Boolean operations on automata

Canonical name	BooleanOperationsOnAutomata
Date of creation	2013-03-22 18:03:09
Last modified on	2013-03-22 18:03:09
Owner	CWoo (3771)
Last modified by	CWoo (3771)
Numerical id	8
Author	CWoo (3771)
Entry type	Definition
Classification	msc 03D05
Classification	msc 68Q45
Defines	complement of an automaton
Defines	union of automata

A Boolean operation is any one of the three set-theoretic operations: union, intersection, and complementation. Boolean operations on automata are operations defined on automata resembling those from set theory.

Union of Two Automata

If $A_1 = (S_1, \Sigma, \delta_1, I_1, F_1)$ and $A_2 = (S_2, \Sigma, \delta_2, I_2, F_2)$. Let S, I, F be the disjoint unions of S_1 and S_2, I_1 and I_2, F_1 and F_2 respectively. For any $(s, a) \in S \times \Sigma$, define δ to be

$$\delta(s,a) := \begin{cases} \delta_1(s,a) & \text{if } s \in S_1, \\ \delta_2(s,a) & \text{if } s \in S_2. \end{cases}$$

The automaton $A = (S, \Sigma, \delta, I, F)$ is called the *union* of A_1 and A_2, and is denoted by $A_1 \cup A_2$. Intuitively, we take the disjoint union of the state diagrams of A_1 and A_2, and let it be the state diagram of A.

It is easy to see that $L(A_1 \cup A_2) = L(A_1) \cup L(A_2)$, and that $A_1 \cup A_2$ is equivalent to $A_2 \cup A_1$.

If only one starting state is required, one can modify the definition of $A_1 \cup A_2$ by adding a new state q and setting it as the new start state, then adding an edge from q to each of the states in I with label λ. The modified $A_1 \cup A_2$ is equivalent to the original $A_1 \cup A_2$. Furthermore, if A_1 and A_2 are DFA's, so is $A_1 \cup A_2$.

Complement of an Automaton

Let $A = (S, \Sigma, \delta, I, F)$ be an automaton. We define the complement A' of A as the quintuple

$$(S, \Sigma, \delta, I, F')$$

where $F' = S - F$. It is clear that A' is a well-defined automaton. Additionally, A is finite iff A' is, and A is deterministic iff A' is.

Visually, the state diagram of A' is a directed graph whose final nodes are exactly the non-final nodes of A.

It is obvious that $(A')' = A$ and that A is a DFA iff A' is.

Suppose A is a DFA. Then it is easy to see that a string $a \in \Sigma^*$ is accepted by A' precisely when a is rejected by A. If $L(A)$ denotes the language consisting of all words accepted by A. Then

$$L(A') = L(A)',$$

where $L(A)' = \Sigma^* - L(A)$, the complement of $L(A)$ in Σ^*.

However, because it is possible that for some (s, a), $\delta(s, a) = \varnothing$, the language accepted by an arbitrary automaton A' is in general not $L(A)'$.

Intersection of Two Autamata

One may define $A_1 \cap A_2$ to be $(A_1' \cup A_2')'$. However, defined this way,

$$L(A_1 \cap A_2) = L(A_1) \cap L(A_2)$$

is in general not true.

The way to ensure that the equation above always holds is to define $A_1 \cap A_2$ via the product of A_1 and A_2. For more details, see the entry on product of automata.

If both A_1 and A_2 are DFA's, then $A_1 \cap A_2$ is equivalent to $(A_1' \cup A_2')'$.