- CUBIC u_1 = True in 3SAT $\rightarrow$ $u_1 = 1$, $\bar{u}_1 = 0$ in CUBIC

  u_1 = False in 3SAT $\rightarrow$ $u_1 = 0$, $\bar{u}_1 = 1$

  $u_1 \neq \bar{u}_1 \Rightarrow u_1 u_1 u_1 + \bar{u}_1 \bar{u}_1 \bar{u}_1 = 1$

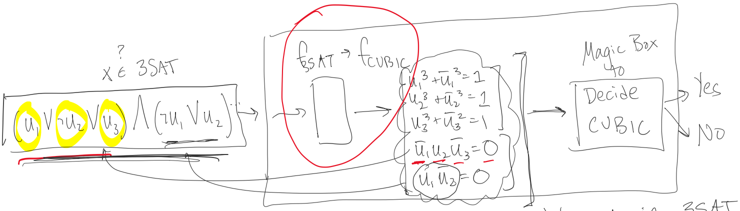
  TRUE FALSE TRUE

  $u_1 \vee \neg u_2 \vee u_3 \rightarrow \bar{u}_1 \cdot u_2 \bar{u}_3 = 0$

WICS, Hacker Thing

Goals ← Nondeterministic Polynomial Time

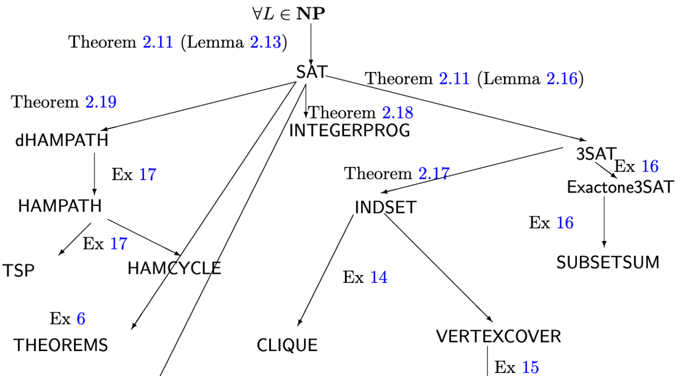
- Place NP in our map
- Prove Nondeterministic Connection


$x \overset{?}{\in}$ 3SAT
$(u_1 \vee \neg u_2 \vee u_3) \wedge (\neg u_1 \vee u_2)$
$f_{3SAT} \rightarrow f_{CUBIC}$
$u_1^3 + \bar{u}_1^3 = 1$
$u_2^3 + \bar{u}_2^3 = 1$
$u_3^3 + \bar{u}_3^2 = 1$
$\bar{u}_1 u_2 \bar{u}_3 = 0$
$u_1 \bar{u}_2 = 0$
Magic Box to
Decide CUBIC
Yes
No

Formal Proof of NP-Hard:

- Describe f →
  - For each variable u_i in 3SAT instance, create an equation $u_i^3 + \bar{u}_i^3 = 1$
  - For each clause in 3SAT create an equation $V_i V_j V_k = 0$

    $V_i = u_i$ or $\bar{u}_i$ base on if $\neg u_i$ or u_i
- $x \in$ 3SAT iff $f(x) \in$ 01PROG
  - $\Rightarrow$ If satisfying assignment for x, then satisfying for f(x)
  - If no satisfying " " no
- f takes polytime. "


$\forall L \in \mathbf{NP}$
Theorem 2.11 (Lemma 2.13)
SAT
Theorem 2.11 (Lemma 2.16)
Theorem 2.19
Theorem 2.18
INTEGERPROG
dHAMPATH
3SAT
Ex 16
Exactone3SAT
Ex 17
Theorem 2.17
HAMPATH
INDSET
Ex 16
Ex 17
TSP
HAMCYCLE
Ex 14
SUBSETSUM
Ex 6
THEOREMS
CLIQUE
VERTEXCOVER
Ex 15

Ex 18
QUADEQ
MAXCUT

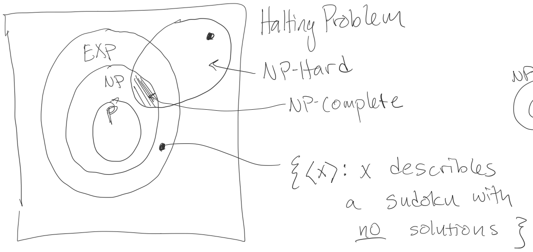
Figure 2.4: Web of reductions between the **NP**-completeness problems described in this chapter and the exercises. Thousands more are known.

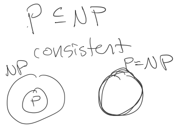
01PROG, 3SAT, CUBIC ∈ NP

def: L is NP-complete if

- L ∈ NP
- L ∈ NP-Hard

Big Picture: Draw NP-Hard, NP-complete


EXP
NP
P
Halting Problem
NP-Hard
NP-complete
{⟨x⟩: x describes a sudoku with no solutions}


P ⊆ NP
consistent
NP
P
P=NP

Goals

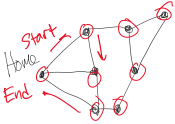
- Create + alleviate NP-Despair

Read + Despair:

- NP-Complete problems are believed to take exponential time to solve
  99.9% of computer scientists
- NP-complete problems are extremely useful + important

Don't fire all your computer scientists yet!

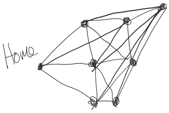
ex: Traveling Salesperson


Start
Home
End

Hit all vertices with less than K steps.

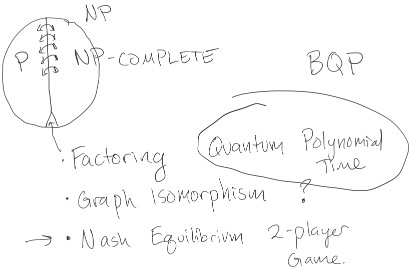
NP-complete

Traveling Dronebot



Is there a path that hits all vertices with length within 10% of optimal.

P

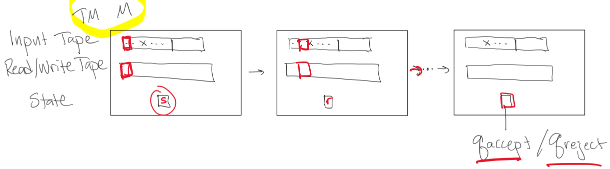

NP
P
NP-COMPLETE
BQP
Quantum Polynomial Time
?
· Factoring
· Graph Isomorphism
→ · Nash Equilibrium 2-player Game.

Goal

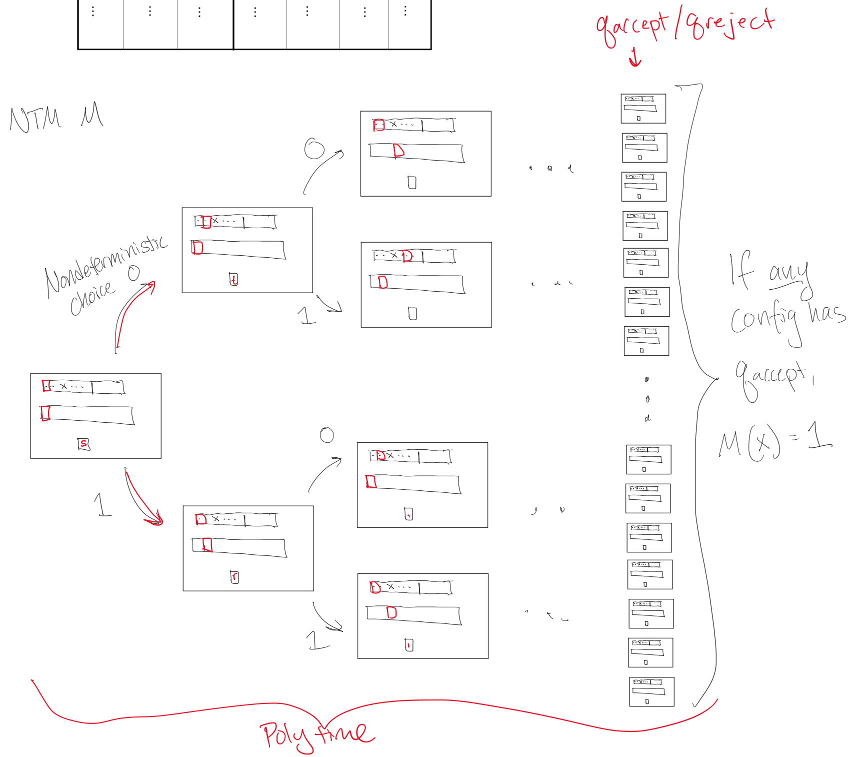
Describe NP using NTMs.

def: $L \in NP$ if $\exists$ a NTM M s.t. $x \in L$ iff $M(x) = 1$ and M halts in polynomial ← time regardless of its non-deterministic choices.

NP = Nondeterministic Polynomial Time


TM M
Input Tape
Read/Write Tape
State
s
qaccept / qreject

IF			THEN			
input symbol read	work/output tape symbol read	current state	move input head	new work/output tape symbol	move work/output tape	new state
⋮	⋮	⋮	⋮	⋮	⋮	⋮
a	b	q	→	b'	←	q'
⋮	⋮	⋮	⋮	⋮	⋮	⋮


NTM M
Nondeterministic choice 0
1
0
1
0
1
s
qaccept/qreject
If any config has qaccept,
M(x) = 1
Poly time