

18.404/6.840 Recitation 9

Abbas Zeitoun

November 6, 2020

A_{LBA} is PSPACE-Complete

Recall that $A_{\text{LBA}} = \{\langle M, w \rangle \mid M \text{ is an LBA that accepts string } w\}$. Show that A_{LBA} is PSPACE-complete.

Solution: To show that A_{LBA} is PSPACE-complete, we need to show that:

- 1) $A_{\text{LBA}} \in \text{PSPACE}$.
- 2) $\forall B \in \text{PSPACE}, B \leq_p A_{\text{LBA}}$.

1) A TM T can recognize A_{LBA} by simulating M on w and returning the same result. To simulate M , T only needs to keep track of its tape contents in addition to the position of the head of M and the state that M is in. All of these only require linear space since an LBA only has a linear number of tape cells to work with by definition.

T also needs to make sure that M doesn't loop on w , so it can maintain a counter and reject if M doesn't halt after computing for more steps than the number of configurations of M on w . The maximum number of configurations of M on w is $|Q| \times m \times |\Gamma|^m$, where Q is the set of M 's states, Γ is M 's tape alphabet, and m is the length of w . Since $m \leq n$, where n is the size of the input string $\langle M, w \rangle$, then the number of configurations is $O(n \times d^n)$ for some constant d . Representing that in bits requires $O(n)$ space, so the total space required by T is $O(n)$. So $A_{\text{LBA}} \in \text{PSPACE}$.

2) We can show this by reducing from TQBF. Note that to decide TQBF, a TM can follow the procedure described in lecture 17 to recursively assign truth values to the variables (in the order that they appear in the quantifiers) and check if the resulting boolean formula evaluates to true for each such assignment. Doing so only requires extra space for a stack that stores the truth values of the variables, so the extra space used is $O(n)$.

If an LBA had extra space at the end of its input string, it would be able to implement the same algorithm without any changes. So given an instance ϕ of TQBF, we can map ϕ to an A_{LBA} instance $\langle M, w \rangle$, where:

- $w = \langle \phi \rangle \# _{}^n$ ($\langle \phi \rangle$ padded with n blank symbols, where n is the length of $\langle \phi \rangle$)
- M is an LBA that first checks if the input string is of the form $\langle \phi \rangle \# _{}^n$ and then uses the extra blank cells at the end of the input string to check if ϕ is satisfiable (following the procedure described above).

It is straightforward to see that M only accepts w if ϕ is satisfiable, and the reduction only requires polynomial time, so $\text{TQBF} \leq_p A_{\text{LBA}}$.