# Approximation of Cost-Sharing and Utilitarian Mechanisms

Stefano Leonardi

Sapienza University of Rome

MITACS Workshop on Internet and Network Economics

SFU Harbour Centre, Vancouver

May 30 - June 1, 2011

# Algorithm and Mechanism Design



Mechanism design offers a conceptual framework for algorithm design and optimization in the age of Internet:

- Input data owned from selfish distributed agents
- Agents can strategize in order to maximize their individual utility
- Algorithms should both provide efficient and correct solutions and incentivize agents (with payments) to reveal true input data
- Our ideal goal is to implement a mechanism in the form of a dominant strategy:

  Reveal true input data maximize individual utility, whatever strategy is played from the other players

# Approximation and Mechanism Design


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


Two main sources of complexity:

- The network problems we like to solve are often computationally hard:

  Develop a theory of approximation algorithms that yield good strategic properties.

- Imposing good strategic properties limit the quality of approximation that can be obtained,

  independently from computational complexity.

# Approximation and Mechanism Design


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


Two main classes of problems:

1. Cost sharing mechanisms: fair share of the cost of providing a service to the agents.
2. Utilitarian mechanisms:
   - minimize the cost of the solution that uses resources provided by the agents; or
   - maximize the utility of the agents that are served from the mechanism.

# Approximation and Mechanism Design


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


Twist methodologies for the design of approximation algorithms to yield good strategic properties

We give examples of application of

- Primal-dual algorithms
- Polynomial time approximation schemes
- Pareto-optimal solutions and Multi-objective optimization
- Lagrangean relaxation

to relevant combinatorial optimization problems in networks.

# Overview of the tutorial


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


## Plan for the two days:

1. Tutorial Part I (today): Approximation of Cost-Sharing Mechanisms
2. Tutorial Part II (tomorrow): Approximation of Utilitarian Mechanisms

# Talk Outline


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


| | |
|---|---|
| Part I | Introduction to cost-sharing mechanisms |
| Part II | Moulin-Shenker mechanisms |
| Part III | The Facility location problem |
| Part IV | The Steiner forest problem |
| Part V | Lower bounds for cross-monotonic cost-sharing methods |
| Part VI | Summary and conclusions |

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


# Cost-Sharing Mechanisms

# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


The ingredients:

- A service provider.
- A set $U$ of potential users (agents, customers).
- Each user $j \in U$ has a (private) utility $u_j$ (the price $j$ is willing to pay to receive the service).
- A cost-function $c$: $c(Q)$ is the cost for servicing a set $Q \subseteq U$. $c(Q)$ is usually given by the solution to an optimization problem.

# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


## The ingredients:

- A service provider.
- A set $U$ of potential users (agents, customers).
- Each user $j \in U$ has a (private) utility $u_j$ (the price $j$ is willing to pay to receive the service).
- A cost-function $c$: $c(Q)$ is the cost for servicing a set $Q \subseteq U$. $c(Q)$ is usually given by the solution to an optimization problem.

## Cost-Sharing Mechanism:

- Receive bids $b_j$ from all users $j \in U$.
- Select recipients $Q \subseteq U$ using bids.
- Distribute service cost $c(Q)$ among users in $Q$: Determine payment $p_j$ for each $j \in Q$.

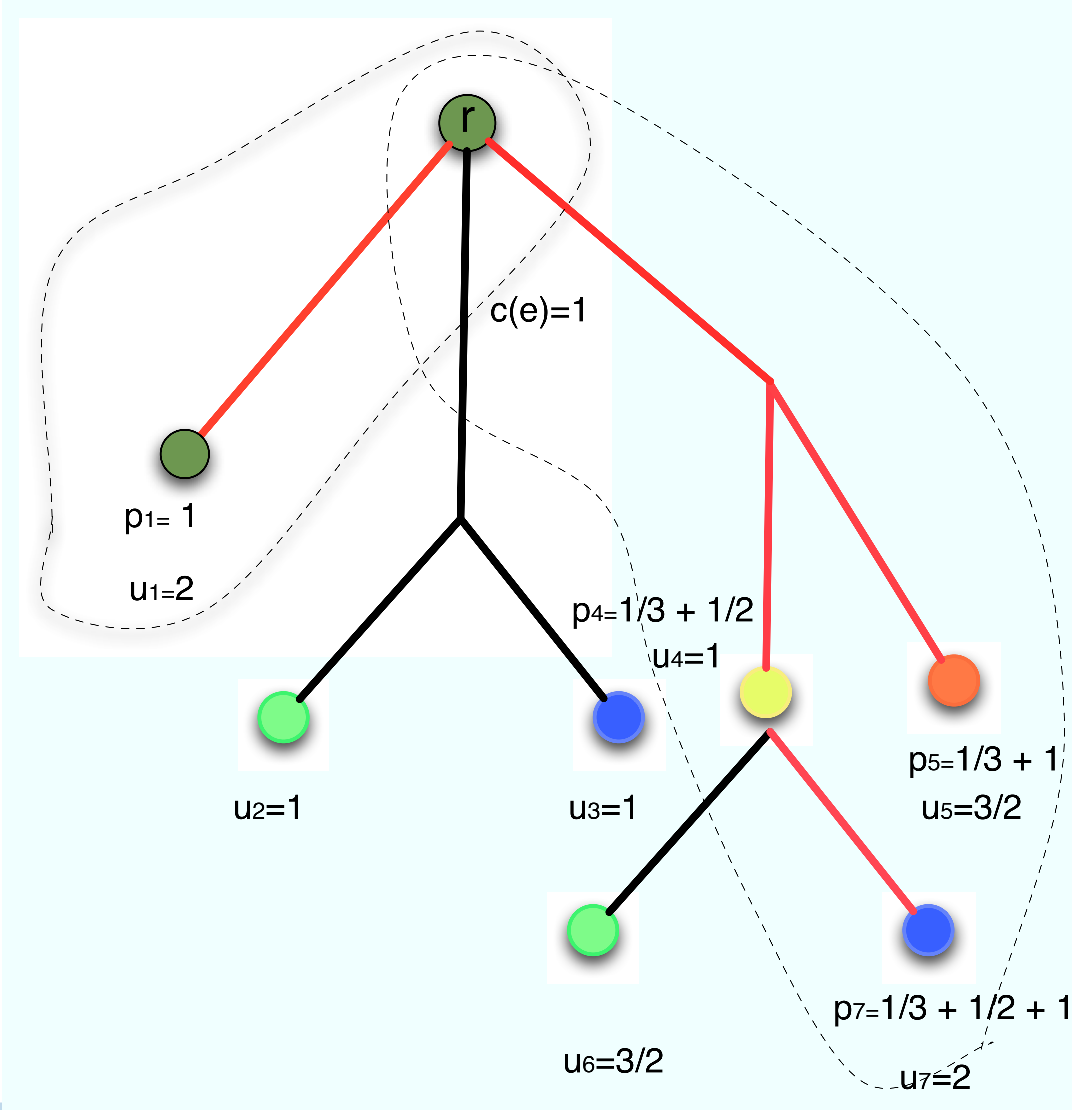
# Example: Multicast Transmission


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


*Shapley cost shares*

- Select a subset $Q$ and a tree $T$ spanning $Q$
- Share the cost of every edge of $T$ evenly between the players served by the edge
- All players in $Q$ should bid more than the individual cost-share



# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


- **Benefit** of user $j$ is $u_j - p_j$ if $j \in Q$, and $0$ otherwise.

# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


- Benefit of user $j$ is $u_j - p_j$ if $j \in Q$, and $0$ otherwise.
- Users may lie about their utilities to increase benefit.

# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


- Benefit of user $j$ is $u_j - p_j$ if $j \in Q$, and $0$ otherwise.
- Users may lie about their utilities to increase benefit.

Objectives:

- **Strategyproofness:** Dominant strategy for each user is to bid true utility.
- **Group-Strategyproofness:** Same holds even if users collaborate. No side payments between users.
- **Cost Recovery or Budget Balance:** $\sum_{j \in Q} p_j \geq c(Q)$.
- **Competitiveness:** $\sum_{j \in Q} p_j \leq \mathtt{opt}_Q$.

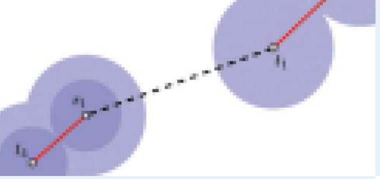
# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


- Finding such cost-shares and a cost-function is hard if underlying problem is hard.



# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


- Finding such cost-shares and a cost-function is hard if underlying problem is hard.
- Finding such cost-shares may be impossible if we want to ensure strategyproofness (later in this talk)

# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


- Finding such cost-shares and a cost-function is hard if underlying problem is hard.
- Finding such cost-shares may be impossible if we want to ensure strategyproofness (later in this talk)
- Relax budget balance condition:
  $\beta$**-budget balance**: $\frac{1}{\beta}c(Q) \leq \sum_{j\in Q} p_j \leq \texttt{opt}_Q, \quad \beta \geq 1$

# Design of Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


The Primal-dual method is a natural choice.

- Many combinatorial optimization problems can be formulated as Integer Linear Programs (ILP)
- Primal-dual algorithms construct a feasible solution to the ILP together with a dual solution to the fractional LP
- The cost of the feasible solution if $\beta$-approximated if its ratio to the value of the dual solution is at most $\beta$
- Dual variables have a natural interpretation as costs to be distributed between players
- Weak duality implies competitiveness
- Approximation ratio $\beta$ implies $\beta$-budget balance.

# Design of Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


The Primal-dual method is a natural choice.

- Many combinatorial optimization problems can be formulated as Integer Linear Programs (ILP)
- Primal-dual algorithms construct a feasible solution to the ILP together with a dual solution to the fractional LP
- The cost of the feasible solution if $\beta$-approximated if its ratio to the value of the dual solution is at most $\beta$
- Dual variables have a natural interpretation as costs to be distributed between players
- Weak duality implies competitiveness
- Approximation ratio $\beta$ implies $\beta$-budget balance.

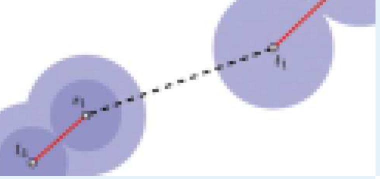
# Design of Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


The Primal-dual method is a natural choice.

- Many combinatorial optimization problems can be formulated as Integer Linear Programs (ILP)
- Primal-dual algorithms construct a feasible solution to the ILP together with a dual solution to the fractional LP
- The cost of the feasible solution if $\beta$-approximated if its ratio to the value of the dual solution is at most $\beta$
- Dual variables have a natural interpretation as costs to be distributed between players
- Weak duality implies competitiveness
- Approximation ratio $\beta$ implies $\beta$-budget balance.



# Design of Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


The Primal-dual method is a natural choice.

- Many combinatorial optimization problems can be formulated as Integer Linear Programs (ILP)
- Primal-dual algorithms construct a feasible solution to the ILP together with a dual solution to the fractional LP
- The cost of the feasible solution if $\beta$-approximated if its ratio to the value of the dual solution is at most $\beta$
- Dual variables have a natural interpretation as costs to be distributed between players
- Weak duality implies competitiveness
- Approximation ratio $\beta$ implies $\beta$-budget balance.

# Design of Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


The Primal-dual method is a natural choice.

- Many combinatorial optimization problems can be formulated as Integer Linear Programs (ILP)
- Primal-dual algorithms construct a feasible solution to the ILP together with a dual solution to the fractional LP
- The cost of the feasible solution if $\beta$-approximated if its ratio to the value of the dual solution is at most $\beta$
- Dual variables have a natural interpretation as costs to be distributed between players
- Weak duality implies competitiveness
- Approximation ratio $\beta$ implies $\beta$-budget balance.

# Metric Facility location


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Input:

- undirected graph $G = (V, E)$
- non-negative edge costs $c : E \rightarrow \mathbb{R}^+$
- set of facilities $F \subseteq V$
- facility $i$ has facility opening cost $f_i$
- set of demand points $D \subseteq V$
- $c_{ij}$: cost of connecting demand point $j$ to facility $i$

Goal: Compute

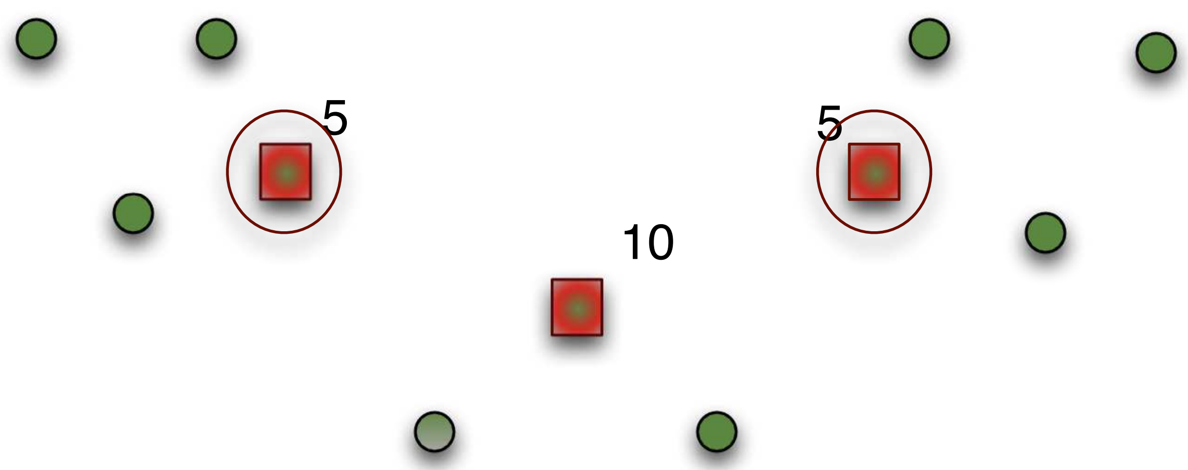
- set $F' \subseteq F$ of opened facilities; and
- function $\phi : \mathcal{D} \rightarrow \mathcal{F}'$ that minimizes

$$\sum_{i \in F'} f_i + \sum_{j \in \mathcal{D}} c_{\phi(j)j}$$

# Example


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Distances are 1 to the nearest facility and 3 to the further facility.

The two facilities of cost 5 are opened

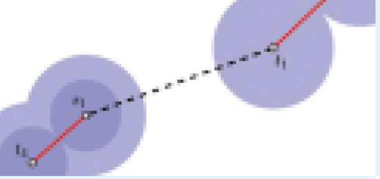
# LP formulation


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


$$
\begin{array}{llcll}
\text{min} & \sum_{i\in F, j\in D} c_{ij}x_{ij} + \sum_{i\in F} f_i y_i & & & \\
\text{s.t.} & \sum_{i\in F} x_{ij} & \geq & 1 & j \in D \\
& y_i - x_{ij} & \geq & 0 & i \in F, j \in D \\
& x_{ij} & \in & \{0,1\} & i \in F, j \in D \\
& y_i & \in & \{0,1\} & i \in F
\end{array}
$$

- $y_i = 1$ if facility $i$ is opened;
- $x_{ij} = 1$ if demand $j$ connected to facility $i$.



# LP relaxation:


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission

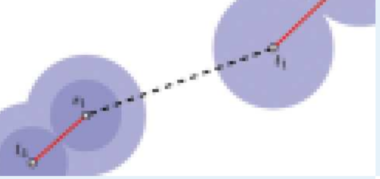

$$
\begin{array}{lrcll}
\textbf{min} & \displaystyle\sum_{i\in F, j\in D} c_{ij}x_{ij} + \sum_{i\in\mathcal{F}} f_i y_i & & & \\
\text{s.t.} & \displaystyle\sum_{i\in F} x_{ij} & \geq & 1 & j \in D \\
 & y_i - x_{ij} & \geq & 0 & i \in F, j \in D \\
 & x_{ij} & \geq & 0 & i \in F, j \in D \\
 & y_i & \geq & 0 & i \in F
\end{array}
$$

# LP DUAL:


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


$$
\begin{array}{rrcll}
\mathrm{DualProgram}: \max & \displaystyle\sum_{j \in D} \alpha_j & & & \\
\text{s.t.} & \alpha_j - \beta_{ij} & \leq & c_{ij} & i \in F, j \in D \\
 & \displaystyle\sum_{j \in D} \beta_{ij} & \leq & f_i & i \in F \\
 & \alpha_j & \geq & 0 & j \in D \\
 & \beta_{ij} & \geq & 0 & i \in F, j \in D
\end{array}
$$



# Primal-dual Algorithm for Facility Location


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


At time 0, set all $\alpha_j = 0$, $\beta_{ij} = 0$ and declare all demands unconnected.

While there is an unconnected demand:

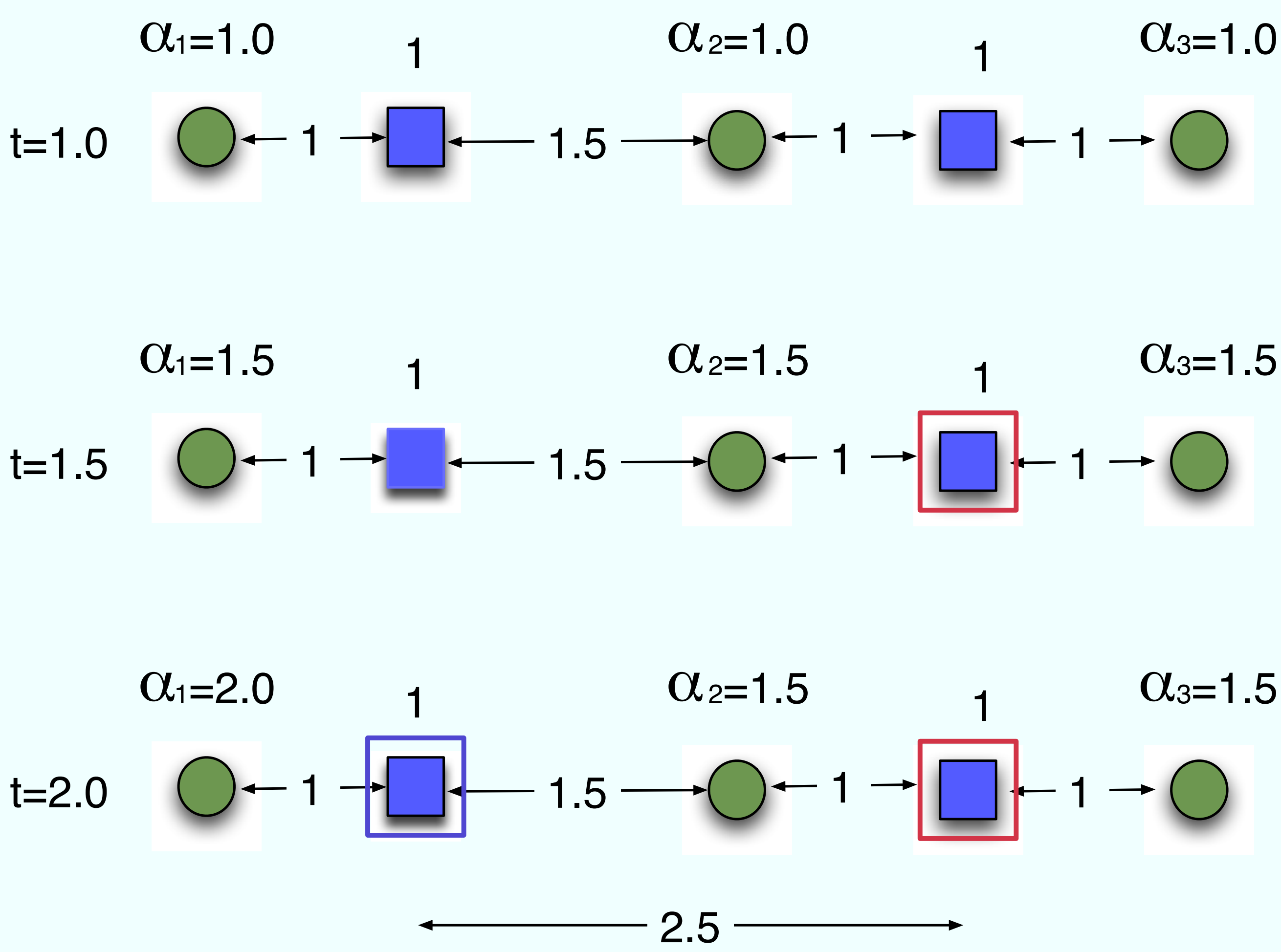
- Raise uniformly all $\alpha_j$'s of unconnected demands
- If $\alpha_j = c_{ij}$, declare demand $j$ tight with facility $i$
- For a tight constraint $ij$, raise both $\alpha_j$ and $\beta_{ij}$
- If $\sum_j \beta_{ij} = f_i$ at time $t_i$, declare:
  - Facility $i$ *temporarily opened* at time $t_i$;
  - Facility $i$ *permanently opened* if there is no permanently opened facility within distance $2t_i$;
  - All unconnected demands $j$ that are tight with $i$ *connected*;

# Example of execution of the algorithm


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission

# Proof of 3 approximation.


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


## Demands connected to opened facilities

- $\alpha_j = c_{ij} + \beta_{ij}$ for demands connected to opened facility $i$.
- $\alpha_j$ pays for connection cost $c_{ij}$ and contribute with $\beta_{ij}$ to $f_i$.
- Since other opened facilities are at distance $> t_i$, $\alpha_j$ does not pay for opening any other facility.

## Demands connected to temporarily opened facilities

- Demand $j$ connected to temporarily opened facility $i$. There exists an opened facility $i'$ with $c_{ii'} \leq 2t_i$.
- Since $c_{ji} \leq \alpha_j$ and $t_i \leq \alpha_j$, $c_{ji'} \leq c_{ji} + c_{ii'} \leq 3\alpha_j$

# A Strategyproof Mechanism


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Agent $j \in D$ has utility $u_j$ and reports bid $b_j$ to the mechanism:

- If $\alpha_j > b_j$ for unconnected city $j$ then discard agent $j$.
- If facility $i$ is opened at time $t_i$: any unconnected city $j$ tight with facility $i$ is connected and it is charged payment $p_j = \alpha_j = t_i$.
- If some unconnected city $j$'s becomes tight at time $\alpha_j$ with opened facility $i$ then connect city $j$ to facility $i$ and charge $p_j = \alpha_j$

[Devanur, Mihail, Vazirani, 2003]

# A Strategyproof Mechanism


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Truthfulness follows from bid independence:

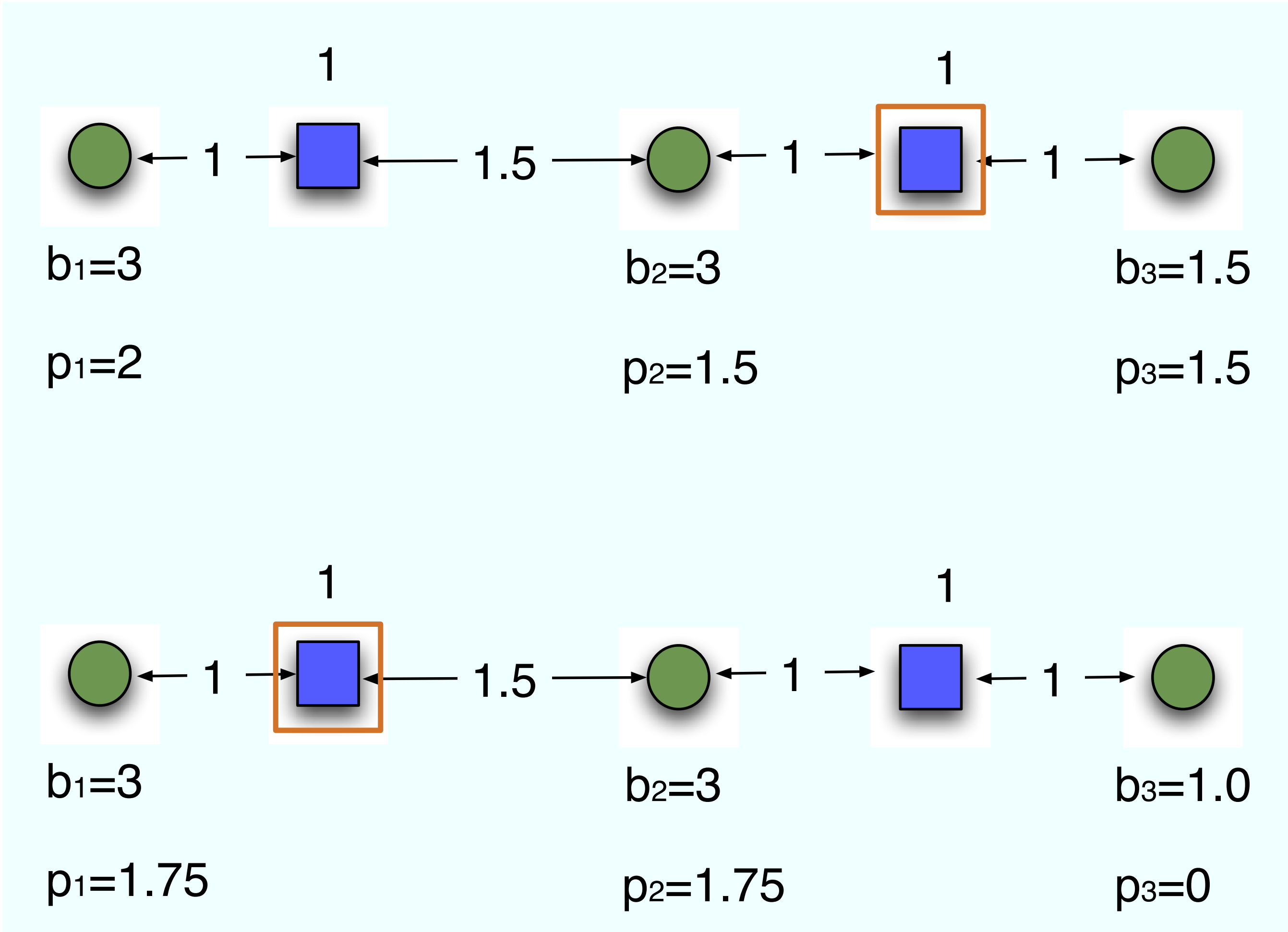
- Lowering the bid might result in early discard: payoff=0
- Raising the bid might result in paying more than the bid: payoff<0

Primal dual algorithms that monotonically increase dual variables often result in truthful cost-sharing mechanism.

# The Mechanism is not Group-strategyproof



Players can collude in order to manipulate the mechanism:



# Design of Group-strategyproof Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


The Primal-dual algorithm needs to be adapted:

- The only way to manipulate the game is early discard of some of the members of the coalition
- This is of interest only for players with 0 payoff!
- This is not beneficial if whenever a player leaves the game the cost share of all other players is not decreased
- We do not allow side payments, i.e., transfer utility between members of the coalition

# Cross-Monotonicity

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission

Formal requirement of group-strategyproof mechanisms:

Cost-Sharing Method:

- Given: Set $Q \subseteq U$ of users.
- Compute: Cost-shares $\xi_Q(j)$ for each $j \in Q$ such that competitiveness and $\beta$-budget balance hold.

# Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Formal requirement of group-strategyproof mechanisms:

Cost-Sharing Method:

- Given: Set $Q \subseteq U$ of users.
- Compute: Cost-shares $\xi_Q(j)$ for each $j \in Q$ such that competitiveness and $\beta$-budget balance hold.

$\xi$ is cross-monotonic if each individual cost-share does not increase as additional players join the game:

# Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Formal requirement of group-strategyproof mechanisms:

Cost-Sharing Method:

- Given: Set $Q \subseteq U$ of users.
- Compute: Cost-shares $\xi_Q(j)$ for each $j \in Q$ such that competitiveness and $\beta$-budget balance hold.

$\xi$ is cross-monotonic if each individual cost-share does not increase as additional players join the game:

$$\forall Q' \subseteq Q,\ \forall j \in Q' : \quad \xi_{Q'}(j) \geq \xi_Q(j).$$

# Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Formal requirement of group-strategyproof mechanisms:

Cost-Sharing Method:

- Given: Set $Q \subseteq U$ of users.
- Compute: Cost-shares $\xi_Q(j)$ for each $j \in Q$ such that competitiveness and $\beta$-budget balance hold.

$\xi$ is cross-monotonic if each individual cost-share does not increase as additional players join the game:

$$\forall Q' \subseteq Q,\ \forall j \in Q' : \quad \xi_{Q'}(j) \geq \xi_Q(j).$$

**Theorem [Moulin, Shenker '97]:** The Moulin–Shenker Mechanism is group-strategyproof, and satisfies cost recovery and competitiveness.

# Moulin–Shenker Mechanism


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Moulin–Shenker mechanism: Use cross-monotonic cost-sharing method to obtain group-strategyproof mechanisms.

# Moulin–Shenker Mechanism


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Moulin–Shenker mechanism: Use cross-monotonic cost-sharing method to obtain group-strategyproof mechanisms.

<u>Moulin–Shenker Mechanism:</u>

1. Initialize: $Q \leftarrow U$.
2. If for each user $j \in Q$: $\xi_Q(j) \leq b_j$ then stop.
3. Otherwise, remove from $Q$ all users with $\xi_Q(j) > b_j$ and repeat.

# Moulin–Shenker Mechanism


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


Designing a cost-sharing mechanism that is group-strategyproof, satisfies competitiveness and (approximate) budget balance.

$\Downarrow$ reduces to

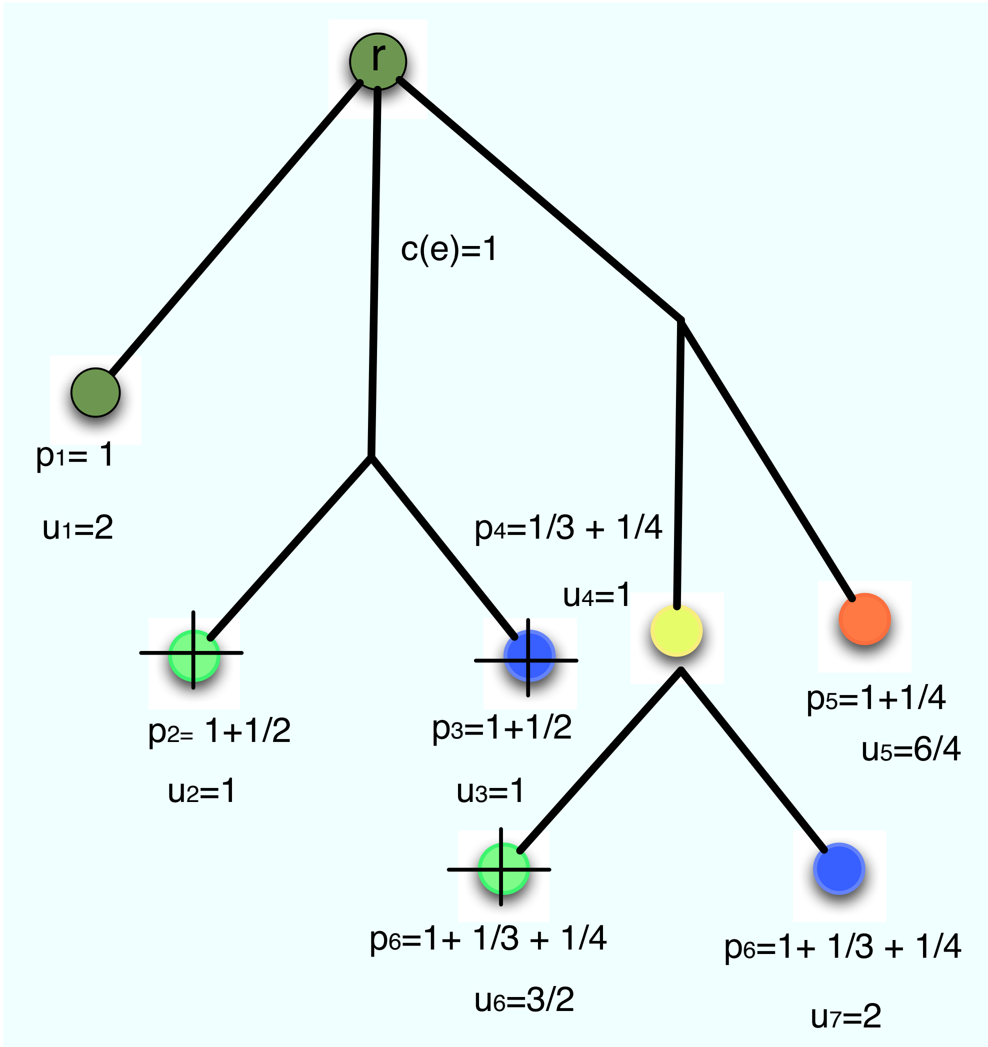
Designing a cross-monotonic cost-sharing method $\xi$ that satisfies competitiveness and (approximate) budget balance.

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission

*Moulin Mechanism for Shapley Cost Shares*

- Shapley is a cross-monotonic cost sharing method for Multicast transmission - Submodular function optimization
- Shapley is budget-balance, i.e. recovers the whole cost



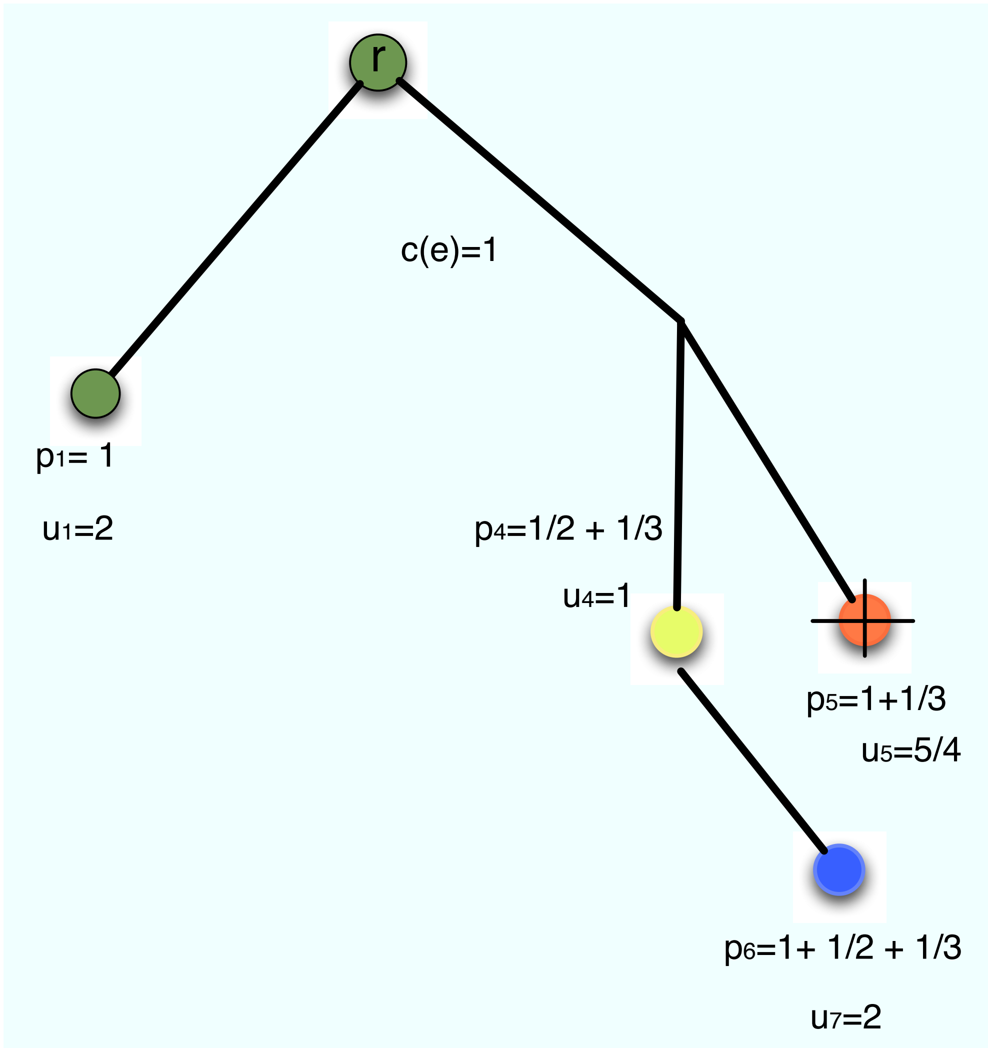
# Example: Multicast Transmission

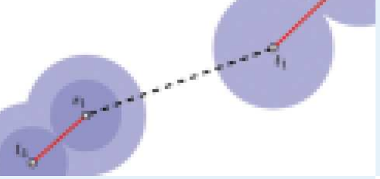

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


*Moulin Mechanism for Shapley Cost Shares*

- Shapley is a cross-monotonic cost sharing method for Multicast transmission - Submodular function optimization
- Shapley id budget-balance, i.e. recovers the whole cost





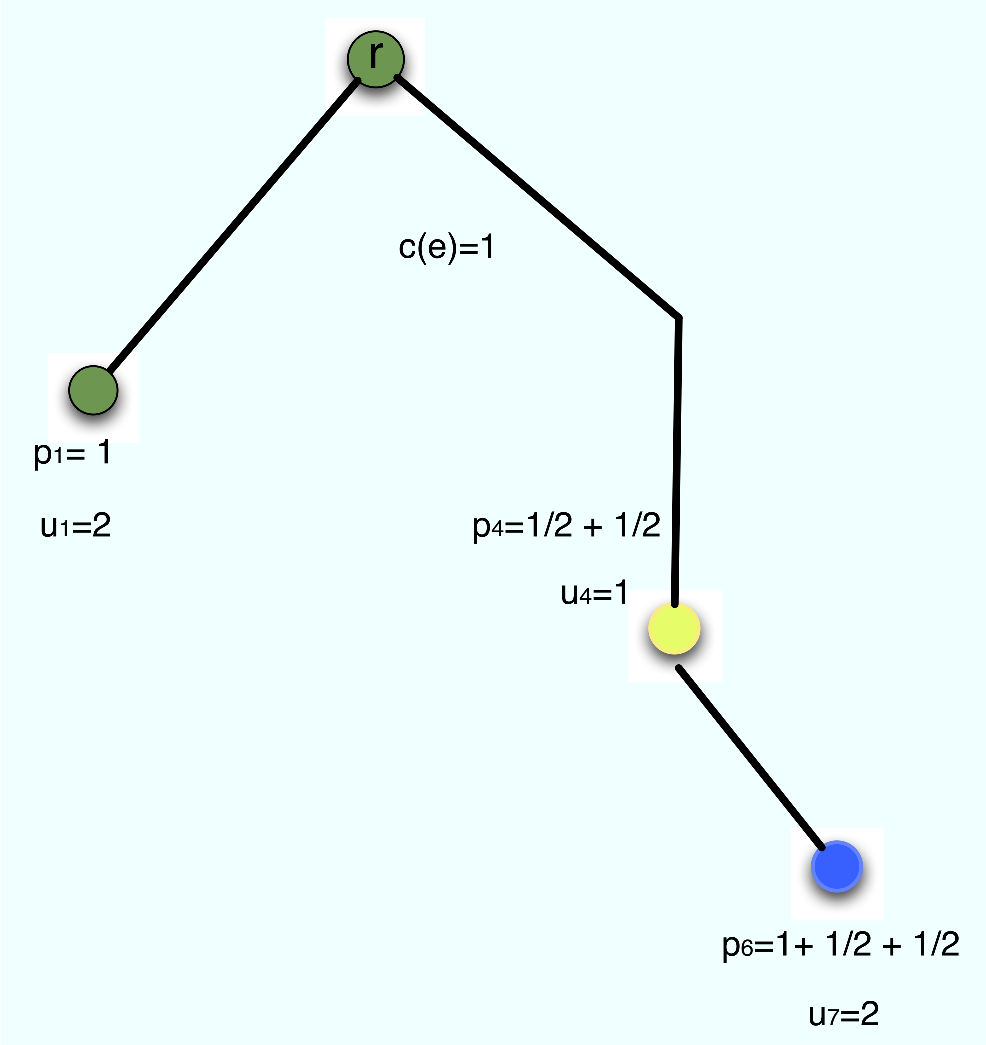
# Example: Multicast Transmission


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


*Moulin Mechanism for Shapley Cost Shares*

- Shapley is a cross-monotonic cost sharing method for Multicast transmission - Submodular function optimization
- Shapley id budget-balance, i.e. recovers the whole cost



# Known Results - Upper Bounds


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms
- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


| **Authors** | **Problem** | $\beta$ |
|---|---|---|
| [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, Schäfer '03], [Gupta et al. '03] | SRoB | $4$ |
| [Leonardi, Schäfer '03] | CFL | $30$ |
| [Könemann, Leonardi, Schäfer '05] | Steiner forest | $2$ |
| [Gupta, Könemann, Leonardi, Ravi, Schäfer '07] | Prize Collecting Steiner Forest | $3$ |
| [Goyal, Gupta, Leonardi, Ravi '07] | 2-Stage Stochastic Steiner Tree | $O(1)$ |

# Known Results - Lower Bounds


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

- Cost-Sharing Mechanisms
- Example: Multicast Transmission
- Metric Facility location
- LP formulation
- Primal-dual Algorithm for Facility Location
- Example of execution of the algorithm
- Proof of 3 approximation.
- A Strategyproof Mechanism
- A Strategyproof Mechanism
- The Mechanism is not Group-strategyproof
- Design of Group-strategyproof Mechanisms
- Cross-Monotonicity
- Moulin–Shenker Mechanism
- Example: Multicast Transmission


| **Authors** | **Problem** | $\beta$ |
|---|---|---|
| | **Lower bounds** | |
| [Immorlica, Mahdian, Mirrokni '05] | edge cover | $2$ |
| | facility location | $3$ |
| | vertex cover | $n^{1/3}$ |
| | set cover | $n$ |
| [Könemann, Leonardi, Schäfer, van Zwam '05] | Steiner tree | $2$ |

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

- The Mechanism
- Cost-shares
- Example of execution of the algorithm

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


# Group-strategyproof for Facility location

# The Mechanism


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

- The Mechanism
- Cost-shares
- Example of execution of the algorithm

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


Demands continue to contribute towards opening facilities even after connection:

- Raise dual variables $\alpha_j$ even after demand $j$ is connected
- The cost share of user $j$ is still the earliest time of connection of user $j$
- How can we limit the number and the cost of opened facilities?
- We still like to recover at least a costant fraction of the opening cost?

[Pal and Tardos, 2003]

# Cost-shares


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location
- The Mechanism
- Cost-shares
- Example of execution of the algorithm

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds


- $S_i$ :users contributing to making facility $i$ full, all within distance $t_i$ from $i$
- Raise cost share $\alpha_j$ even after $j$ becomes tight with an opened facility:

$$\xi_j = \min\{\min_{i:j\in S_i} t_i, \min_{i:j\notin S_i} c_{ij}\}$$

- Cost shares are cross-monotonic since by adding more users, every facility becomes full earlier
- Do not open a facility at time $t_i$ if one at distance $\leq 2t_i$ already exists.

The mechanism is still 3-budget balanced!

# Example of execution of the algorithm


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

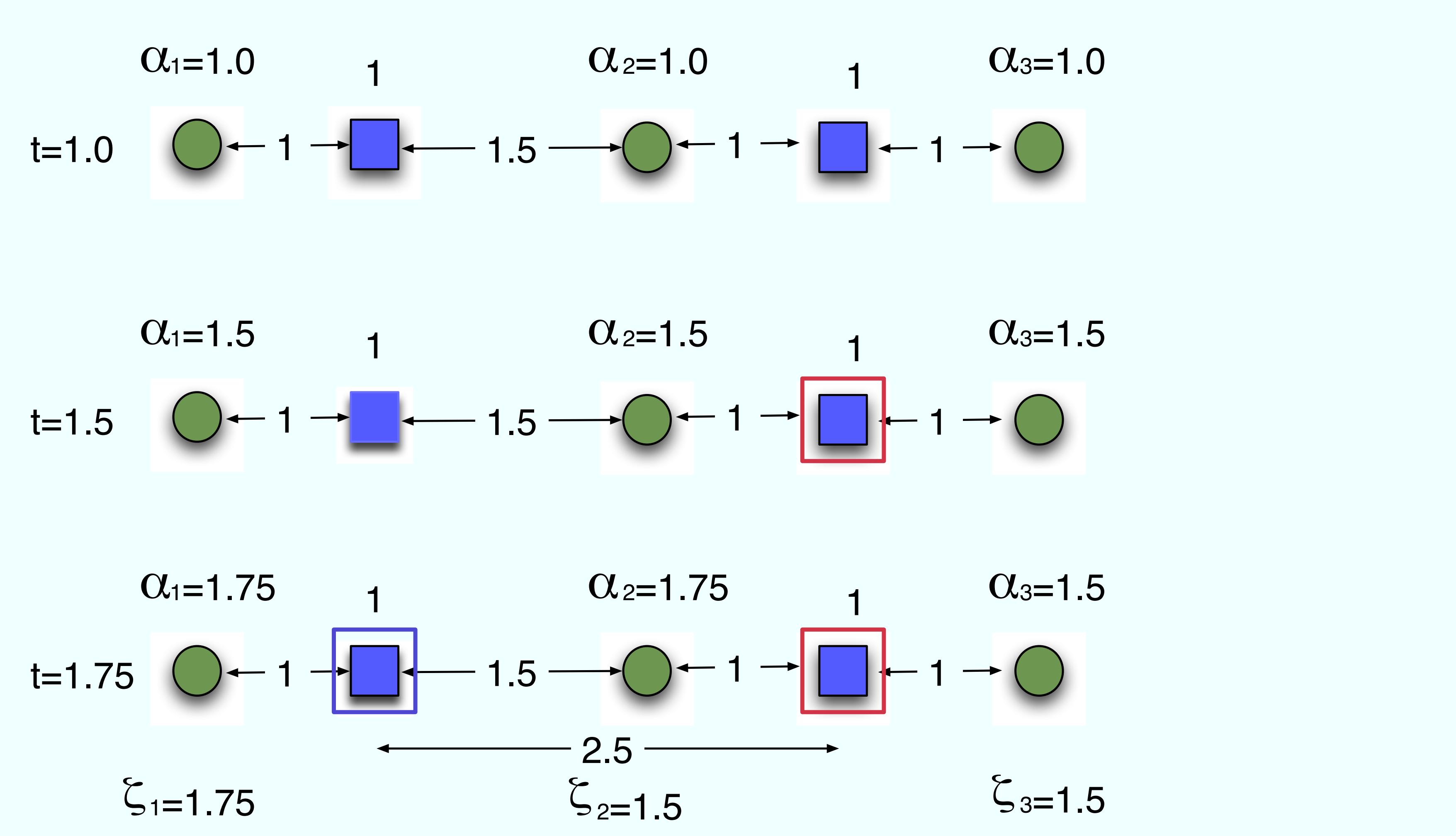
Cost-Sharing Mechanisms

Group-strategyproof for Facility location
- The Mechanism
- Cost-shares
- Example of execution of the algorithm

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds

# Steiner Forests

# Steiner forests


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- **S**teiner forests

# Steiner forests


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- **S**teiner forests

  Input:
  - undirected graph $G = (V, E)$;
  - non-negative edge costs $c : E \to \mathbb{R}^+$;
  - terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V \times V$.

# Steiner forests


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- **S**teiner forests

  Input:
  - undirected graph $G = (V, E)$;
  - non-negative edge costs $c : E \to \mathbb{R}^+$;
  - terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V \times V$.

  Goal:

  Compute min-cost forest $F$ in $G$ such that $s$ and $t$ are in same tree for all $(s, t) \in R$.

# Steiner forests


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- **S**teiner forests

  Input:
  - undirected graph $G = (V, E)$;
  - non-negative edge costs $c : E \to \mathbb{R}^+$;
  - terminal-pairs $R = \{(s_1, t_1), \ldots, (s_k, t_k)\} \subseteq V \times V$.

  Goal:

  Compute min-cost forest $F$ in $G$ such that $s$ and $t$ are in same tree for all $(s, t) \in R$.
- Special case: Steiner trees.

  Compute a min-cost tree spanning a teminal-set $R \subseteq V$.

# Steiner forests: Example


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

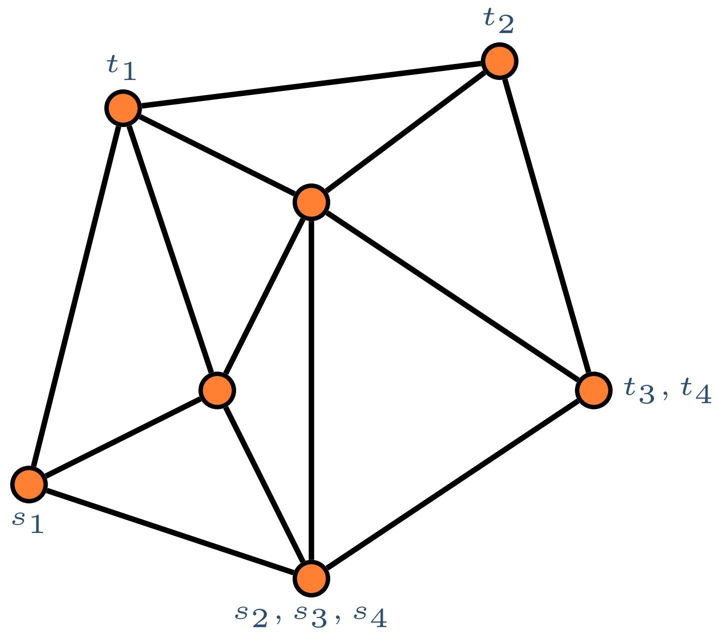
Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- Example with four terminal pairs: $R = \{(s_i, t_i)\}_{1 \leq i \leq 4}$
- All edges have unit cost.



Algorithm and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Overview of the tutorial
Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

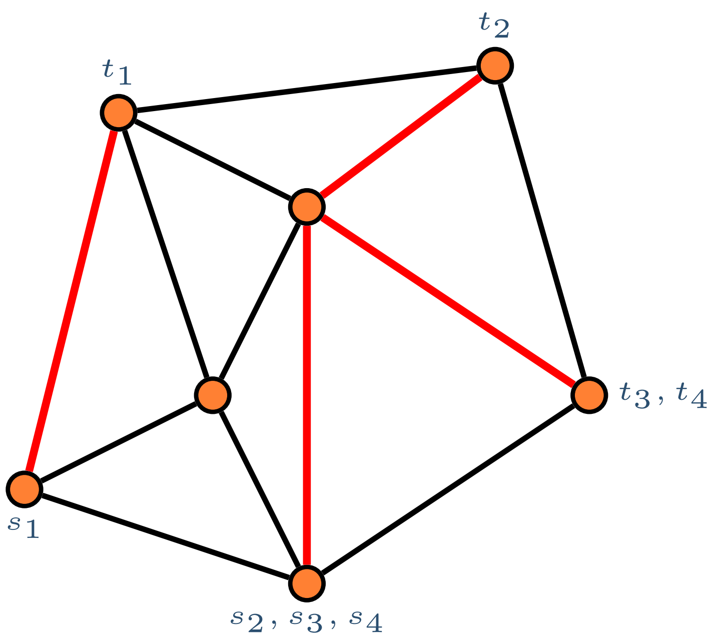
Steiner Forests
Steiner forests
Steiner forests: Example
Our Result
Primal-Dual
Primal LP: Steiner Cuts
Dual LP
Pictorial View
Algorithm SF: Example
PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds

# Steiner forests: Example

- Example with four terminal pairs: $R = \{(s_i, t_i)\}_{1 \leq i \leq 4}$
- All edges have unit cost.




Total cost is 4!

# Previous Work and cross-monotonic result


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- [Agrawal, Klein, Ravi ’95] (see also [Goemans, Williamson ’95]):
  Primal-dual $2$-approximation for Steiner forests.

# Previous Work and cross-monotonic result


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- [Agrawal, Klein, Ravi '95] (see also [Goemans, Williamson '95]):
  Primal-dual $2$-approximation for Steiner forests.
- [Jain, Vazirani '01]:
  Group-strategyproof cost-sharing mechanism for Steiner trees that satisfies competitiveness and $2$-budget balance.

# Previous Work and cross-monotonic result

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds

- [Agrawal, Klein, Ravi '95] (see also [Goemans, Williamson '95]):
  Primal-dual $2$-approximation for Steiner forests.
- [Jain, Vazirani '01]:
  Group-strategyproof cost-sharing mechanism for Steiner trees that satisfies competitiveness and $2$-budget balance.
- **[Könemann, L., Schäfer, 2005]:**
  Group-strategyproof cost-sharing mechanism for Steiner forests that satisfies competitiveness and $2$-budget balance.

# Previous Work and cross-monotonic result


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- [Agrawal, Klein, Ravi '95] (see also [Goemans, Williamson '95]):
  Primal-dual $2$-approximation for Steiner forests.
- [Jain, Vazirani '01]:
  Group-strategyproof cost-sharing mechanism for Steiner trees that satisfies competitiveness and $2$-budget balance.
- **[Könemann, L., Schäfer, 2005]:**
  Group-strategyproof cost-sharing mechanism for Steiner forests that satisfies competitiveness and $2$-budget balance.

# Steiner Forests: Primal-dual algorithm


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- We sketch primal-dual algorithm SF due to [Agrawal, Klein, Ravi '95] (see also [Goemans, Williamson '95]).

# Steiner Forests: Primal-dual algorithm


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- We sketch primal-dual algorithm SF due to [Agrawal, Klein, Ravi '95] (see also [Goemans, Williamson '95]).
- Algorithm SF computes
  - feasible Steiner forest $F$, and
  - feasible dual solution $y$

  at the same time.

  Key trick: Use dual $y$ and weak duality to bound cost of $F$.

# Primal LP: Steiner Cuts


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- Primal has variables $x_e$ for all $e \in E$.
  $x_e = 1$ if $e$ is in Steiner forest, $0$ otherwise

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

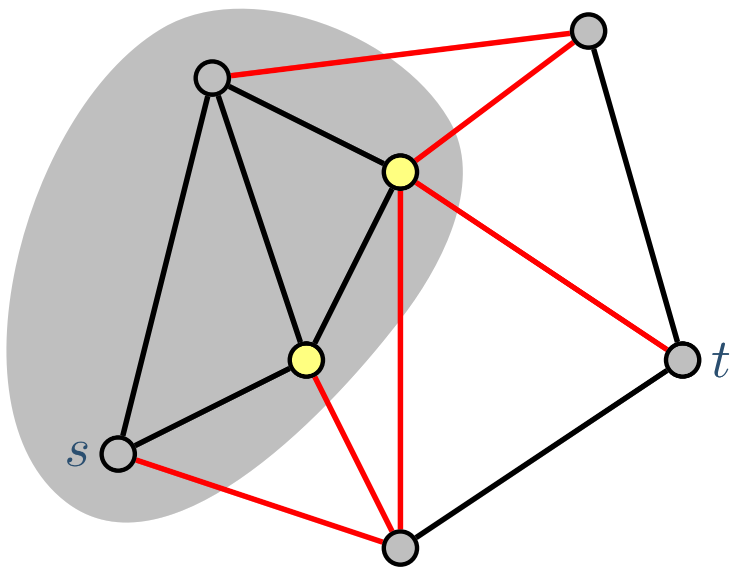
Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

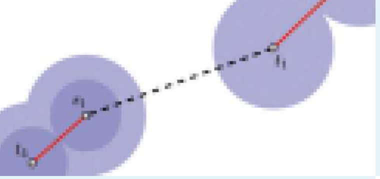
Lower Bounds


# Primal LP: Steiner Cuts

- Primal has variables $x_e$ for all $e \in E$.
  $x_e = 1$ if $e$ is in Steiner forest, $0$ otherwise
- Steiner cut: Subset of nodes that separates at least one terminal pair $(s, t) \in R$.




Any feasible Steiner forest must contain at least one of the red edges!



# Primal LP: Steiner Cuts


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

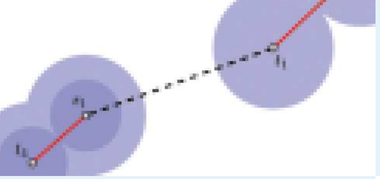
Steiner Forest CS-Mechanism

Lower Bounds


Primal LP has one constraint for each Steiner cut.

$$\begin{array}{lrcll} \text{min} & \sum_{e \in E} c_e x_e & & & \\ \text{s.t.} & \sum_{e \in \delta(U)} x_e & \geq & 1 & \forall \text{ Steiner cut } U \\ & x_e & \geq & 0 & \forall e \in E \end{array}$$

$\delta(U)$: Edges with exactly one endpoint in $U$.



# Steiner trees: Dual LP


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


Dual LP has a variable $y_U$ for all Steiner cuts $U$.

$$\begin{array}{rlcll} \text{max} & \sum_{U} y_U & & & \\ \text{s.t.} & \sum_{U:\ e \in \delta(U)} y_U & \leq & c_e & \forall e \in E \\ & y_U & \geq & 0 & \forall U \end{array}$$

$\delta(U)$: Edges with exactly one endpoint in $U$.

# Dual LP: Pictorial View


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- Can visualize $y_U$ as disks around $U$ with radius $y_U$.
  Example: Terminal pair $(s, t) \in R$, edge $(s, t)$ with cost $4$


$$y_s = y_t = 0$$

Algorithm and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Overview of the tutorial
Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

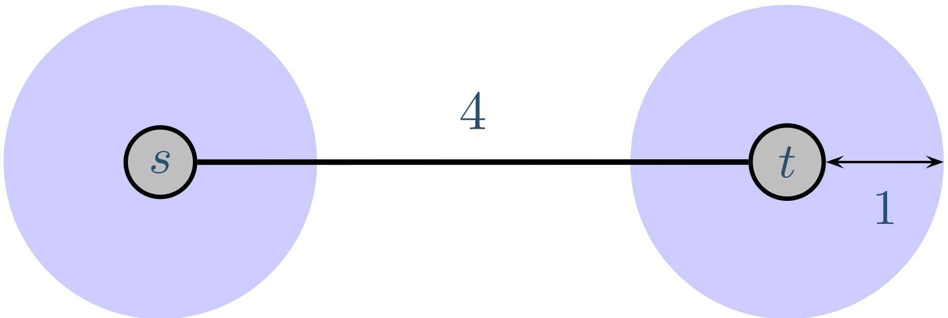
Steiner Forests
Steiner forests
Steiner forests: Example
Our Result
Primal-Dual
Primal LP: Steiner Cuts
Dual LP
Pictorial View
Algorithm SF: Example
PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


# Dual LP: Pictorial View

- Can visualize $y_U$ as disks around $U$ with radius $y_U$.
  Example: Terminal pair $(s,t) \in R$, edge $(s,t)$ with cost $4$




$$y_s = y_t = 1$$

# Dual LP: Pictorial View


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

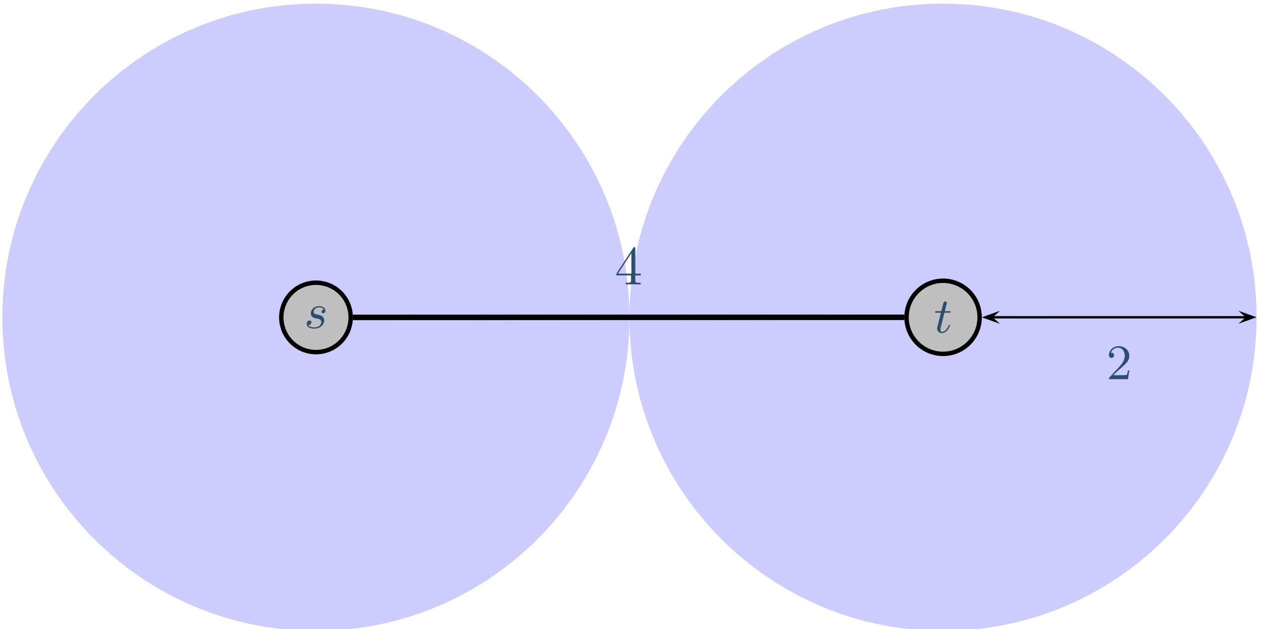
Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


- Can visualize $y_U$ as disks around $U$ with radius $y_U$.
  Example: Terminal pair $(s,t) \in R$, edge $(s,t)$ with cost $4$




$y_s = y_t = 2$

Have: $y_s + y_t = 4 = c_{st}$. Edge $(s,t)$ is tight.

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

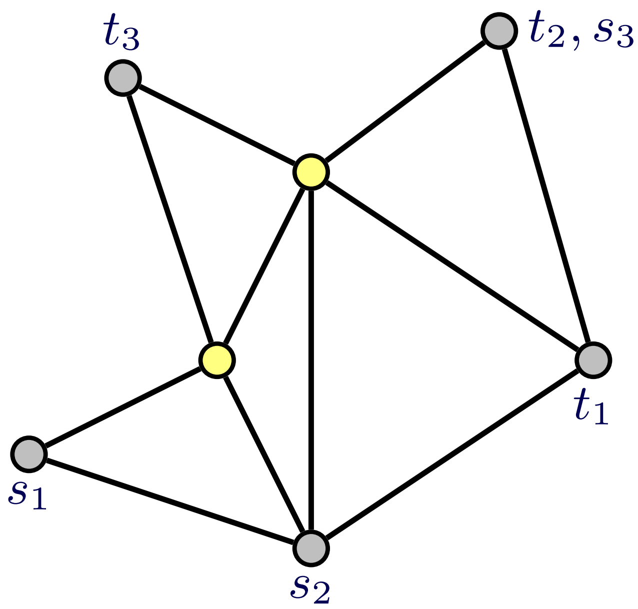
Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds

Algorithm grows duals of connected components.



- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

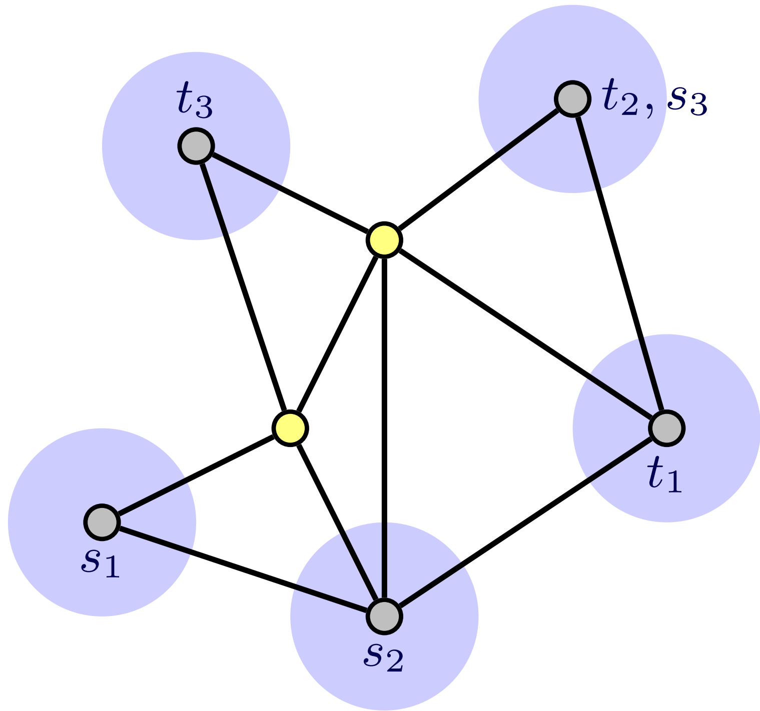
Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


Algorithm grows duals of connected components.



- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

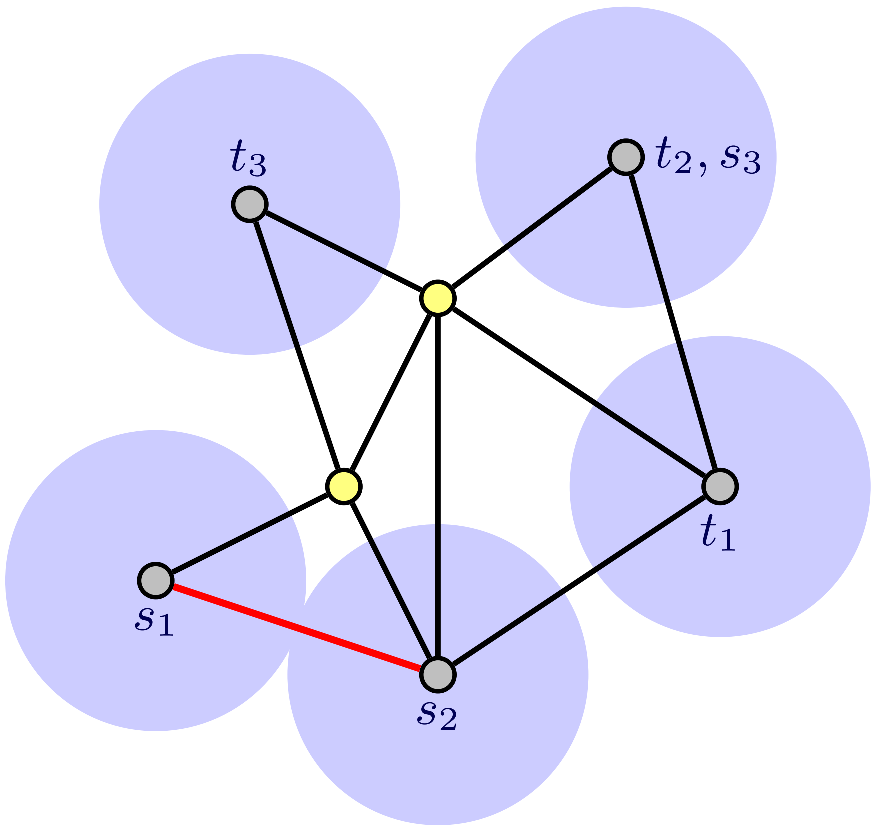
Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


Algorithm grows duals of connected components.



- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

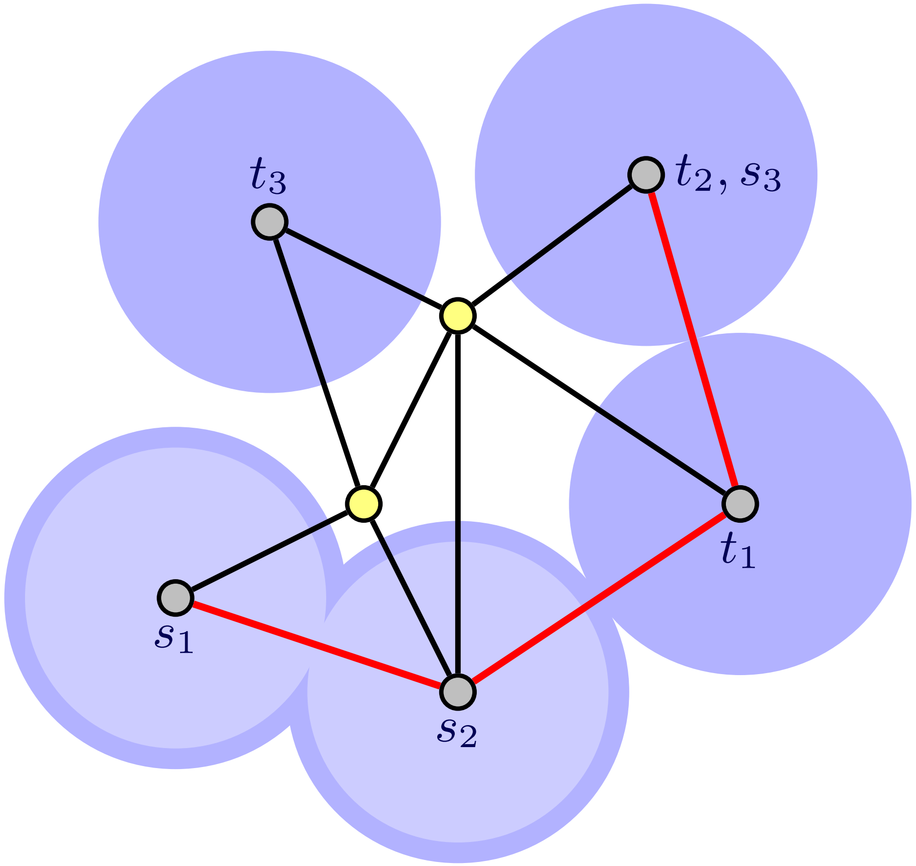
Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


# Algorithm SF: Example

Algorithm grows duals of connected components.



- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

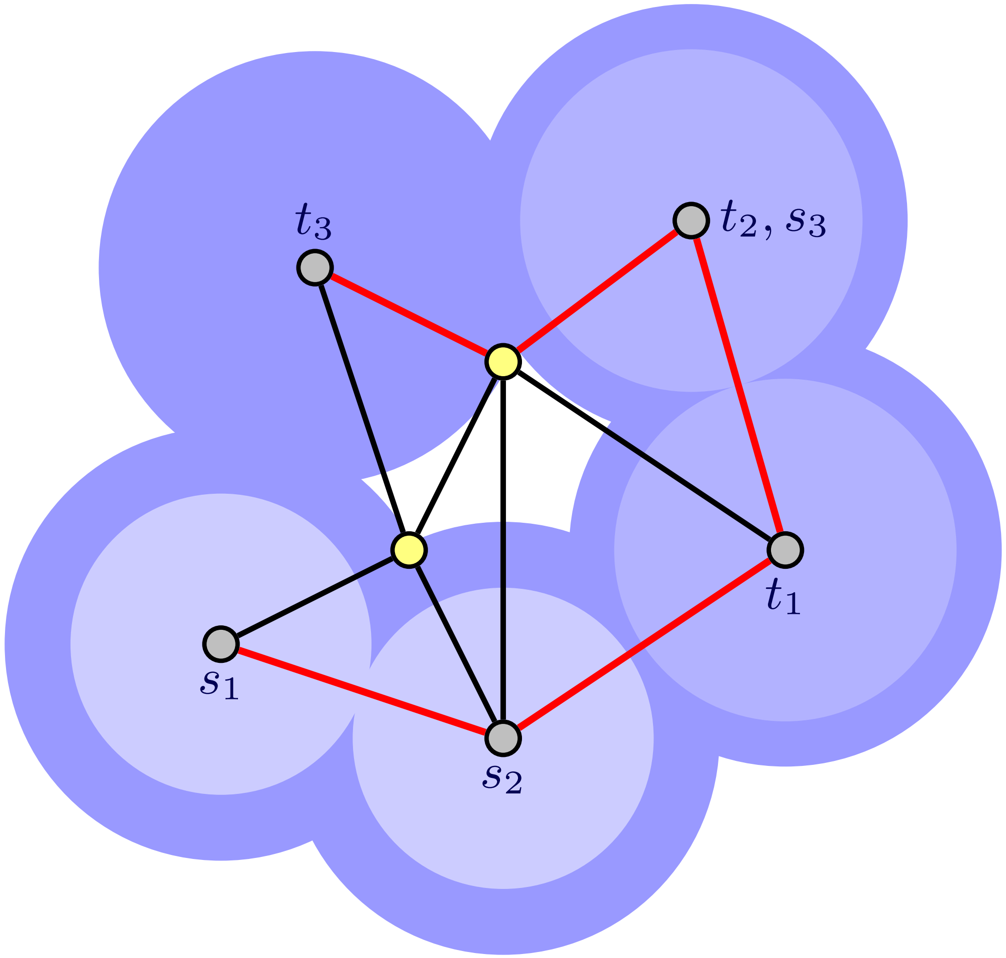
Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


Algorithm grows duals of connected components.



# PD-Algorithm: Properties


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

Steiner Forest CS-Mechanism

Lower Bounds


**Theorem [Agrawal, Klein, Ravi ’95]:** Algorithm computes forest $F$ and dual $y$ such that

$$c(F) \leq (2 - 1/k) \cdot \sum_U y_U \leq (2 - 1/k) \cdot \mathtt{opt}_R.$$

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests
- Steiner forests
- Steiner forests: Example
- Our Result
- Primal-Dual
- Primal LP: Steiner Cuts
- Dual LP
- Pictorial View
- Algorithm SF: Example
- PD-Algorithm: Properties

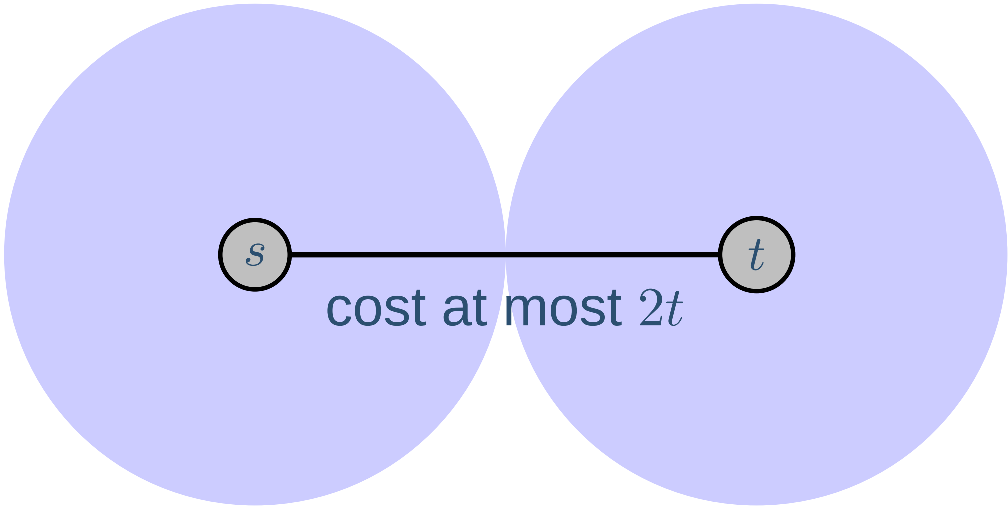
Steiner Forest CS-Mechanism

Lower Bounds


**Theorem [Agrawal, Klein, Ravi '95]:** Algorithm computes forest $F$ and dual $y$ such that

$$c(F) \leq (2 - 1/k) \cdot \sum_U y_U \leq (2 - 1/k) \cdot \texttt{opt}_R.$$

Main trick: Edge $(s, t)$ becomes tight at time $t$.




Use twice the dual around $s$ and $t$ to pay for cost of path.

Algorithm and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Overview of the tutorial
Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds

# Steiner Forest Cost-Sharing Mechanism

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

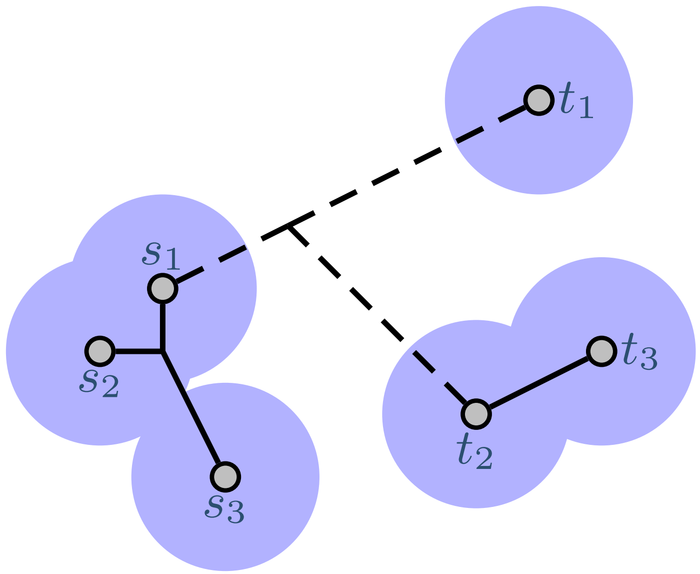
Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Say: terminal pair $(s, t)$ is active at time $t$ if $s$ and $t$ are not in same moat.
  Example: All terminals are active.

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

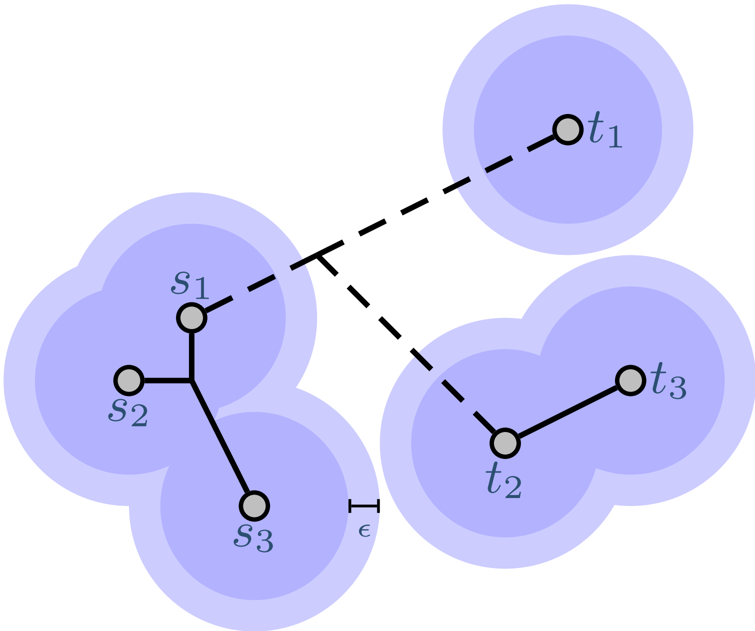
Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Say: terminal pair $(s, t)$ is active at time $t$ if $s$ and $t$ are not in same moat.
  Example: All terminals are active.
- Grow active moats by $\epsilon$.

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

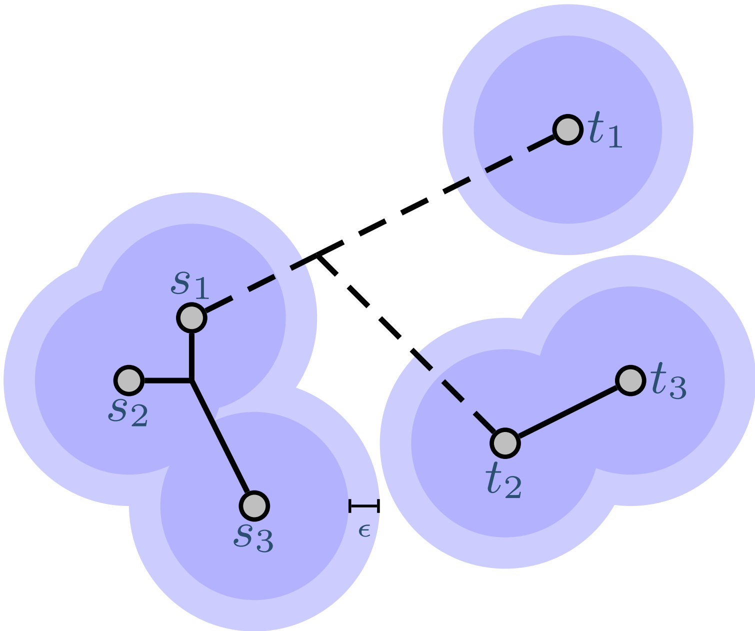
Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Say: terminal pair $(s, t)$ is active at time $t$ if $s$ and $t$ are not in same moat.
  Example: All terminals are active.
- Grow active moats by $\epsilon$.
- Growth of moats is shared among active terminals.

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

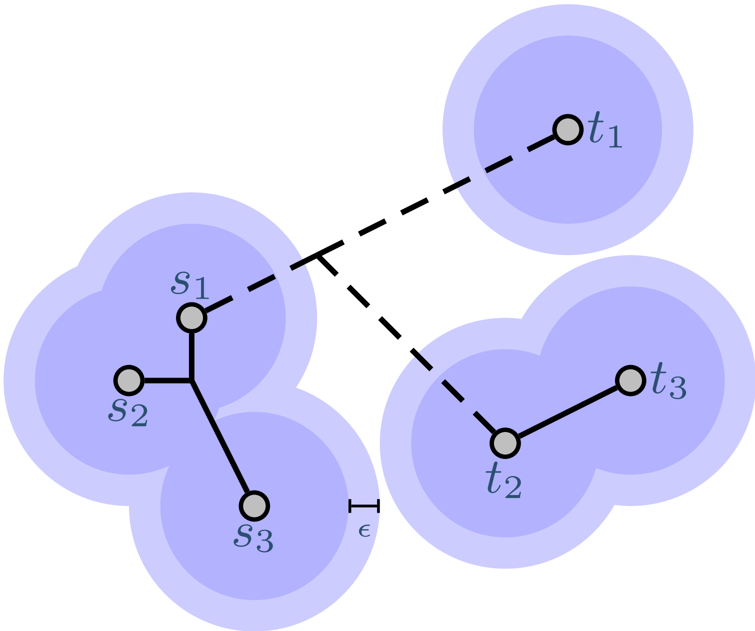
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds

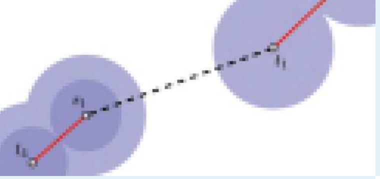




- Say: terminal pair $(s, t)$ is **active** at time $t$ if $s$ and $t$ are not in same moat. Example: All terminals are active.
- Grow active moats by $\epsilon$.
- Growth of moats is **shared** among active terminals.
- Cost-share increase for ...

$$s_1 : \epsilon/3$$

$$t_2 : \epsilon/2$$

$$t_1 : \epsilon$$



# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

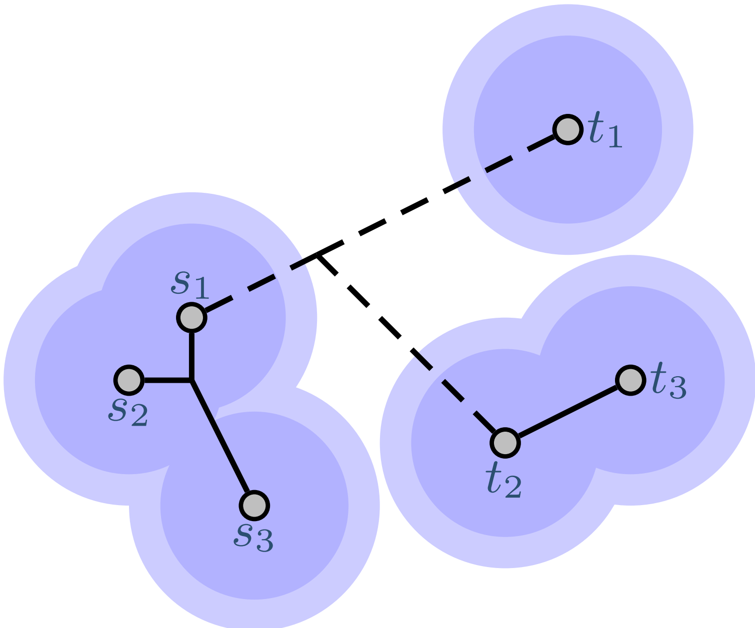
Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- $U^t(r)$ : moat of terminal $r$ at time $t$.

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

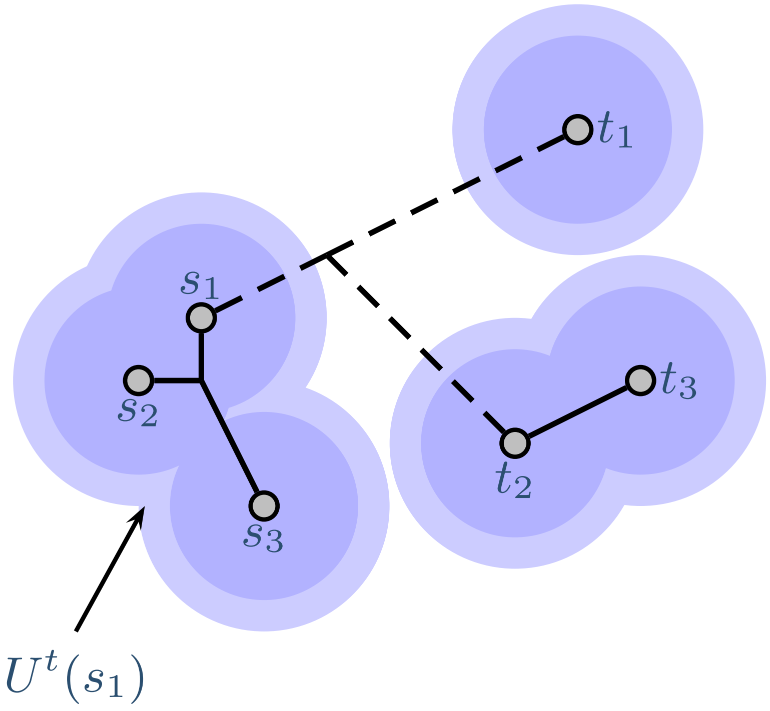
Cost-Sharing Mechanisms

Group-strategyproof for Facility location

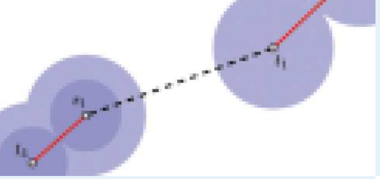
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

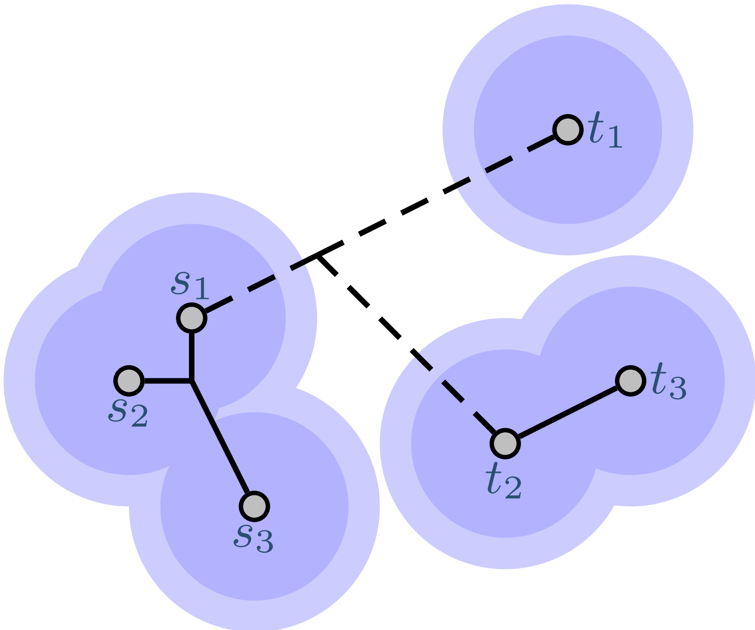
Lower Bounds


- $U^t(r)$ : moat of terminal $r$ at time $t$.



# Try 1: SF and Shapley Value




- $U^t(r)$ : moat of terminal $r$ at time $t$.
- $a^t(r)$ : number of active terminals in $U^t(r)$; e.g., $a^t(s_1) = 3$.


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

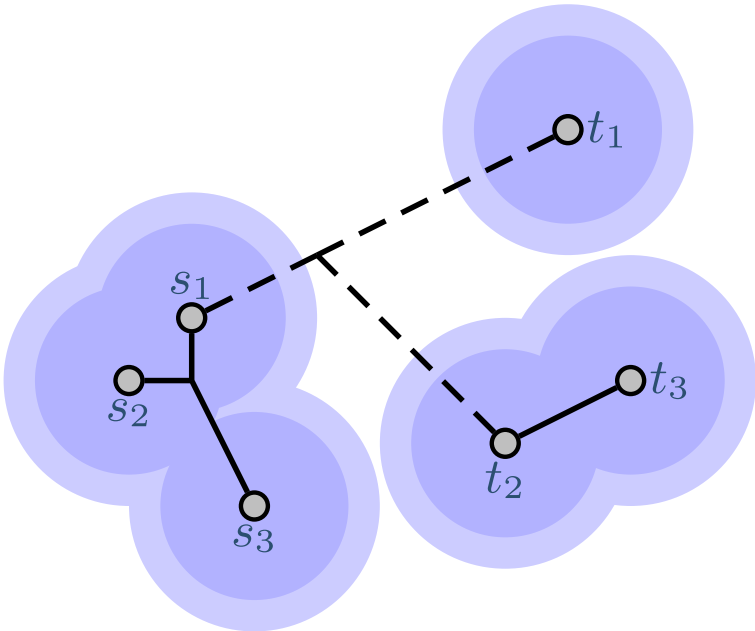
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds

Algorithm and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Approximation and Mechanism Design
Overview of the tutorial
Talk Outline
Cost-Sharing Mechanisms
Group-strategyproof for Facility location
Steiner Forests
Steiner Forest CS-Mechanism
Try 1: SF and Shapley Value
Try 2: Independent Activity Time
Proving Cross-Monotonicity
Proving Cost Recovery and Competitiveness
Bounding $\sum_r \xi_R(r)$
Lower Bounds


# Try 1: SF and Shapley Value




- $U^t(r)$ : moat of terminal $r$ at time $t$.
- $a^t(r)$ : number of active terminals in $U^t(r)$; e.g., $a^t(s_1) = 3$.
- Suppose terminal $r \in R$ becomes inactive at time $T$. Cost-share:

$$\xi_Q(r) = \int_0^T \frac{1}{a^t(r)}\, dt$$

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

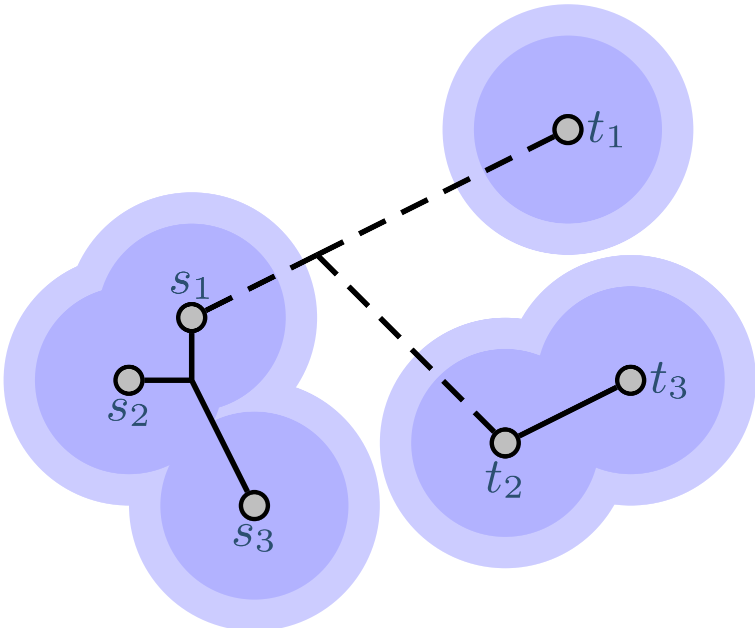
Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- $U^t(r)$ : moat of terminal $r$ at time $t$.
- $a^t(r)$ : number of active terminals in $U^t(r)$; e.g., $a^t(s_1) = 3$.
- Suppose terminal $r \in R$ becomes inactive at time $T$. Cost-share:

$$\xi_Q(r) = \int_0^T \frac{1}{a^t(r)}\, dt$$

- For terminal-pair $(s,t) \in R$:

$$\xi_Q(s,t) = \xi_Q(s) + \xi_Q(t)$$

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

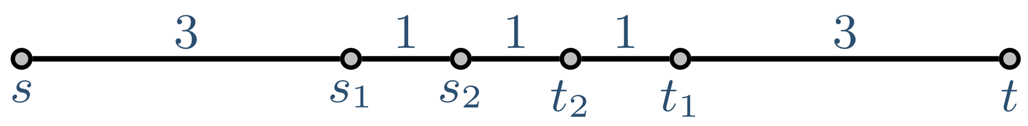
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


Q: Is $\xi$ cross-monotonic? A: No!

Simple example: $R = \{(s,t),(s_1,t_1),(s_2,t_2)\}$, $R_0 = R \setminus \{(s_2,t_2)\}$



# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

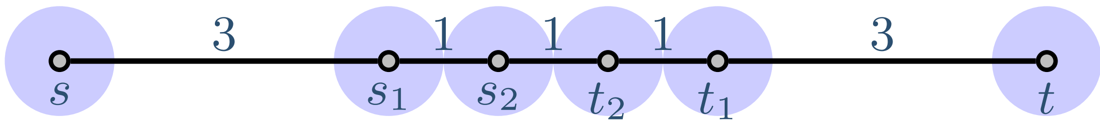
Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


Q: Is $\xi$ cross-monotonic? A: No!

Simple example: $R = \{(s,t),(s_1,t_1),(s_2,t_2)\}$, $R_0 = R \setminus \{(s_2,t_2)\}$

$t = 0.5$



# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

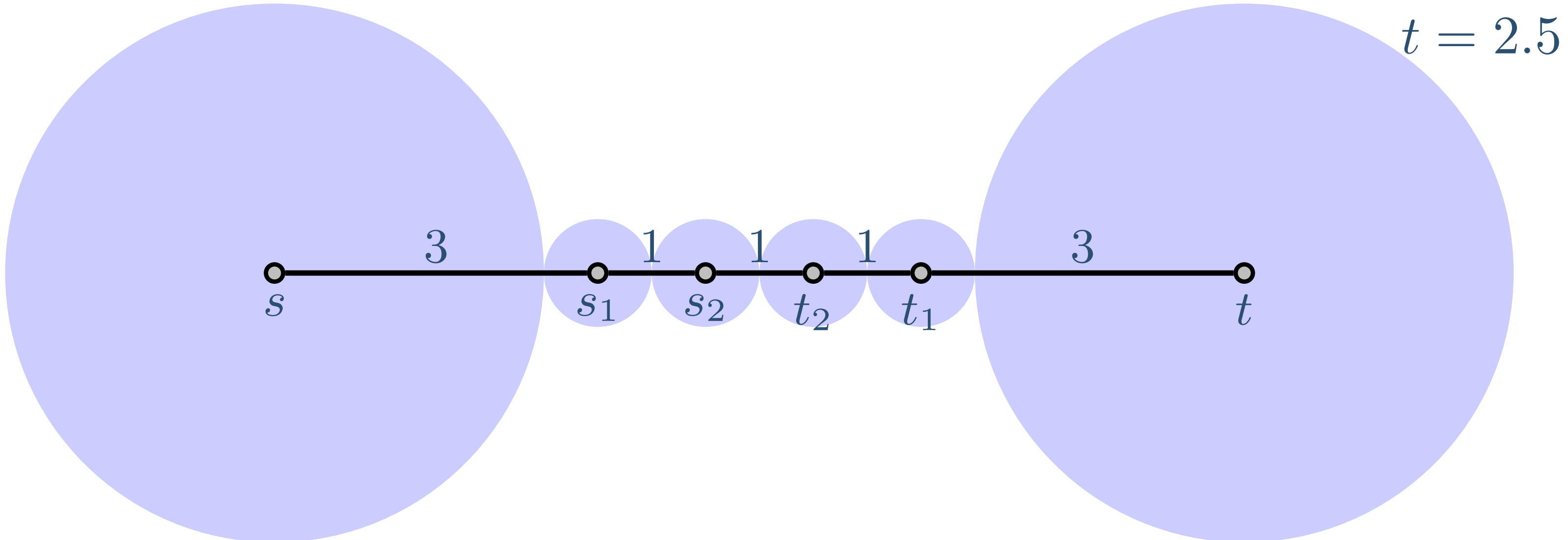
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


Q: Is $\xi$ cross-monotonic? A: No!

Simple example: $R = \{(s,t),(s_1,t_1),(s_2,t_2)\}$, $R_0 = R \setminus \{(s_2,t_2)\}$




- $\xi_R(s,t) = 5$

# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

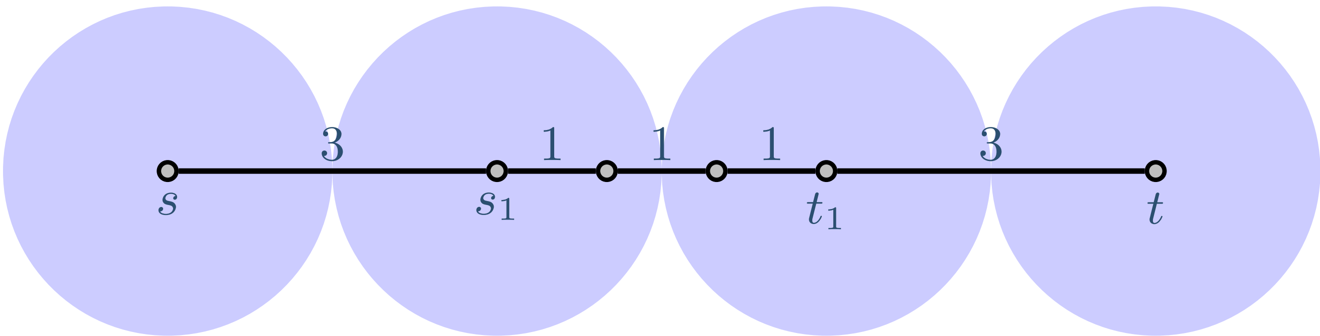
Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds

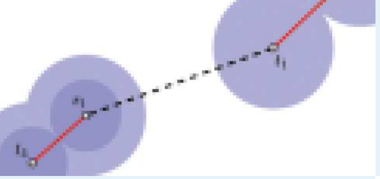

Q: Is $\xi$ cross-monotonic? A: No!

Simple example: $R = \{(s,t),(s_1,t_1),(s_2,t_2)\}$, $R_0 = R \setminus \{(s_2,t_2)\}$

$t = 1.5$




- $\xi_R(s,t) = 5$
- $\xi_{R_0}(s,t) = 3$



# Try 1: SF and Shapley Value


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

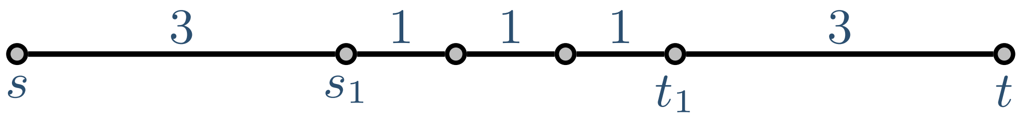
Steiner Forest CS-Mechanism

- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


Q: Is $\xi$ cross-monotonic? A: No!

Simple example: $R = \{(s,t),(s_1,t_1),(s_2,t_2)\}$, $R_0 = R \setminus \{(s_2,t_2)\}$




- $\xi_R(s,t) = 5$
- $\xi_{R_0}(s,t) = 3$
- Activity time of $(s,t)$ depends on $(s_2,t_2)$!

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Previous try: Activity-times of terminal pairs inter-dependent.

**[Könemann, L., Schäfer, van Zwam, 2008]:**

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

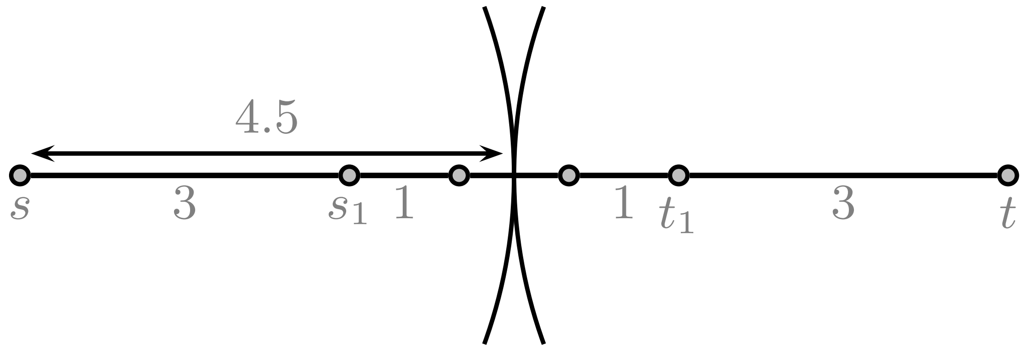
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Previous try: Activity-times of terminal pairs inter-dependent. How long would they need to connect if no other terminal was in the game?




**[Könemann, L., Schäfer, van Zwam, 2008]:**

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

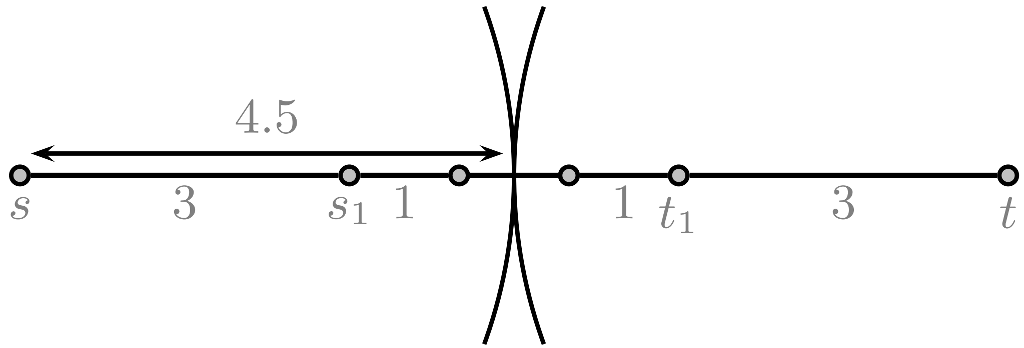
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Previous try: Activity-times of terminal pairs inter-dependent.
  How long would they need to connect if no other terminal was in the game?




- Death time of terminal-pair $(s,t) \in R$:

$$\mathrm{d}(s,t) = \frac{c(s,t)}{2},$$

where $c(s,t)$ is cost of minimum-cost $s,t$-path.

**[Könemann, L., Schäfer, van Zwam, 2008]:**

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

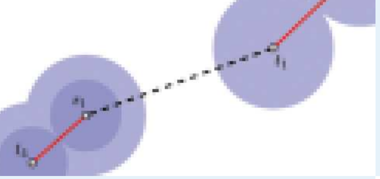
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: $\mathrm{d}(r) = \mathrm{d}(s,t)$ for $r \in \{s,t\}$.



# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: d$(r) =$ d$(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time d$(r)$.

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: $\mathrm{d}(r) = \mathrm{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathrm{d}(r)$.
- SF grows moats as long as they contain active terminals.

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt.$$

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

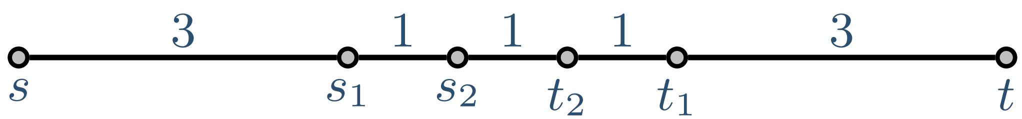
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt.$$



# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

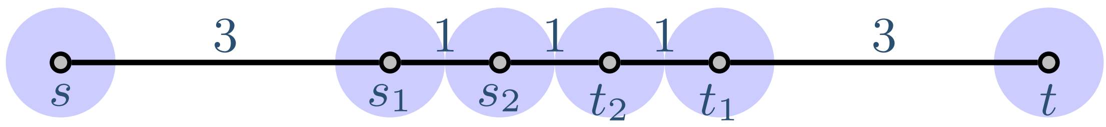
Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt.$$

$t = 0.5$



# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

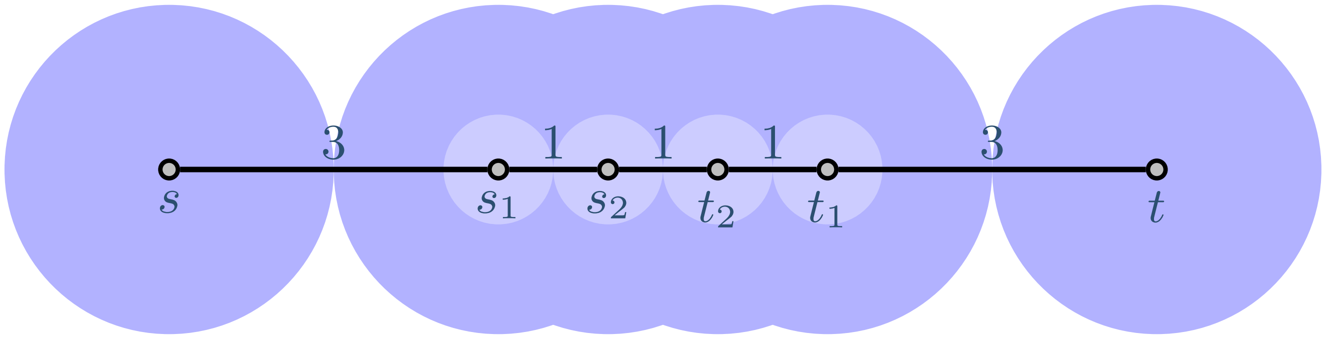
Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds

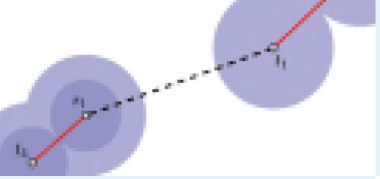

- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\,dt.$$

$t = 1.5$




- $\xi_R(s_1, t_1) = 2$



# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

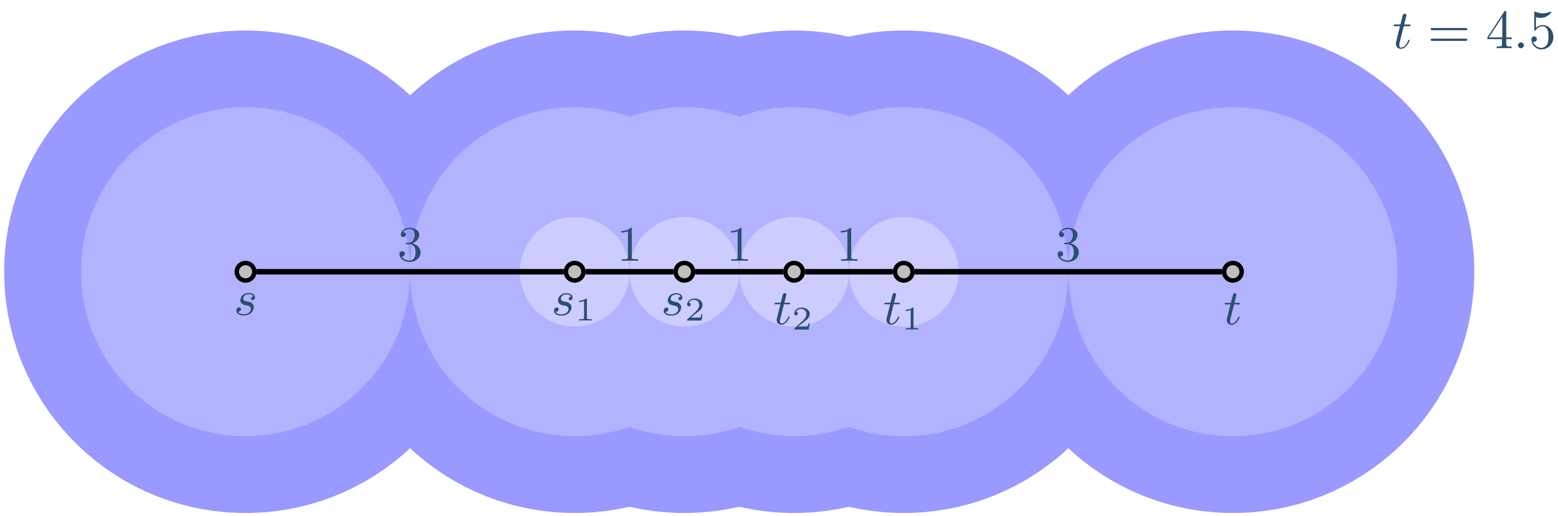
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

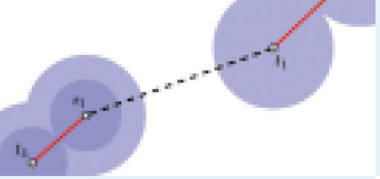
Lower Bounds


- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt.$$




- $\xi_R(s_1,t_1) = 2$, $\xi_R(s,t) = 6$.



# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

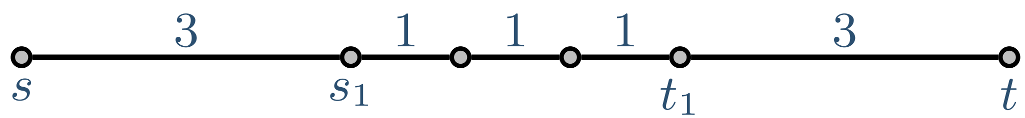
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt.$$




- $\xi_R(s_1,t_1) = 2$, $\xi_R(s,t) = 6$.

# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

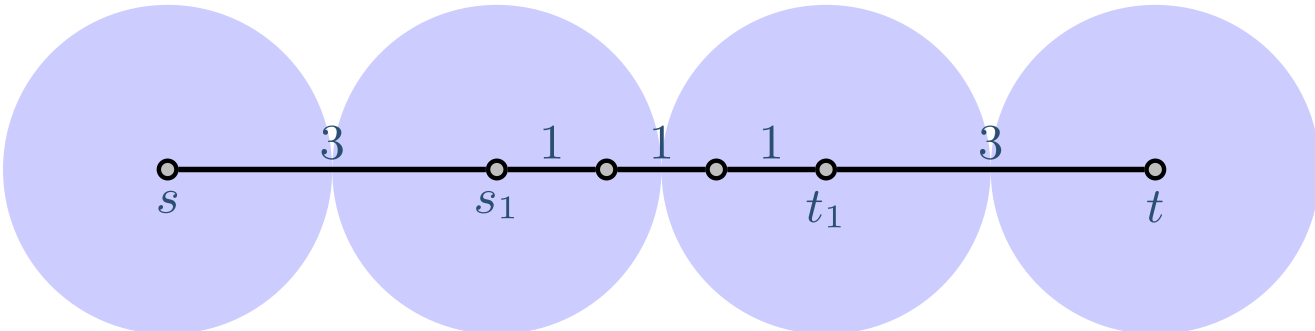
Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds

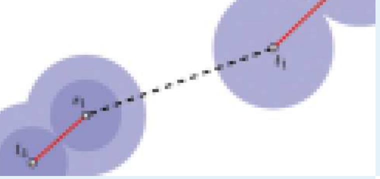

- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt.$$

$t = 1.5$




- $\xi_R(s_1,t_1) = 2$, $\xi_R(s,t) = 6$.
- $\xi_{R_0}(s_1,t_1) = 3$



# Try 2: Independent Activity Time


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

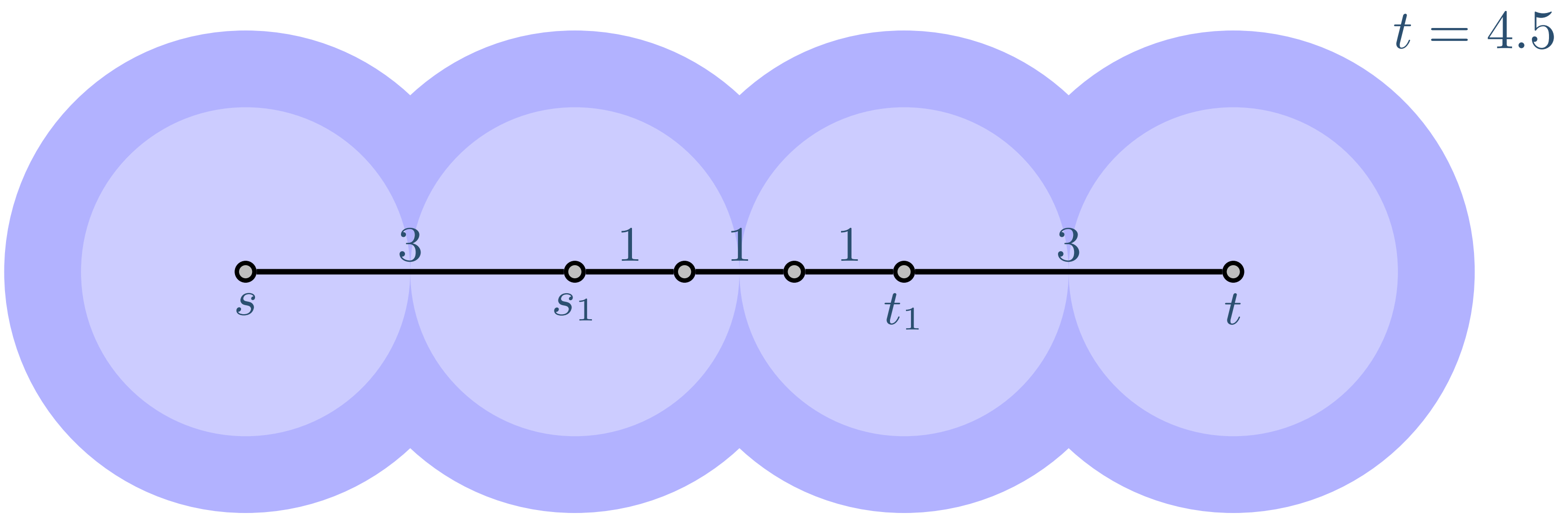
Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


- Extend to terminal nodes: $\mathsf{d}(r) = \mathsf{d}(s,t)$ for $r \in \{s,t\}$.
- Terminal $r$ is active until time $\mathsf{d}(r)$.
- SF grows moats as long as they contain active terminals.
- Cost-share of terminal $r$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt.$$




- $\xi_R(s_1,t_1) = 2$, $\xi_R(s,t) = 6$.
- $\xi_{R_0}(s_1,t_1) = 3$, $\xi_{R_0}(s,t) = 6$.

# Proving Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

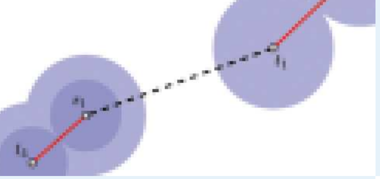
Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ is cross-monotonic.

Proof:

- $R_0 = R \setminus \{(s,t)\}$.



# Proving Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ is cross-monotonic.

Proof:

- $R_0 = R \setminus \{(s,t)\}$.
- $U_0^t(r)$: Moat of $r$ at time $t$ in SF$(R_0)$.

# Proving Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

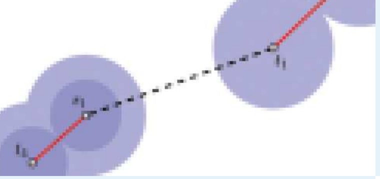
Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ is cross-monotonic.

Proof:

- $R_0 = R \setminus \{(s,t)\}$.
- $U_0^t(r)$: Moat of $r$ at time $t$ in SF$(R_0)$.
- $a_0^t(r)$: Number of active terminals in $U_0^t(r)$.



# Proving Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ is cross-monotonic.

Proof:

- $R_0 = R \setminus \{(s,t)\}$.
- $U_0^t(r)$: Moat of $r$ at time $t$ in SF$(R_0)$.
- $a_0^t(r)$: Number of active terminals in $U_0^t(r)$.
- Death-times of terminal-pairs are instance independent! Therefore: For each $r \in R_0$:

$$U_0^t(r) \text{ active } \Longrightarrow U^t(r) \text{ active and } U_0^t(r) \subseteq U^t(r).$$

# Proving Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds

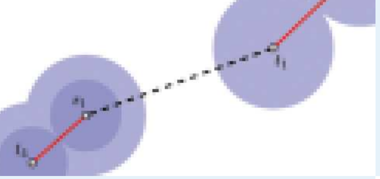

**Lemma:** $\xi$ is cross-monotonic.

Proof:

- $R_0 = R \setminus \{(s,t)\}$.
- $U_0^t(r)$: Moat of $r$ at time $t$ in SF$(R_0)$.
- $a_0^t(r)$: Number of active terminals in $U_0^t(r)$.
- Death-times of terminal-pairs are instance independent! Therefore: For each $r \in R_0$:

$$U_0^t(r) \text{ active } \Longrightarrow U^t(r) \text{ active and } U_0^t(r) \subseteq U^t(r).$$

- Implies: $a_0^t(r) \leq a^t(r)$ for all $t \geq 0$ and $r \in R_0$.



# Proving Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ is cross-monotonic.

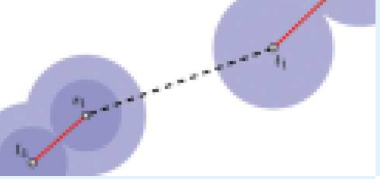
Proof:

- $R_0 = R \setminus \{(s,t)\}$.
- $U_0^t(r)$: Moat of $r$ at time $t$ in SF$(R_0)$.
- $a_0^t(r)$: Number of active terminals in $U_0^t(r)$.
- Death-times of terminal-pairs are instance independent! Therefore: For each $r \in R_0$:

$$U_0^t(r) \text{ active } \implies U^t(r) \text{ active and } U_0^t(r) \subseteq U^t(r).$$

- Implies: $a_0^t(r) \leq a^t(r)$ for all $t \geq 0$ and $r \in R_0$.
- We obtain: For each $r \in R_0$:

$$\xi_R(r) = \int_0^{\mathsf{d}(r)} \frac{1}{a^t(r)}\, dt \leq \int_0^{\mathsf{d}(r)} \frac{1}{a_0^t(r)}\, dt = \xi_{R_0}(r).$$



# Proving Cost Recovery and Competitiveness


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ satisfies cost recovery and $2$-approximate competitiveness.

Proof:

- Let $F$ and $y$ be forest and corresponding dual computed by SF.

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


# Proving Cost Recovery and Competitiveness

**Lemma:** $\xi$ satisfies cost recovery and $2$-approximate competitiveness.

Proof:

- Let $F$ and $y$ be forest and corresponding dual computed by SF.
- SF-Theorem implies

$$c(F) \leq 2 \cdot \sum_{U \subseteq V} y_U = 2 \cdot \sum_{r \in R} \xi_R(r).$$

$y$ is not dual feasible! Some active moats do not correspond to Steiner cuts.

# Proving Cost Recovery and Competitiveness


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ satisfies cost recovery and $2$-approximate competitiveness.

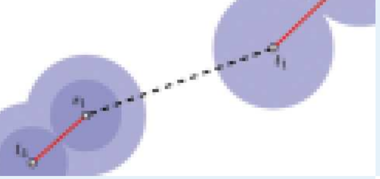
Proof:

- Let $F$ and $y$ be forest and corresponding dual computed by SF.
- SF-Theorem implies

$$c(F) \le 2 \cdot \sum_{U \subseteq V} y_U = 2 \cdot \sum_{r \in R} \xi_R(r).$$

$y$ is not dual feasible! Some active moats do not correspond to Steiner cuts.

- Can show: $\sum_{r \in R} \xi_R(r) \le \texttt{opt}_R$.



# Proving Cost Recovery and Competitiveness


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism
- Try 1: SF and Shapley Value
- Try 2: Independent Activity Time
- Proving Cross-Monotonicity
- Proving Cost Recovery and Competitiveness
- Bounding $\sum_r \xi_R(r)$

Lower Bounds


**Lemma:** $\xi$ satisfies cost recovery and $2$-approximate competitiveness.

Proof:

- Let $F$ and $y$ be forest and corresponding dual computed by SF.
- SF-Theorem implies

$$c(F) \leq 2 \cdot \sum_{U \subseteq V} y_U = 2 \cdot \sum_{r \in R} \xi_R(r).$$

  $y$ is not dual feasible! Some active moats do not correspond to Steiner cuts.
- Can show: $\sum_{r \in R} \xi_R(r) \leq \texttt{opt}_R$.
- This implies:

$$c(F) \leq 2 \cdot \sum_{r \in R} \xi_R(r) \leq 2 \cdot \texttt{opt}_R.$$

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds

- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


# Lower bounds for cross-monotonic cost-sharing mechanisms

# Lower Bound for Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- [Immorlica, Mahdian, Mirrokni ’05]: Give bounds on budget balance of cross-monotonic cost-sharing methods for facility location (3), vertex cover ($n^{1/3}$) and edge cover (2).

# Lower Bound for Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- [Immorlica, Mahdian, Mirrokni '05]: Give bounds on budget balance of cross-monotonic cost-sharing methods for facility location (3), vertex cover ($n^{1/3}$) and edge cover (2).
- We prove a lower bound of $2$ for Steiner trees.

# Lower Bound for Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- [Immorlica, Mahdian, Mirrokni '05]: Give bounds on budget balance of cross-monotonic cost-sharing methods for facility location (3), vertex cover ($n^{1/3}$) and edge cover (2).
- We prove a lower bound of $2$ for Steiner trees.
- $\Rightarrow$ our result for Steiner forest is tight.

# Lower Bound for Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- [Immorlica, Mahdian, Mirrokni '05]: Give bounds on budget balance of cross-monotonic cost-sharing methods for facility location (3), vertex cover ($n^{1/3}$) and edge cover (2).
- We prove a lower bound of $2$ for Steiner trees.
- $\Rightarrow$ our result for Steiner forest is tight.

- Lower bounds are irrespective of time complexity.

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


# Lower Bound for Cross-Monotonicity

- [Immorlica, Mahdian, Mirrokni '05]: Give bounds on budget balance of cross-monotonic cost-sharing methods for facility location (3), vertex cover ($n^{1/3}$) and edge cover (2).
- We prove a lower bound of $2$ for Steiner trees.
- $\Rightarrow$ our result for Steiner forest is tight.

- Lower bounds are irrespective of time complexity.
- Proofs exploit the core property (weaker than cross-monotonicity):

$$\forall Q \subseteq V, \sum_{j \in Q} \xi_V(j) \leq \texttt{opt}_Q$$

# Lower Bound for Cross-Monotonicity


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- [Immorlica, Mahdian, Mirrokni '05]: Give bounds on budget balance of cross-monotonic cost-sharing methods for facility location (3), vertex cover ($n^{1/3}$) and edge cover (2).
- We prove a lower bound of $2$ for Steiner trees.
- $\Rightarrow$ our result for Steiner forest is tight.

- Lower bounds are irrespective of time complexity.
- Proofs exploit the core property (weaker than cross-monotonicity):

$$\forall Q \subseteq V, \sum_{j \in Q} \xi_V(j) \leq \texttt{opt}_Q$$

- Turns into a lower bound on budget-balance of group-strategyproof methods only if there are no free riders.

# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

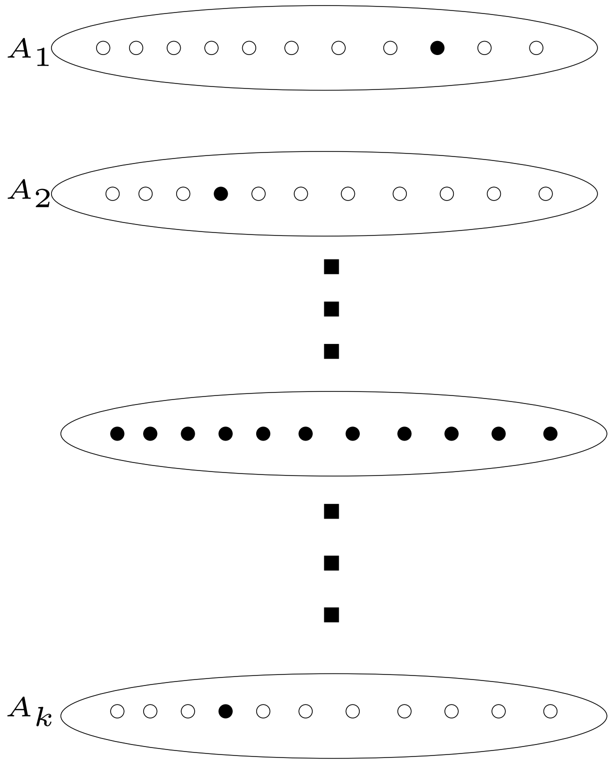
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $k$ pairwise disjoint classes $A_i$ of $m$ vertices.



# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

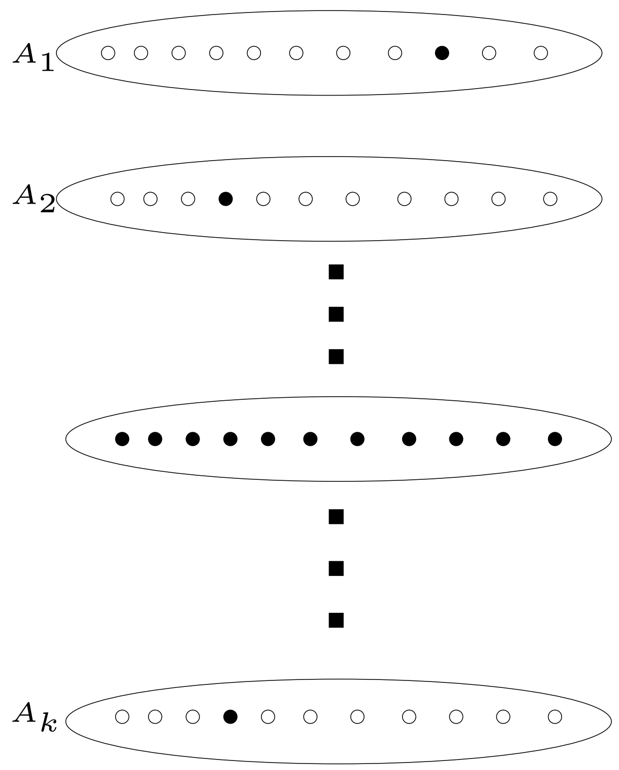
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $k$ pairwise disjoint classes $A_i$ of $m$ vertices.
- Select a random class $A_i = \{c_1, \ldots, c_m\}$.



# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

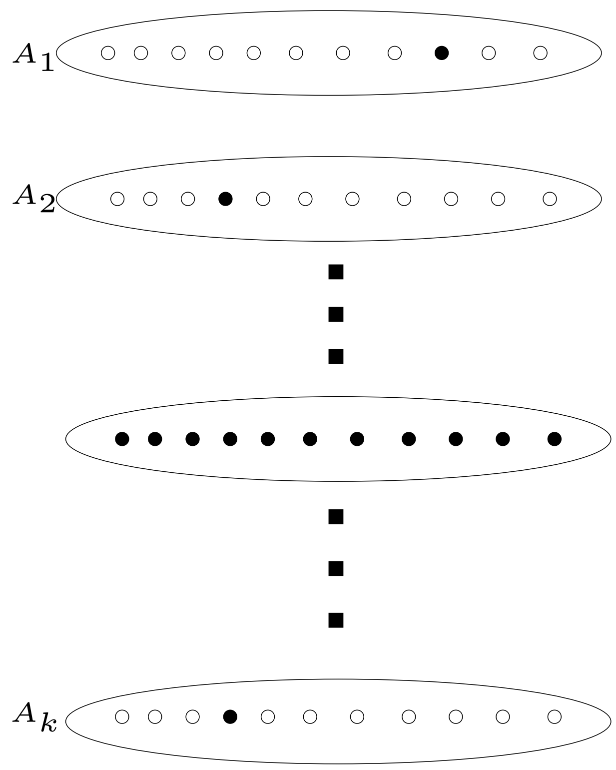
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $k$ pairwise disjoint classes $A_i$ of $m$ vertices.
- Select a random class $A_i = \{c_1, \ldots, c_m\}$.
- For each class $j \neq i$ select a random vertex $a_j$.



# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

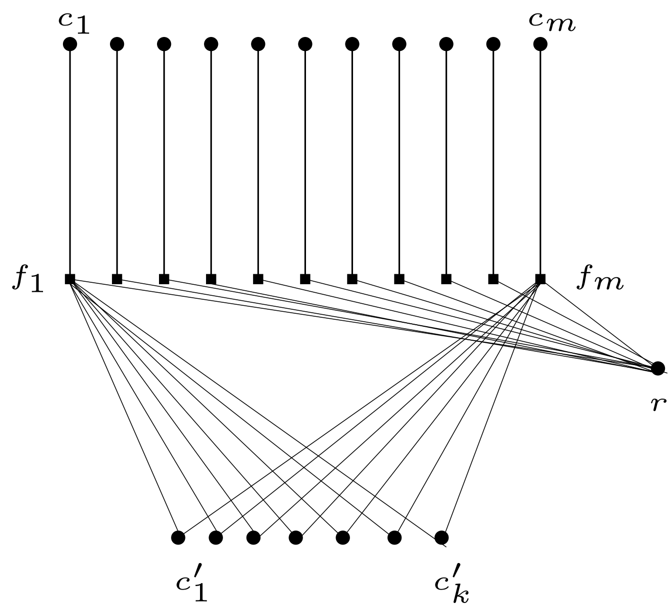
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $\mathcal{B} := \{\{a_1, \ldots, a_k\} : a_i \in A_i,\ i = 1, \ldots, k\}$.



# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

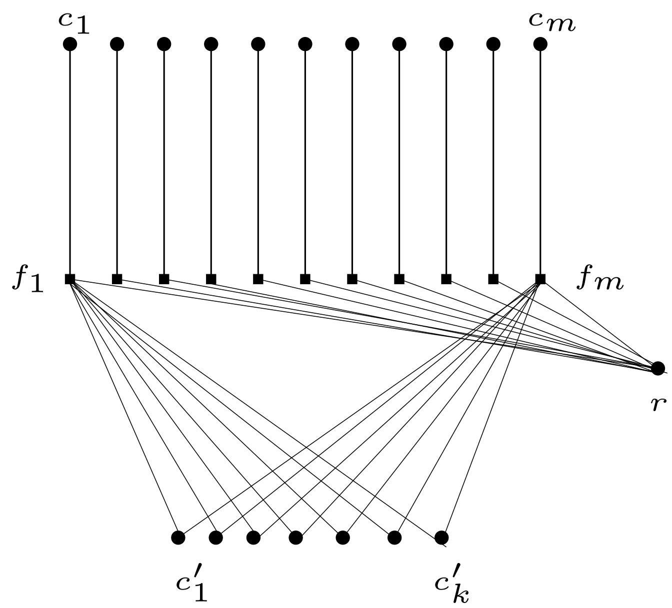
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $\mathcal{B} := \{\{a_1, \ldots, a_k\} : a_i \in A_i,\ i = 1, \ldots, k\}$.
- For each $B \in \mathcal{B}$: vertex $f_B$ with distance 1 to all vertices in $B$.



- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

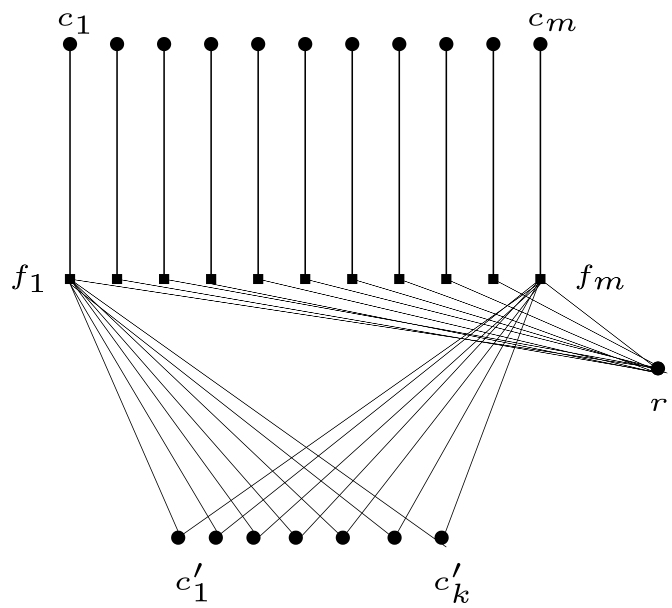
Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues

# Lower Bound for Steiner Trees

- $\mathcal{B} := \{\{a_1, \ldots, a_k\} : a_i \in A_i,\ i = 1, \ldots, k\}$.
- For each $B \in \mathcal{B}$: vertex $f_B$ with distance 1 to all vertices in $B$.
- $f_B$ is connected to the root $r$, with edges of length 3.



# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

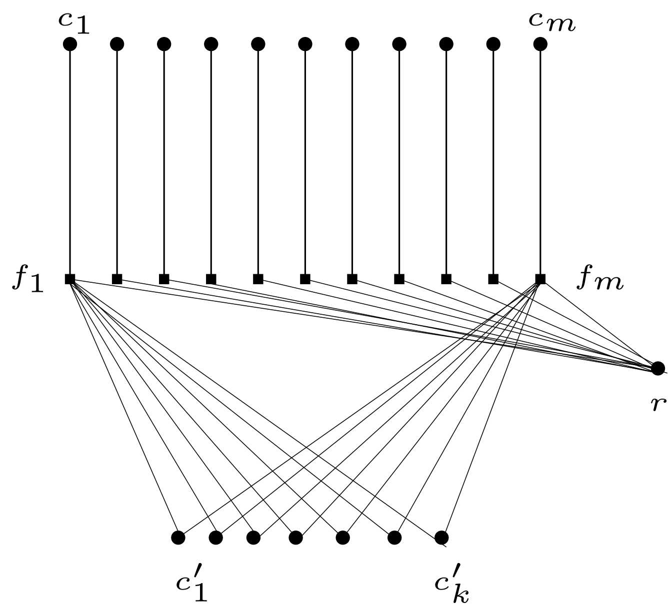
Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $\mathcal{B} := \{\{a_1, \ldots, a_k\} : a_i \in A_i,\ i = 1, \ldots, k\}$.
- For each $B \in \mathcal{B}$: vertex $f_B$ with distance 1 to all vertices in $B$.
- $f_B$ is connected to the root $r$, with edges of length 3.
- $f_B$ has distance 3 to vertices not in $B$.



- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

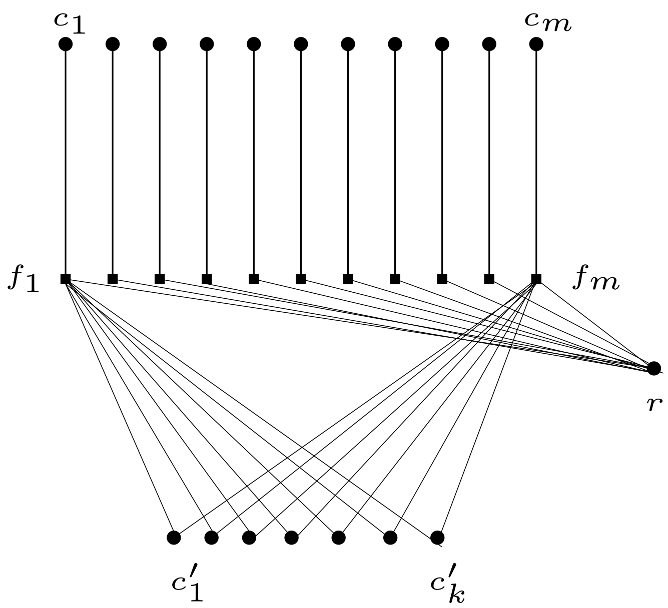
Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


# Lower Bound for Steiner Trees

- $\mathcal{B} := \{\{a_1, \ldots, a_k\} : a_i \in A_i,\ i = 1, \ldots, k\}$.
- For each $B \in \mathcal{B}$: vertex $f_B$ with distance 1 to all vertices in $B$.
- $f_B$ is connected to the root $r$, with edges of length 3.
- $f_B$ has distance 3 to vertices not in $B$.
- For each $c_l$, $l = 1, \ldots, m$, $c(\{a_1, \ldots, a_{i-1}, c_l, a_{i+1}, a_k\}) = k + 3$ implies $\xi(c_l) = \frac{k+3}{k}$



# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

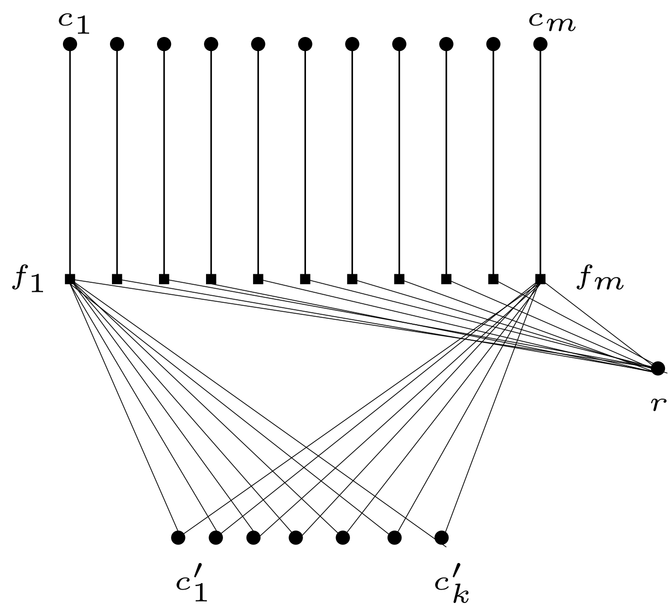
Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $\mathcal{B} := \{\{a_1, \ldots, a_k\} : a_i \in A_i,\ i = 1, \ldots, k\}$.
- For each $B \in \mathcal{B}$: vertex $f_B$ with distance 1 to all vertices in $B$.
- $f_B$ is connected to the root $r$, with edges of length 3.
- $f_B$ has distance 3 to vertices not in $B$.
- For each $c_l$, $l = 1, \ldots, m$, $c(\{a_1, \ldots, a_{i-1}, c_l, a_{i+1}, a_k\}) = k + 3$ implies $\xi(c_l) = \frac{k+3}{k}$
- Total cost share:

$$\sum_{c \in A_i} \xi(c) + \sum_{j \neq i} \xi(a_j) \leq m \times \frac{k+3}{k} + k + 2$$



# Lower Bound for Steiner Trees


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

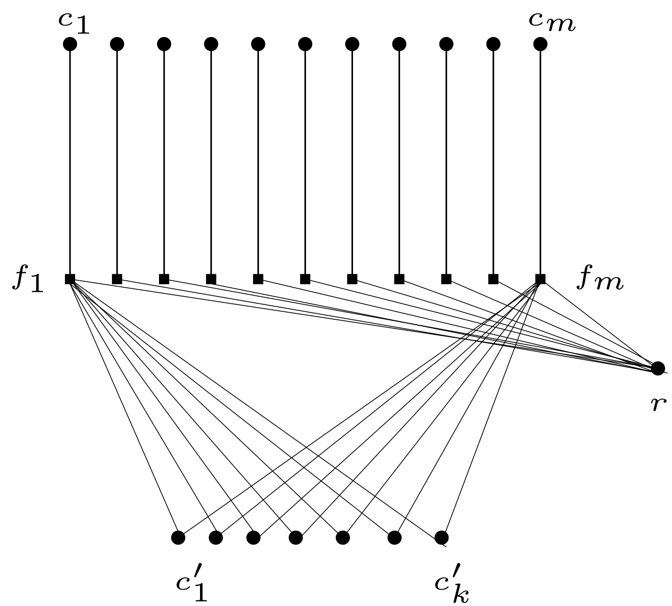
Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- $\mathcal{B} := \{\{a_1, \ldots, a_k\} : a_i \in A_i,\ i = 1, \ldots, k\}$.
- For each $B \in \mathcal{B}$: vertex $f_B$ with distance 1 to all vertices in $B$.
- $f_B$ is connected to the root $r$, with edges of length 3.
- $f_B$ has distance 3 to vertices not in $B$.
- For each $c_l$, $l = 1, \ldots, m$, $c(\{a_1, \ldots, a_{i-1}, c_l, a_{i+1}, a_k\}) = k + 3$ implies $\xi(c_l) = \frac{k+3}{k}$
- Total cost share:

$$\sum_{c \in A_i} \xi(c) + \sum_{j \neq i} \xi(a_j) \leq m \times \frac{k+3}{k} + k + 2$$




- $\texttt{opt} \geq 2m + k + 3$

# Cost-Sharing Mechanisms


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


Objectives:

- **Strategyproofness:** Dominant strategy for each user is to bid true utility.
- **Group-Strategyproofness:** Same holds even if users collaborate. No side payments between users.
- **Cost Recovery or Budget Balance:** $\sum_{j \in Q} p_j \geq c(Q)$.
- **Competitiveness:** $\sum_{j \in Q} p_j \leq \texttt{opt}_Q$.
- $\alpha$**-Efficiency approximate maximum social welfare:**

$$u(Q) - c(Q) \geq \frac{1}{\alpha} \cdot \max_{S \subseteq U} [u(S) - C(S)], \quad \alpha \geq 1$$

No mechanism can achieve (approximate) budget balance, truthfullness and efficiency [Feigenbaum et al. '01]

- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

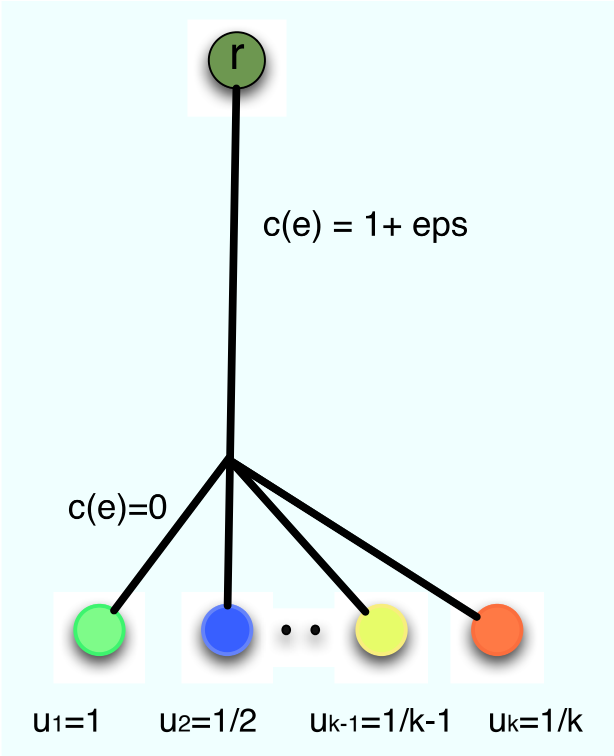
Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


# Limitations of Moulin mechanisms

- Moulin mechanism ends with dropping all players
- (1+$\epsilon$)-budget balance solution achieves $H(k)$ social welfare.



- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


# Objectives

1. $\beta$-budget balance: approximate total cost

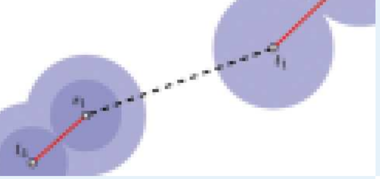
$$\frac{1}{\beta} c(Q) \leq p(Q) \leq \texttt{opt}_Q, \quad \beta \geq 1$$

2. Group-strategyproofness: bidding truthfully $b_i = u_i$ is a dominant strategy for every user $i \in U$, even if users cooperate
3. $\alpha$-approximate: approximate minimum social cost

$$\Pi(Q) \leq \alpha \cdot \min_{S \subseteq U} \Pi(S), \quad \alpha \geq 1$$

where $\Pi(S) := u(U \setminus S) + C(S)$

[Roughgarden and Sundararajan '06]



# Known Results - Social Cost


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds

- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


| **Authors** | **Problem** | $\beta$ | $\alpha$ |
|---|---|---|---|
| [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, Könemann, Leonardi, Ravi, Schäfer '07] | prize-collecting Steiner forest | $3$ | $\Theta(\log^2 n)$ |
| [Goyal, Gupta, Leonardi, Ravi '07] | 2-stage Stochastic Steiner Tree | $O(1)$ | $\Theta(\log^2 n)$ |

# Summary


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds

- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- Introduced cost-sharing mechanisms for network design problems
- Presented cross-monotonic cost-sharing methods for Steiner forests and facility location.
- Presented a lower bounds on budget balance for cross-monotonic cost-sharing methods.
- Presenteed bounds on efficiency loss.

# Open Issues


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- Give better and cross-monotonic cost-sharing methods.

# Open Issues

● Algorithm and Mechanism Design
● Approximation and Mechanism Design
● Approximation and Mechanism Design
● Approximation and Mechanism Design
● Overview of the tutorial
● Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
● Lower Bound for Cross-Monotonicity
● Lower Bound for Steiner Trees
● Limitations of Moulin mechanisms
● Objectives
● Known Results - Social Cost
● Summary
● Open Issues

- Give better and cross-monotonic cost-sharing methods.
- Characterize classes of problems yielding mechanisms with good cost recovery.

# Open Issues


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds

- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- Give better and cross-monotonic cost-sharing methods.
- Characterize classes of problems yielding mechanisms with good cost recovery.
- A more satisfactory definition of group-strategyproofness.

# Open Issues


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- Give better and cross-monotonic cost-sharing methods.
- Characterize classes of problems yielding mechanisms with good cost recovery.
- A more satisfactory definition of group-strategyproofness.
- Achieve better efficiency loss with randomized mechanisms?

# Open Issues


- Algorithm and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Approximation and Mechanism Design
- Overview of the tutorial
- Talk Outline

Cost-Sharing Mechanisms

Group-strategyproof for Facility location

Steiner Forests

Steiner Forest CS-Mechanism

Lower Bounds
- Lower Bound for Cross-Monotonicity
- Lower Bound for Steiner Trees
- Limitations of Moulin mechanisms
- Objectives
- Known Results - Social Cost
- Summary
- Open Issues


- Give better and cross-monotonic cost-sharing methods.
- Characterize classes of problems yielding mechanisms with good cost recovery.
- A more satisfactory definition of group-strategyproofness.
- Achieve better efficiency loss with randomized mechanisms?
- Players with 0 utility seem to play a crucial role for manipulation. Can this be avoided by using randomization?