COMPSCI 311: Introduction to Algorithms

Lecture 20: Network Flow Applications

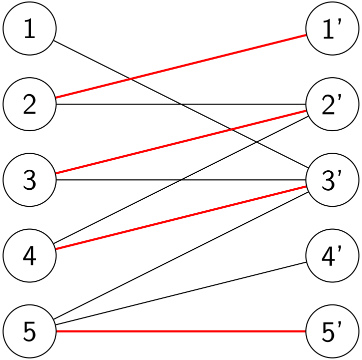
Dan Sheldon

University of Massachusetts Amherst

First Application of Network Flows: Bipartite Matching

- Given a bipartite graph $G = (L \cup R, E)$, a subset of edges $M \subseteq E$ is a matching if each node appears in at most one edge in M.
- The maximum matching problem is to find the matching with the most edges.
- We'll design an efficient algorithm for maximum matching in a bipartite graph.

Bipartite Matching

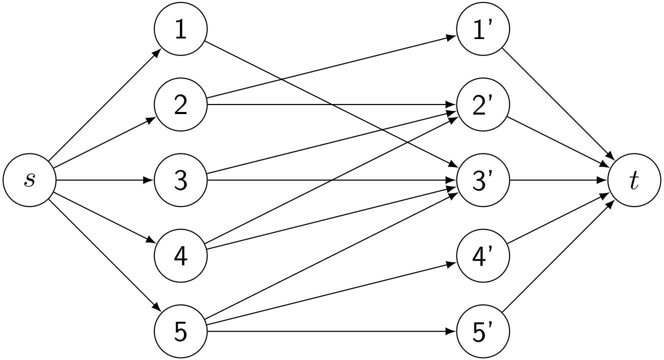


Formulating Matching as Network Flow problem

- **Goal**: given matching instance $G = (L \cup R, E)$:
  - create a flow network G',
  - find a maximum flow f in G'
  - use f to construct a maximum matching M in G.
- Exercise
- Convert undirected bipartite graph G to flow network G'
  - Direction of edges?
  - Capacities?
  - Source and sink?

Maximal Matching as Network Flow

- Add a source s and sink t
- For each edge $(u, v) \in E$, add $u \to v$ (directed), capacity 1
- Add an edge with capacity 1 from s to each node $u \in L$
- Add an edge with capacity 1 from each node $v \in R$ to t.



Clicker

Let G' be the flow network as constructed above and let e be an edge from L to R.

A. For every flow f, either $f(e) = 0$ or $f(e) = 1$.

B. For every maximum flow f, either $f(e) = 0$ or $f(e) = 1$.

C. There is some maximum flow f such that either $f(e) = 0$ or $f(e) = 1$.

D. B and C

E. A, B, and C

Maximum Matching: Analysis

- Run F-F to get an integral max-flow f
- Set M to the set of edges from L to R with flow $f(e) = 1$
- Claim: The set M is a maximum matching.

Correctness: We will show that for every integer flow of value k we can construct a matching M of size k and vice versa. Therefore, a maximum integer-valued flow yields a maximum matching.

Correctness 1

1. Integral flow f of value $k \Rightarrow$ matching M of size k

- Suppose f is a flow of value k
- Let $M =$ edges from L to R carrying one unit of flow
- There are k such edges, because the net flow across cut between L and R is k, and there are no edges from R to L
- There is at most 1 unit of flow entering $u \in L$, and therefore at most 1 unit of flow leaving u
- Since all flow values are 0 or 1, this means M has at most one edge incident to u.
- A similar argument for $v \in L$ means that M has at most one edge incident to v
- Therefore, M is a matching with size k

Correctness 2 (Review on Own)

2. Matching M of size $k \Rightarrow$ integral flow f of value k

- Suppose M is a matching of size k
- Send one unit of flow from s to $u \in L$ if u is matched
- Send one unit of flow from $v \in R$ to t if t is matched
- Send one unit of flow on e if e is in M
- All other edge flow values are zero
- Verify that capacity and flow conservation constraints are satisfied, and that $v(f) = k$.

Clicker

What is the running time of the Ford-Fulkerson algorithm to find a maximum matching in a bipartite graph with $|L| = |R| = n$?
(Assume each node has at least one incident edge.)

A. $O(m+n)$

B. $O(mn)$

C. $O(mn^2)$

D. $O(m^2n)$

Perfect Matchings in Bipartite Graphs

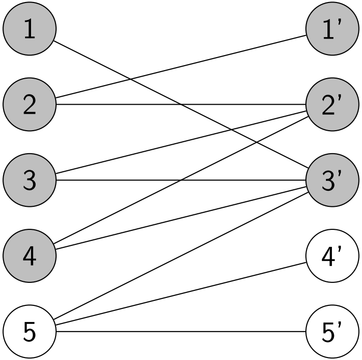
Recall: A matching M is **perfect** if every node appears in (exactly) one edge in M.

Question: When does a bipartite graph have a perfect matching?

- Clearly, we must have $|L| = |R|$
- Clearly, every node must have at least one edge
- What other conditions are necessary? Sufficient?

Perfect Matchings in Bipartite Graphs

For $S \subseteq L$, let $N(S) \subseteq R$ be the set of all neighbors of nodes in S



Observation: For a perfect matching we need

$$\forall S \subseteq L, \quad |N(S)| \geq |S| \qquad (*)$$

Otherwise we can't match all nodes in S

Hall's Marriage Theorem

Assume G is bipartite with $|L| = |R| = n$.

Simple Observation: If G has a perfect matching then:

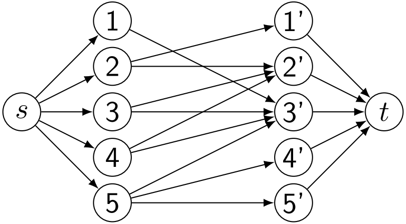
$$\forall S \subseteq L, \quad |N(S)| \geq |S| \tag{*}$$

Theorem (Hall 1935, earlier by Frobenius, Kőnig): G has a perfect matching **if and only if** (*)

We will prove: if G does not have a perfect matching then (*) does not hold $\implies$ there is some $S \subseteq L$ with $|N(S)| < |S|$.

Use max-flow / min-cut theorem on bipartite-matching flow network.

Clicker



Consider the flow network construction for bipartite matching. Which of the following is true?

A. The construction still works if edges from s to L have infinite capacity.

B. The construction still works if edges from L to R have infinite capacity.

C. The construction still works if edges from R to t have infinite capacity.

Hall's Marriage Theorem

Picture on board: G' w/ infinite-capacity $L \to R$ edges

- Suppose G does not have a perfect matching
- Let (A, B) be the minimum-cut in $G' \implies c(A, B) < n$
- Let $S = A \cap L$
- All neighbors of nodes in S are also in A, else an edge of infinite capacity is cut $\implies N(S) \subseteq A \cap R$
- The cut capacity is

$$\begin{aligned} n > c(A, B) &= |B \cap L| + |A \cap R| \\ &= n - |S| + |A \cap R| \\ &\geq n - |S| + |N(S)| \end{aligned}$$

- $\implies |S| > |N(S)|$

Baseball Elimination?

Board work

Bokeh Effect: Blurring Background

- Using an expensive camera and appropriate lenses, you can get a "bokeh" effect on portrait photos in which the background is blurred and the foreground is in focus.

- Can fake effect using cheap phone cameras and appropriate software

Formulating the Problem

Problem: given set V of pixels, classify each as foreground or background. Assume you have:

- Numeric "cost" for assigning each pixel foreground/background
- Numeric penalty for assigning neighboring pixels to different classes

Sketch of approach: other slides, board work, demo