EECS 336: Lecture 10: Introduction to Algorithms

P vs. NP: indep set, hamiltonian cycle, 3d matching

Reading: 8.4, 8.5, 8.6.

`"guide to reductions"`

Last Time:

- reductions (cont)
- tractability and intractability
- 3-SAT $\leq_p$ INDEP-SET

Today:

- 3-SAT $\leq_p$ INDEP-SET
- 3-SAT $\leq_p$ HAMILTONIAN-CYCLE
- 3-SAT $\leq_p$ 3D-MATCHING

Reduction Illustrated

Problems	3-SAT	INDEP-SET
Instance	f	(V^f, E^f, θ^f)
Solution	$\mathbf{z}$	S^f

Problem Y: 3-SAT

input: boolean formula $f(\mathbf{z}) = \bigwedge_{j=1}^{m}(l_{j1} \vee l_{j2} \vee l_{j3})$

- literal l_{jk} is variable "z_i" or negation "$\bar{z}_i$"
- "and of ors"
- e.g., $f(\mathbf{z}) = (z_1 \vee \bar{z}_2 \vee z_3) \wedge (z_2 \vee \bar{z}_5 \vee z_6) \wedge ...$

output:

- "Yes" if assignment $\mathbf{z}$ with $f(\mathbf{z}) = T$ exists

  e.g., $\mathbf{z} = (T, T, F, T, F, ...)$
- "No" otherwise.

Problem X: INDEP-SET

input: $G = (V, E), k$

output: "yes" if $\exists S \subset V$

- satisfying $\forall v \in S, (u, v) \notin E$
- $|S| \geq \theta$

Independent Set Reduction

Lemma: 3-SAT $\leq_p$ INDEP-SET

Part 1: forward instance construction

convert 3-SAT instance f into INDEP-SET instance (V^f, E^f, θ^f).

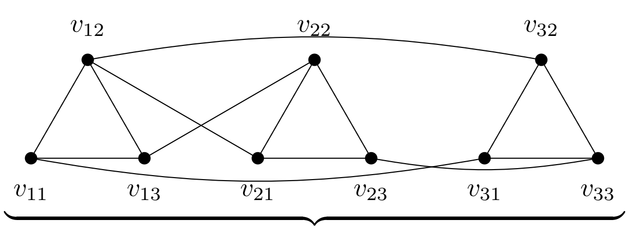
- goal: "at least one true literal per clause" $\Leftrightarrow$ "independent set of size at least θ"
- literal l_{ij} $\Rightarrow$ vertices $v_{ij} \in V^f$
- "all clauses true" $\Rightarrow$ $\theta^f = m$
- "literal conflicts" $\Rightarrow$ conflict edges.

  $\forall i$: $l_{jk} =$ "z_i" and $l_{j'k'} =$ "$\bar{z}_i$" $\Rightarrow (v_{jk}, v_{j'k'}) \in E^f$
- "one representative per clause" $\Rightarrow$ clause edges.

  $\forall j$: $(v_{j1}, v_{j2}),\ (v_{j2}, v_{j3}),\ (v_{j3}, v_{j1}) \in E^f$

Example:

$f(\mathbf{z}) = (z_1 \vee z_2 \vee z_3) \wedge (\bar{z}_2 \vee \bar{z}_3 \vee \bar{z}_4) \wedge (\bar{z}_1 \vee \bar{z}_2 \vee z_4)$



Runtime Analysis: linear time (one vertex per literal.)

Part II: reverse certificate construction

construct assignment $\mathbf{z}$ from S^f

(if (V^f, E^f) has indep. set S^f size $\geq \theta^f = m$ then f is satisfiable.)

For each z_i:

- if exists vertex in S labeled by "z_i"

  set $z_i = T$
- else

  set $z_i = F$

Claim: if vertex in S is labeled by "$\bar{z}_i$" then no vertices in S are labeled by "z_i" and z_i is set to False.

(because of conflict edge between vertex labeled "$\bar{z}_i$" and all vertices labeled "z_i".)

Claim: S independent and $|S| \geq m \Rightarrow f(\mathbf{z}) = T$:

- S has $|S| = m$

  $\Rightarrow S$ has one vertex per clause.
- for clause j and $v_{jk} in S$:

  if l_{jk} is "z_i", then z_i is true (by construction)

  if l_{jk} is "$\bar{z}_i$", then z_i is false (by claim)
- So $f(\mathbf{z}) = T$.

Part III: forward certificate construction

construct independent set S from z

(if f is satisfiable then (V^f, E^f) has indep. set size $\geq m = \theta^f$.)

- let S' be nodes in (V^f, E^f) corresponding to true literals.
- if more than one vertex in S' in same triangle drop all but one.

  $\Rightarrow S$.
- $|S| = m$
- for all $u, v \in S$,
  - $u \& v$ not in same triangle.
  - l_u and l_v both true

    $\Rightarrow$ must not conflict

    $\Rightarrow$ no (l_u, l_v) edge in (V^f, E^f).
  - so S is independent.

Reductions From 3-SAT

Must Encode:

a) "at least one true literal per clause"

b) "true literals for each z_i either all"z_i" or all "$\bar{z}_i$"

Problem: Hamiltonian Cycle

input: directed graph (V, E)

output: "yes" if exists cycle C that visits each vertex exactly once.

Lemma: hamiltonian cycle is NP-hard

Proof: (reduction from 3-SAT)

Part I: construction

(turn 3-SAT formula f in to graph (V^f, E^f) with hamiltonian cycle iff f is satisfiable)

- idea: variable = isolated path, right-to- left = true, left-to-right = false.
- idea: clause is node, which needs to be hit by at most one literal being true.
- construction:
- left-right path per variable.
- splice in clause nodes.

Runtime: $O(nm)$

Part II: reverse certificate construction

- high-level: ensure "other paths" do not exist.

Part III: forward certificate construction

- high-level: confirm "desired path" exists.

Problem: Traveling Salesman (TSP)

Lemma: TSP is $\mathcal{NP}$-hard.

Proof: reduction from Hamiltonian Cycle

Part I: forward instance construction

- encode edges with cost 1
- encode non-edges with cost n.

Part II & III: exists HC iff TSP cost $\leq n$

Problem: 3D-MATCHING

Input: tripartite hypergraph (A, B, C, E) * vertices A, B, C, * edges $E \subset A \times B \times C$

Output: "yes" if exist prefect matching $M \subset E$

3D Matching

Lemma: 3-SAT $\leq_p$ 3D-MATCHING

Part I: forward instance construction

(convert 3-SAT instance f into 3D-MATCHING instance (A^f, B^f, C^f, E^f))

variable gadget i:

- vertices $a_{i1}, \ldots, a_{im}$, $b_{i1}, \ldots, b_{im}$, $c_{i1T}, \ldots, c_{imT}$, $c_{i1F}, \ldots, c_{imF}$
- true edges $\{(a_{ij}, b_{ij}, c_{ijT}) : j \in [m]\}$
- false edges $\{(a_{ij}, b_{ij}, c_{ijF}) : j \in [m]\}$
- m true tips, m false tips.

clause gadget j:

- two vertices a_j, b_j

literal edge l_{jk}:

- "z_i" $\Rightarrow (a_j, b_j, c_{ijT})$
- "$\bar{z}_i$" $\Rightarrow (a_j, b_j, c_{ijF})$

cleanup gadgets $r \in \{1, \ldots, 2mn - m\}$:

- two vertices a'_r, b'_r
- edges $\{(a'_r, b'_r, c_{ijB}) : i \in [n], j \in [m], B \in \{T, F\}\}$

Parts II & III: see book.