Groups, Formal Language Theory and Decidability

Sam Jones

Supervised by: Rick Thomas

Department of Computer Science, University of Leicester

August 2013

What is an algorithm?

Informally: A *finite* sequence of steps to follow in order to solve a problem.

A problem is said to be *decidable* if an algorithm solving it exists and is said to be *undecidable* if there does not exist an algorithm solving it.

Formal language theory basics

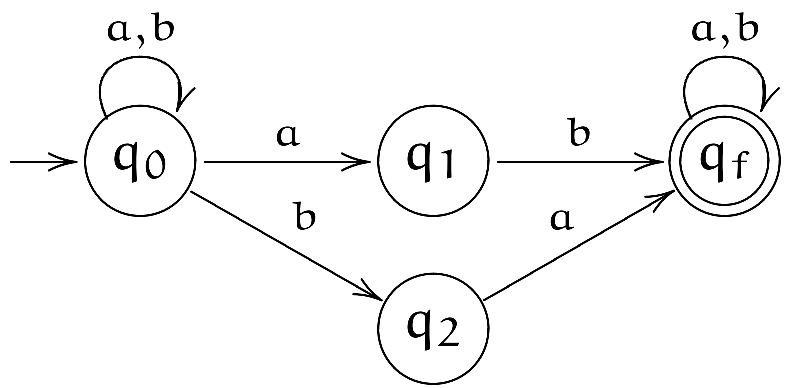
Given a finite alphabet (set of symbols) Σ, Σ^* is the set of all finite words consisting of symbols from Σ.

We call any subset L of Σ^* a *language*.

Finite automata

Finite automata have no memory (other than the states).

Finite automata accept a class of languages known as the *regular* languages.

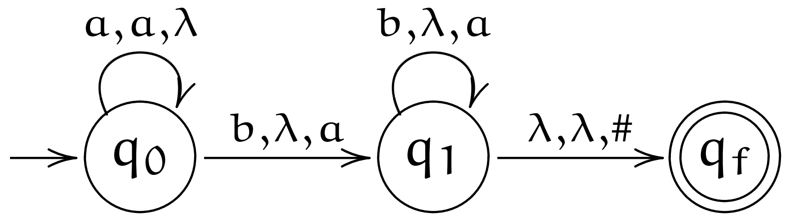


The language accepted by this automaton is the set of all finite words which contain the subword ab or the subword ba.

Pushdown automata

Finite automaton with an added memory device: a *stack*.

Pushdown automata accept a class of languages known as the *context-free* languages.



The language accepted by this pushdown automaton is the set of words of the form $a^n b^n$

One-counter automata

Pushdown automaton where the stack alphabet is restricted to one symbol (other than the bottom stack marker).

One-counter automata accept precisely the *one-counter* languages.

The word problem

Given a finite presentation $< X|R >$ for a group G, the word problem asks whether two words α and β over the alphabet $\Sigma = X \cup X^{-1}$ represent the same element of G.

The word problem as a formal language

$$\alpha = \beta \iff \alpha\beta^{-1} = 1 \text{ in } G.$$

Consider

the set $WP(X, G)$ of all words in Σ^* which represent the identity element of G.

The problem of determining whether two words are equal (in G) is now equivalent to determining membership of this language.

The word problem as a formal language

Does the word problem change if we change our choice of X?

It depends what we mean by this.

Inverse homomorphism to the rescue

If $\mathcal{F}$ is a class of languages closed under inverse homomorphism and $\mathcal{WP}(X, G) \in \mathcal{F}$ for some

finite generating set X then we have that $\mathcal{WP}(Y, G) \in \mathcal{F}$ for all finite

generating sets Y.

Classification of groups by their word problem

Group	*Language*
Finite	Regular
Virtually Cyclic	One-Counter
Virtually Cyclic	Deterministic One-Counter
Virtually Free	Context-Free
Virtually Free	Deterministic Context-Free

Are there any other groups here?

In some sense, no: Herbst proved that if your class of languages has certain closure properties and lies inside the context-free languages then you either get the finite groups, the one-counter groups or all of the context-free groups.

The word problem and decidability

Fix a class of languages $\mathcal{F}$. Is it decidable, given a language $L \in \mathcal{F}$, whether or not $L = \mathcal{WP}(X, G)$ for some group G?

Regular - yes

Context-Free - no

The word problem and decidability

Fix a class of languages $\mathcal{F}$. Is it decidable, given a language $L \in \mathcal{F}$, whether or not $L = WP(X, G)$ for some group G?

One-counter - no

Deterministic Context-Free - yes

Characterisation of word problems

$L \subseteq \Sigma^*$, $L = WP(X, G)$ for some group $G \iff$

1. for all $\alpha \in \Sigma^*$ there exists $\beta \in \Sigma^*$ such that $\alpha\beta \in L$

   AND

2. $\alpha u \beta \in L, u \in L \Rightarrow \alpha\beta \in L$

Decidability results

1	2	Language
yes	yes	Regular
no	no	One-Counter
yes	?	Deterministic Context-Free
no	no	Context-Free