Announcements

- Quiz due tonight
- HW 6 due 5/1 (Tuesday night!), and extra credit
- Midterms back on wednesday (Solutions up tonight)
- Last discussion on friday
- Final Exam: Friday 5/4, 3:30-5:30pm, Marcus Hall 131.

Recap

- Problem $X$ is a set of strings $s$, the YES instances.
- Algorithm $A$ solves $X$ if $A(s) = \text{true}$ iff $s \in X$.
- $B$ is polytime certifier for $X$ if
  - $B$ is polytime algorithm of two inputs $s$ and $t$ (a hint).
  - $s \in X$ iff exists $t$ with $|t| \leq p(|s|)$ and $B(s, t) = \text{True}$.
- $\mathcal{P}$ – class of problems with polytime algorithm.
- $\mathcal{NP}$ – class of problems with polytime certifier.
- $X$ is NP-Complete iff $Y \leq_{P} X$ for all $Y \in \mathcal{NP}$.

Example

<table>
<thead>
<tr>
<th>Problem ($X$)</th>
<th>INDEPENDENTSET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance ($s$)</td>
<td>Graph $G$ and number $k$</td>
</tr>
<tr>
<td>Algorithm ($A$)</td>
<td>Try all subsets and check (Runtime?)</td>
</tr>
<tr>
<td>Hint ($t$)</td>
<td>Which nodes are in the answer?</td>
</tr>
<tr>
<td>Certifier ($B$)</td>
<td>Are those nodes independent and size $k$?</td>
</tr>
</tbody>
</table>

Plan for today

- Review 3-SAT $\leq_{P}$ CIRCUITSAT
- HAMCYCLE
- TSP

Back to 3-SAT

Claim. If $Y$ is NP-complete and $Y \leq_{P} X$, then $X$ is NP-complete.

Theorem. 3-SAT is NP-Complete.

- Clearly in $\mathcal{NP}$.
- Prove by reduction from CIRCUITSAT.

Example.
The Reduction

- One variable \( x_v \) per circuit node \( v \).
- Clauses to enforce circuit computations.
  - If \( v = \neg \) then \( v \) has one input \( u \) and can add clauses \((x_v \lor x_u), (\neg x_v \lor \neg x_u)\).
  - If \( v = \lor \) with \( u, w \) incoming then \((x_v \lor \neg x_u), (x_v \lor \neg x_w), (\neg x_v \lor x_u \lor x_w)\).
  - If \( v = \land \) then \((\neg x_v \lor x_u), (\neg x_v \lor x_w), (x_v \lor \neg x_u \lor \neg x_w)\).
- Input bits get set with \((x_v)\) if fixed to one and \((\neg x_v)\) otherwise.
- Clause \((x_o)\) for output bit.

Final steps

- This formula satisfiable iff circuit is satisfiable.
- But not a 3-sat formula! It has clauses of size 1 and 2.
  - Fix: 4 new variables \( z_1, \ldots, z_4 \) where \( z_1, z_2 \) forced to be 0.
  - Include those two in any short clause.

Theorem. IndependentSet, VertexCover, SetCover, SAT, 3-SAT are all NP-Complete.

Finding NP-Complete Problems.

Want to prove problem \( X \) is NP-complete.

- Check \( X \in NP \).
- Choose known NP-complete problem \( Y \).
- Prove \( Y \leq_P X \).
- Often suffices to do single transformation from \( y \rightarrow x \) where
  - \( y \in Y \) if \( x \in X \).
  - \( y \notin Y \) if \( x \notin X \).
  - Known as Karp Reduction.

Touring problems.

Two new problems.

- TSP – Traveling Salesman. Given points \( v_1, \ldots, v_n \) with distances \( d(v_i, v_j) \geq 0 \), can we visit all points and return home with total distance less than \( B \)?
  \[
  \text{cost}(\sigma) = \sum_{i=1}^{n} d(v_{\sigma(i)}, v_{\sigma(i+1)})
  \]
- HamCycle – Hamiltonian Cycle. Given directed graph \( G = (V, E) \), is there a cycle that visits each vertex exactly once?

HamCycle Example

\[
\begin{array}{c}
\text{HamCycle Example} \\
\begin{array}{c}
\text{Example}
\end{array}
\end{array}
\]

HamCycle

Theorem. HamCycle is NP-Complete.

- It is in \( NP \).
- Need to reduce from some NP-Complete problem. Which one?

Claim. 3-SAT \( \leq_P \) HamCycle.

Reduction has two main parts.

- Make a graph with \( 2^n \) Hamiltonian cycles, one per assignment.
- Augment graph with clauses to invalidate assignments.
Graph skeleton

Skeleton Construction

- $n$ rows (one per variable).
- Row has $4m + 2$ vertices connected in forward and backward path.
- First and last vertex of row $i$ connected to first and last of $i+1$.
- Source $s$ connected to first and last of row 1.
- First and last of row $n$ connected to $t$.
- Edge $(t, s)$.

Augmenting

For clause $C_l = x_i \lor \neg x_j \lor x_k$ new node $c_l$ in graph.

- Edges $(v_i, 4l, c_l)$ and $(c_l, v_i, 4l+1)$.
- Edges $(v_j, 4l+1, c_l)$ and $(c_l, v_j, 4l)$.
- Edges $(v_k, 4l, c_l)$ and $(c_l, v_k, 4l+1)$.

Can only visit $c_l$ on row $i$ if traverse $i$ from left to right.

Example

(x_1 \lor x_2 \lor \neg x_3) \land (\neg x_1 \lor \neg x_2 \lor \neg x_3)

Proof

If $\phi$ is satisfying assignment

- If $\phi(x_i) = 1$ traverse left to right, else right to left.
- For each $C_l$, it is satisfied, so one term is traversed in the correct direction
  - We can therefore splice it into our cycle.

If $P$ is a Hamiltonian cycle

- If $P$ visits $c_l$ from row $i$, it will also leave to row $i$.
- Splice out clause variables leaves cycle on skeleton.
  - Cycles on skeleton correspond to assignments!

Traveling Salesman

- TSP – Traveling Salesman. Given points $v_1, \ldots, v_n$ with distances $d(v_i, v_j) \geq 0$, can we visit all points and return home with total distance less than $B$?

  \[
  \text{cost}(\sigma) = \sum_{i=1}^{n} d(v_{\sigma(i)}, v_{\sigma(i+1)})
  \]

Theorem. TSP is NP-Complete

- Clearly in $\mathcal{NP}$.
- Reduction from $\text{HAMCYCLE}$. 
**TSP reduction**

Given HamCycle instance $G = (V, E)$ make TSP instance

- One point per vertex.
- $d(v_i, v_j) = 1$ if $(v_i, v_j) \in E$, else 2. (asymmetric).
- Set bound to be $n$.

TSP of distance $n$ iff HamCycle of length $n$

---

**HamPath**

Similar to Hamiltonian Cycle, visit every vertex exactly once.

**Theorem.** HamPath is NP-Complete.

Two proofs.

- Modify 3-SAT to HamCycle reduction.
- Reduce from HamCycle directly.

---

**Graph Coloring**

**Def.** A $k$-coloring of a graph $G = (V, E)$ is a function $f : V \rightarrow \{1, \ldots, k\}$ such that for all $(u, v) \in E$, $f(u) \neq f(v)$.

**Problem.** Given $G = (V, E)$ and number $k$, does $G$ have a $k$-coloring?

Many applications

- Actually coloring maps!
- Scheduling jobs on machine with competing resources.
- Allocating variables to registers in a compiler.

---

**Graph Coloring**

**Claim.** 2-COLORING $\in \mathcal{P}$.

**Proof.**

- 2-coloring equivalent to bipartite testing.

**Theorem.** 3-COLORING is NP-Complete.

---

**Reduction**

Reduce from 3-SAT.

- Skeleton – Idea: 1 color for True, 1 for False
- 3 extra nodes in a clique $T, F, B$.
- For each variable $x_i$, two nodes $v_{i0}, v_{i1}$.
- Edges $(v_{i0}, B), (v_{i1}, B), (v_{i0}, v_{i1})$.
- Either $v_{i0}$ or $v_{i1}$ gets the $T$ color.

For clause $x_i \lor \neg x_j \lor x_k$
Proof

- Graph is polynomial in $n + m$.
- If satisfying assignment
  - Color $B, T, F$ then $v_{11}$ as $T$ if $\phi(x_i) = 1$.
  - Since clauses satisfied, can color each gadget.
- If graph 3-colorable
  - One of $v_{01}, v_{11}$ must get $T$ color.
  - Clause gadget colorable iff clause satisfied.

**Question.** What about $k$-coloring?