



planetmath.org

Math for the people, by the people.

linear bounded automaton

Canonical name	LinearBoundedAutomaton
Date of creation	2013-03-22 18:57:40
Last modified on	2013-03-22 18:57:40
Owner	CWoo (3771)
Last modified by	CWoo (3771)
Numerical id	10
Author	CWoo (3771)
Entry type	Definition
Classification	msc 68Q45
Synonym	LBA
Synonym	NLBA
Synonym	DLBA
Synonym	deterministic LBA
Related topic	ContextSensitiveGrammar
Related topic	ContextSensitiveLanguage
Defines	deterministic linear bounded automaton

A *linear bounded automaton*, or LBA for short, is a restricted form of a non-deterministic Turing machine with a single tape and a single tape head, such that, given an input word on the tape, the tape head can only scan and rewrite symbols on the cells occupied by the initial input word.

Formally, a *linear bounded automaton* is a 9-tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F, \prec, \succ)$, where

1. Q is the state alphabet
2. Σ is the input alphabet
3. Γ is the tape alphabet and $\Sigma \subset \Gamma$
4. $\prec, \succ \in \Sigma$ are the left and right end markers.
5. δ is a function from $Q \times \Gamma$ to $P(Q \times \Gamma \times \{L, R\})$, called the *next move function*
6. $q_0 \in Q$ is the start state
7. $B \in \Gamma - \Sigma$ is the blank symbol
8. $F \subset Q$ is the set of final states

The interpretation of $(q, b, D) \in \delta(p, a)$ is the same as that of a non-deterministic TM: if the M is in state p and the tape head is reading a tape cell containing a , then it replaces a by b in that tape cell, and move in the direction of D ($D = L$ for left, and $D = R$ for right), and changes its state to q . In addition, the next move function δ has the following properties:

- if $(q, b, D) \in \delta(p, \prec)$, then $b = \prec$ and $D = R$, and
- if $(q, b, D) \in \delta(p, \succ)$, then $b = \succ$ and $D = L$.
- if $(q, b, D) \in \delta(p, a)$ and $a \notin \{\prec, \succ\}$, then $b \notin \{\prec, \succ\}$.

In other words, if the tape head is reading the left end marker \prec , then if it has a next move, it can only move right without replacing the end marker \prec . Similarly, if it is reading the right end marker \succ , then it can only move left if it has any next move at all, without replacing \succ . Furthermore, no symbol can be replaced by an end marker unless the symbol itself is the end marker.

An LBA is also known as a non-deterministic LBA, or NLBA for short. An LBA is said to be *deterministic* (abbreviated as DLBA) if $\delta(p, a)$ is at most a singleton for every pair $(p, a) \in Q \times \Gamma$.

A configuration of M is a triple (p, u, i) , where $p \in Q$ is the current state of M , $u \in \{\prec\}\Gamma^*\{\succ\}$ is the content of the tape (including the end markers), and i a non-negative integer, the position of the tape head, where $i = 0$ is the position of \prec , the left end marker.

Define a binary relation \Rightarrow on the set of all configurations of M : for $a_i, b \in \Gamma$,

$$(p, a_0 \cdots a_{i-1} a a_{i+1} \cdots a_{n+1}, i) \Rightarrow (q, a_0 \cdots a_{i-1} b a_{i+1} \cdots a_{n+1}, j),$$

where $i = 1, \dots, n$ and n a non-negative integer, iff any one of the following holds:

$$(q, b, L) \in \delta(p, a_i) \text{ and } j = i - 1 \quad \text{or} \quad (q, b, R) \in \delta(p, a_i) \text{ and } j = i + 1.$$

Notice that $a_0 = \prec$ and $a_{n+1} = \succ$. If $i = 0$, then the former case can not happen, and if $i = n + 1$, then the latter case can not happen.

Take the reflexive transitive closure \Rightarrow^* of \Rightarrow , and set

$$L(M) := \{u \in \Sigma^* \mid (q_0, \prec u \succ, 0) \Rightarrow^* (q, \prec v \succ, j) \text{ for some } q \in F\}$$

the language accepted by M .

Remarks

- For a given input word u , the workspace $w(u)$ of an LBA is defined as the portion of the tape between the end markers (including the end markers). In the definition above, we see that $|w(u)| = |u| + 2$, where $|\cdot|$ is the length function. One can enlarge the workspace so that $|w(u)| = r|u| + s$, where r, s are positive integers with $r \geq 1$ and $s \geq 2$. This is the reason for the name “linear bounded” in LBA. However, the computational power of an LBA with enlarged workspace is not increased: the language accepted by such an LBA can be accepted by an LBA defined above.
- A language is context-sensitive iff it can be accepted by an LBA.
- Every context-free language can be accepted by a DLBA, but not conversely.
- It is an open question whether every context-sensitive language can be accepted by a DLBA.

References

- [1] J.E. Hopcroft, J.D. Ullman, *Formal Languages and Their Relation to Automata*, Addison-Wesley, (1969).