DFG Research Center MATHEON
Mathematics for key technologies

An Efficient Cost Sharing Mechanism for the Prize-Collecting Steiner Forest Problem

Guido Schäfer
Institute of Mathematics, TU Berlin, Germany

Workshop on Flexible Network Design
Bertinoro, October 2–6, 2006

joint work with: A. Gupta, J. Könemann, S. Leonardi, R. Ravi

Outline

- Part I: Cost Sharing Mechanisms
  - cost sharing model, definitions, objectives
  - state of affairs, new trade-offs
  - tricks of the trade
- Part II: Prize-Collecting Steiner Forest
  - primal-dual algorithm PCSF
  - cross-monotonicity and budget balance
  - general reduction technique
- Conclusions and Open Problems

Part I

Cost Sharing Mechanisms

Cost Sharing Model

Service provider: offers some service

- set U of n potential users, interested in service
- cost function $C : 2^U \rightarrow \mathbb{R}^+$
  $C(S) =$ cost to serve user-set $S \subseteq U$
- determines who receives service and distributes cost

Every user $i \in U$:

- has a (private) utility $u_i \geq 0$ for receiving the service
- announces bid $b_i \geq 0$, the maximum amount he is willing to pay for the service

Cost Sharing Model

Service provider: offers some service

- set U of n potential users, interested in service
- cost function $C : 2^U \rightarrow \mathbb{R}^+$
  $C(S) =$ cost to serve user-set $S \subseteq U$
- determines who receives service and distributes cost

Every user $i \in U$:

- has a (private) utility $u_i \geq 0$ for receiving the service
- announces bid $b_i \geq 0$, the maximum amount he is willing to pay for the service

Cost Sharing Mechanism

Cost sharing mechanism M:

- collects all bids $\{b_i\}_{i \in U}$ from users
- decides a set $S^M \subseteq U$ of users that receive service
- determines a payment p_i for every user $i \in S^M$

Properties:

1. user is not paid for receiving service
2. user is charged at most his bid if he receives service, zero otherwise
3. user receives service if his bid is large enough

Cost Sharing Mechanism

Cost sharing mechanism M:

- collects all bids $\{b_i\}_{i \in U}$ from users
- decides a set $S^M \subseteq U$ of users that receive service
- determines a payment p_i for every user $i \in S^M$

Properties:

1. user is not paid for receiving service
2. user is charged at most his bid if he receives service, zero otherwise
3. user receives service if his bid is large enough

Budget Balance

β-budget balance: total payment of users in S^M approximates overall cost

$$C(S^M) \leq \sum_{i \in S^M} p_i \leq \beta \cdot C(S^M), \quad \beta \geq 1$$

Truthfullness

Benefit: user i receives benefit $u_i - p_i$ if served, zero otherwise

Strategic behaviour: every user $i \in U$ acts selfishly and attempts to maximize his benefit (using his bid)

Strategyproofness: benefit of every user $i \in U$ is maximized if he bids truthfully, i.e., bidding $b_i = u_i$ is a dominant strategy for every user $i \in U$

Group-strategyproofness: same holds true even if users form coalitions and coordinate their biddings

Truthfullness

Benefit: user i receives benefit $u_i - p_i$ if served, zero otherwise

Strategic behaviour: every user $i \in U$ acts selfishly and attempts to maximize his benefit (using his bid)

Strategyproofness: benefit of every user $i \in U$ is maximized if he bids truthfully, i.e., bidding $b_i = u_i$ is a dominant strategy for every user $i \in U$

Group-strategyproofness: same holds true even if users form coalitions and coordinate their biddings

Truthfullness

Benefit: user i receives benefit $u_i - p_i$ if served, zero otherwise

Strategic behaviour: every user $i \in U$ acts selfishly and attempts to maximize his benefit (using his bid)

Strategyproofness: benefit of every user $i \in U$ is maximized if he bids truthfully, i.e., bidding $b_i = u_i$ is a dominant strategy for every user $i \in U$

Group-strategyproofness: same holds true even if users form coalitions and coordinate their biddings

Truthfullness

Benefit: user i receives benefit $u_i - p_i$ if served, zero otherwise

Strategic behaviour: every user $i \in U$ acts selfishly and attempts to maximize his benefit (using his bid)

Strategyproofness: benefit of every user $i \in U$ is maximized if he bids truthfully, i.e., bidding $b_i = u_i$ is a dominant strategy for every user $i \in U$

Group-strategyproofness: same holds true even if users form coalitions and coordinate their biddings

Efficiency

Social welfare: for a set $S \subseteq U$, define

$$W(S) := \sum_{i \in S} u_i - C(S)$$

α-efficiency: assuming truthfull bidding, social welfare of S^M approximates maximum social welfare

$$W(S^M) \geq \frac{1}{\alpha} \cdot W(S) \quad \forall S \subseteq U, \quad \alpha \geq 1$$

Efficiency

Social welfare: for a set $S \subseteq U$, define

$$W(S) := \sum_{i \in S} u_i - C(S)$$

α-efficiency: assuming truthfull bidding, social welfare of S^M approximates maximum social welfare

$$W(S^M) \geq \frac{1}{\alpha} \cdot W(S) \quad \forall S \subseteq U, \quad \alpha \geq 1$$

Bad News...

Exact: truthfullness $+$ budget balance $+$ efficiency
$=$ impossible [Green et al. '76], [Roberts '79]

Approximate: truthfullness $+$ approximate budget balance $+$ approximate efficiency
$=$ impossible [Feigenbaum et al. '03]

Remark: impossibility results hold even for strategyproofness and simple cost functions

Consequence: researchers concentrated on proper subsets of these objectives

Bad News...

Exact: truthfullness $+$ budget balance $+$ efficiency
$=$ impossible [Green et al. '76], [Roberts '79]

Approximate: truthfullness $+$ approximate budget balance $+$ approximate efficiency
$=$ impossible [Feigenbaum et al. '03]

Remark: impossibility results hold even for strategyproofness and simple cost functions

Consequence: researchers concentrated on proper subsets of these objectives

Bad News...

Exact: truthfullness $+$ budget balance $+$ efficiency
$=$ impossible [Green et al. '76], [Roberts '79]

Approximate: truthfullness $+$ approximate budget balance $+$ approximate efficiency
$=$ impossible [Feigenbaum et al. '03]

Remark: impossibility results hold even for strategyproofness and simple cost functions

Consequence: researchers concentrated on proper subsets of these objectives

Bad News...

Exact: truthfullness + budget balance + efficiency
= impossible [Green et al. '76], [Roberts '79]

Approximate: truthfullness + approximate budget balance + approximate efficiency
= impossible [Feigenbaum et al. '03]

Remark: impossibility results hold even for strategyproofness and simple cost functions

Consequence: researchers concentrated on proper subsets of these objectives

Cost Sharing Mechanisms

Authors	Problem	β
[Moulin, Shenker '01]	submodular cost	1
[Jain, Vazirani '01]	MST	1
	Steiner tree and TSP	2
[Devanur, Mihail, Vazirani '03]	set cover	$\log n$
(strategyproof only)	facility location	1.61
[Pal, Tardos '03]	facility location	3
	SRoB	15
[Leonardi, S. '03], [Gupta et al. '03]	SRoB	4
[Leonardi, S. '03]	CFL	30
[Könemann, Leonardi, S. '05]	Steiner forest	2
Lower bounds		
[Immorlica, Mahdian, Mirrokni '05]	edge cover	2
	facility location	3
	vertex cover	$n^{1/3}$
	set cover	n
[Könemann, Leonardi, S., van Zwam '05]	Steiner tree	2

Alternative Efficiency Measure: Social Cost

Social cost: for a set $S \subseteq U$, define

$$\Pi(S) := \sum_{i \notin S} u_i + C(S)$$

$$= \sum_{i \in U} u_i - \sum_{i \in S} u_i + C(S) = -W(S) + \sum_{i \in U} u_i$$

Thus: S maximizes $W(S)$ iff S minimizes $\Pi(S)$

α-approximate: approximate minimimum social cost

$$\Pi(S^M) \leq \alpha \cdot \Pi(S) \quad \forall S \subseteq U, \quad \alpha \geq 1$$

[Roughgarden and Sundararajan '06]

Alternative Efficiency Measure: Social Cost

Social cost: for a set $S \subseteq U$, define

$$\Pi(S) := \sum_{i \notin S} u_i + C(S)$$
$$= \sum_{i \in U} u_i - \sum_{i \in S} u_i + C(S) = -W(S) + \sum_{i \in U} u_i$$

Thus: S maximizes $W(S)$ iff S minimizes $\Pi(S)$

α-approximate: approximate minimimum social cost

$$\Pi(S^M) \leq \alpha \cdot \Pi(S) \quad \forall S \subseteq U, \quad \alpha \geq 1$$

[Roughgarden and Sundararajan '06]

Alternative Efficiency Measure: Social Cost

Social cost: for a set $S \subseteq U$, define

$$\Pi(S) := \sum_{i \notin S} u_i + C(S)$$
$$= \sum_{i \in U} u_i - \sum_{i \in S} u_i + C(S) = -W(S) + \sum_{i \in U} u_i$$

Thus: S maximizes $W(S)$ iff S minimizes $\Pi(S)$

α-approximate: approximate minimimum social cost

$$\Pi(S^M) \leq \alpha \cdot \Pi(S) \quad \forall S \subseteq U, \quad \alpha \geq 1$$

[Roughgarden and Sundararajan '06]

Alternative Efficiency Measure: Social Cost

Social cost: for a set $S \subseteq U$, define

$$\Pi(S) := \sum_{i \notin S} u_i + C(S)$$
$$= \sum_{i \in U} u_i - \sum_{i \in S} u_i + C(S) = -W(S) + \sum_{i \in U} u_i$$

Thus: S maximizes $W(S)$ iff S minimizes $\Pi(S)$

α-approximate: approximate minimimum social cost

$$\Pi(S^M) \leq \alpha \cdot \Pi(S) \quad \forall S \subseteq U, \quad \alpha \geq 1$$

[Roughgarden and Sundararajan '06]

Alternative Efficiency Measure: Social Cost

Social cost: for a set $S \subseteq U$, define

$$\Pi(S) := \sum_{i \notin S} u_i + C(S)$$
$$= \sum_{i \in U} u_i - \sum_{i \in S} u_i + C(S) = -W(S) + \sum_{i \in U} u_i$$

Thus: S maximizes $W(S)$ iff S minimizes $\Pi(S)$

α-approximate: approximate minimimum social cost

$$\Pi(S^M) \leq \alpha \cdot \Pi(S) \quad \forall S \subseteq U, \quad \alpha \geq 1$$

[Roughgarden and Sundararajan '06]

Cost Sharing Mechanisms

Authors	Problem	β	α
[Roughgarden, Sundararajan '06]	submodular cost	1	$\Theta(\log n)$
	Steiner tree	2	$\Theta(\log^2 n)$
[Chawla, Roughgarden, Sundararajan '06]	Steiner forest	2	$\Theta(\log^2 n)$
[Roughgarden, Sundararajan ?]	facility location	3	$\Theta(\log n)$
	SRoB	4	$\Theta(\log^2 n)$
[Gupta et al. '07]	prize-collecting Steiner forest	3	$\Theta(\log^2 n)$

How to achieve

β-budget balance?

$$\left(C(S) \leq \sum_{i \in S^M} p_i \leq \beta \cdot C(S) \right)$$

... use techniques from approximation algorithms

How to achieve

group-strategyproofness?

(not everybody in the coalition is better off by misreporting his utility)

Cross-Monotonic Cost Sharing Method

Cost sharing method: function $\xi : U \times 2^U \rightarrow \mathbb{R}^+$
$\xi(i, S) =$ cost share of user i with respect to set $S \subseteq U$

β-budget balance:

$$C(S) \leq \sum_{i \in S} \xi(i, S) \leq \beta \cdot C(S) \quad \forall S \subseteq U$$

Cross-monotonicity: cost share of user i does not increase as additional users join the game:

$$\forall S' \subseteq S, \forall i \in S' : \quad \xi(i, S') \geq \xi(i, S)$$

Cross-Monotonic Cost Sharing Method

Cost sharing method: function $\xi : U \times 2^U \to \mathbb{R}^+$
$\xi(i,S) =$ cost share of user i with respect to set $S \subseteq U$

β-budget balance:

$$C(S) \leq \sum_{i \in S} \xi(i,S) \leq \beta \cdot C(S) \quad \forall S \subseteq U$$

Cross-monotonicity: cost share of user i does not increase as additional users join the game:

$$\forall S' \subseteq S,\ \forall i \in S' : \quad \xi(i,S') \geq \xi(i,S)$$

Moulin Mechanism

Given: cross-monotonic and β-budget balanced cost sharing method ξ

Moulin mechanism $M(\xi)$:

1: Initialize: $S^M \leftarrow U$
2: If for each user $i \in S^M$: $\xi(i, S^M) \leq b_i$ then STOP
3: Otherwise, remove from S^M all users with $\xi(i, S^M) > b_i$ and repeat

Thm: Moulin mechanism $M(\xi)$ is a group-strategyproof cost sharing mechanism that is β-budget balanced

[Moulin, Shenker '01], [Jain, Vazirani '01]

How to achieve

α-approximability?

$$\left(\Pi(S^M) \leq \frac{1}{\alpha} \cdot \Pi(S) \quad \forall S \subseteq U \right)$$

Summability of Cost Sharing Methods

Given: arbitrary order σ on users in U

Order subset $S \subseteq U$ according to σ:

$$S := \{i_1, \ldots, i_{|S|}\}$$

Let $S_j :=$ first j users of S

α-summability: ξ is α-summable if

$$\forall \sigma, \forall S \subseteq U: \quad \sum_{j=1}^{|S|} \xi(i_j, S_j) \leq \alpha \cdot C(S)$$

Summability of Cost Sharing Methods

Given: arbitrary order σ on users in U

Order subset $S \subseteq U$ according to σ:

$$S := \{i_1, \ldots, i_{|S|}\}$$

Let $S_j :=$ first j users of S

α-summability: ξ is α-summable if

$$\forall \sigma, \forall S \subseteq U : \quad \sum_{j=1}^{|S|} \xi(i_j, S_j) \leq \alpha \cdot C(S)$$

Summability of Cost Sharing Methods

Given: arbitrary order σ on users in U

Order subset $S \subseteq U$ according to σ:

$$S := \{i_1, \ldots, i_{|S|}\}$$

Let $S_j :=$ first j users of S

α-summability: ξ is α-summable if

$$\forall \sigma, \forall S \subseteq U : \quad \sum_{j=1}^{|S|} \xi(i_j, S_j) \leq \alpha \cdot C(S)$$

Summability of Cost Sharing Methods

Given: arbitrary order σ on users in U

Order subset $S \subseteq U$ according to σ:

$$S := \{i_1, \ldots, i_{|S|}\}$$

Let $S_j :=$ first j users of S

α-summability: ξ is α-summable if

$$\forall \sigma, \forall S \subseteq U : \quad \sum_{j=1}^{|S|} \xi(i_j, S_j) \leq \alpha \cdot C(S)$$

Summability implies Approximability

Given: cross-monotonic cost sharing method ξ that satisfies β-budget balance and α-summability

Thm: Moulin mechanism $M(\xi)$ is a group-strategyproof cost sharing mechanism that is β-budget balanced and $(\alpha+\beta)$-approximate

[Roughgarden, Sundararajan '06]

Summability implies Approximability

Given: cross-monotonic cost sharing method ξ that satisfies β-budget balance and α-summability

Thm: Moulin mechanism $M(\xi)$ is a group-strategyproof cost sharing mechanism that is β-budget balanced and $(\alpha+\beta)$-approximate

[Roughgarden, Sundararajan '06]

Part II

Prize-Collecting Steiner Forest Problem

Prize-Collecting Steiner Forest Problem (PCSF)

Given:

- network $N = (V, E, c)$ with edge costs $c : E \to \mathbb{R}^+$
- set of n terminal pairs $R = \{(s_1, t_1), \dots, (s_n, t_n)\} \subseteq V \times V$
- penalty $\pi_i \geq 0$ for every pair $(s_i, t_i) \in R$.

Feasible solution: forest F and subset $Q \subseteq R$ such that for all $(s_i, t_i) \in R$: either s_i, t_i are connected in F, or $(s_i, t_i) \in Q$

Objective: compute feasible solution (F, Q) such that $c(F) + \pi(Q)$ is minimized

Prize-Collecting Steiner Forest Problem (PCSF)

Given:

- network $N = (V, E, c)$ with edge costs $c : E \to \mathbb{R}^+$
- set of n terminal pairs $R = \{(s_1, t_1), \ldots, (s_n, t_n)\} \subseteq V \times V$
- penalty $\pi_i \geq 0$ for every pair $(s_i, t_i) \in R$.

Feasible solution: forest F and subset $Q \subseteq R$ such that for all $(s_i, t_i) \in R$: either s_i, t_i are connected in F, or $(s_i, t_i) \in Q$

Objective: compute feasible solution (F, Q) such that $c(F) + \pi(Q)$ is minimized

Prize-Collecting Steiner Forest Problem (PCSF)

Given:

- network $N = (V, E, c)$ with edge costs $c : E \to \mathbb{R}^+$
- set of n terminal pairs $R = \{(s_1, t_1), \ldots, (s_n, t_n)\} \subseteq V \times V$
- penalty $\pi_i \geq 0$ for every pair $(s_i, t_i) \in R$.

Feasible solution: forest F and subset $Q \subseteq R$ such that for all $(s_i, t_i) \in R$: either s_i, t_i are connected in F, or $(s_i, t_i) \in Q$

Objective: compute feasible solution (F, Q) such that $c(F) + \pi(Q)$ is minimized

PCSF in a Cost Sharing Context

- every user is associated with a terminal pair: $U = R$
- user i wants to connect s_i and t_i
- service provider can either build this connection himself, or buy connection at a price of π_i from another provider
- cost function $C(S)$ for user set $S \subseteq U$ is given by the cost of an optimal solution for PCSF(S)

PCSF in a Cost Sharing Context

- every user is associated with a terminal pair: $U = R$
- user i wants to connect s_i and t_i
- service provider can either build this connection himself, or buy connection at a price of π_i from another provider
- cost function $C(S)$ for user set $S \subseteq U$ is given by the cost of an optimal solution for PCSF(S)

PCSF in a Cost Sharing Context

- every user is associated with a terminal pair: $U = R$
- user i wants to connect s_i and t_i
- service provider can either build this connection himself, or buy connection at a price of π_i from another provider
- cost function $C(S)$ for user set $S \subseteq U$ is given by the cost of an optimal solution for PCSF(S)

PCSF in a Cost Sharing Context

- every user is associated with a terminal pair: $U = R$
- user i wants to connect s_i and t_i
- service provider can either build this connection himself, or buy connection at a price of π_i from another provider
- cost function $C(S)$ for user set $S \subseteq U$ is given by the cost of an optimal solution for PCSF(S)

Our Results

- cost sharing method ξ that is cross-monotonic and 3-budget balanced for PCSF
  Byproduct: simple primal-dual 3-approximate algorithm
- reduction technique that shows that Moulin mechanism $M(\xi)$ is $\Theta(\log^2 n)$-approximate
- simple proof of $O(\log^3 n)$-summability for Steiner forest cost sharing method

joint work with: A. Gupta, J. Könemann, S. Leonardi, R. Ravi
to appear in SODA 2007

LP Formulation

$$\begin{aligned}
\texttt{min} \quad & \sum_{e \in E} c_e \cdot x_e + \sum_{(u,\bar{u}) \in R} \pi(u,\bar{u}) \cdot x_{u\bar{u}} \\
\text{s.t.} \quad & \sum_{e \in \delta(S)} x_e + x_{u\bar{u}} \geq 1 \quad \forall S \in \mathscr{S}, \forall (u,\bar{u}) \odot S \\
& x_e \geq 0 \quad \forall e \in E \\
& x_{u\bar{u}} \geq 0 \quad \forall (u,\bar{u}) \in R
\end{aligned}$$

$\mathscr{S}$ = set of all Steiner cuts (separate at least one pair)

$\delta(S)$ = edges that cross cut defined by S

$(u,\bar{u}) \odot S$ = terminal pair $(u,\bar{u})$ separated by S

Dual LP

$$
\begin{aligned}
\texttt{max} \quad & \sum_{S \in \mathscr{S}} \sum_{(u,\bar{u}) \odot S} \xi_{S,u\bar{u}} \\
\text{s.t.} \quad & \sum_{S : e \in \delta(S)} \sum_{(u,\bar{u}) \odot S} \xi_{S,u\bar{u}} \leq c_e \quad \forall e \in E \\
& \sum_{S : (u,\bar{u}) \odot S} \xi_{S,u\bar{u}} \leq \pi(u,\bar{u}) \quad \forall (u,\bar{u}) \in R \\
& \xi_{S,u\bar{u}} \geq 0 \quad \forall S \in \mathscr{S}, \forall (u,\bar{u}) \odot S
\end{aligned}
$$

$\xi_{S,u\bar{u}} =$ cost share that $(u,\bar{u})$ receives from Steiner cut S

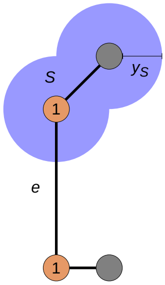
Dual LP — Simplified

$$\xi_{u\bar{u}} := \sum_{S:(u,\bar{u})\odot S} \xi_{S,u\bar{u}} \quad \text{(total cost share of } (u,\bar{u})\text{)}$$

$$y_S := \sum_{(u,\bar{u})\odot S} \xi_{S,u\bar{u}} \quad \text{(total dual of Steiner cut } S\text{)}$$

$$\begin{aligned} \texttt{max} \quad & \sum_{S\in\mathscr{S}} y_S \\ \text{s.t.} \quad & \sum_{S:e\in\delta(S)} y_S \le c_e \quad \forall e \in E \\ & \xi_{u\bar{u}} \le \pi(u,\bar{u}) \quad \forall (u,\bar{u}) \in R \\ & \xi_{S,u\bar{u}} \ge 0 \quad \forall S \in \mathscr{S}, \forall (u,\bar{u}) \odot S \end{aligned}$$

Visualizing the Dual

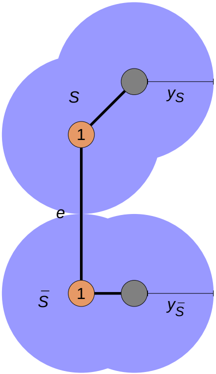


- dual y_S of Steiner cut S is visualized as moat around S of radius y_S
- edge e is tight if

$$\sum_{S:e\in\delta(S)} y_S = c_e$$

- growth of moat corresponds to an increase in the dual value

Visualizing the Dual

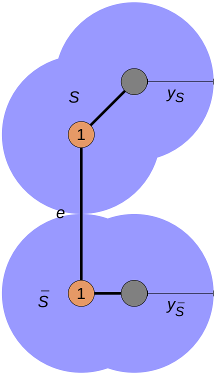


- dual y_S of Steiner cut S is visualized as moat around S of radius y_S
- edge e is tight if

$$\sum_{S: e \in \delta(S)} y_S = c_e$$

- growth of moat corresponds to an increase in the dual value

Visualizing the Dual



- dual y_S of Steiner cut S is visualized as moat around S of radius y_S
- edge e is tight if

$$\sum_{S: e \in \delta(S)} y_S = c_e$$

- growth of moat corresponds to an increase in the dual value

Activity Notion

Death time: let $d_G(u,\bar{u})$ be distance between $u,\bar{u}$ in G

$$d(u,\bar{u}) := \frac{1}{2} d_G(u,\bar{u})$$

Activity: terminal $u \in R$ is active at time τ iff

$$\xi^{\tau}_{u\bar{u}} < \pi(u,\bar{u}) \quad \text{and} \quad \tau \leq d(u,\bar{u}).$$

Call a moat active if it contains at least one active terminal

Activity Notion

Death time: let $d_G(u,\bar{u})$ be distance between $u,\bar{u}$ in G

$$\mathrm{d}(u,\bar{u}) := \frac{1}{2} d_G(u,\bar{u})$$

Activity: terminal $u \in R$ is active at time τ iff

$$\xi^{\tau}_{u\bar{u}} < \pi(u,\bar{u}) \quad \text{and} \quad \tau \le \mathrm{d}(u,\bar{u}).$$

Call a moat active if it contains at least one active terminal

Activity Notion

Death time: let $d_G(u,\bar{u})$ be distance between $u,\bar{u}$ in G

$$\mathrm{d}(u,\bar{u}) := \frac{1}{2} d_G(u,\bar{u})$$

Activity: terminal $u \in R$ is active at time τ iff

$$\xi^{\tau}_{u\bar{u}} < \pi(u,\bar{u}) \quad \text{and} \quad \tau \leq \mathrm{d}(u,\bar{u}).$$

Call a moat active if it contains at least one active terminal

Algorithm

- process over time
- at every time τ: grow all active moats uniformly
- share dual growth of a moat evenly among active terminals contained in it
- if two active moats collide: add all new tight edges on path between them to the forest F
- if a terminal pair $(u, \bar{u})$ becomes inactive since its cost share reaches its penalty, add $(u, \bar{u})$ to the set Q
- terminate if all moats are inactive

Algorithm

- process over time
- at every time τ: grow all active moats uniformly
- share dual growth of a moat evenly among active terminals contained in it
- if two active moats collide: add all new tight edges on path between them to the forest F
- if a terminal pair $(u, \bar{u})$ becomes inactive since its cost share reaches its penalty, add $(u, \bar{u})$ to the set Q
- terminate if all moats are inactive

Algorithm

- process over time
- at every time τ: grow all active moats uniformly
- share dual growth of a moat evenly among active terminals contained in it
- if two active moats collide: add all new tight edges on path between them to the forest F
- if a terminal pair $(u, \bar{u})$ becomes inactive since its cost share reaches its penalty, add $(u, \bar{u})$ to the set Q
- terminate if all moats are inactive

Algorithm

- process over time
- at every time τ: grow all active moats uniformly
- share dual growth of a moat evenly among active terminals contained in it
- if two active moats collide: add all new tight edges on path between them to the forest F
- if a terminal pair $(u, \bar{u})$ becomes inactive since its cost share reaches its penalty, add $(u, \bar{u})$ to the set Q
- terminate if all moats are inactive

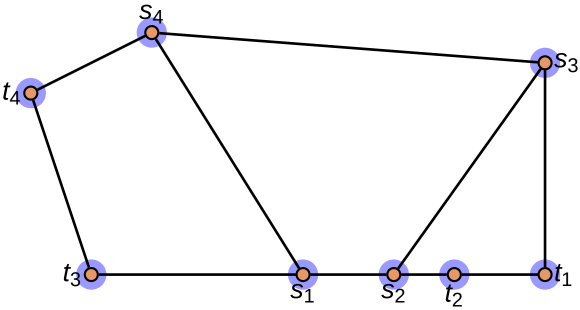
Algorithm

- process over time
- at every time τ: grow all active moats uniformly
- share dual growth of a moat evenly among active terminals contained in it
- if two active moats collide: add all new tight edges on path between them to the forest F
- if a terminal pair $(u, \bar{u})$ becomes inactive since its cost share reaches its penalty, add $(u, \bar{u})$ to the set Q
- terminate if all moats are inactive

Algorithm

- process over time
- at every time τ: grow all active moats uniformly
- share dual growth of a moat evenly among active terminals contained in it
- if two active moats collide: add all new tight edges on path between them to the forest F
- if a terminal pair $(u, \bar{u})$ becomes inactive since its cost share reaches its penalty, add $(u, \bar{u})$ to the set Q
- terminate if all moats are inactive

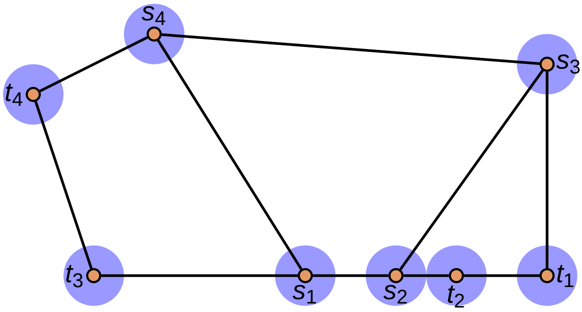
Illustration



$\tau = 0.5$

	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^τ	1	1	1	1

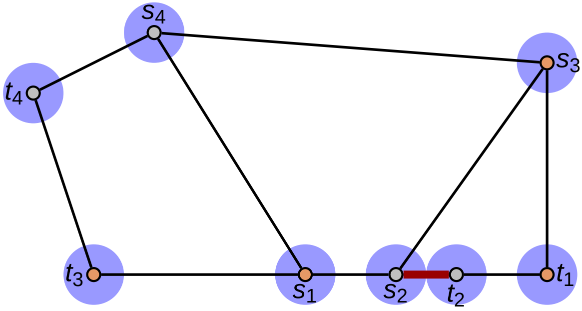
Illustration



$\tau = 1$

	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^τ	2	2	2	2

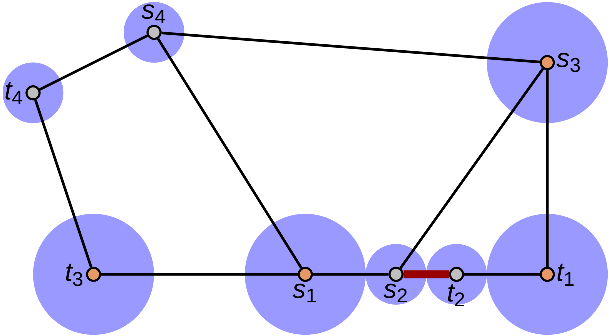
Illustration



$\tau = 1$

	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^τ	2	2	2	2

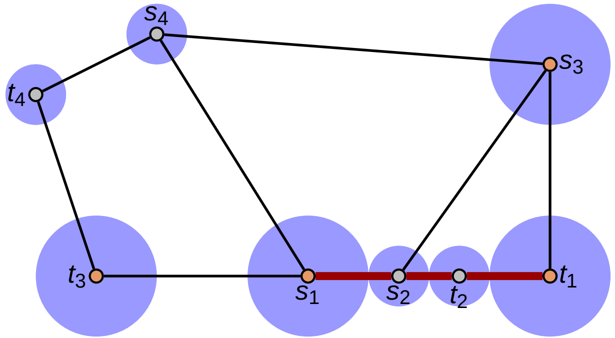
Illustration



$\tau = 2$

	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^τ	4	2	4	2

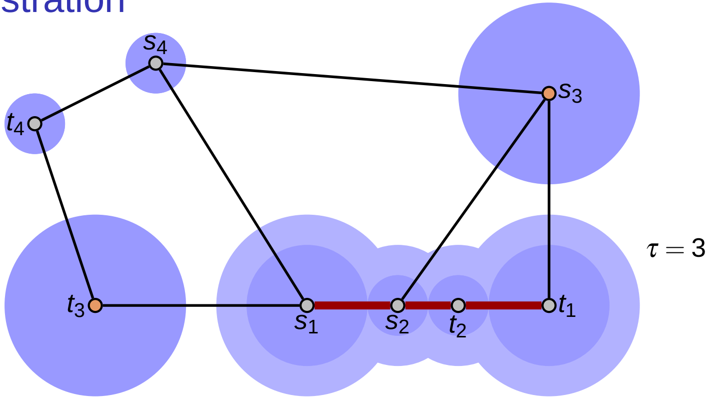
Illustration



$\tau = 2$

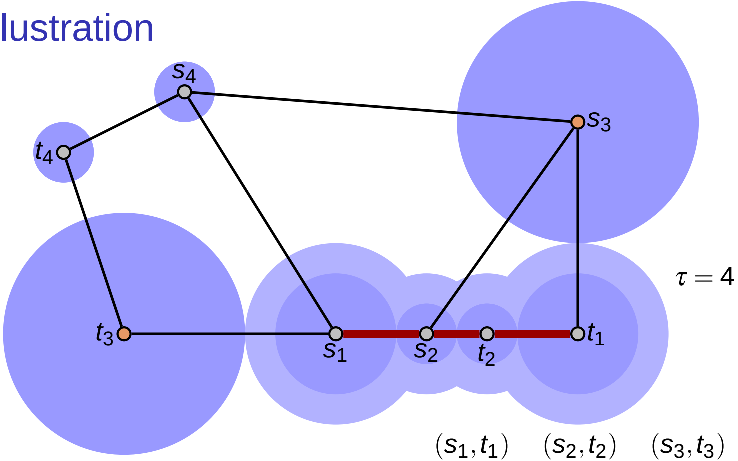
	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^{τ}	4	2	4	2

Illustration



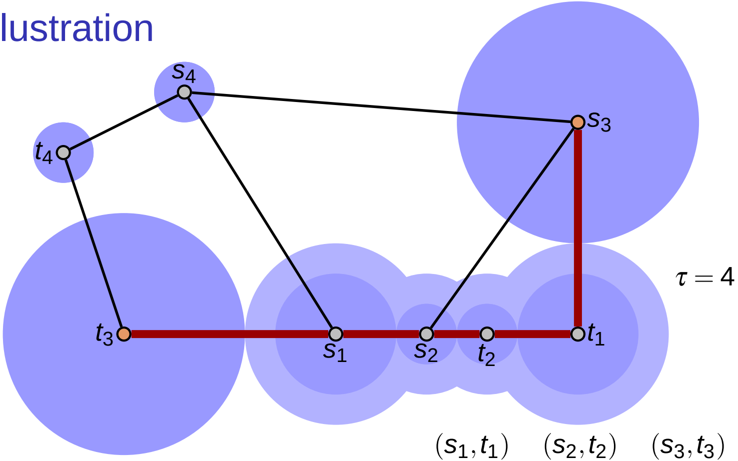
	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^τ	5	2	6	2

Illustration



	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^{τ}	5	1	8	1

Illustration



	(s_1, t_1)	(s_2, t_2)	(s_3, t_3)	(s_4, t_4)
$d(\cdot)$	4	1	22	3
$\pi(\cdot)$	5	5	∞	2
ξ^{τ}	5	1	8	1

Two Quick Proofs

Lem: ξ is cross-monotonic

Proof (idea): at every time τ and for any $S \subseteq S'$

- moat system wrt. S is a refinement of moat system wrt. S'
- cost share of u wrt. S is at least cost share of u wrt. S'

Lem: ξ is 3-budget balanced

Proof (idea):

- cost of solution is at most $2 \sum y_S$ for Steiner forest and $\sum \xi_{u\bar{u}}$ for total penalty
- need to prove that $\sum y_S \leq C(S)$ (hard part)

Two Quick Proofs

Lem: ξ is cross-monotonic

Proof (idea): at every time τ and for any $S \subseteq S'$

- moat system wrt. S is a refinement of moat system wrt. S'
- cost share of u wrt. S is at least cost share of u wrt. S'

Lem: ξ is 3-budget balanced

Proof (idea):

- cost of solution is at most $2\sum y_S$ for Steiner forest and $\sum \xi_{u\bar{u}}$ for total penalty
- need to prove that $\sum y_S \leq C(S)$ (hard part)

Two Quick Proofs

Lem: ξ is cross-monotonic

Proof (idea): at every time τ and for any $S \subseteq S'$

- moat system wrt. S is a refinement of moat system wrt. S'
- cost share of u wrt. S is at least cost share of u wrt. S'

Lem: ξ is 3-budget balanced

Proof (idea):

- cost of solution is at most $2 \sum y_S$ for Steiner forest and $\sum \xi_{u\bar{u}}$ for total penalty
- need to prove that $\sum y_S \leq C(S)$ (hard part)

Two Quick Proofs

Lem: ξ is cross-monotonic

Proof (idea): at every time τ and for any $S \subseteq S'$

- moat system wrt. S is a refinement of moat system wrt. S'
- cost share of u wrt. S is at least cost share of u wrt. S'

Lem: ξ is 3-budget balanced

Proof (idea):

- cost of solution is at most $2\sum y_S$ for Steiner forest and $\sum \xi_{u\bar{u}}$ for total penalty
- need to prove that $\sum y_S \leq C(S)$ (hard part)

Partitioning Lemma

Given: cross-monotonic cost sharing method ξ on U that is β-budget balanced for C

Lem: If there is a partition $U = U_1 \dot{\cup} U_2$ such that the Moulin mechanism $M(\xi)$ is α_i-approximate on U_i for all $i \in \{1,2\}$, then $M(\xi)$ is $(\alpha_1 + \alpha_2)\beta$-approximate on U

Partitioning Lemma

Given: cross-monotonic cost sharing method ξ on U that is β-budget balanced for C

Lem: If there is a partition $U = U_1 \,\dot{\cup}\, U_2$ such that the Moulin mechanism $M(\xi)$ is α_i-approximate on U_i for all $i \in \{1, 2\}$, then $M(\xi)$ is $(\alpha_1 + \alpha_2)\beta$-approximate on U

High-Utility Users

U_1 = set of all users i with $u_i \geq \pi_i$

Lem: (High-Utility Lemma): $M(\xi)$ is 1-approximate on U_1.

Proof: By construction, $\xi(i,S) \leq \pi_i \leq u_i$ for all i, for all $S \subseteq U_1$.
Thus, set S^M output by Moulin mechanism $M(\xi)$ is U.
Moreover, U minimizes social cost.

High-Utility Users

U_1 = set of all users i with $u_i \geq \pi_i$

Lem: (High-Utility Lemma): $M(\xi)$ is 1-approximate on U_1.

Proof: By construction, $\xi(i, S) \leq \pi_i \leq u_i$ for all i, for all $S \subseteq U_1$.
Thus, set S^M output by Moulin mechanism $M(\xi)$ is U.
Moreover, U minimizes social cost.

High-Utility Users

$U_1 =$ set of all users i with $u_i \geq \pi_i$

Lem: (High-Utility Lemma): $M(\xi)$ is 1-approximate on U_1.

Proof: By construction, $\xi(i, S) \leq \pi_i \leq u_i$ for all i, for all $S \subseteq U_1$.
Thus, set S^M output by Moulin mechanism $M(\xi)$ is U.
Moreover, U minimizes social cost.

High-Utility Users

$U_1 =$ set of all users i with $u_i \geq \pi_i$

Lem: (High-Utility Lemma): $M(\xi)$ is 1-approximate on U_1.

Proof: By construction, $\xi(i,S) \leq \pi_i \leq u_i$ for all i, for all $S \subseteq U_1$.
Thus, set S^M output by Moulin mechanism $M(\xi)$ is U.
Moreover, U minimizes social cost.

High-Utility Users

$U_1 =$ set of all users i with $u_i \geq \pi_i$

Lem: (High-Utility Lemma): $M(\xi)$ is 1-approximate on U_1.

Proof: By construction, $\xi(i,S) \leq \pi_i \leq u_i$ for all i, for all $S \subseteq U_1$.
Thus, set S^M output by Moulin mechanism $M(\xi)$ is U.
Moreover, U minimizes social cost.

Low-Utility Users

U_2 = set of all users i with $u_i < \pi_i$

ξ' = cross-monotonic cost sharing method for Steiner forest problem

Similarity Property: For every $S \subseteq U_2$: If there is a user $i \in S$ with $\xi(i,S) > u_i$ or $\xi'(i,S) > u_i$ then there exists a user $j \in S$ with $\xi(j,S) > u_j$ and $\xi'(j,S) > u_j$.

Lem: When starting with a low-utility set $S \subseteq U_2$, the final user sets produced by $M(\xi)$ and $M(\xi')$ are the same

Low-Utility Users

$U_2 =$ set of all users i with $u_i < \pi_i$

$\xi' =$ cross-monotonic cost sharing method for Steiner forest problem

Similarity Property: For every $S \subseteq U_2$: If there is a user $i \in S$ with $\xi(i,S) > u_i$ or $\xi'(i,S) > u_i$ then there exists a user $j \in S$ with $\xi(j,S) > u_j$ and $\xi'(j,S) > u_j$.

Lem: When starting with a low-utility set $S \subseteq U_2$, the final user sets produced by $M(\xi)$ and $M(\xi')$ are the same

Low-Utility Users

$U_2 =$ set of all users i with $u_i < \pi_i$

$\xi' =$ cross-monotonic cost sharing method for Steiner forest problem

Similarity Property: For every $S \subseteq U_2$: If there is a user $i \in S$ with $\xi(i,S) > u_i$ or $\xi'(i,S) > u_i$ then there exists a user $j \in S$ with $\xi(j,S) > u_j$ and $\xi'(j,S) > u_j$.

Lem: When starting with a low-utility set $S \subseteq U_2$, the final user sets produced by $M(\xi)$ and $M(\xi')$ are the same

Low-Utility Users

U_2 = set of all users i with $u_i < \pi_i$

ξ' = cross-monotonic cost sharing method for Steiner forest problem

Similarity Property: For every $S \subseteq U_2$: If there is a user $i \in S$ with $\xi(i,S) > u_i$ or $\xi'(i,S) > u_i$ then there exists a user $j \in S$ with $\xi(j,S) > u_j$ and $\xi'(j,S) > u_j$.

Lem: When starting with a low-utility set $S \subseteq U_2$, the final user sets produced by $M(\xi)$ and $M(\xi')$ are the same

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^* = \alpha \Pi^*$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \le C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \le u(U \setminus S) + C'(S) = \Pi'(S) \le \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Low-Utility Users

Lem: (Low-Utility Lemma): $M(\xi)$ is α-approximate on U_2 if $M(\xi')$ is α-approximate on U_2

Proof: Solution for set with minimum social cost never pays a penalty, as $u_i < \pi_i$. Thus, optimal social cost for PCSF and SF are the same. Furthermore, $C(S) \leq C'(S)$ for all $S \subseteq U_2$. Due to the similarity property, both mechanisms output the same set S.

$$\Pi(S) = u(U \setminus S) + C(S) \leq u(U \setminus S) + C'(S) = \Pi'(S) \leq \alpha \Pi'^{*} = \alpha \Pi^{*}$$

Putting the Pieces together...

We showed:

- $M(\xi)$ is 1-approximate on high-utility users
- $M(\xi)$ is $\Theta(\log^2 n)$-approximate on low-utility users

Thm: $M(\xi)$ is a group-strategyproof cost sharing mechanism for PCSF that is 3-budget balanced and $\Theta(\log^2 n)$-approximate

Remark: technique extends to other prize-collecting problems, e.g., prize-collecting facility location

Putting the Pieces together...

We showed:

- $M(\xi)$ is 1-approximate on high-utility users
- $M(\xi)$ is $\Theta(\log^2 n)$-approximate on low-utility users

Thm: $M(\xi)$ is a group-strategyproof cost sharing mechanism for PCSF that is 3-budget balanced and $\Theta(\log^2 n)$-approximate

Remark: technique extends to other prize-collecting problems, e.g., prize-collecting facility location

Putting the Pieces together...

We showed:

- $M(\xi)$ is 1-approximate on high-utility users
- $M(\xi)$ is $\Theta(\log^2 n)$-approximate on low-utility users

Thm: $M(\xi)$ is a group-strategyproof cost sharing mechanism for PCSF that is 3-budget balanced and $\Theta(\log^2 n)$-approximate

Remark: technique extends to other prize-collecting problems, e.g., prize-collecting facility location

Part III

Conclusions and Open Problems

Conclusions

New efficiency measure:

- circumvents classical intractability results
- enables to differentiate the solution quality of different cost sharing mechanisms
- motivates the design of “good” cost sharing mechanisms
- ... but still might be too restrictive!?

Obs: Suppose that there is a set $S \subseteq U$ with $C(S') \geq C(S)/\delta$ for all $S' \subseteq S$ and some constant $\delta \geq 1$. Then there is no $\Omega(\log |S|)$-approximate Moulin mechanism that satisfies cost recovery.

[Brenner, S. 06]

Conclusions

New efficiency measure:

- circumvents classical intractability results
- enables to differentiate the solution quality of different cost sharing mechanisms
- motivates the design of “good” cost sharing mechanisms
- ... but still might be too restrictive!?

Obs: Suppose that there is a set $S \subseteq U$ with $C(S') \geq C(S)/\delta$ for all $S' \subseteq S$ and some constant $\delta \geq 1$. Then there is no $\Omega(\log |S|)$-approximate Moulin mechanism that satisfies cost recovery.

[Brenner, S. 06]

Conclusions

New efficiency measure:

- circumvents classical intractability results
- enables to differentiate the solution quality of different cost sharing mechanisms
- motivates the design of “good” cost sharing mechanisms
- ... but still might be too restrictive!?

Obs: Suppose that there is a set $S \subseteq U$ with $C(S') \geq C(S)/\delta$ for all $S' \subseteq S$ and some constant $\delta \geq 1$. Then there is no $\Omega(\log |S|)$-approximate Moulin mechanism that satisfies cost recovery.

[Brenner, S. 06]

Conclusions

New efficiency measure:

- circumvents classical intractability results
- enables to differentiate the solution quality of different cost sharing mechanisms
- motivates the design of “good” cost sharing mechanisms
- ... but still might be too restrictive!?

Obs: Suppose that there is a set $S \subseteq U$ with $C(S') \geq C(S)/\delta$ for all $S' \subseteq S$ and some constant $\delta \geq 1$. Then there is no $\Omega(\log |S|)$-approximate Moulin mechanism that satisfies cost recovery.

[Brenner, S. 06]

Open Problems

- LP formulation for PCSF primal-dual algorithm
- study other problems in cost sharing context (appealing from both sides, game theory and algorithm design)
- come up with alternative reasonable objectives (group-strategyproofness sometimes asks for too much)