Chapter 6

Transition Graphs

6.1 Introduction

Each FA has the following properties (among others):

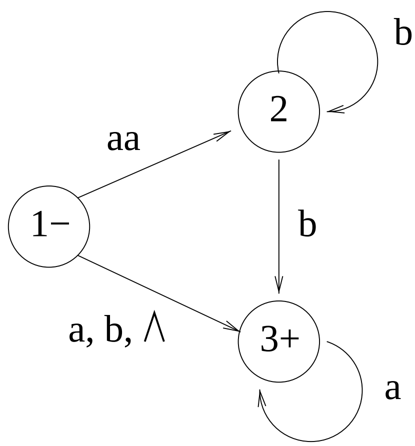
- For each state x and each letter $\ell \in \Sigma$, there is exactly one arc leaving x labeled with ℓ.
- Can only read one letter at a time when traversing an arc.
- Exactly one start state.

Now we want a different kind of machine that relaxes the above requirements:

- For each state x and each letter $\ell \in \Sigma$, we do *not* require that there is exactly one arc leaving x labeled with ℓ.
- Able to read any number of letters at a time when traversing an arc. Specifically, each arc is now labeled with a string $s \in \Sigma^*$, so the string s might be Λ or it might be a single letter $\ell \in \Sigma$.
- If an arc is labeled with Λ, we traverse the arc without reading any letters from the input string.
- If an arc is labeled with a non-empty string $s \in \Sigma^*$, we can traverse the arc if and only if the next unread letter(s) from the original input string are the string s.

- Suppose that we are in a state and we cannot leave the state because there is no arc leaving the state labeled with a string that corresponds to the next unread letters from the input string. Then if there are still more unread letters from the original input string, the machine *crashes*.
- There may be more than one way to process a string on the machine, and so the machine may be *nondeterministic*.
- If there is at least one way of processing the string on the machine such that it ends in a final state with no unread letters left and without crashing, then the string is accepted; otherwise, the string is rejected.
- There can be more than one start state.

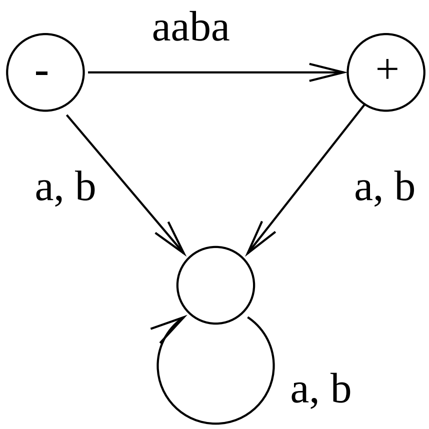
Example: Consider the following machine that processes strings over the alphabet $\Sigma = \{a, b\}$:



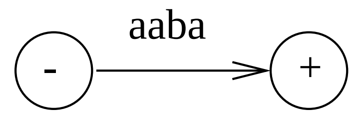
Note that this machine is not a finite automaton:

- The arc from state 1 to state 2 is labeled with the string aa, which is not a single letter.
- There are two arcs leaving state 2 labeled with b.
- There is no arc leaving state 2 labeled with a.
- There is an arc from state 1 to state 3 labeled with Λ, which is not a letter from Σ.
- There is no arc leaving state 3 labeled with b.

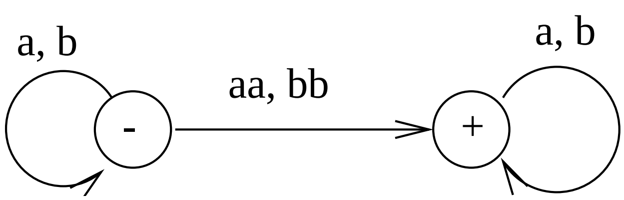
Example: Only accepts the word *aaba*



or



Example: Accepts all words that contain a doubled letter.



- Note that we must decide how many letters to read from the input string each time we go back for more.
- Depending on how we process the string *abb*, the machine may or may not accept it.
- Thus, we say that a string is *accepted* by a machine if there is some way (called a successful path) to process all of the letters in the string and end in a final state without having crashed.
- If there is no way to do this, then the string is not accepted.
- For example, consider the string *baba*, which is not accepted.

6.2 Definition of Transition Graph

Definition: A *transition graph* (TG) is a collection $M = (K, \Sigma, \Pi, S, F)$ where:

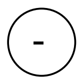
1. K is a finite set of states.
   - $S \subset K$ is a set of start states with $S \neq \emptyset$ (but possibly with more than one state), where each start state is designated pictorially by $\ominus$.
   - $F \subset K$ is a set of final states (possibly empty, possibly all of K), where each final state is designated pictorially by $\oplus$.
2. An alphabet Σ of possible input letters from which input strings are formed.
3. $\Pi \subset K \times \Sigma^* \times K$ is a finite set of transitions, where each transition (arc) from one state to another state is labeled with a string $s \in \Sigma^*$.
   - If an arc is labeled with Λ, we traverse the arc without reading any letters from the input string.
   - If an arc is labeled with an non-empty string $s \in \Sigma^*$, we can traverse the arc if and only if the next unread letter(s) from the original input string are the string s.
   - We allow for the possibility that for any state $x \in K$ and any string $s \in \Sigma^*$, there is more than one arc leaving x labeled with string s.
   - Also, we allow for the possibility that for any state $x \in K$ and any letter $\ell \in \Sigma$, there is no arc leaving state x labeled with ℓ.

Remarks:

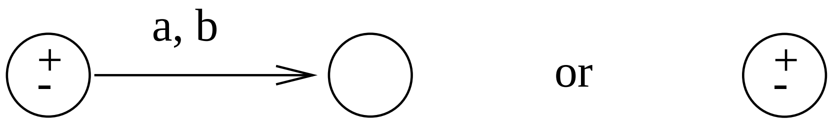
- when an edge is labeled with Λ, we can take that edge without consuming any letters from the input string.
- We can have more than one start state.
- Note that every FA is also a TG.
- However, not every TG is an FA.

6.3 Examples of Transition Graphs

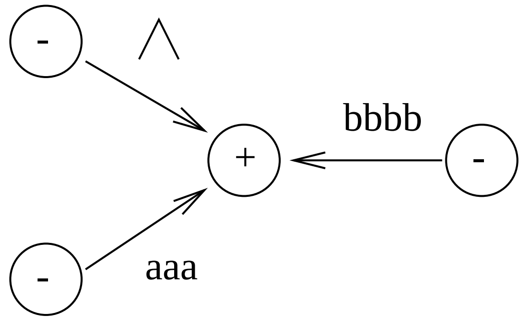
Example: this TG accepts nothing, not even Λ.



Example: this TG accepts only the string Λ.

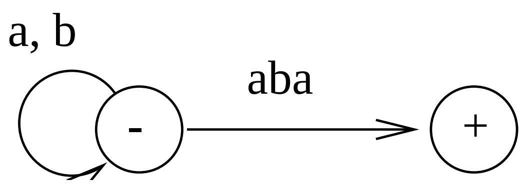


Example: This TG accepts only the words Λ, *aaa* and *bbbb*.

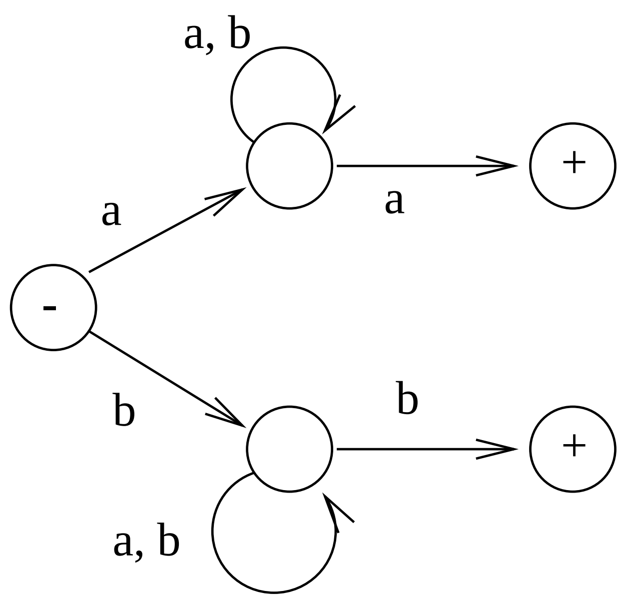


Example: this TG accepts only words that end in *aba*; i.e., the language generated by the regular expression

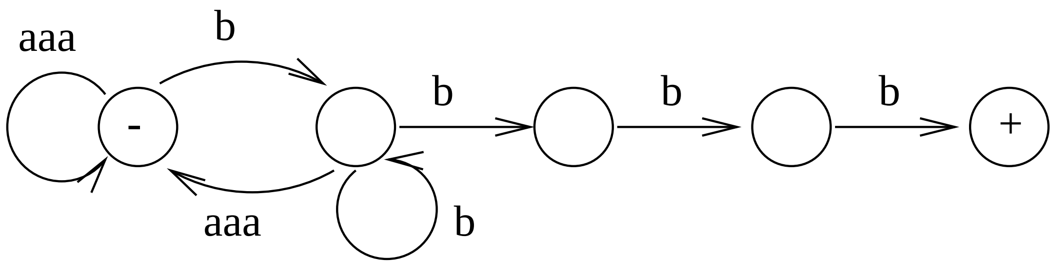
$$(\mathbf{a}+\mathbf{b})^*\mathbf{aba}$$



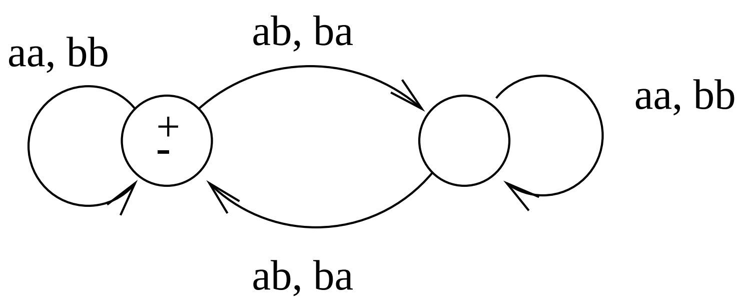
Example: this TG accepts the language of all words that begin and end with the same letter and have at least two letters.



Example: this TG accepts the language of all words in which the a's occur in clumps of three and that end in four or more b's.



Example: this is the TG for EVEN-EVEN



Example: Is the word *baaabab* accepted by this machine?