)]$$



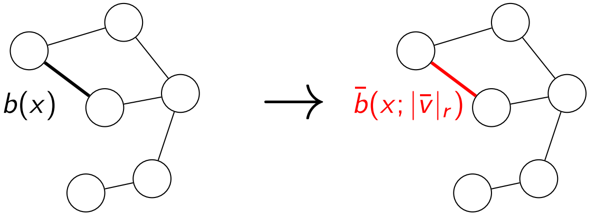
Key ingredients:

P1: $\bar{b}(x; v)$ solves crucial recursion
P2: $\bar{v}$ solves continuous relaxation of min $SC_P(a)$

$$SC(a^{\mathrm{NE}}) \overset{P1}{\leq} SC_P(\bar{v}) \overset{P2}{\leq} SC_P(a^{\mathrm{OPT}})$$

Matching polynomial time algorithm - Proof Ideas

$$SC(a) = \sum_{r \in a} |a|_r b(|a|_r) \qquad SC_P(a) = \sum_{r \in a} \mathbb{E}_{P \sim \mathrm{Poi}(|a|_r)}[Pb(P)]$$



Key ingredients:

P1: $\bar{b}(x; v)$ solves crucial recursion

P2: $\bar{v}$ solves continuous relaxation of min $SC_P(a)$

$$SC(a^{\mathrm{NE}}) \overset{P1}{\leq} SC_P(\bar{v}) \overset{P2}{\leq} SC_P(a^{\mathrm{OPT}}) \overset{\text{def } \rho_b}{\leq} \rho_b SC(a^{\mathrm{OPT}})$$

Conclusion and open questions

Conclusion and open questions

Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\Longrightarrow$ first poly algo optimal approx

Conclusion and open questions

Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\implies$ first poly algo optimal approx

Remarks:

* Competitive decision making + incentives = best-centralized

Conclusion and open questions

Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\Longrightarrow$ first poly algo optimal approx

Remarks:

* Competitive decision making + incentives = best-centralized
* Surprising that "*taxes are enough*"

Conclusion and open questions

Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\Longrightarrow$ first poly algo optimal approx

Remarks:

* Competitive decision making + incentives = best-centralized
* Surprising that "*taxes are enough*"
* Poly-time algo requires centralized solution of cvx opt

Conclusion and open questions

Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\Longrightarrow$ first poly algo optimal approx

Remarks:

* Competitive decision making $+$ incentives $=$ best-centralized
* Surprising that "*taxes are enough*"

* Poly-time algo requires centralized solution of cvx opt
  If undesirable $\rightsquigarrow$ optimal local tax [Paccagnan/Chandan/Ferguson/Marden, TEAC'21]
  very little performance loss, e.g., 2.012 vs 2 for affine

Conclusion and open questions

Problem: minimum social cost in atomic congestion games

Main result I: tight NP-hardness of approximation

Main result II: taxes achieve matching approximation
$\Longrightarrow$ first poly algo optimal approx

Remarks:

* Competitive decision making $+$ incentives $=$ best-centralized
* Surprising that "*taxes are enough*"

* Poly-time algo requires centralized solution of cvx opt
  If undesirable $\rightsquigarrow$ optimal local tax [Paccagnan/Chandan/Ferguson/Marden, TEAC'21]
  very little performance loss, e.g., 2.012 vs 2 for affine

* Main result II extends to network CG

“Judiciously designed taxes achieve optimal approximation, and no other tractable intervention can improve upon this result”