The NP-completeness of Subset Sum

Pilu Crescenzi and Viggo Kann

University of Florence and KTH

October 2011

Basic definitions

- Class NP
  - Set of decision problems that admit "short" and efficiently verifiable solutions
  - Formally, $L \in \mathtt{NP}$ if and only if there exist
    - polynomial p
    - polynomial-time machine V
    - such that, for any x,

$$x \in L \Leftrightarrow \exists y \, (|y| \leq p(|x|) \wedge V(x, y) = 1)$$

- Polynomial-time reducibility
  - $L_1 \leq L_2$ if there exists polynomial-time computable function f such that, for any x,

$$x \in L_1 \Leftrightarrow f(x) \in L_2$$

- $\mathtt{NP}$-complete problem
  - $L \in \mathtt{NP}$ is $\mathtt{NP}$-complete if any language in $\mathtt{NP}$ is polynomial-time reducible to L
    - Hardest problem in $\mathtt{NP}$

Basic results

- Cook-Levin theorem
  - Sat problem
    - Given a boolean formula in conjunctive normal form (disjunction of conjunctions), is the formula satisfiable?
  - Sat is NP-complete
- 3-Sat
  - Each clause contains exactly three literals
- 3-Sat is NP-complete
  - Simple proof by local substitution
    - $l_1 \Rightarrow (l_1 \vee y \vee z) \wedge (l_1 \vee y \vee \overline{z}) \wedge (l_1 \vee \overline{y} \vee z) \wedge (l_1 \vee \overline{y} \vee \overline{z})$
    - $l_1 \vee l_2 \Rightarrow (l_1 \vee l_2 \vee y) \wedge (l_1 \vee l_2 \vee \overline{y})$
    - $l_1 \vee l_2 \vee l_3 \Rightarrow l_1 \vee l_2 \vee l_3$
    - $l_1 \vee l_2 \vee \cdots \vee l_k \Rightarrow$

$$(l_1 \vee l_2 \vee y_1) \wedge (\overline{y_1} \vee l_3 \vee y_2) \wedge (\overline{y_2} \vee l_4 \vee y_3) \wedge \cdots \wedge (\overline{y_{k-3}} \vee l_{k-1} \vee l_k)$$

Problem definition: Subset Sum

Given a (multi)set A of integer numbers and an integer number s, does there exist a subset of A such that the sum of its elements is equal to s?

- No polynomial-time algorithm is known
- Is in NP (short and verifiable certificates):
  - If a set is "good", there exists subset $B \subseteq A$ such that the sum of the elements in B is equal to s
  - Length of B encoding is polynomial in length of A encoding
  - There exists a polynomial-time algorithm that verifies whether B is a set of numbers whose sum is s:
    - Verify that $\sum_{a \in B} a = s$

NP-completeness

- Reduction of 3−Sat to Subset Sum:
  - n variables x_i and m clauses c_j
- For each variable x_i, construct numbers t_i and f_i of $n+m$ digits:
  - The i-th digit of t_i and f_i is equal to 1
  - For $n+1 \leq j \leq n+m$, the j-th digit of t_i is equal to 1 if x_i is in clause c_{j-n}
  - For $n+1 \leq j \leq n+m$, the j-th digit of f_i is equal to 1 if $\overline{x_i}$ is in clause c_{j-n}
  - All other digits of t_i and f_i are 0
- Example:
  $(x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee x_3)$

	i			j			
Number	1	2	3	1	2	3	4
t_1	1	0	0	1	0	0	1
f_1	1	0	0	0	1	1	0
t_2	0	1	0	1	0	1	0
f_2	0	1	0	0	1	0	1
t_3	0	0	1	1	1	0	1
f_3	0	0	1	0	0	1	0

- For each clause c_j, construct numbers x_j and y_j of $n+m$ digits:
  - The $(n+j)$-th digit of x_j and y_j is equal to 1
  - All other digits of x_i and y_j are 0
- Example:
  $(x_1 \vee x_2 \vee x_3) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee x_3)$

	i			j			
Number	1	2	3	1	2	3	4
x_1	0	0	0	1	0	0	0
y_1	0	0	0	1	0	0	0
x_2	0	0	0	0	1	0	0
y_2	0	0	0	0	1	0	0
x_3	0	0	0	0	0	1	0
y_3	0	0	0	0	0	1	0
x_4	0	0	0	0	0	0	1
y_4	0	0	0	0	0	0	1

- Finally, construct a sum number s of $n+m$ digits:
  - For $1 \leq j \leq n$, the j-th digit of s is equal to 1
  - For $n+1 \leq j \leq n+m$, the j-th digit of s is equal to 3

Proof of correctness

- Show that Formula satisfiable $\Rightarrow$ Subset exists:
  - Take t_i if x_i is true
  - Take f_i if x_i is false
  - Take x_j if number of true literals in c_j is at most 2
  - Take y_j if number of true literals in c_j is 1
  - Example
    - $(x_1 \lor x_2 \lor x_3) \land (\overline{x_1} \lor \overline{x_2} \lor x_3) \land (\overline{x_1} \lor x_2 \lor \overline{x_3}) \land (x_1 \lor \overline{x_2} \lor x_3)$
    - All variables true

	i			j			
Number	1	2	3	1	2	3	4
t_1	1	0	0	1	0	0	1
t_2	0	1	0	1	0	1	0
t_3	0	0	1	1	1	0	1
x_2	0	0	0	0	1	0	0
y_2	0	0	0	0	1	0	0
x_3	0	0	0	0	0	1	0
y_3	0	0	0	0	0	1	0
x_4	0	0	0	0	0	0	1
s	1	1	1	3	3	3	3

- Show that Subset exists ⇒ Formula satisfiable:
  - Assign value true to x_i if t_i is in subset
  - Assign value false to x_i if f_i is in subset
  - Exactly one number per variable must be in the subset
    - Otherwise one of first n digits of the sum is greater than 1
  - Assignment is consistent
  - At least one variable number corresponding to a literal in a clause must be in the subset
    - Otherwise one of next m digits of the sum is smaller than 3
  - Each clause is satisfied