Finite State Automata and Automatic Groups

Jake Adicoff, Kyle Stanley, and Kevin Chen

May 17th, 2017

Table of contents

Introduction

Why study Finite State Automata?

- A basic circuit that allows for remarkably powerful computing
- FSA end structure to Automatic Groups

Why Study Automatic Groups?

- 3 Manifold groups
- Solvable word problem
- Quadratic Dehn Function
- Normal Form

Finite State Automata

Definition of Finite State Automata

Definition (Finite State Automaton)

A FSA A consists of five objects:

(1) a set I, called the input alphabet of input symbols;

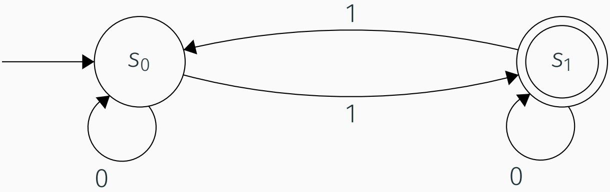
(2) a set S of states the automaton can be in;

(3) a designated state s_0, called the initial state;

(4) a designated set of states called the set of accepted states;

(5) a next-state function $N : S \times I \to S$ that associates a ”next-state” to each ordered pair consisting of a ”current state” and ”current input.”

Next-State Diagram and Table

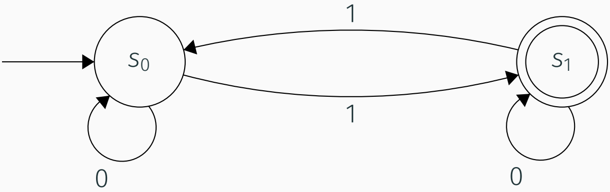


State	0	1
s_0 (*initial*)	s_0	s_1
s_1 (*accepting*)	s_1	s_0

The Language Accepted by a FSA

Definition (Language)

The language accepted by an automaton A is the set of all words $w \in I^*$ corresponding to direct paths that begin at an initial state and end at an accept state of A. The language is denoted $L(A)$.

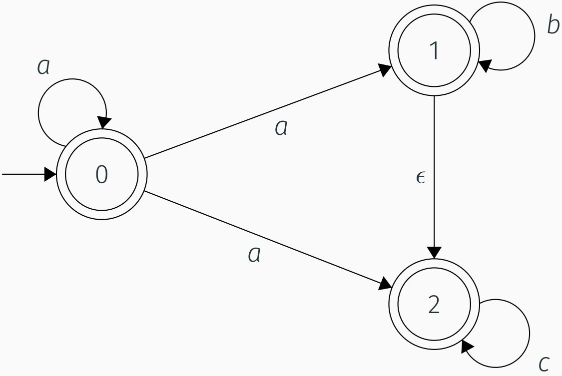


Regular Languages

Definition (Non-Deterministic Automaton)

A non-deterministic automaton is an automaton where some edges are labelled by a new letter, ϵ, which is not in the original alphabet, I, of the automaton. The language of a such an automaton is the set of all words corresponding to paths that begin at a start state and end at an accept state, with all ϵ edges removed.

Example



This FSA is non-deterministic because it has three *a* edges leaving state 0 and an ϵ edge from state 1 to state 2.

Deterministic Automaton

Definition (Deterministic Automaton)

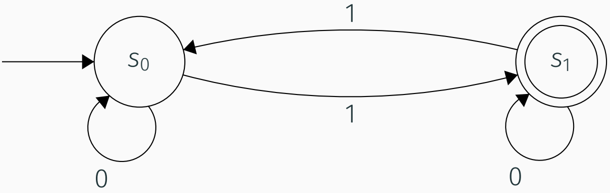
A deterministic automaton is a finite-state automaton with the following additional requirements:

(1) There is exactly one start state.

(2) No two edges leaving a vertex have the same label.

A deterministic automaton is complete if for each vertex v, and each letter a in the alphabet I, there is an edge leaving v labelled a.

Example



This FSA is deterministic because it has only 1 start state, no ϵ edges, and no states with two or more outbound edges with the same label.

Definition of Regular Language

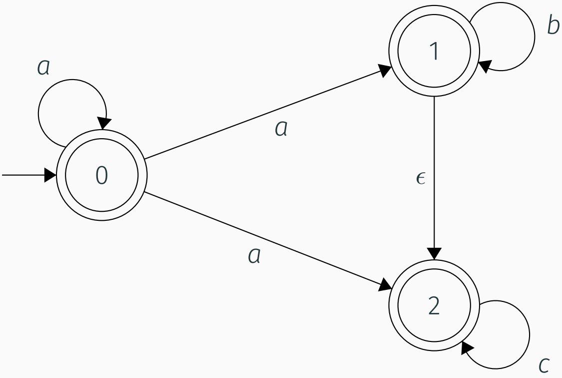
Definition (Regular Language)
A regular language is any language that is accepted by a deterministic automaton.

Theorem 1

Theorem

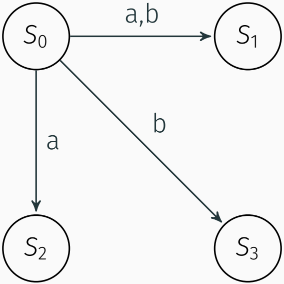
The set of languages accepted by non-deterministic FSAs is the same as the set of regular languages.

Example



Remove ϵ edges by making new automaton $\mathcal{M}_\epsilon$ with states equal to ϵ-closed subsets S of $V(\mathcal{M})$.

Example



Remove duplicate edge labels by creating a new graph $\mathcal{D}$ with states equal to up to the power set of the original states! (Takes n states to a maximum of $2^n - 1$ states)

Theorem 2

Theorem

Let K and L be regular languages with a common input alphabet I. Then the following languages are also regular languages.

(1) The complementary language $S^* - K$

(2) The union $K \cup L$

(3) The intersection $K \cap L$

(4) The concatenation KL consisting of all words of the form

$$KL = \{w_K w_L \mid w_K \in K, w_L \in L\}$$

(5) $K^* = K \cup KK \cup KKK \cup \ldots$

Lemma (Pumping Lemma)

Let L be a regular language. Then there is an integer $n \geq 1$ such that any word $x \in L$ of length greater than n can be expressed as $x = uvw$ where v is a non-empty word, and

(1) $|u| \leq n$

(2) $uv^iw \in L \ \forall i \geq 0$

Two-Variable Padded Language

Definition (Two-Variable Padded Language)

Given an alphabet A, we can add padding symbol $\$ \notin A$ to form the alphabet $A \cup \{\$\}$, and we can consider a FSA M, but this time with labels in $(A \cup \$) \times (A \cup \$) \setminus (\$, \$)$. Given a pair of words $(u, v) \in A^* \times A^*$, say $u = u_1 u_2 ... u_n$, $v = v_1 v_2 ... v_m$ with $m \leq n$, we pad v with the symbol $\$$ so that the resulting words have equal length. We will say (u, v) is accepted by M if we can read off the edges $(u_1, v_1), ..., (u_m, v_m), (u_{m+1}, \$), ..., (u_n, \$)$ and end up at an accept state of M. The set of accepted pairs (u, v) is said to be regular over the padded alphabet A.

Automatic Groups

Definition of Automatic Group

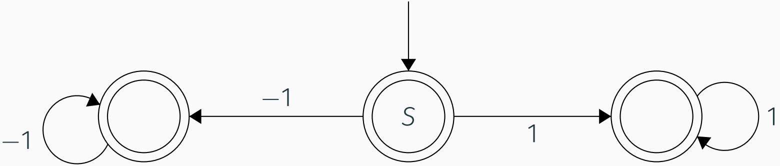
Definition (Automatic Group)
An group is automatic if it has two main properties:

1. It has a normal form that is accepted by some deterministic FSA. This FSA is called the *word acceptor*, M.
2. For each generator s, there is a deterministic FSA M_s (*word comparator*) which accepts pairs (g, gs) – that is, the machine can recognize when two normal form words differ by a particular generator. There is also a deterministic FSA $M_=$ (*equality checker*) which accepts pairs (g, g) - that is, $M_=$ can tell when two words are equivalent.

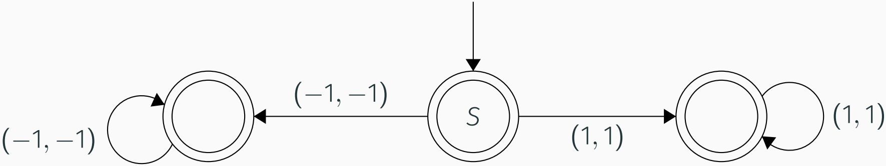
A Simple Example

- $\mathbb{Z}$ is an Automatic Group.
- We can show $\mathbb{Z}$ is automatic by constructing the word acceptor (M), the equality checker ($M_=$), and the two word comparators (M_1 and M_{-1}).

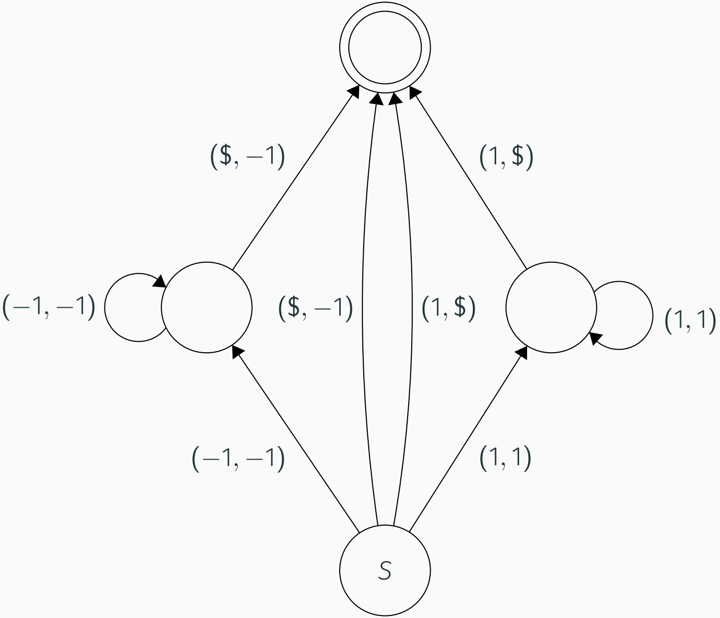
Word Acceptor M



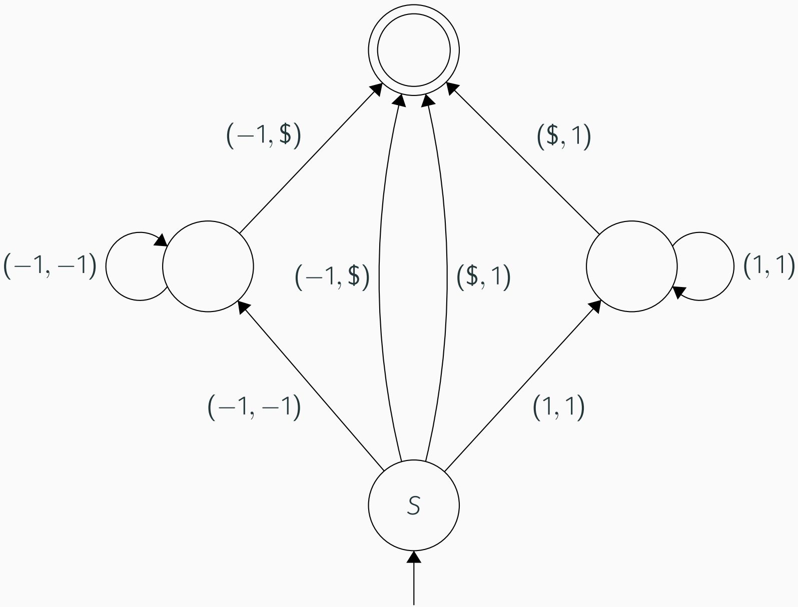
Equality Checker $M_=$



First Word Comparator M_1



Second Word Comparator M_{-1}



k-Fellow Traveler Property

Definition (k-fellow traveller property)

Let u be some path in $\Gamma_S(G)$ and define the map $u : [0, \infty) \to \Gamma_S(G)$, where $u(t)$ is the element of G given by the first t letters of u if t is less than the length of u, and $u(t) = \bar{u}$ if t is greater than or equal to the length of u (where $\bar{u}$ is the element of G represented by the word u in the free group on S). Two paths u and v in $\Gamma_S(G)$ are said to satisfy the k-fellow traveller property if $d_{\Gamma_S(G)}(\overline{u(t)}, \overline{v(t)}) \leq k$ for all $t \geq 0$.

Alternative Definition of Automatic Group

Theorem (Alternative Definition of Automatic Group)

A group G is automatic if and only if the following properties hold:

(1) G has a word acceptor M with regular language $L(M)$ over some finite monoid generating set A , as in condition 1 of the definition of automatic group. Recall that the natural map $L(M) \to G$ is required to be onto, and that the image of a word w is denoted by $\bar{w}$.

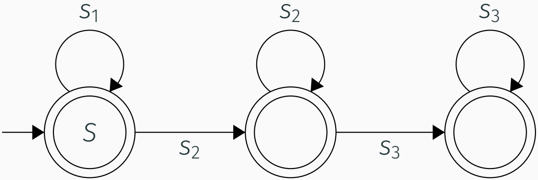
(2) There is a constant k such that if $u, v \in L(M)$ represent elements of G which are distance 1 apart in $\Gamma_A(G)$, then the paths u and v satisfy the k-fellow traveler property.

Examples of Automatic Groups

- Finite Groups
- The free group on two generators, F_2
- Finitely Generated Abelian Groups
- BS(1,2) - not automatic

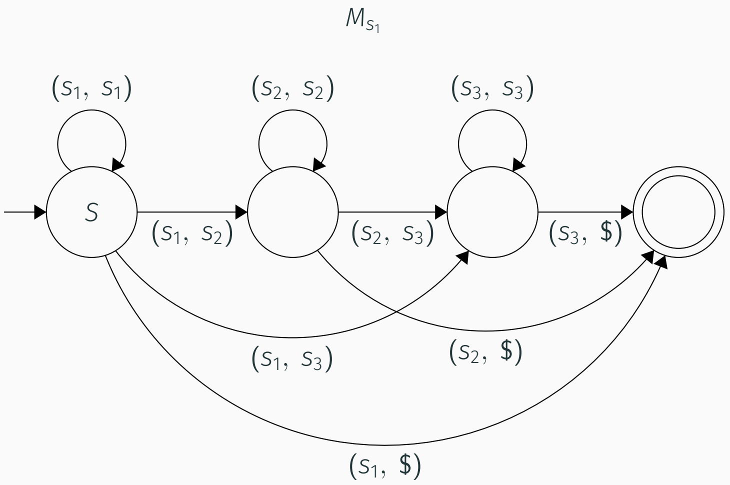
$S = \{s_1, s_2, s_3\}$

Normal form: $g = s_1^n s_2^m s_3^r$ for $n, m, r \in \mathbb{N} \cup \{0\}$



The equality checker $M_=$ is the same diagram with ordered pairs (s_i, s_i) replacing s_i.

Finitely Generated Abelian Groups (cont.)



References I

- *Discrete Mathematics With Applications.* **Cengage, 1990.**
- *Groups Graphs and Trees - An Introduction to the Geometry of Infinite Groups.* **London Mathematical Society Student Texts, 2008.**
- Benson Farb. **Automatic groups: A guided tour.** *L'Enseignement Mathematique*, 1992.