The Maximum Carpool Matching Problem

Gilad Kutiel
gkutiel@cs.technion.ac.il

CSR 2017

The Maximum Carpool Matching Problem (informal)

- A group of people want to carpool from point A to point B

The Maximum Carpool Matching Problem (informal)

- A group of people want to carpool from point A to point B
- We want to maximize the total welfare

The Maximum Carpool Matching Problem (informal)

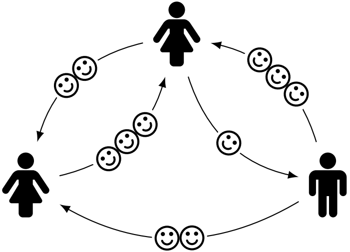
- A group of people want to carpool from point A to point B
- We want to maximize the total welfare
- How they should be matched?

The Maximum Carpool Matching Problem (informal)

- A group of people want to carpool from point A to point B
- We want to maximize the total welfare
- How they should be matched?

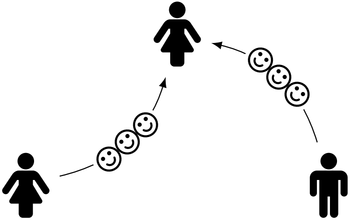
The Maximum Carpool Matching Problem (informal)

- A group of people want to carpool from point A to point B
- We want to maximize the total welfare
- How they should be matched?



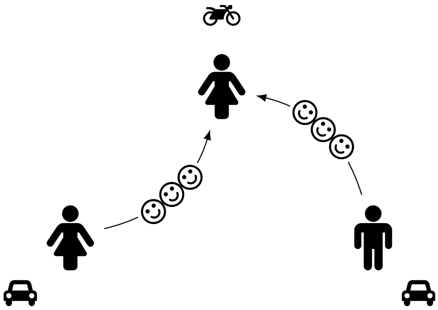
The Maximum Carpool Matching Problem (informal)

- A group of people want to carpool from point A to point B
- We want to maximize the total welfare
- How they should be matched?



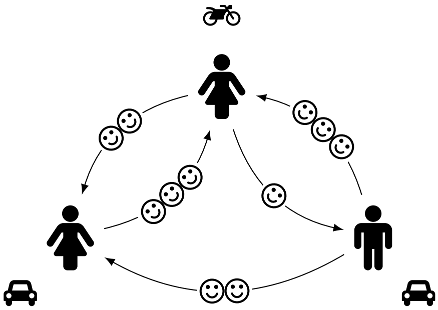
The Maximum Carpool Matching Problem (informal)

- A group of people want to carpool from point A to point B
- We want to maximize the total welfare
- How they should be matched?



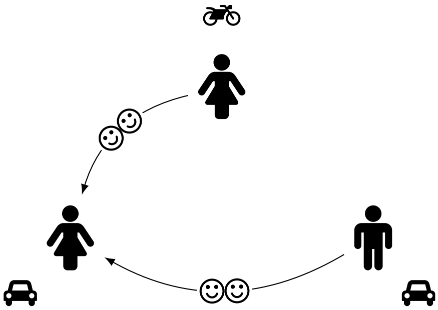
The Maximum Carpool Matching Problem (informal)

- A group of people want to carpool from point A to point B
- We want to maximize the total welfare
- How they should be matched?



The Maximum Carpool Matching Problem (informal)

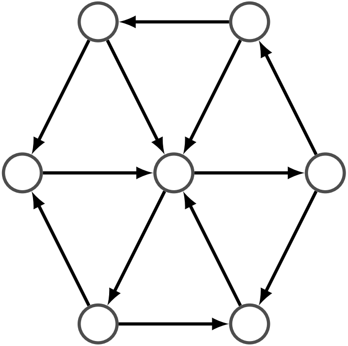
- A group of people want to carpool from point A to point B
- We want to maximize the total welfare
- How they should be matched?



Formal Definition

Input: (D, w, c)

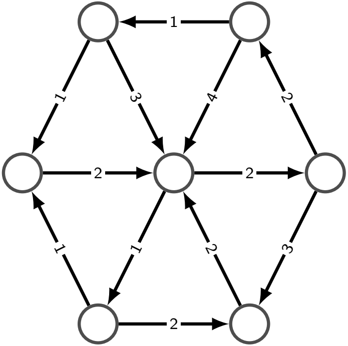
- $D = (V, A)$



Formal Definition

Input: (D, w, c)

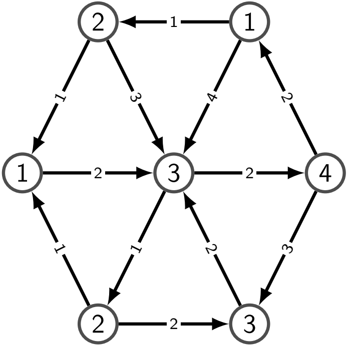
- $D = (V, A)$
- $w : A \to \mathbb{R}_+$



Formal Definition

Input: (D, w, c)

- $D = (V, A)$
- $w : A \to \mathbb{R}_+$
- $c : V \to \mathbb{N}$

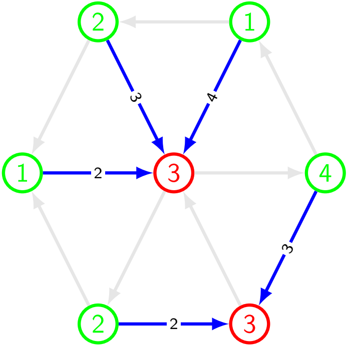


Formal Definition

Input: (D, w, c)

- $D = (V, A)$
- $w : A \to \mathbb{R}_+$
- $c : V \to \mathbb{N}$

Matching: (P, D, M)



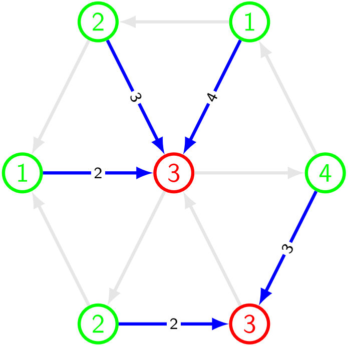
Formal Definition

Input: (D, w, c)

- $D = (V, A)$
- $w : A \to \mathbb{R}_+$
- $c : V \to \mathbb{N}$

Matching: (P, D, M)

- $P \cap D = \emptyset,\ P \cup D = V$



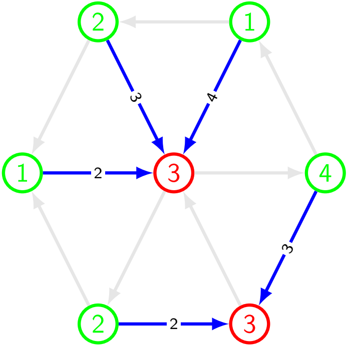
Formal Definition

Input: (D, w, c)

- $D = (V, A)$
- $w : A \to \mathbb{R}_+$
- $c : V \to \mathbb{N}$

Matching: (P, D, M)

- $P \cap D = \emptyset$, $P \cup D = V$
- $M \subseteq A \cap (P \times D)$



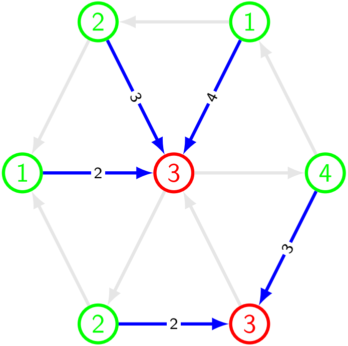
Formal Definition

Input: (D, w, c)

- $D = (V, A)$
- $w : A \to \mathbb{R}_+$
- $c : V \to \mathbb{N}$

Matching: (P, D, M)

- $P \cap D = \emptyset,\ P \cup D = V$
- $M \subseteq A \cap (P \times D)$
- $\forall p \in P, deg_{out}^{M}(p) \leq 1$



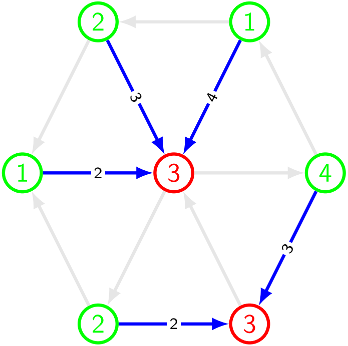
Formal Definition

Input: (D, w, c)

- $D = (V, A)$
- $w : A \to \mathbb{R}_+$
- $c : V \to \mathbb{N}$

Matching: (P, D, M)

- $P \cap D = \emptyset,\ P \cup D = V$
- $M \subseteq A \cap (P \times D)$
- $\forall p \in P, deg^M_{out}(p) \leq 1$
- $\forall d \in D, deg^M_{in}(d) \leq c(d)$



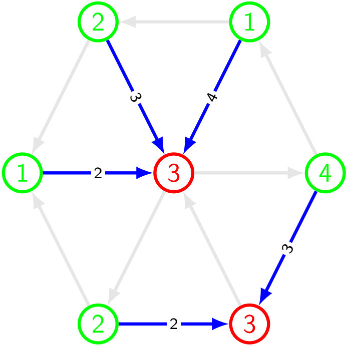
Formal Definition

Input: (D, w, c)

- $D = (V, A)$
- $w : A \to \mathbb{R}_+$
- $c : V \to \mathbb{N}$

Matching: (P, D, M)

- $P \cap D = \emptyset,\ P \cup D = V$
- $M \subseteq A \cap (P \times D)$
- $\forall p \in P, deg_{out}^M(p) \leq 1$
- $\forall d \in D, deg_{in}^M(d) \leq c(d)$

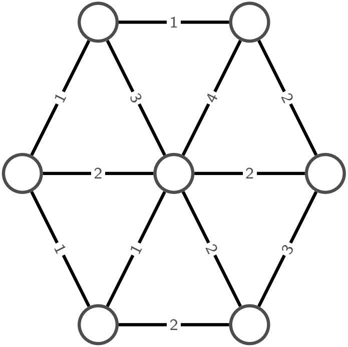


We seek for a matching that maximizes

$$w(M) = \sum_{e \in M} w(e)$$

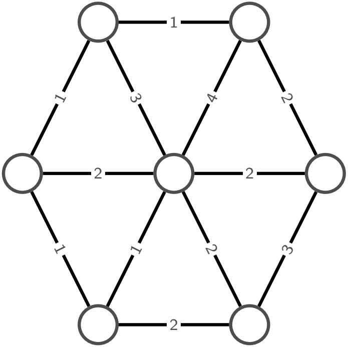
Maximum Spanning Star Forest (MSSF)

- A special case of the MAXIMUM CARPOOL MATCHING problem (MCM)



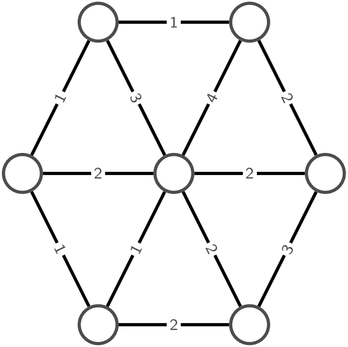
Maximum Spanning Star Forest (MSSF)

- A special case of the MAXIMUM CARPOOL MATCHING problem (MCM)
- Undirected graph



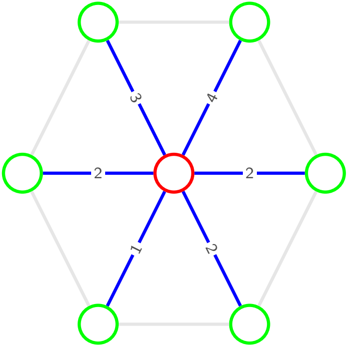
Maximum Spanning Star Forest (MSSF)

- A special case of the MAXIMUM CARPOOL MATCHING problem (MCM)
- Undirected graph
- No capacities



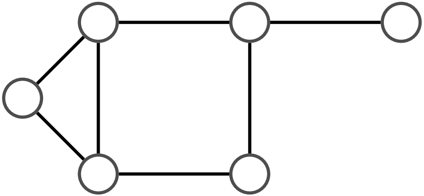
Maximum Spanning Star Forest (MSSF)

- A special case of the Maximum Carpool Matching problem (MCM)
- Undirected graph
- No capacities



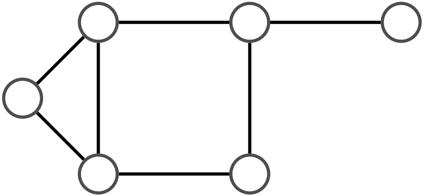
Hardness

- MSSF (and thus MCM) is APX-hard



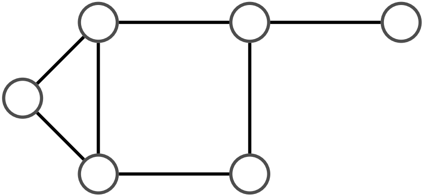
Hardness

- MSSF (and thus MCM) is APX-hard
- Unweighted MSSF (uniform weight)



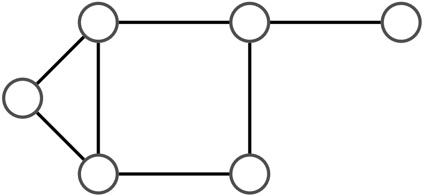
Hardness

- MSSF (and thus MCM) is APX-hard
- Unweighted MSSF (uniform weight)
  - Maximize the number of leaves



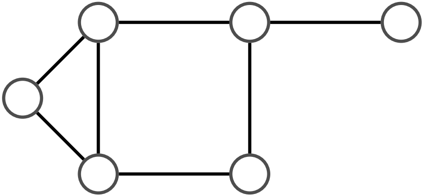
Hardness

- MSSF (and thus MCM) is APX-hard
- Unweighted MSSF (uniform weight)
  - Maximize the number of leaves
  - Minimize the number of centers



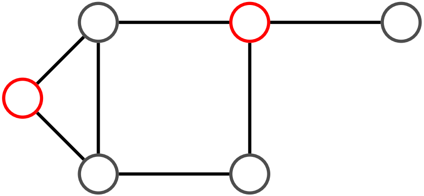
Hardness

- MSSF (and thus MCM) is APX-hard
- Unweighted MSSF (uniform weight)
  - Maximize the number of leaves
  - Minimize the number of centers
    - (Minimum Dominating Set problem)



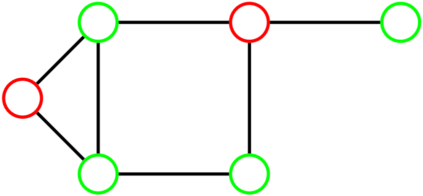
Hardness

- MSSF (and thus MCM) is APX-hard
- Unweighted MSSF (uniform weight)
  - Maximize the number of leaves
  - Minimize the number of centers
    - (Minimum Dominating Set problem)



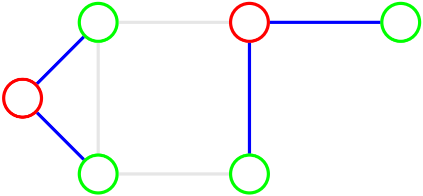
Hardness

- MSSF (and thus MCM) is APX-hard
- Unweighted MSSF (uniform weight)
  - Maximize the number of leaves
  - Minimize the number of centers
    - (Minimum Dominating Set problem)

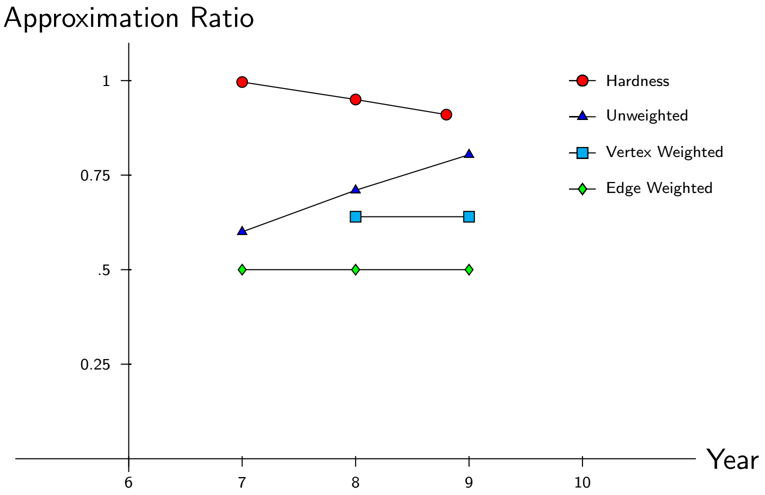


Hardness

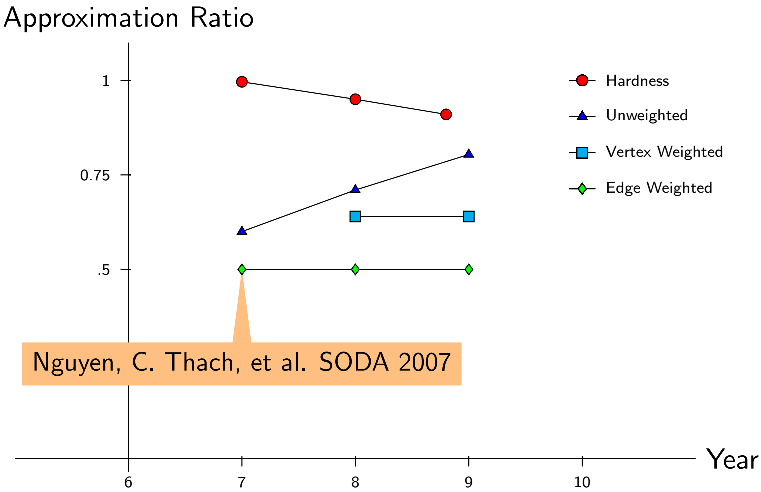
- MSSF (and thus MCM) is APX-hard
- Unweighted MSSF (uniform weight)
  - Maximize the number of leaves
  - Minimize the number of centers
    - (Minimum Dominating Set problem)



Previous Work (MSSF)

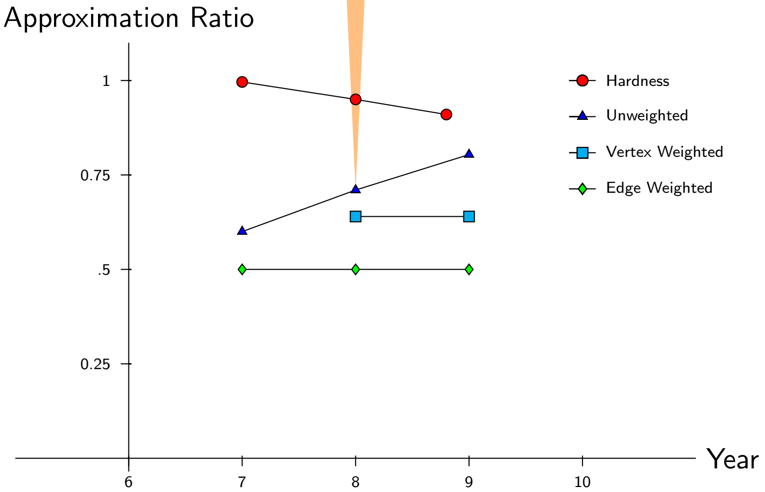


Previous Work (MSSF)

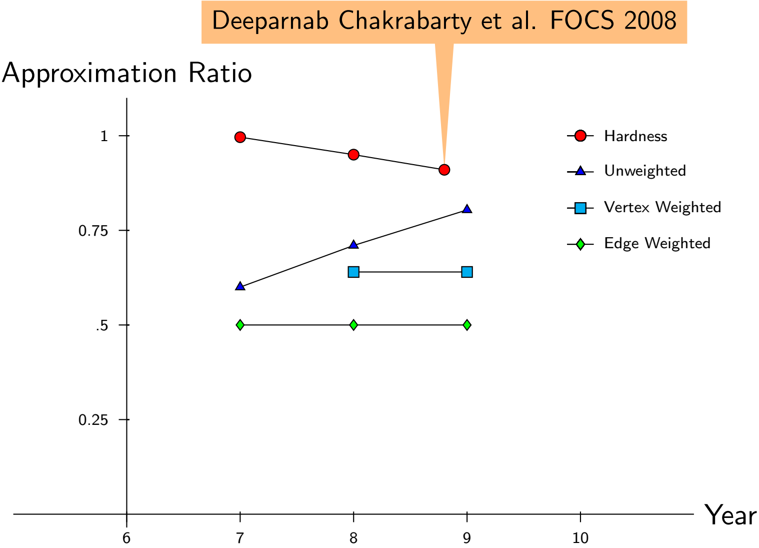


Previous Work (MSSF)

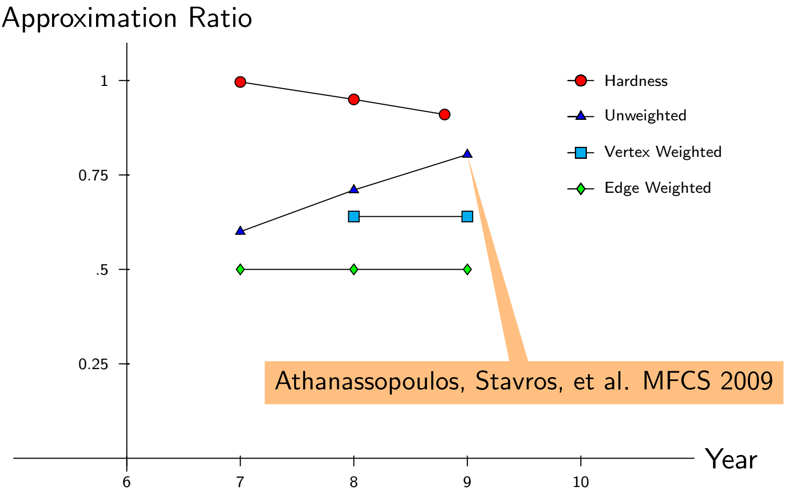
Chen, Ning, et al. APPROX 2007



Previous Work (MSSF)



Previous Work (MSSF)



Previous Work (MCM)

- Heuristics + experiments

Previous Work (MCM)

- Heuristics + experiments
  - Solution's quality matters

Previous Work (MCM)

- Heuristics + experiments
  - Solution's quality matters
- No approximation algorithm

Previous Work (MCM)

- Heuristics + experiments
  - Solution's quality matters
- No approximation algorithm
- The algorithms for MSSF do not generalize to MCM

Our Result (MCM)

- First approximation algorithms

Our Result (MCM)

- First approximation algorithms
  - $1/2$-approximation (unweighted)

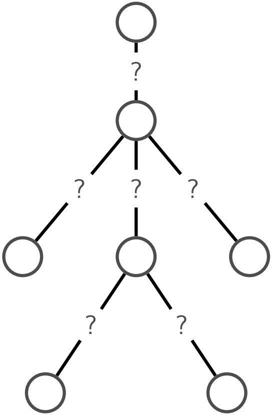
Our Result (MCM)

- First approximation algorithms
  - 1/2-approximation (unweighted)
  - 1/3-approximation (weighted)

Maximum Carpool Matching

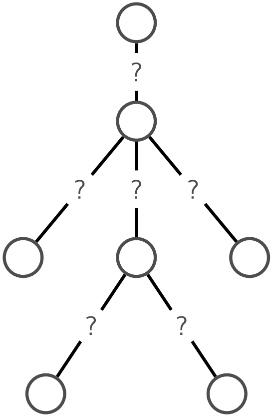
MSSF on Trees

- Optimally solvable (using DP)



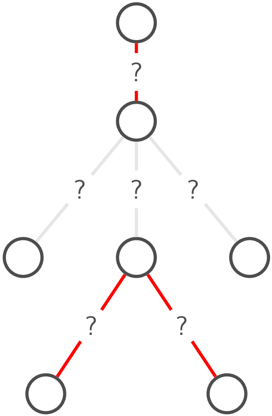
MSSF on Trees

- Optimally solvable (using DP)
- At least 1/2 of the weight



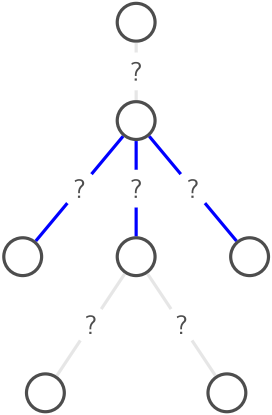
MSSF on Trees

- Optimally solvable (using DP)
- At least 1/2 of the weight



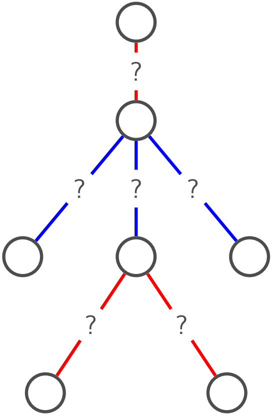
MSSF on Trees

- Optimally solvable (using DP)
- At least 1/2 of the weight



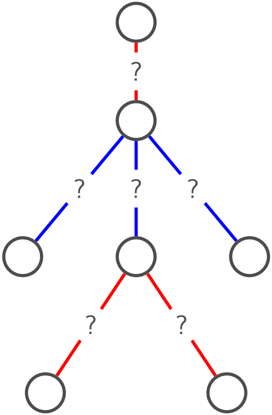
MSSF on Trees

- Optimally solvable (using DP)
- At least 1/2 of the weight



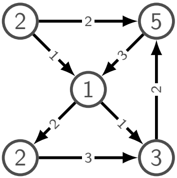
MSSF on Trees

- Optimally solvable (using DP)
- At least 1/2 of the weight
- 1/2-approximation algorithm to MSSF (Max Spanning Tree $\geq$ OPT)



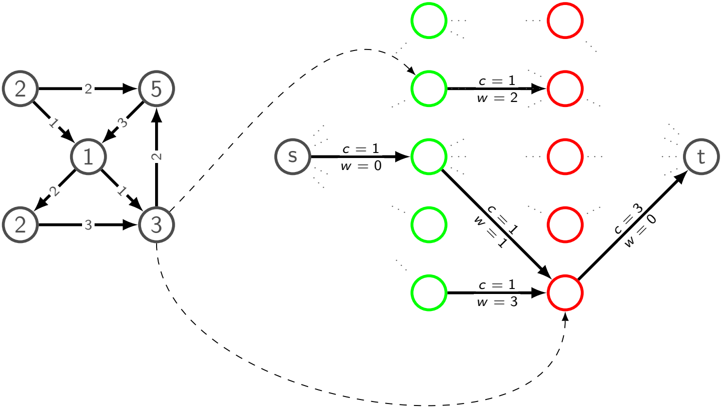
Relaxed Matching

- Every vertex can be a driver and a passenger at the same time



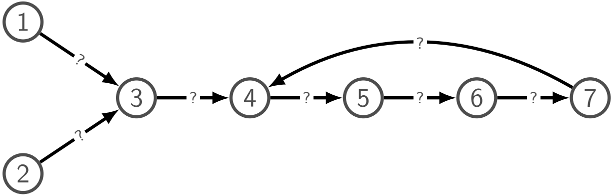
Relaxed Matching

- Every vertex can be a driver and a passenger at the same time



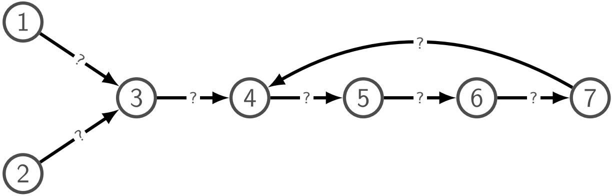
Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)



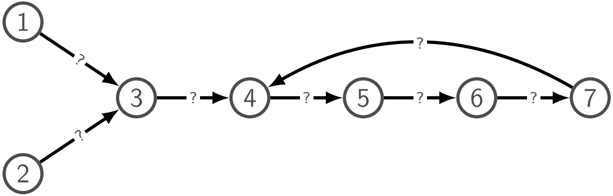
Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)
- Can be partitioned into 3 feasible solutions



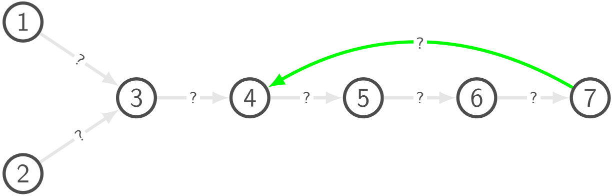
Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)
- Can be partitioned into 3 feasible solutions
- One has at least $1/3$ of the weight



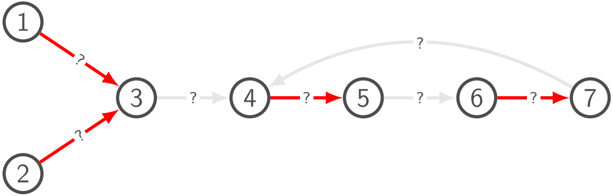
Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)
- Can be partitioned into 3 feasible solutions
- One has at least $1/3$ of the weight



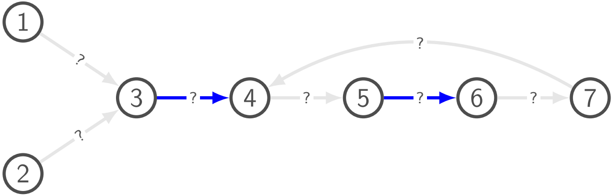
Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)
- Can be partitioned into 3 feasible solutions
- One has at least $1/3$ of the weight



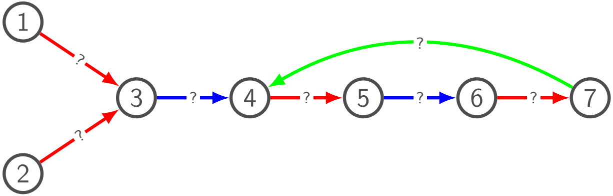
Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)
- Can be partitioned into 3 feasible solutions
- One has at least $1/3$ of the weight



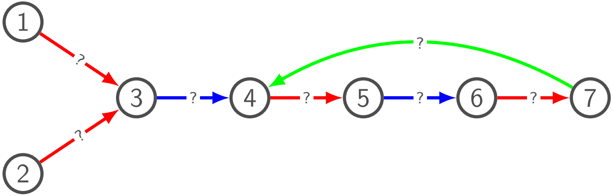
Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)
- Can be partitioned into 3 feasible solutions
- One has at least $1/3$ of the weight



Relaxed Matching (Observations)

- A relaxed matching is a pseudo-forest (out degree ≤ 1)
- Can be partitioned into 3 feasible solutions
- One has at least $1/3$ of the weight
- $1/3$-approximation algorithm to MCM (Relaxed Matching $\geq$ OPT)

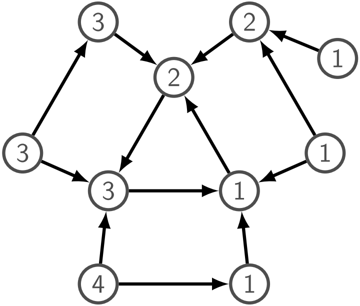


Maximum Carpool Matching

Unweighted

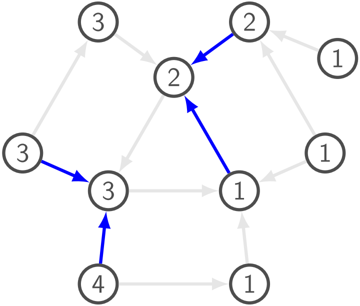
Unweighted MCM (1/2-approximation algorithm)

- Start with any maximal solution (bad in general)



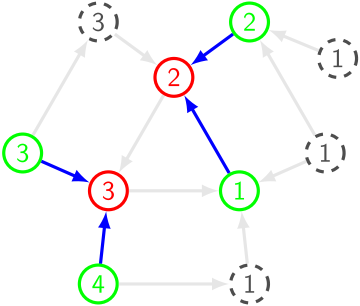
Unweighted MCM (1/2-approximation algorithm)

- Start with any maximal solution (bad in general)



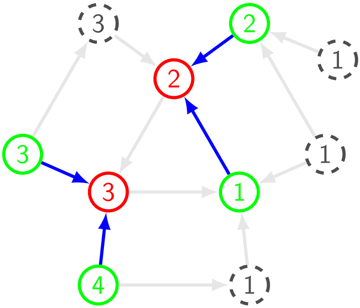
Unweighted MCM (1/2-approximation algorithm)

- Start with any maximal solution (bad in general)



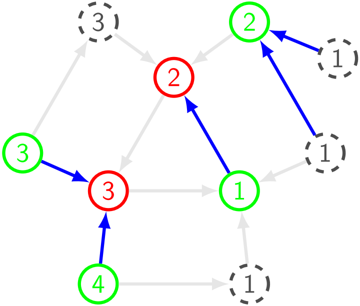
Unweighted MCM (1/2-approximation algorithm)

- Start with any maximal solution (bad in general)
- Improve $\iff$ a passenger has two incoming arcs from unmatched vertices



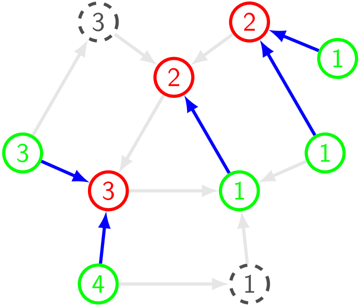
Unweighted MCM (1/2-approximation algorithm)

- Start with any maximal solution (bad in general)
- Improve $\iff$ a passenger has two incoming arcs from unmatched vertices



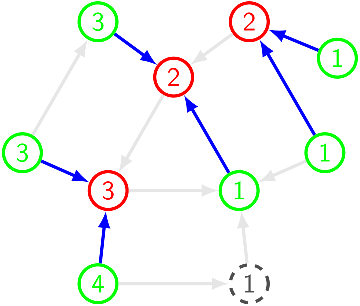
Unweighted MCM (1/2-approximation algorithm)

- Start with any maximal solution (bad in general)
- Improve $\iff$ a passenger has two incoming arcs from unmatched vertices
- Make it maximal



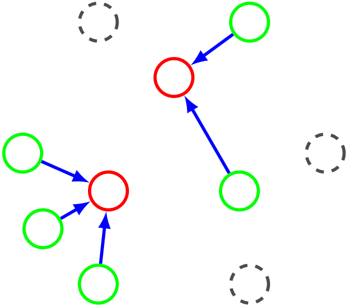
Unweighted MCM (1/2-approximation algorithm)

- Start with any maximal solution (bad in general)
- Improve $\iff$ a passenger has two incoming arcs from unmatched vertices
- Make it maximal



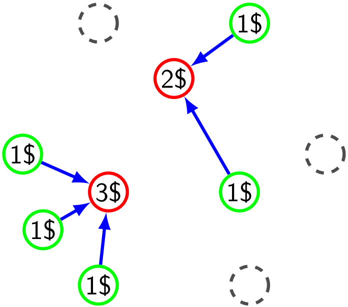
Unweighted MCM (Analysis)

- Load every arc with 2$



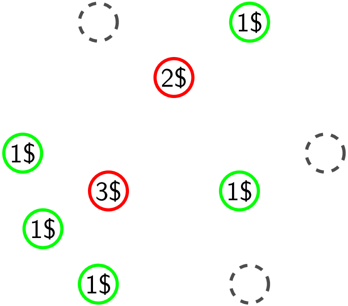
Unweighted MCM (Analysis)

- Load every arc with 2$



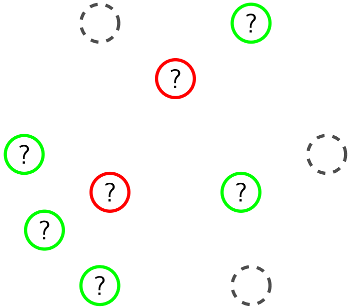
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution



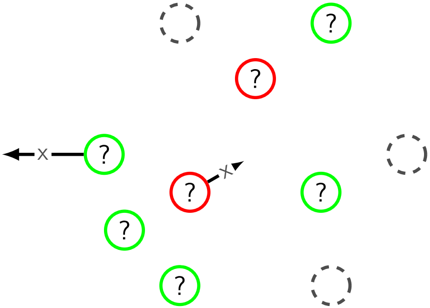
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)



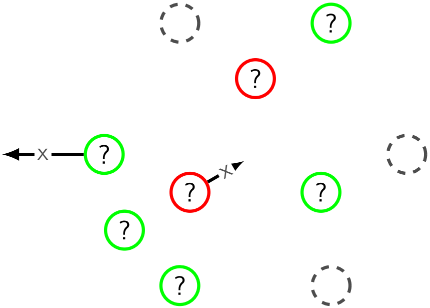
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)



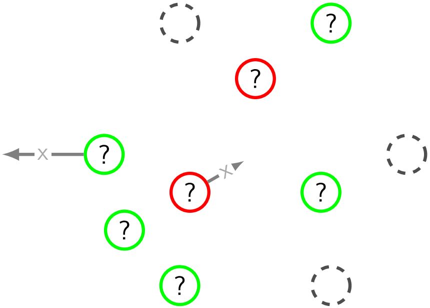
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)
- Left with outgoing arcs from unmatched vertices



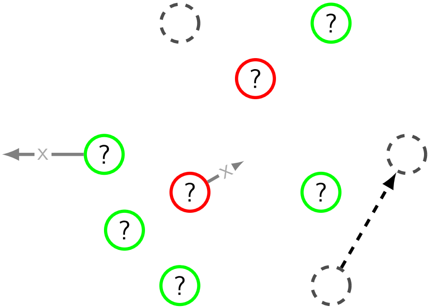
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)
- Left with outgoing arcs from unmatched vertices
- Can be paid by their destination



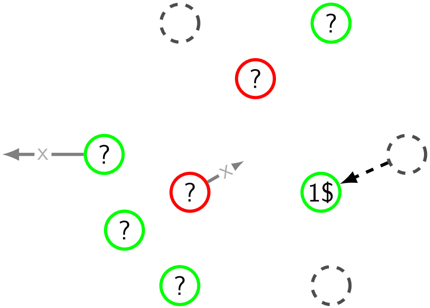
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)
- Left with outgoing arcs from unmatched vertices
- Can be paid by their destination



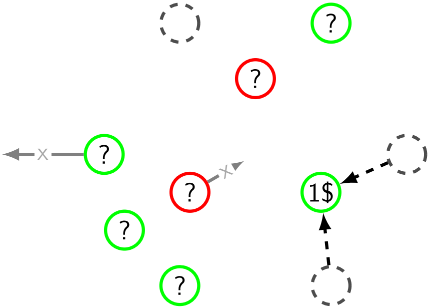
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)
- Left with outgoing arcs from unmatched vertices
- Can be paid by their destination



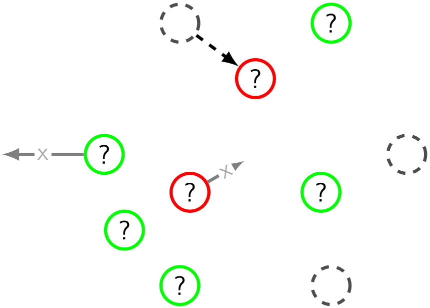
Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)
- Left with outgoing arcs from unmatched vertices
- Can be paid by their destination



Unweighted MCM (Analysis)

- Load every arc with 2$
- Consider another solution
- Each vertex pays (if it can) for its outgoing arc (if exists)
- Left with outgoing arcs from unmatched vertices
- Can be paid by their destination



Summary

- MSSF

Summary

- MSSF
  - APX-hard problem (10/11)

Summary

- MSSF
  - APX-hard problem (10/11)
  - No better than 1/2-approximation algorithm is known

Summary

- MSSF
  - APX-hard problem (10/11)
  - No better than 1/2-approximation algorithm is known
- MCM

Summary

- MSSF
  - APX-hard problem (10/11)
  - No better than 1/2-approximation algorithm is known
- MCM
  - Generalization of MSSF

Summary

- MSSF
  - APX-hard problem (10/11)
  - No better than 1/2-approximation algorithm is known
- MCM
  - Generalization of MSSF
  - First approximation algorithms:

Summary

- MSSF
  - APX-hard problem (10/11)
  - No better than 1/2-approximation algorithm is known
- MCM
  - Generalization of MSSF
  - First approximation algorithms:
    - 1/2-approximation to the unweighted variant

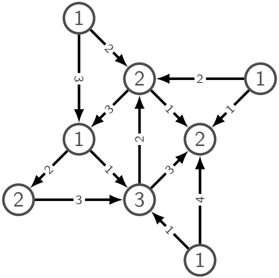
Summary

- MSSF
  - APX-hard problem (10/11)
  - No better than 1/2-approximation algorithm is known
- MCM
  - Generalization of MSSF
  - First approximation algorithms:
    - 1/2-approximation to the unweighted variant
    - 1/3-approximation to the general problem

Followups

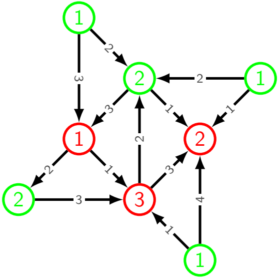
Fixed Matching

- P and D are given (tractable)



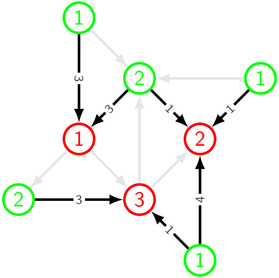
Fixed Matching

- P and D are given (tractable)



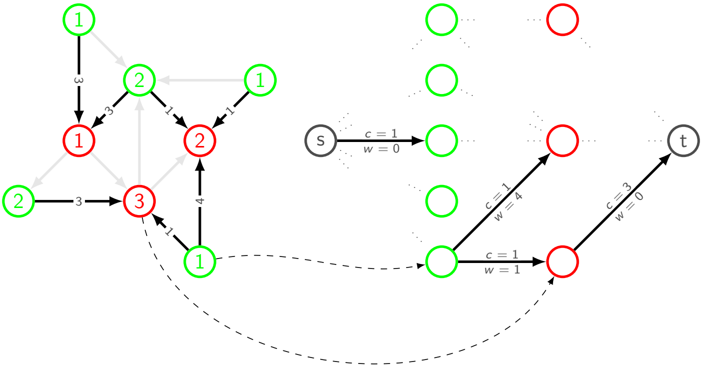
Fixed Matching

- *P* and *D* are given (tractable)



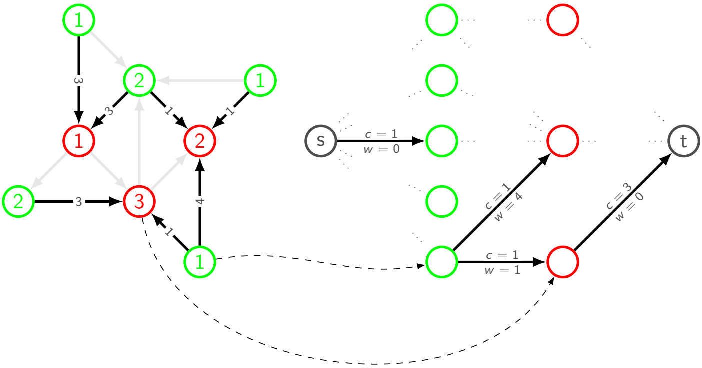
Fixed Matching

- P and D are given (tractable)



Fixed Matching

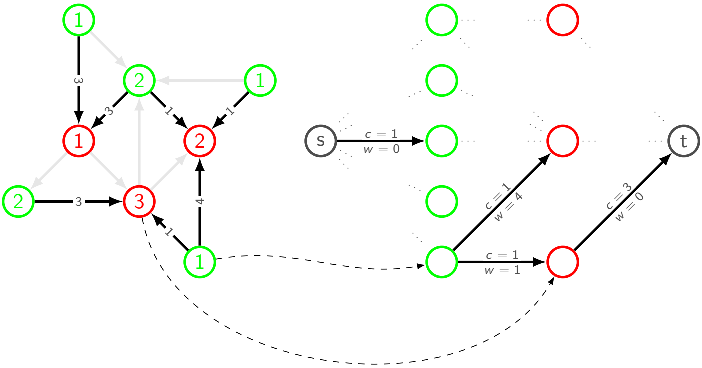
- P and D are given (tractable)



- $f(D) = OPT(D)$ is submodular (David Adjiashvili, personal communication)

Fixed Matching

- P and D are given (tractable)



- $f(D) = OPT(D)$ is submodular (David Adjiashvili, personal communication)
- $\frac{1}{2}$-approximation

Group Carpool

- Some people insists to carpool together

Group Carpool

- Some people insists to carpool together
- Not submodular anymore

Group Carpool

- Some people insists to carpool together
- Not submodular anymore
- Still admits a $\frac{1}{2}$-approximation

Open Question

Better than $1/2$-approximation?

Thank You !