CSL 356: Analysis and Design of Algorithms

Ragesh Jaiswal

CSE, IIT Delhi

Computational Intractability

NP and NP-completeness

Computational Intractability: NP & NP-complete

- **NP**: A problem X is in **NP** if and only if there is an *efficient certifier* for X.
- **NP**-complete: These are all problems X with the following properties:
  1. X is in **NP**.
  2. For all Y in **NP**, $Y \leq_p X$.
- Theorem (Cook-Levin): 3-SAT is **NP**-complete.
- **NP**-hard: These are all problems X with the following property:
  1. For all Y in **NP**, $Y \leq_p X$.
  - Example: Given a graph G, find the maximum independent set in G.

Computational Intractability

NP-complete problems

Computational Intractability: NP-complete problems

- We now know that 3-SAT is **NP**-complete.
- Claim: Independent-set, Vertex-cover, Set-cover are also **NP**-complete.

Computational Intractability: NP-complete problems

- We now know that 3-SAT is **NP**-complete.
- Claim: Independent-set, Vertex-cover, Set-cover are also **NP**-complete.
  - Proof: These problem are in **NP** and

    3-SAT $\leq_p$ Independent-set $\leq_p$ Vertex-cover $\leq_p$ Set-cover.

Computational Intractability: NP-complete problems

- Problem (TSP): Given a *complete,* weighted, directed graph G and a number k, determine if there is a tour in the graph of total length at most k.
- Claim 1: TSP is in **NP**.
  - Proof: A tour of length at most k is a certificate.

Computational Intractability: NP-complete problems

- Problem (TSP): Given a *complete*, weighted, directed graph G and a number k, determine if there is a tour in the graph of total length at most k.
- Claim 1: TSP is in **NP**.
  - Proof: A tour of length at most k is a certificate.
- Claim 2: 3-SAT $\leq_p$ TSP
  - Proof:
    - Claim 2.1: 3-SAT $\leq_p$ Hamiltonian-cycle.
    - Claim 2.2: Hamiltonian-cycle $\leq_p$ TSP.

Computational Intractability: NP-complete problems

- Problem (TSP): Given a *complete*, weighted, directed graph G and a number k, determine if there is a tour in the graph of total length at most k.
- Claim 1: TSP is in **NP**.
  - Proof: A tour of length at most k is a certificate.
- Claim 2: 3-SAT $\leq_p$ TSP
  - Proof:
    - Claim 2.1: 3-SAT $\leq_p$ Hamiltonian-cycle.
    - Claim 2.2: Hamiltonian-cycle $\leq_p$ TSP.
- Problem (Hamiltonian-cycle): Given an unweighted, directed graph, determine if there is a Hamiltonian cycle in the graph.
  - Hamiltonian cycle: A cycle that visits each vertex exactly once.

Computational Intractability: NP-complete problems

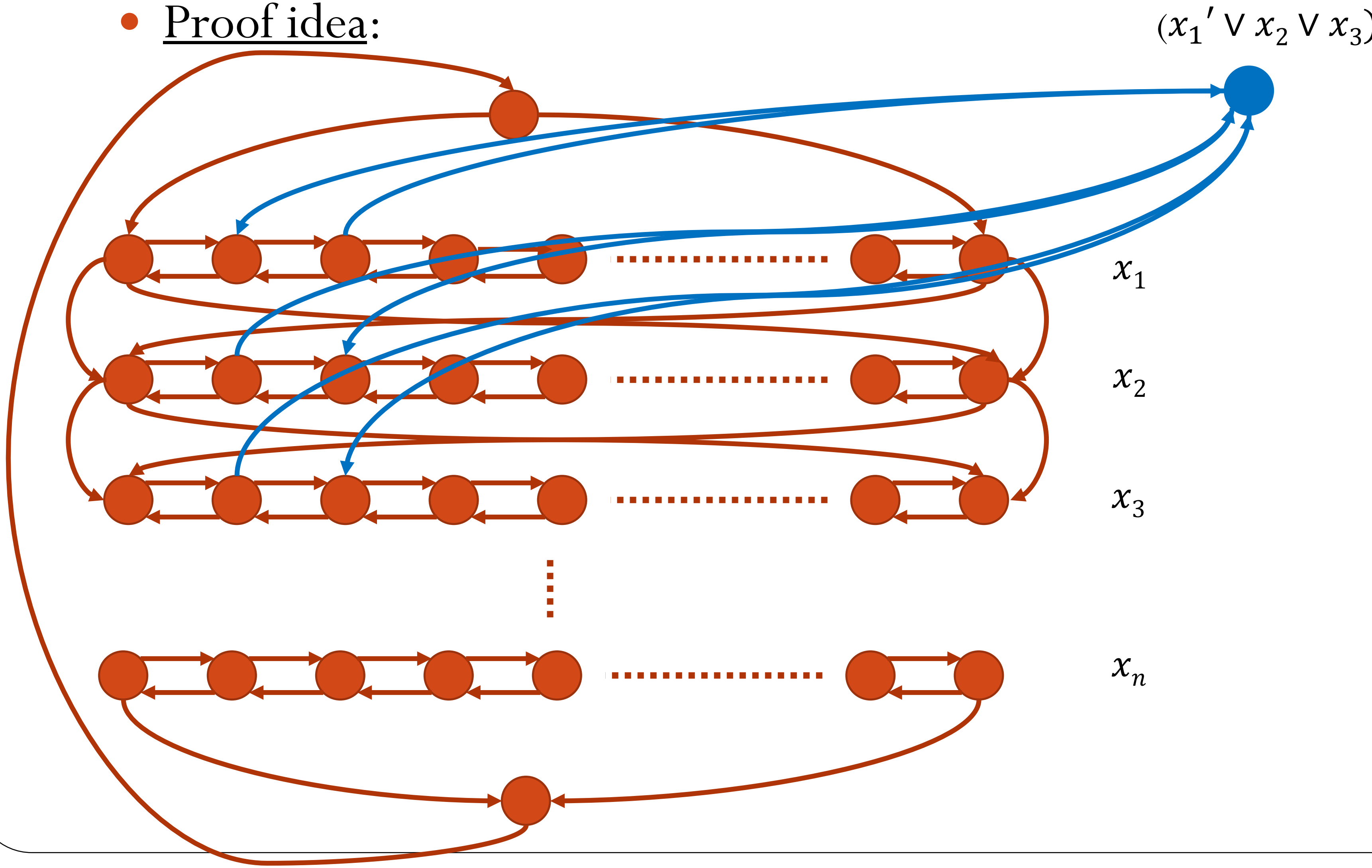
- Claim 2.2: Hamiltonian-cycle $\leq_p$ TSP.
  - Proof: Given a unweighted, directed graph G, construct the following complete, directed, weighted graph G:
    - For each edge (u, v) in G give the weight of 1 to edge (u, v) in G'.
    - For each pair (u, v) such that there is no edge from u to v in G, add an edge (u, v) with weight 2 in G'.
    - Claim2.2.1: G has a Hamiltonian cycle if and only if G' has a tour of size at most n.

Computational Intractability: NP-complete problems

- Claim 2.1: 3-SAT $\leq_p$ Hamiltonian-cycle.
  - Proof: Given an instance of the 3-SAT problem (a formula Ω with n variables and m clauses), we need to create a directed graph G such that Ω is satisfiable if and only if G has a Hamiltonian cycle.

Computational Intractability: NP-complete problems

- Claim 2.1: 3-SAT $\leq_p$ Hamiltonian-cycle.
  - Proof idea:



Computational Intractability: NP-complete problems

- Claim 2.1.1: If the 3-SAT formula is satisfiable, then there is a Hamiltonian cycle in the constructed graph.

Computational Intractability: NP-complete problems

- Claim 2.1.1: If the 3-SAT formula is satisfiable, then there is a Hamiltonian cycle in the constructed graph.
- Claim 2.1.2: If the constructed graph has a Hamiltonian cycle, then the 3-SAT formula has a satisfying assignment.

Computational Intractability: NP-complete problems

- Claim 2.1.1: If the 3-SAT formula is satisfiable, then there is a Hamiltonian cycle in the constructed graph.
- Claim 2.1.2: If the constructed graph has a Hamiltonian cycle, then the 3-SAT formula has a satisfying assignment.
  - Main idea: Any Hamiltonian cycle in the constructed graph goes through each of the bi-directional chains only in one direction.

Computational Intractability

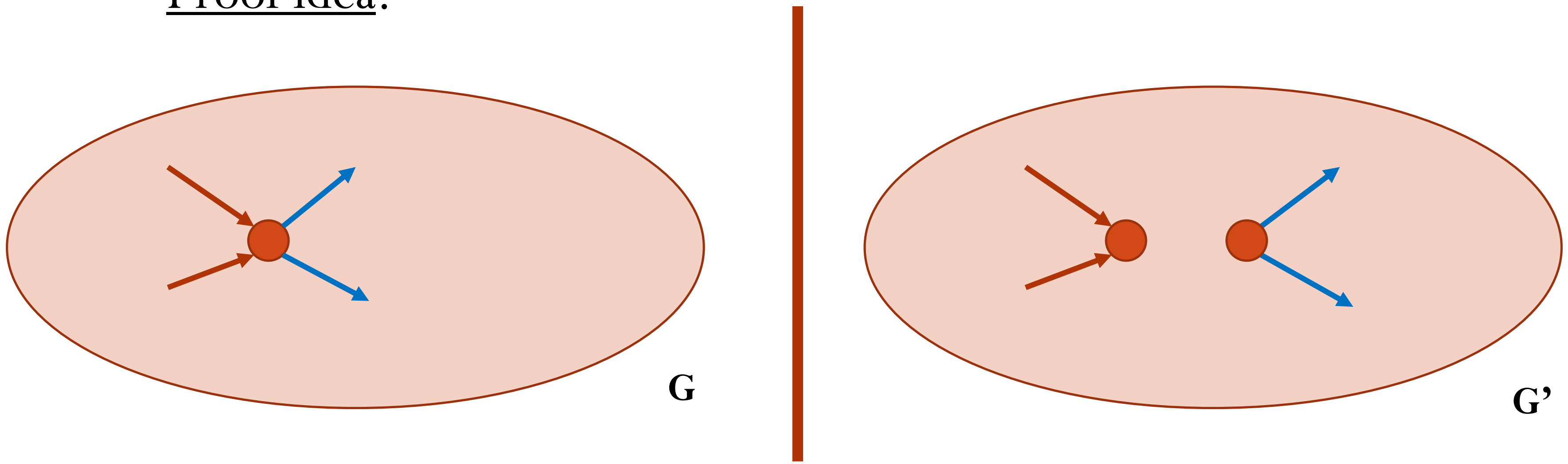
NP-complete problems: Hamiltonian Path

Computational Intractability: NP-complete problems

- Problem (Hamiltonian-path): Given a directed graph G, determine if there is a Hamiltonian path in the graph.
  - Hamiltonian path: A path that visits each vertex exactly once.
- Claim 1: Hamiltonian-path is in **NP**.
  - Proof: A Hamiltonian path acts as a certificate.
- Claim 2: Hamiltonian-cycle $\leq_p$ Hamiltonian-path.

Computational Intractability: NP-complete problems

- Problem (Hamiltonian-path): Given a directed graph G, determine if there is a Hamiltonian path in the graph.
  - Hamiltonian path: A path that visits each vertex exactly once.
- Claim 1: Hamiltonian-path is in **NP**.
  - Proof: A Hamiltonian path acts as a certificate.
- Claim 2: Hamiltonian-cycle $\leq_p$ Hamiltonian-path.
  - Proof idea:

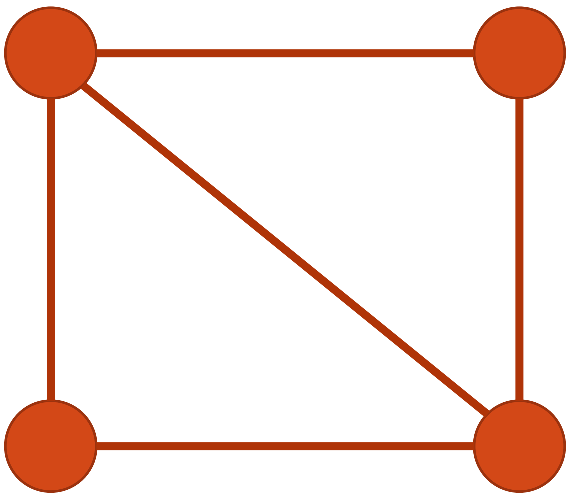


Computational Intractability

NP-complete problems: k-COLORING

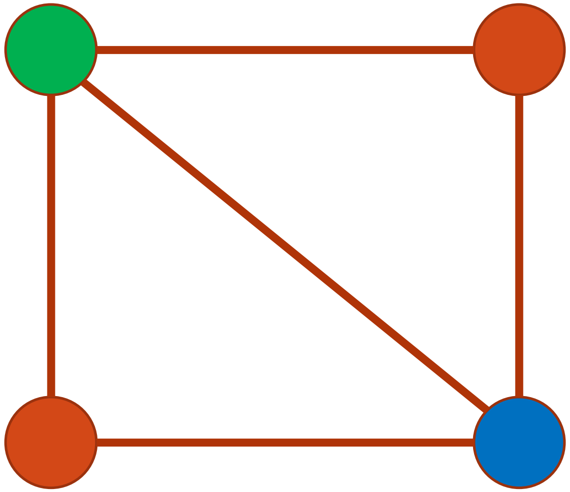
Computational Intractability: NP-complete problems

- Problem (k-coloring): Given a graph G, determine if G is k-colorable.
  - k-colorable: A graph is said to be k colorable if it is possible to assign one of k colors to each node such that for every edge (u, v), u and v are assigned different colors.



Computational Intractability: NP-complete problems

- Problem (k-coloring): Given a graph G, determine if G is k-colorable.
  - k-colorable: A graph is said to be k colorable if it is possible to assign one of k colors to each node such that for every edge (u, v), u and v are assigned different colors.



Computational Intractability: NP-complete problems

- Problem (2-coloring): Given a graph G, determine if G is 2-colorable.
- How hard is this problem?

Computational Intractability: NP-complete problems

- Problem (2-coloring): Given a graph G, determine if G is 2-colorable.
- How hard is this problem?
- Claim: G is 2-colorable if and only if G is bipartite.

End