Network Design Games

Éva Tardos
Cornell University

Interacting selfish users

Users with diverse economic interests
- Routers on the Internet
- Companies
- Servers

Selfishness:
- Parties will deviate from their protocol if it is in their interest
- But are not malicious

Study resulting issues:

Algorithmic game theory

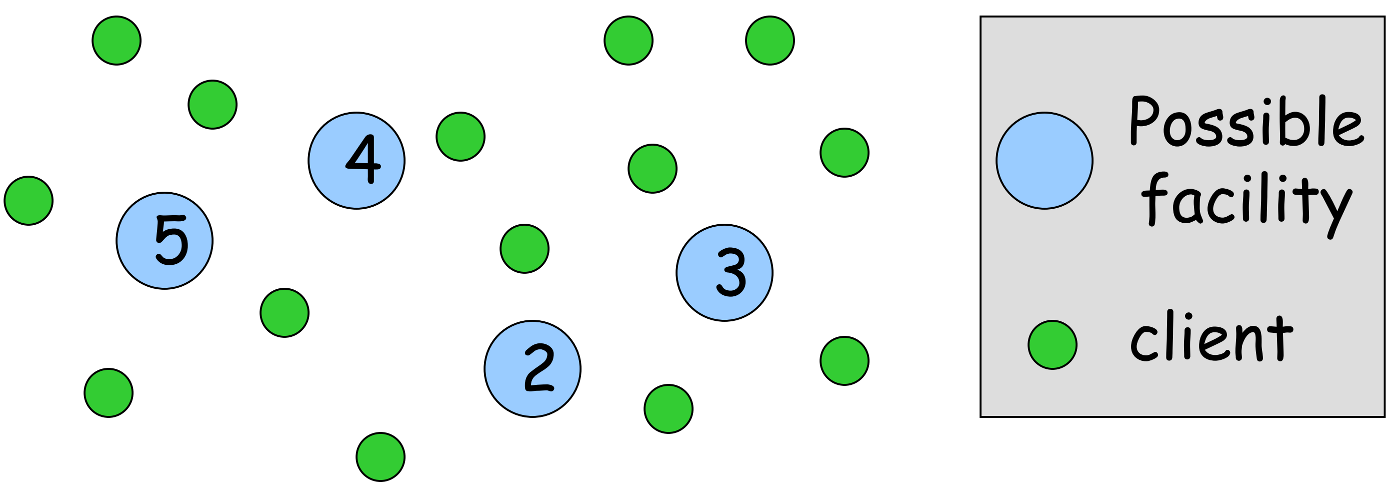
algorithms + game theory

Facility Location

F is a set of facilities (servers).
D is a set of clients.

dist is the distance function.

Facility i in F has cost f_i



Problem Statement

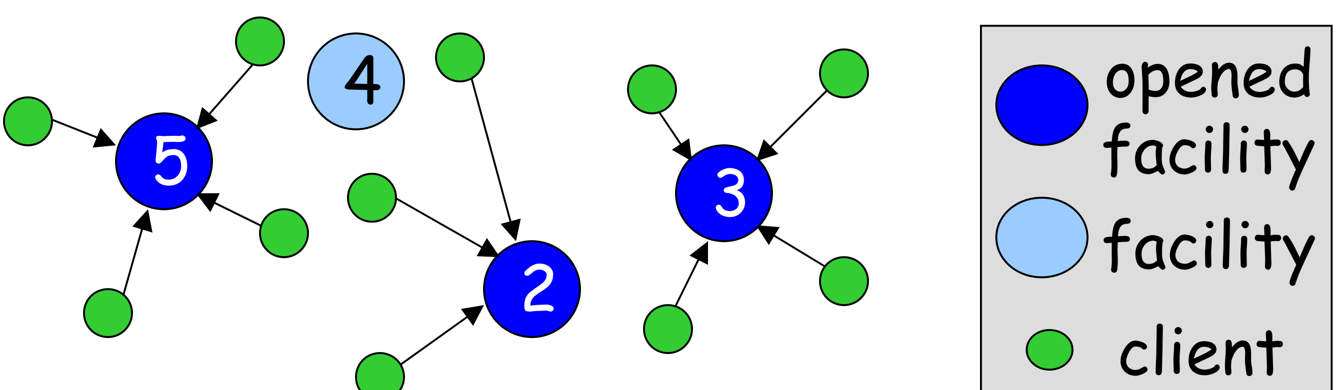
We need to:

- Pick a set S of facilities to open.
- Assign every client to an open facility (a facility in S).

Goal: Minimize cost:

$$\text{cost}(S) + \Sigma_p \text{ dist}(p,S).$$

Facility cost + service cost



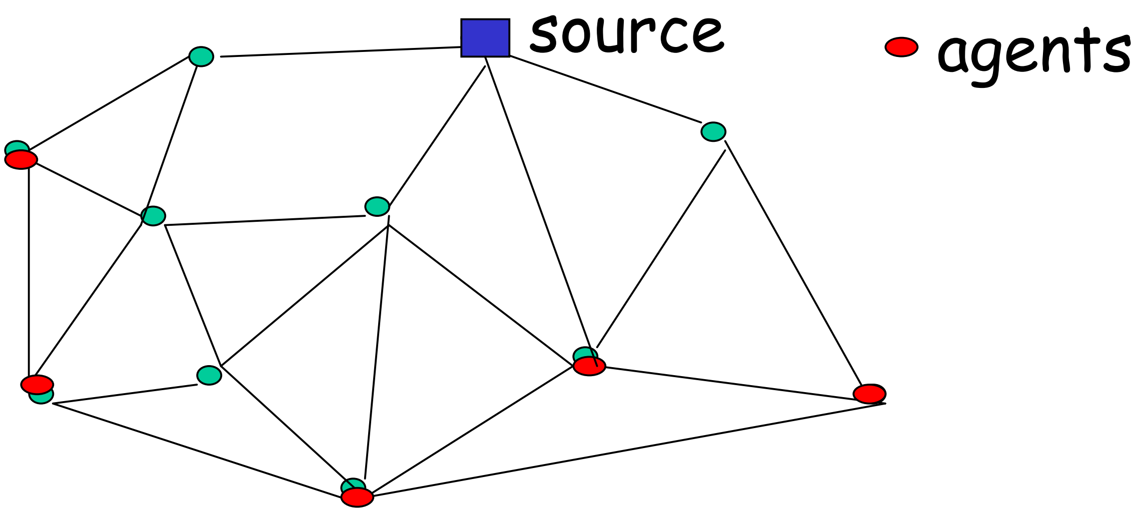
What is known?
algorithms

NP-complete, but

all techniques can be used to get close to optimal solutions:

- Clever greedy [Jain, Mahdian, Saberi '02]
- Local search [starting with Korupolu, Plaxton, and Rajaraman '98], can handle capacities
- LP and rounding: [starting with Shmoys, T, Aardal '97]
- primal-dual [starting with Jain-Vazirani'99]

What is network design I
Steiner Tree

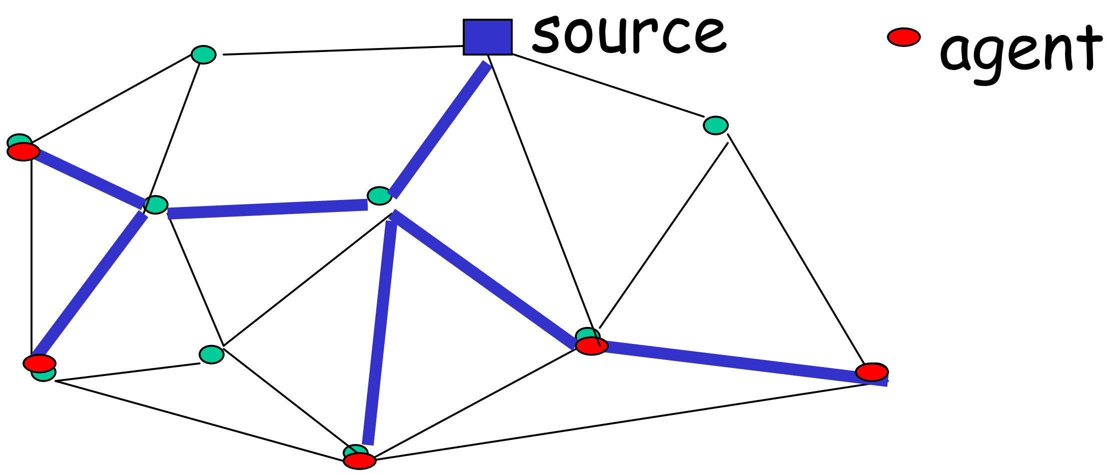


Network of possible edges with edge costs $c_e \geq 0$

Agents want to connect subsets of nodes.

- E.g., agent wants to connect his node to a common source.

What is network design I
Steiner Tree

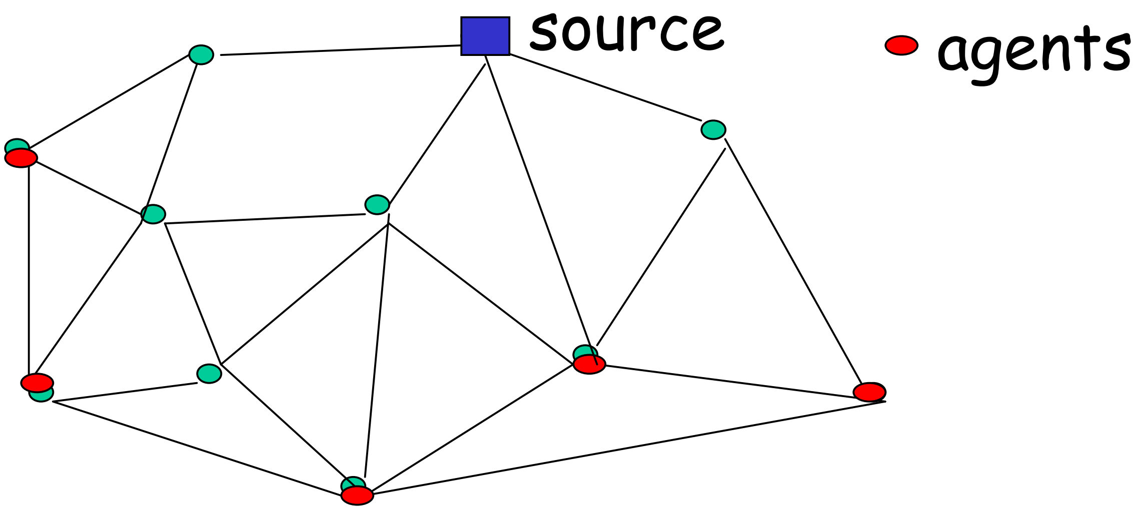


A possible design.

Agent's goal:

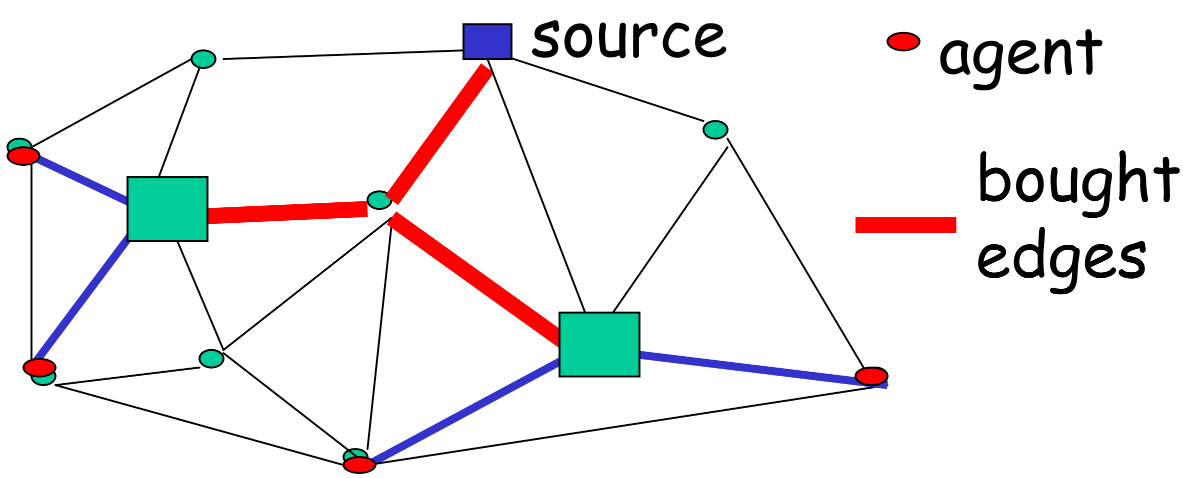
- Connect his terminal to source
- Pay as little as possible

What is network design II
Rent-on-Buy
(connected facility location)



- Agents want to connect to a common source.
- Network of possible edges
  - Costs c_e to rent and
  - Costs Mc_e to buy
- Rented edge: charged by user
- Bought edge: fixed cost

What is network design II
Rent-on-Buy
(connected facility location)



A possible design.

Agent's goal:

- Connect his terminal to source
- Pay as little as possible

What is known?
algorithms

NP-complete, but good approximations known

- combining Steiner tree and facility location [Karger-Minkoff '00], [Guha-Munagala-Meyerson '00],
- primal-dual [Swamy-Kumar '02]
- randomized [Gupta, Kumar, Roughgarden'03]

Best approximation 3.55

What is a game?

Cost-sharing:

- Central algorithm selects network.
- Agents have to share the cost (or contribute to)

Nash games:

- Agents decide on how much to contribute to each edge
- Study Nash equilibrium
  - Stable strategies

Cost-sharing vs. Nash?

Cost-sharing:

- Central algorithm selects network.
- Cooperative game: goal to make it group strategyproof:
  - no group of users want to get together to “cheat” the algorithm

Nash games:

  - Stable strategy for individual agents
  - Agents only observe others, do not cooperate
- Central algorithm only needed to help payers reach equilibrium [Greenwald, Friedman and Shenker]

How to evaluate a game?

Nash games: compare cost of Nash equilibrium to optimum cost

- Minimize cost in Steiner tree (and generalized SteinerTree) [Anshelevich-Dasgupta-T-Wexler 2003]
- Maximize profit for facility location [Vetta 2002]
- Routing to minimize delay [Roughgarden-Tardos 2000]

This talk: Approximate cost-sharing

Cost sharing Games

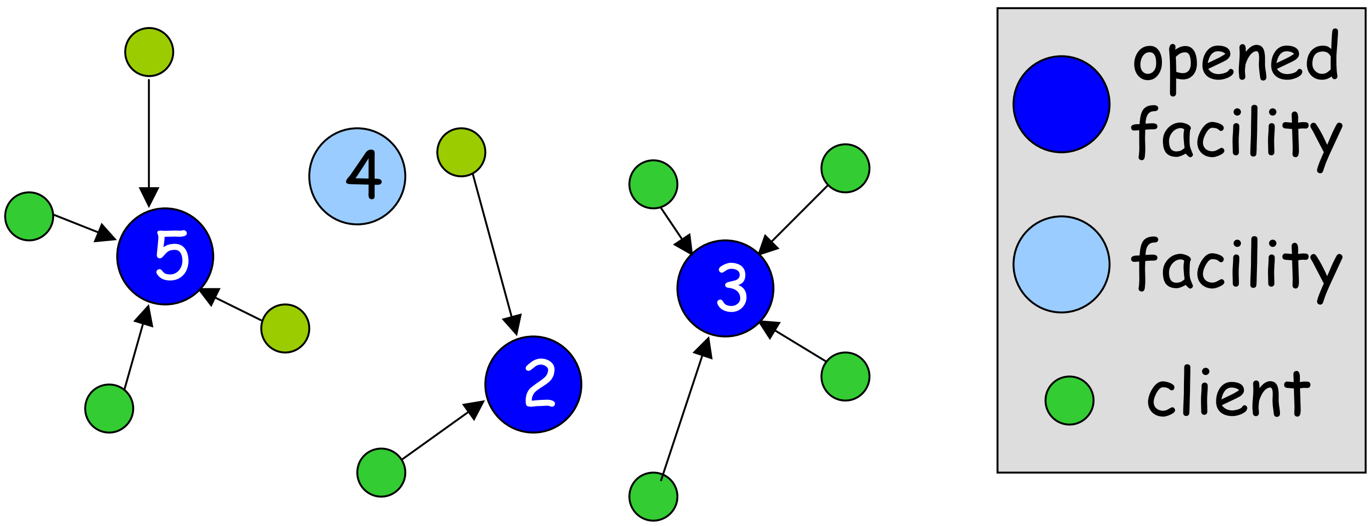
Core/fairness: Shapley'67, Bondareva'63 Chardaire'98; Goemans-Skutella'00

How to deal with limited individual utility?

Population monotone ⇒ strategy proof [Moulin-Shenker'99]

- Steiner Tree: [Kent & Skorin-Kapov '96], [Jain & Vazirani'01]
- Facility Location [Devanur, Vazirani & Mihail EC'03, Pál & T '03]
- Rent-or-Buy [Pál & T '03]

Cost Sharing



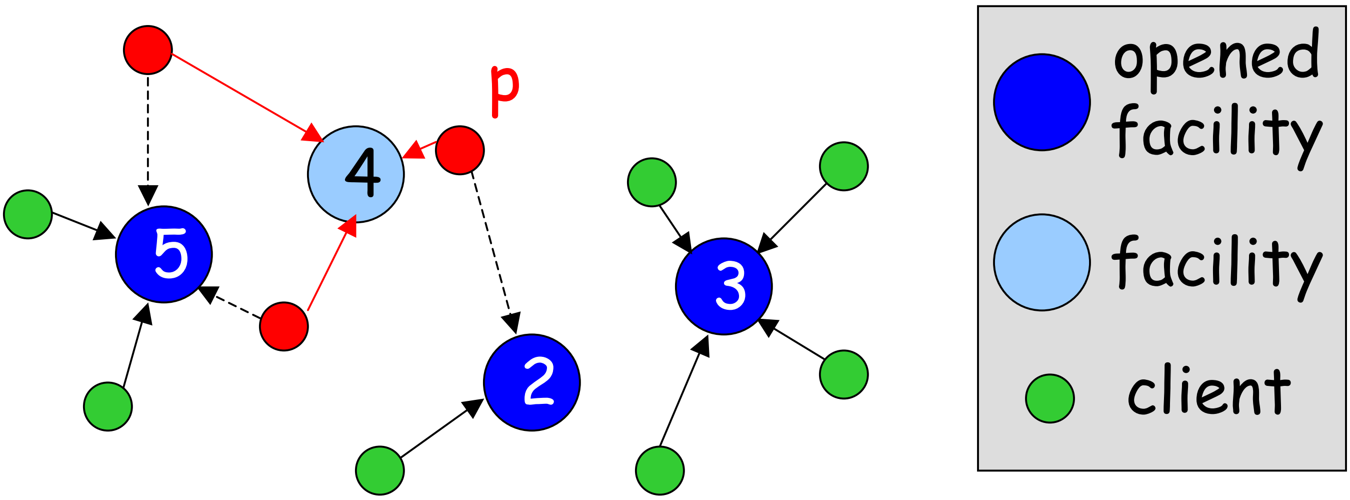
- Build the a solution for S at cost c(S)
  - by a centralized algorithm

- users only share the cost:
  - cost share of agent p is ξ_p

- Budget-balanced if

$$\Sigma_{p \in S}\, \xi_p = c(S)$$

What is “fair cost” share?

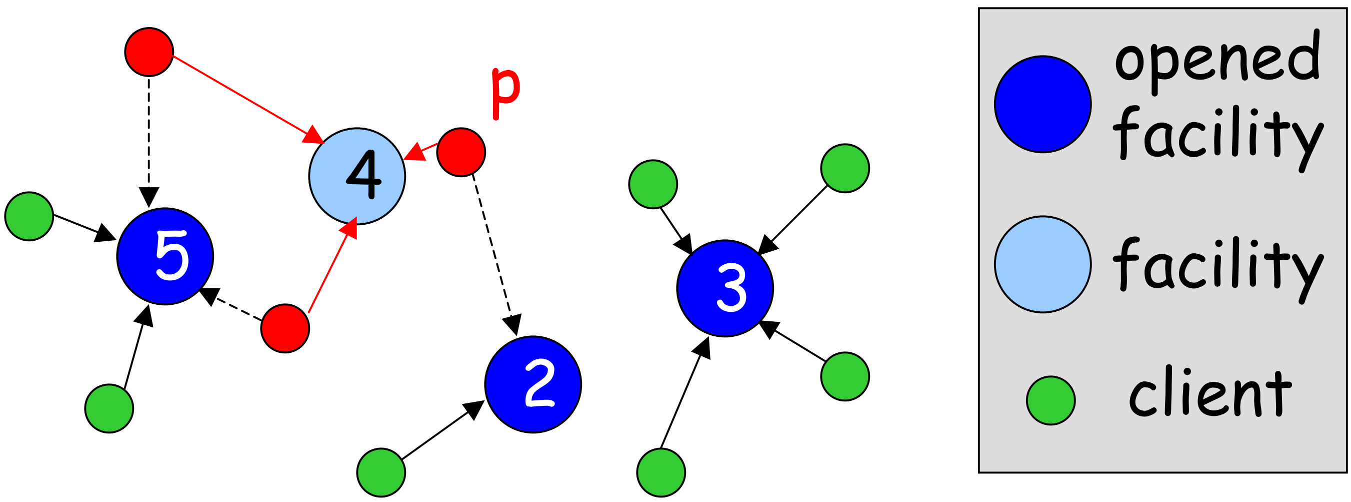


Unfair to charge a group of users more than it costs to serve them alone

Fair: if no group A of users is overcharged:

- $\Sigma_{p \in A}\ \xi_p \leq c(A)$ for all $p \in A \subseteq S$

Core: fair cost-sharing



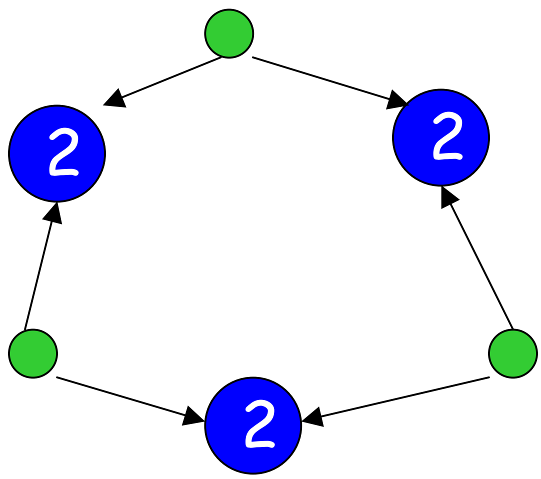
Core =

Budget balanced cost-sharing where no subset A is overcharged:

- $\Sigma_{p \in A} \xi_p \leq c(A)$ for all $A \subseteq S$

$\Rightarrow$ No subset of clients wants to deviate

Is there fair & budget balanced cost-sharing?



No! no pair of clients can be charged more than 2, so max total 3,

but it costs 4 to build a solution…

But there is fair cost-sharing that recovers a large fraction of the budget

Outline

Algorithmic game theory

- Nash equilibrium games vs.
- Cost-sharing

Cost sharing games

- Core/fairness Shapley'67, Bondareva'63 Chardaire'98; Goemans-Skutella'00
- Dealing with limited individual utility: Population monotone/strategy proof [Devanur, Mihail, Vazirani EC'03], [Pál-T '03]
    - Facility location
    - Rent-or-Buy network design

What is known?

Core closely related to linear programming dual: Shapley'67, Bondareva'63

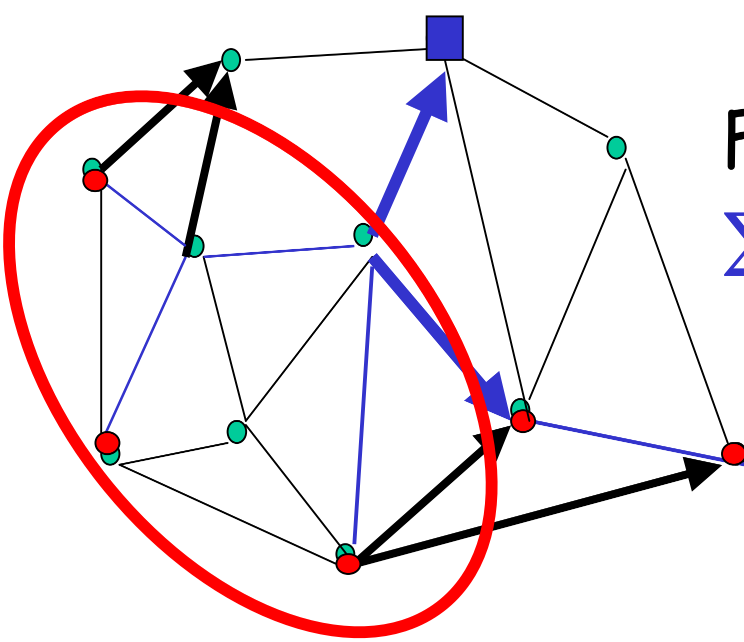
facility location game: **[Chardaire'98; Goemans-Skutella'00]**

Theorem Budget balance gap = LP integrality gap!

Core and linear programming

Desires of agents can be expressed by linear inequalities:

- Use variables x_e for edge e



For subset S:

$\Sigma_{e \text{ leaving } S}\, x_e \geq 1$

associated with the group of agents in S

Linear Programming

Variables x_e 0 or 1

Constraints $\Sigma_{e \text{ leaving } A} x_e \geq 1$
for all subsets A

Goal minimize cost $\Sigma_e c_e x_e$

Linear Programming

Variables x_e 0 or 1

Constraints $\Sigma_{e \text{ leaving } A}\, x_e \geq 1$ for all subsets S

Goal minimize cost $\Sigma_e\, c_e\, x_e$

LP dual variables: $y_A \geq 0$

a cost-share associated with each inequality (each subset *A*)

- Constraint: no subset can be over-changed
- LP duality: minimum cost of LP solution = maximum total of cost-shares
  - $\Sigma_A\, y_A = \Sigma_e\, c_e\, x_e$

LP Duals ⇔ Cost-shares

Primal dual LP pair = for each requirement someone willing to pay to make it true

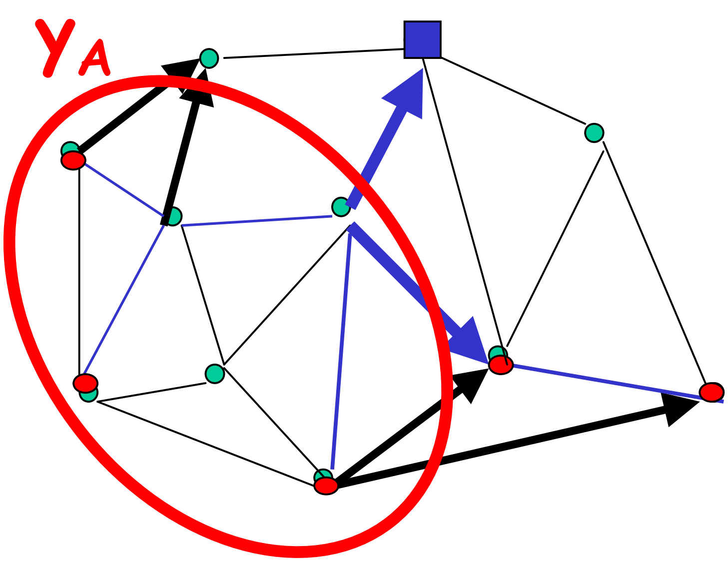
Cost-sharing: only players can have shares.

- Not all requirements are naturally associated with individual players.
- Real players need to share the cost.

Core and linear programming

Agents in A must share the cost y_A for all sets A

- Any way of sharing is in core!
  - any agent in A can be made to pay y_A



Bondareva '63
Shapley '67

≈Approximation algorithm

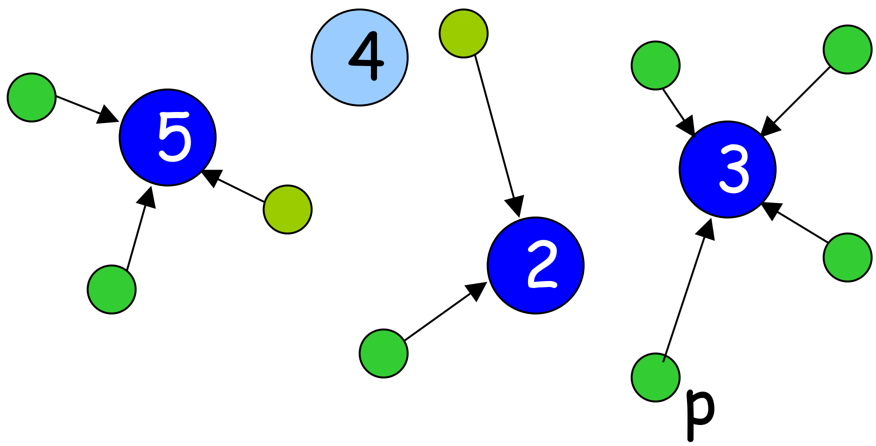
If no budget balanced core allocation exists, then we'll use approximate budget balancing:

- $\Sigma_p \xi_p \geq \kappa\, c(S)$

How small a fraction κ of the budget can be recover?

Corollary: Approximate core ≈ LP based approximation

Dealing with limited individual utility: should all clients get connected?



Assume each user has a utility u_p max amount they are willing to pay.

How to deal with utilities?

core: no subset is overcharged

- $\Sigma_{p \in A}\, \xi_p \leq c(A)$ for all $p \in A \subseteq S$

How do we deal with utilities?

- Want share ξ_p to be limited by utility $\xi_p \leq u_p$?

How to deal with utilities?

core: no subset is overcharged

- $\Sigma_{p \in A} \xi_p \leq c(A)$ for all $p \in A \subseteq S$

How do we deal with utilities?

- Want share ξ_p to be limited by utility $\xi_p \leq u_p$?
  - But but these utilities are known *only to the clients*
  - client p can lie about is utility u_p and decrease its share

Is core good enough?

Better idea:

Do not limit shares by utility:

Users who cannot afford their share will have to drop out!

Share of client p may depend on set S receiving service: $\xi(p,S)$

Is core good enough?

Better idea:

Do not limit shares by utility:

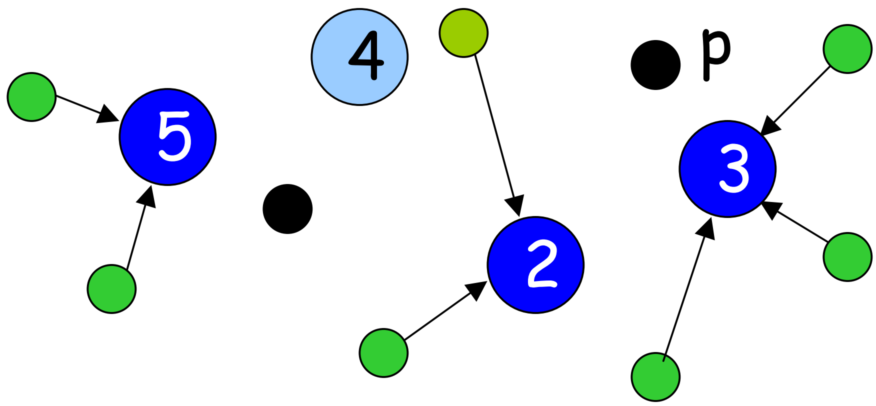
Users who cannot afford their share will have to drop out!

Share of client p may depend on set S receiving service: $\xi(p,S)$

Will agents drop out if $\xi(p,S) > u_p$?

- Is this method truthful?

From cost sharing game to mechanism design



Assume each user has a utility u_p max they are willing to pay.

mechanism design

- Algorithm must elicit utilities from users.
- Decide who to serve,
- Charge served users at most their utility.

Stronger desirable property

population monotone (cross-monotone):

Extra clients do not increase cost-shares.

- $\xi(p,A) \leq \xi(p,B)$ for all $p \in B \subset A$

Population monotone $\Rightarrow$ fair

$\Sigma_{p \in A} \xi(p,S) \leq \Sigma_{p \in A} \xi(p,A) \leq c(A)$
for all $p \in A \subseteq S$

From cost-sharing to mechanism design

Natural mechanism:

While not all users happy
- Assume all clients get served
- compute cost shares
- drop users who cannot afford their share

Moulin-Shenker 1992:

Mechanism group strategy proof, iff cost-sharing is population monotone
- Adding more users cannot decrease the cost

Group strategy proof = it is in no group of agents' interest to lie

Are cost shares population monotone ?

Linear program based cost-shares often not population monotone

- facility location cost-shares by **Chardaire'98; Goemans-Skutella'00** not monotone

Devanur, Vazirani, Mihail EC'03 strategy proof cost shares from primal-dual approximation algorithms for facility location

Not monotone, and not group strategy proof.

population monotone cost shares?

Single source tree-game: cross monotonic cost-sharing that recovers ½ of the cost: Kent and Skorin-Kapov'96 and Jain Vazirani'01

- by sharing LP dual variables y_S evenly among the agents in S

More on monotone cost-shares

[Pál-T '03] Primal-dual ⇒ monotone cost-shares

Theorem 3-approximately budget balanced cross-monotonic cost-sharing for facility location

⇒ group strategy proof cost-sharing mechanism, that is 3-approximately budget balanced.

Extension to rent-or-buy network design (combines features of facility location and Steiner tree), that is cross monotonic and approximately budget balanced.

Monotone cost-shares from any cost-shares?

Any cost-sharing can be made monotone:

$$\gamma(p,J) = \mathbf{min}_{p \in I \subset J}\, \xi(\mathbf{p},\mathbf{I})$$

- Obviously population monotonic
- But how unbalanced does this make the budget?
- And how do we compute it without trying all subsets?

Primal-Dual algorithm for monotone cost-sharing

What is Primal-Dual algorithm?

Uses ideas from cost sharing

- For each requirement, someone has to pay to make it true…

Sets cost-shares and solution together:

Here: Monotonic cost-sharing for a facility location game

Pál-T '03

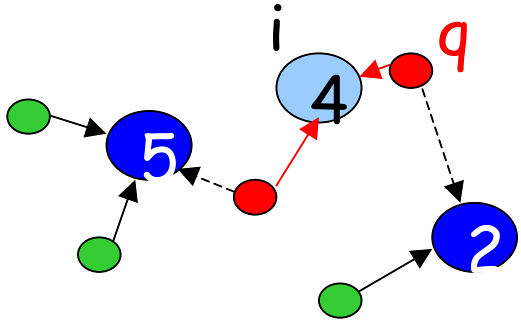
Cost-sharing problem for facility location

Client q has a fee α_q (cost-share)

Goal: collect as much as possible $\max \Sigma_q \alpha_q$

Fairness: Do no overcharge: for any subset A of clients and any possible facility i we must have

$$\Sigma_{q \in A} \underbrace{[\alpha_q - dist(i,q)]}_{\text{amount client q could contribute to building facility i.}} \leq f_i$$



Exact cost-sharing

- All clients connected to a facility
- Cost share α_q covers connection costs for each client q
- No set is over-changed
- Cost f_i of selecting a facility i is covered by clients using it

$\Rightarrow \Sigma_p \alpha_p = f(S) + \Sigma_p \, dist(p,S)$, and

both facilities and fees are optimal

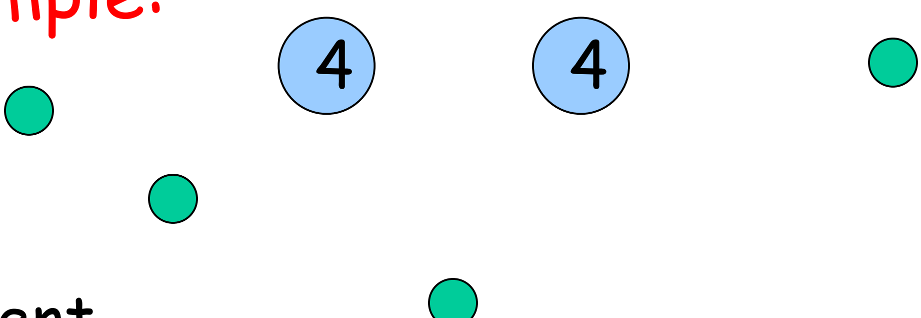
Idea of Approximate cost-sharing algorithm

Idea 1: [Jain Vazirani] each client starts unconnected, and with fee $\alpha_p = 0$

Then it starts raising what it is willing to pay to get connected

- Raise all shares evenly α

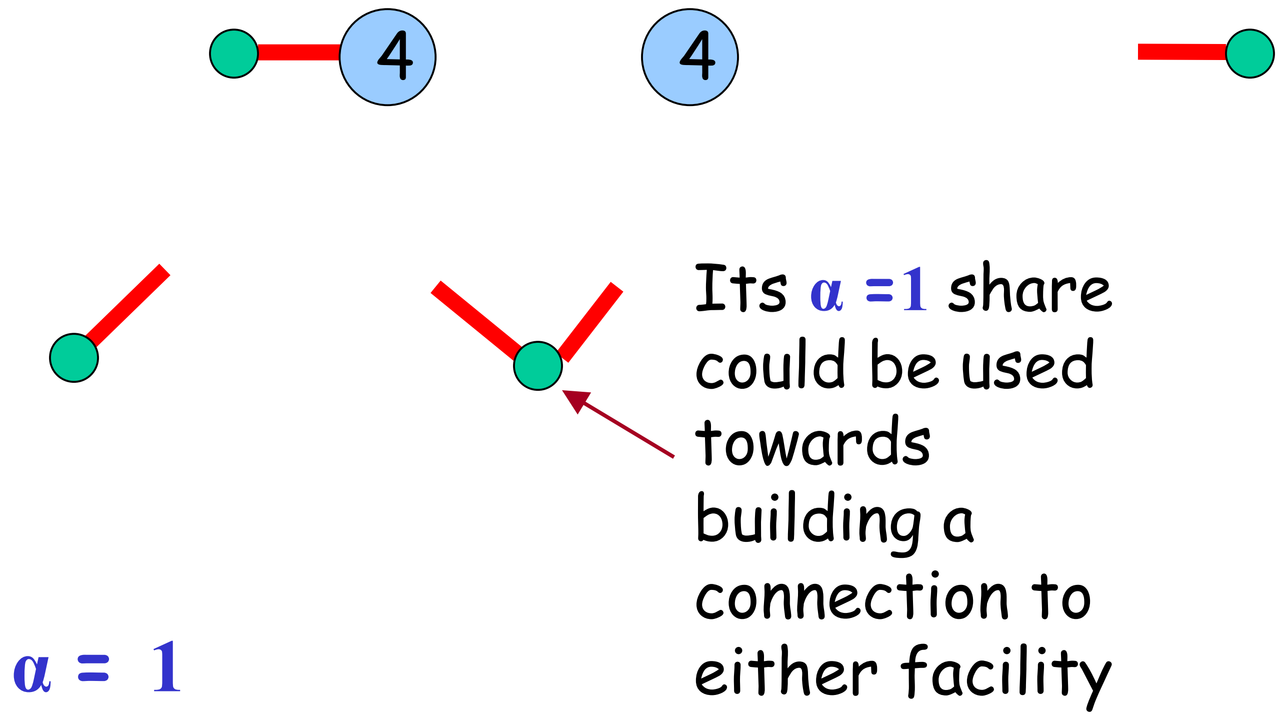
Example:



= client

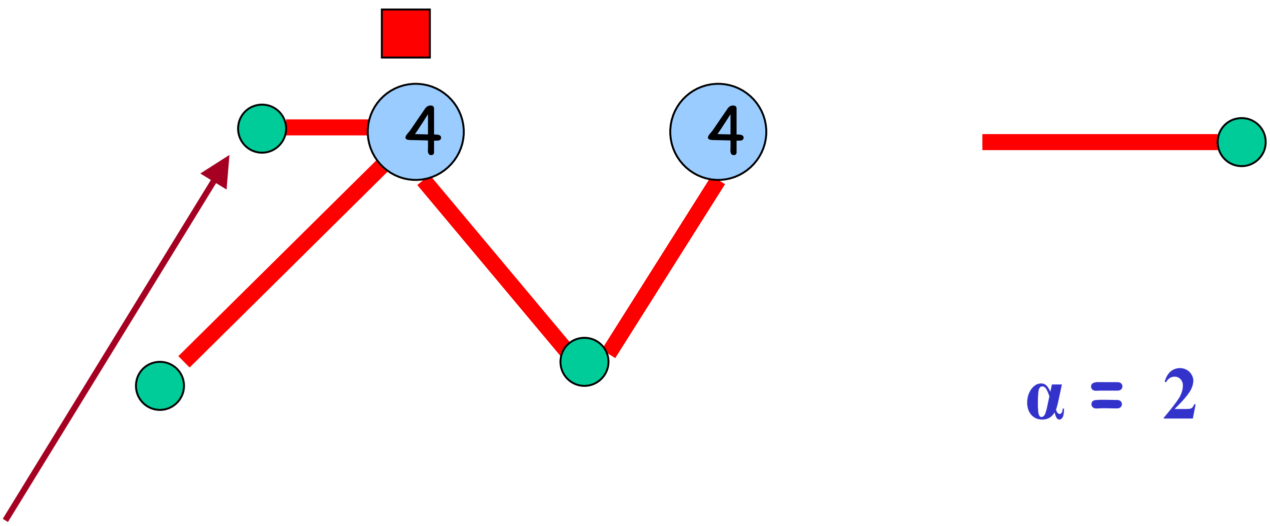
4 = possible facility with its cost

Primal-Dual Algorithm (1)



- Each client raises his fee α evenly what it is willing to pay

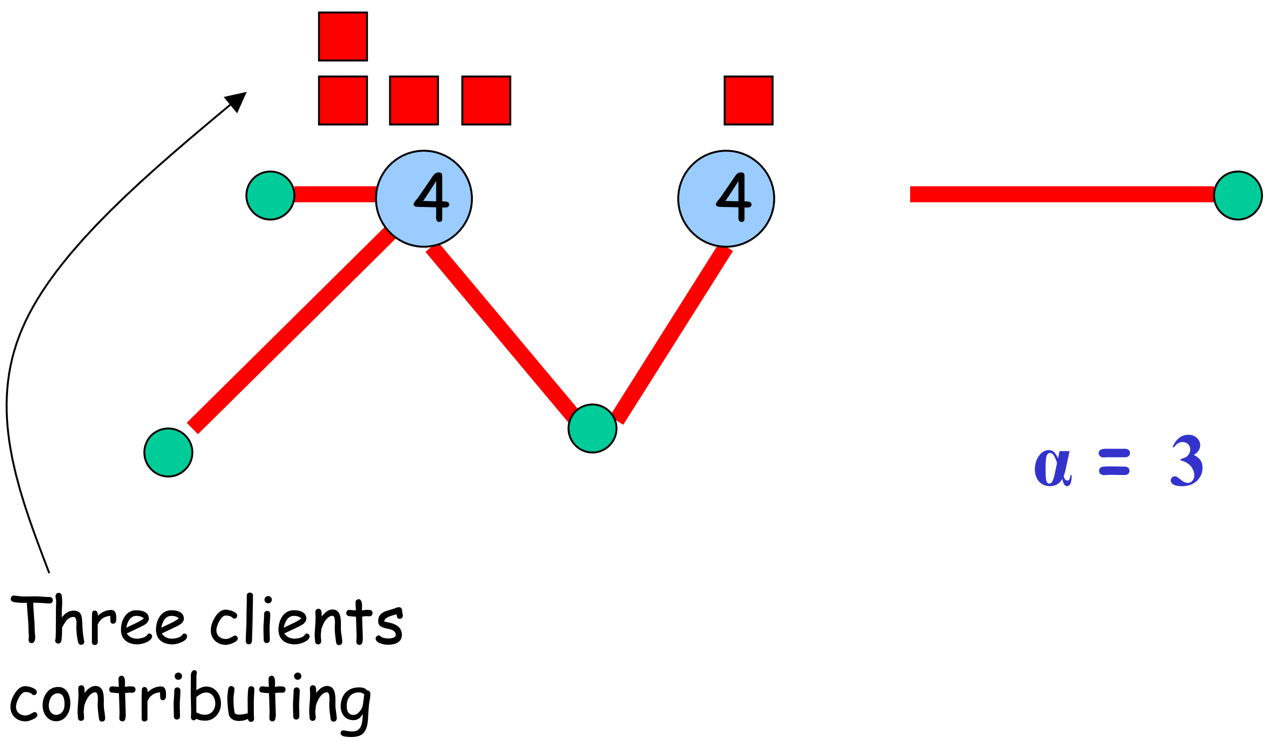
Primal-Dual Algorithm (2)



Starts contributing towards facility cost

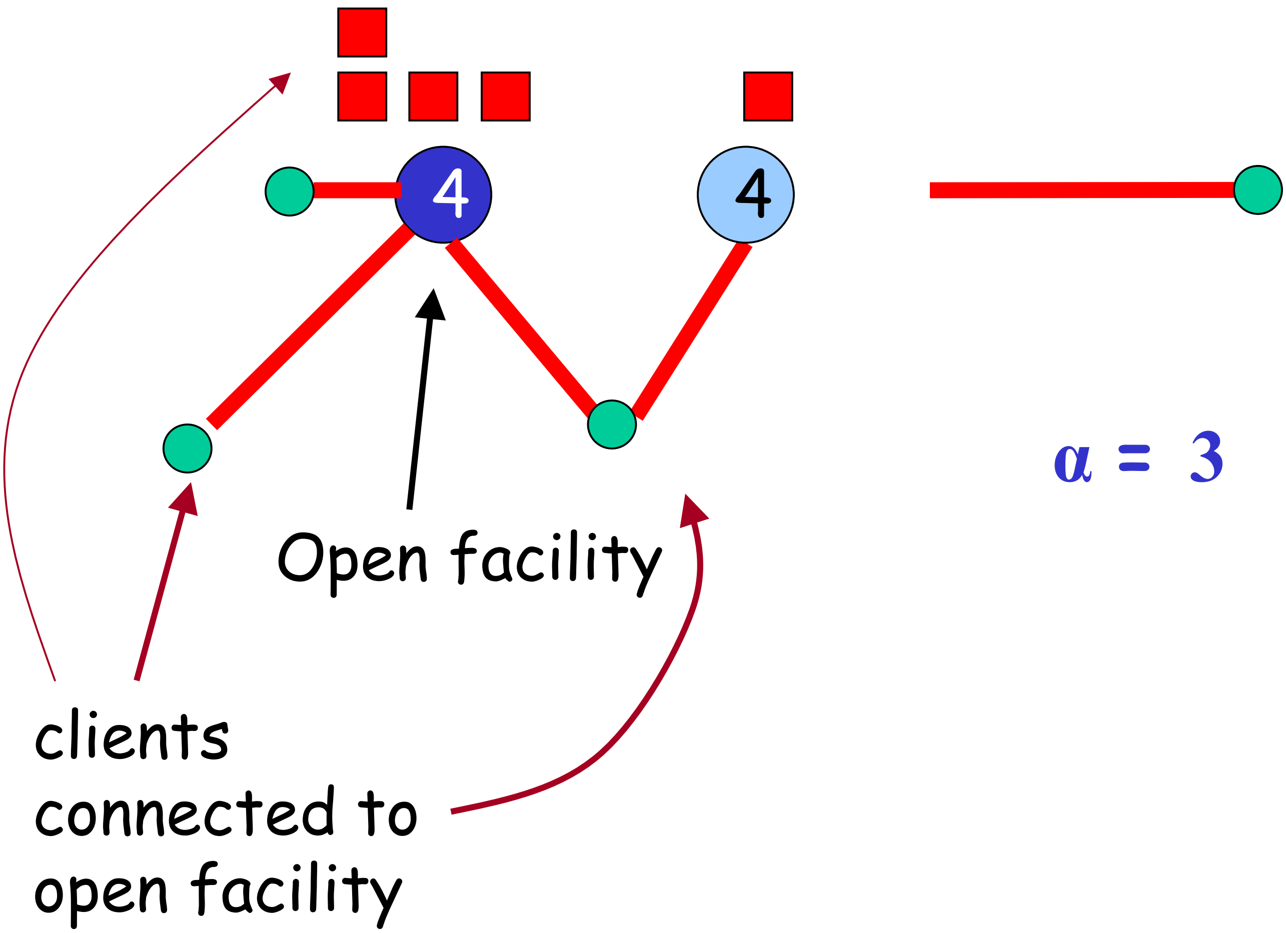
- Each client raises evenly what it is willing to pay

Primal-Dual Algorithm (3)



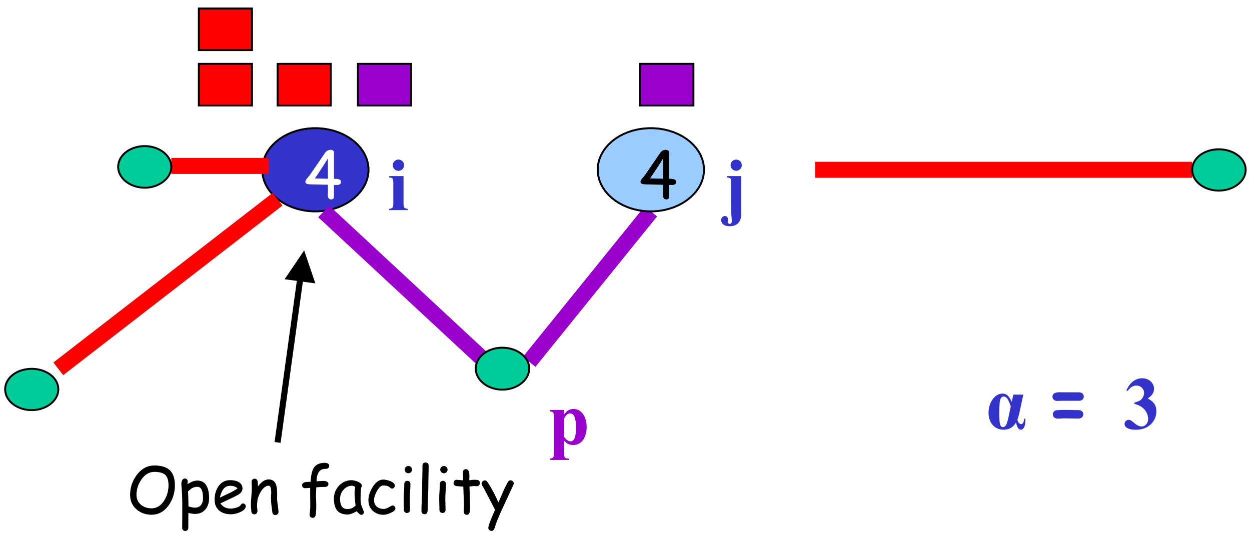
- Each client raises evenly what it is willing to pay

Primal-Dual Algorithm (4)



Open facility, when cost is covered by contributions

Primal-Dual Algorithm: Trouble



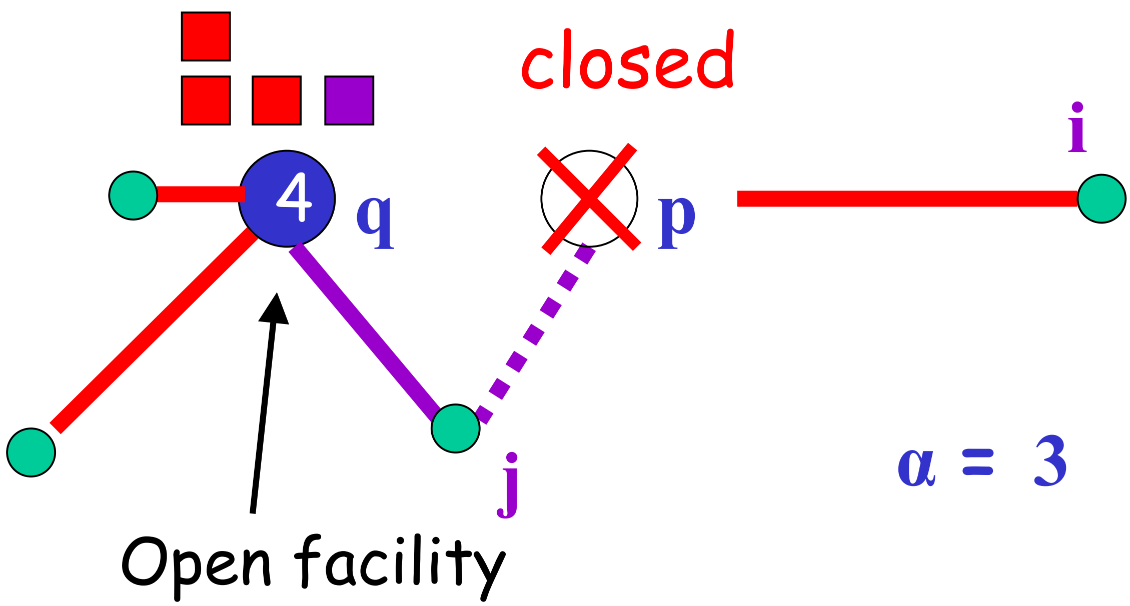
Trouble:

- one client p connected to facility i, but contributes to also to facility i;

Good news:

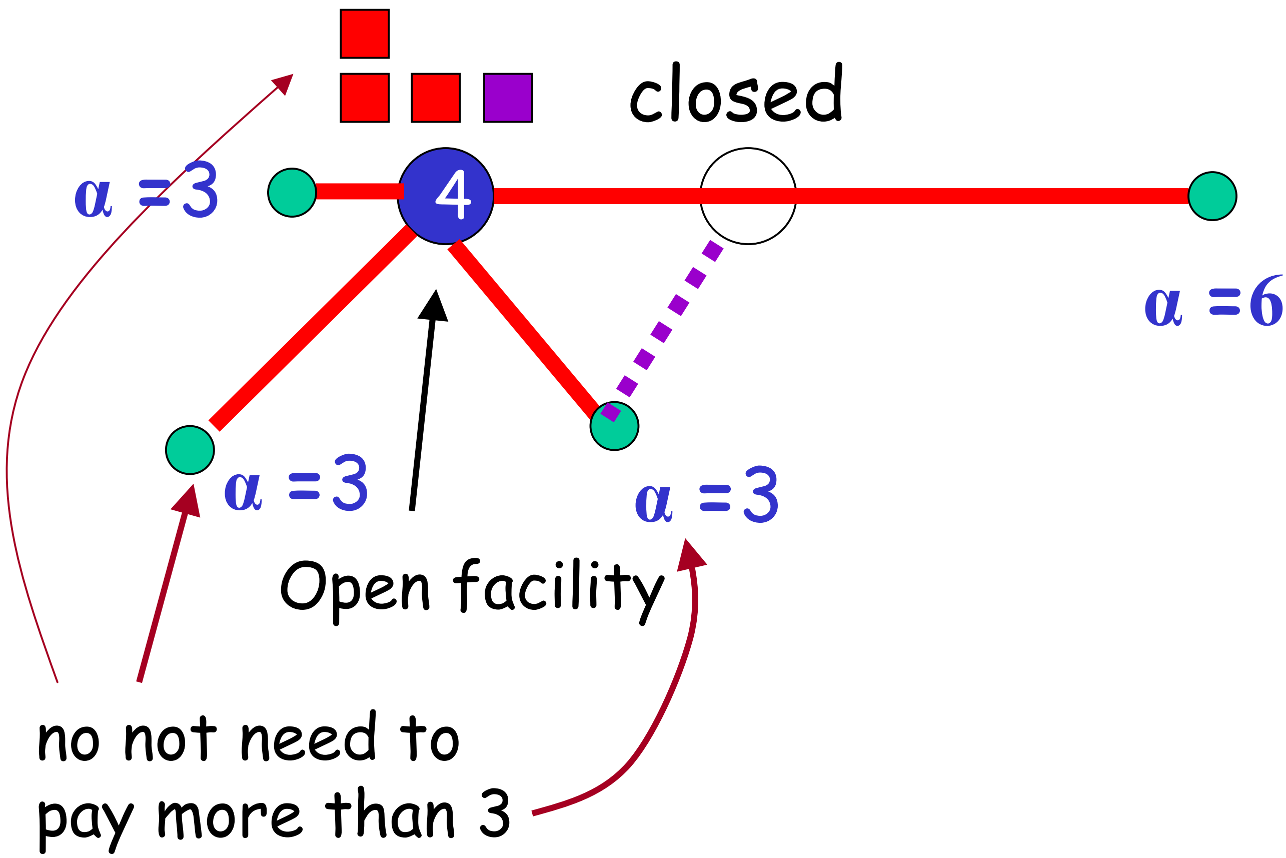
- Client p is close to both i and j ⇒ facilities i and j are at most **2α** from each other.

Primal-Dual Algorithm (5)



Idea close facility p and never open it!

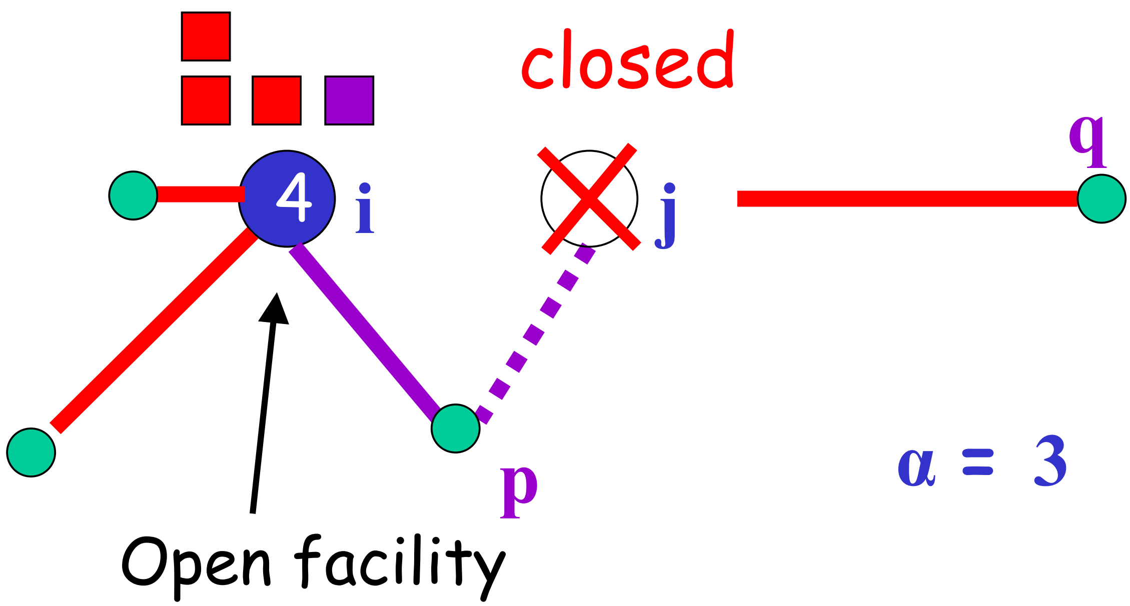
Primal-Dual Algorithm (6)



Not yet connected clients raise their fee evenly

Until all clients get connected

Are the cost-shares monotone?

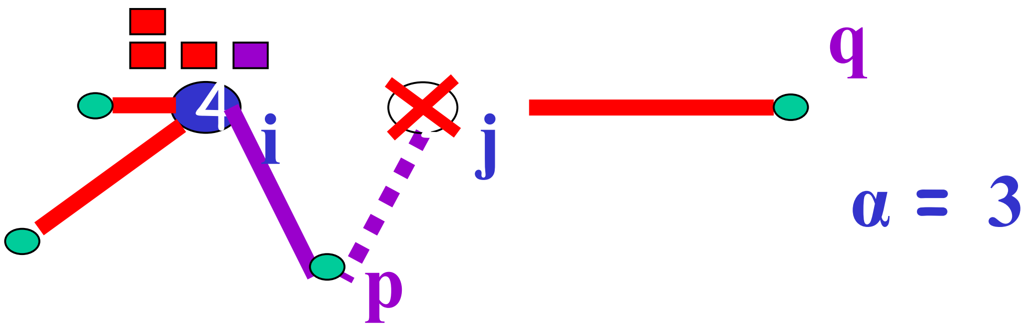


Idea close facility j and never open it!

But: events caused by p at i can make q pay more!!

⇒ Shares not monotone!

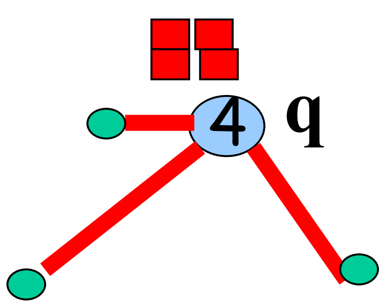
How to make monotone cost shares



Monotone share of client p=

- the earliest p can be connected:

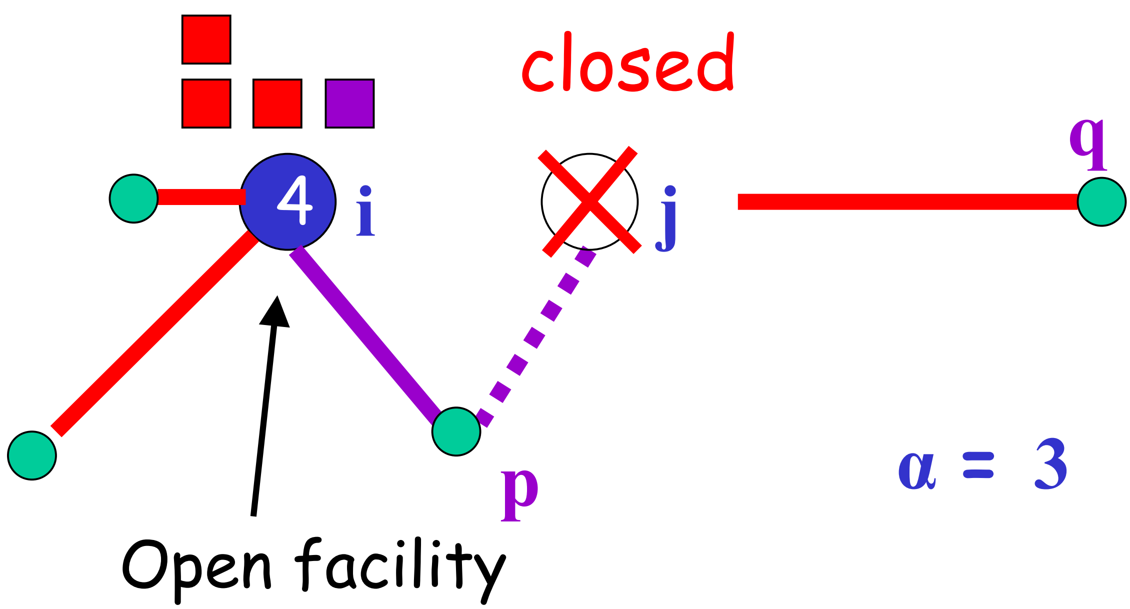
- $\xi(j,S)=\min_i \max(t(i), dist(p,i))$



t(i) = earliest facility i can open

Monotone Primal-Dual

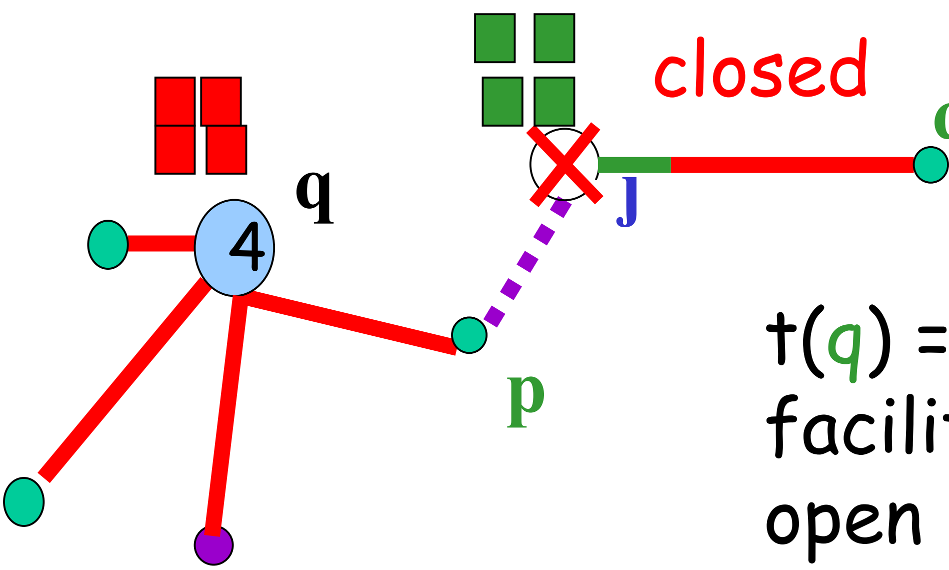
Keep closed facilities around to limit cost shares



How good is cost-share ξ_j ?

Costs ξ_j does not over-change

ξ is population monotone: extra clients can only decrease t(q)



t(q) = earliest facility q can open

t(q) monotone $\Rightarrow$ ξ monotone

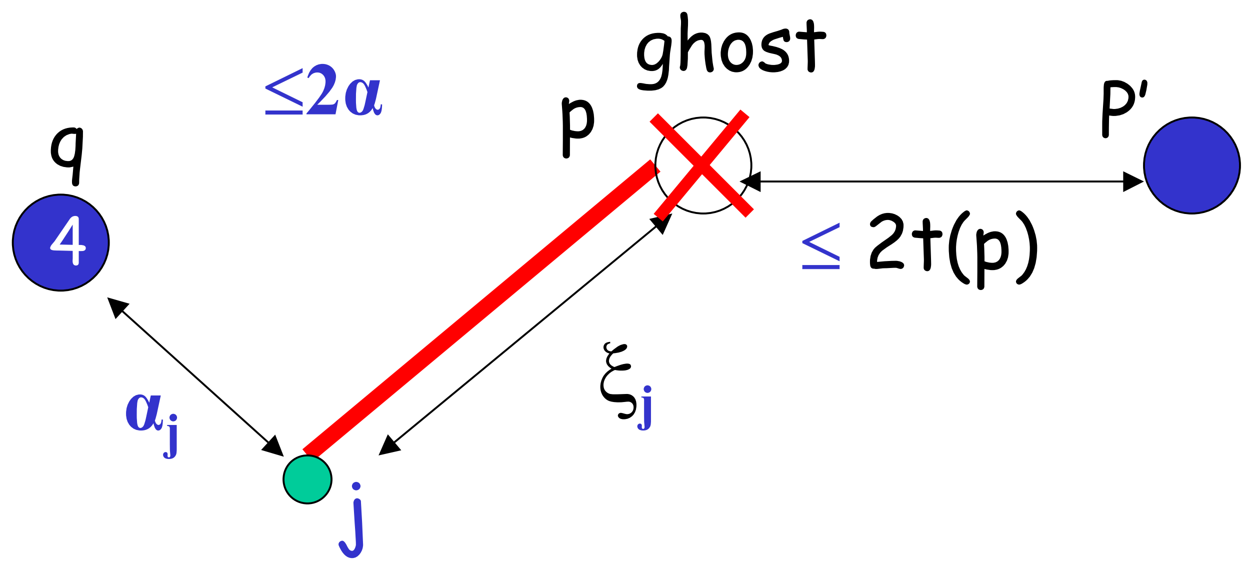
$$\xi_j = \min_q \max\,(t(q), \mathrm{dist}(j,q))$$

Feasibility + fairness ??

- √ All clients connected to a facility
- √ Cost share α_j covers connection costs for client j
- √ Cost α_j also covers cost of selected a facilities
- √ Costs ξ_j population monotone, and does not over-change

How much smaller is $\xi \leq \alpha$??

How much smaller is $\xi \leq \alpha$?



p defines ξ_j

p may be closed, but then there is an opened p' not too far:

dist(j,p') $\leq 3\,\xi_j$

How big can be α_j? $\Rightarrow \alpha_j \leq 3\,\xi_j$

Approximate budget balance = approximation quality

Theorem [Pál-T '03] 3-approximately budget balanced cross-monotonic cost-sharing for facility location

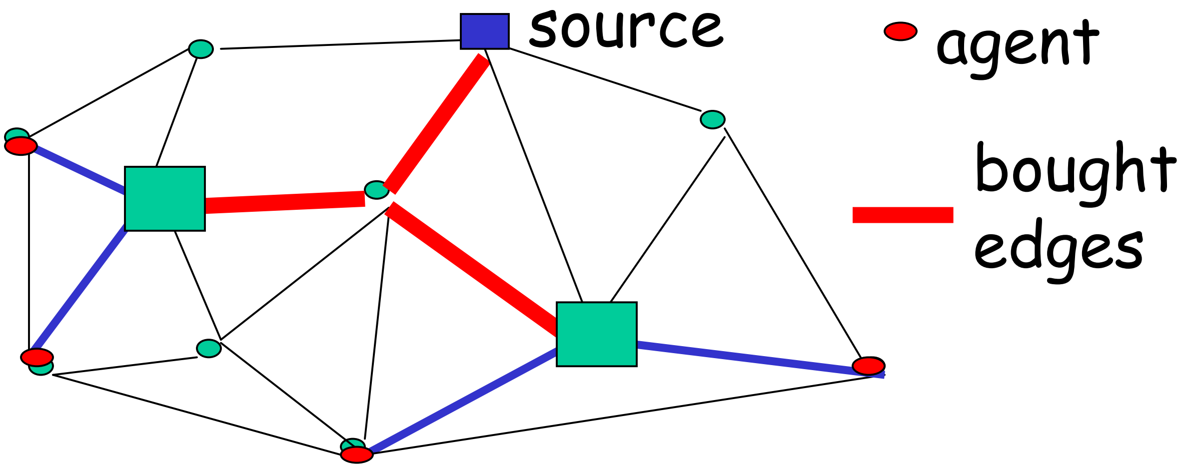
$\Rightarrow$ group strategy proof cost-sharing mechanism, that is 3-approximately budget balanced.

Approximate budget balance = approximation quality

[Pál-T '03] Extension to rent-or-buy network design (combines features of facility location and Steiner tree): cross monotonic and approximately budget balanced cost-sharing:

- Recovers $1/15^{th}$ fraction of the budget

rent-or-buy network design



Approximation Algorithms:

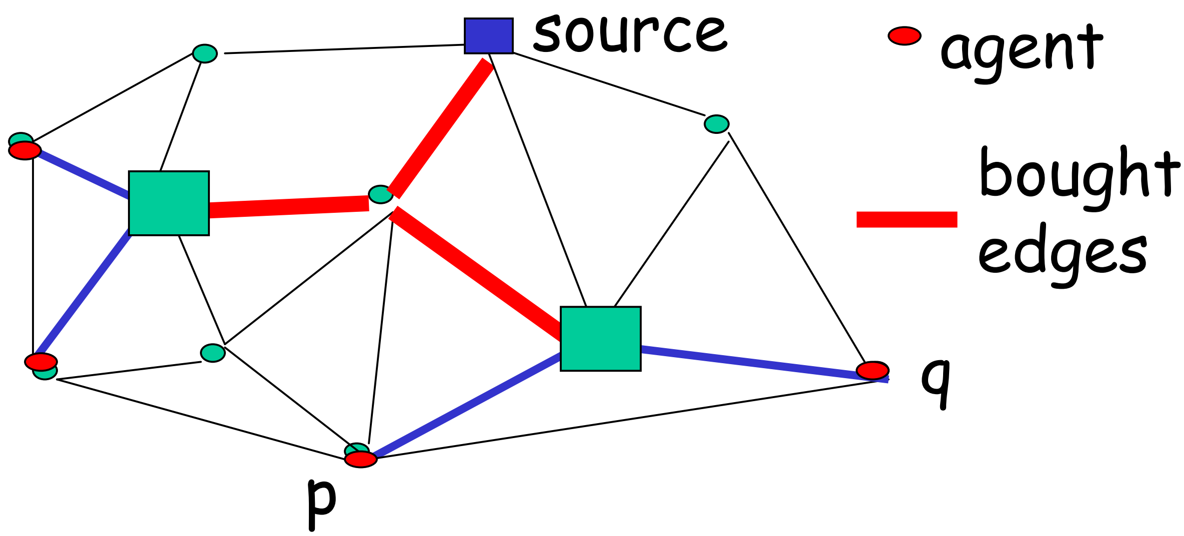
≈ Facility location ■: places where M or more clients can get together

≈ Steiner tree on selected gathering points

We have population monotone cost-sharing for both…

But: selection of gathering points not monotone…

rent-or-buy cost-sharing

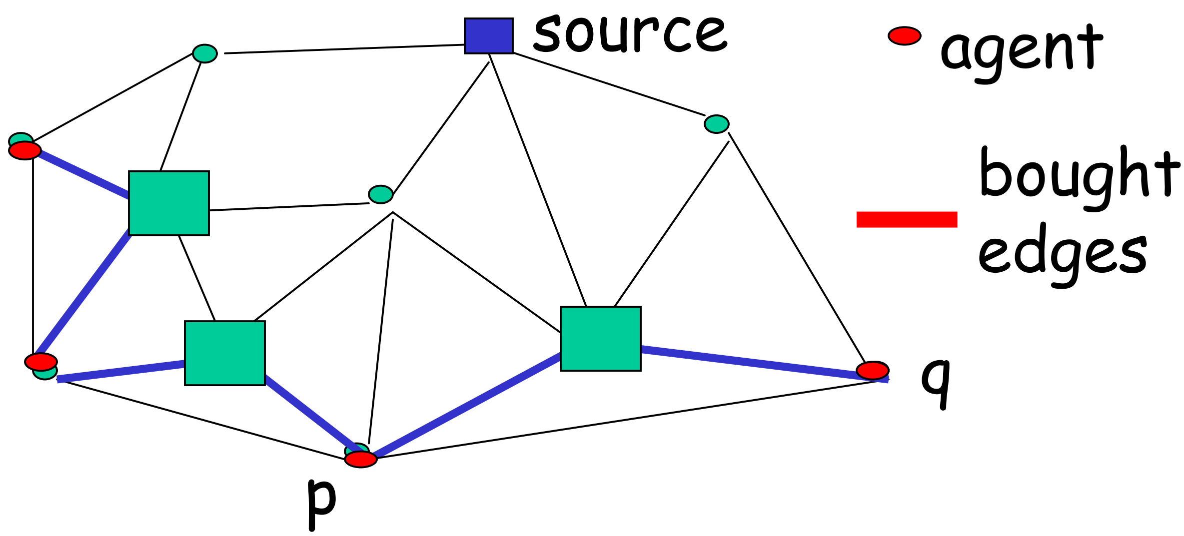


We have population monotone cost-sharing for both…

But: selection of gathering points not monotone…

- new client p' near old one p
- Helps add a center near p
- This hurt q (non-monotone)

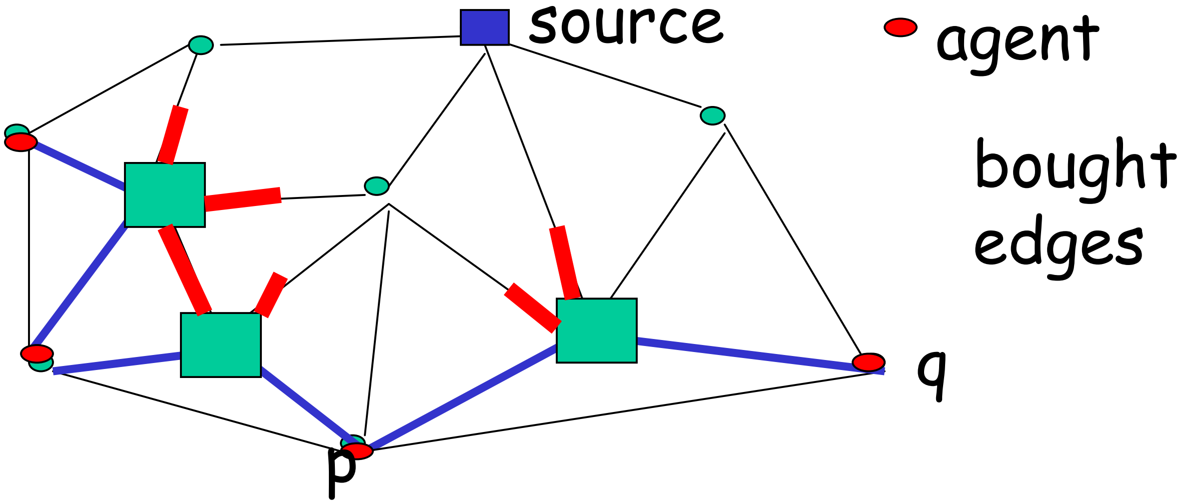
rent-or-buy cost-sharing



Combining facility location and Steiner tree:

- Add possible center whenever M or more clients can get together
  - M= multiplier for buying

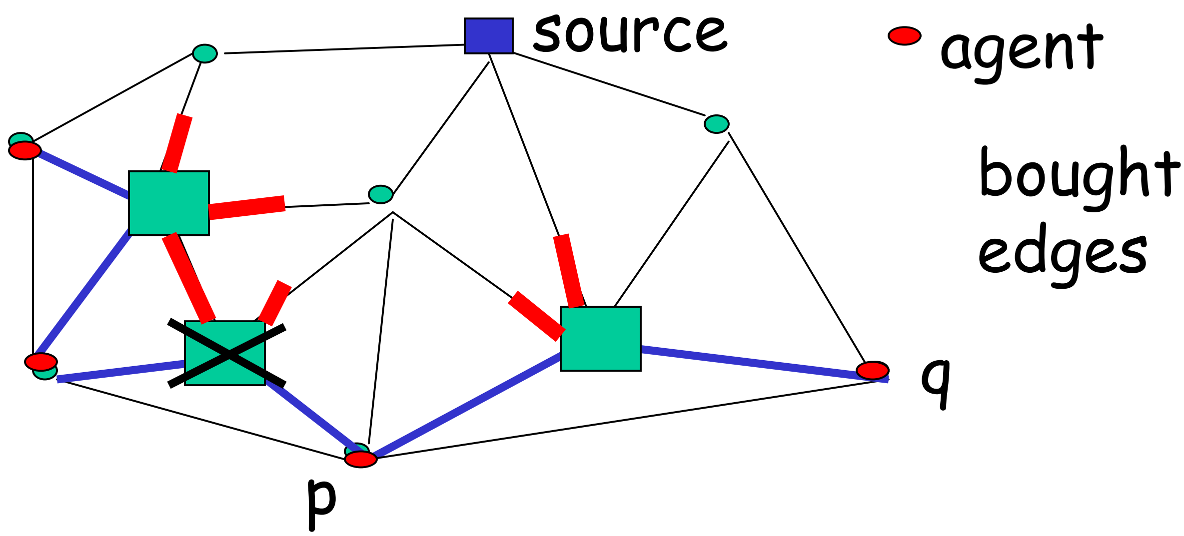
rent-or-buy cost-sharing



Combining facility location and Steiner tree:

- Add possible center whenever M or more clients can get together
- Each center builds Steiner tree and evenly shares cost
- Client select cheapest option for its share

rent-or-buy cost-sharing

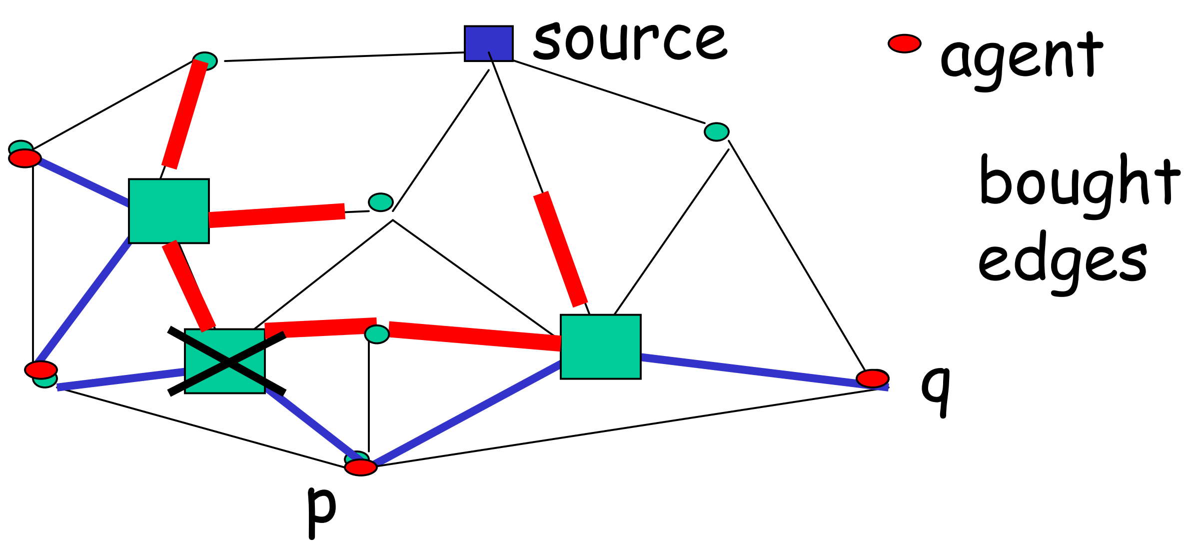


Combining facility location and Steiner tree:

Trouble:

- Client select cheapest option
- For actual tree: select subset of centers (analogous to ghost centers).

Rent-or-buy cost-sharing competitive?



Troubles:

- One client counted towards multiple centers?
- Two real centers can connect via closed ones

Theorem: shares are approx budget balanced.

Idea: centers closed if there is open center nearby

Conclusion

Facility Location/ Network Design gives rise to many interesting games

Price of Anarchy ↔ Best Nash ↔ Cost-Sharing?

Cost-sharing: central agent also builds an maintains the edges, using collected fees

- Core: no set of agents may be over-changed
  - Closely related to LP dual
- Population monotone: group strategy-proof for dealing with limited individual utility