BU CS 332 – Theory of Computation

https://forms.gle/4op1rh6EDQPPKaJg6

Lecture 23:

- More NP-completeness

Reading:

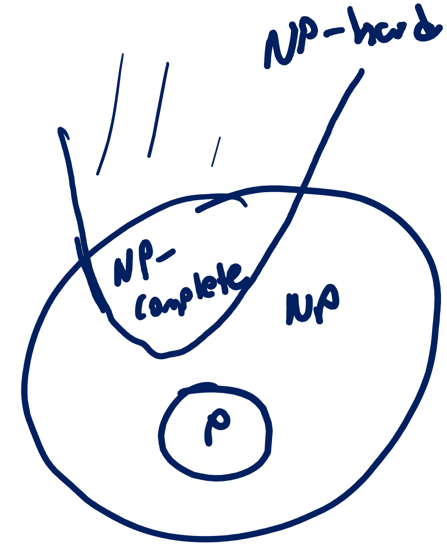
Sipser Ch 7.4-7.5

Final: Thursday 5/9
3-5
Room TBD

Mark Bun

April 29, 2024

NP-completeness



"The hardest languages in NP"

Definition: A language B is NP-complete if

1) $B \in \mathrm{NP}$, and
2) B is NP-hard: Every language $A \in \mathrm{NP}$ is poly-time reducible to B, i.e., $A \leq_{\mathrm{p}} B$

Last time:

$TMSAT = \{\langle N, w, 1^t \rangle \mid$ NTM N accepts input w within t steps$\}$ is NP-complete

<u>Cook-Levin Theorem:</u>
$\{\langle \varphi \rangle \mid$ Boolean formula φ is satisfiable$\}$ is NP-complete

New NP-complete problems from old

Lemma: If $A \leq_{\mathrm{p}} B$ and $B \leq_{\mathrm{p}} C$, then $A \leq_{\mathrm{p}} C$

(poly-time reducibility is transitive)

Theorem: If $B \leq_{\mathrm{p}} C$ for some NP-hard language B, then C is also NP-hard

The usual way to prove NP-completeness:

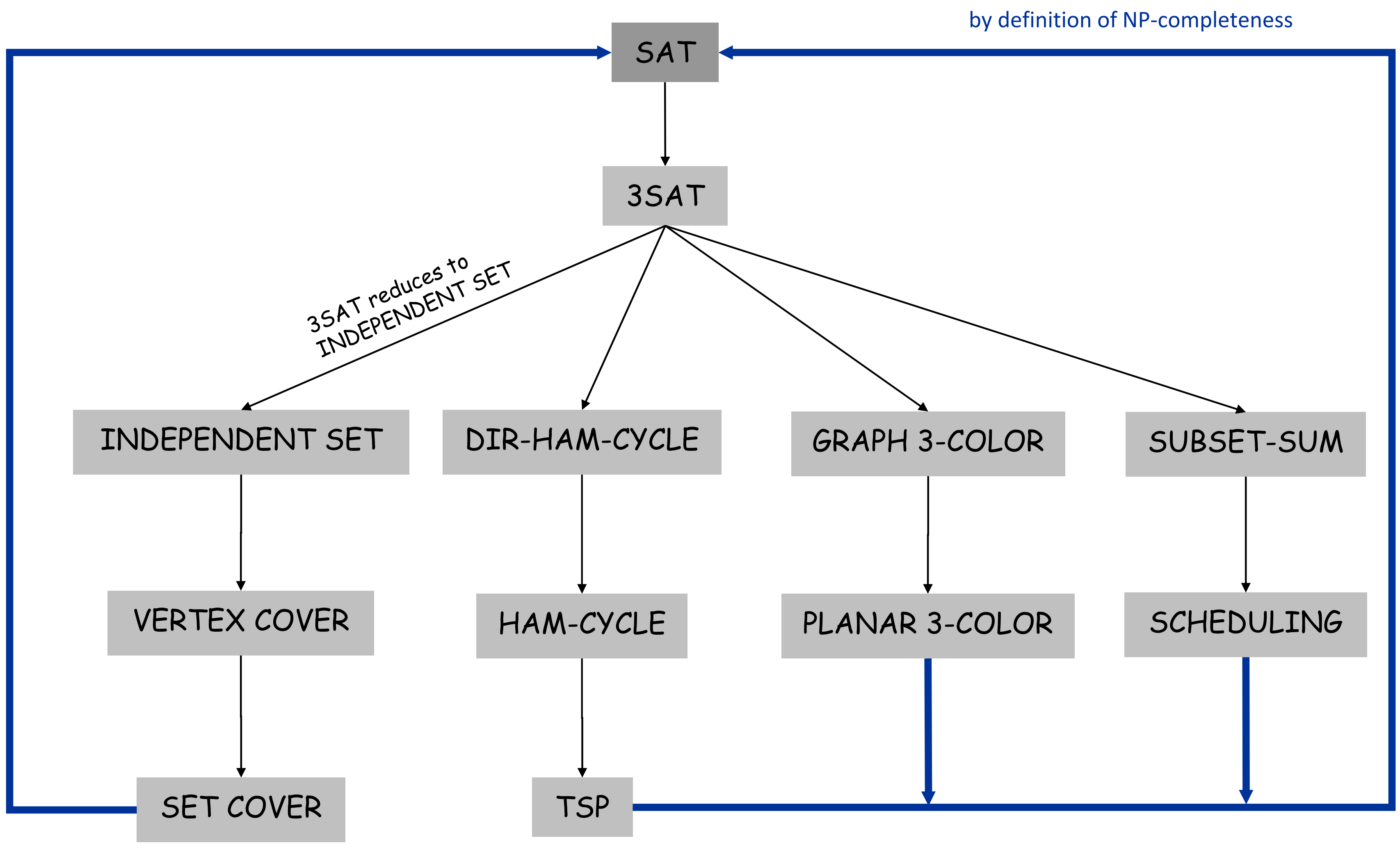
If

1) $C \in \mathrm{NP}$ and
2) There is an NP-complete language B (e.g., 3SAT, VERTEX-COVER, IND-SET, …) such that $B \leq_{\mathrm{p}} C$,

then C is also NP-complete.

New NP-complete problems from old

All problems below are NP-complete and hence poly-time reduce to one another!



$3SAT$ (3-CNF Satisfiability)

Definitions:

- A literal either a variable of its negation $\qquad x_5\ ,\ \overline{x_7}$
- A clause is a disjunction (OR) of literals $\qquad$ Ex. $x_5 \lor \overline{x_7} \lor x_2$
- A 3-CNF is a conjunction (AND) of clauses where each clause contains exactly 3 literals

Ex. $C_1 \land C_2 \land \ldots \land C_m =$

$$(x_5 \lor \overline{x_7} \lor x_2) \land (\overline{x_3} \lor x_4 \lor x_1) \land \cdots \land (x_1 \lor x_1 \lor x_1)$$

$$3SAT = \{\langle \varphi \rangle \mid 3-\text{CNF formula } \varphi \text{ is satisfiable }\}$$

Last time: We argued SAT $\leq_p$ 3-SAT
so by Cook-Levin Thm (i.e., SAT is NP-complete) know 3-SAT is also NP-complete

Some general reduction strategies

- Reduction from special case to general case

  Ex. $VERTEX - COVER \leq_p SET - COVER$

  $3SAT \leq_p SAT$

  SAT $\leq_p$ SAT

- Reduction by simple equivalence

  Ex. $IND - SET \leq_p VERTEX - COVER$

  $VERTEX - COVER \leq_p IND - SET$

- "Gadget" reductions

  Ex. $SAT \leq_p 3SAT$

  $3SAT \leq_p IND - SET$

Independent Set

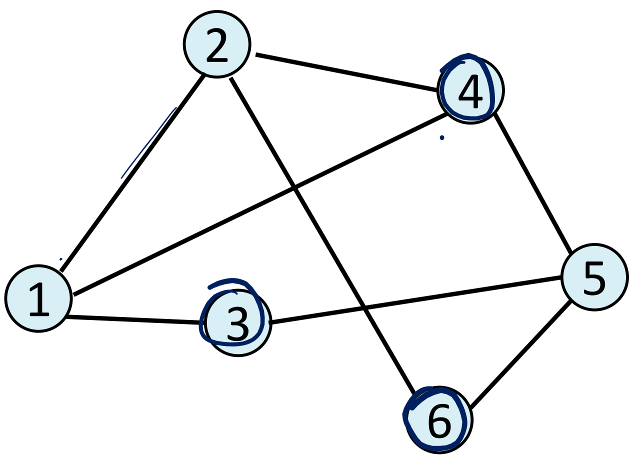
An **independent set** in an undirected graph G is a set of vertices that includes at most one endpoint of every edge.

$$IND-SET = \{\langle G, k\rangle | G \text{ is an undirected graph containing an independent set with} \geq k \text{ vertices}\}$$

$\langle G, 2\rangle \in$ IND-SET
$\langle G, 3\rangle \in$ IND-SET

$\langle G, 4\rangle \notin$ IND-SET

$G =$



Which of the following are independent sets in this graph?

a) $\{1\}$
b) $\{1, 5\}$
c) $\{2, 3, 6\}$
d) $\{3, 4, 6\}$

Independent Set is NP-complete

1) $IND - SET \in \mathrm{NP}$
2) Reduce $3SAT \leq_{\mathrm{p}} IND - SET$

[⇒ IND-SET is NP-hard, since we know 3SAT is NP-hard]

Proof of 1) The following gives a poly-time verifier for $IND - SET$

Certificate: Vertices $v_1, \ldots, v_k$ which form an alleged independent set of size k

Verifier:

"On input $\langle G, k; v_1, \ldots, v_k \rangle$, where G is a graph, k is a natural number,

1. Check that $v_1, \ldots v_k$ are distinct vertices in G
2. Check that there are no edges between the v_i's."

Check that $v_1, \ldots, v_k$ form an independent set of size k in G.

Independent Set is NP-complete

- $B \in NP$
- A (known NP-complete)
  $A \leq_p B$
  $\Rightarrow B$ is NP-complete

1) $IND-SET \in \mathrm{NP}$
2) Reduce $3SAT \leq_{\mathrm{p}} IND-SET$

3CNF formula φ is satisfiable $\Leftrightarrow$ G has an independent set of size $\geq k$

Proof of 2) The following TM computes a poly-time reduction.

"On input $\langle\varphi\rangle$, where φ is a 3CNF formula,

1. Construct graph G from φ
   - G contains 3 vertices for each clause, one for each literal.
   - Connect 3 literals in a clause in a triangle.
   - Connect every literal to each of its negations.
2. Output $\langle G, k\rangle$, where k is the number of clauses in φ."

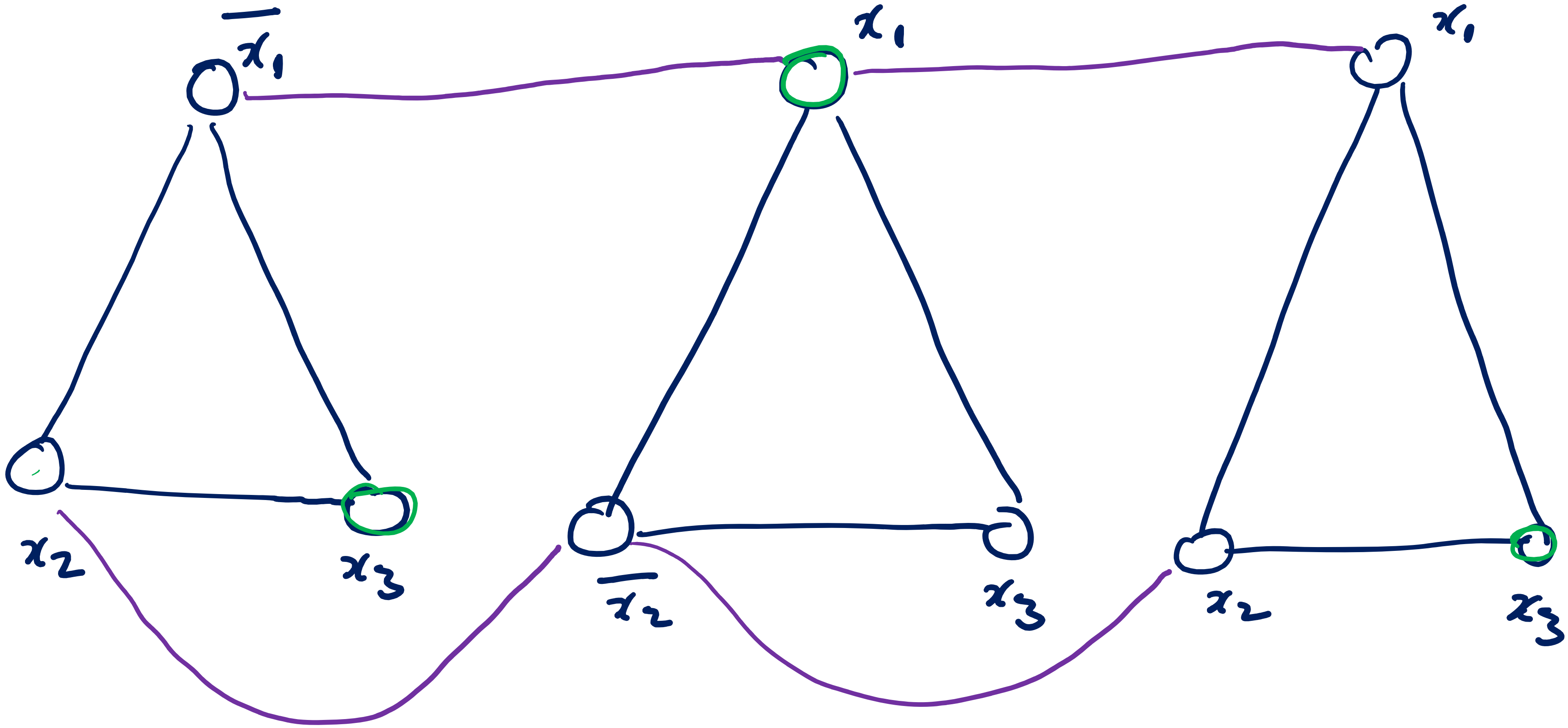
Example of the reduction

Claim: Any satisfying assignment to φ induces an independent set of size 3 in G

$$\varphi = (\overline{x_1} \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_3)$$

Example sat. assignment: $x_1 = 1$, $x_2 = 1$, $x_3 = 1$

$G =$



$x_3 = 1$ $x_1 = 1$ x_2 = either 0 or 1

$k = 3$

Proof of correctness for reduction

Let k = # clauses and l = # literals in φ

Correctness: φ is satisfiable iff G has an independent set of size k

φ satisfiable $\Rightarrow$ $\exists$ a satisfying assignment

$\Longrightarrow$ Given a satisfying assignment, select one true literal from each triangle. This is an independent set of size k

$\Longleftarrow$ Let S be an independent set in G of size k

- S must contain exactly one vertex in each triangle
- Set these literals to true, and set all other variables arbitrarily
- Truth assignment is consistent and all clauses are satisfied

Runtime: $O(k + l^2)$ which is polynomial in input size

Some general reduction strategies

- Reduction by simple equivalence

  Ex. $\boxed{\begin{array}{l} IND - SET \leq_{\mathrm{p}} VERTEX - COVER \\ VERTEX - COVER \leq_{\mathrm{p}} IND - SET \end{array}}$

- Reduction from special case to general case

  Ex. $VERTEX - COVER \leq_{\mathrm{p}} SET - COVER$

  $3SAT \leq_{\mathrm{p}} SAT$

- "Gadget" reductions

  Ex. $\boxed{SAT \leq_{\mathrm{p}} 3SAT}$

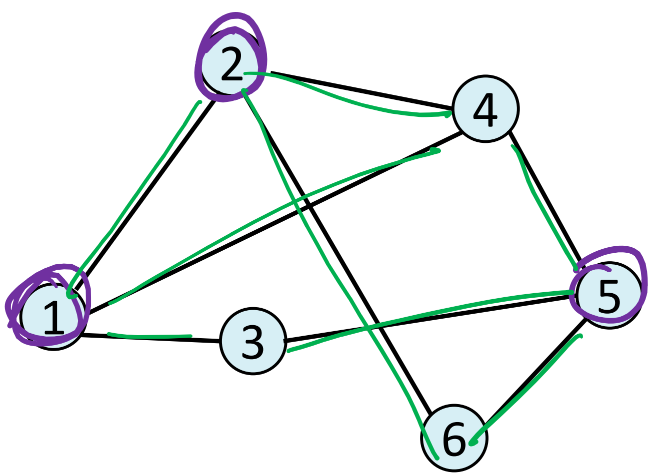
  $\boxed{3SAT \leq_{\mathrm{p}} IND - SET}$

Vertex Cover

Set S of vertices s.t. $\forall$ edges $(u,v) \in E$, either u or $v \in S$

Given an undirected graph G, a **vertex cover** in G is a subset of nodes which includes at ***least*** one endpoint of every edge.

$VERTEX-COVER$ = $\{\langle G, k\rangle$ | G is an undirected graph which has a vertex cover with $\leq k$ vertices$\}$



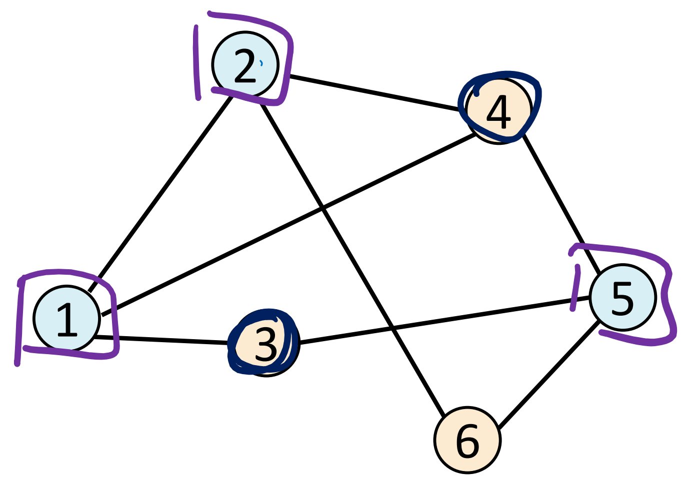
Independent Set and Vertex Cover

Claim. S is an independent set iff $V \setminus S$ is a vertex cover.

$\Longrightarrow$ Let S be any independent set.

- Consider an arbitrary edge (u, v).
- S is independent $\Longrightarrow u \notin S$ or $v \notin S \Longrightarrow u \in V \setminus S$ or $v \in V \setminus S$.
- Thus, $V \setminus S$ covers (u, v).

$\forall$ edge (u,v), either



$\Longleftarrow$ Let $V \setminus S$ be any vertex cover.

- Consider two nodes $u \in S$ and $v \in S$.
- Then $(u, v) \notin E$ since $V \setminus S$ is a vertex cover.
- Thus, no two nodes in S are joined by an edge $\Longrightarrow S$ is an independent set.

INDEPENDENT SET reduces to VERTEX COVER

Theorem. IND-SET $\leq_{\mathrm{p}}$ VERTEX-COVER.

What do we need to do to prove this theorem?

a) Construct a poly-time nondet. TM deciding IND-SET

b) Construct a poly-time deterministic TM deciding IND-SET

c) Construct a poly-time nondet. TM mapping instances of IND-SET to instances of VERTEX-COVER

d) Construct a poly-time deterministic TM mapping instances of IND-SET to instances of VERTEX-COVER

e) Construct a poly-time nondet. TM mapping instances of VERTEX-COVER to instances of IND-SET

f) Construct a poly-time deterministic TM mapping instances of VERTEX-COVER to instances of IND-SET

INDEPENDENT SET reduces to VERTEX COVER

S is an ind set in G
$\Leftrightarrow$ V \ S is a vertex cover

Theorem. IND-SET $\leq_{\mathrm{p}}$ VERTEX-COVER.

Proof. The following TM computes the reduction.

"On input $\langle G, k \rangle$, where G is an undirected graph and k is an integer,

instance of IND-SET

1. Output $\langle G, n-k \rangle$, where n is the number of nodes in G."

instance of VERTEX-COVER

Correctness:

$\exists$ a set S of size $\geq k$ that is independent in G

$\Updownarrow$

- G has an independent set of size at least k iff it has a vertex cover of size at most $n - k$.

$\Leftrightarrow \exists$ a set V \ S of size $\leq n-k$ that is a vertex cover in G

Runtime:

- Reduction runs in linear time.

on a multi-tape TM, at least

VERTEX COVER reduces to INDEPENDENT SET

Theorem. VERTEX-COVER $\leq_{\mathrm{p}}$ IND-SET

Proof. The following TM computes the reduction.

"On input $\langle G, k\rangle$, where G is an undirected graph and k is an integer,

an instance of VERTEX-COVER

1. Output $\langle G, n-k\rangle$, where n is the number of nodes in G."

an instance of IND-SET

Correctness:

- G has a vertex cover of size at most k iff it has an independent set of size at least $n-k$. so $\langle G,k\rangle \in$ VERTEX-COVER $\Leftrightarrow \langle G, n-k\rangle \in$ IND-SET

Runtime:

- Reduction runs in linear time.

Subset Sum

SUBSET-SUM = $\{\langle w_1, \ldots, w_m, t\rangle \mid$
there exists a subset of natural numbers $w_1, \ldots, w_m$ that sum to $t\}$

Ex: $\langle 3, 7, 2, 1, 4, 9\rangle \in$ SUBSET-SUM
$w_1 \; w_2 \; w_3 \; w_4 \; w_5 \; t$

because $w_2 + w_3 = 7+2 = 9 \; (=t)$

Theorem: SUBSET-SUM is NP-complete

Claim 1: SUBSET-SUM is in NP NTM:

1. Nondet. guess $S \subseteq [m]$
2. Check that $w_{i_1} + w_{i_2} + \ldots + w_{i_\ell} = t$ where $S = \{i_1, \ldots, i_\ell\}$

Claim 2: SUBSET-SUM is NP-hard

$3\mathrm{SAT} \leq_p$ SUBSET–SUM

Goal: Given a 3CNF formula φ on v variables and k clauses, construct a SUBSET-SUM instance $w_1, \ldots, w_m, t$ such that

φ is satisfiable iff there exists a subset of $w_1, \ldots, w_m$ that sum to t

$x_1, \bar{x}_1, x_2, \bar{x}_2 \ldots$

First attempt: Encode each literal ℓ of φ as a k-digit *decimal* number $w_\ell = c_1 \ldots c_k$ where

$$c_i = \begin{cases} 1 \text{ if } \ell \text{ appears in clause } i \\ 0 \text{ otherwise} \end{cases}$$

Example of the first attempt reduction

$$\varphi = (\overline{x_1} \lor x_2 \lor x_3) \land (x_1 \lor \overline{x_2} \lor x_3)$$

Ex. sat assignment

$x_1 = 1 \quad x_2 = 1 \quad x_3 = 1$

		Clause 1	clause 2	
x_1	$\rightarrow$	0	1	01
$\overline{x_1}$	$\rightarrow$	1	0	10
x_2	$\rightarrow$	1	0	10
$\overline{x_2}$	$\rightarrow$	0	1	01
x_3	$\rightarrow$	1	1	11
$\overline{x_3}$	$\rightarrow$	0	0	00

$+$

22

Note: $2 \geq 1$ and $1 \geq 1$

Idea: Every sat assignment to φ induces a subset of #s that digit-wise sum up to $\geq 11\ldots1$

$3SAT \leq_p$ SUBSET−SUM

First attempt: Encode each literal ℓ of φ as a k-digit *decimal* number $w_\ell = c_1 \dots c_k$ where

$$c_i = \begin{cases} 1 \text{ if } \ell \text{ appears in clause } i \\ \quad 0 \text{ otherwise} \end{cases}$$

Claim: If φ is satisfiable, then there exists a subset of the w_ℓ's that "digit-wise" add up to "at least" $111 \dots 11$

Two issues:

1) Need to enforce that exactly one of $\ell, \overline{\ell}$ is set to 1
2) Need the subset to add up to exactly some target

$3SAT \leq_p$ SUBSET−SUM

Actual reduction: Encode each literal ℓ of φ as a $(v+k)$-digit *decimal* number $w_\ell = b_1 \dots b_v |\ c_1 \dots c_k$ where

$$b_i = \begin{cases} 1 \text{ if } \ell \in \{x_i, \overline{x_i}\} \\ 0 \text{ otherwise} \end{cases} \qquad c_i = \begin{cases} 1 \text{ if } \ell \text{ appears in clause } i \\ 0 \text{ otherwise} \end{cases}$$

Also, include two copies each of 000…0|100…0, 000…0|010…0, … 000…0|0…01

Claim: φ is satisfiable if and only if there exists a subset of the numbers that add up to $t = 111 \dots 11|333 \dots 33$

Example of the reduction

$\varphi = (\overline{x_1} \vee x_2 \vee x_3) \wedge (x_1 \vee \overline{x_2} \vee x_3)$

	b_1	b_2	b_3	c_1	c_2
x_1	1	0	0	0	1
$\overline{x_1}$	1	0	0	1	0
x_2	0	1	0	1	0
$\overline{x_2}$	0	1	0	0	1
x_3	0	0	1	1	1
$\overline{x_3}$	0	0	1	0	0
p_{11}	0	0	0	1	0
p_{12}	0	0	0	1	0
p_{21}	0	0	0	0	1
p_{22}	0	0	0	0	1

Encode literal ℓ as $w_\ell = b_1 \dots b_v | \; c_1 \dots c_k$ where

$$\boxed{b_i = \begin{cases} 1 \text{ if } \ell \in \{x_i, \overline{x_i}\} \\ 0 \text{ otherwise} \end{cases}}$$

$$c_i = \begin{cases} 1 \text{ if } \ell \text{ appears in clause } i \\ 0 \text{ otherwise} \end{cases}$$

Include two copies each of 000…0|100…0, 000…0|010…0, … 000…0|0…01

Eg.

assignment $x_1 = 1$, $x_2 = 1$, $x_3 = 1$

```
  100 01
  010 10
  001 11
+ 000 10
  000 01
  ------
  111 33
```

target: 111 33