Lecture 25 Cook's Theorem

In this lecture we will prove the *NP*-hardness of CNFSat by exhibiting a reduction from an arbitrary problem in *NP* to CNFSat. We use the standard definition of one-tape deterministic and nondeterministic Turing machines; see for example [3, pp. 25ff.]. This landmark result was proved by S. Cook in 1971 [22]. A similar result was proved independently by L. Levin in the Soviet Union in 1973 [71].

Theorem 25.1 *If $A \in NP$ then $A \leq_{\mathrm{m}}^{\mathrm{p}}$* CNFSat.

Proof. Let $A \subseteq \Sigma^*$ be an arbitrary but fixed language in *NP*. Then A is accepted by some nondeterministic Turing machine M. We will describe a function σ that from a given $x \in \Sigma^*$ computes a Boolean formula $\mathcal{B} = \sigma(x)$ that is satisfiable iff M accepts x. The function σ must be computable in polynomial time deterministically, and its description may depend on M.

Here is the main idea. The possible executions of M on input $x \in \Sigma^*$ form a branching tree of *configurations*, where each configuration gives a snapshot of the current instantaneous state of the computation and includes all relevant information that can affect the computation, such as tape contents, head position, and current state of the finite control. Since M is polynomially time bounded, we can assume that the depth of this tree is at most $N = |x|^k$ for some fixed k. The exponent k may depend on M but does not depend on x. A valid computation sequence of length N can use no more than N tape cells, since at the very worst the machine moves right one tape cell in each step. Thus there are at most N time units and N tape cells we need to consider.

We will encode computations of M on input x as truth assignments to various arrays of Boolean variables, which describe things like where the read head is at time i, which symbol is occupying cell j at time i, and so forth. We will write down clauses involving these variables that will describe legal moves of the machine and legal starting and accepting configurations of M on x. A truth assignment will simultaneously satisfy all these clauses iff it describes a valid computation sequence of M on input x. We will then take $\mathcal{B} = \sigma(x)$ to be the conjuction of all these clauses. Then the satisfying truth assignments to $\mathcal{B}$ correspond in a one-to-one fashion to the accepting computations of M on x, therefore $\mathcal{B}$ will be satisfiable iff M has an accepting computation on input x, *i.e.* iff $x \in A$.

Here are the Boolean variables, along with their intuitive interpretations. Let Q denote the set of states of the finite control of M, and let Σ denote the tape alphabet of M.

- Q_i^q, $0 \le i \le N$, $q \in Q$; intuitively,
$$Q_i^q \quad = \quad \text{"At time } i\text{, the machine is in state } q\text{."}$$
- H_{ij}, $0 \le i, j \le N$; intuitively,
$$H_{ij} \quad = \quad \text{"At time } i\text{, the machine's read/write head is scanning tape cell } j\text{."}$$
- S_{ij}^a, $0 \le i, j \le N$, $a \in \Sigma$; intuitively,
$$S_{ij}^a \quad = \quad \text{"At time } i\text{, tape cell } j \text{ contains symbol } a\text{."}$$

The machine starts in its start state s scanning the left endmarker $\vdash$ with the input x filling the first $|x|$ spaces on the tape followed by blank characters $\natural$. This situation is captured by the following formula:

$$Q_0^s \wedge H_{00} \wedge S_{00}^{\vdash} \wedge \bigwedge_{1 \le j \le |x|} S_{0j}^{x_j} \wedge \bigwedge_{|x|+1 \le j \le N} S_{0j}^{\natural} \ .$$

Assume that M never prints its left endmarker $\vdash$ anyplace except in the leftmost cell of the tape, and that upon seeing $\vdash$ in any state, it never moves left. Assume further that if M wants to accept, it first erases its tape and moves its head all the way to the left before entering the accept state t, and subsequently does not move its head or change state. These assumptions are without loss of generality, since if A is accepted by a nondeterministic polynomial time machine at all, then it is accepted by another machine that satisfies these conditions.

The acceptance condition can then be represented by the formula

$$Q_N^t \wedge H_{N,0} \wedge S_{N,0}^{\vdash} \wedge \bigwedge_{1 \le j \le N} S_{0,j}^{\natural} \ .$$

The computation of the machine obeys certain constraints, which are represented by various formulas:

- "At any time, the machine is in exactly one state."

$$\bigwedge_{0\le i\le N}\,(\bigvee_{q\in Q} Q_i^q) \quad\wedge\quad \bigwedge_{0\le i\le N}\ \bigwedge_{\substack{p,q\in Q\\ p\ne q}} (\neg Q_i^p \vee \neg Q_i^q)$$

- "At any time, each tape cell contains exactly one symbol."

$$\bigwedge_{0\le i,j\le N}\,(\bigvee_{a\in\Sigma} S_{i,j}^a) \quad\wedge\quad \bigwedge_{0\le i,j\le N}\ \bigwedge_{\substack{a,b\in\Sigma\\ a\ne b}} (\neg S_{ij}^a \vee \neg S_{ij}^b)$$

- "At any time, the machine is scanning exactly one cell."

$$\bigwedge_{0\le i\le N}\,(\bigvee_{0\le j\le N} H_{ij}) \quad\wedge\quad \bigwedge_{0\le i\le N}\ \bigwedge_{0\le j<k\le N} (\neg H_{ij} \vee \neg H_{ik})$$

The last set of conditions we need to write down are the most crucial to this construction. They say that computation follows the transition relation of M. There are clauses that specify, based on the state, head position, and contents of the tape at time i, the possible state, head position, and contents of the tape at time $i+1$.

The transition relation of M is the part of the specification of M that tells which actions M can take in a given situation. Formally, it is a finite set δ of tuples of the form $((p,a),(q,b,d))$, where

- p and q are states of the finite control,
- a and b are tape symbols, and
- d is a direction, either -1 (left), 0 (stationary), or $+1$ (right).

If the tuple $((p,a),(q,b,d))$ is in δ, this says that whenever the machine is in state p scanning symbol a, it can take the following actions: print b on that tape cell, move the head in direction d, and enter state q. Since δ is a relation and not a function, there may be several such actions (q,b,d) possible for a given (p,a), but it is important to note that the number of such (q,b,d) depends only on M and is independent of the size of the input x.

The following two formulas express that the configuration at time $i+1$ follows from that at time i according to the transition relation δ. Formula (34) says that for a given $p\in Q$, $a\in\Sigma$, and $0\le i,j\le N$, if M at time i is in state p scanning cell j on which is written the symbol a, then for some tuple $((p,a),(q,b,d))\in\delta$, at time $i+1$ there will be a b occupying cell j and M will be in state q scanning cell $j+d$. Formula (35) says that any cell not being scanned at time i contains the same symbol at time $i+1$ as at time i.

$$Q_i^p \wedge H_{ij} \wedge S_{ij}^a \quad\to \bigvee_{((p,a),(q,b,d))\in\delta} (Q_{i+1}^q \wedge H_{i+1,j+d} \wedge S_{i+1,j}^b) \qquad (34)$$

$$S_{ij}^a \wedge \neg H_{ij} \quad\to\quad S_{i+1,j}^a\ . \qquad (35)$$

These formulas are not in CNF as they stand, but can be transformed into equivalent CNF formulas using the distributive and DeMorgan laws of propositional logic. Do not worry about (34) and (35) getting too big in this process—their lengths depend only on M and not on the size of the input x, hence are $O(1)$. We take the conjunction of (34) and (35) over all i and j in the range 0 to N.

The formula $\mathcal{B}$ is the conjuction of all these formulas. It is in conjunctive normal form, and its length is polynomial in $|x|$. Moreover, it can be constructed from x in polynomial time. Every satisfying truth assignment to $\mathcal{B}$ gives rise to an accepting computation of the machine, and vice-versa. □