Proof book

Updated Oct 26 2018

Lemma 1.1: Equivalent states s_1 and s_2 of machines M_1 and M_2 (machines with same input and output alphabets but different states) go into equivalent states under each input.

Proof:

Suppose for some $s_1' = \delta(s_1, x)$ and $s_2' = \delta(s_2, x)$, we have $\bar{\lambda_1}(s_1', \bar{w}) \neq \bar{\lambda_2}(s_2', \bar{w})$ for some $\bar{w}$. Then, we have found an input sequence $a\bar{w}$ such that $\bar{\lambda_1}(s_1, a\bar{w}) \neq \bar{\lambda_2}(s_2, a\bar{w})$ which is a contradiction. $\square$

Theorem 1.1: Given a machine M_1, there is a reduced machine M equivalent to M_1. Furthermore, if M_2 is any machine equivalent to M_1, then there exists a state homomorphism which maps M_2 onto M.

Proof:

a. For any unreduced machine M_0 where there exists different but equivalent states s_1 and s_2, we can group these two states into one new state s such that all inward transitions to s_1 or s_2 now go into s, and all outward transitions from s_1 or s_2 now go out of s. Additionally, s is equivalent to both s_1 and s_2. This reduces the number of equivalent states (in terms of s_1/s_2) by one and does not introduce new equivalent states. We can repeat this process until every state has no other equivalent states but itself. Hence, by definition of *reduced*, we have a reduced machine, and it is shown that it must exist for every machine.

b. Since M_2 is equivalent to M, for every state s_2 in S_2 of M_2, there exists an (and only one as M is reduced) equivalent state s in S of M. Hence, we can define $h: S_2 \to S$ if and only if $s_2 \equiv s$. This mapping is also onto since equivalence goes both ways. By Lemma 1.1 and $s_2 \equiv h(s_2)$ for all s_2, we have $\delta_2(s_2, a) \equiv \delta(h(s_2), a)$ for all s_2 and a. By the definition of h, $h[\delta_2(s_2, a)] = \delta(h(s_2), a)$, which implies homomorphism from M_2 to M. $\square$

Corollary 1.1: If two reduced machines are equivalent, then they are isomorphic.

Justification:

Because $rM_1 \equiv rM_2$, there is a two-way state homomorphism, which is an isomorphism.

Theorem 1.4: If M' is a realization of M, then $\bar{\lambda}(s, \bar{x}) = \xi[\bar{\lambda}'(s', \iota(\bar{x}))]$.

Proof:

Lemma 1.4.a: $\bar{\delta}'[\alpha(s), \iota(\bar{x})] \subseteq \alpha[\bar{\delta}(s, \bar{x})]$

Proof:

We induce on $\bar{x}$. The base case holds by definition.

Inductively, we have $\bar{\delta}'[\alpha(s), \iota(\bar{x}')] \subseteq \alpha[\bar{\delta}(s, \bar{x}')]$. Assume $v \in \bar{\delta}'[\alpha(s), \iota(\bar{x}')]$, then $v \in \alpha[\bar{\delta}(s, \bar{x}')]$. Hence, $\delta'[v, \iota(x)] \in \delta'[\alpha[\bar{\delta}(s, \bar{x}')], \iota(x)]$. By the property of assginment, $\delta'[\alpha[\bar{\delta}(s, \bar{x}')], \iota(x)] \subseteq \alpha[\delta[\bar{\delta}(s, \bar{x}'), x]] = \alpha[\bar{\delta}(s, \bar{x})]$. That is, $v \in \alpha[\bar{\delta}(s, \bar{x})]$, and that $\bar{\delta}'[\alpha(s), \iota(\bar{x}')] \subseteq \alpha[\bar{\delta}(s, \bar{x})]$. Since in general we have $\bar{\delta}'[\alpha(s), \iota(\bar{x})] \subseteq \bar{\delta}'[\alpha(s), \iota(\bar{x}')]$, thus $\bar{\delta}'[\alpha(s), \iota(\bar{x})] \subseteq \alpha[\bar{\delta}(s, \bar{x})]$. $\square$