NCP Group
Overview and New Results

Table of Contents

Node Swap Processes

Overview

	SNSP		**WPNSP**		**SPNSP**	
	AVE	**MAX**	**AVE**	**MAX**	**AVE**	**MAX**
Existence of OE	no		yes	open	yes	
Existence of PE	no		yes	no	yes	
Always convergence	no		yes	no	yes	
Convergence speed	∞		$\Theta(\lvert F\rvert \operatorname{diam}(G))$	∞	$\Theta(\lvert F\rvert \operatorname{diam}(G))$	$\Omega(\lvert V_F\rvert^2 \operatorname{diam}(G))$ $O\left(\binom{\lvert V_F\rvert + \operatorname{diam}(G) - 1}{\lvert V_F\rvert}\right)$
oPoA						
General graphs	$\Theta(\operatorname{diam}(G))$		$\Theta(\operatorname{diam}(G))$		$\Theta(\operatorname{diam}(G))$	
Layered graphs	$\Theta(\operatorname{diam}(G))$	$\Theta(\sqrt{\operatorname{diam}(G)})$	——		——	
oPoS	$\Theta(\operatorname{diam}(G))$		1	$\Theta(\operatorname{diam}(G))$	1	$O(\operatorname{diam}(G))$

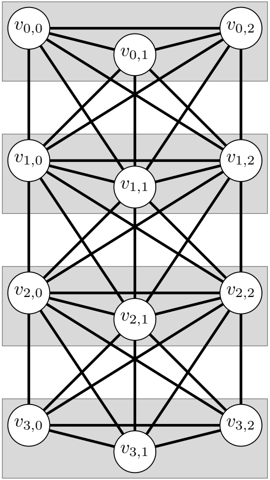
OE: operation equilibrium – Nash equilibrium isomorphic to initial graph
PE: process equilibrium – Nash equilibrium reachable from initial graph via allowed node swaps
oPoA: operational Price of Anarchy
oPoS: operational Price of Stability

Node Swap Processes

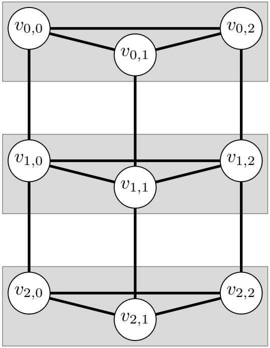
Layered Graphs

New Results on Path-Layered Graphs

Complete layered graph



Path-layered graph



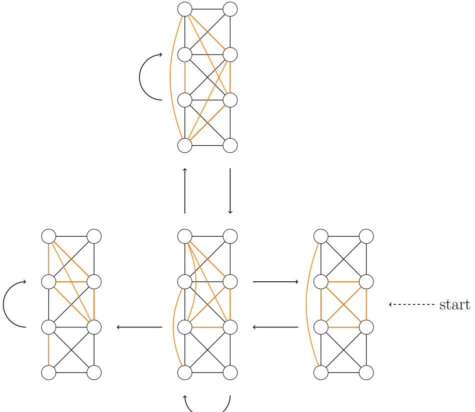
New Results on Path-Layered Graphs

- Former Result: Reachable SNSP with layered graph and friendship matching $\Rightarrow$ SNSP can reach PE

New Results on Path-Layered Graphs

- Former Result: Reachable SNSP with layered graph and friendship matching $\Rightarrow$ SNSP can reach PE
- This is not true for friendships that are isomorphic to a subgraph of the layered graph

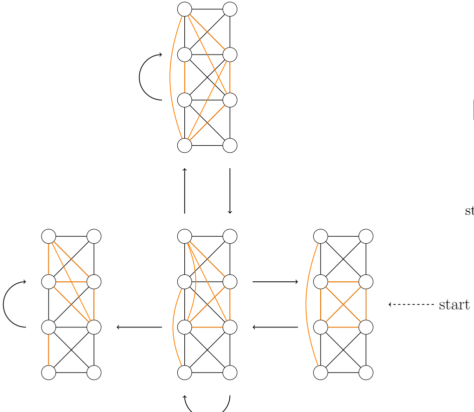
Complete layered graph



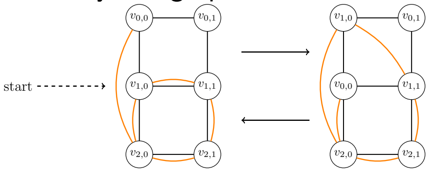
New Results on Path-Layered Graphs

- Former Result: Reachable SNSP with layered graph and friendship matching ⇒ SNSP can reach PE
- This is not true for friendships that are isomorphic to a subgraph of the layered graph

Complete layered graph

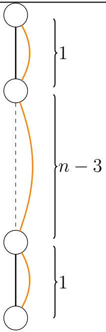


Path-layered graph



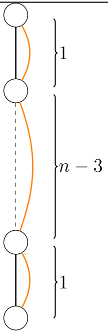
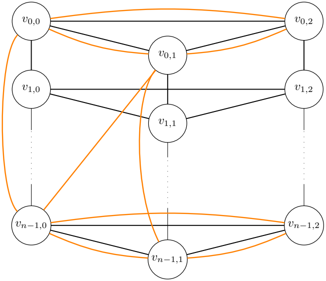
New Results on Path-Layered Graphs

Layered graphs with n layers, average cost function:

	Complete layered graph	
Max. distance between friends in OE:	$n-3$	
oPoA	$\Theta(n)$	
		

New Results on Path-Layered Graphs

Layered graphs with n layers, average cost function:

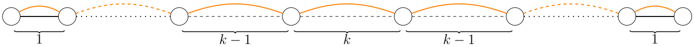
	Complete layered graph	Path-layered graph
Max. distance between friends in OE:	$n-3$	n
oPoA	$\Theta(n)$	$\Theta(n)$
		

New Results on Path-Layered Graphs

Layered graphs with n layers, maximum cost function:

	Complete layered graph	
Max. distance between friends in OE:	$\lfloor \sqrt{n-1} \rfloor$	
oPoA	$\Theta(\sqrt{n})$	

- Complete layered graph:

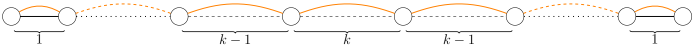


New Results on Path-Layered Graphs

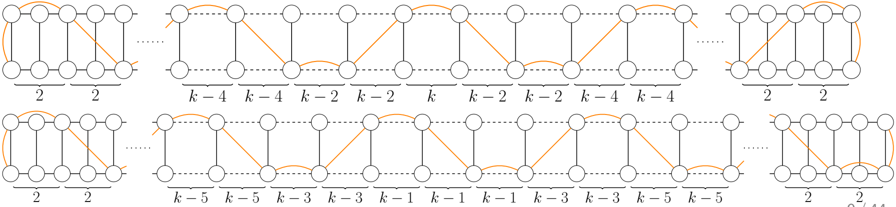
Layered graphs with n layers, maximum cost function:

	Complete layered graph	Path-layered graph
Max. distance between friends in OE:	$\lfloor\sqrt{n-1}\rfloor$	$\left\lfloor\sqrt{n-\frac{3}{4}}+\frac{1}{2}\right\rfloor$
oPoA	$\Theta(\sqrt{n})$	$\Theta(\sqrt{n})$

- Complete layered graph:

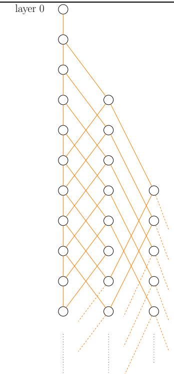


- Path-layered graph:



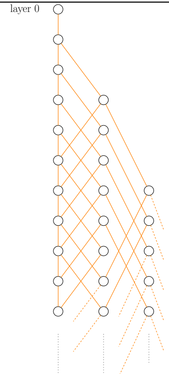
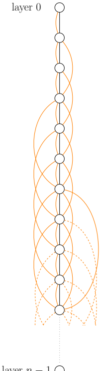
New Results on Path-Layered Graphs

Layered graphs with n layers, maximum cost function:

	Complete layered graph	Path-layered graph
Max. distance between friends in OE:	$\lfloor \sqrt{n-1} \rfloor$	$\left\lfloor \sqrt{n - \frac{3}{4}} + \frac{1}{2} \right\rfloor$
oPoA	$\Theta(\sqrt{n})$	$\Theta(\sqrt{n})$
		

New Results on Path-Layered Graphs

Layered graphs with n layers, maximum cost function:

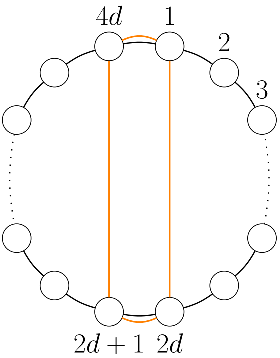
	Complete layered graph	Path-layered graph
Max. distance between friends in OE:	$\lfloor\sqrt{n-1}\rfloor$	$\left\lfloor\sqrt{n-\frac{3}{4}}+\frac{1}{2}\right\rfloor$
oPoA	$\Theta(\sqrt{n})$	$\Theta(\sqrt{n})$
		

Node Swap Processes

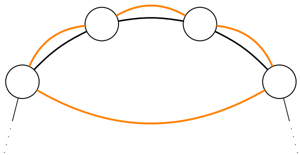
Operational Price of Stability

Operational Price of Stability
SNSP–AVE

oPoS $= \Theta(\text{diam}(G))$:



reachable PE

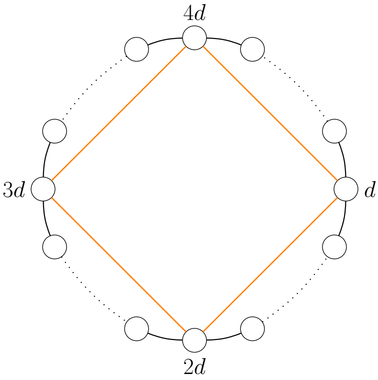


constant social costs

$\Rightarrow$ any social optimum has constant social costs,
any OE has social costs of $\Theta(d)$

Operational Price of Stability
SNSP–MAX

oPoS $= \Theta(\mathrm{diam}(G))$:



non-reachable PE

$\Rightarrow$ any social optimum has constant social costs,
any OE has social costs of $\Theta(d)$

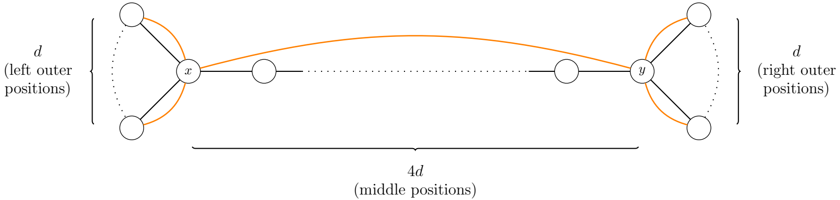
Operational Price of Stability

WPNSP

- AVE: oPoS $= 1$:
  every social optimum is OE

Operational Price of Stability
WPNSP

- AVE: oPoS $= 1$:
  every social optimum is OE
- MAX: oPoS $= \Theta(\text{diam}(G))$:

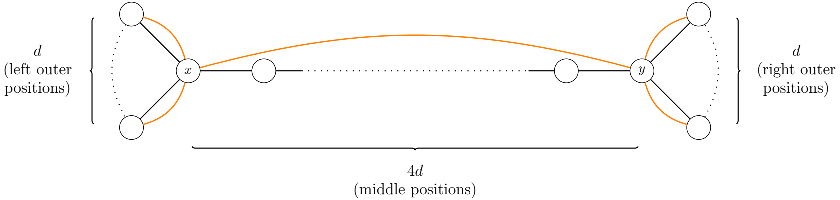


reachable example that has PE

$\Rightarrow$ any social optimum has social costs of $\Theta(d)$,
any OE has social costs of $\Theta(d^2)$

Operational Price of Stability
SPNSP

- AVE: oPoS $= 1$:
  every social optimum is OE
- MAX: oPoS between 1 and $\Theta(\text{diam}(G))$ ...:



construction not working here

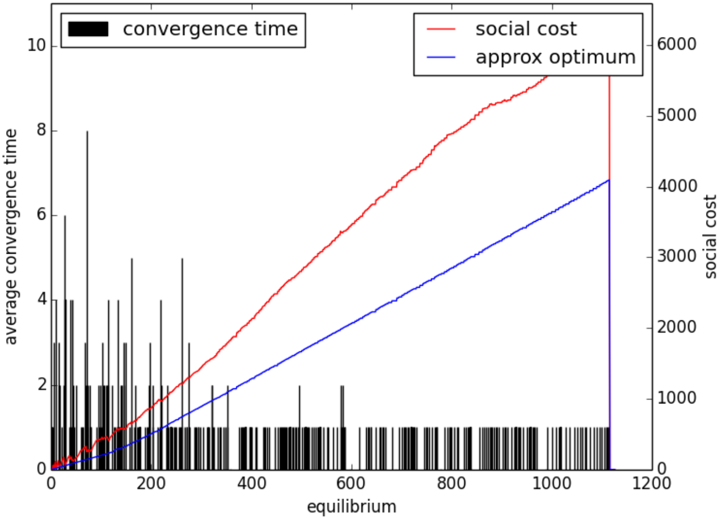
Shortcut Process

Shortcut Process

Simulations

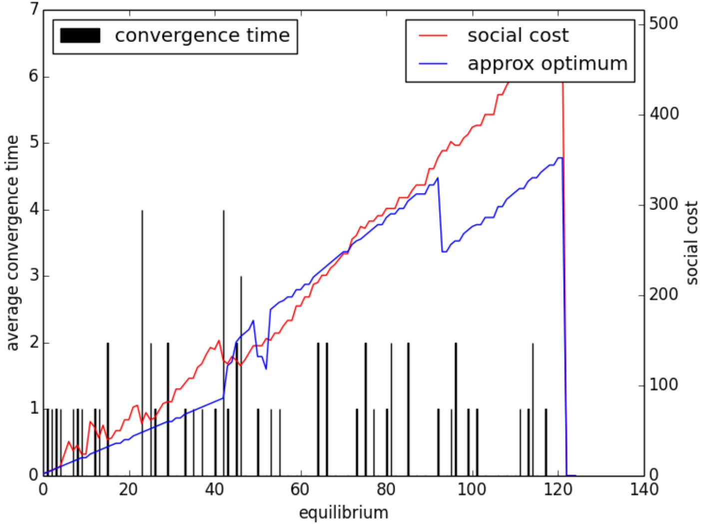
Social Costs

150 nodes, 1 instance

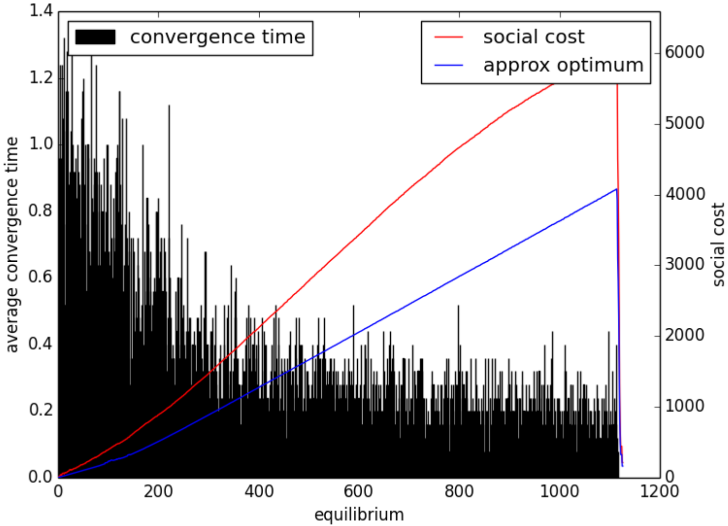


Social Costs

50 nodes, 1 instance

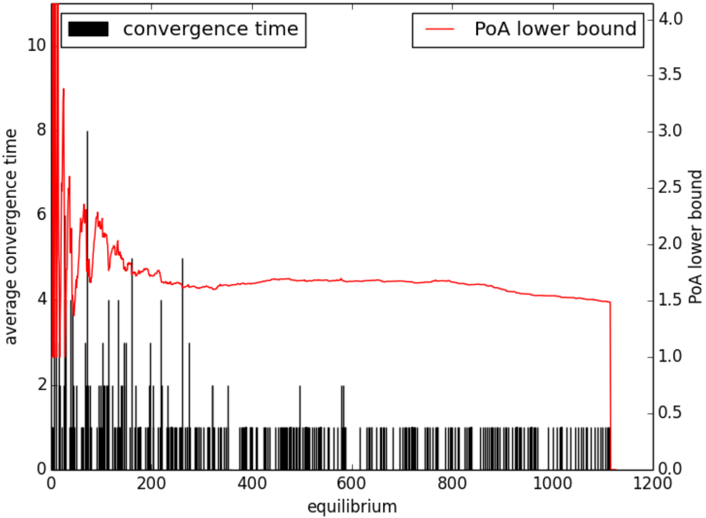


Social Costs
150 nodes, average



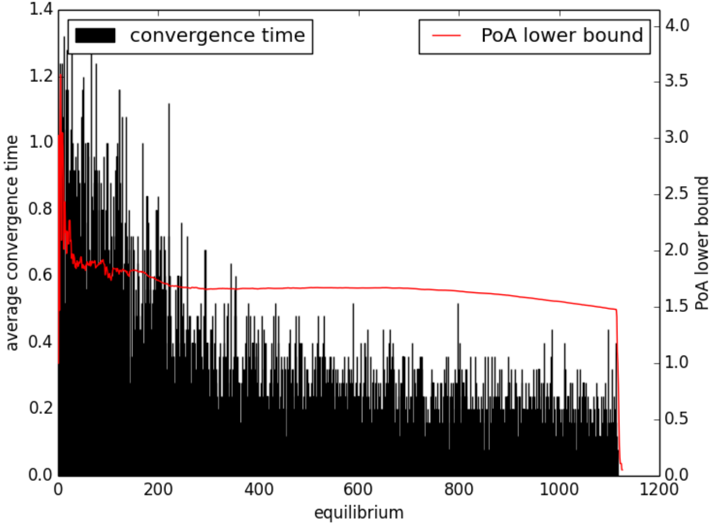
Price of Anarchy

150 nodes, 1 instance



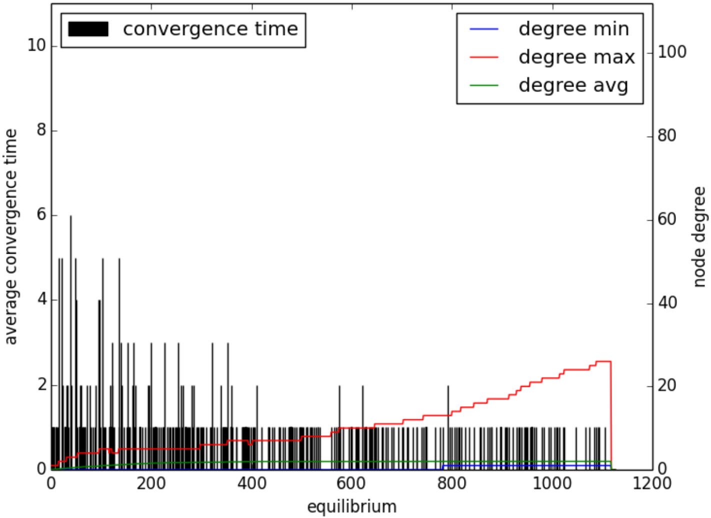
Price of Anarchy

150 nodes, average



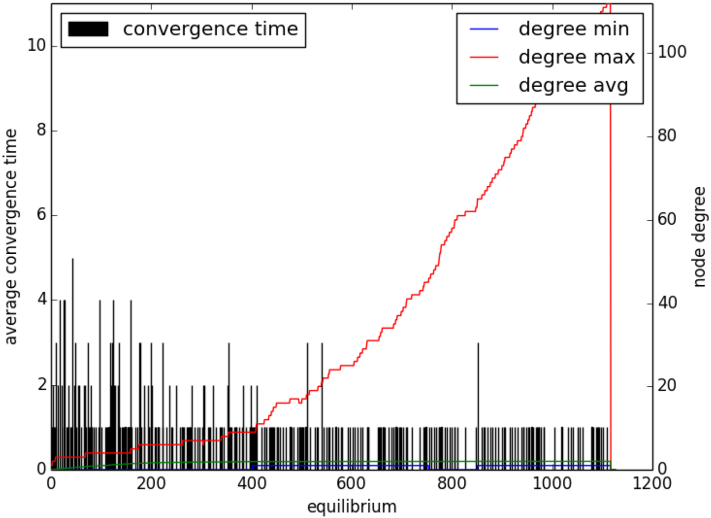
Node Degree

150 nodes, 1 instance



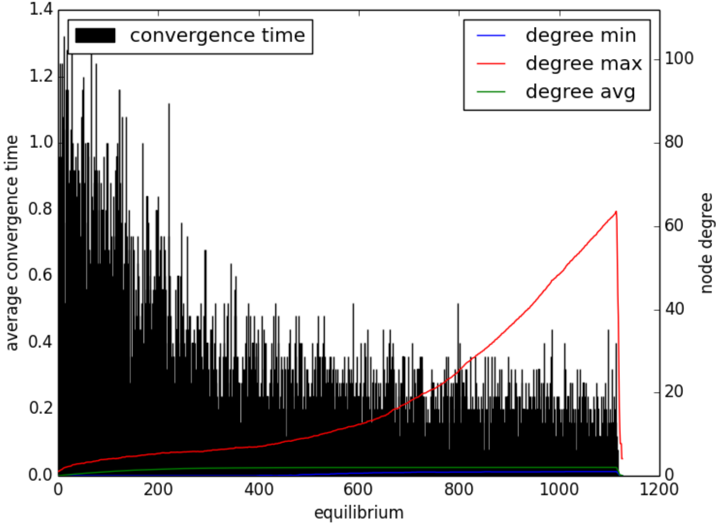
Node Degree

150 nodes, 1 instance



Node Degree

150 nodes, average



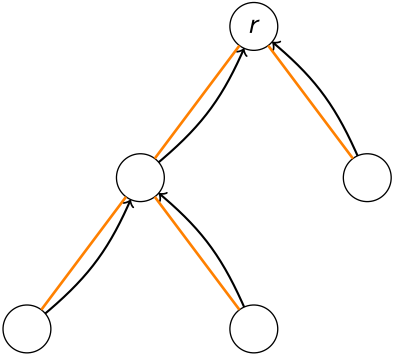
Shortcut Process

Approximation Algorithms for Social Optimum

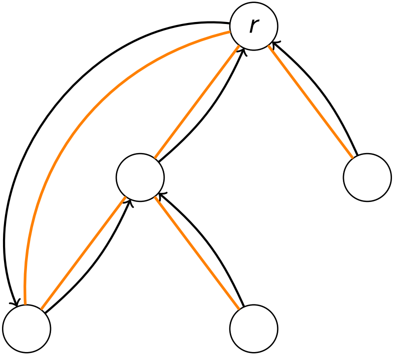
Approximation Algorithms for Social Optimum

- Trivial algorithm with factor 2: Build a star
- Better idea: Treat friendship components independently
  Easy cases:

Trees:

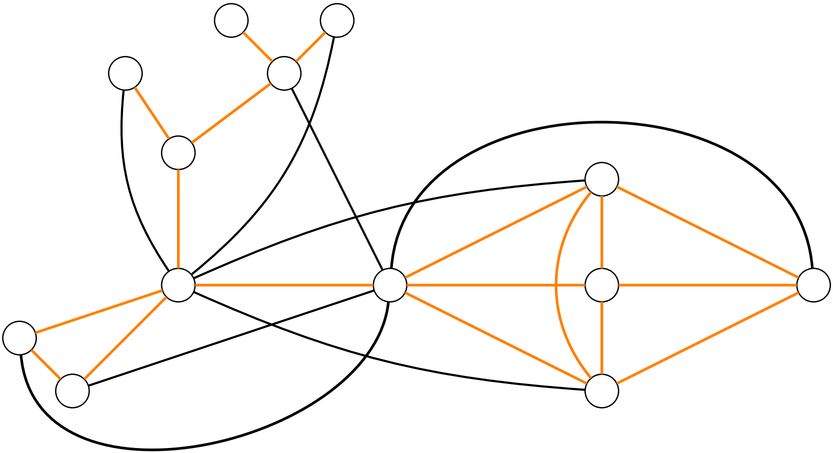


One-cycle-graphs:

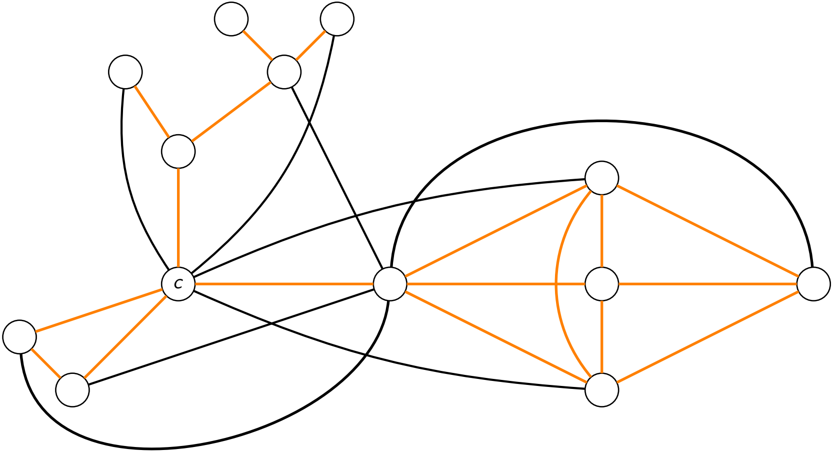


Approximation Algorithms for Social Optimum

AVE

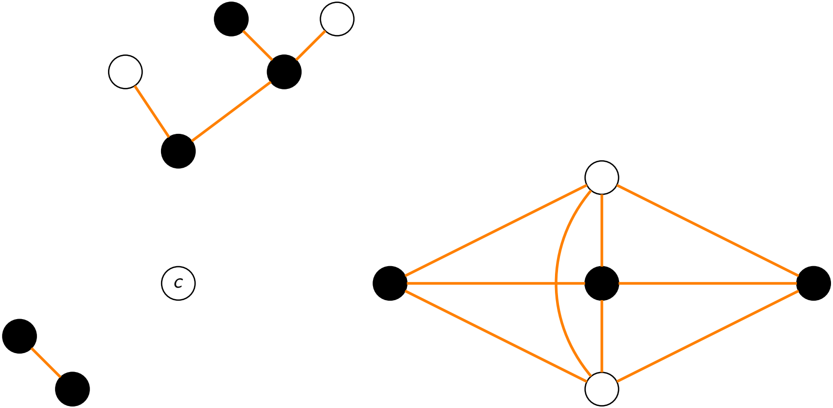


Approximation Algorithms for Social Optimum
AVE



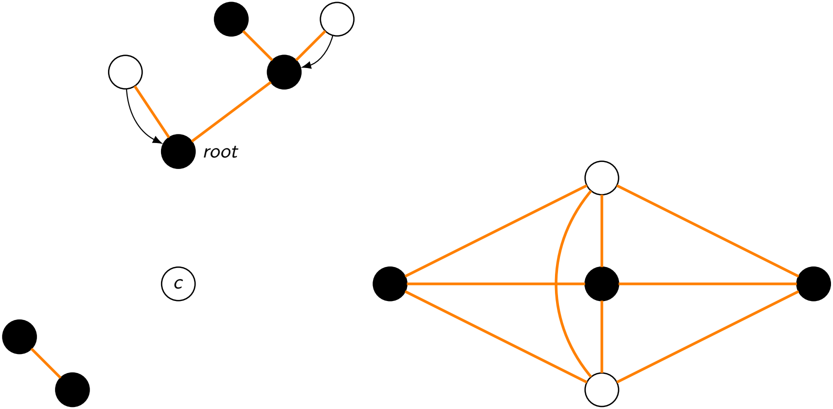
Choose star center c with highest number of friends plus static edges in component

Approximation Algorithms for Social Optimum
AVE



Black nodes have shortcut to c

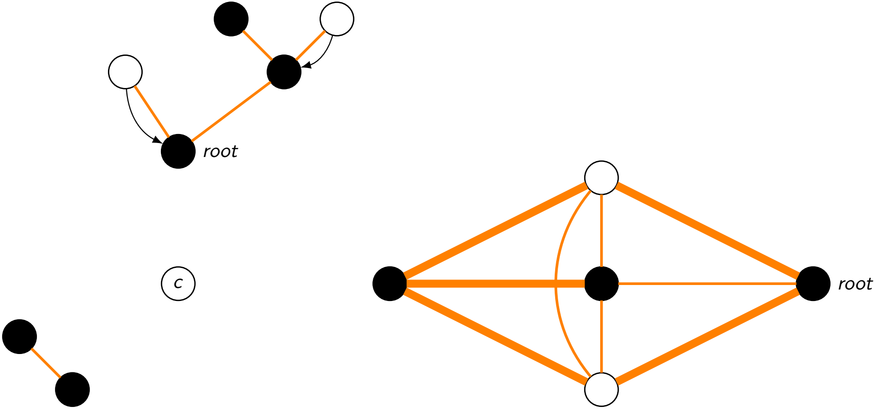
Approximation Algorithms for Social Optimum
AVE



Treat new friendship components:

- Trees: As before

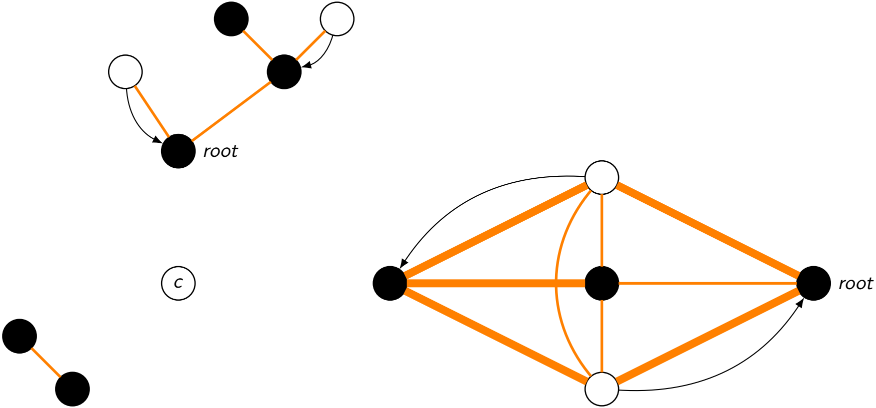
Approximation Algorithms for Social Optimum
AVE



Treat new friendship components:

- Trees: As before
- Other components: Find spanning one-cycle-graph

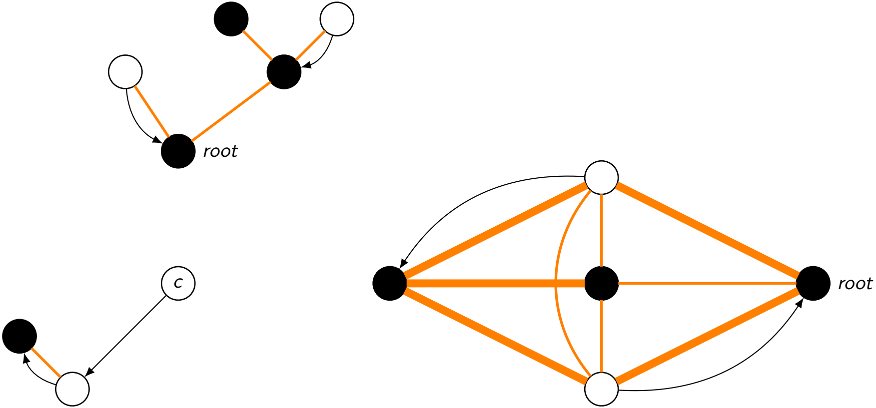
Approximation Algorithms for Social Optimum
AVE



Treat new friendship components:

- Trees: As before
- Other components: Find spanning one-cycle-graph

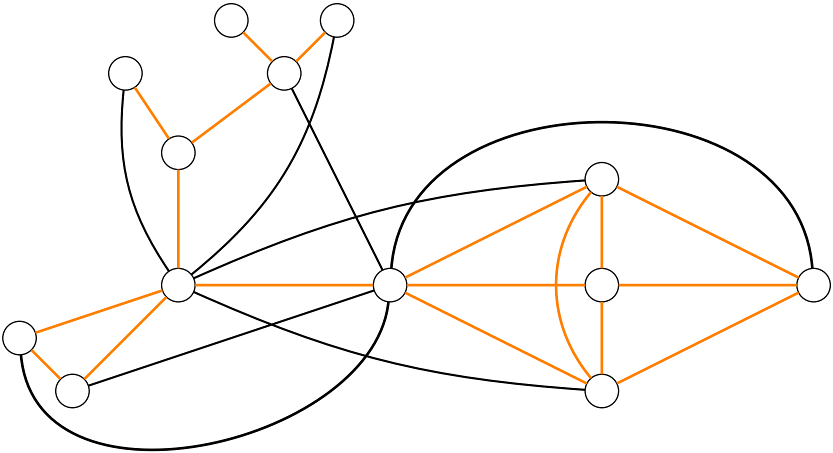
Approximation Algorithms for Social Optimum
AVE



Use shortcut of c

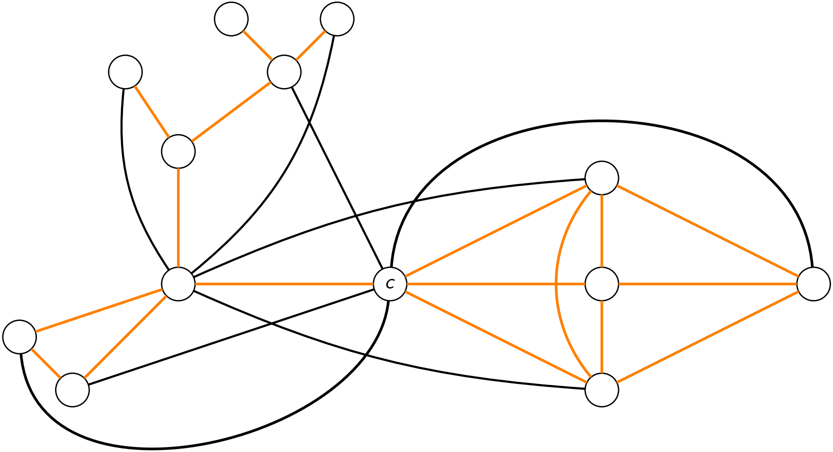
Approximation Algorithms for Social Optimum

MAX



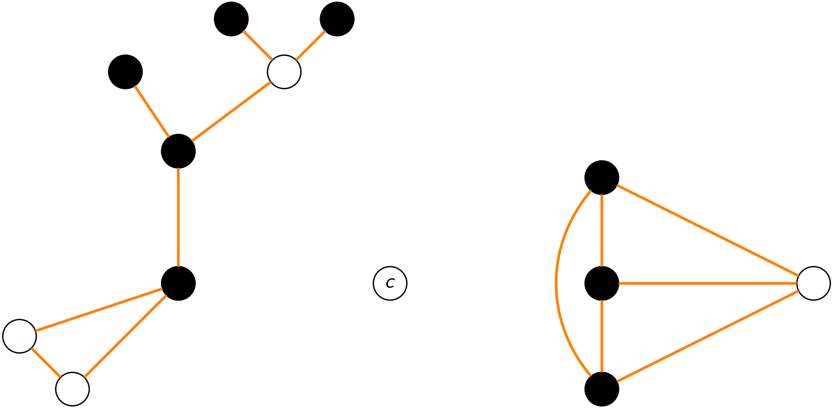
Approximation Algorithms for Social Optimum

MAX



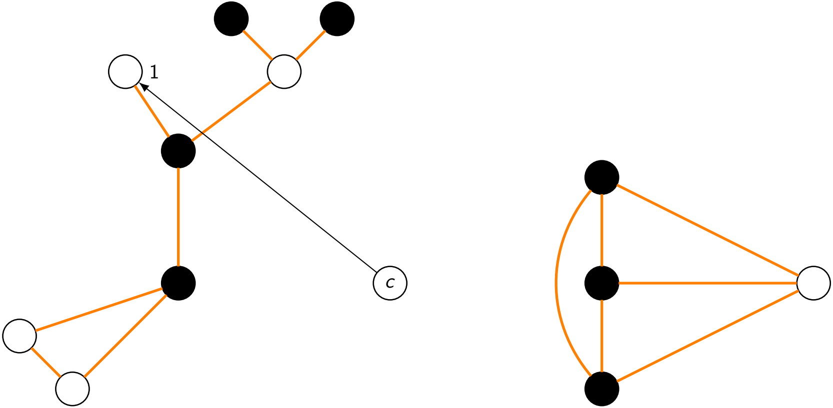
Choose star center c with highest number of static edges in component

Approximation Algorithms for Social Optimum
MAX



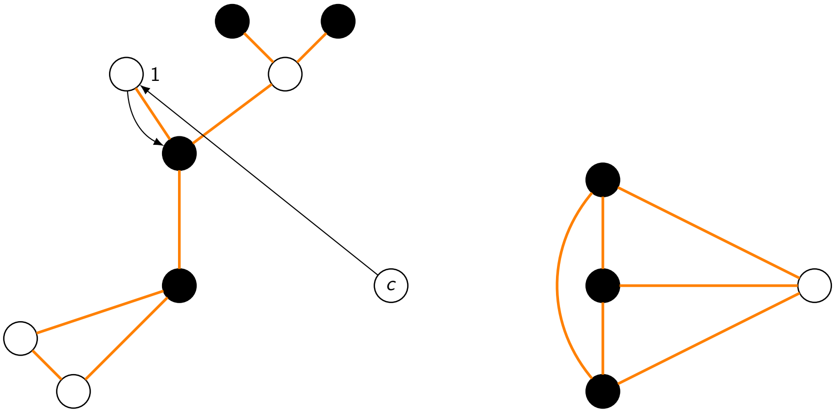
Black nodes have shortcut to c

Approximation Algorithms for Social Optimum
MAX



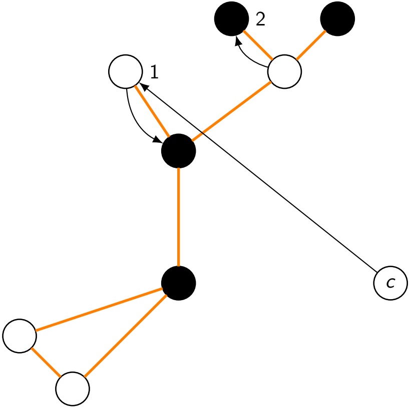
Sort nodes with friends by non-decreasing number of friends:
c puts shortcut to first node

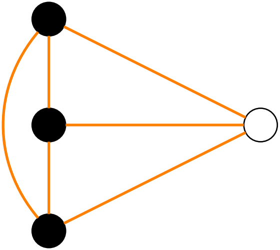
Approximation Algorithms for Social Optimum
MAX



Sort nodes with friends by non-decreasing number of friends:
first node uses own and neighboring shortcuts to satisfy itself

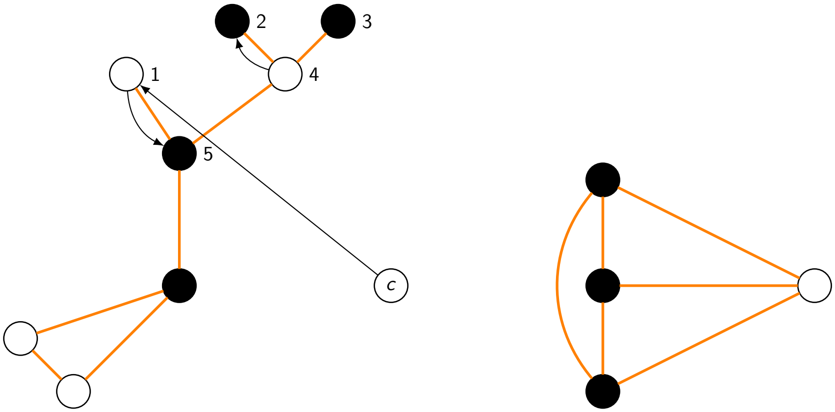
Approximation Algorithms for Social Optimum
MAX





Sort nodes with friends by non-decreasing number of friends:
next node uses own and neighboring shortcuts to satisfy itself

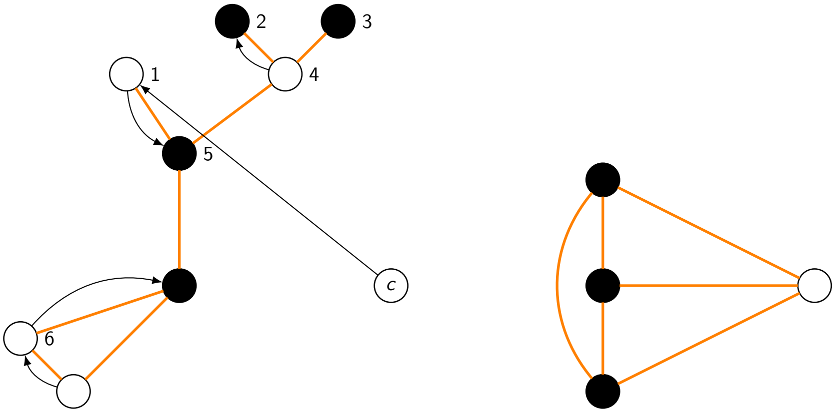
Approximation Algorithms for Social Optimum
MAX



Sort nodes with friends by non-decreasing number of friends: next node uses own and neighboring shortcuts to satisfy itself

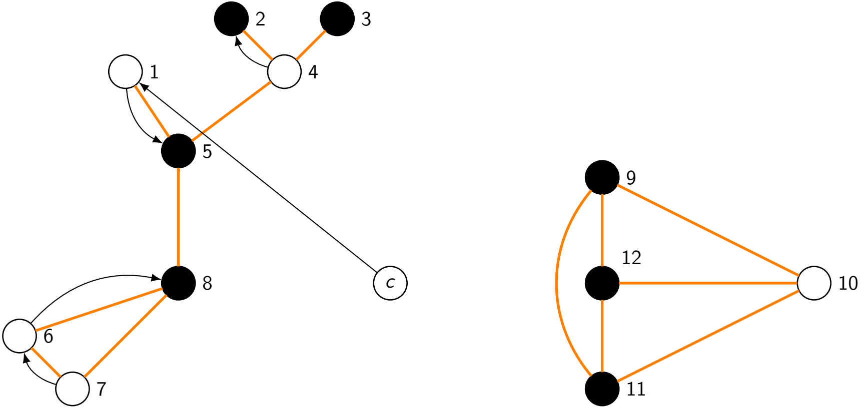
Approximation Algorithms for Social Optimum

MAX



Sort nodes with friends by non-decreasing number of friends:
next node uses own and neighboring shortcuts to satisfy itself

Approximation Algorithms for Social Optimum
MAX



Sort nodes with friends by non-decreasing number of friends:
next node uses own and neighboring shortcuts to satisfy itself

Approximation Algorithms for Social Optimum

Approximation Factors

- Observation: There is a solution that covers all friendships
  $\Rightarrow$ algorithms find one
- AVE: $\frac{2|F|-4}{|F|+1}$
  (algorithm covers at least 4 friendship if $|F| \geq 4$)
- MAX: $\frac{2|V_F|-1}{|V_F|+2}$ with $V_F := |\{v \in V \mid v \text{ has friends}\}|$
  (algorithm satisfies at least 1 center node)