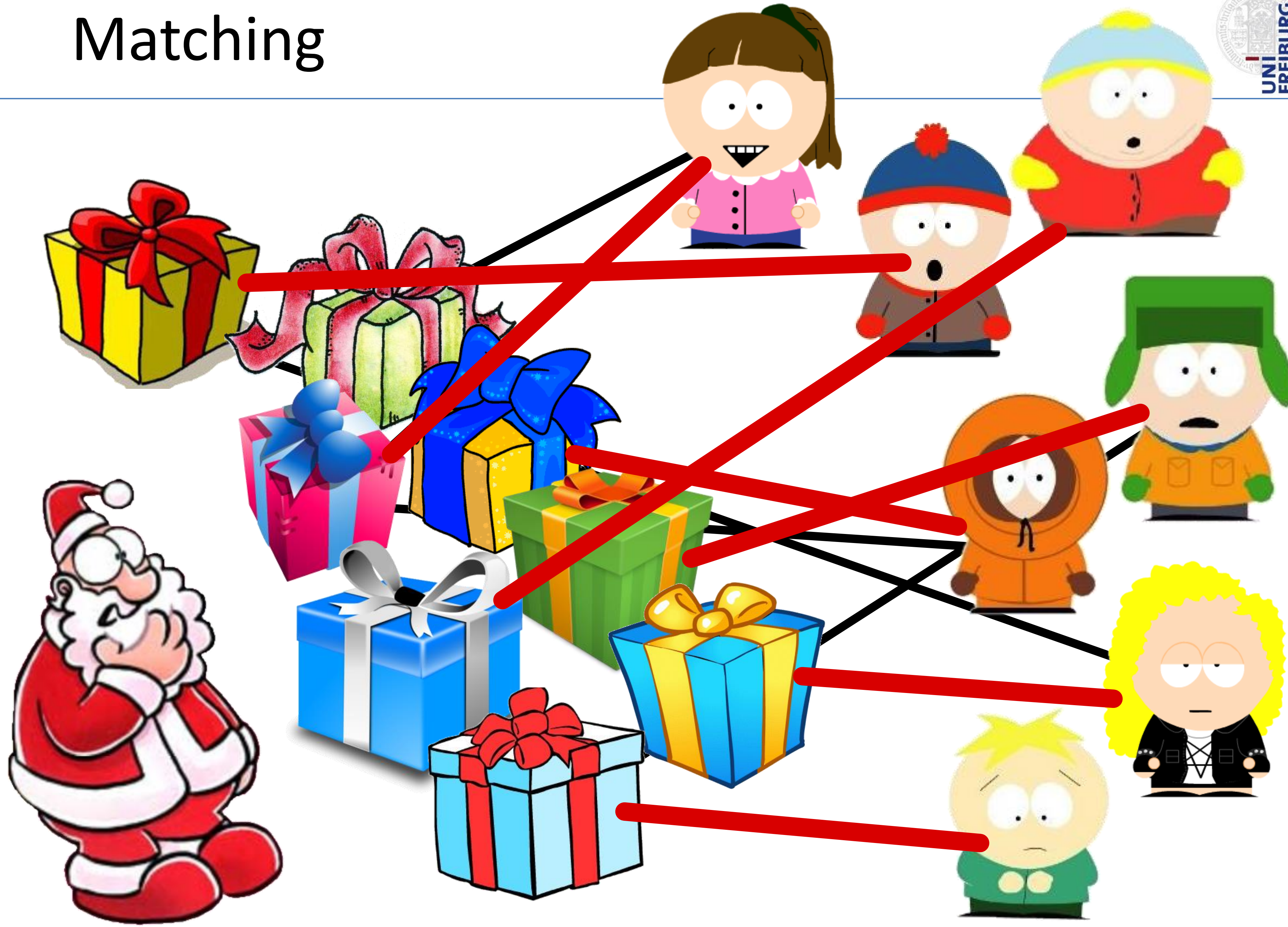
Algorithm Theory

Chapter 6
Graph Algorithms

Part VIII:
Bipartite Maximum Matching

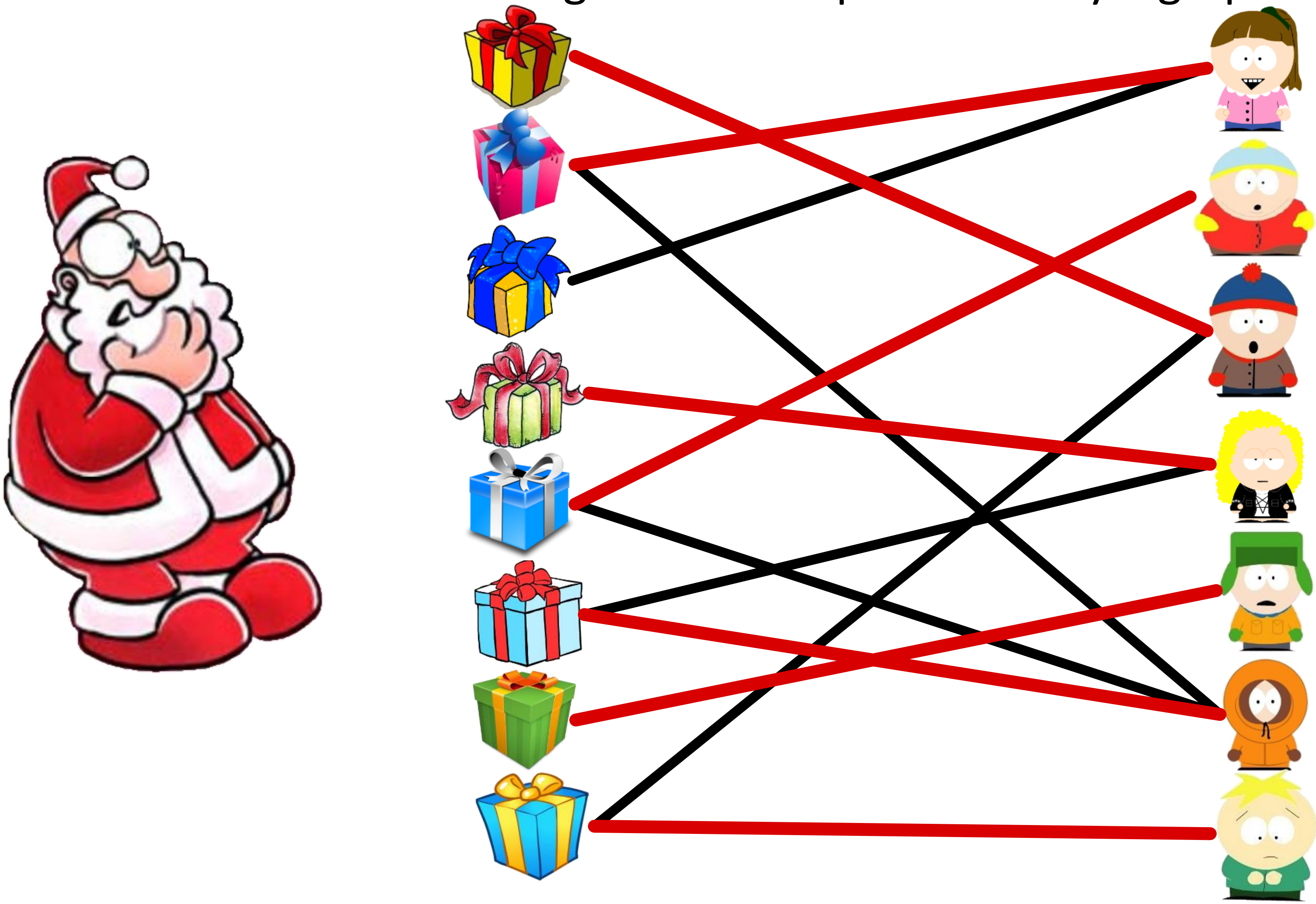
Fabian Kuhn

Matching



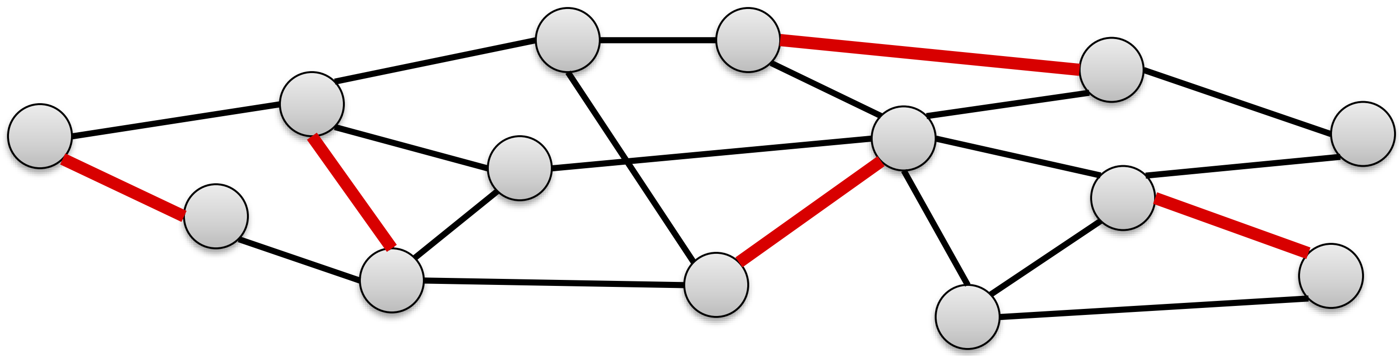
Gifts-Children Graph

- Which child likes which gift can be represented by a graph



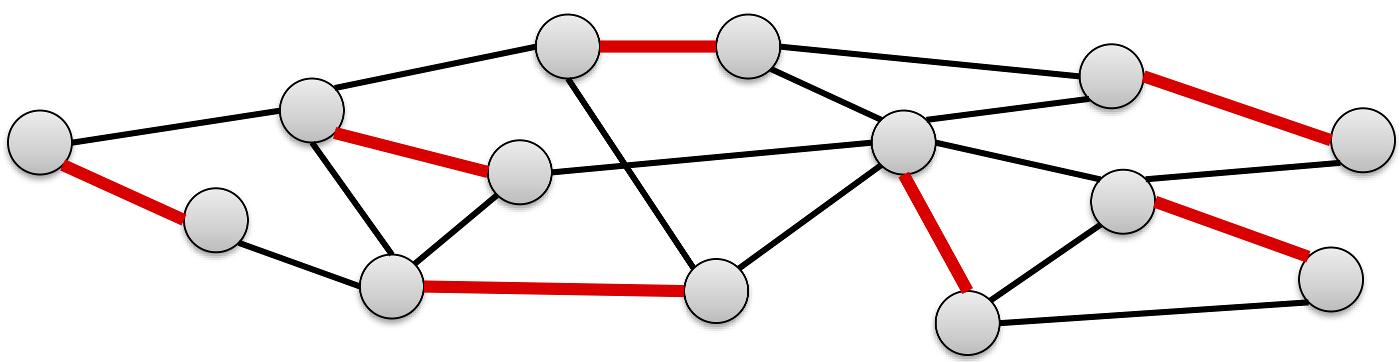
Matching

Matching: Set of pairwise non-incident edges



Maximal Matching: A matching s.t. no more edges can be added

Maximum Matching: A matching of maximum possible size



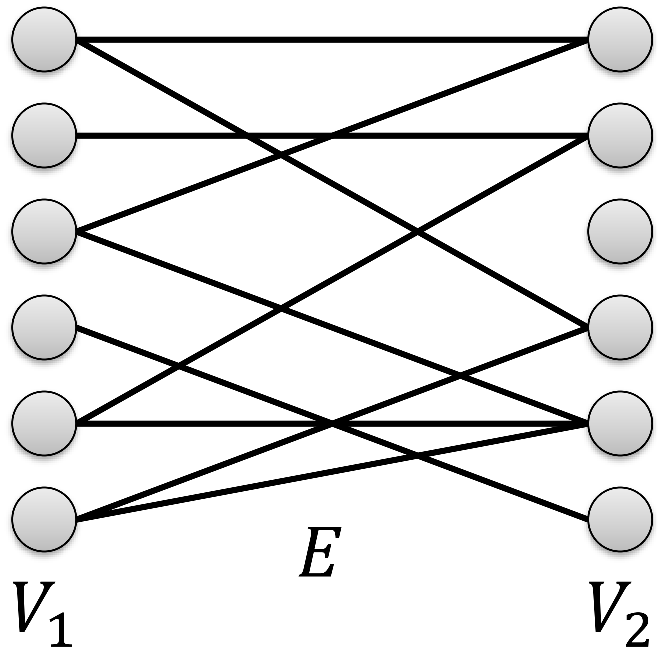
Perfect Matching: Matching of size $n/2$ (every node is matched)

Bipartite Graph

Definition: A graph $G = (V, E)$ is called bipartite iff its node set can be partitioned into two parts $V = V_1 \,\dot\cup\, V_2$ such that for each edge $\{\mathrm{u}, \mathrm{v}\} \in E$,

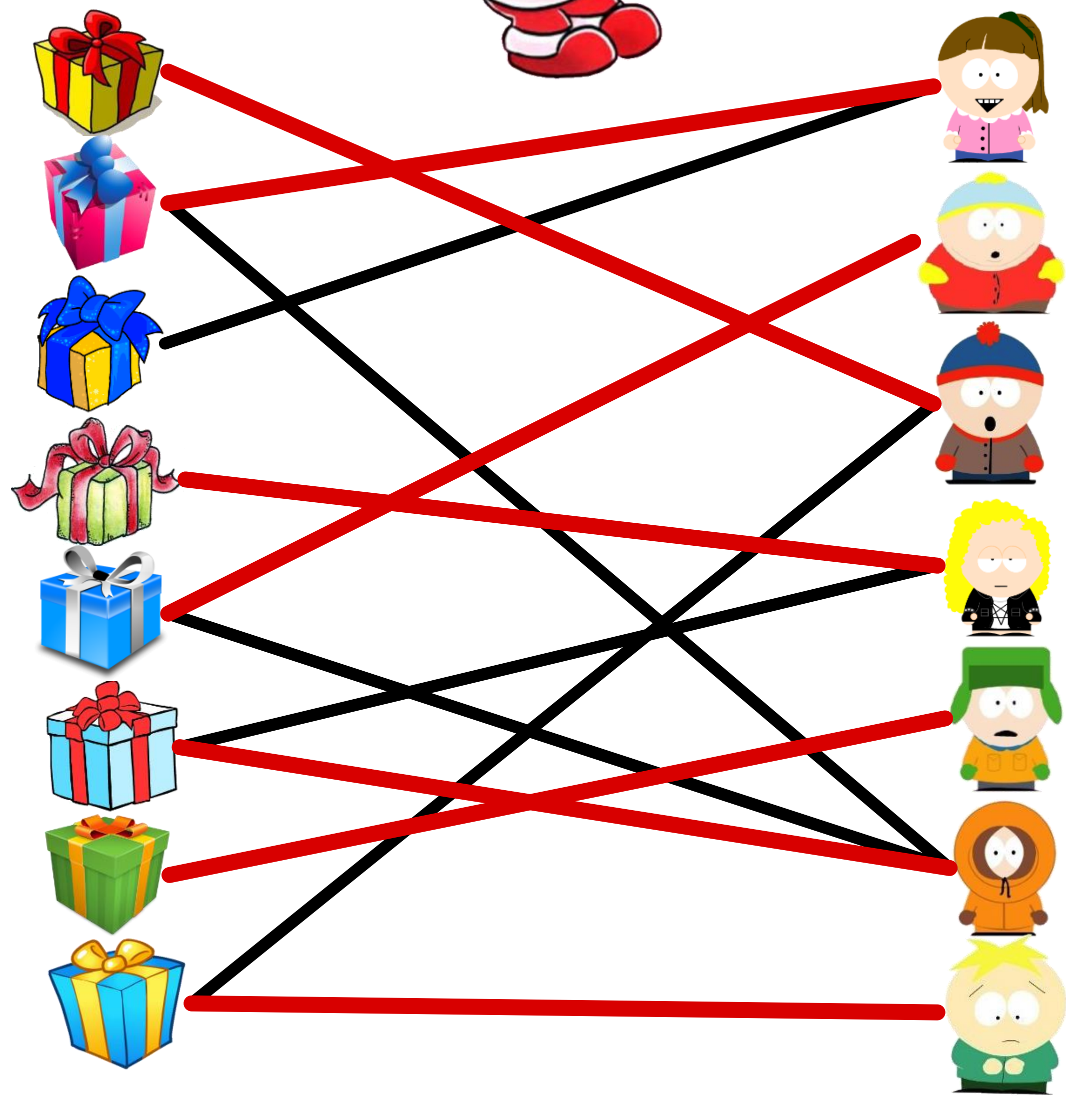
$$|\{u, v\} \cap V_1| = 1.$$

- Thus, edges are only between the two parts



Santa's Problem

Maximum Matching in Bipartite Graphs:



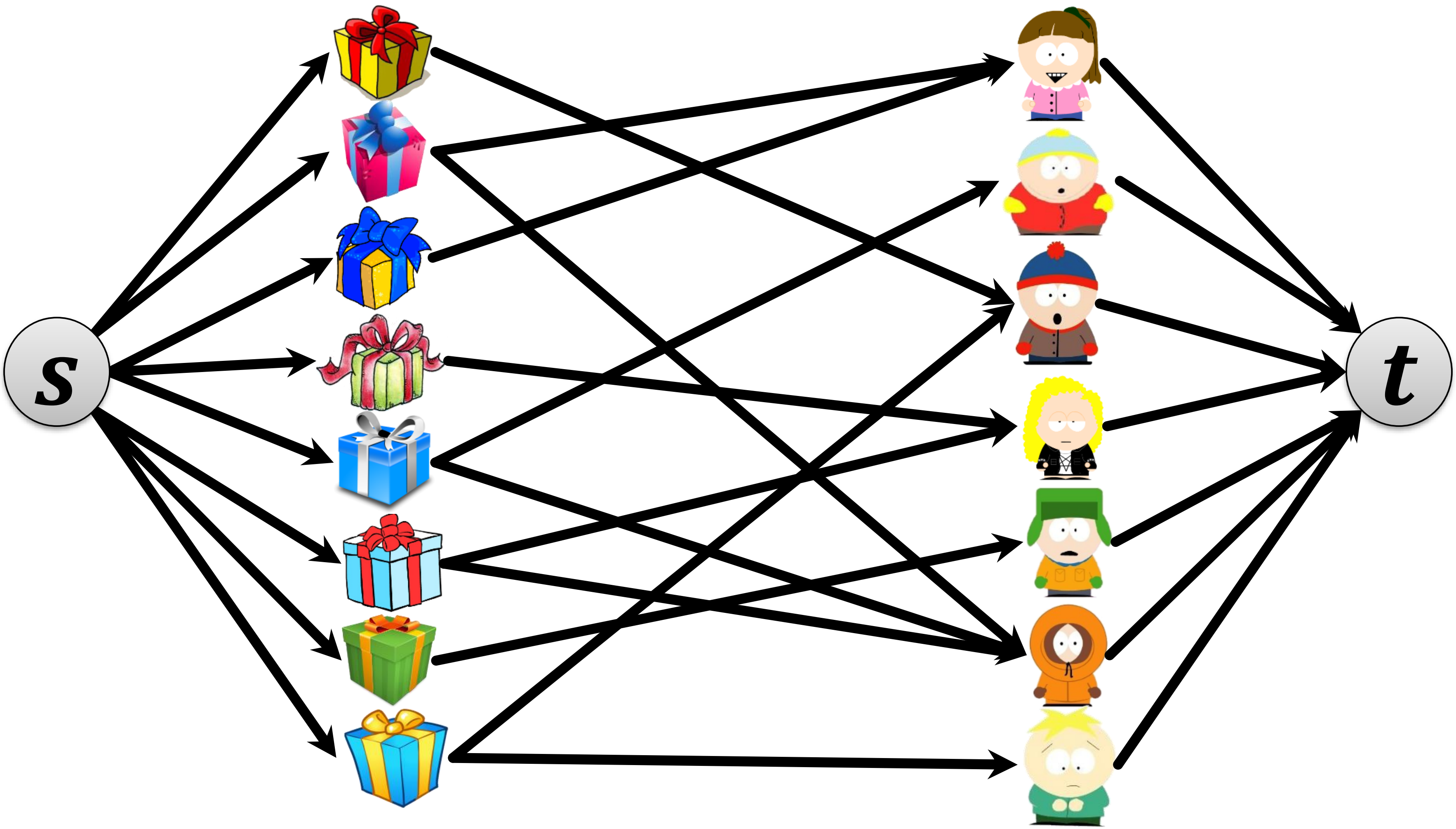
Every child can get a gift iff there is a matching of size $\#\text{children}$

Clearly, every matching is at most as big

If $\#\text{children} = \#\text{gifts}$, there is a solution iff there is a perfect matching

Reducing to Maximum Flow

- Like edge-disjoint paths…



all capacities are 1

Reducing to Maximum Flow

Theorem: Every integer solution to the max flow problem on the constructed graph induces a maximum bipartite matching of G.

Proof:

1. An integer flow f of value $|f|$ induces a matching of size $|f|$
   - Left nodes (gifts) have incoming capacity 1
   - Right nodes (children) have outgoing capacity 1
   - Left and right nodes are incident to ≤ 1 edge e of G with $f(e) = 1$

2. A matching of size k implies a flow f of value $|f| = k$
   - For each edge $\{u, v\}$ of the matching:

$$f\big((s,u)\big) = f\big((u,v)\big) = f\big((v,t)\big) = 1$$

   - All other flow values are 0

Running Time of Max. Bipartite Matching

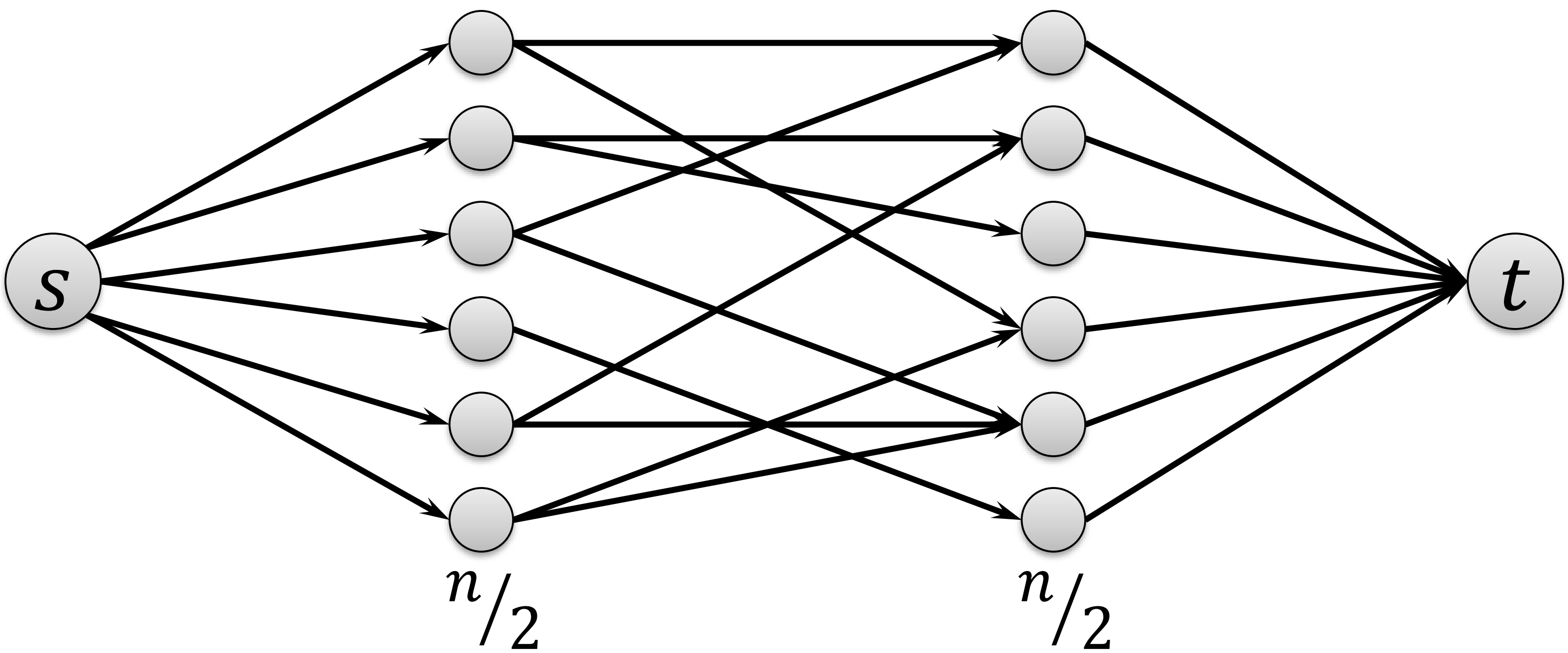
Theorem: A maximum matching M^* of a bipartite graph can be computed in time $O(m \cdot |M^*|) = O(m \cdot n)$.

- The problem can be reduced to a maximum flow problem on a flow network with $O(m)$ edges and all capacities $= 1$
- The Ford-Fulkerson algorithm solves the maximum flow problem in time $O(m \cdot C)$, where C is the value of the maximum flow (i.e., $C = |M^*|$).
- A maximum matching M^* has size $|M^*| \leq {}^n/_2 = O(n)$.

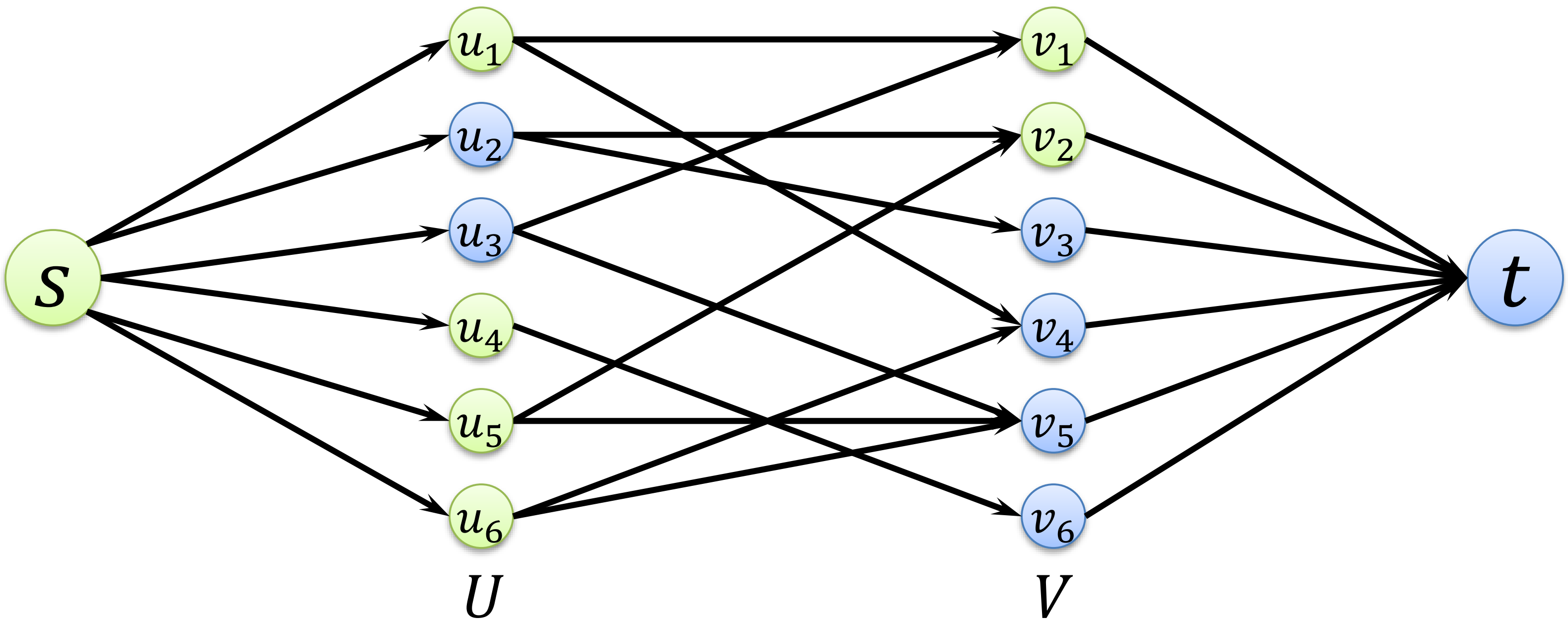
Perfect Matching?

- There can only be a perfect matching if both sides of the partition have size $n/2$.

- There is no perfect matching, iff there is an s-t cut of size $< n/2$ in the flow network.



s-t Cuts



Partition (A, B) of node set such that $s \in A$ and $t \in B$

- If $v_i \in A$: edge (v_i, t) is in cut (A, B)
- If $u_i \in B$: edge (s, u_i) is in cut (A, B)
- Otherwise (if $u_i \in A$, $v_i \in B$), all edges from u_i to some $v_j \in B$ are in cut (A, B)

Hall's Theorem

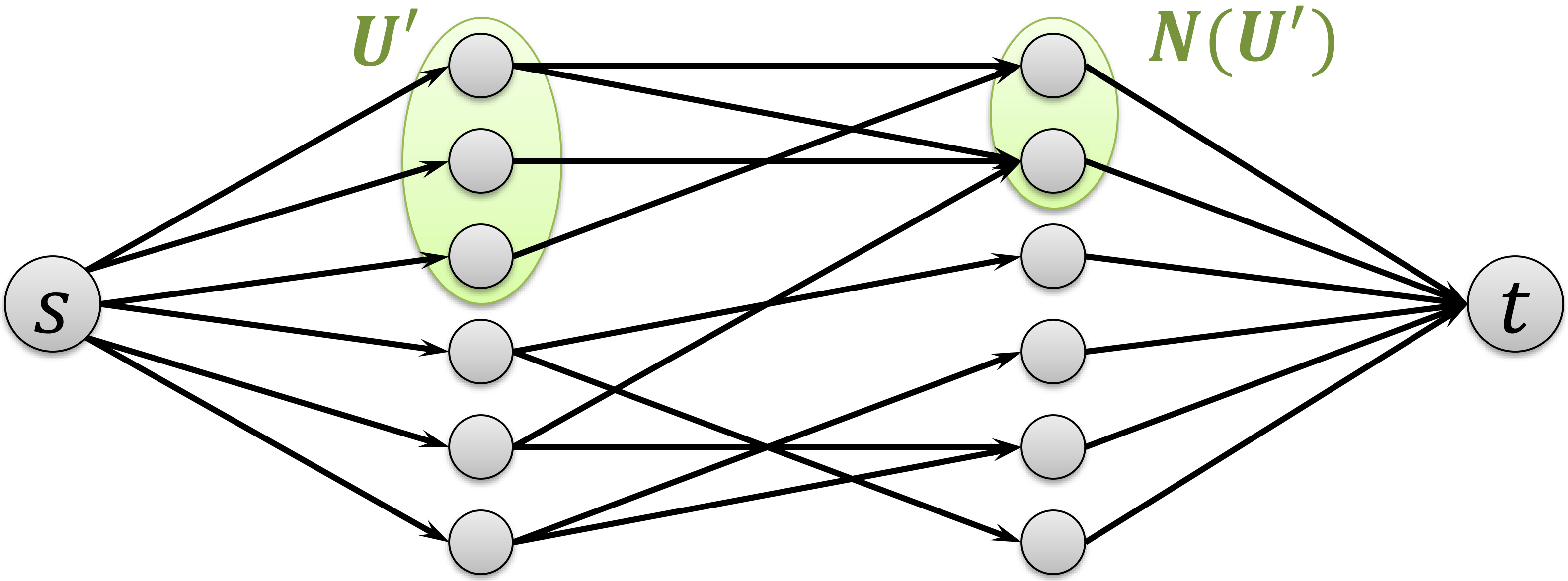
Theorem: A bipartite graph $G = (U \cup V, E)$ for which $|U| = |V|$ has a perfect matching if and only if

$$\forall \boldsymbol{U'} \subseteq \boldsymbol{U}: |\boldsymbol{N}(\boldsymbol{U'})| \geq |\boldsymbol{U'}|,$$

where $N(U') \subseteq V$ is the set of neighbors of nodes in U'.

Proof: No perfect matching $\Leftrightarrow$ some s-t cut has capacity $< n/2$

1. Assume there is U' for which $|N(U')| < |U'|$:



Hall's Theorem

Theorem: A bipartite graph $G = (U \cup V, E)$ for which $|U| = |V|$ has a perfect matching if and only if

$$\forall U' \subseteq U: |N(U')| \geq |U'|,$$

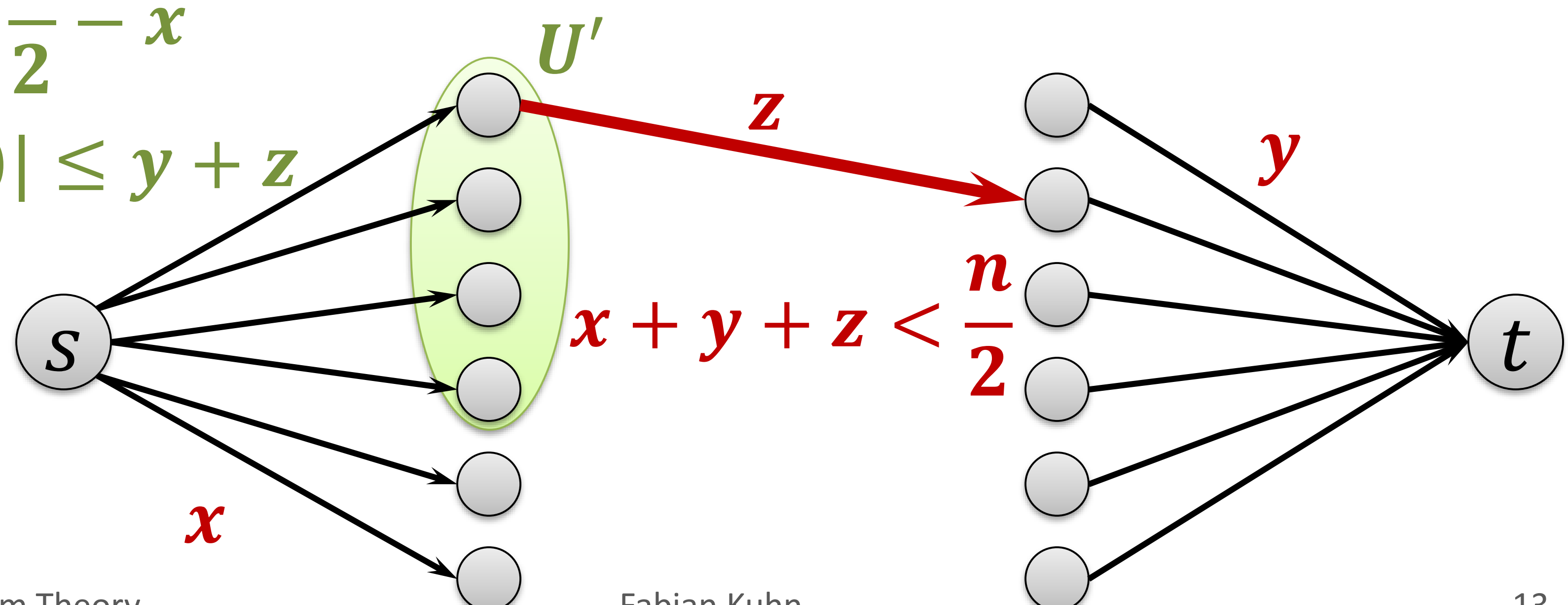
where $N(U') \subseteq V$ is the set of neighbors of nodes in U'.

Proof: No perfect matching $\Leftrightarrow$ some s-t cut has capacity $< n/2$

2. Assume that there is a cut (A, B) of capacity $< n/2$

$$|U'| = \frac{n}{2} - x$$

$$|N(U')| \leq y + z$$



Hall's Theorem

Theorem: A bipartite graph $G = (U \cup V, E)$ for which $|U| = |V|$ has a perfect matching if and only if

$$\forall \boldsymbol{U'} \subseteq \boldsymbol{U}: |\boldsymbol{N}(\boldsymbol{U'})| \geq |\boldsymbol{U'}|,$$

where $N(U') \subseteq V$ is the set of neighbors of nodes in U'.

Proof: No perfect matching $\Leftrightarrow$ some s-t cut has capacity $< n$

2. Assume that there is a cut (A, B) of capacity $< n$

$$\boldsymbol{x} + \boldsymbol{y} + \boldsymbol{z} < \frac{\boldsymbol{n}}{\boldsymbol{2}} \quad \Rightarrow \quad \boldsymbol{y} + \boldsymbol{z} < \frac{\boldsymbol{n}}{\boldsymbol{2}} - \boldsymbol{x}$$

$$\Downarrow$$

$$|\boldsymbol{U'}| = \frac{\boldsymbol{n}}{\boldsymbol{2}} - \boldsymbol{x} \quad \Rightarrow \quad \boldsymbol{y} + \boldsymbol{z} < |\boldsymbol{U'}|$$

$$\Downarrow$$

$$|\boldsymbol{N}(\boldsymbol{U'})| \leq \boldsymbol{y} + \boldsymbol{z} \quad \Rightarrow \quad |\boldsymbol{N}(\boldsymbol{U'})| < |\boldsymbol{U'}|$$