Polytime reductions, SAT

Tuesday, April 21, 2020 3:21 PM

Polynomial Reductions
- "Problem X is no harder than problem Y"
  Understanding des'ns and logical consequences
- Problem solving
- More common problems

- Recap reductions
- add poly time.

Problem X reduces to problem Y

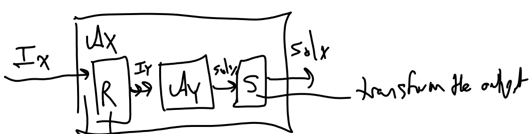
set of problem instances

SSSP is a problem.

An instance for SSSP would be (G, s)

A solution to problem X is an alg. A_X that gives a solution for every instance I_X — an instance of problem X.

A solution for Y, $A_Y(I_Y) \to sol_Y$ implies a solution for X, $A_X(I_X) \to sol_X$



R transforms an instance I_X into an instance of Y

NFAs to DFAs

eg. "does NFA n accept string x?" ... or all strings?

"does DFA d accept string x?"

let $n = (\Sigma, Q, S, A)$ then

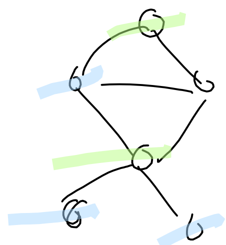
Subset construction define $d = (\Sigma, P(Q) \ldots$

Decision Problem
- for Yes/no Qs, Don't transform output,

I_Y is YES iff I_X is Yes

I_Y is IES ...

Maximal independent set



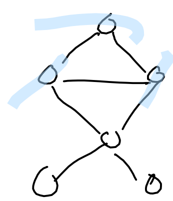
G =

- an IS a set of vertices none of which are adjacent
- maximal: no other vertex you can add and it's still independent
- largest maximal IS. MIS of largest size.

Decision Problem: Is there an MIS of size k?

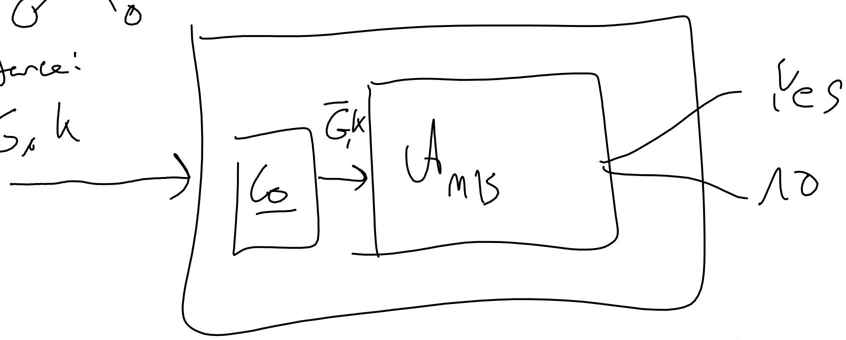
Clique:

Clique is a set of nodes that are all connected.
Decision: Is there a clique of size k?

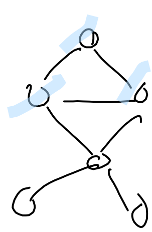


Claim: Clique reduces to MIS.

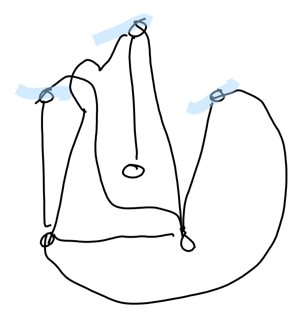
Clique instance: G, k



G =



G^c, $\bar{G}$ =



C is a clique in G,

$\forall u,v \in C \quad u-v \in E \implies \forall u,v \in C \quad u \not- v \in \bar{E}$

Clique reduces to MIS

Clique $\leq$ MIS

Clique is ... harder than MIS.

Clique is no harder ...

MIS is at least as hard as Clique

Polynomial time.

running time is $O(\text{poly}(n))$ where n is t

Complexity theory domain

polynomial vs. exponential

$O(n^c)$ for some c.

n^3 vs n^4 still poly time.

$n \log n$ vs n still poly time

2^n exponential

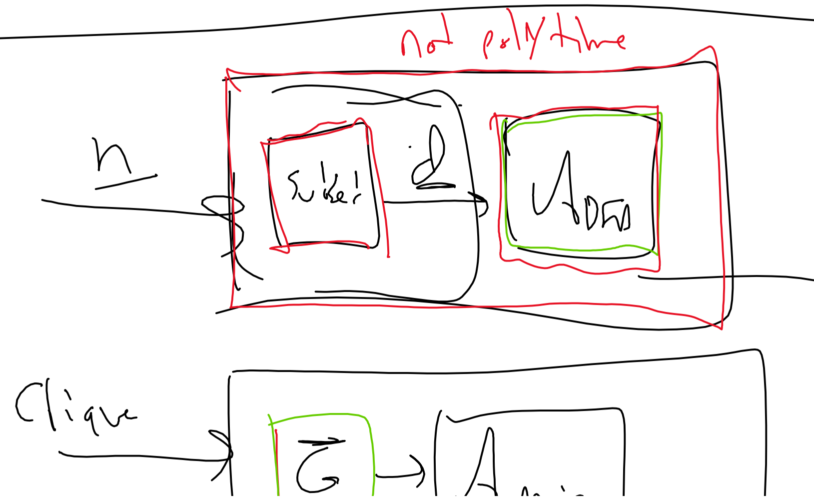
narrower theory asymptotic improvements

n^3, n^2

$n \log n$ vs n quasilinear vs.

engineering applied

Constant factors



$|d| = 2$

not poly time transform exponen

How large is $\overline{G}$ as a function of m, $G = (V, E$

$n = m$, $|\overline{E}| = n^2$

up to quadratic overhead in transforming.

poly(n)

$$X \leq_p Y$$

X polytime reduces to Y.

Direction matters

Suppose: X and Y are decision problems.

X is solvable in poly time

and $X \leq_p Y$.

Which are true?

— Y solvable in poly time? X

— Y cannot be solved in poly time? Dunno. X

A solution for Y, $A_Y(I_Y) \to sol_Y$
implies a solution for X, $A_X(I_X) \to sol_X$

Decision problems: In order to show $X \leq Y$

problem transformation is one way:

$I_X \Longrightarrow I_Y$ Do not have to show $I_Y \Rightarrow$

Do have to show I_Y is YES $\Longleftrightarrow I_X$

Boolean Satisfiability: SAT

Given a CNF formula, is there a satisfying assignment

CNF formulas:

- $X_1 \ldots X_n$ boolean vars.
- $X_i, \neg X_i$ literals (var, or negation of var)
- $X_i \vee \neg X_j \ldots$ clause ("OR" of literals)
- $(X_i \vee \neg X_j) \wedge \ldots$ ("And" of clauses)

"And of ORs"

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

$\wedge \left| \begin{array}{ll} (X_1 \vee \ldots X_i \vee \ldots) & \text{clause } 1 \\ (\quad \vee \ldots \vee \quad) & \text{clause } 2 \\ \vdots & \end{array} \right.$

Satisfying assignment is: a boolean value for each var, s.t. all clauses hold

ex $(X_1 \vee X_2 \vee \underbrace{\neg X_4}_{\text{literal}}) \wedge \underbrace{(X_2 \vee \neg X_3)}_{\text{clause}}$

$x_1 = 1, x_2 = 1, x_3 = 1, x_4 = 1$ ✓

ex: unsatisfiable $(X_1) \land (\neg X_1)$ ✗

3SAT. Every clause has exactly 3 literals. w/ distinct vars.

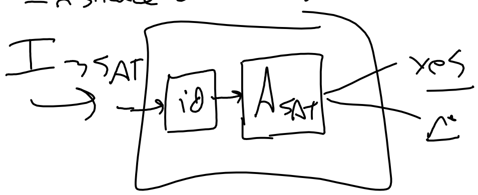
$(X_1) \land (\neg X_1)$ not in 3SAT.

$$3SAT \leq SAT \ ? \ ✓$$

$$SAT \leq 3SAT \ ?$$

Which one is trivial?

Instance of 3SAT is also an instance of SAT.



Turning truth tables into SAT instances.

$$Y = X \land (\neg Z \lor Y)$$

X	Y	Z	$X \land (\neg Z \lor Y)$	Y= ...
0	0	0	0	1
0	0	1	0	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	0
1	0	1	0	1
1	1	0	1	1
1	1	1	1	1

"not this row" $(X \lor \neg Y \lor Z) \land$

"not this row" $(X \lor \neg Y \lor \neg Z) \land$

"not this row" $(\neg X \lor Y \lor Z)$

Converting SAT instance to 3SAT

$(X \lor \neg Y) \land (\ldots$

— padding small clause (...) ...

— breaking up large clauses $(X \vee \neg Y \vee Z \vee A) \wedge \ldots$

$(X \vee Y)$ is satisfiable iff

$$(X \vee Y \vee u) \wedge (X \vee Y \vee \neg u)$$

where u is a fresh variable.

Claim: Let P and Q be clauses

$P \vee Q$ is satisfiable iff

$(P \vee Z) \wedge (Q \vee \neg Z)$ is satisfiable. for a fresh variable Z.

$Z = 1 \Longrightarrow Q$ must hold, $(P \vee Z)$ holds regardless of P

$Z = 0 \Longrightarrow \ldots P$ must hold.

$$\Longrightarrow (\underbrace{X_1 \vee \ldots}_{P} \underbrace{X_2 \vee \ldots \vee X_k}_{Q})$$

P: 2 literals

Q: rest of the literals.

So ...

SAT $\leq$ 3SAT