COMP 355
Advanced Algorithms

Extensions of Network Flows
Section 7.7- 7.12(KT)

Runtimes of Various Max-Flow Algorithms

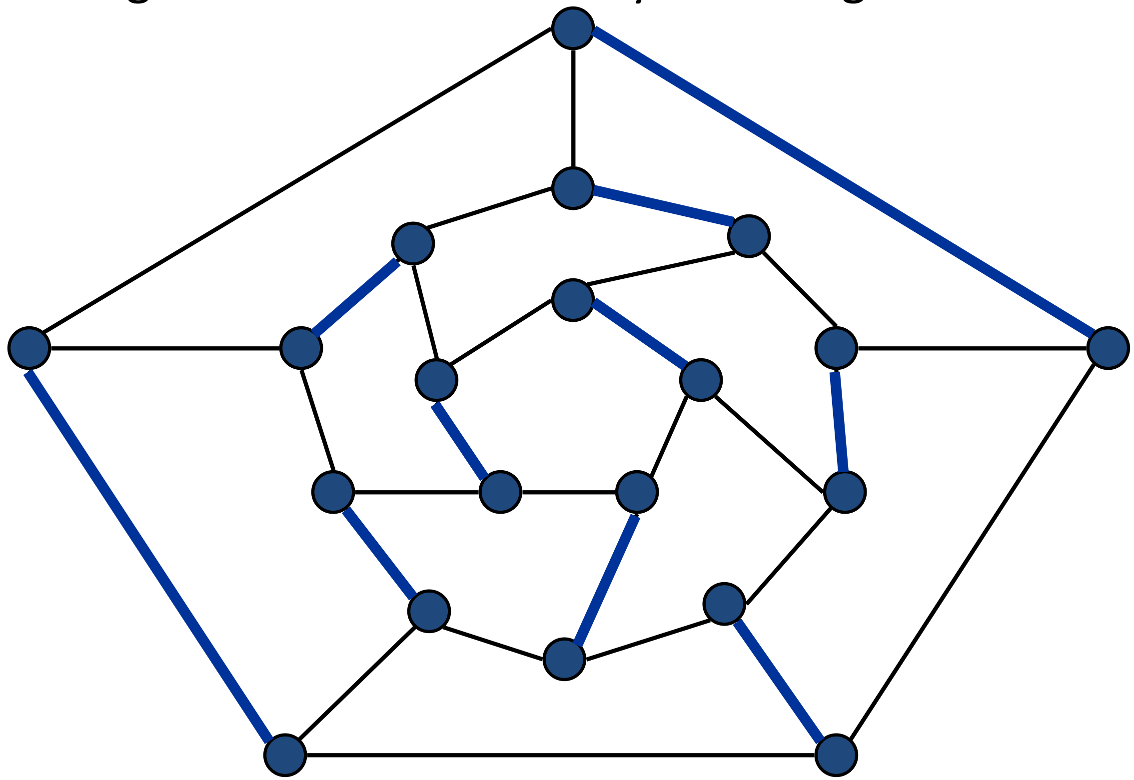
Algorithm	Year	Time	Notes
Ford-Fulkerson	1956	$O(mC)$	
Gabow	1985	$O(nm \log C)$	Scaling
Edmonds-Karp	1972	$O(nm^2)$	Ford-Fulkerson + augment shortest paths
Dinic	1970	$O(n^2 m)$	Blocking flows in a layered graph
Dinic + Tarjan	1983	$O(nm \log n)$	Dinic + better data structures
Preflow push	1986	$O(nm \log(n^2/m))$	Goldberg and Tarjan
King, Rao, Tarjan	1994	$O(mn \log_{\frac{m}{n \log n}} n)$	$O(nm)$ if $m = O(n^{1+\varepsilon})$
Orlin + KRT	2013	$O(nm)$	Orlin: $O(nm)$ time for $m \leq O(n^{16/15-\varepsilon})$ KRT: $O(nm)$ for $m > n^{1+\varepsilon}$

Current state of the art is $O(nm)$

Matching

Matching.

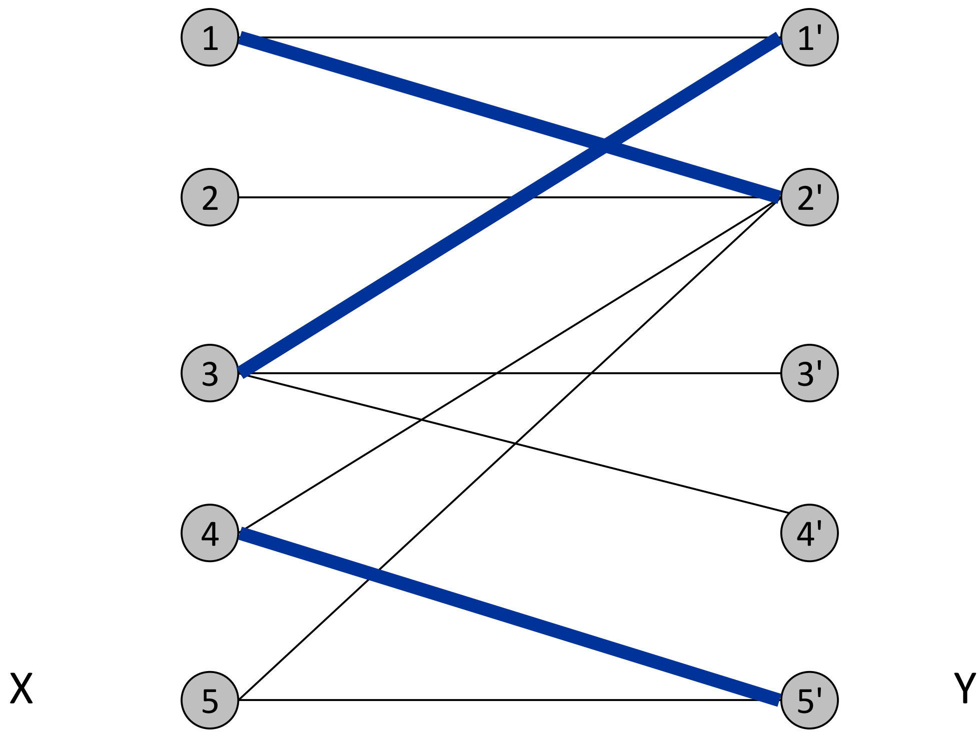
- Input: undirected graph $G = (V, E)$.
- $M \subseteq E$ is a matching if each node appears in at most one edge in M.
- Max matching: find a max cardinality matching.



Bipartite Matching

Bipartite matching.

- Input: undirected, bipartite graph $G = (X \cup Y, E)$.
- $M \subseteq E$ is a matching if each node appears in at most one edge in M.
- Max matching: find a max cardinality matching.



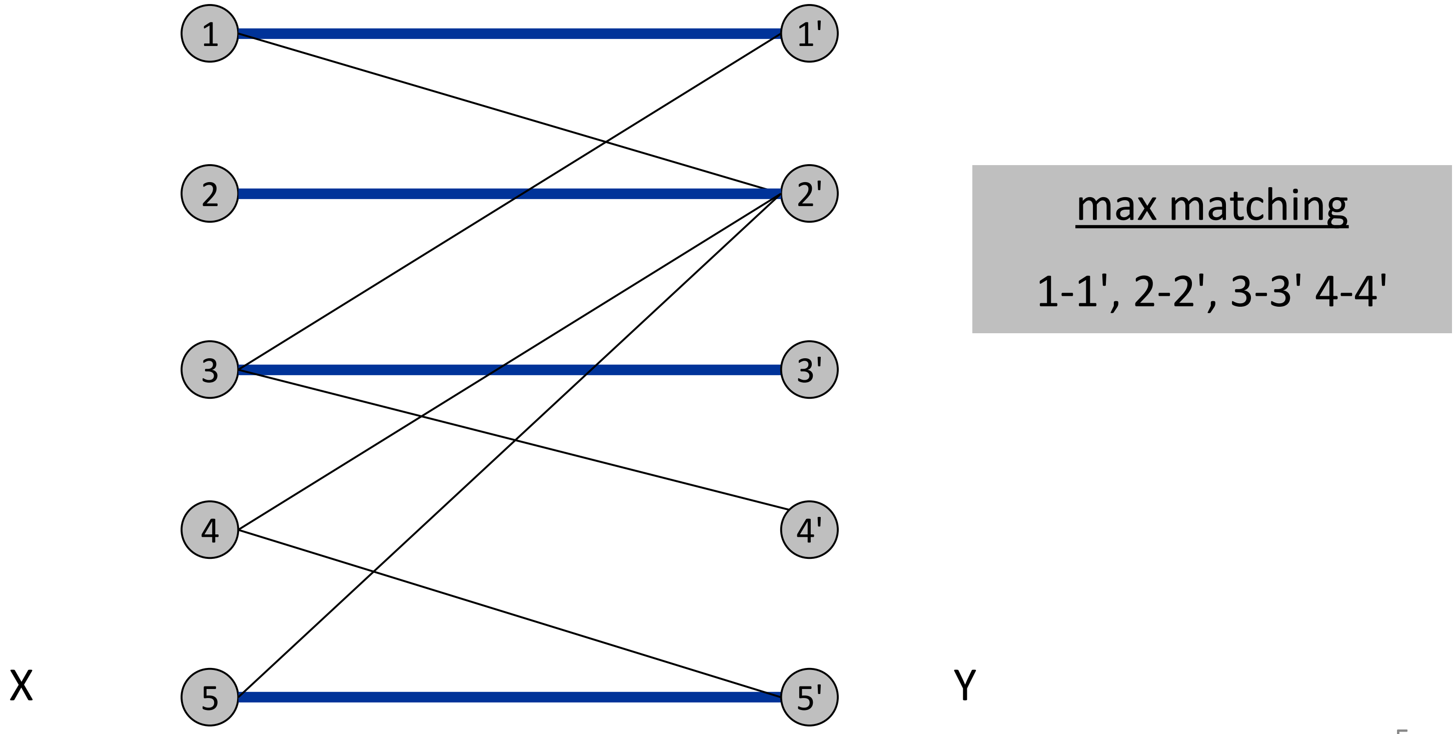
<u>matching</u>

1-2', 3-1', 4-5'

Bipartite Matching

Bipartite matching.

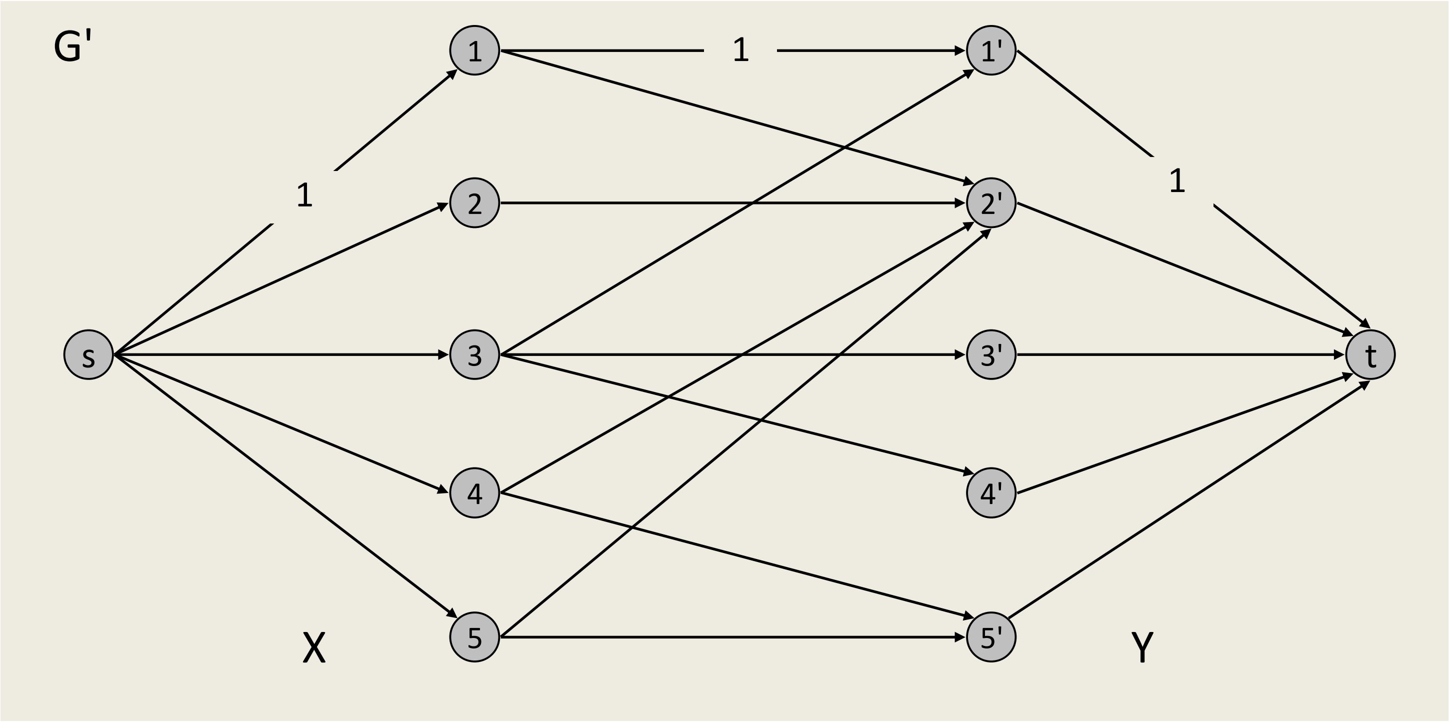
- Input: undirected, bipartite graph G = (X $\cup$ Y, E).
- M $\subseteq$ E is a matching if each node appears in at most edge in M.
- Max matching: find a max cardinality matching.



Bipartite Matching

Max flow formulation.

- Create digraph G' = (X ∪ Y ∪ {s, t}, E').
- Direct all edges from X to Y, and assign unit (or infinite) capacity.
- Add source s, and unit capacity edges from s to each node in X.
- Add sink t, and unit capacity edges from each node in Y to t.

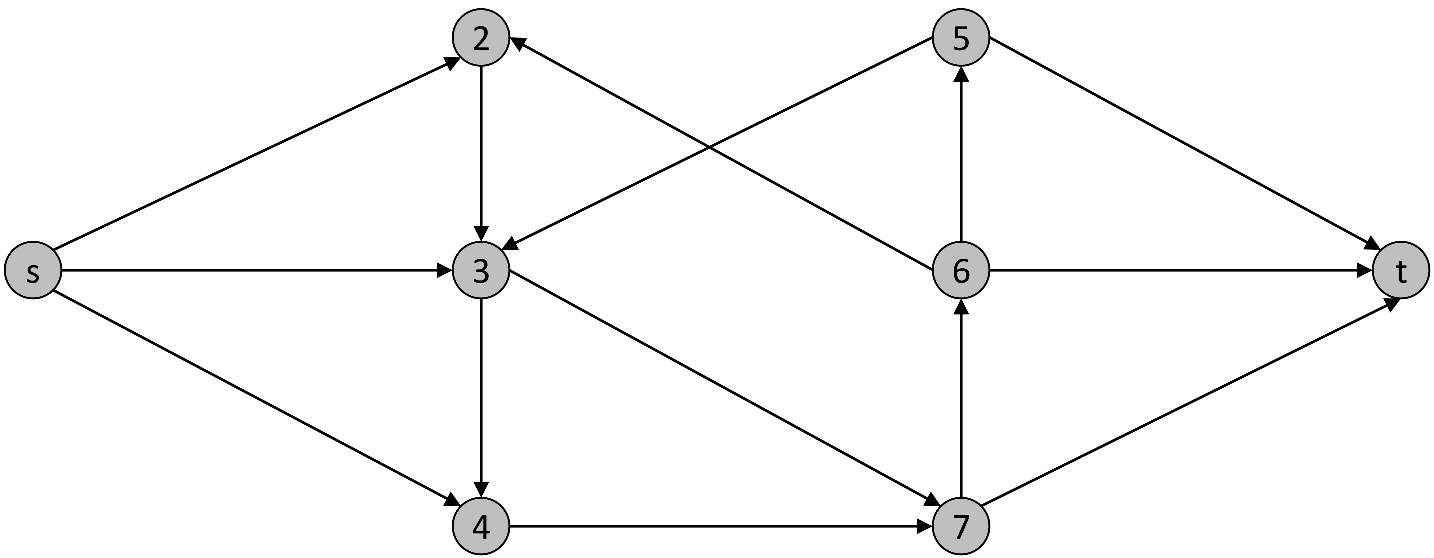


Edge Disjoint Paths

Disjoint path problem. Given a digraph G = (V, E) and two nodes s and t, find the max number of edge-disjoint s-t paths.

Def. Two paths are edge-disjoint if they have no edge in common.

Ex: communication networks.

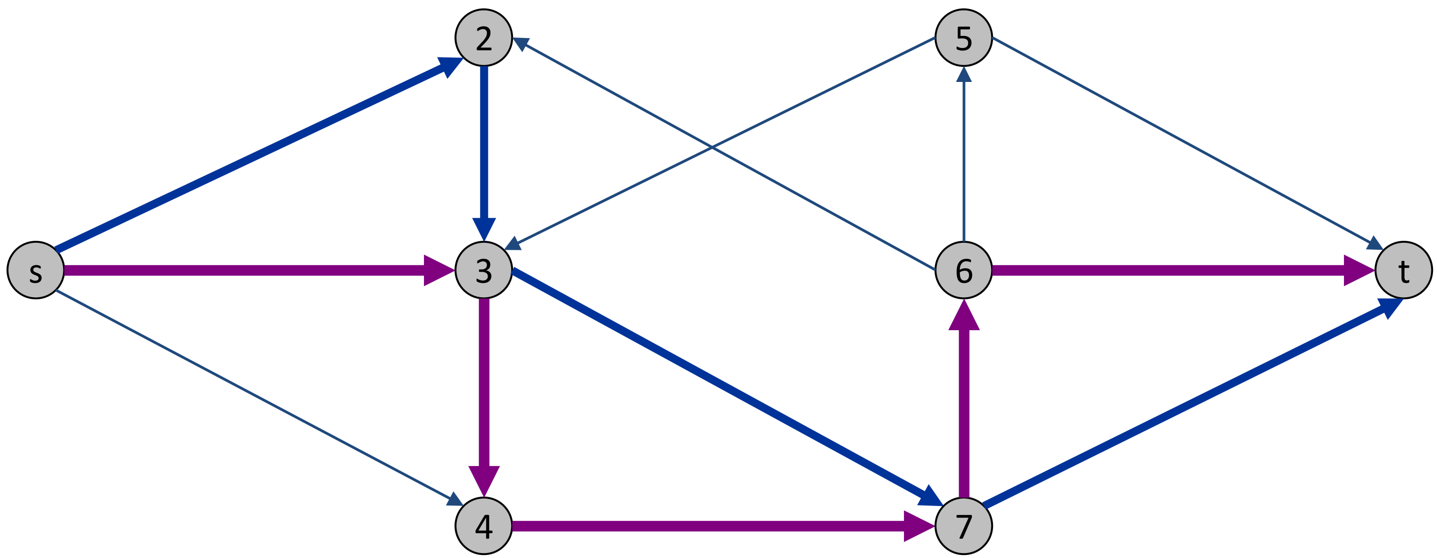


Edge Disjoint Paths

Disjoint path problem. Given a digraph G = (V, E) and two nodes s and t, find the max number of edge-disjoint s-t paths.

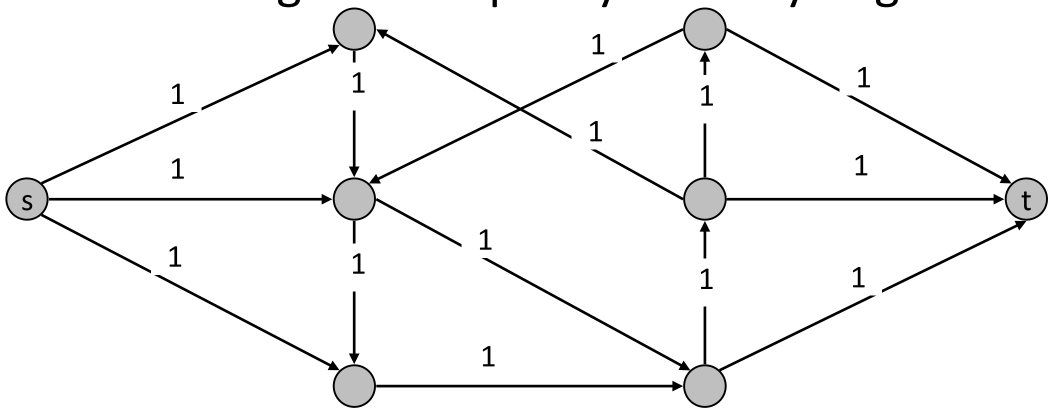
Def. Two paths are edge-disjoint if they have no edge in common.

Ex: communication networks.



Edge Disjoint Paths

Max flow formulation: assign unit capacity to every edge.



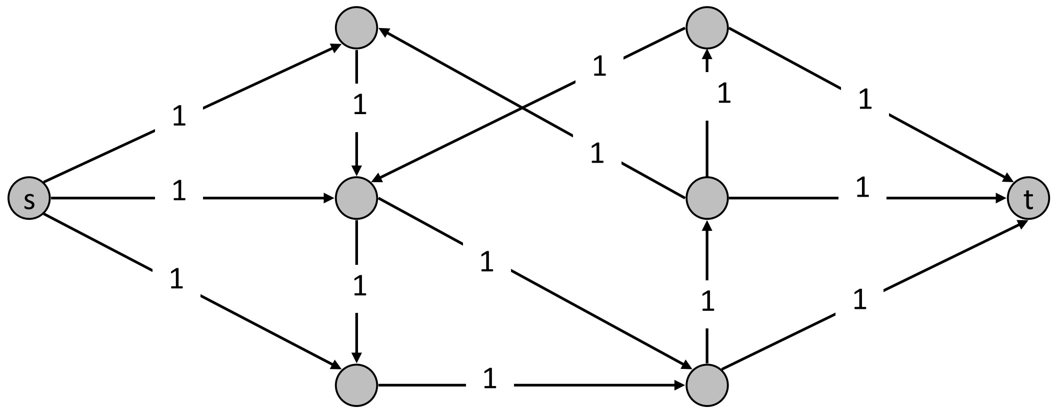
Theorem. Max number edge-disjoint s-t paths equals max flow value.

Pf. $\leq$

- Suppose there are k edge-disjoint paths $P_1, \ldots, P_k$.
- Set f(e) = 1 if e participates in some path P_i ; else set f(e) = 0.
- Since paths are edge-disjoint, f is a flow of value k. ▪

Edge Disjoint Paths

Max flow formulation: assign unit capacity to every edge.



Theorem. Max number edge-disjoint s-t paths equals max flow value.

Pf. $\geq$

- Suppose max flow value is k.
- Integrality theorem $\Rightarrow$ there exists 0-1 flow f of value k.
- Consider edge (s, u) with f(s, u) = 1.
  - by conservation, there exists an edge (u, v) with f(u, v) = 1
  - continue until reach t, always choosing a new edge
- Produces k (not necessarily simple) edge-disjoint paths. ▪

can eliminate cycles to get simple paths if desired

Extensions of Network Flow

Network flow - useful in a wide variety of applications

We will discuss two useful extensions to the network flow problem.

- Both can be reduced to network flow
- Single algorithm will solve them both.

Many computational problems that would seem to have little to do with flow of fluids through networks can be expressed as one of these two extended versions.

Circulation with Demands

Circulation with demands.

- Directed graph G = (V, E).
- Edge capacities c(e), $e \in E$.
- Node supply and demands d(v), $v \in V$.

demand if d(v) > 0; supply if d(v) < 0; transshipment if d(v) = 0

Def. A circulation is a function that satisfies:

- For each $e \in E$: $0 \leq f(e) \leq c(e)$ (capacity)
- For each $v \in V$: $\sum_{e \text{ in to } v} f(e) - \sum_{e \text{ out of } v} f(e) = d(v)$ (conservation)

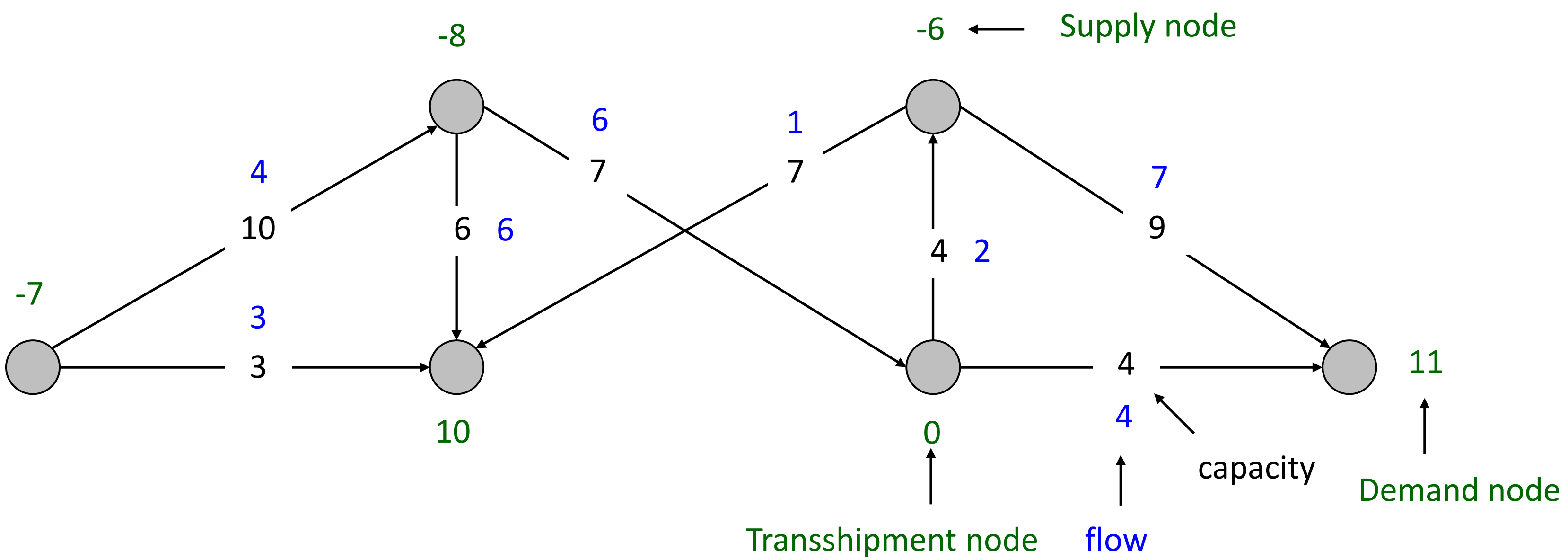
Circulation problem: Given (V, E, c, d), does there exist a circulation?

Circulation with Demands

Necessary condition: sum of supplies = sum of demands.

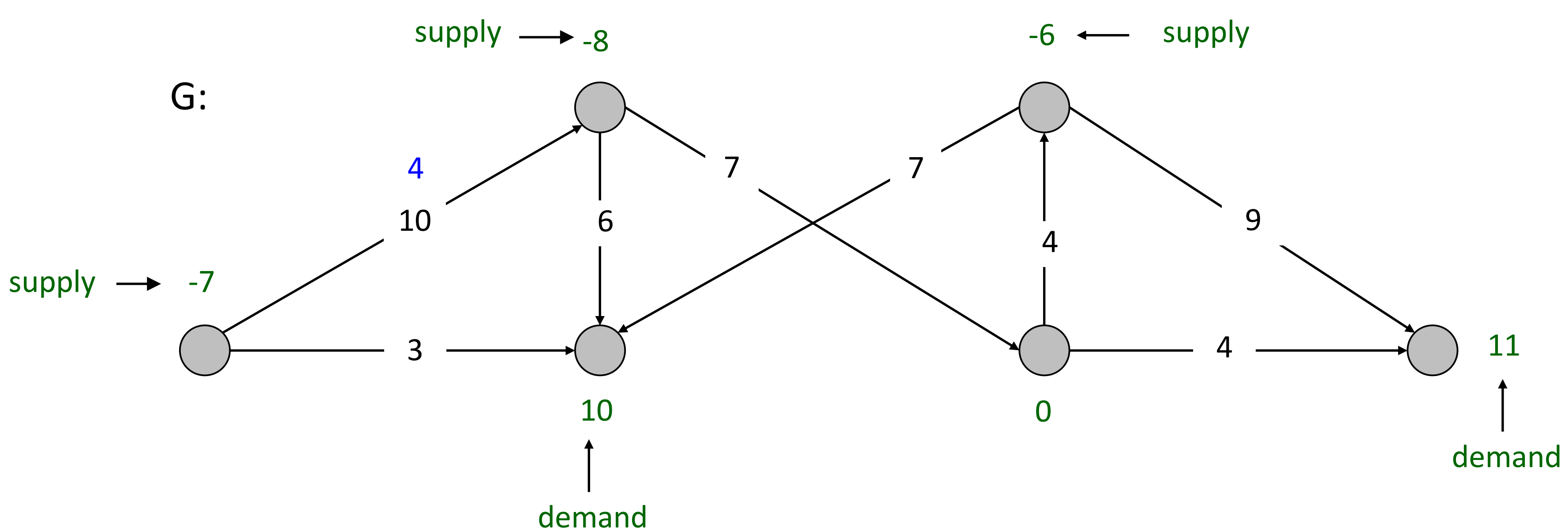
$$\sum_{v : d(v) > 0} d(v) = \sum_{v : d(v) < 0} -d(v) =: D$$

Pf. Sum conservation constraints for every demand node v.



Circulation with Demands

Max flow formulation.

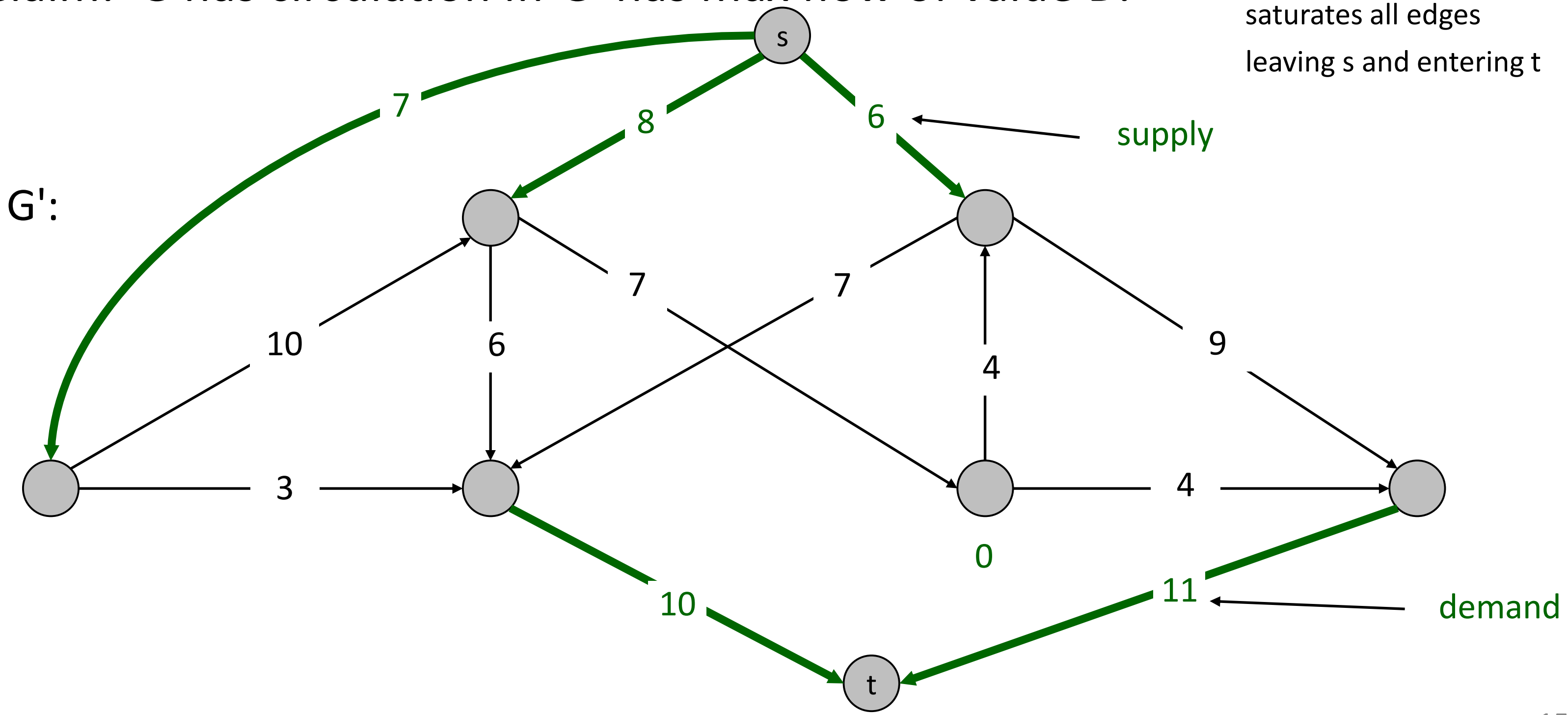


Circulation with Demands

Max flow formulation.

- Add new source s and sink t.
- For each v with d(v) < 0, add edge (s, v) with capacity -d(v).
- For each v with d(v) > 0, add edge (v, t) with capacity d(v).
- Claim: G has circulation iff G' has max flow of value D.

saturates all edges
leaving s and entering t



Lemma: There is a feasible circulation in G if and only if G' has an s^*-t^* flow of value D.

Proof: ($\Rightarrow$) Suppose that there is a feasible circulation f in G. The value of this circulation (the net flow coming out of all supply nodes) is clearly D. We can create a flow f' of value D in G', by saturating all the edges coming out of s^* and all the edges coming into t^*. We claim that this is a valid flow for G'. Clearly it satisfies all the capacity constraints. To see that it satisfies the flow balance constraints observe that for each vertex $v \in V$, we have one of three cases:

- ($v \in S$) The flow into v from s^* matches the supply coming out of v from the circulation.
- ($v \in T$) The flow out of v to t^* matches the demand coming into v from the circulation.
- ($v \in V \setminus (S \cup T)$) We have $d_v = 0$, which means that it satisfies flow conservation by the supply/demand constraints.

($\Leftarrow$) Conversely, suppose that we have a flow f' of value D in G'. It must be that each edge leaving s^* and each edge entering t^* is saturated. Therefore, by the flow conservation of f', all the supply nodes and all the demand nodes have achieved their desired supply/demand constraints. All the other nodes satisfy their supply/demand constraints because by the flow conservation of f' the incoming flow equals the outgoing flow. Therefore, the resulting flow is a circulation for G.

It is not hard to see to that the reduction can be performed in $O(n+m)$ time by a simple analysis of the network's structure. Thus, the overall running time is dominated by the time to compute the network flow (which is $O(nm)$ according to the current state-of-art).

Circulation with Demands and Lower Bounds

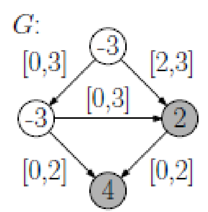
Feasible circulation.

- Directed graph G = (V, E).
- Edge capacities c(e) and lower bounds ℓ (e), e $\in$ E.
- Node supply and demands d(v), v $\in$ V.

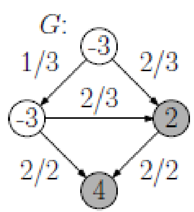
Def. A circulation is a function that satisfies:

- For each e $\in$ E: $\ell(e) \leq f(e) \leq c(e)$ (capacity)
- For each v $\in$ V: $\sum_{e \text{ in to } v} f(e) - \sum_{e \text{ out of } v} f(e) = d(v)$ (conservation)

Circulation problem with lower bounds. Given (V, E, ℓ, c, d), does there exists a circulation?



Initial network (edges labeled with $[\ell, c]$ bounds)

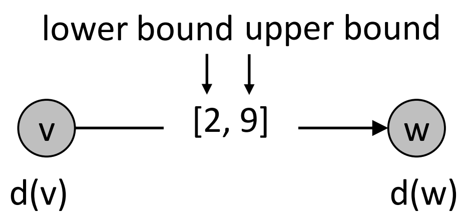


A valid flow

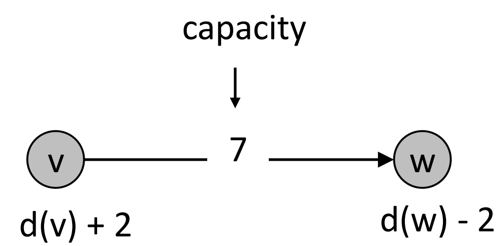
Circulation with Demands and Lower Bounds

Idea. Model lower bounds with demands.

- Send $\ell(e)$ units of flow along edge e.
- Update demands of both endpoints.



G

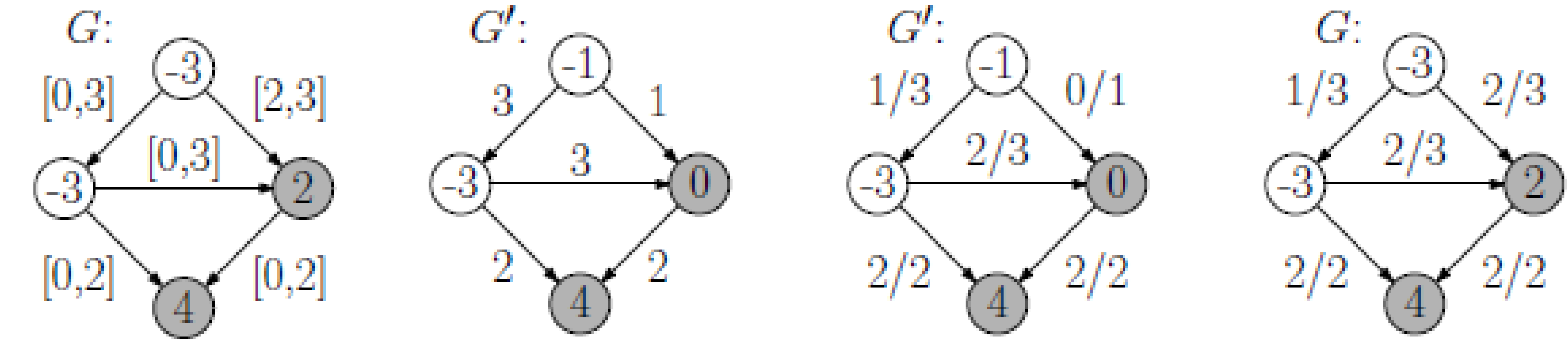


G'

Theorem. There exists a circulation in G iff there exists a circulation in G'. If all demands, capacities, and lower bounds in G are integers, then there is a circulation in G that is integer-valued.

Pf sketch. f(e) is a circulation in G iff $f'(e) = f(e) - \ell(e)$ is a circulation in G'.

Circulation with Demands and Lower Bounds



Initial network (edges labeled with $[\ell, c]$ bounds)

(a)

Modified (with no lower capacity bounds)

(b)

Valid circulation f_1

(c)

Final circulation (for original network)

(d)

Survey Design

Survey design.

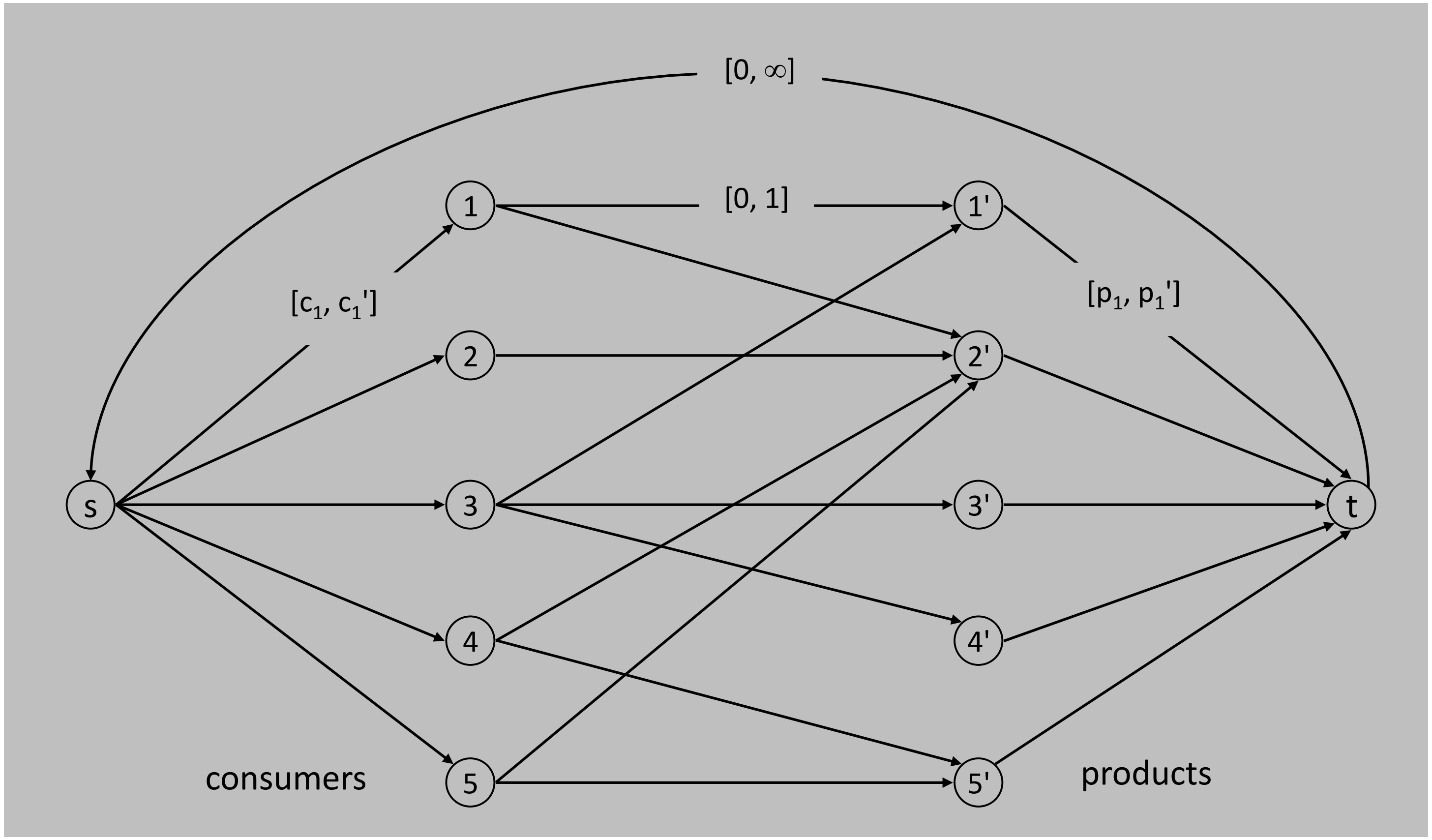
- Design survey asking n_1 consumers about n_2 products.
- Can only survey consumer i about a product j if they own it.
- Ask consumer i between c_i and c_i' questions.
- Ask between p_j and p_j' consumers about product j.

Goal. Design a survey that meets these specs, if possible.

Survey Design

Algorithm. Formulate as a circulation problem with lower bounds.

- Include an edge (i, j) if customer i owns product j.
- Integer circulation ⇔ feasible survey design.



Lemma: There exists a valid circulation in G if and only there is a valid survey design.

Proof: ($\Rightarrow$) Suppose that G has a valid (integer-valued) circulation. For each customer-product edge (i, j) that carries one unit of flow, customer i is surveyed about product j. By definition of the edges, customers are surveyed only about products they purchased. From our capacity constraints and the fact that demands are all zero, it follows that the total flow into each customer node is between c_i and c'_i, implying that this customer is asked about this many products. Similarly, the total flow out of each product node is between p_j and p'_j, implying that this product is involved in this many products surveys

($\Leftarrow$) Suppose that there is a valid survey design. We construct a flow in G as follows. For each customer-product pair (i, j) involved in the survey, we create a flow of one unit on edge (i, j). We set the flow along the edge (s, i) to the number of surveys that customer i answers, we set the flow along the edge (j, t) to the number of surveys involving product j, and we set the flow on edge (t, s) to the total number of surveys. It is straightforward to see that (by the rules of a valid survey design) this is a valid flow in G, and in particular, it satisfies the lower and upper capacity constraints and the supply and demand constraints.