Notes on Automata Theory

Caffeinemachine
khetan@math.tifr.res.in*

March 29, 2021

Contents

1 Monoids

A **monoid** is a pair $(M, \cdot)$, there $\cdot$ is a binary operation $\cdot : M \times M \to M$ such that

Associativity. $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for all $a, b, c \in M$.
Identity. There is $e \in M$ such that $e \cdot a = a \cdot e$ for all $a \in M$.

We often suppress the '$\cdot$' and simply write ab in place of ab when there is no confusion about the binary operation.

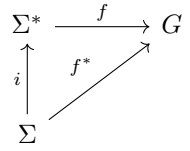
Example 1.1. Let X be any set and $M = \text{Maps}(X)$ denote the set of all the maps $X \to X$. Then $(M, \circ)$ is a monoid, where '$\circ$' is the composition operation. For $f \in \text{Maps}(X)$ and $x \in X$, we write xf to denote the image of x under f, instead of the more standard $f(x)$, since it more suitable for the purpose of studying automata. ♢

Example 1.2. Let X be a set and $\text{Rel}(X)$ denote the set of all the relations on X. For two relations R and S on X, we define the composition of R and S, which we write as $R \circ S$, or simply as RS, as follows: For $x, y \in X$, we have $xRSy$ if and only if there is $z \in X$ such that xRz and zSy. It is easy to see that $\text{Rel}(X)$ is a monoid under the composition operation. ♢

Example 1.3. Let Σ be any finite non-empty set. For a positive integer n, a **word** over Σ of length n is a function $f : \{1, \ldots, n\} \to \Sigma$, which we denote by $f(1) \cdots f(n)$. Given two words f and g of length m and n respectively over Σ, we define the **concatenation** of f and g, written fg, as $f(1) \cdots f(m)g(1) \cdots g(n)$. Let ε be a formal symbol, which we will refer to as the **empty word**. For any word f, we define $\varepsilon f = f\varepsilon = f$. Let Σ^* be the set of all the words over Σ along with ε. It can be easily checked that Σ^* is a monoid under concatenation. ♢

A **monoid homomorphism** is a map $h : G \to H$ between two monoids G and H such that h takes the identity of G to the identity of H, and h respects multiplication, that is $h(xy) = h(x)h(y)$ for all $x, y \in G$. Note that for a finite alphabet Σ, the monoid Σ^* is nothing but the "free monoid" generated by Σ: given Σ and given any monoid G along with a map $f : \Sigma \to G$, there is a unique monoid homomorphism $f^* : \Sigma^* \to G$ which makes the following diagram commute.

*For any questions, comments, suggestions, corrections, or typographical errors please write to me at this email address.



where $i : \Sigma \to \Sigma^*$ sends $\sigma \in \Sigma$ to $\sigma \in \Sigma^*$.

2 Quotient Monoids

Let M and M' be monoids and $f : M \to M'$ be a monoid homomorphism. Let $\sim$ be the equivalence relation on M induced by f. Write $M/\sim$ to denote the set of all all the equivalence classes of M under $\sim$, which are nothing by the fibers of f. Then we have a map $\bar{f} : M/\sim \to M'$ which sends $[x]$ to $f(x)$, making the following diagram commute.

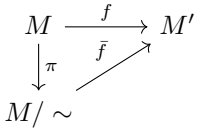


Figure 1: Factoring through quotient.

where $\pi : M \to M/\sim$ is the map which sends x to $[x]$ for all $x \in M$. In the above diagram M and M' are monoids while $M/\sim$ is, so far, just a set. It is thus natural to ask if one can endow $M/\sim$ with a monoid structure such that π and $\bar{f}$ become monoid homomorphisms, and also how many such monoid structures exist.

Suppose there is a monoid structure on $M/\sim$ such that π is a monoid homomorphism. Then we must have $[xy] = [x][y]$ for all $x, y \in M$. For this to be well-defined, we must have that whenever $y, y \in M$ are such that $y \sim y'$, then $xy \sim xy'$ for all $x \in M$, and whenever $x, x' \in M$ are such that $x \sim x'$, then $xy \sim x'y$ for all $y \in M$. This motivates the following definition.

Let M be a monoid and $\sim$ be an equivalence relation on M. We say that $\sim$ is a **left congruence** if $xy \sim xy'$ for all $x \in M$ whenever $y \sim y'$ for some $y, y' \in M$. We similarly define the notion of **right congruence**. We say that $\sim$ is a **bicongruence** relation if it is both left and right congruence. The following lemma is immediate.5

Lemma 2.1. *Let M be a monoid and $\sim$ be a bicongruence equivalence relation on M. Then $M/\sim$ has a unique monoid structure such that $\pi : M \to M/\sim$ is a monoid homomorphism.*

Note that the equivalence relation induced on M by a monoid homomorphism $f : M \to M'$ is a bicongruence equivalence relation and thus f factors through $M/\sim$ to induce a monoid homomorphism $\bar{f} : M/\sim \to M'$ making the digram in Figure 1 commute.

3 Automatons and Regular Languages

A **deterministic semiautomaton**, or DSA, is a triple $S = (Q, \Sigma, \mathcal{F})$, where Q is a finite set whose elements are called **states**, Σ is a finite set which we call the **alphabet**, and $\mathcal{F}$ is a collection $\{f_\sigma : \sigma \in \Sigma\}$ of maps $f_\sigma : Q \to Q$, one for each $\sigma \in \Sigma$. For $q \in Q$, the image of q under f_σ will be written as qf_σ rather than $f_\sigma(q)$.

Now consider the monoid homomorphism $h : \Sigma^* \to \mathrm{Maps}(Q)$ given by $\sigma \mapsto f_\sigma$. Let q_0 be a state in Q and w be a word in Σ^*. The **run** of the semiautomaton S on the word w starting at q_0 is defined as the state $q_0 h(w)$.

An **deterministic finite automaton** is a 5-tuple $A = (Q, \Sigma, \mathcal{F}, q_0, F)$, where (Q, Σ, q_0) is a DSA, q_0 is a distinguished state which we refer to as the **start state**, and F is a distinguished collection of states which we refer to as **final states**.

We say that a word $w \in \Sigma^*$ is **accepted** by the automaton A is its run on w starting at q_0 is in F. The set of all the words that are accepted by A is called the **language** of A. A subset L of Σ^* is called a **regular language**, or a **regular set**, if there is an automaton whose language is L.

Theorem 3.1. *Let Σ be a finite alphabet. Then a language $L \subseteq \Sigma^*$ is regular if and only if is the union of some of the equivalence classes of some finite bicongruence relation on Σ^*.*

Proof. Let L be a regular language. Thus there is an automaton $A = (Q, \Sigma, \mathcal{F}, q_0, F)$ whose language is L. Consider the homomorphism $h : \Sigma^* \to \text{Maps}(Q)$ given by $h(\sigma) = f_\sigma$ for each $\sigma \in \Sigma$. Let $\sim$ be the equivalence relation induced on Σ^* by the fibers of h. As noted earlier, $\sim$ is a bicongruence relation. Also, $\sim$ is finite since $\text{Maps}(S)$ is finite. Any two words in the same equivalence class thus have the same run starting at any state, and hence in particular starting at q_0. Therefore, if a words $w \in \Sigma^*$ is accepted by A, so are all the words in the equivalence class of w. Hence the set of all the words in L is the union of some of these equivalence classes.

Conversely, let $\sim$ be a finite bicongruence relation on Σ^* and L be the union of some of the equivalence classes. We want to show that L is regular. Let $Q = \Sigma^*/\sim$. For any $w \in \Sigma^*$, let $[w]$ denote its equivalence class. For each $\sigma \in \Sigma$ define a map $f_\sigma : Q \to Q$ by writing $[w]f_\sigma = [w\sigma]$ for all $w \in \Sigma^*$. This is well defined thanks to the bicongruence of $\sim$.[1] Let $q_0 = [\varepsilon]$, that is, q_0 is the equivalence class of the empty word. Lastly, define F as the set of all the equivalence classes whose union is L. We claim that the language of the automaton $A = (Q, \Sigma, \mathcal{F}, q_0, F)$ is same as L. To do this, note that for any $w \in \Sigma^*$ we have the run of A on w starting at q_0 is nothing but $[w]$. Now if $w \in \Sigma^*$ is accepted by A, then $[w] \in F$, which, by definition of F, implies that $w \in L$. By the same token, if $w \in L$ then $[w] \in F$ and hence w is accepted by A, showing that L is indeed the language of A, and we are done. ■

Theorem 3.2. *Let Σ and Δ be finite alphabets and let $h : \Sigma^* \to \Delta^*$ be a monoid homomorphism. Let L be a regular language over Δ. Then $h^{-1}(L)$ is a regular language over Σ.*

Proof. Let $\sim$ be a finite bicongruence equivalence relation on Δ^* such that L is the union of some of the equivalence classes of $\sim$. Let $\pi : \Delta^* \to \Delta^*/\sim$ be the natural projection map. Let $\cong$ be the equivalence relation induced on Σ^* by the fibers of the map $\pi \circ h$. Clearly, then, $h^{-1}(L)$ is the union of some of the equivalence classes of $\cong$. More precisely, if $S \subseteq \Delta^*/\sim$ is such that $\bigcup S = L$, then $h^{-1}(L) = \bigcup(\pi \circ h)^{-1}(S)$. ■

4 Non-Deterministic Finite Automaton

A **non-deterministic semiautomaton**, or NDSA, is a triple $(Q, \Sigma, \mathcal{R})$, where Q is a finite sets whose elements are called **states**, Σ is a finite set called the **alphabet**, and $\mathcal{R} = \{r_\sigma : \sigma \in \Sigma\}$ is a set of relations on Q indexed by Σ.

A **non-deterministic finite automaton**, or NFA, is a 5-tuple $(Q, \Sigma, \mathcal{R}, q_0, F)$, where $(Q, \Sigma, \mathcal{R})$ is a NDSA, q_0 is a distinguished state called the **start state**, and F is a distinguished collection of states whose elements are called **final states**. We say that an NFA $(Q, \Sigma, \mathcal{R}, q_0, F)$ accepts a word $w = \sigma_1 \cdots \sigma_k$ in Σ^* if there is $q \in F$ such that the composite relation $r_{\sigma_1} \cdots r_{\sigma_k}$ satisfies

$$q_0 \; r_{\sigma_1} \cdots r_{\sigma_k} \; q \tag{4.1}$$

Since functions are special types of relations, we see that any DFA is also an NFA, and hence every regular language is accepted by some NFA. Interestingly, this apparent increase in power is illusory.

Theorem 4.1. *A language L over a finite alphabet Σ is regular if and only if there is an NFA whose language is L.*

Proof. For the non-trivial direction we need to show that if L is the language of an NFA then L is regular. Let $(Q, \Sigma, \mathcal{R}, q_0, F)$ be a NFA whose language is L. Define a map $f : \Sigma \to \text{Rel}(Q)$ as $f(\sigma) = r_\sigma$, which gives a homomorphism $f^* : \Sigma^* \to \text{Rel}(Q)$. Let $\sim$ be the equivalence relation on Σ^* induced on Σ^* by f^*. Since f^* is a monoid homomorphism, we know that $\sim$ is a bicongruence equivalence relation. Further, since $\text{Rel}(Q)$ is finite, we know that $\sim$ is finite.

Now let $R \subseteq \text{Rel}(Q)$ be defined as the set

$$R = \{r \in \text{Rel}(Q) : \quad \text{there exists } q \in F \text{ such that } q_0 \sim q\} \tag{4.2}$$

Then note that L is nothing by $\bigcup_{r \in R}(f^*)^{-1}(r)$, and thus L is the union of finitely many equivalence classes of $\sim$. Applying Theorem 3.1 now leads to the conclusion that L is regular and we are done. ■

Theorem 4.2. *Let Σ and Δ be finite alphabets and let $h : \Sigma^* \to \Delta^*$ be a monoid homomorphism. Then $h(L)$ is regular whenever $L \subseteq \Sigma^*$ is regular.*

Proof. (Sketch). Let A be a (deterministic) automaton whose language is L. We construct a non-deterministic automaton out of A whose language will be $h(L)$. Let $\sigma \in \Sigma$ be arbitrary and $h(\sigma) = \delta_1 \cdots \delta_k$. Replace each edge labelled σ in the graph of A by a sequence of k edges and $k-1$ auxiliary states and label these sequence of edges

[1] We only need *right* congruence for well-definedness.

by $\delta_1, \ldots, \delta_k$ in sequence. Do this for each $\sigma \in \Sigma$. Keep the rest the same. The language resulting NFA has $h(L)$ as its language. ■

5 Epsilon-NFA

An ε-NFA is a 5-tuple $(Q, \Sigma, \mathcal{R}, q_0, F)$, where Q is a finite set whose elements are called **states**, Σ is a finite set which we refer to as the **alphabet**, $\mathcal{R}$ is a set $\{r_\sigma : \sigma \in \Sigma\} \cup \{r_\varepsilon\}$, where r_σ is a relation on Q for each $\sigma \in \Sigma$ and r_ε is a reflexive relation on Q, the elementq_0 is a distinguished state which we call the **start state**, and F is a distinguished collection of states whose elements are called **final states**.

Let $A = (Q, \Sigma, \mathcal{R}, q_0, F)$ be an ε-NFA. For any state q in Q we define the ε-**closure** of q as the set of all the states q' such that $q\ r_\varepsilon\ q'$. For each $\sigma \in \Sigma$ we define a relation r_σ^ε as follows. For two states q and q', we write $q\ r_\sigma^\varepsilon\ q'$ if and only if there is a state q'' in the ε-closure of q such that $q''\ r_\sigma\ q'$. This gives us an NFA $A' = (Q, \Sigma, \mathcal{R}', q_0, F)$, where $\mathcal{R}' = \{r_\sigma^\varepsilon : \sigma \in \Sigma\}$. We say that a word $w \in \Sigma^*$ is **accepted** by A if and only if it is accepted by A'.