Lecture 20

**SPACE** \((t(n)) = \{ L \mid L \in L(M), M \text{ takes } t(n) \text{ space to decide on length } n \} \)

**NSPACE** \((t(n)) = \{ L \mid L \in L(M), M \text{ is an NTM takes } t(n) \text{ space} \}

**SAVITCH's thm**: For \(t(n) \geq n\), \(\text{NSPACE}(t(n)) \subseteq \text{SPACE}(t^2(n))\)

**Lemma 1**: \(\text{TIME}(t(n)) \subseteq \text{SPACE}(t(n))\)

**Lemma 2**: \(\text{SPACE}(t(n)) \subseteq \text{TIME}(2^{O(t(n))})\)

**Pf of Savitch's thm**: Let us be given \(N\), in NTM needing \(t(n)\) space to decide \(L\).

**WLOG**, let \(N\) have the unique accepting config. \(\# \text{accept} \# \) (also on input \(w = w_1 \ldots w_n\)). Unique starting config \(\#s_{\text{start}}w_1 \ldots w_n \# \) \(t(n)\)

**WANT**: deterministic procedure that, on input \(C_1, C_2\), time \(T\) determine if \(N\) can get from \(C_1\) to \(C_2\) in \(T\) steps.

\[ \text{CANYIELD}(C_1, C_2, 2^c) : \]

- if \(c = 0\), check if the transition function of \(N\) allows it
- else
  - for every possible \(C\) of the form \(C_{\text{mid}}, 2^{-c}\)
    - if \(\text{CANYIELD}(C_1, C_{\text{mid}}, 2^{-c-1})\)
    - and \(\text{CANYIELD}(C_{\text{mid}}, C_2, 2^{-c-1})\)

  - reject

**Analysis**

- Correctness: induction on \(c\), if \(c = 0\), base case correct.
  - Inductive step also correct if possible if inductive step also exists.

**Space complexity**: \(S(i)\)

- if \(\text{S}(0) \leq t(n)\) for the lookup,
- for \(i \geq 1\)

\[ S(i) \leq \text{constant} \cdot t(n) + S(i-1) = i \cdot \text{constant} \cdot t(n) \]

To determine if \(N\) accepts \(w\), run \(\text{CANYIELD}(\text{start}, \text{accept}, 2^{O(t(n))})\)

Need space \(S(O(t(n))) = O(t^2(n))\)

\(\checkmark\)
\[ \text{PSPACE} = \bigcup_{k=0}^{\infty} \text{SPACE}(n^k) \]

\[ \text{NPSPACE} = \bigcup_{k=0}^{\infty} \text{NPSPACE}(n^k) \]

By Savitch's thm, \( \text{PSPACE} = \text{NPSPACE} \)

\[ \text{SATEPSPACE (last class)} \]
\[ \text{NP} \subseteq \text{PSPACE} \]

One of these containments is proper: 
\[ P \neq \text{EXPTIME} \]

But \[ P \supseteq \text{NP} \supseteq \text{PSPACE} = \text{EXPTIME} \]

True quantified boolean formula (TQBF)

\[ \forall x \exists y \left[ (x \land y) \land (\bar{x} \land \bar{y}) \right] \]

This is TQBF bc set \( y = \bar{x} \) to make it true.

\[ \exists x \forall y \left[ (x \land y) \land (\bar{x} \land \bar{y}) \right] \]

False if \( x = T, y = T \) makes statement false, same for \( x = F, y = F \)

\[ \psi = Q_1 x_1, Q_2 x_2, \ldots, Q_n x_n \phi(x_1, \ldots, x_n) \]

Unquantified boolean formula.

True if \( \phi(x_1, \ldots, x_n) = T \) whenever \( x_1, \ldots, x_n \) defined according to quantifiers

if \( Q_1 = \exists \) \( \psi \) is true if \( Q_2 x_2, \ldots, Q_n x_n \phi(T, x_2, \ldots, x_n) \)

or \( Q_2 x_2, \ldots, Q_n x_n \phi(F, x_2, \ldots, x_n) \)

if \( Q_2 = \forall \) \( \psi \) is true if \( Q_2 x_2, \ldots, Q_n x_n \phi(T, x_2, \ldots, x_n) \)

and \( Q_1 x_1, \ldots, Q_n x_n \phi(F, x_2, \ldots, x_n) \)
$$\text{TQBF} = \{ \phi \mid \forall \phi \text{ is a true } \text{qbf} \}$$

**Thm**  \( \text{TQBF} \in \text{PSPACE} \)

\( \text{TQBF} \) is \( \text{PSPACE} \)-complete

**Thm**  \( \forall \phi \in \text{PSPACE} \), \( \phi \in \text{PTQBF} \)

(i.e. TQBF is \( \text{PSPACE} \)-hard)

**How to decide TQBF using poly space?**

**TQBF decider:** On input \( \phi \)

\[ \phi = Q_1 x_1 Q_2 x_2 ... Q_n x_n \phi_0 (x_1, ..., x_n) \]

If \( Q_1 = \exists \)

let \( \phi_T = Q_2 x_2 ... Q_n x_n \phi_0 (x_1, ..., x_n) \)

let \( \phi_F = Q_2 x_2 ... Q_n x_n \phi_0 (F_1, ..., x_n) \)

accept if \( \text{TQBF decider} (\phi_T) \) or \( \text{TQBF decider} (\phi_F) \) accepts

else \( Q = \forall \)

accept if \( \text{TQBF decider} (\phi_T) \) and \( \text{TQBF decider} (\phi_F) \) accepts

In polynomial space, if \( \phi_T \) and \( \phi_F \) is not defined until it has to be run, and each recursive call reuses the same space.

**Analysis**

**Correctness:** \( \checkmark \)

**Space complexity:** \( n \) is \# of variables, \( m \) is size of \( \phi \)

\[ S(n, m) = O(m) \]

\[ S(n, m) = O(n^2 + m) \]

\[ n \cdot O(n^2 + m) = O(n^2 + m) \]
Assuming TQBF is P-space complete.

Formula game:

2 players, Alice and Ernie.

Given a formula:

\[ \exists x_1 \forall y_1 \exists x_2 \forall y_2 \ldots \exists x_n \forall y_n \phi(x_1, \ldots, x_n, y_1, \ldots, y_n) \]

Ernie's job is to set \( x \)'s and make \( \phi \) true.

Alice's job is to set \( y \)'s and make \( \phi \) false.

Formula Game - \( \{<x,y> | \text{Ernie has a winning strat} \} \)