equivalent automata

Canonical name	EquivalentAutomata
Date of creation	2013-03-22 18:03:21
Last modified on	2013-03-22 18:03:21
Owner	CWoo (3771)
Last modified by	CWoo (3771)
Numerical id	6
Author	CWoo (3771)
Entry type	Definition
Classification	msc 03D05
Classification	msc 68Q45

Two automata are said to be *equivalent* if they accept the same language. Explicitly, if $A_1 = (S_1, \Sigma_1, \delta_1, I_1, F_1)$ and $A_2 = (S_2, \Sigma_2, \delta_2, I_2, F_2)$ are two automata, then A_1 is equivalent to A_2 if $L(A_1) = L(A_2)$. We write $A_1 \sim A_2$ when they are equivalent. It is clear that $\sim$ is an equivalence relation on the class of automata.

First, note that if $A_1 \sim A_2$, then every symbol $\alpha \in \Sigma_1$ in a word $a \in L(A_1)$ is a symbol α in Σ_2. In other words, every symbol in a word accepted by A_1 (or A_2) belongs to $\Sigma := \Sigma_1 \cap \Sigma_2$. As a result, $L(A_1) = L(A_2) \subseteq \Sigma^*$. If B_i is an automaton obtained from A_i by replacing the alphabet Σ_i with Σ, where $i = 1, 2$, then $B_i \sim A_i$. This shows that we may, without loss of generality, assume outright, in the definition of equivalence of A_1 and A_2, that they have the same underlying alphabet.

The most striking aspect of equivalence of automata is the following:

Proposition 1. *Every non-deterministic automaton is equivalent to a deterministic one.*

Proof. Suppose $A = (S_1, \Sigma, \delta_1, I_1, F_1)$ be a non-deterministic automaton. We seek a deterministic automaton $B = (S_1, \Sigma, \delta_2, I_2, F_2)$ such that $A \sim B$. Recall that the difference between A and B lie in the transition functions: δ_1 is a function from $S_1 \times \Sigma$ to $P(S_1)$, whereas δ_2 is a function from $S_2 \times \Sigma$ to S_2, and the fact that I_2 is required to be a singleton. The key to finding B is to realize that δ_1 can be converted into a function from $P(S_1) \times \Sigma$ to $P(S_1)$.

Now, define $S_2 := P(S_1)$, $I_2 := I_1$. For $T \subseteq S_1$ and $\alpha \in \Sigma$, let

$$\delta_2(T, \alpha) := \bigcup_{s \in T} \delta_1(s, \alpha).$$

As usual, we extend δ_2 so it is defined on all of $S_2 \times \Sigma^*$. We want to show that

$$\delta_2(\{s\}, a) = \delta_1(s, a)$$

for any $s \in S_1$ and any $a \in \Sigma^*$. This can be done by induction on the length of a:

- if $a = \lambda$, then $\delta_2(\{s\}, \lambda) = \{s\} = \delta_1(s, \lambda)$ by definition;
- if $a \in \Sigma$, then $\delta_2(\{s\}, a) = \bigcup_{s \in \{s\}} \delta_1(s, a) = \delta_1(s, a)$, again by definition;
- if $a = b\alpha$, where $b \in \Sigma^*$ and $\alpha \in \Sigma$, then by the induction step, $\delta_2(\{s\}, b) = \delta_1(s, b)$, so that $\delta_2(\{s\}, a) = \delta_2(\{s\}, b\alpha) = \delta_2(\delta_2(\{s\}, b), \alpha) = \delta_2(\delta_1(s, b), \alpha) = \bigcup_{t \in \delta_1(s,b)} \delta_1(t, \alpha) = \delta_1(\delta_1(s, b), \alpha) = \delta_1(s, b\alpha) = \delta_1(s, a)$.

Suppose a is accepted by A, so that $\delta_1(s,a) \cap F_1 \neq \varnothing$ for some $s \in I_1$. Then

$$\delta_2(I_2, a) = \bigcup_{s \in I_2} \delta_1(s,a) = \bigcup_{s \in I_1} \delta_1(s,a), \tag{1}$$

which has non-empty intersection with F_1. So, we want F_2 to consists of every element of S_2 that has non-empty intersection with F_1. Formally, we define $F_2 := \{F \subseteq S_1 \mid F \cap F_1 \neq \varnothing\}$. So what we have just shown is that $L(A) \subseteq L(B)$.

On the other hand, if a is accepted by B, then (1) above says that $\bigcup_{s \in I_1} \delta_1(s,a) \in F_2$, or $(\bigcup_{s \in I_1} \delta_1(s,a)) \cap F_1 \neq \varnothing$, or $\delta_1(s,a) \cap F_1 \neq \varnothing$ for some $s \in I_1$, which means a is accepted by A, proving the proposition. $\square$