Lec 9: Weighted Non-Bipartite Matching

Prof. Peng-Jun Wan

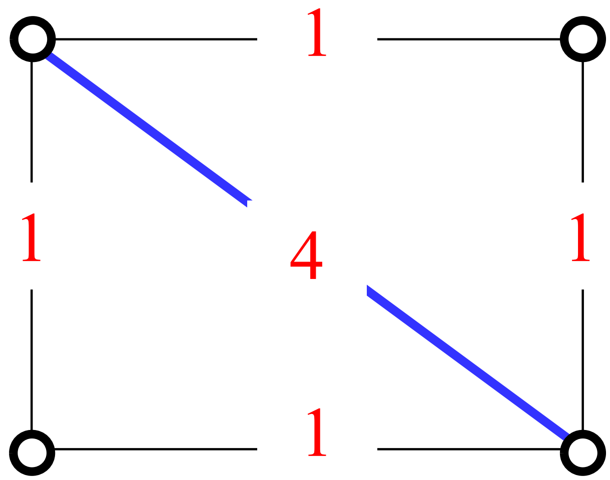
Outline

- Reduction to Min-Cost Perfect Matching
- Collective Bidding for Perfect Matching
- Blossom Algorithm
- Applications

0. Reduction to Min-Cost Perfect Matching

Recap: Weighted matching

- Input: an edge-weighted graph $G = (V, E;\ w)$
- Objective: find a matching M with maximum weight $w(M)$.
  - we may assume all weights are positive



Minimum-cost perfect matching

- Input: $G = (V, E; c)$ with edge costs and having a perfect matching
- Objective: find a perfect matching M with minimum cost $c(M)$.
  - we may, and shall, assume all costs are positive

Max-Weight Perfect Matching ⇒ Min–Cost Perfect Matching

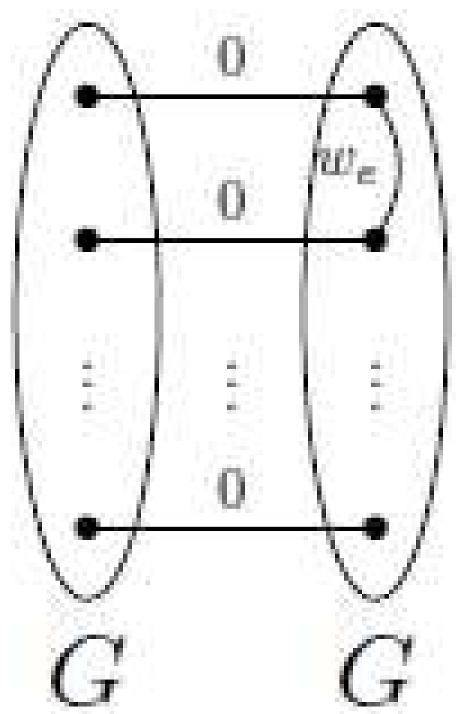
- flipping the weights

Reduction to perfect matching

Max-Weight Matching ⇒ Max–Weight Perfect Matching

replication:

- take two copies of G
- connect each node with its copy by an edge of zero weight



1. Collective Bidding for Perfect Matching

Odd-set cuts

- $\delta(v)$: edges incident to v
- $\delta(U)$: edges with exactly one end in U (i.e. leaving U), U-cut

For any perfect matching M

- $|M \cap \delta(v)| = 1$, for any $v \in V$;
- $|M \cap \delta(U)| \geq 1$, for any odd subset $U \subset V$ with $|U| \geq 3$.

Frugal bidding

(Ω, π)

- Ω: all singletons and a nested family of odd subsets (groups) $U \subset V$

- π: bids (payments) offered by members of Ω s.t.
  - for any non-singleton $U \in \Omega$, $\pi(U) \geq 0$

- frugalness: $\sum_{U \in \Omega : \mathrm{e} \in \delta(U)} \pi(U) \leq c(e)$ for each edge e
  - e can only collect the bids by groups to which it creates outside connection.

Participation in matching

Residual edge cost c_π: $c_\pi(e) = c(e) - \sum_{U \in \Omega : e \in \delta(U)} \pi(U) \geq 0$

- e is tight w.r.t. (Ω, π) if $c_\pi(e) = 0$ // collected payment covers cost
- **only tight edges are willing to join the matching** M

Selection of M: M is tight and has **no competing for bids** i.e.

$$|M \cap \delta(U)| \leq 1 \text{ for each } U \in \Omega$$

Weak duality

between a frugal bidding (Ω, π) and an arbitrary perfect matching M

Thm. $c(M) \geq \sum_{U \in \Omega} \pi(U)$.

Lemma. $c(M) - c_\pi(M) \geq \sum_{U \in \Omega} \pi(U)$ and equality holds iff $|M \cap \delta(U)| = 1$ for each non-singleton $U \in \Omega$ with $\pi(U) > 0$

Pf.

$$
\begin{aligned}
c(M) - c_\pi(M) &= \sum_{e \in M} [c(e) - c_\pi(e)] \\
&= \sum_{e \in M} \sum_{U \in \Omega : e \in \delta(U)} \pi(U) \\
&= \sum_{U \in \Omega} \pi(U) |M \cap \delta(U)| \\
&= \sum_{v \in V} \pi(\{v\}) + \sum_{U \in \Omega : |U| \geq 3} \pi(U) |M \cap \delta(U)| \\
&\geq \sum_{v \in V} \pi(\{v\}) + \sum_{U \in \Omega : |U| \geq 3} \pi(U) \\
&= \sum_{U \in \Omega} \pi(U)
\end{aligned}
$$

Achieving a min-cost perfect matching

Theorem. If a frugal bidding (Ω, π) admits a perfect matching M which is tight and competing-free, them M is a min-cost perfect matching.

Pf. $c_\pi(M) = 0$ and $c(M) = \sum_{U \in \Omega} \pi(U)$.

The general bidding process

- Initially Ω consists of all singletons offering 0 payment, and $M = \emptyset$
- Iterative bidding process:
  - Raise a frugal bidding in net amount to attract more edges
  - Grow a tight and competing-free matching (via swapping)
- In the end, the matching is perfect

3. Blossom Algorithm

Overview of the bidding evolution

- At any time, only maximal (outermost) members of Ω reset bids
  - $\Omega^{\max}$: the collection of maximal members // a partition of V
  - $G/\Omega^{\max}$: a multigraph with each vertex being a member of Ω^{max}

- New members of Ω are generated from tight blossoms on maximal members
  - keeps a tight Hamiltonian circuit on its children
  - initially offers 0 payment

- Maximal members of Ω may be dismantled after their bids drop to 0
  - effectively deshrinkg the outermost blossoms

Overview of the matching evolution

- Only a tight and competing-free matching M on $G/\Omega^{\max}$ is maintained.
- A tight M-alternating forest F on $G/\Omega^{\max}$ is maintained for finding a tight M-augmenting path
  - Pool of candidates for joining M
- M is always augmented along a tight M-augmenting path
- After dismantling a maximal member of $\Omega^{\max}$, both M and F are lifted by using the inner Hamiltonian chain
  - Blossoms may appear at odd level
- After a perfect matching M on $G/\Omega^{\max}$ is discovered, M is lifted to a perfect matching and stops.

Different need for blossom deshrinking

- Needed when dismantling a maximal member of Ω.
- No augmenting path lift, and hence no deshrinking in augmenting a matching
- Needed in the final matching lift

Blossom Algorithm

initialization: $M \leftarrow \emptyset, F \leftarrow \emptyset, \Omega \leftarrow \{\{v\}: v \in V\}$, and $\pi(\{v\}) \leftarrow 0\ \forall v \in V$.

while M is not a perfect matching on $G/\Omega^{\max}$

Reset bids;

Case 0: Deshrink a blossom;

Case 1: Extend the forest;

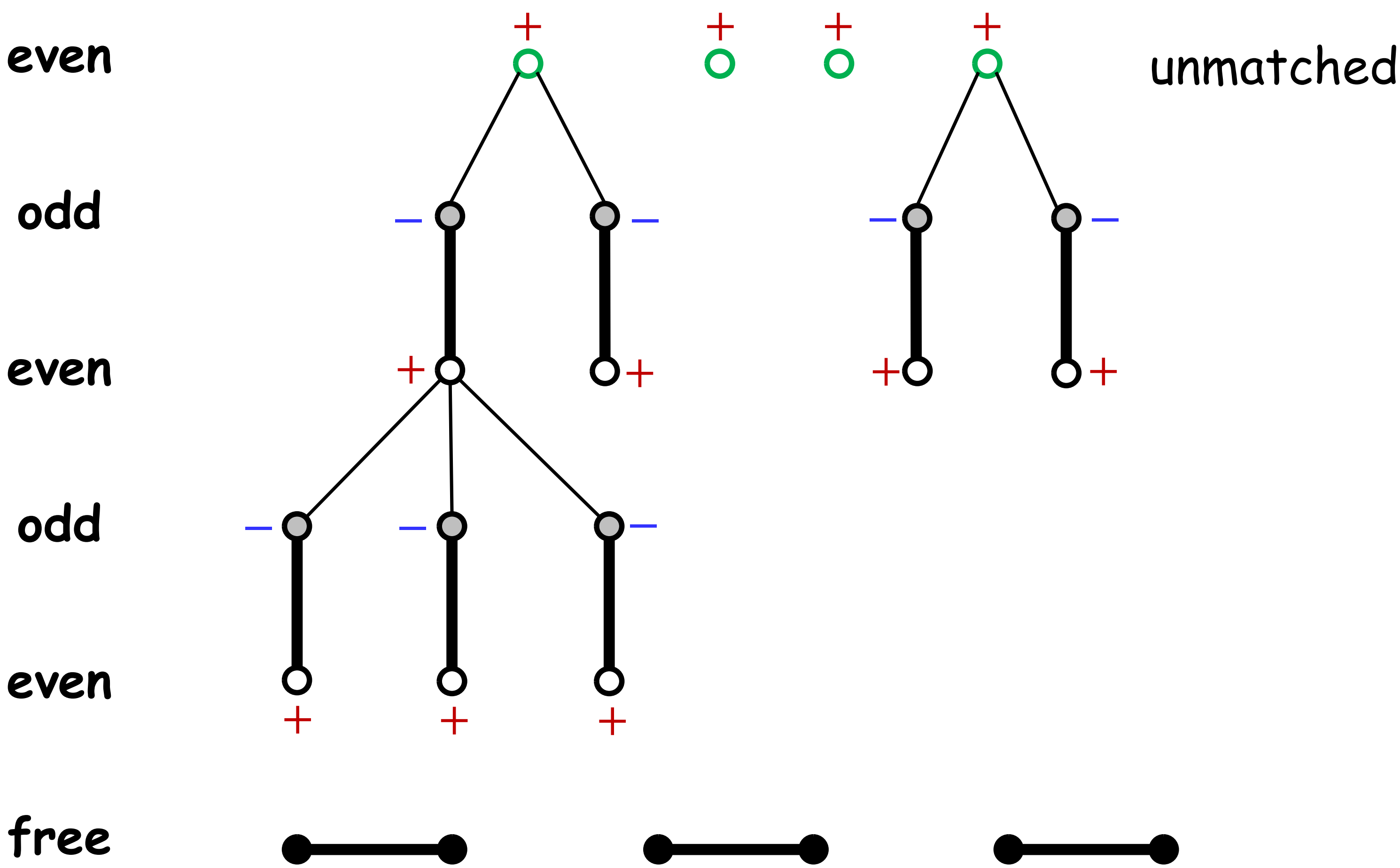
Case 2: Shrink a blossom;

Case 3: Augment the matching;

lift M to a perfect matching in G, and return M.

Resetting bids

- Unmatched vertices need to pay more to get matched .
- To preserve tightness of F,
  - for each $U \in even(F)$, $\pi(U) \leftarrow \pi(U) + \varepsilon$
  - for each $U \in odd(F)$, $\pi(U) \leftarrow \pi(U) - \varepsilon$



Greedy resetting amount

Maintain non-negativity of bids by blossoms and frugalness

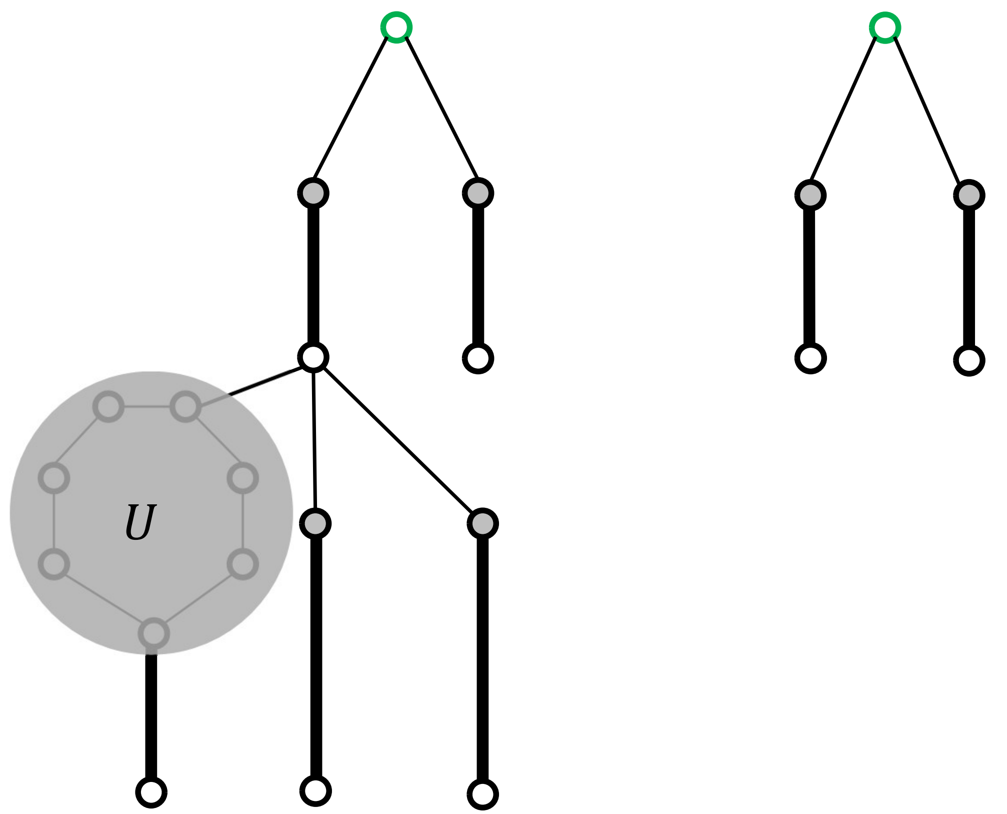
$\varepsilon_0 \leftarrow \min\{\pi(U): U \in odd(F), |U| \geq 3\}$
$\varepsilon_1 \leftarrow \min\{c_\pi(e):\ e \text{ is between } even(F) \text{ and } free(F)\}$
$\varepsilon_2 \leftarrow (1/2)\min\{c_\pi(e): \text{both ends of } e \text{ belong to } even(F)\}$
$\varepsilon \leftarrow \min\{\varepsilon_0, \varepsilon_1, \varepsilon_2\}$ // could be 0

After resetting,

- either some blossom $U \in odd(F)$ with $|U| \geq 3$ drops its bid to 0
- or some edge e between $even(F)$ and $even(F) \cup free(F)$ is tight

Case 0: deshrink a blossom

Case 0: 0-bid blossom $U \in odd(F)$ with $|U| \geq 3$.



Case 0: deshrink a blossom

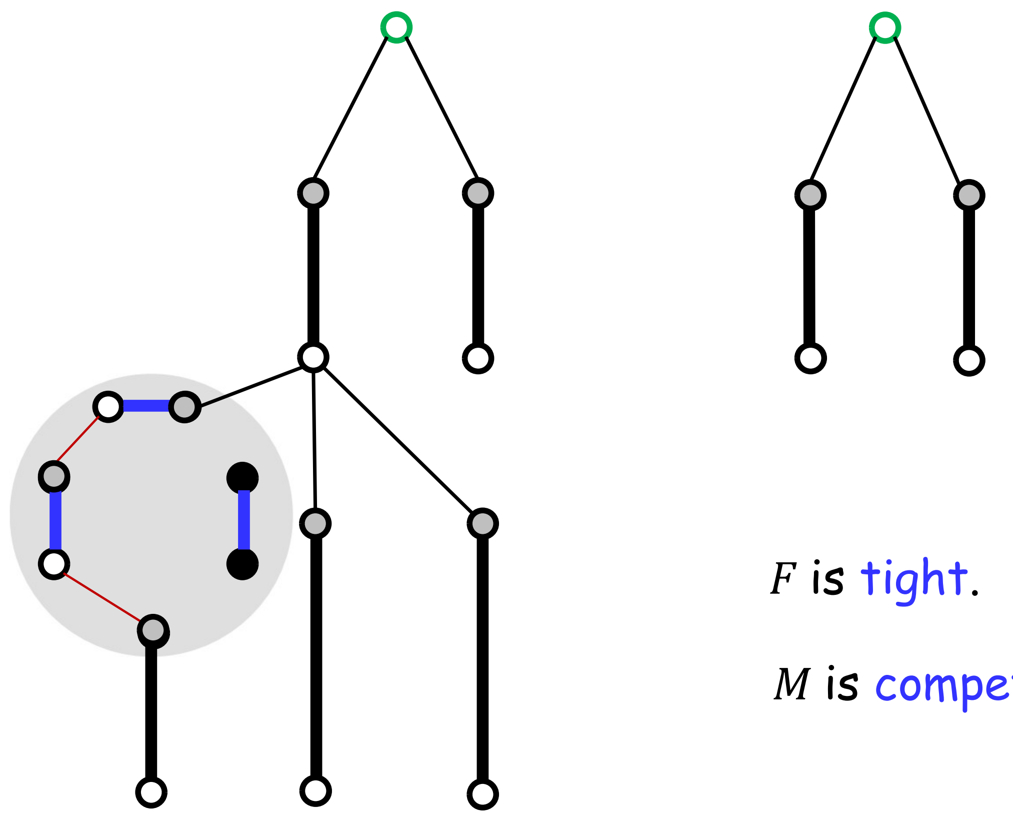
$\Omega \leftarrow \Omega - \{U\}$;

$N \leftarrow$ the matching in C_U covering all vertices in U missed by M;

$M \leftarrow M \cup N$;

$P \leftarrow$ the even path in C_U connecting the two edges in F incident to U;

$F \leftarrow F \cup N \cup P$;



F is tight.

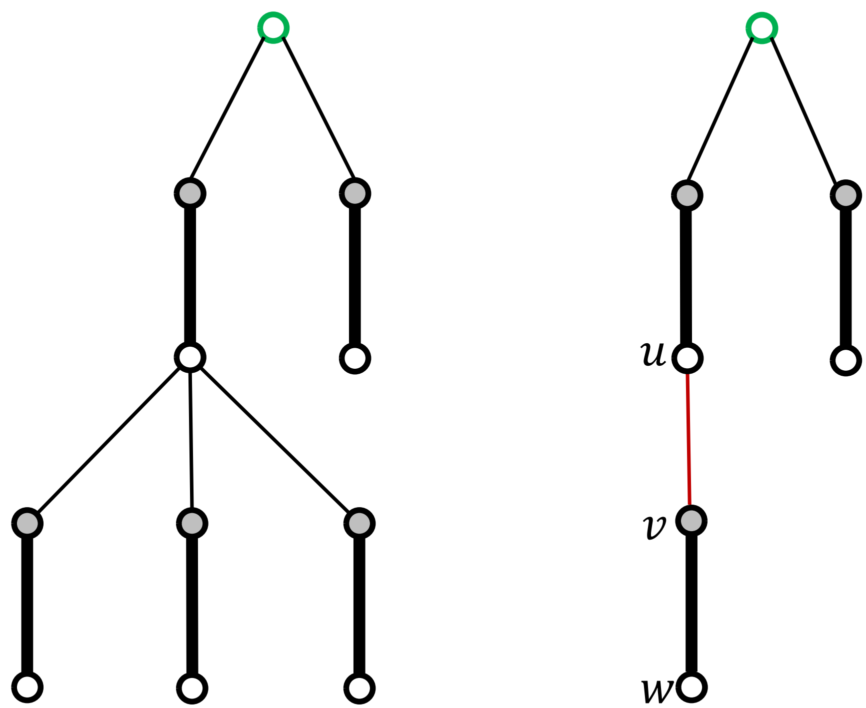
M is competing-free.

Case 1: extend the forest

Case 1: new tight e between $even(F)$ and $free(F)$

$F \leftarrow F \cup \{e\}$;

F is tight.



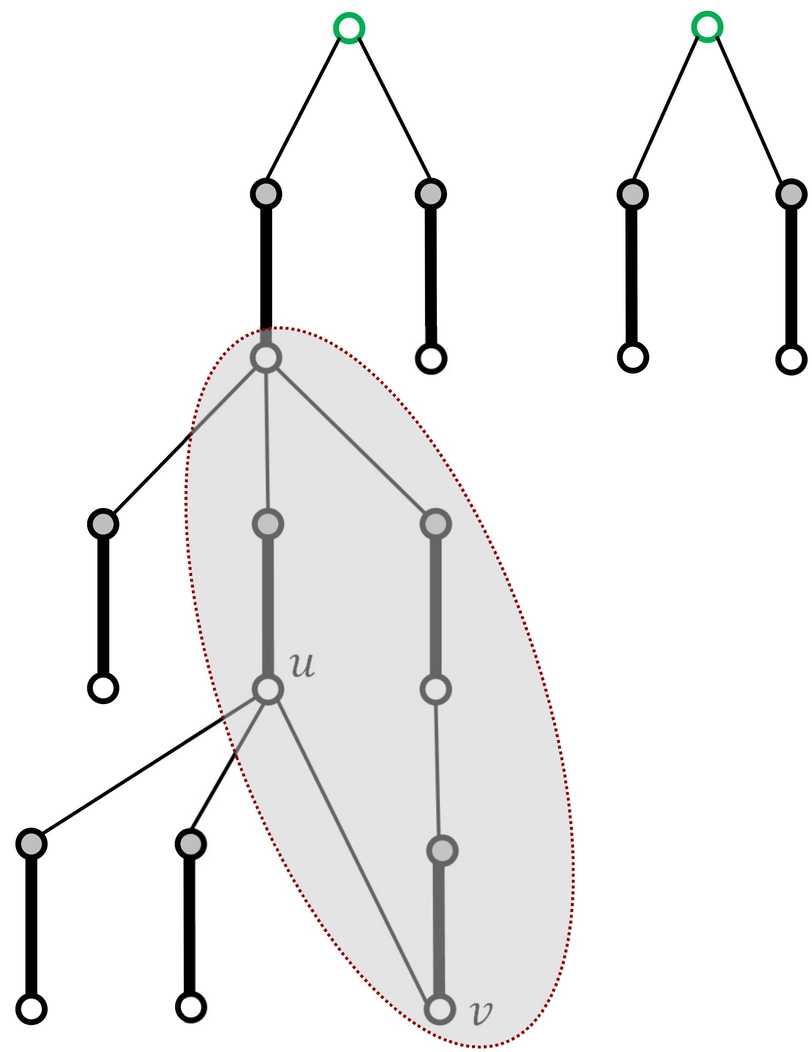
Case 2: shrink a blossom

Case 2: new tight e with both ends in $even(F)$, and in the same tree

$B \leftarrow$ the blossom in $F \cup \{e\}$;
$U \leftarrow$ the vertex set of B; $C_U \leftarrow B$;

$\Omega \leftarrow \Omega \cup \{U\}$; $\pi(U) \leftarrow 0$;
$F \leftarrow F/U$; $M \leftarrow M/U$;

The new $U \in even(F)$.



C_U and F are tight.

M is competing-free.

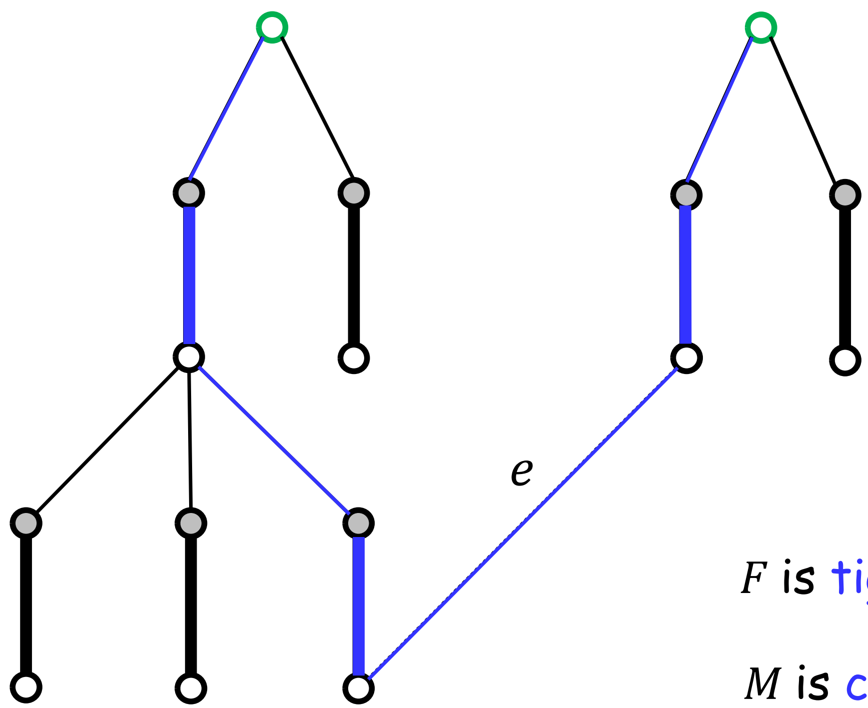
Case 3. augmenting the matching

Case 3: new tight e with both ends in $even(F)$ but in different trees

$P \leftarrow$ the M-augmenting path in $F \cup \{e\}$;

$M \leftarrow M \oplus P$

$F \leftarrow M$



F is tight.

M is competing-free.

Analysis

Correctness: M is perfect, tight, and competing-free

# of iterations: $O(n^2)$:

- # of augmenting iterations $\leq n/2$
- # of iterations between two successive augmenting iterations $\leq 2n$

Implementation:

- With simple data structure, the running time is $O(n^3)$
- Fastest-known time [Gabow 1990]: $O(n(m + n\log n))$

Number of augmentations (Case 3)

Lemma. The number of augmenting iterations $\leq n/2$

Pf. evolution of the potential $|\Omega^{\max}| - 2|M|$ (# of vertices missed by M):

- initially n;
- decreases by 2 in each augmenting iteration;
- no change in each of the other three iterations.

Number of iterations between successive augmentations

Lemma. $\leq 2n$ iterations between two successive augmenting iterations

Pf. evolution of the potential $2|V_{even}| + |odd(F)| + |free(F)|$ where

$V_{even} :=$ {nodes $v \in V$ shrunk to a vertex of $G/\Omega^{\max}$ in $even(F)$}

- at most $2n$

$$\begin{aligned} &2|V_{even}| + |odd(F)| + |free(F)| \\ &\leq 2|V_{even}| + |V| - |V_{even}| \leq |V_{even}| + |V| \leq 2n \end{aligned}$$

- strictly increases with iterations between augmenting iterations
  - straightforward exercise.

Strong duality

$\min c(M)$
s.t. M is a perfect matching

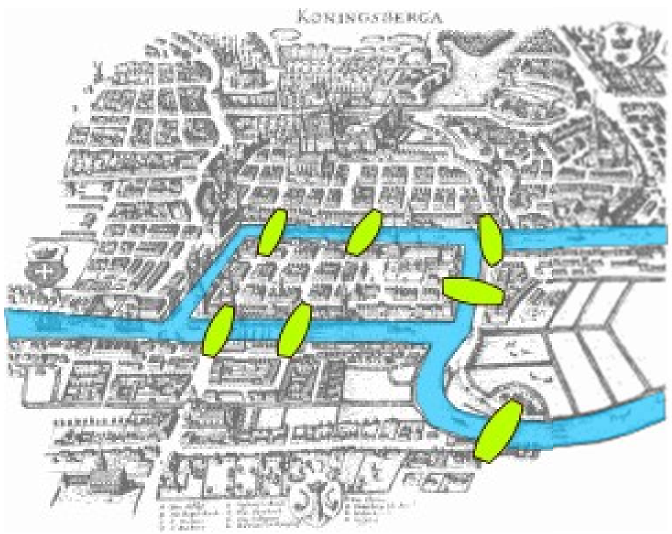
Ω: the family of all odd subsets of V

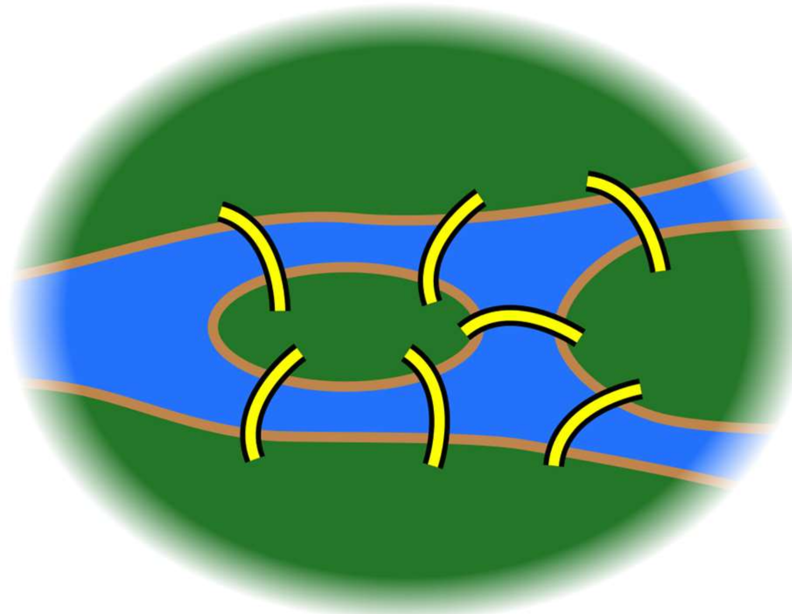
$\max \sum_{U \in \Omega} \pi(U)$
s.t. $\pi(U) \geq 0$, for all non-singleton $U \in \Omega$
$\quad \sum_{U \in \Omega : e \in \delta(U)} \pi(U) \leq c(e)$, for all edge e

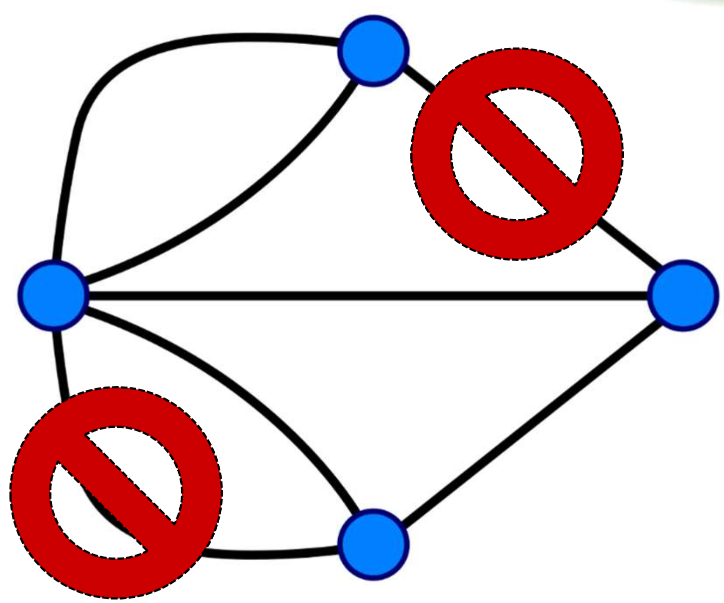
Thm. The two problems have the same value.

4. Applications

Seven Bridges of Königsberg



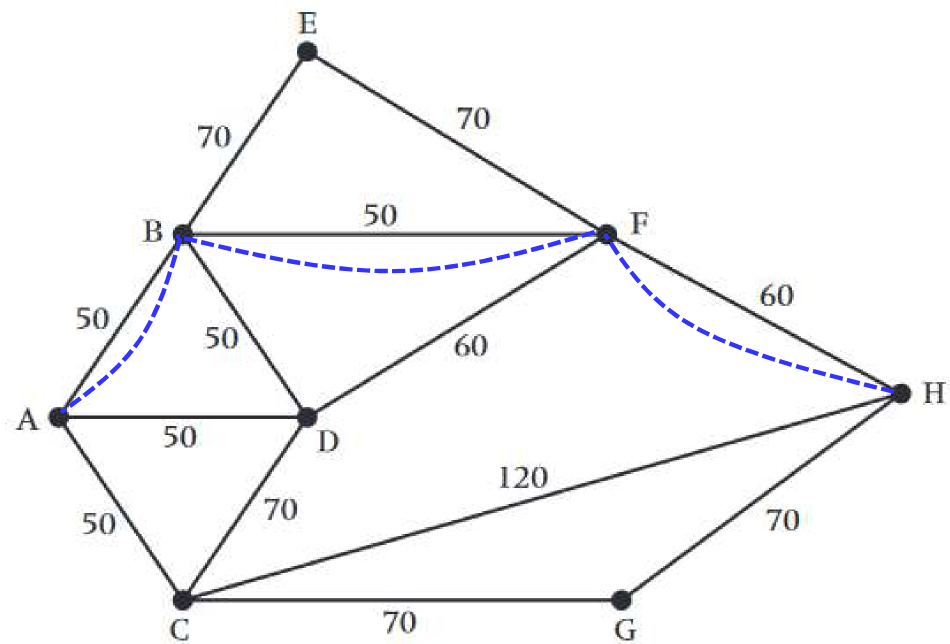




The Chinese Postman Problem

$G = (V, E; \ell)$: A graph with non-negative length function ℓ

Chinese postman tour: A closed walk C visiting each edge of G at least once.

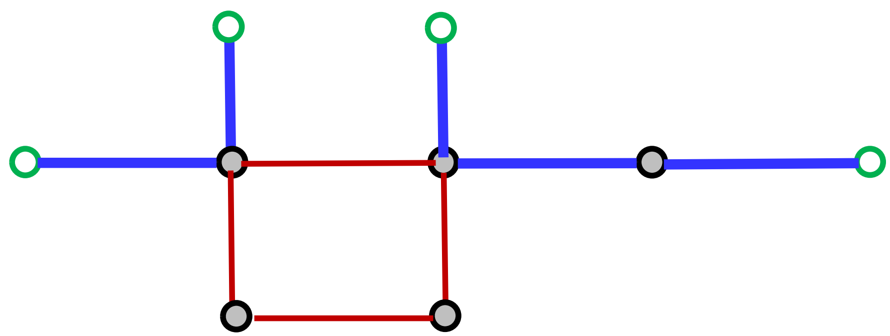


The Chinese postman problem: find a shortest Chinese postman tour C.
first studied by Guan [1960], and named by Edmonds [1965]

T-join

Def. Given $G = (V, E)$ and $T \subseteq V$, a subset $J \subseteq E$ is a T-join if T is the set of nodes with odd degree in the graph (V, J), i.e.,

$$T = \{v \in V : |\delta(v) \cap J| \text{ is odd}\}.$$



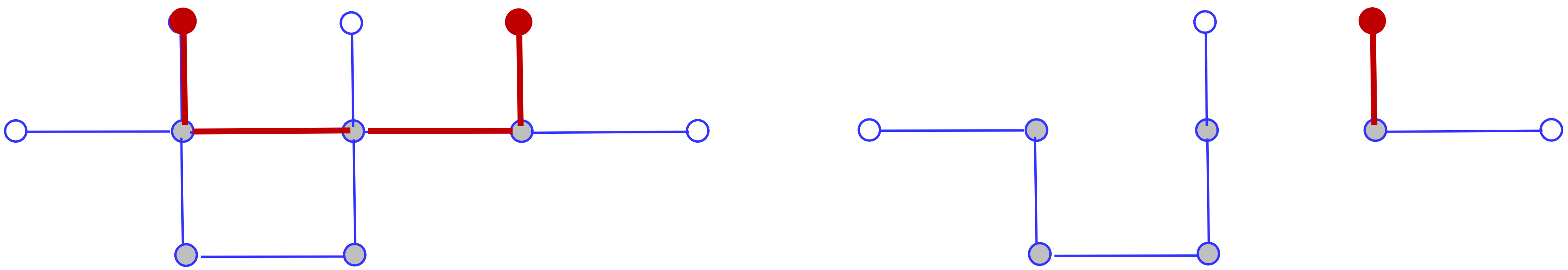
Remark: $|T|$ must be even.

Lemma. J is a T-join iff J can be decomposed into $|T|/2$ paths connecting disjoint pairs of nodes in T and some circuits.

Examples: $\emptyset$-join is a cycle; $\{s,t\}$-join is an $s-t$ path + circuits

Symmetric difference of T-joins

Lemma. J_1 is a T_1-join and J_2 is a T_2-join $\Rightarrow J_1 \oplus J_2$ is a $T_1 \oplus T_2$-join.



Pf. For any $v \in V$:

$$|\delta(v) \cap (J_1 \oplus J_2)| = |(\delta(v) \cap J_1) \oplus (\delta(v) \cap J_2)| \text{ is odd}$$

$\Leftrightarrow |\delta(v) \cap J_1|$ and $|\delta(v) \cap J_2|$ have different parity

$\Leftrightarrow v \in T_1 \oplus T_2$

Shortest T-join

Def. Given $G = (V, E; \ell)$ with edge length ℓ and $T \subseteq V$, find a T-join with minimum total length

[Edmonds 1965]: Reduction to min-cost perfect matching

Non-negative edge-length

P_{st}: a shortest s - t path in G for each pair $\{s, t\}$ in T; its length is $c(st)$
K_T: the complete graph on T with edge cost c
M: a min-cost perfect matching in K_T

Claim: The symmetric difference of the paths P_{st} for $st \in M$ is a shortest T-join in G.

Remark: Simply take union if all edges have positive length [as disjoint]

Arbitrary edge-length

N: the set of edges in E with negative length
U: the set of vertices incident to an odd number of edges in N
Then, N is a U-join.

Lem. If J is a $T \oplus U$-join, then $J \oplus N$ is a T-join and

$$\ell(J \oplus N) = |\ell|(J) + \ell(N).$$

Pf. $(T \oplus U) \oplus U = T$ and

$$\ell(J \oplus N) = \ell(J \setminus N) - \ell(N \cap J) + \ell(N)$$
$$= |\ell|(J \setminus N) + |\ell|(N \cap J) + \ell(N) = |\ell|(J) + c(N)$$

Arbitrary edge-length

Thm. If J is a shortest $T \oplus U$-join w.r.t. $|\ell|$, then $J \oplus N$ is a shortest T-join w.r.t. ℓ.

Pf. For any T-join J', $J' \oplus N$ is a $T \oplus U$-join and

$$\ell(J') = \ell(J' \oplus N \oplus N) = |\ell|(J' \oplus N) + \ell(N) \geq |\ell|(J) + \ell(N) = \ell(J \oplus N)$$

Shortest path in undirected graphs

$G = (V, E; \ell)$: a graph with length function ℓ
$s, t \in V$

Thm. G has a negative circuit iff G has a negative $\emptyset$-join.

Thm. Suppose G has no negative circuit, and J is a shortest $\{s, t\}$-join. Partition J into an s - t path P and circuits. Then, P is a shortest s - t path.

Minimum-mean circuit in undirected graphs

Fact. MMCs are invariant with uniform length changes
Assumption. G has a negative circuit

```
repeat
    find a shortest ∅-join J;
    if ℓ(J) < 0 add −ℓ(J)/|J| to all edge-lengths;
    if ℓ(J) = 0, return a circuit in J.
```

Claim. $|J|$ strictly decreases in all but the last iterations.

Pf. Two subsequent iterations: $\ell, J; \ell', J'$ with $\ell'(J') < 0$

$$0 > \ell'(J') = \ell(J') - \frac{\ell(J)}{|J|}|J'| \geq \ell(J) - \frac{\ell(J)}{|J|}|J'| = \frac{\ell(J)}{|J|}(|J| - |J'|)$$