CS 381 – FALL 2019

Week 15.1, Monday, Nov 25

Homework 7 Due Tomorrow: November 26th at 11:59PM on Gradescope

Monday Office Hours (Today): ~~2:30-3:30PM~~ 5:30-6:30PM
No Class on Wed/Fri (Happy Thanksgiving!)
No PSOs this Week

iClicker

Assume P ≠ NP. Which of the following problems are not in NP ∩ co-NP?

↑ Should have assumed coNP ≠ NP

A. FACTOR
- FACTOR is in NP ∩ co-NP (but not known to be in P)
  (prime factorization is either a witness/disqualifier)

B. PRIMES
- Primes is in P (contained in NP ∩ co-NP)

C. 3COLOR
- If 3COLOR is in co-NP then co-NP = NP since 3COLOR is NP-Complete
- More precise problems statement: ``Assume coNP ≠ NP"

D. BIPARTITE MATCHING (Given a bipartite graph G and integer k is there a matching that contains at least k edges)?
- Primes is in P (contained in NP ∩ co-NP)

E. All of the problems are in NP ∩ co-NP
- This is actually true if NP = co-NP!
- (Full Credit if you picked choice E since we don't know)

Partition

SUBSET-SUM. Given natural numbers $w_1, \ldots, w_n$ and an integer W, is there a subset that adds up to exactly W?

PARTITION. Given natural numbers $v_1, \ldots, v_m$, can they be partitioned into two subsets that add up to the same value?

$\uparrow$ $\frac{1}{2} \Sigma_i v_i$

Claim. SUBSET-SUM $\leq_P$ PARTITION.

Pf. Let $W, w_1, \ldots, w_n$ be an instance of SUBSET-SUM.

- Create instance of PARTITION with $m = n+2$ elements.
  - $v_1 = w_1, v_2 = w_2, \ldots, v_n = w_n, \quad v_{n+1} = 2 \Sigma_i w_i - W, \quad v_{n+2} = \Sigma_i w_i + W$

- There exists a subset that sums to W iff there exists a partition since two new elements cannot be in the same partition. ▪

iClicker

Assume co-NP $\neq$ NP. Which of the following claims are necessarily true?

1. PARTITION is not in P
2. PRIMES is in co-NP
3. TAUTOLOGY is in coNP

A. Claim 1 only

B. Claim 2 only

C. Claim 3 only

D. Claims 1 and 3

E. All of the claims are true

iClicker

Assume co-NP $\neq$ NP. Which of the following claims are necessarily true?

1. PARTITION is not in P
2. PRIMES is in co-NP
3. TAUTOLOGY is in coNP

A. Claim 1 only
- If PARTITION were in P then P=NP since PARTITION is NP-Complete. This would imply co-NP=NP.

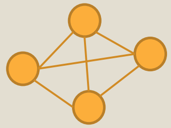
B. Claim 2 only
- PRIMES is in P which is contained in co-NP.

C. Claim 3 only
- TAUTOLOGY is in coNP
- We could also conclude that TAUTOLOGY is not in NP (otherwise we would have co-NP = NP since TAUTOLOGY is coNP-complete)

D. Claims 1 and 3

E. All of the claims are true

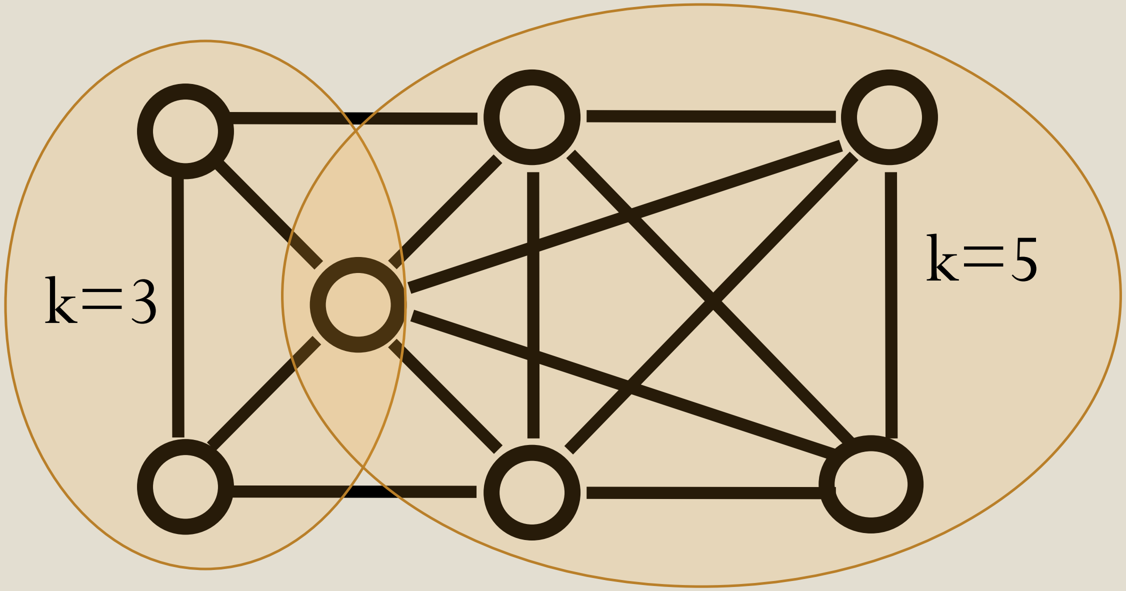
Let G be an undirected graph. A clique of size k is a complete graph on k vertices.



K_4: Clique of size 4

Clique (decision) problem:

Given G and an integer k, does G contain a subgraph that is a clique of size k?

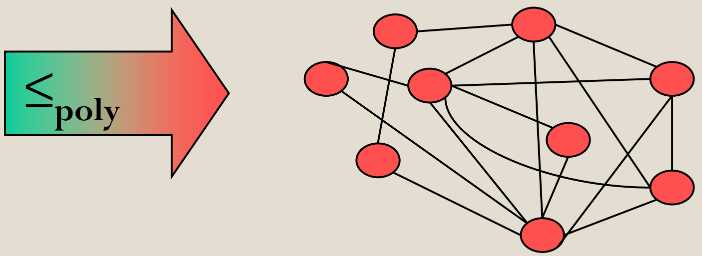


Claim: The clique problem is NP-complete

Proof:

1. The clique problem is in NP.
   - The certificate is a set of k vertices.
   - We need to check whether the k vertices induce a complete graph of size k in G. Check that all possible edges between these k vertices are present.
2. Choose an NP-complete problem: 3-SAT
3. Show that **3-SAT $\leq_{poly}$ Clique**

$(.. \vee .. \vee ..) \wedge \ldots \wedge (.. \vee .. \vee ..)$

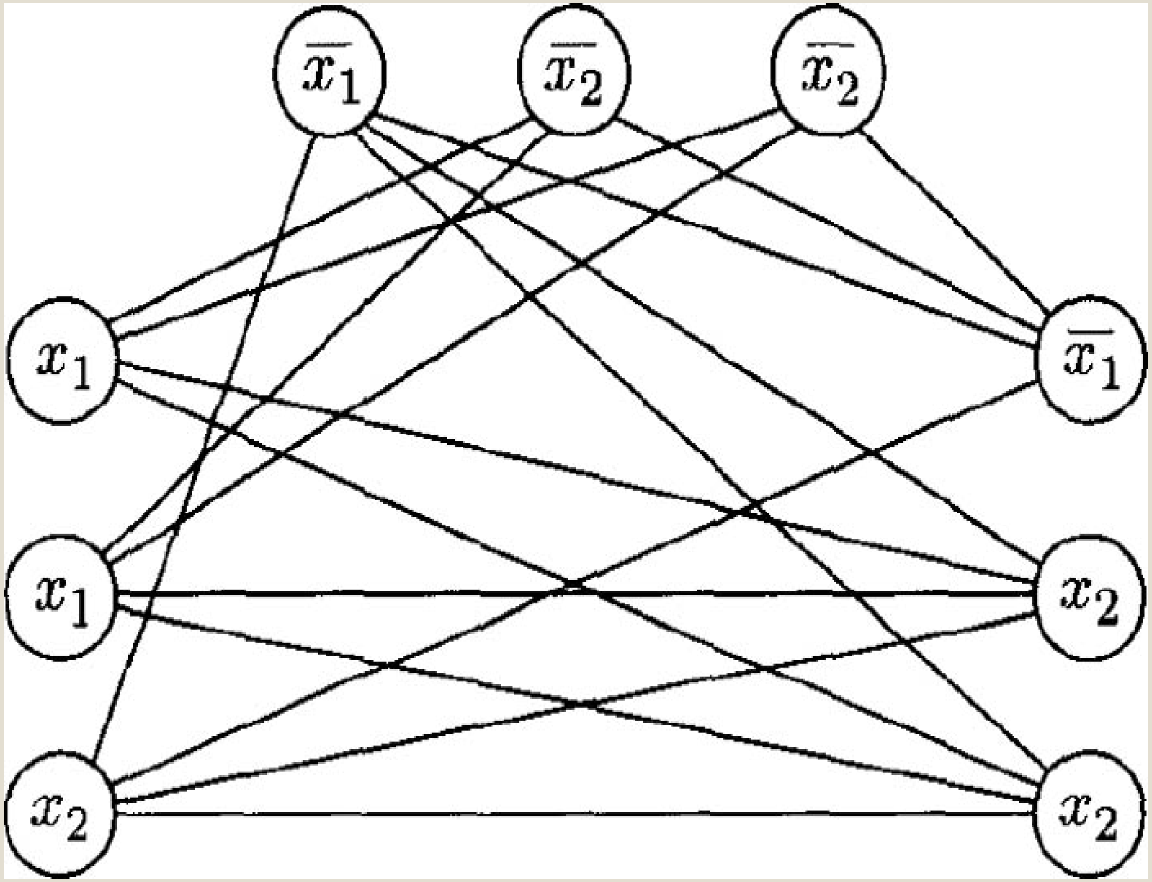


Transformation

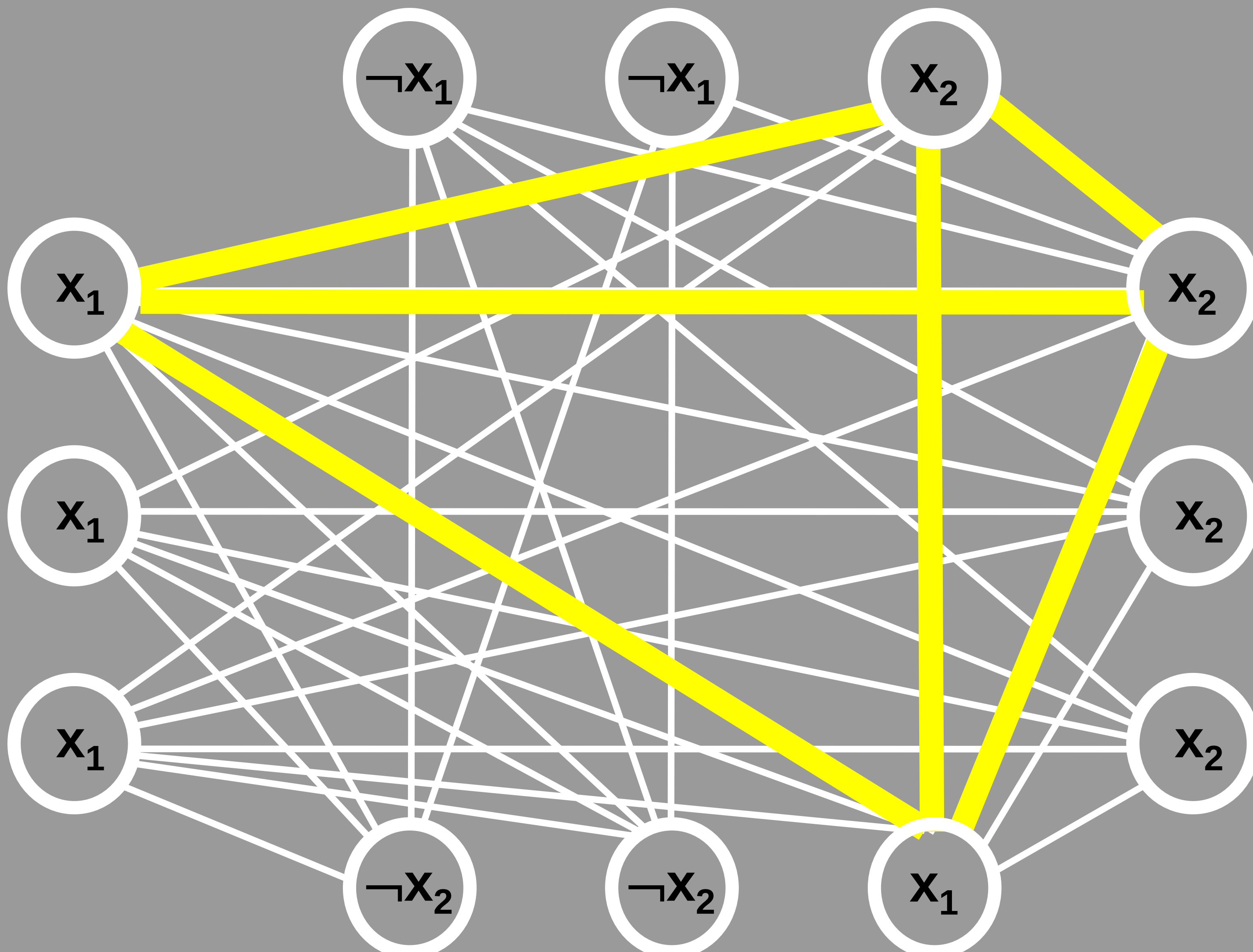
- Given is a 3-SAT formula C consisting of p clauses
- For every clause, create three **vertices** and label them with the literals in the clause
  - creates 3p vertices
- Form the **edges**: Connect each vertex to the literals in other clauses that are not the negation of it
- Set k=p

Is a polynomial time transformation generating from formula C an undirected graph G and an integer k

$$\phi = (x_1 \vee x_1 \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2 \vee x_2)$$



$$(x_1 \vee x_1 \vee x_1) \wedge (\neg x_1 \vee \neg x_1 \vee x_2) \wedge (x_2 \vee x_2 \vee x_2) \wedge (\neg x_2 \vee \neg x_2 \vee x_1)$$



Graph G has a clique of size k iff formula C can be satisfied.

$\Leftarrow$ *Assume G contains a clique of size k*

- To have a clique of size k, every triplet of vertices corresponding to a clause must contribute exactly one vertex (no edges connecting this triplet)
- If a vertex in the clique corresponds to an unnegated variable, set x_i = true; if negated, set it to false.
- no edges between $\neg x_i$ and x_i → assignment is consistent

This gives an assignment that makes every one of the k=p clauses true and thus formula C is satisfiable.

Graph G has a clique of size k iff formula C can be satisfied.

$\Rightarrow$ *Assume the formula is satisfiable*

Then, at least one literal in every clause is true.

Since x and its negation cannot be true at the same time, graph G contains an edge between every pair of literals set true.

Hence, we have a clique of size p=k.

$\Rightarrow$ (For each clause pick one ``true" node from the corresponding triplet)

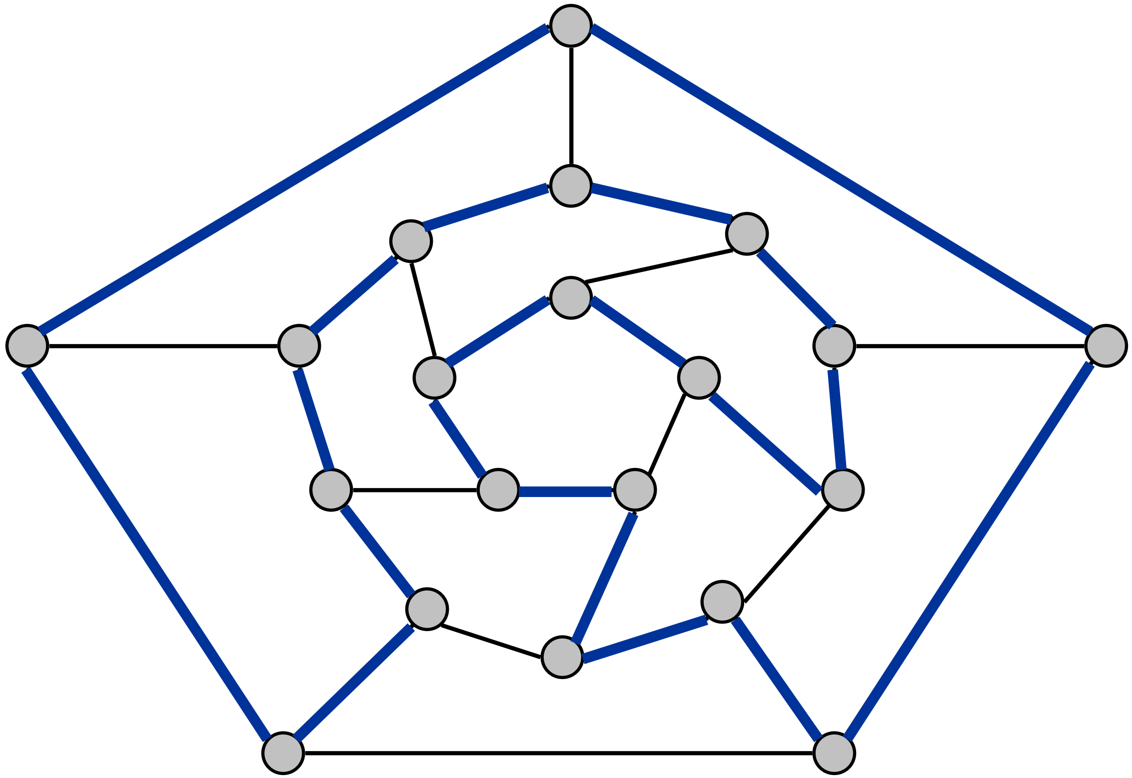
8.5 Sequencing Problems

Basic genres.

- Packing problems: SET-PACKING, INDEPENDENT SET.
- Covering problems: SET-COVER, VERTEX-COVER.
- Constraint satisfaction problems: SAT, 3-SAT.
- Sequencing problems: HAMILTONIAN-CYCLE, TSP.
- Partitioning problems: 3D-MATCHING, 3-COLOR.
- Numerical problems: SUBSET-SUM, KNAPSACK.

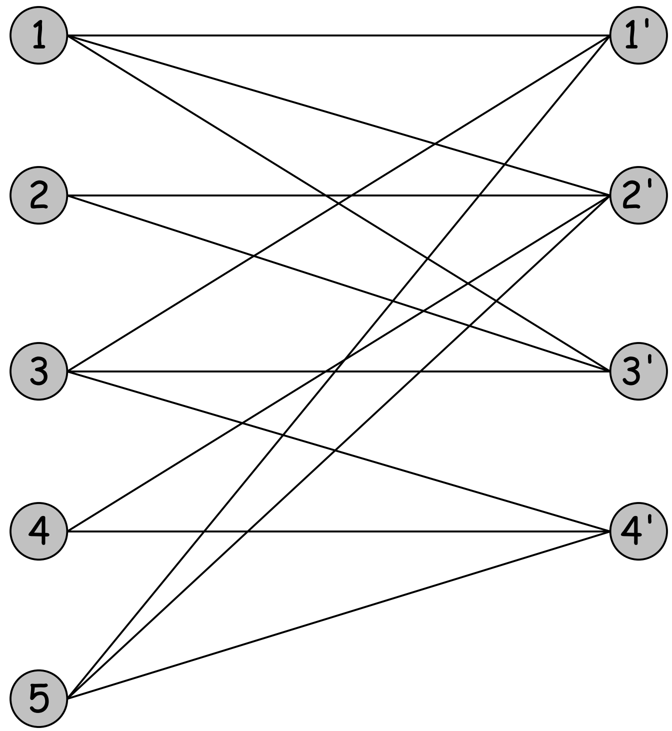
Hamiltonian Cycle

HAM-CYCLE: given an undirected graph G = (V, E), does there exist a simple cycle Γ that contains every node in V.



YES: vertices and faces of a dodecahedron.

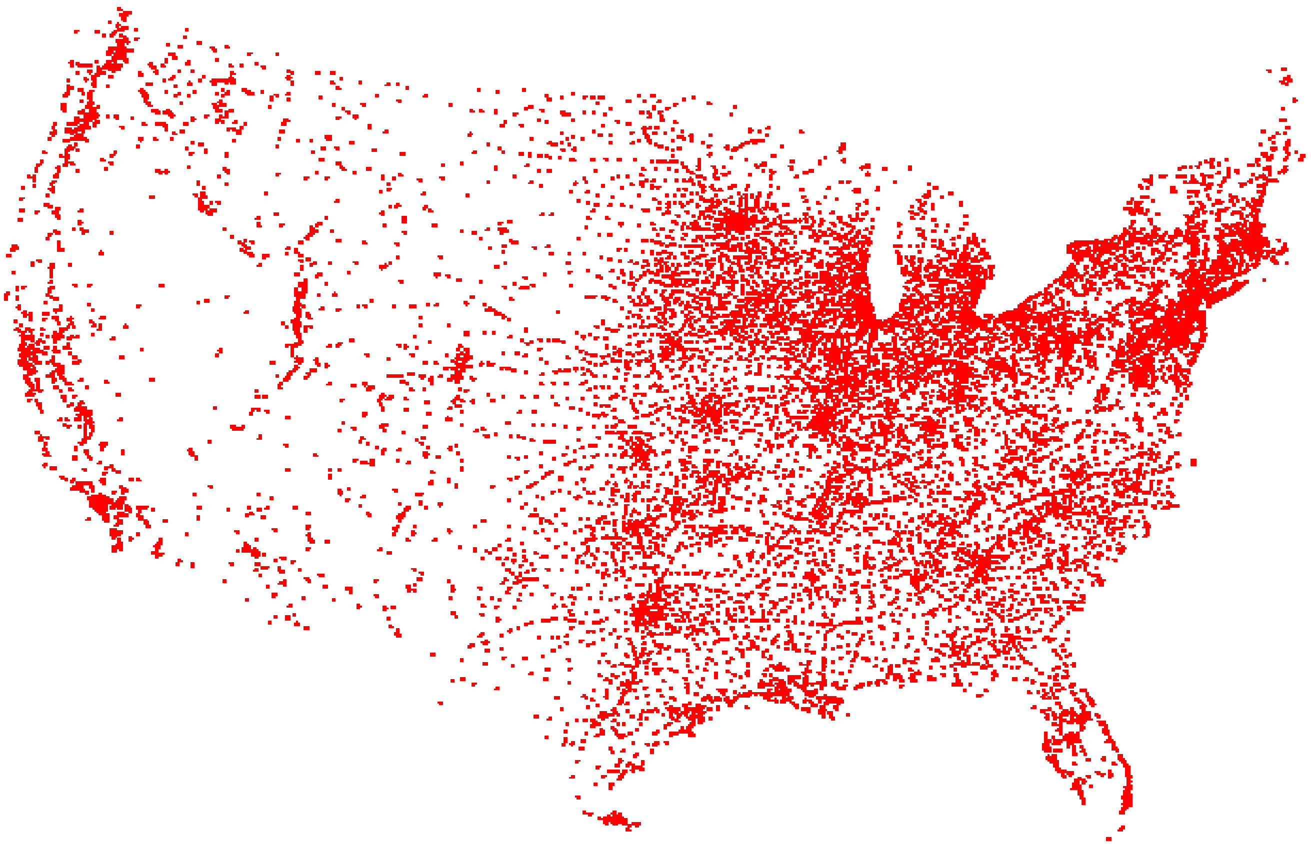
HAM-CYCLE: given an undirected graph G = (V, E), does there exist a simple cycle Γ that contains every node in V.



NO: bipartite graph with odd number of nodes.

Traveling Salesperson Problem

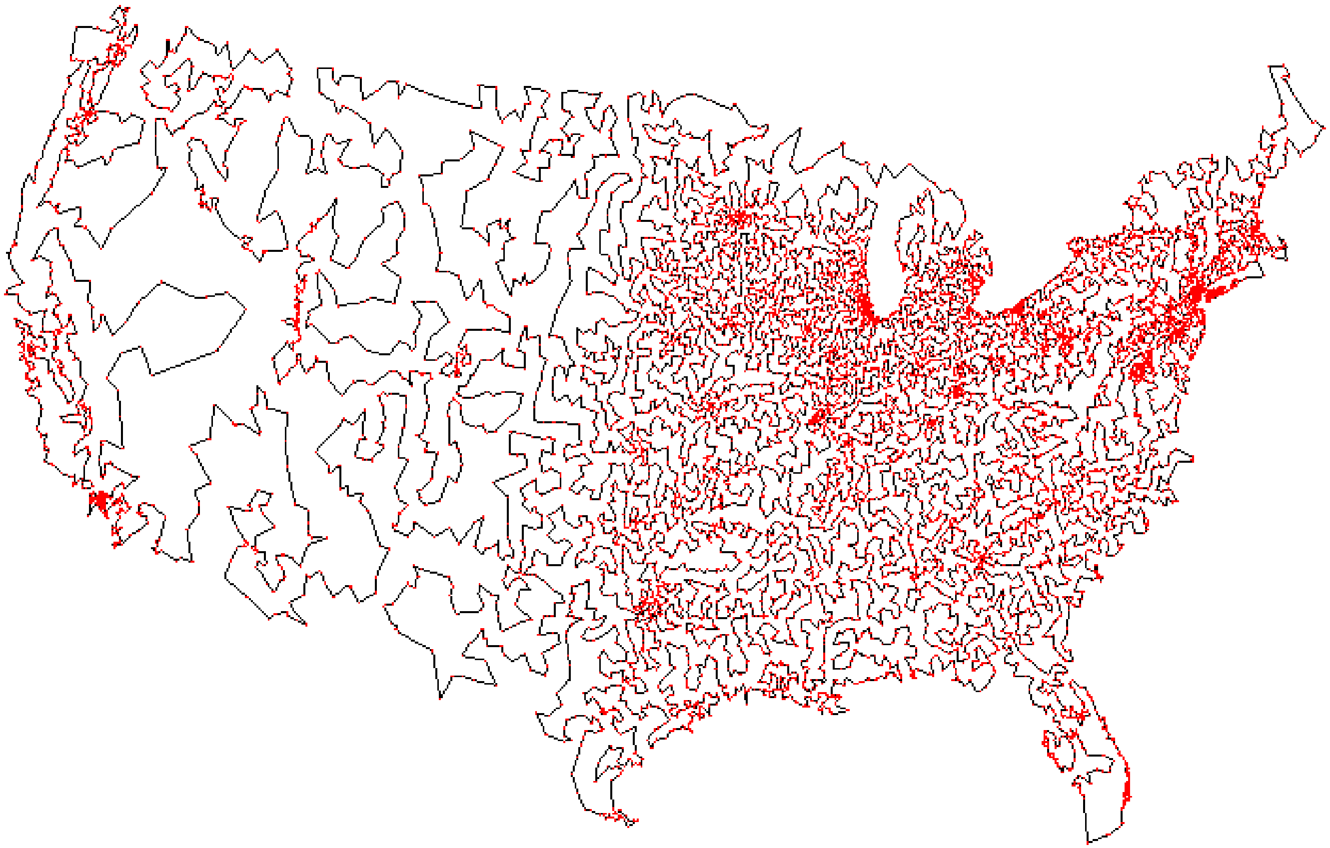
TSP. Given a set of n cities and a pairwise distance function d(u, v), is there a tour of length $\leq$ D?



All 13,509 cities in US with a population of at least 500
Reference: http://www.tsp.gatech.edu

Traveling Salesperson Problem

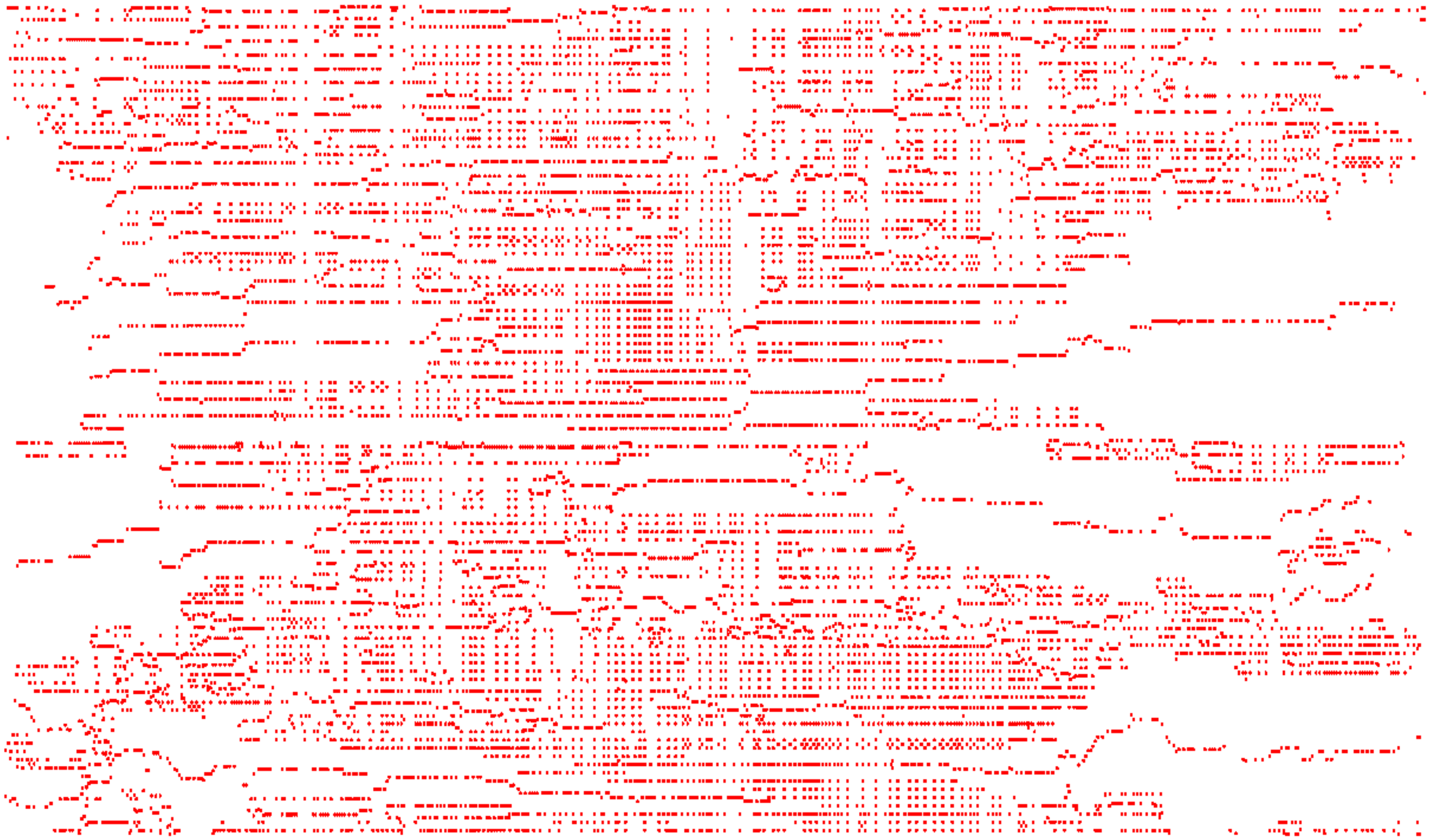
TSP. Given a set of n cities and a pairwise distance function d(u, v), is there a tour of length ≤ D?



Optimal TSP tour
Reference: http://www.tsp.gatech.edu

Traveling Salesperson Problem

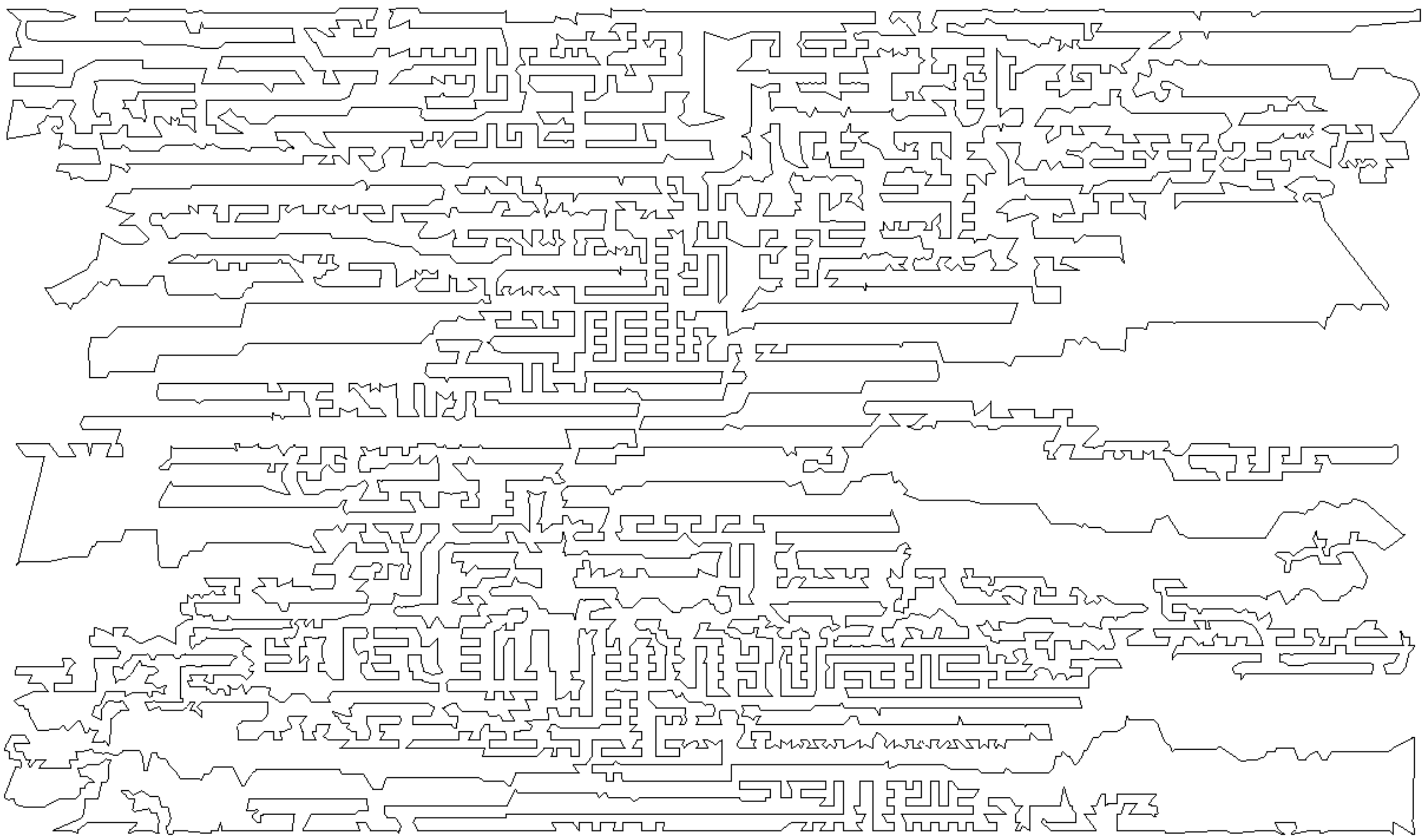
TSP. Given a set of n cities and a pairwise distance function d(u, v), is there a tour of length ≤ D?



11,849 holes to drill in a programmed logic array
Reference: http://www.tsp.gatech.edu

Traveling Salesperson Problem

TSP. Given a set of n cities and a pairwise distance function d(u, v), is there a tour of length $\leq$ D?



Optimal TSP tour
Reference: http://www.tsp.gatech.edu

Traveling Salesperson Problem

TSP. Given a set of n cities and a pairwise distance function d(u, v), is there a tour of length $\leq$ D?

HAM-CYCLE: given a graph G = (V, E), does there exists a simple cycle that contains every node in V?

Claim. HAM-CYCLE $\leq_P$ TSP.

Pf.

- Given instance G = (V, E) of HAM-CYCLE, create n cities with distance function

$$d(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E \\ 2 & \text{if } (u, v) \notin E \end{cases}$$

- TSP instance has tour of length $\leq$ n iff G is Hamiltonian. ▪

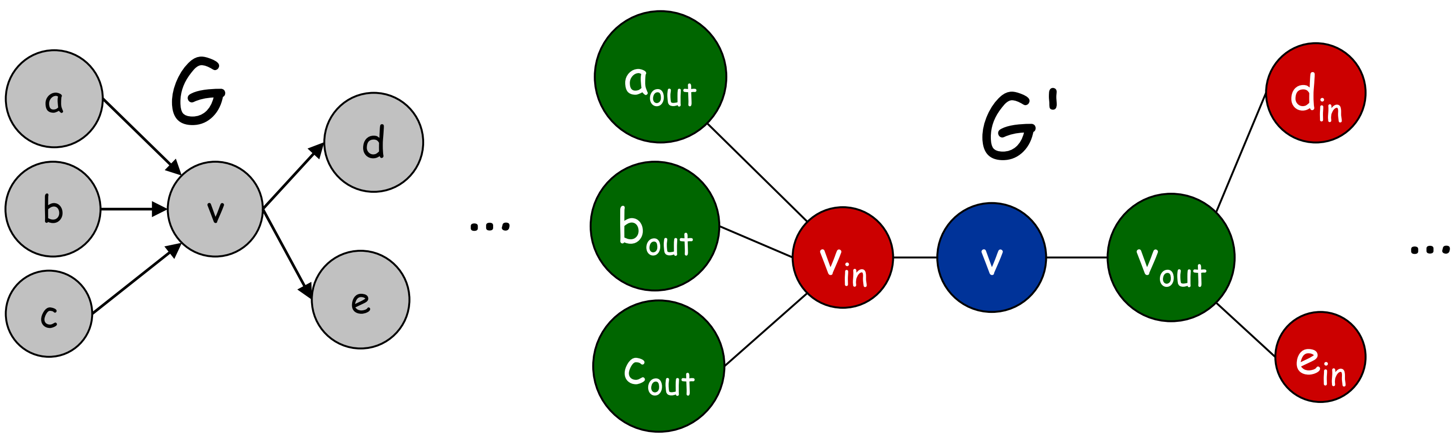
Remark. TSP instance in reduction satisfies Δ-inequality.

Directed Hamiltonian Cycle

DIR-HAM-CYCLE: given a digraph G = (V, E), does there exists a simple directed cycle Γ that contains every node in V?

Claim. DIR-HAM-CYCLE $\leq_P$ HAM-CYCLE.

Pf. Given a directed graph G = (V, E), construct an undirected graph G' with 3n nodes.

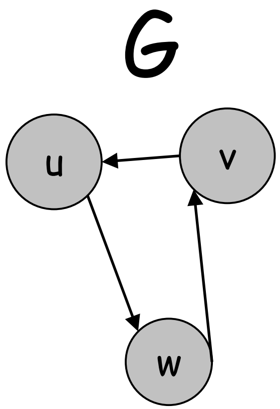


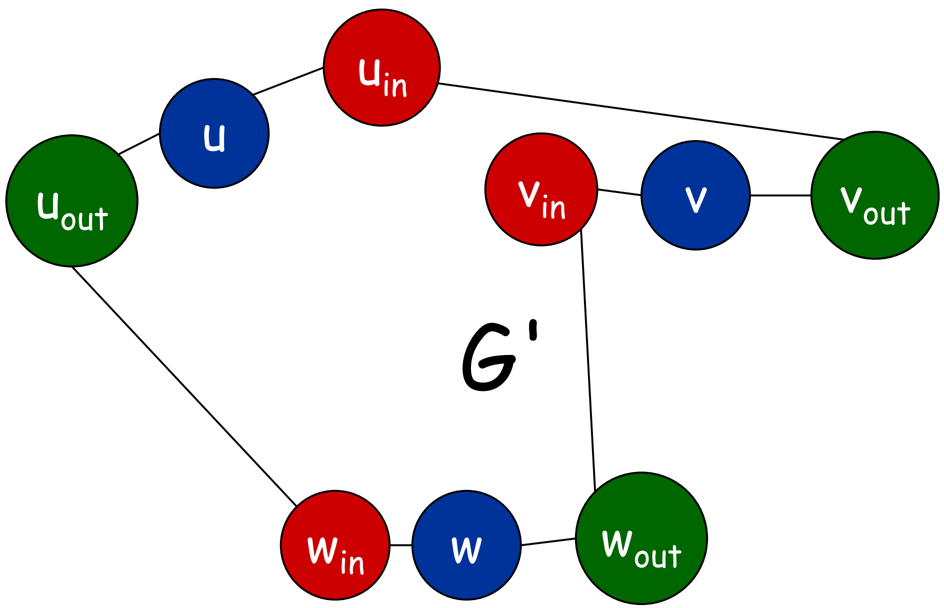
Directed Hamiltonian Cycle

Claim. G has a Hamiltonian cycle iff G' does.

Pf. ⇒

- Suppose G has a directed Hamiltonian cycle Γ (e.g., (u,w,v)).
- Then G' has an undirected Hamiltonian cycle (same order).
  - For each node v in directed path cycle replace v with v_{in},v,v_{out}





Directed Hamiltonian Cycle

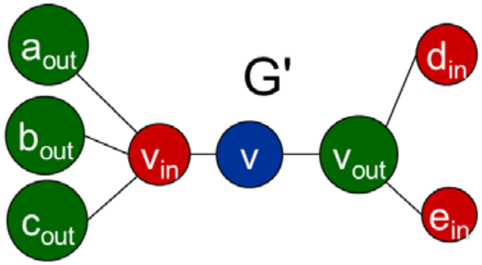
Claim. G has a Hamiltonian cycle iff G' does.

Pf. $\Rightarrow$

- Suppose G has a directed Hamiltonian cycle Γ.
- Then G' has an undirected Hamiltonian cycle (same order).
  - For each node v in directed path cycle replace v with v_{in}, v, v_{out}

Pf. $\Leftarrow$

- Suppose G' has an undirected Hamiltonian cycle Γ'.
- Γ' must visit nodes in G' using one of following two orders:
  …, B, G, R, B, G, R, B, G, R, B, …
  …, B, R, G, B, R, G, B, R, G, B, …
- Blue nodes in Γ' make up directed Hamiltonian cycle Γ in G, or reverse of one. ▪



More Hard Computational Problems

Aerospace engineering: optimal mesh partitioning for finite elements.
Biology: protein folding.
Chemical engineering: heat exchanger network synthesis.
Civil engineering: equilibrium of urban traffic flow.
Economics: computation of arbitrage in financial markets with friction.
Electrical engineering: VLSI layout.
Environmental engineering: optimal placement of contaminant sensors.
Financial engineering: find minimum risk portfolio of given return.
Game theory: find Nash equilibrium that maximizes social welfare.
Genomics: phylogeny reconstruction.
Mechanical engineering: structure of turbulence in sheared flows.
Medicine: reconstructing 3-D shape from biplane angiocardiogram.
Operations research: optimal resource allocation.
Physics: partition function of 3-D Ising model in statistical mechanics.
Politics: Shapley-Shubik voting power.
Pop culture: Minesweeper consistency.
Statistics: optimal experimental design.