The Satisfiability Problem: Given a boolean expression E in the boolean variables $x_1, x_2, ..., x_n$, is there an assignment of `true` and `false` values to each the variables such that the expression E evaluates to `true` ?

It is very easy to cast the satisfiability problem in terms of a language. Let,

$$\text{SAT} = \{ E \mid E \text{ is a satisfiable boolean expression.} \}$$

Theorem The language SAT is $\mathcal{NP}$-complete.

Proof: We need to show that every language L in NP is polynomial-time mapping reducible to SAT. Let L be an arbitrary language in NP, and let N denote a non-deterministic Turing machine that decides L.

$$N = \left(Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}} \right)$$

Consider the computation performed by N on input w. Machine N accepts input w if and only if there exists an accepting computation history:

$$I_0 \vdash I_1 \vdash I_2 \vdash .... \vdash I_t$$

where $t \leq p(n)$ for some polynomial p, and where $n = |w|$. Each I_j, for $0 \leq j \leq t$, is called an *instantaneous description*, abbreviated as ID. To simplify our proof, we will assume that the machine is modified to "keep running" until exactly $p(n)$ steps have been taken. I.e., $t = p(n)$.

The proof proceeds by constructing a boolean expression (in polynomial time) that simulates the computation history for N. I.e., we will construct a boolean expression w_0 which is satisfiable if and only if N accepts w. For discussion purposes, we define:

$$m = |\Gamma| \qquad \text{and} \qquad s = |Q| \quad .$$

The key argument in this proof is given as follows: Machine N accepts w if and only if the following seven conditions hold:

1. the tape head is scanning exactly one cell in each ID
2. each ID has exactly one symbol in each tape cell
3. each I has exactly one state
4. at most one tape cell, the one scanned by the head is modified from one ID to the next
5. the change in state, head location, and tape cell contents between successive IDs is allowed by the move function δ.
6. the first ID is an initial ID
7. the state in the last ID is the accepting state.

To formalize conditions 1 through 7, we introduce the following boolean variables:

1. $C[i, j, t]$ is `true` if and only if the i^{th} cell contains symbol j at time t. Observe: $1 \leq i \leq p(n)$, $1 \leq j \leq m$, and $0 \leq t \leq p(n)$.

   Notice $C[i, j, t]$ defines $\mathcal{O}(p^2(n))$ boolean variables.

2. $S[k, t]$ is `true` if and only if machine N is in state q_k at time t. Observe: $1 \leq k \leq s$ and $0 \leq t \leq p(n)$.

   Notice $S[k, t]$ defines $\mathcal{O}(p(n))$ boolean variables.

3. $H[i, t]$ is `true` if and only if the head is scanning tape cell i at time t. Observe: $1 \leq i \leq p(n)$ and $0 \leq t \leq p(n)$.

   Notice $H[i, t]$ defines $\mathcal{O}(p^2(n))$ boolean variables.

To simplify notation, we introduce a utility boolean expression which asserts that exactly one of the boolean parameters $x_1, x_2, ..., x_r$ is true. Define:

$$U(x_1, x_2, ..., x_r) = (x_1 \vee x_2 \vee ... \vee x_r) \wedge \left(\bigwedge_{\substack{i,j \\ i \neq j}} (\neg x_i \vee \neg x_j) \right)$$

We are now ready to express each of the conditions 1 through 7 in terms of the boolean variables $C[i, j, t]$, $S[k, t]$, and $H[i, t]$.

1. Expression A asserts that at each point in time, machine N is scanning exactly one tape cell. Define:

   $$A_t = U(\, H[1, t], H[2, t], ..., H[p(n), t]\,)$$

   and

   $$A = A_0 \wedge A_1 \wedge A_2 \wedge ... \wedge A_{p(n)}$$

   Notice the length of the boolean expression A is $\mathcal{O}(p^3(n))$ and can be written down in that time.

2. Expression B asserts that at each point in time, each tape cell contains exactly one symbol. Define:

   $$B_{i,t} = U(\, C[i, 1, t], C[i, 2, t], ..., C[i, m, t]\,)$$

   where $1 \leq i \leq p(n)$ and $0 \leq t \leq p(n)$. Further, we combine the $B_{i,t}$ to define B.

   $$B = \bigwedge_{i,t} B_{i,t}$$

   Notice the length of the boolean expression B is $\mathcal{O}(p^2(n))$

3. Expression C asserts that at each point in time, N is in exactly one state.

   $$C = \bigwedge_{0 \leq t \leq p(n)} U(S(1, t), S(2, t), ..., S(s, t))$$

   Notice the length of the boolean expression C is $\mathcal{O}(p(n))$.

4. Expression D asserts that the contents of at most one tape call is modified at any point in time t.

$$D = \bigwedge_{i,j,t} [\ \ (C[i,j,t] \equiv C[i,j,t+1])\ \vee H[i,t]\ \ ]$$

Here, the symbol $\equiv$ denotes "if and only if". In terms of boolean operations, $x \equiv y$ can be expressed as $(x \wedge y) \vee (\neg x \wedge \neg y)$.

Notice the length of the boolean expression D is $\mathcal{O}(p^2(n))$.

5. Expression E asserts each successive ID in the computation history follows from the previous ID by one operation allowed by the transition function δ. This is the most complicated of the boolean expressions in this proof.

Define $E_{i,j,k,t}$ to assert one of the following:

- that the i^{th} cell does not contain symbol j at time t,
- that the tape head is not scanning cell i at time t,
- that N is not in state k at time t, or
- that the next ID of N is obtained from the previous ID by a transition allowed by δ.

$$E_{i,j,k,t} = \neg C[i,j,t] \vee \neg H[i,t] \vee \neg S[k,t] \vee \bigvee_{\ell} (\ C[i,j_\ell,t+1] \wedge S[k_\ell,t+1] \wedge H[i_\ell,t+1]\ )$$

Here, ℓ varies over all possible moves when machine N is scanning symbol j in state k.

We now define E as:

$$E = \bigwedge_{i,j,k,t} E_{i,j,k,t}$$

Notice the length of the boolean expression E is $\mathcal{O}(p^2(n))$.

6. F asserts that the first configuration is an initial configuration.

$$F = S[1,0] \wedge H[1,0]\ \wedge \bigwedge_{1 \le i \le n} C[i,j_i,0]\ \wedge \bigwedge_{n < i \le p(n)} C[i,1,0]$$

$S[1,0]$ asserts that at time $t = 0$, machine N is in state q_1, which we take to be the start state. $H[1,0]$ asserts that at time $t = 0$, N is scanning the leftmost tape cell. The remaining two factors assert that the first n tape cells contain the input string, and the remaining tape cells are blank (where symbol X_1 is the blank).

Notice the length of the boolean expression F is $\mathcal{O}(p(n))$.

7. G asserts that N enters the accepting state. Let $\hat{q}$ denote the accepting state. Then,

$$G = S(\ \hat{q}, p(n)\ )$$

Notice the length of the boolean expression F is $\mathcal{O}(1)$.

Finally,

$$w_0 = A \wedge B \wedge C \wedge D \wedge E \wedge F \wedge G$$

The boolean expression w_0 is satisfiable if and only if input string w is in $L(N)$.