COMP251: Bipartite graphs

Jérôme Waldispühl
School of Computer Science
McGill University

Based on slides from M. Langer (McGill) & P. Beame (UofW)

Bipartite graphs

Vertices are partitioned into 2 sets.
All edges cross the sets.

Examples

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Women</td>
<td>Traditional marriage</td>
</tr>
<tr>
<td>Students</td>
<td>registration</td>
</tr>
<tr>
<td>People</td>
<td>employment</td>
</tr>
<tr>
<td>People</td>
<td>Have read/wen</td>
</tr>
<tr>
<td></td>
<td>Courses</td>
</tr>
<tr>
<td></td>
<td>Companies</td>
</tr>
<tr>
<td></td>
<td>Books/Movies</td>
</tr>
</tbody>
</table>

Examples

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Counter-examples

Easy to identify.
But not always...

Cycles

Claim: If a graph is bipartite if and only if does not contain an odd cycle.

Proof: Q5 of assignment 2.

Is it a bipartite graph?

Assuming $G=(V,E)$ is an undirected connected graph.

1. Run DFS and build a DFS tree.
2. Color vertices by layers (e.g., red & black)
3. If all non-tree edges join vertices of different color then the graph is bipartite.

Non-tree edges in DFS tree cross 2 or more levels. Why?
**Bipartite matching**
Consider an undirected bipartite graph.

A matching is a subset of the edges \( \{ (\alpha, \beta) \} \) such that no two edges share a vertex.

**Perfect matching**
Suppose we have a bipartite graph with \( n \) vertices in each A and B. A perfect matching is a matching that has \( n \) edges.

Note: It is not always possible to find a perfect matching.

**Complete bipartite graph**
A complete bipartite graph is a bipartite graph that has an edge for every pair of vertex \( (\alpha, \beta) \) such that \( \alpha \in A, \beta \in B. \)

**The algorithm of happiness**

**Resident matching program**
- **Goal:** Given a set of preferences among hospitals and medical school students, design a self-reinforcing admissions process.
- **Unstable pair:** applicant \( x \) and hospital \( y \) are unstable if:
  - \( x \) prefers \( y \) to their assigned hospital.
  - \( y \) prefers \( x \) to one of its admitted students.
- **Stable assignment:** Assignment with no unstable pairs.
  - Natural and desirable condition.
  - Individual self-interest will prevent any applicant/hospital deal from being made.

**Stable marriage problem**
**Goal:** Given \( n \) men and \( n \) women, find a "suitable" matching. Participants rate members of opposite sex.
- Each man lists women in order of preference from best to worst.
- Each woman lists men in order of preference from best to worst.

<table>
<thead>
<tr>
<th>Men's preferences</th>
<th>Women's preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>2nd</td>
</tr>
<tr>
<td>Xavier</td>
<td>Amy</td>
</tr>
<tr>
<td>Yuri</td>
<td>Brenda</td>
</tr>
</tbody>
</table>
Stable marriage problem

- **Perfect matching**: everyone is matched monogamously.
  - Each man gets exactly one woman.
  - Each woman gets exactly one man.

- **Stability**: no incentive for some pair of participants to undermine assignment by joint action.
  - In matching $M$, an unmatched pair $m-w$ is unstable if man $m$ and woman $w$ prefer each other to current partners.
  - Unstable pair $m-w$ could each improve by eloping.

- **Stable matching**: perfect matching with no unstable pairs.

- **Stable matching problem**: Given the preference lists of $n$ men and $n$ women, find a stable matching (if one exists).

Example

Q: Is $X-C$, $Y-B$, $Z-A$ a good assignment?

A: No! Brenda and Xavier will hook up...

![Graph showing unstable pair](image)

Example

Q: Is $X-A$, $Y-B$, $Z-C$ a good assignment?

A: Yes!

![Graph showing stable pair](image)

Stable marriage problem

Consider a complete bipartite graph such that $|A| = |B| = n$.

- Each member of $A$ has a preference ordering of members of $B$.
- Each member of $B$ has a preference ordering of members of $A$.

Algorithm for finding a matching:

- Each $A$ member proposes to a $B$, in preference order.
- Each $B$ member accepts the first proposal from an $A$, but then rejects that proposal if/when it receives a proposal from an $A$ that it prefers more.

In our example: Men propose to women. Woman accept the first offer made to them, but women will drop their partner when/if a preferred man proposes to them.

Note the asymmetry between $A$ and $B$.

Gale-Shapley algorithm

For each $a \in A$, let $\text{pref}[a]$ be the ordering of its preferences in $B$.
For each $b \in B$, let $\text{pref}[b]$ be the ordering of its preferences in $A$.
Let matching be a set of crossing edges between $A$ and $B$.

```plaintext
matching ← ∅
while there is $a \in A$ not yet matched do
    $β ← \text{pref}[a].\text{removeFirst}()$
    if $β$ not yet matched then
        matching ← matching $∪$ $\{(a,β)\}$
    else
        $γ ← β$'s current match
        if $β$ prefers $a$ over $γ$ then
            matching ← matching $∪$ $\{(γ,β)\}$ $∪$ $\{(a,β)\}$
    end if
end while
return matching
```
<table>
<thead>
<tr>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xavier</td>
<td>Brenda</td>
<td>Amy</td>
</tr>
<tr>
<td>Yuri</td>
<td>Brenda</td>
<td>Claire</td>
</tr>
<tr>
<td>Zoran</td>
<td>Amy</td>
<td>Claire</td>
</tr>
</tbody>
</table>

**Men's preferences**

<table>
<thead>
<tr>
<th>1st</th>
<th>2nd</th>
<th>3rd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amy</td>
<td>Zoran</td>
<td>Xavier</td>
</tr>
<tr>
<td>Yuri</td>
<td>Brenda</td>
<td>Claire</td>
</tr>
<tr>
<td>Zoran</td>
<td>Amy</td>
<td>Claire</td>
</tr>
</tbody>
</table>

**Women's preferences**
Correctness (termination)

Observations:
1. Men propose to women in decreasing order of preference.
2. Once a woman is matched, she never becomes unmatched; she only "trades up."

Claim: Algorithm terminates after at most \( n^2 \) iterations of while loop (i.e. \( O(n^2) \) running time).

Proof: Each time through the while loop a man proposes to a new woman. There are only \( n^2 \) possible proposals.

Correctness (perfection)

Claim: All men and women get matched.

Proof: (by contradiction)
- Suppose, for sake of contradiction, that Zoran is not matched upon termination of algorithm.
- Then some woman, say Amy, is not matched upon termination.
- By Observation 2 (only trading up, never becoming unmatched), Amy was never proposed to.
- But, Zoran proposes to everyone. Contradiction.

Correctness (stability)

Claim: No unstable pairs.

Proof: (by contradiction)
- Suppose A-Z is an unstable pair: each prefers each other to partner in Gale-Shapley matching.
  - Case 1: Z never proposed to A.
    \( \Rightarrow \) A prefers his GS partner to A.
    \( \Rightarrow \) A-Z is stable.
  - Case 2: Z proposed to A.
    \( \Rightarrow \) A rejected Z (right away or later)
    \( \Rightarrow \) A prefers her GS partner to Z.
    \( \Rightarrow \) A-Z is stable.
- In either case A-Z is stable. Contradiction.
Optimality

**Definition:** Man $m$ is a valid partner of woman $w$ if there exists some stable matching in which they are matched.

**Man-optimal assignment:** Each man receives best valid partner (according to his preferences).

**Claim:** All executions of GS yield a man-optimal assignment, which is a stable matching!

---

Man-Optimality

**Claim:** GS matching $S^*$ is man-optimal.

**Proof:** (by contradiction)

- Suppose some man is paired with someone other than his best partner. Men propose in decreasing order of preference so some man is rejected by a valid partner.
- Let $Y$ be first such man, and let $A$ be the first valid woman that rejects him.
- Let $S$ be a stable matching where $A$ and $Y$ are matched.
- In building $match1sg$, when $Y$ is rejected, $A$ forms (or reaffirms) engagement with a man, say $Z$, whom she prefers to $Y$.
- Let $B$ be $Z$'s partner in $S$.
- In building $match1sg$, $Z$ is not rejected by any valid partner at the point when $Y$ is rejected by $A$.
- Thus, $Z$ prefers $A$ to $B$.
- But $A$ prefers $Z$ to $Y$.
- Thus $A-Z$ is unstable in $S$. 