LBA - CSL

$\alpha A\beta \rightarrow \alpha X\beta$

$\alpha, q_1, q_2, q_3, q, a_4, a_5, a_6$

$M$ is an LBA w/ $q$ states $\gamma$ symbols and input is length $n$

distinct config. = $q \cdot n \cdot \gamma^n$
Computational Histories
- sequence of configurations

Accepting Comp. History
$C_1, C_2, C_3, \ldots C_l$

a) $C_i$: initial configuration
b) $C_l$: accepting configuration
c) each $C_i$ follows from $C_{i-1}$ according to transitions of $M$. 
\[ A_{LBA} = \{ <m, w> \mid M \text{ is a LBA that accepts string } w \} \]

\[ A_{LBA} \text{ is decidable } \]

\[ L = \text{on input } <m, w> \]
   1. run \( M \) on \( w \) for \( \geq n^3 \) steps, or until \( M \) halts.
   2. If \( M \) accepts, accept
   Else reject (loops forever)
$$E_{LBA} = \{ <M> | \text{M is an LBA and } L(M) = \emptyset \}$$

is not decidable

Proof (reduction from ATM)

Suppose \( E_{LBA} \) is decidable

decider R, ATM

Construct B such that \( L(B) \neq \emptyset \)
iff M accepts w.

LBA B recognizes all accepting computational histories of M on input w.

\( B \) on input \( \#C_1\#C_2\#C_3\# \ldots \#C_q\# \)
where \( C_i \) is a configuration of M

1. Check if \( C_q \) is the initial configuration of M on w.
   \( e.g., \quad w, \overline{w}, w, \overline{w}, \ldots, \overline{w} \)
2. Check if \( C_q \) is an accepting config. of M on w.
3. For each pair \( C_i, \#C_{i+1} \) check that \( C_{i+1} \) follows from \( C_i \) given M.

\( w, \overline{w}, \ldots, w_k, \overline{w}, \ldots \)

if \( L(B) \) is empty, then there are no accepting comp. histories for M on w.
if \( L(B) \) is not empty, then there is some accepting comp. histoy.

\( S = \) on input \( <M, w> \)

1. Construct LBA B as described.
2. Run R on \( <B, w> \).
3. If R rejects, accept.
   If R accepts, reject.