Matching algorithms (Cont...)

RAMESH K. JALLU

LEC-03

DT. 17//01//22

Recap

- Alternating paths and augmenting paths with respect to a matching
- The augmenting path theorem
  - A matching M is a maximum matching in graph G ***if and only*** if there are **no** M-augmenting paths in G

THE MATCHING ALGORITHM

{

1. Start with any matching.
2. Find an augmenting path with respect to the current matching.
3. Augment the current matching.
4. Repeat the above two steps as long as possible.

}

Algorithm for bipartite graphs

- Let $G = (V, E)$ be a bipartite graph, where $V = V_1 \cup V_2$
- **Approach**: Repeatedly find an augmenting path and obtain a new matching using the path, until we discover a matching for which there is no augmenting path
- **Question**: How can we obtain augmenting paths wrt a matching M in G in a systematic and efficient way?
- For ease of understanding assume that the vertices in V_1 are colored red and the vertices in V_2 are colored blue

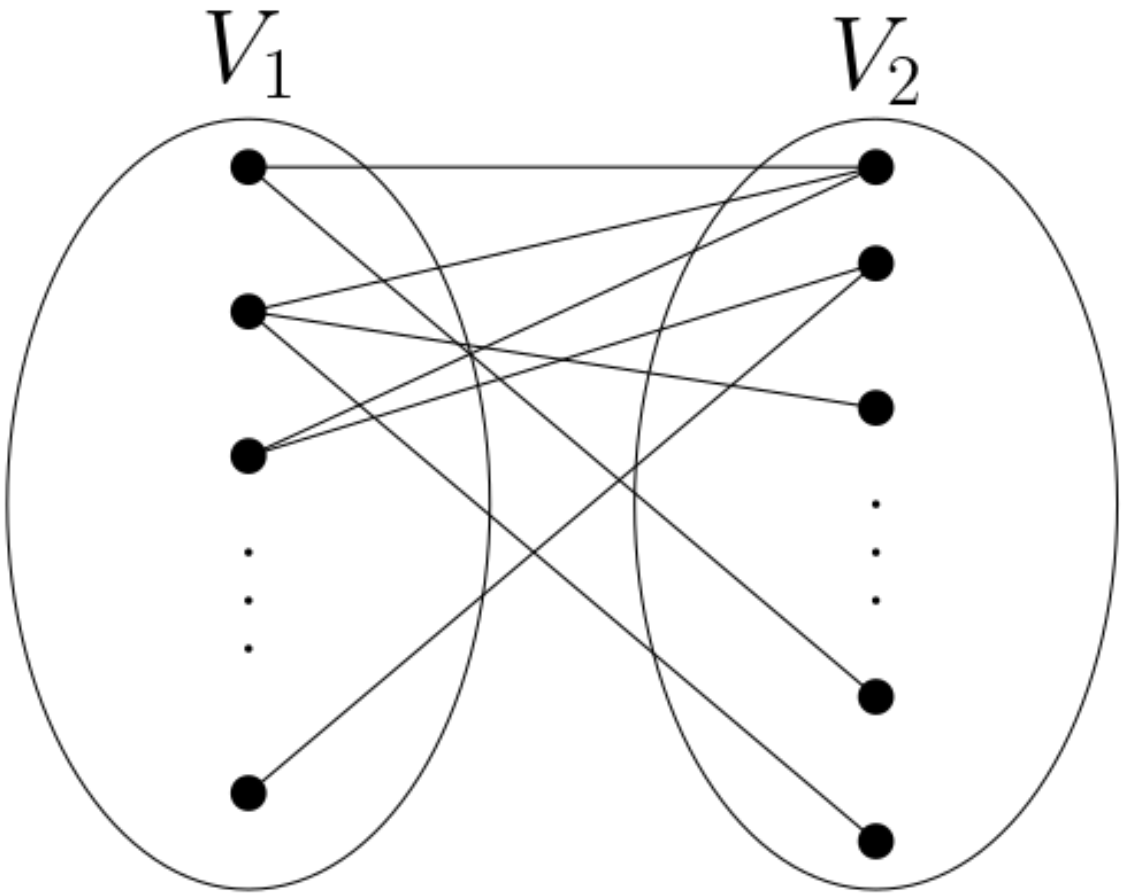
Alternating tree

- With the aid of alternating trees we try find augmenting path wrt to matching M, if exists
- The approach will work properly in bipartite graphs
  - We will extend it later so that it would also be able to cope with non-bipartite graphs
- We construct a forest of alternating trees
- An ***alternating tree*** is a tree whose root is an unmatched vertex
- All root-leaf paths in an alternating tree are alternating paths wrt to M
- If we manage to add an unmatched vertex, other than the root, to an alternating tree, we have identified an augmenting path

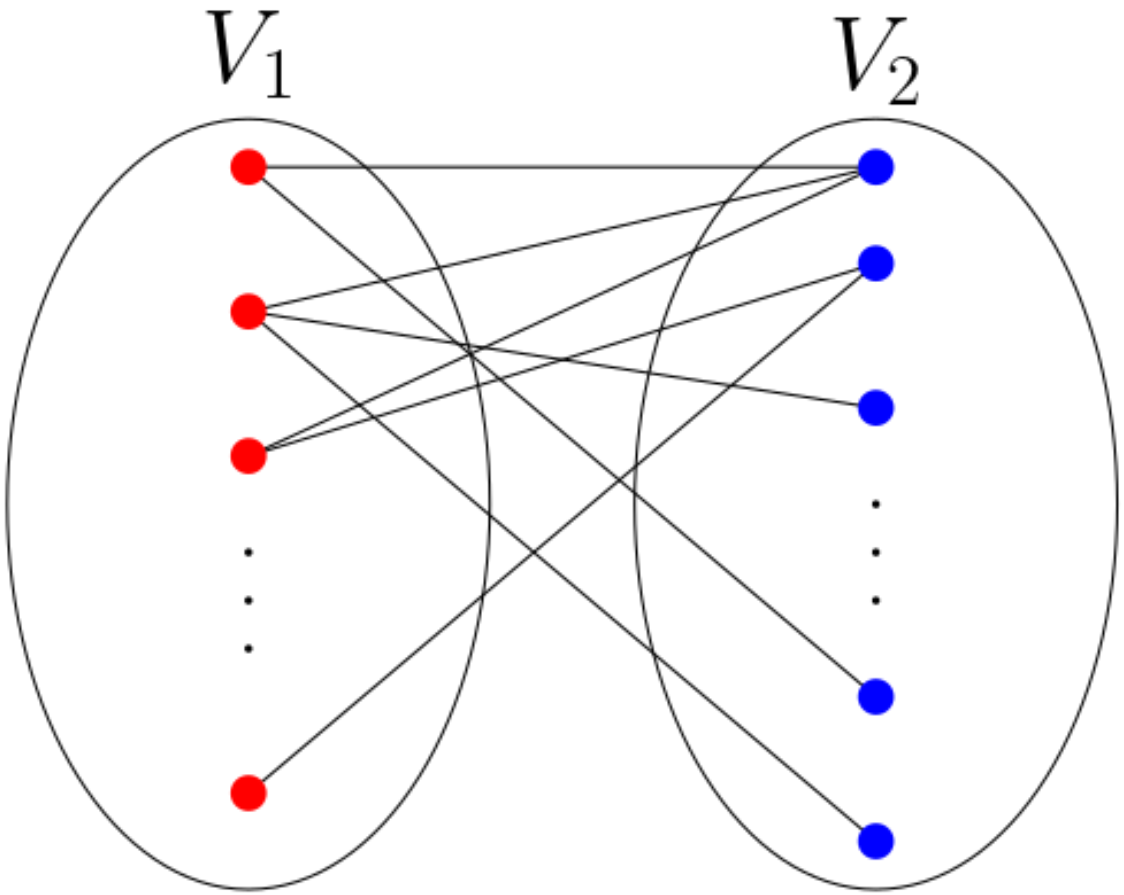
Alternating tree algorithm

- We start with an empty matching and iteratively augment the matching using the following procedure
  1. Start with an unmatched red vertex v
  2. Consider all the unmatched edges out of it
     a. If one of these leads to an unmatched vertex w, add the edge (v, w) to the matching
     b. Otherwise, go to step 3
  3. From the set of blue vertices obtained in step 2, consider all the matched edges
  4. From the set of red vertices obtained in the previous step, consider all the unmatched edges
     a. If one of these leads to an unmatched vertex, we have found an augmenting path; flip along this path to increase the size of the matching by 1
     b. Otherwise, go back to step 3

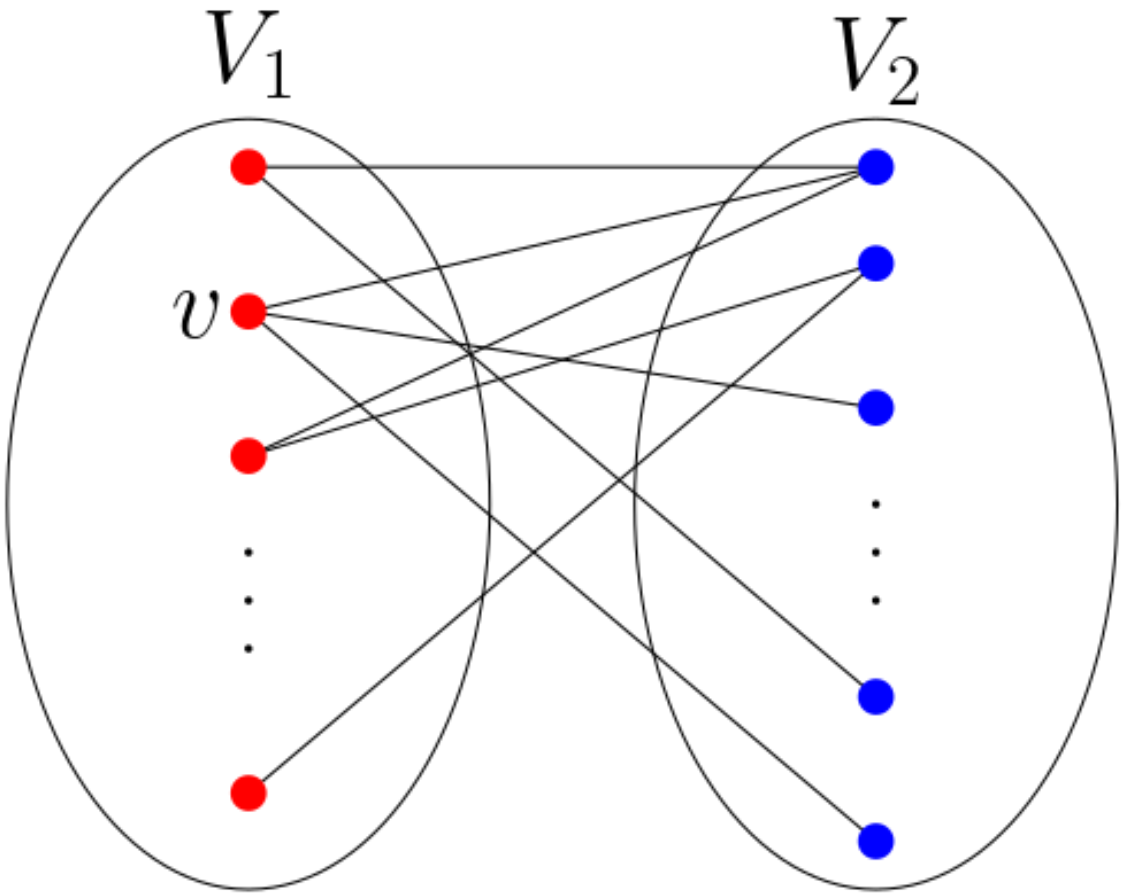
Example



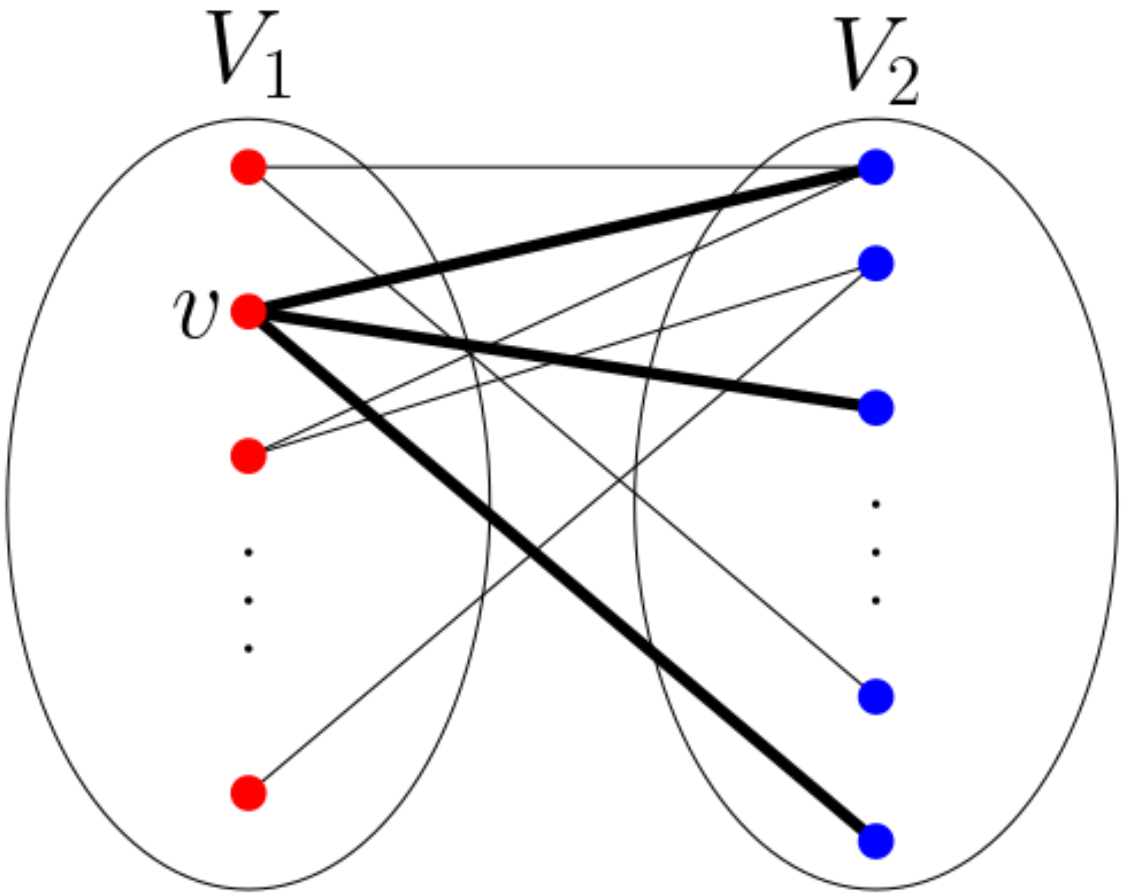
Example



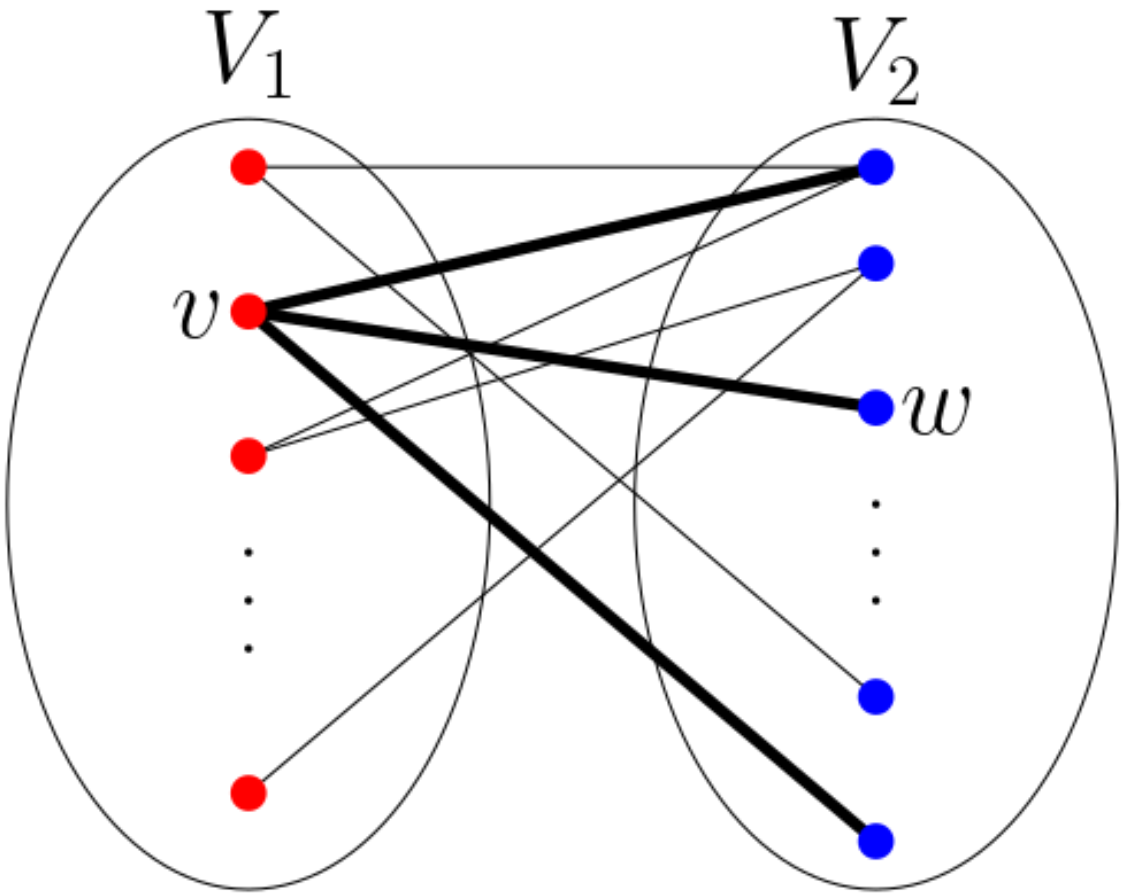
Example



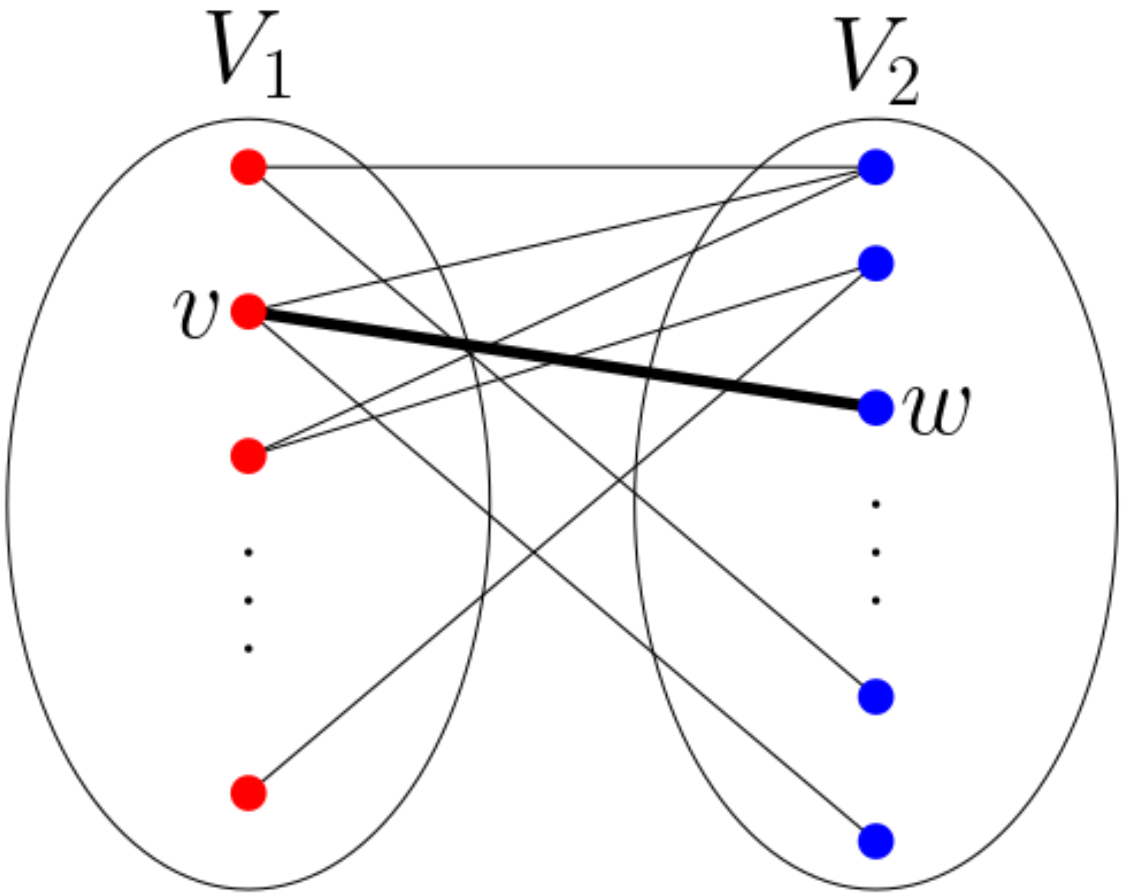
Example



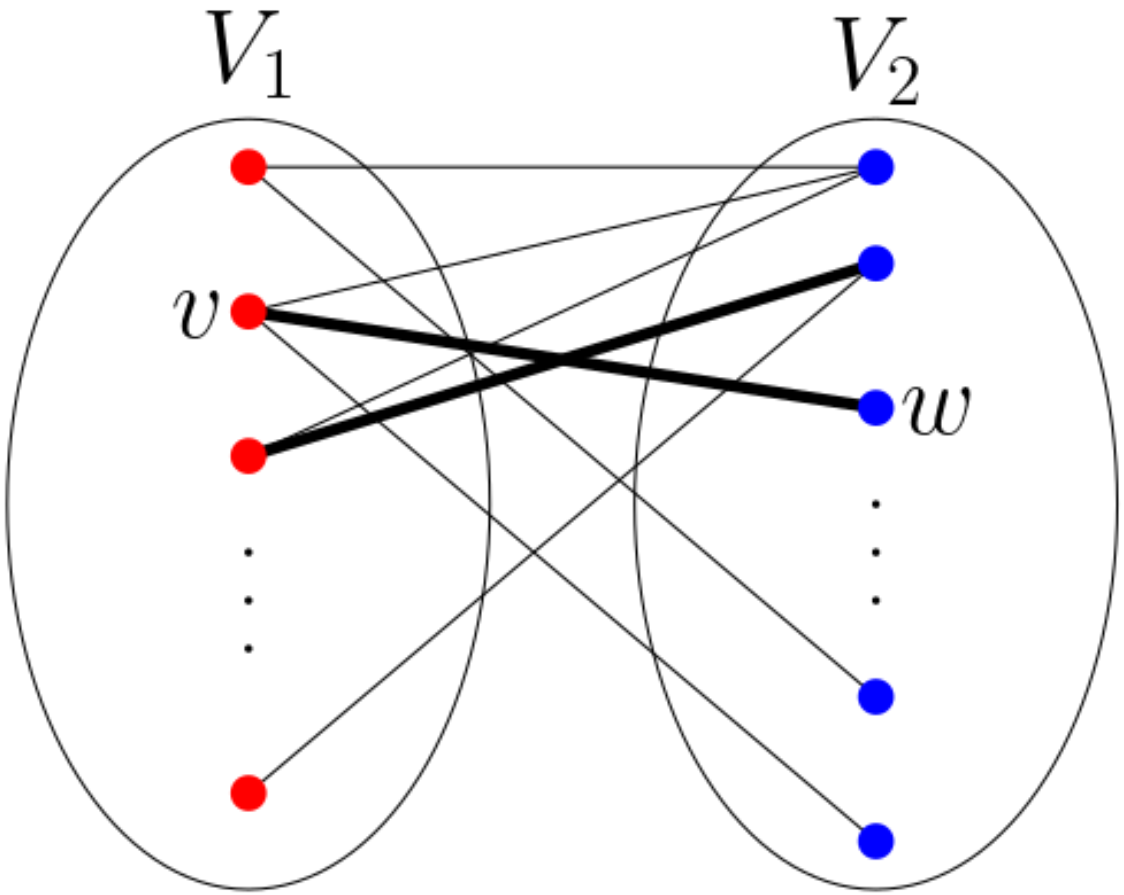
Example



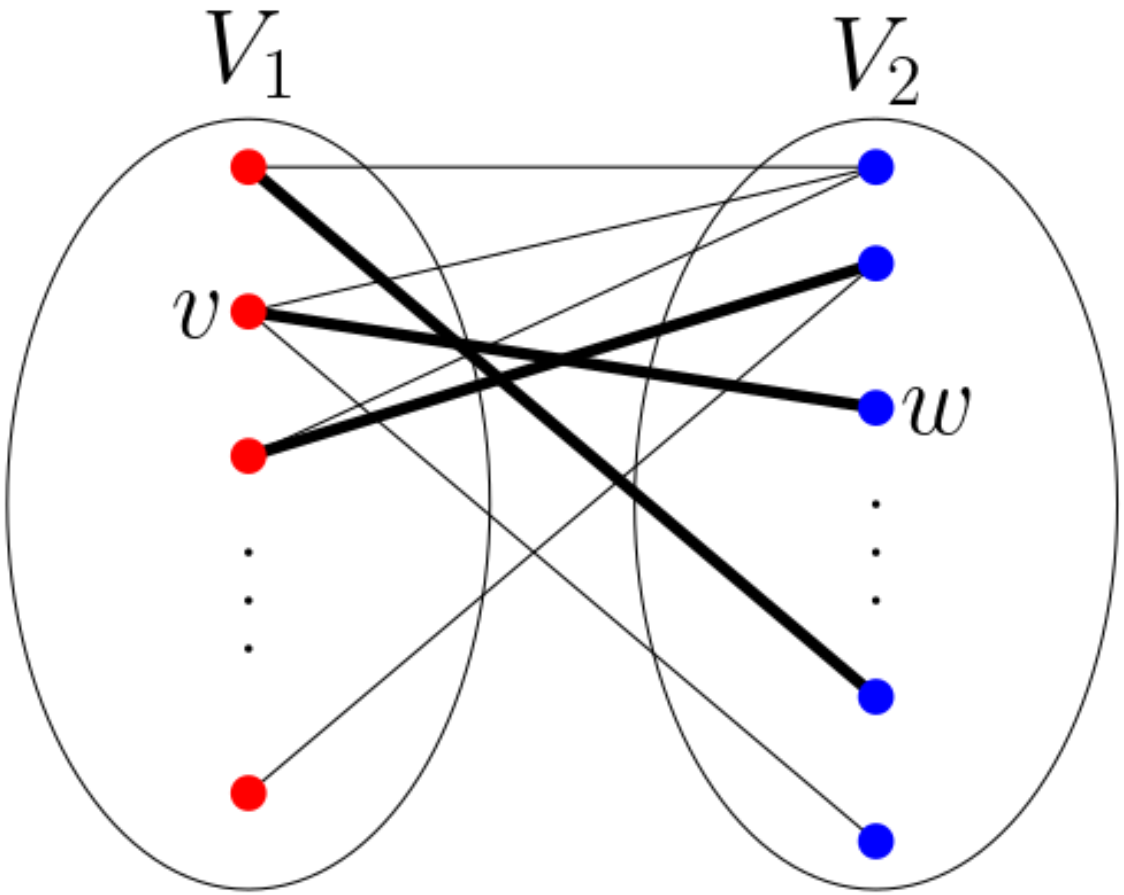
Example



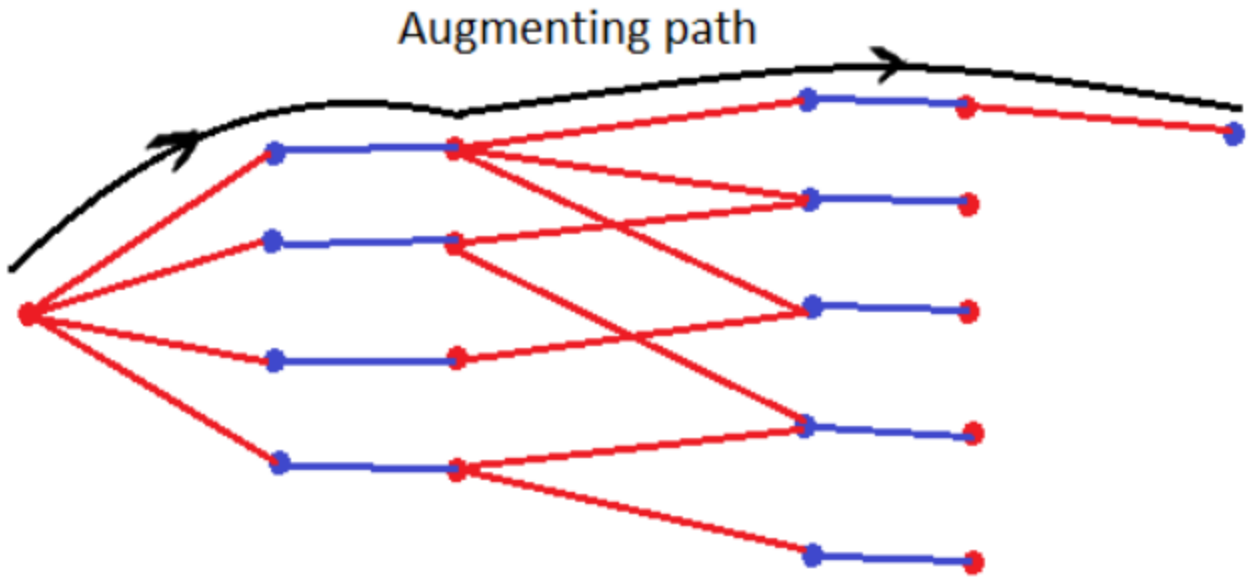
Example



Example



Example



- Thank you!