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 $\Gamma$ that contains every node in $V$.

**YES**: vertices and faces of a dodecahedron.
Hamiltonian Cycle

**HAM-CYCLE**:

Given an undirected graph $G = (V, E)$, does there exist a simple cycle $\Gamma$ that contains every node in $V$?

*NO*: bipartite graph with odd number of nodes.

**Directed Hamiltonian Cycle**

**DIR-HAM-CYCLE**:

Given a digraph $G = (V, E)$, does there exist a simple directed cycle $\Gamma$ that contains every node in $V$?

**Claim**. $\text{DIR-HAM-CYCLE} \leq P_{\text{HAM-CYCLE}}$.

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

---

**3-SAT Reduces to Directed Hamiltonian Cycle**

**Claim**. $\text{3-SAT} \leq P_{\text{DIR-HAM-CYCLE}}$.

**Pf**. Given an instance $\Phi$ of 3-SAT, we construct an instance of $\text{DIR-HAM-CYCLE}$ that has a Hamiltonian cycle iff $\Phi$ is satisfiable.

**Construction**. First, create graph that has $2^n$ Hamiltonian cycles which correspond in a natural way to $2^n$ possible truth assignments.

---

**Longest Path**

**SHORTEST-PATH**. Given a digraph $G = (V, E)$, does there exist a simple path of length at most $k$ edges?

**LONGEST-PATH**. Given a digraph $G = (V, E)$, does there exist a simple path of length at least $k$ edges?

**Claim**. $\text{3-SAT} \leq P_{\text{LONGEST-PATH}}$.

**Pf 1**. Redo proof for $\text{DIR-HAM-CYCLE}$, ignoring back-edge from $t$ to $s$.

**Pf 2**. Show $\text{HAM-CYCLE} \leq P_{\text{LONGEST-PATH}}$. 

---
The Longest Path

Lyrics. Copyright © 1988 by Daniel J. Barrett.
Music. Sung to the tune of The Longest Time by Billy Joel.

Woh-oh-oh-oh, find the longest path!
Woh-oh-oh-oh, find the longest path!
If you said P is NP tonight,
There would still be papers left to write,
I have a weakness,
I'm addicted to completeness,
And I keep searching for the longest path.
The algorithm I would like to see
Is of polynomial degree,
But it's elusive:
Nobody has found conclusive
Evidence that we can find a longest path.
I have been hard working for so long,
I swear it's right, and he marks it wrong.
Some how I'll feel sorry when it's done:
GPA 2.1
Is more than I hope for.
Garey, Johnson, Karp and other men (and women)
Tried to make it order N log N.
Are I a mad fool
If I spend my life in grad school,
Forever following the longest path?
Woh-oh-oh-oh, find the longest path!
Woh-oh-oh-oh, find the longest path!
Woh-oh-oh-oh, find the longest path.

† Recorded by Dan Barrett while a grad student at Johns Hopkins during a difficult algorithms final.

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?

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

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

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

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

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

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 $\leq D$?

**Optimal TSP tour Reference:** [http://www.tsp.gatech.edu](http://www.tsp.gatech.edu)

---

**8.6 Partitioning 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.

---

**3-Dimensional Matching**

**3D-MATCHING.** Given $n$ instructors, $n$ courses, and $n$ times, and a list of the possible courses and times each instructor is willing to teach, is it possible to make an assignment so that all courses are taught at different times?

<table>
<thead>
<tr>
<th>Instructor</th>
<th>Course</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wayne</td>
<td>COS 423</td>
<td>MW 11-12.20</td>
</tr>
<tr>
<td>Wayne</td>
<td>COS 423</td>
<td>TTh 11-12.20</td>
</tr>
<tr>
<td>Wayne</td>
<td>COS 226</td>
<td>TTh 11-12.20</td>
</tr>
<tr>
<td>Wayne</td>
<td>COS 126</td>
<td>TTh 11-12.20</td>
</tr>
<tr>
<td>Tardos</td>
<td>COS 523</td>
<td>TTh 3-4.20</td>
</tr>
<tr>
<td>Tardos</td>
<td>COS 423</td>
<td>TTh 3-4.20</td>
</tr>
<tr>
<td>Tardos</td>
<td>COS 423</td>
<td>MW 11-12.20</td>
</tr>
<tr>
<td>Kleinberg</td>
<td>COS 226</td>
<td>TTh 3-4.20</td>
</tr>
<tr>
<td>Kleinberg</td>
<td>COS 423</td>
<td>MW 11-12.20</td>
</tr>
</tbody>
</table>

---

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

**Claim.** $\text{HAM-CYCLE} \leq_p \text{TSP}$.

**Pf.**
- Given instance $G = (V, E)$ of $\text{HAM-CYCLE}$, 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 $\Delta$-inequality.
3-Dimensional Matching

30-DIMENSIONAL MATCHING. Given disjoint sets \( X, Y, \) and \( Z, \) each of size \( n, \) and a set 
\( T \subseteq X \times Y \times Z \) of triples, does there exist a set of \( n \) triples in \( T \) such 
that each element of \( X \cup Y \cup Z \) is in exactly one of these triples?

Claim. 3-SAT \( \leq \) P INDEPENDENT-COVER.

Pf. Given an instance \( \Phi \) of 3-SAT, we construct an instance of 3D-
matching that has a perfect matching iff \( \Phi \) is satisfiable.

8.7 Graph Coloring

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.

3-Colorability

3-COLOR: Given an undirected graph \( G, \) does there exists a way to
color the nodes red, green, and blue so that no adjacent nodes have the
same color?

Register Allocation

Register allocation. Assign program variables to machine register so
that no more than \( k \) registers are used and no two program variables
that are needed at the same time are assigned to the same register.

Interference graph. Nodes are program variables names, edge
between \( u \) and \( v \) if there exists an operation where both \( u \) and
\( v \) are “live” at the same time.

Observation. [Chaitin 1982] Can solve register allocation problem iff
interference graph is \( k \)-colorable.

Fact. 3-COLOR \( \leq \) k-REGISTER-ALLOCATION for any constant \( k \neq 3.\)
8.8 Numerical 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: 3-COLOR, 3D-MATCHING.
- Numerical problems: SUBSET-SUM, KNAPSACK.

8.10 A Partial Taxonomy of Hard Problems

Subset Sum

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

Ex: $(1, 4, 6, 4, 256, 1040, 1041, 1093, 1284, 1344)$, $W = 3754$.

Yes: $1 + 16 + 256 + 1040 + 1093 + 1284 = 3754$.

Remark. With arithmetic problems, input integers are encoded in binary. Polynomial reduction must be polynomial in binary encoding.

Claim. 3-SAT $\leq_P$ SUBSET-SUM.

Pf. Given an instance $\Phi$ of 3-SAT, we construct an instance of SUBSET-SUM that has solution iff $\Phi$ is satisfiable.

Polynomial-Time Reductions
Planarity Testing

Planarity testing. [Hopcroft-Tarjan 1974] \(O(n)\).

Remark. Many intractable graph problems can be solved in poly-time if the graph is planar; many tractable graph problems can be solved faster if the graph is planar.

Planar 3-Colorability

Claim. \(3\)-COLOR \(\leq_p\) PLANAR-3-COLOR.
Proof sketch: Given instance of 3-COLOR, draw graph in plane, letting edges cross if necessary.
- Replace each edge crossing with the following planar gadget \(W\).
  - in any 3-coloring of \(W\), opposite corners have the same color
  - any assignment of colors to the corners in which opposite corners have the same color extends to a 3-coloring of \(W\)

Planar k-Colorability

PLANAR-2-COLOR. Solvable in linear time.

PLANAR-3-COLOR. NP-complete.

PLANAR-4-COLOR. Solvable in \(O(1)\) time.

Theorem. [Appel-Haken, 1976] Every planar map is 4-colorable.
- Resolved century-old open problem.
- Used 50 days of computer time to deal with many special cases.
- First major theorem to be proved using computer.

False intuition. If PLANAR-3-COLOR is hard, then so is PLANAR-4-COLOR and PLANAR-5-COLOR.