Analysis and Synthesis of Algorithms
Design of Algorithms

Maximum Flow Algorithms

Maximum Flow Application to Maximal Bipartite Matching

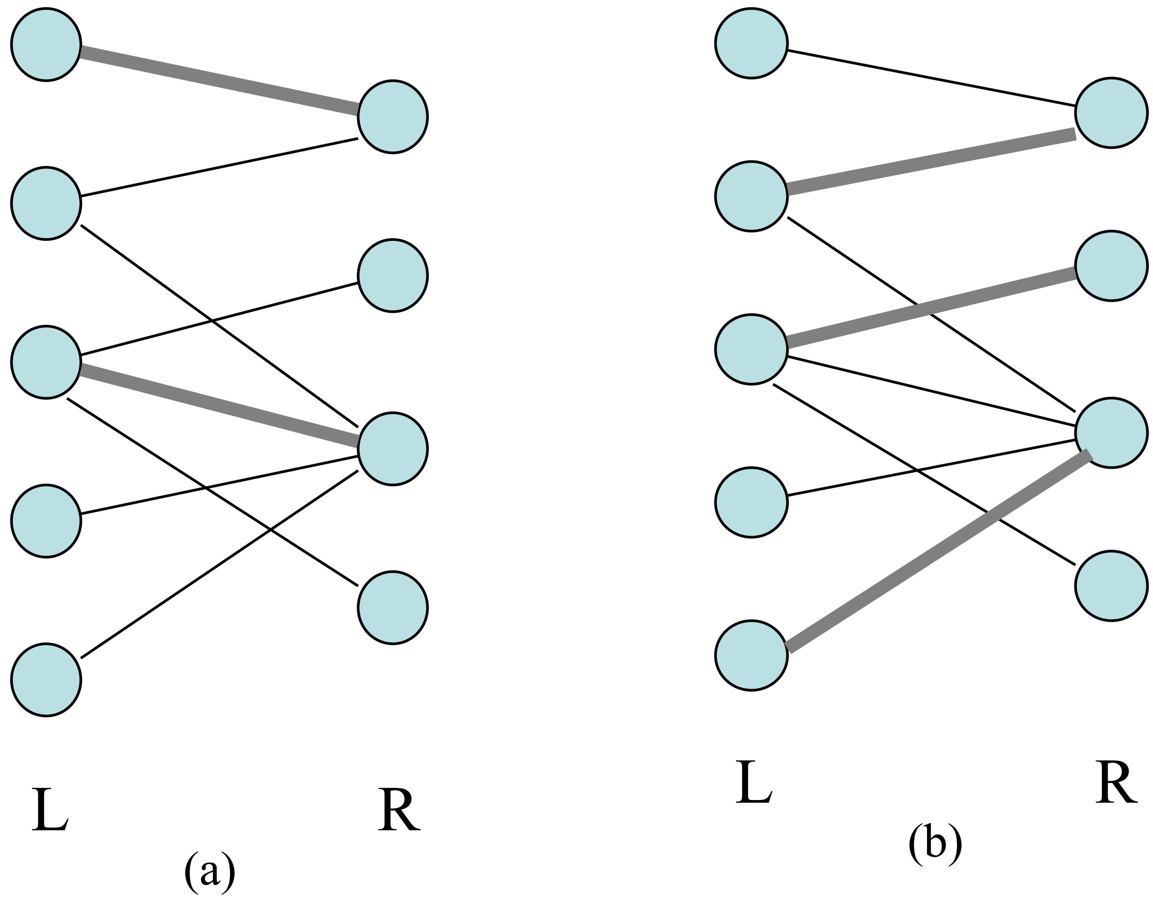
Maximal Bipartite Matching

- Undirected Graph $G = (V, E)$
- **Matching:**
  - $M \subseteq E$, such that for any vertex $v \in V$ ***at most*** one edge in M is incident on v
- **Maximal Matching:**
  - Matching of Maximal Cardinality (in M)
- **Bipartite Graph:**
  - Graph can be divided in $V = L \cup R$, where L and R are disjoint and all edges in E are between L and R
- **Maximal Bipartite Graph:**
  - Maximal Matching in which G is Bipartite

Maximal Bipartite Matching

- **Bipartite graph**: a graph (V, E), where $V = L \cup R$, $L \cap R = \emptyset$, and for every $(u, v) \in E$, $u \in L$ and $v \in R$.
- Given an undirected graph G=(V,E), a **matching** is a subset of edges $M \subseteq E$ such that for all vertices $v \in V$, at most one edge of M is incident on v.
- We say that a vertex $v \in V$ is matched by **matching** M if some edge in M is incident on v; otherwise, v is unmatched.
- A **maximum matching** is a matching of maximum cardinality, that is, a matching M such that for any matching M', we have $|M| \geq |M'|$.

Bipartite Matching: Examples



A bipartite graph G=(V,E) with vertex partition $V=L\cup R$.

(a) A matching with cardinality 2.

(b) A maximum matching with cardinality 3.

Maximal Bipartite Matching

- Using Maximum Flow Algorithms
  - Build an auxiliary graph G'
  - Define edge capacities

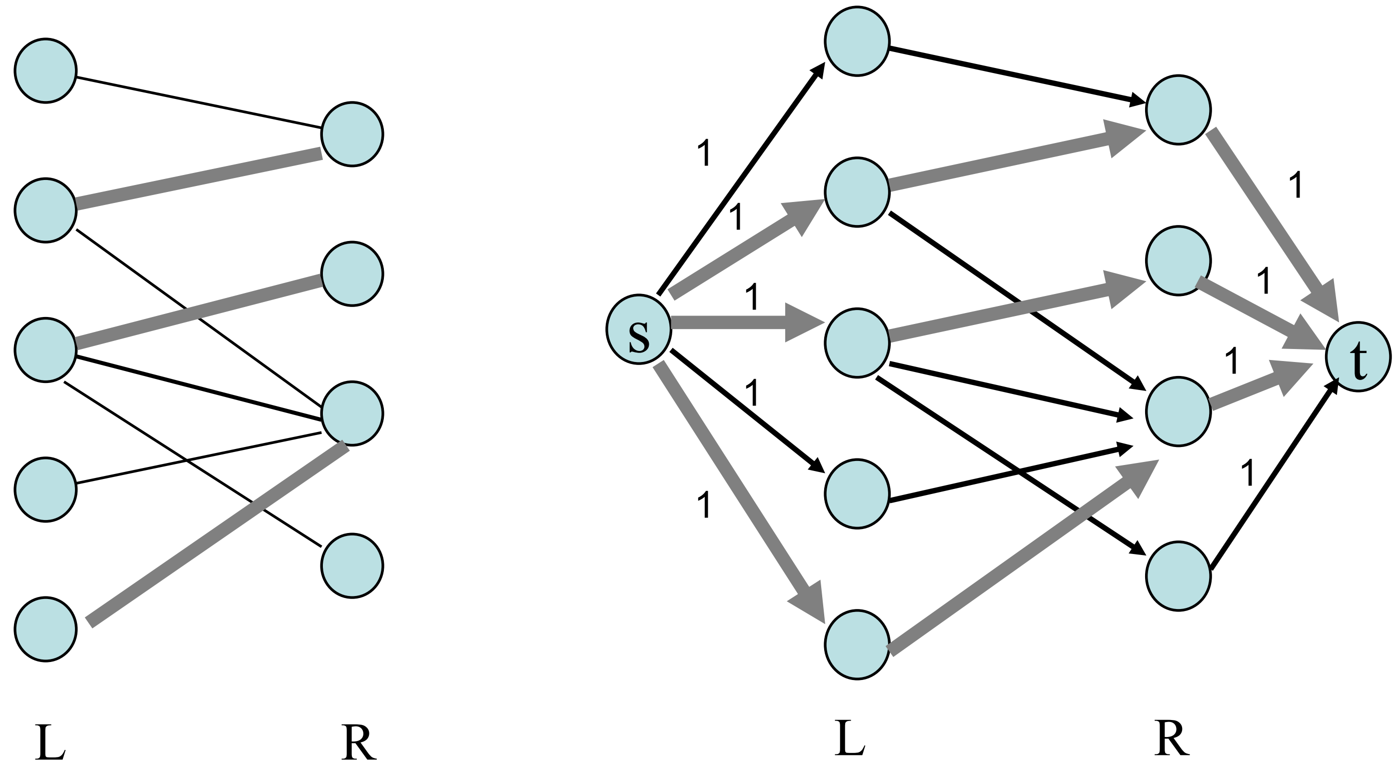
- Build G':
  - $V' = V \cup \{s, t\}$

$$
\begin{aligned}
E' \;=\; & \{(s,u) : u \in L\} \\
\cup\; & \{(u,v) : u \in L, v \in R, \text{ and } (u,v) \in E\} \\
\cup\; & \{(v,t) : v \in R\}
\end{aligned}
$$

  - Assign **unit** capacity to each edge in E'

- Maximal Bipartite Matching in G is equivalent to finding a maximum flow in G'

Recasting as a Max-Flow Problem



(a) (b)

(a) The bipartite graph G=(V,E) with vertex partition V=L∪R.
A maximum matching is shown by shaded edges.

(b) The corresponding flow network. Each edge has unit capacity.
Shaded edges have a flow of 1, and all other edges carry no flow.

Maximal Bipartite Matching

- Given G and G':
  1. If M is a matching in G, exists an integer flow f in G', with $|f| = |M|$

  – Let M be a matching, and $(u,v) \in M$.
    - Define f using edges of M, $f(s,u) = f(u,v) = f(v,t) = 1$.
    - For the remainder edges $(u,v) \in E'$, $f(u,v) = 0$
  – The paths $\mathbf{s} \rightarrow \mathbf{u} \rightarrow \mathbf{v} \rightarrow \mathbf{t}$ for all $(u,v) \in M$ are disjoint in terms of vertices with the exception of s and t
  – As there are $|M|$ paths, each of which with a single unit contribution to the flow for a total of flow f, $|f| = |M|$

Maximal Bipartite Matching

- Given G and G':
  2. If $|f|$ is an integer flow in G', then there exists a matching M in G, with $|M| = |f|$

  – Define $M = \{(u,v): u \in L, v \in R \text{ and } f(u,v) > 0\}$
    - For each $u \in L$, exists at most one $v \in R$ such that $f(u,v)=1$
      – Only a single edge with capacity 1
      – Capacities are integer
    - Similarly for $v \in R$

  – Therefore M is a matching
  – $|M| = f(L,R) = f(s,L) = |f|$

Maximal Bipartite Matching

- If all edge capacities are integers, then the maximum flow |f| is an integer
  - Induction on the number of iterations of the Ford-Fulkerson algorithm

- Maximal Bipartite Matching |M| in G corresponds to |f|, where f is the maximum flow in G'
  - If |M| is a maximal bipartite matching in G, and |f| is not maximal in G', then there exists f' which is maximal
  - f' is integer, |f'| > |f|
  - f' corresponds to a matching |M'|, with |M'| > |M| a contradiction
  - …

Maximal Bipartite Matching

- Using the generic Ford-Fulkerson algorithm leads to a complexity O(E |f*|)

- Maximal bipartite matching is not greater then min(|L|,|R|) = O(V) and is an integer
  - i.e., in the worst case, |f*| = O(V)

- The complexity of the maximal biparite matching is O(V E)