Carlos Mc Gregor c.mcgregormuro@mail.utoronto.ca

CSC373S: Algorithm Design, Analysis & Complexity

Lecture 30

Wednesday March 22, 2017
based on notes by Denis Pankratov

NP Complete Problems *(cont.)*

Subset Sum

Input: $A = \{a_1, .., a_n\} \subseteq \mathbb{Z}_{>0}$
$t \in \mathbb{Z}_{>0}$ **Output:** 1 if $\exists S \subseteq [n], \sum_{i \in S} a_i = t$
0 otherwise
$L_{SUB-SUM} = \{< A = \{a_1, .., a_n\}, t > | \exists S \subseteq [n], \sum_{i \in S} a_1 = t$

Claim: $L_{SUB-SUM} \in NPC$
Proof:

1. $L_{SUB-SUM} \in NP$
   Certificate - $S \subseteq [n]$. Verifier checks $\sum_{i \in S} a_i = t$
   Clearly, certificate is of *polysize*, verifier runs in *polytime*

2. $L_{3-SAT} \leq_p L_{SUB-SUM}$
   Given φ-3-CNF over variables $x_1, .., x_n$ & has clauses $C_1, .., C_m$, we want to construct in polytime $A \subseteq \mathbb{Z}_{>0}, t \in \mathbb{Z}_{>0}$ so that φ is satisfied $\Leftrightarrow < A, t > \in L_{SUB-SUM}$.

   Construct integers of the form:

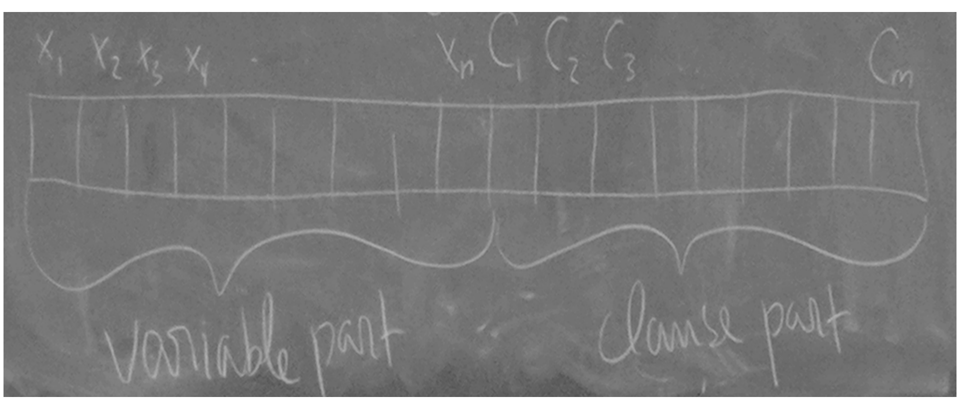

Figure 1: *Visualization of variable and clause parts*

   Integers will be over base 10. For each variable x_i, introduce integer v_i - hast 1 at position x_i in the variable part & 1 at position C_i for each C_i that is satisfied by $x_i = T$ (it has 0s everywhere else).

v_i' - has 1 at position x_i in the variable part, and for each position C_i so that C_i is satisfied by $x_i = F$.
0s everywhere else.

For each clause, introduce ints:
S_i - has 1 at position C_i & 0s everywhere else
S_i' - has 2 at position C_i & 0s everywhere else

Example:
$C_1 = x_1 \vee \neg x_2 \vee \neg x_3$
$C_2 = \neg x_1 \vee \neg x_2 \vee \neg x_3$
$C_3 = \neg x_1 \vee \neg x_2 \vee x_3$

	x_1	x_2	x_3	c_1	c_2	c_3
v_1	1	0	0	1	0	0
v_1'	1	0	0	0	1	1
v_2	0	1	0	0	0	0
v_2'	0	1	0	1	1	1
v_3	0	0	1	0	0	1
v_3'	0	0	1	1	1	0

	x_1	x_2	x_3	c_1	c_2	c_3
s_1	0	0	0	1	0	0
s_1'	0	0	0	2	0	0
s_2	0	0	0	0	1	0
s_2'	0	0	0	0	2	0
s_3	0	0	0	0	0	1
s_3'	0	0	0	0	0	2

Target t - has 1s in the variable part, 4s in the clause part.

Example:

	x_1	x_2	x_3	c_1	c_2	c_3
$t =$	1	1	1	4	4	4

Can construct v_i, v_i', s_i, s_i', t in polytime.
NTS: φ is satisfied $\Leftrightarrow < v_1, .., v_n, v_1', .., v_n', s_1, .., s_m, s_1', .., s_m', t > \in L_{SUB-SUM}$
$\Rightarrow \varphi$ is satisfied. Let x be a satisfied assigment. Select the following ints for the set:
if $x_i = T$, select v_i
if $x_i = F$, select v_i'

If C_i is satisfied by 1 literal, select $s_i \& s_i'$
If C_i is satisfied by 2 literals, select s_i'
If C_i is satisfied by 3 literal, select s_i
$\Rightarrow$ subset adds up to t

$\Leftarrow$ Let S be the subset of ints that add up to t.
Comment: assume that the same variable does not appear twice in the same clause.
No clauses like:

- $x_1 \vee x_1 \vee \neg x_3 \equiv x_1 \vee \neg x_3$ -2-CNF
- $x_1 \vee \neg x_1 \vee x_3 \equiv T$

Note: for every subset of integers, if you add up those integers, there are no carries (because max a digit can be is 6).

This implies that exactly *one* of v_i, v_i' belongs to S for each i.

$\Leftarrow$ Let x be the assignment:
if $v_i \in S$, set $x_i = T$
if $v_i' \in S$, set $x_i = F$

$\Rightarrow$ x is a satisfied assignment (*Exercise*: check this) ☑

Knapsack

Input: $v_1, ..v_n \in \mathbb{Z}_{>0}$ - values of n items
$w_1, ..w_n \in \mathbb{Z}_{>0}$ - weights
W - capacity of knapsack

Output: $\max_{S \subseteq [n]} \sum_{i \in S} v_i$ so that $\sum_{i \in S} w_i \leq W$
$L_{KNAPSACK} = \{< v_1, .., v_n, w_1, .., w_n, W, k > | \exists S \subseteq [n] : \sum_{i \in S} v_i \geq k \ \& \ \sum_{i \in S} w_i \geq W\}$

Claim: $L_{SUB-SUM} \leq_p L_{KNAPSACK}$
Proof: Given $A = \{a_1, .., a_n\}, t \in \mathbb{Z}_{>0}$
Construct:
$\exists S \subseteq [n] \sum_{i \in S} a_i = t$
$v_i = a_i$, $w_i = a_i \Leftrightarrow \sum_{i \in S} a_i \leq t \wedge \sum_{i \in S} a_i \geq t$
$W = t$, $k = t \Leftrightarrow \sum_{i \in S} v_i \geq t \wedge \sum_{i \in S} w_i \leq t$
☑
$L_{PART} = \{< A = \{a_1, .., a_n\} > | \exists S \subseteq [n] : \sum_{i \in S} a_i = \sum_{i \notin S} a_i\}$
Special case of SS: $t = \frac{\sum_{i=1}^n a_i}{2}$

Exercise:
$L_{SUB-SUM} \leq_p L_{PART}$
Hint: Introduce $a_{n+1} = \sum_{i=1}^n a_1 + t$
$a_{n+2} = 2 \sum_{i=1}^n a_1 - t$