

COL351: Analysis and Design of Algorithms

Ragesh Jaiswal, CSE, IITD

Network Flow

Main Idea in terms of Bipartite Matching

- Main Idea: Reduction

- ① We will obtain an algorithm A for a *Network Flow* problem.
 - We discussed Ford-Fulkerson and there are many more.
- ② Given a new problem, we will *rephrase* this problem as a Network Flow problem.
 - Given a bipartite graph, we constructed a network graph by adding edges from a source to vertices in the left and edges from vertices on the right to a sink etc.
- ③ We will then use algorithm A to solve the rephrased problem and obtain a solution.
 - We used Ford-Fulkerson to obtain a flow with maximum value for the network graph constructed.
- ④ Finally, we build a solution for the original problem using the solution to the rephrased problem.
 - We used the flow to construct a matching and then argued that this will be a maximum matching.

Definition (Perfect Matching)

A given bipartite graph is said to have a perfect matching iff there is a matching such that all vertices are present in the matching.

- Claim 1: If a bipartite graph $G = (X, Y, E)$ has a perfect matching, then $|X| = |Y|$.

Definition (Perfect Matching)

A given bipartite graph is said to have a perfect matching iff there is a matching such that all vertices are present in the matching.

- Claim 1: If a bipartite graph $G = (X, Y, E)$ has a perfect matching, then $|X| = |Y|$.
- For a subset $A \subseteq X$, let $N(A)$ denote the neighboring vertices of A in G .
- Claim 2: There is no perfect matching if there is an A such that $|A| > |N(A)|$.

Definition (Perfect Matching)

A given bipartite graph is said to have a perfect matching iff there is a matching such that all vertices are present in the matching.

- Claim 1: If a bipartite graph $G = (X, Y, E)$ has a perfect matching, then $|X| = |Y|$.
- For a subset $A \subseteq X$, let $N(A)$ denote the neighboring vertices of A in G .
- Claim 2: There is no perfect matching if there is an A such that $|A| > |N(A)|$.
- Is the converse of Claim 2 also true?

Definition (Perfect Matching)

A given bipartite graph is said to have a perfect matching iff there is a matching such that all vertices are present in the matching.

- Claim 1: If a bipartite graph $G = (X, Y, E)$ has a perfect matching, then $|X| = |Y|$.
- For a subset $A \subseteq X$, let $N(A)$ denote the neighboring vertices of A in G .
- Claim 2: There is no perfect matching if there is an A such that $|A| > |N(A)|$.
- Is the converse of Claim 2 also true?

Theorem (Hall's Theorem)

Given any bipartite graph $G = (X, Y, E)$, there is a perfect matching in G if and only if for every subset $A \subseteq X$, we have $|A| \leq |N(A)|$.

Network Flow

Hall's Theorem

Theorem (Hall's Theorem)

Given any bipartite graph $G = (X, Y, E)$, there is a perfect matching in G if and only if for every subset $A \subseteq X$, we have $|A| \leq |N(A)|$.

Proof

- Claim 3: If there is a perfect matching, then for all subsets $A \subseteq X$, $|A| \leq |N(A)|$.

Network Flow

Hall's Theorem

Theorem (Hall's Theorem)

Given any bipartite graph $G = (X, Y, E)$, there is a perfect matching in G if and only if for every subset $A \subseteq X$, we have $|A| \leq |N(A)|$.

Proof

- Claim 3: If there is a perfect matching, then for all subsets $A \subseteq X$, $|A| \leq |N(A)|$.
- Claim 4: If there is no perfect matching, then there is a subset $A \subseteq X$ such that $|A| > |N(A)|$.



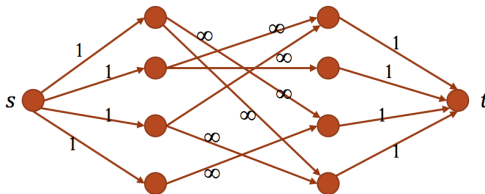
Network Flow

Hall's Theorem

- Claim 4: If there is no perfect matching, then there is a subset $A \subseteq X$ such that $|A| > |N(A)|$.

Proof of Claim 4

- Consider the flow network in figure below constructed using the bipartite graph.
- Claim 4.1: The max-flow in the network is equal to the maximum matching in G .



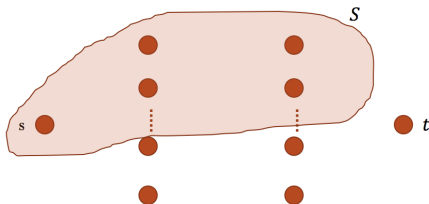
Network Flow

Hall's Theorem

- Claim 4: If there is no perfect matching, then there is a subset $A \subseteq X$ such that $|A| > |N(A)|$.

Proof of Claim 4

- Consider the flow network in figure below constructed using the bipartite graph.
- Claim 4.1: The max-flow in the network is equal to the maximum matching in G .
- Let f be the max integer flow in the network. Consider the residual graph G_f . Let S be the set of vertices reachable from s in G_f . Let A' be vertices of X in S and B' be vertices of Y in S .



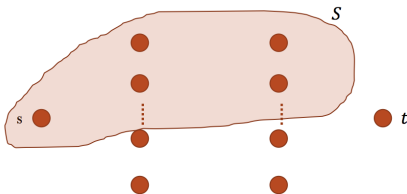
Network Flow

Hall's Theorem

- Claim 4: If there is no perfect matching, then there is a subset $A \subseteq X$ such that $|A| > |N(A)|$.

Proof of Claim 4

- Consider the flow network in figure below constructed using the bipartite graph.
- Claim 4.1: The max-flow in the network is equal to the maximum matching in G .
- Let f be the max integer flow in the network. Consider the residual graph G_f . Let S be the set of vertices reachable from s in G_f . Let A' be vertices of X in S and B' be vertices of Y in S .
- Claim 4.2: $B' = N(A')$.



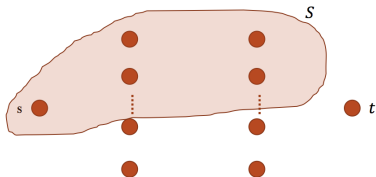
Network Flow

Hall's Theorem

- Claim 4: If there is no perfect matching, then there is a subset $A \subseteq X$ such that $|A| > |N(A)|$.

Proof of Claim 4

- Consider the flow network in figure below constructed using the bipartite graph.
- Claim 4.1: The max-flow in the network is equal to the maximum matching in G .
- Let f be the max integer flow in the network. Consider the residual graph G_f . Let S be the set of vertices reachable from s in G_f and T all remaining vertices. Let A' be vertices of X in S and B' be vertices of Y in S .
- Claim 4.2: $B' = N(A')$.
- Capacity of the cut $(S, T) = n - |A'| + |N(A')|$.



Network Flow

Hall's Theorem

- Claim 4: If there is no perfect matching, then there is a subset $A \subseteq X$ such that $|A| > |N(A)|$.

Proof of Claim 4

- Consider the flow network in figure below constructed using the bipartite graph.
- Claim 4.1: The max-flow in the network is equal to the maximum matching in G .
- Let f be the max integer flow in the network. Consider the residual graph G_f . Let S be the set of vertices reachable from s in G_f and T all remaining vertices. Let A' be vertices of X in S and B' be vertices of Y in S .
- Claim 4.2: $B' = N(A')$.
- Capacity of the cut $(S, T) = n - |A'| + |N(A')|$.
- From Max-flow-min-cut argument, we have:
$$n - |A'| + |N(A')| = \text{max flow} < n \Rightarrow |A'| > |N(A')|.$$



Network Flow

Hall's Theorem

- Claim 4: If there is no perfect matching, then there is a subset $A \subseteq X$ such that $|A| > |N(A)|$.

Proof of Claim 4

- Consider the flow network in figure below constructed using the bipartite graph.
- Claim 4.1: The max-flow in the network is equal to the maximum matching in G .
- Let f be the max integer flow in the network. Consider the residual graph G_f . Let S be the set of vertices reachable from s in G_f and T all remaining vertices. Let A' be vertices of X in S and B' be vertices of Y in S .
- Claim 4.2: $B' = N(A')$.
- Capacity of the cut $(S, T) = n - |A'| + |N(A')|$.
- From Max-flow-min-cut argument (Slide 4 of this lecture), we have:
$$n - |A'| + |N(A')| = \text{max flow} < n \Rightarrow |A'| > |N(A')|.$$



- This is a constructive proof since we can find a subset A' such that $|A'| > |N(A')|$.
- Such an A' may be interpreted as a *certificate* of the fact that there is no perfect matching in G .

- Suppose there are four teams in IPL with their current number of wins:
 - Daredevils: 10
 - Sunrisers: 10
 - Lions: 10
 - Supergiants: 8
- There are 7 more games to be played. These are as follows:
 - Supergiants plays all other 3 teams.
 - Daredevils Vs Sunrisers, Sunrisers Vs Lions, Daredevils Vs Lions, Sunrisers Vs Daredevils

- Suppose there are four teams in IPL with their current number of wins:
 - Daredevils: 10
 - Sunrisers: 10
 - Lions: 10
 - Supergiants: 8
- There are 7 more games to be played. These are as follows:
 - Supergiants plays all other 3 teams.
 - Daredevils Vs Sunrisers, Sunrisers Vs Lions, Daredevils Vs Lions, Sunrisers Vs Daredevils
- A team is said to be eliminated if it cannot end with maximum number of wins.
- Can we say that Supergiants have been eliminated give the current scenario?

- Suppose there are four teams in IPL with their current number of wins:
 - Daredevils: 10
 - Sunrisers: 10
 - Lions: 9
 - Supergiants: 8
- There are 7 more games to be played. These are as follows:
 - Supergiants plays all other 3 teams.
 - 4 games between Daredevils and Sunrisers.
- Can we say that Supergiants have been eliminated give the current scenario?

Problem

There are n teams. Each team i has a current number of wins denoted by $w(i)$. There are $G(i, j)$ games yet to be played between team i and j . Design an algorithm to determine whether a given team x has been eliminated.

Network Flow

Team Elimination

Problem

There are n teams. Each team i has a current number of wins denoted by $w(i)$. There are $G(i, j)$ games yet to be played between team i and j . Design an algorithm to determine whether a given team x has been eliminated.

- Consider the following flow network

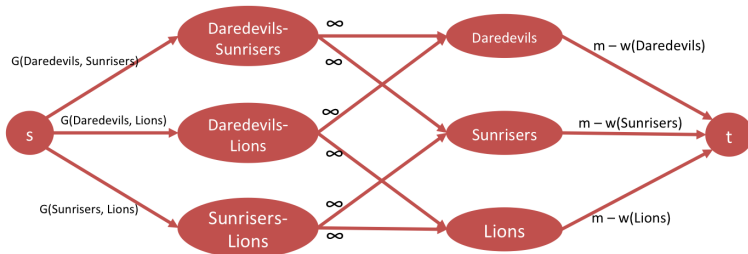


Figure: Team x can end with at most m wins, i.e., $m = w(x) + \sum_j G(x, j)$

Network Flow

Team Elimination

Problem

There are n teams. Each team i has a current number of wins denoted by $w(i)$. There are $G(i, j)$ games yet to be played between team i and j . Design an algorithm to determine whether a given team x has been eliminated.

- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i, j \text{ s.t. } x \notin \{i, j\}} G(i, j)$.

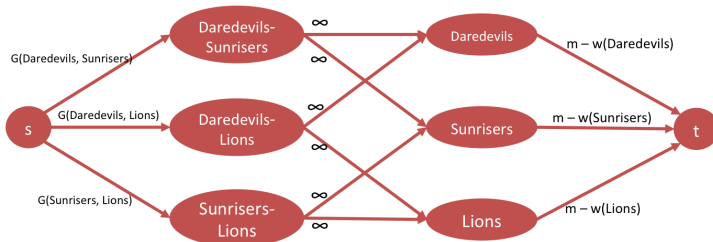


Figure: Team x can end with at most m wins, i.e., $m = w(x) + \sum_j G(x, j)$

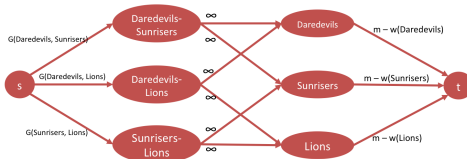
Network Flow

Team Elimination

Problem

There are n teams. Each team i has a current number of wins denoted by $w(i)$. There are $G(i, j)$ games yet to be played between team i and j . Design an algorithm to determine whether a given team x has been eliminated.

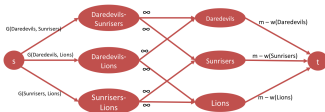
- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i, j \text{ s.t. } x \notin \{i, j\}} G(i, j)$.
- Comment: If we can somehow find a subset T of teams (not including x) such that $\sum_{i \in T} w(i) + \sum_{i < j \text{ and } i, j \in T} G(i, j) > m \cdot |T|$. Then we have a witness to the fact that x has been eliminated.



Problem

There are n teams. Each team i has a current number of wins denoted by $w(i)$. There are $G(i, j)$ games yet to be played between team i and j . Design an algorithm to determine whether a given team x has been eliminated.

- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i,j \text{ s.t. } x \notin \{i,j\}} G(i, j)$.
- Comment: If we can somehow find a subset T of teams (not including x) such that $\sum_{i \in T} w(i) + \sum_{i < j \text{ and } i, j \in T} G(i, j) > m \cdot |T|$. Then we have a witness to the fact that x has been eliminated.
- Can we find such a subset T ?



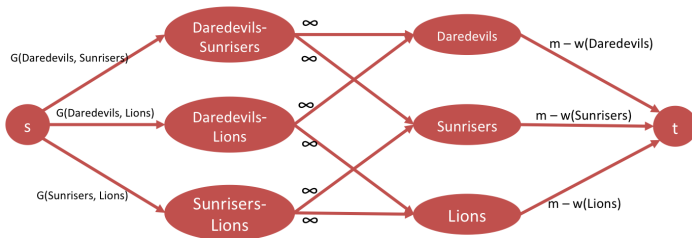
Network Flow

Team Elimination

- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i,j \text{ s.t. } x \notin \{i,j\}} G(i,j)$.

Proof.

- Claim 1.1: If x has been eliminated, then the max flow in the network is $< g^*$.



Network Flow

Team Elimination

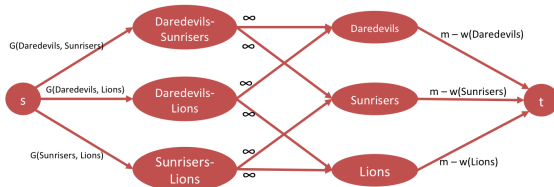
- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i,j \text{ s.t. } x \notin \{i,j\}} G(i,j)$.

Proof of Claim 1

- Claim 1.1: If x has been eliminated, then the max flow in the network is $< g^*$.
- Claim 1.2: If the max flow is $< g^*$, then team x has been eliminated.

Proof of Claim 1.2

- Consider any s - t min-cut (A, B) in the graph.
- Claim 1.2.1: If v_{ij} is in A , then both v_i and v_j are in A .



Network Flow

Team Elimination

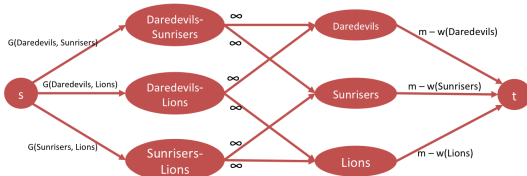
- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i,j \text{ s.t. } x \notin \{i,j\}} G(i,j)$.

Proof of Claim 1

- Claim 1.1: If x has been eliminated, then the max flow in the network is $< g^*$.
- Claim 1.2: If the max flow is $< g^*$, then team x has been eliminated.

Proof of Claim 1.2

- Consider any s - t min-cut (A, B) in the graph.
- Claim 1.2.1: If v_{ij} is in A , then both v_i and v_j are in A .
- Claim 1.2.2: If both v_i and v_j are in A , then v_{ij} is in A .



Network Flow

Team Elimination

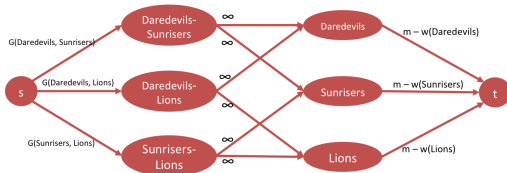
- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i,j \text{ s.t. } x \notin \{i,j\}} G(i,j)$.

Proof of Claim 1

- Claim 1.1: If x has been eliminated, then the max flow in the network is $< g^*$.
- Claim 1.2: If the max flow is $< g^*$, then team x has been eliminated.

Proof of Claim 1.2

- Consider any s - t min-cut (A, B) in the graph.
- Claim 1.2.1: If v_{ij} is in A , then both v_i and v_j are in A .
- Claim 1.2.2: If both v_i and v_j are in A , then v_{ij} is in A .
- Let T be the set of teams such that $i \in T$ **iff** $v_i \in A$.



Network Flow

Team Elimination

- Claim 1: Team x has been eliminated **iff** the maximum flow in the network is $< g^*$, where $g^* = \sum_{i,j \text{ s.t. } x \notin \{i,j\}} G(i,j)$.

Proof of Claim 1

- Claim 1.1: If x has been eliminated, then the max flow in the network is $< g^*$.
- Claim 1.2: If the max flow is $< g^*$, then team x has been eliminated.

Proof of Claim 1.2

- Consider any s - t min-cut (A, B) in the graph.
- Claim 1.2.1: If v_{ij} is in A , then both v_i and v_j are in A .
- Claim 1.2.2: If both v_i and v_j are in A , then v_{ij} is in A .
- Let T be the set of teams such that $i \in T$ **iff** $v_i \in A$. Then we have:

$$\begin{aligned} C(A, B) &= \sum_{i \in T} (m - w(i)) + \sum_{\{i,j\} \not\subset T} G(i,j) < g^* \\ \Rightarrow m \cdot |T| - \sum_{i \in T} w(i) + (g^* - \sum_{\{i,j\} \subset T} G(i,j)) &< g^* \\ \Rightarrow \sum_{i \in T} w(i) + \sum_{\{i,j\} \subset T} G(i,j) &> m \cdot |T| \quad \square \end{aligned}$$



End