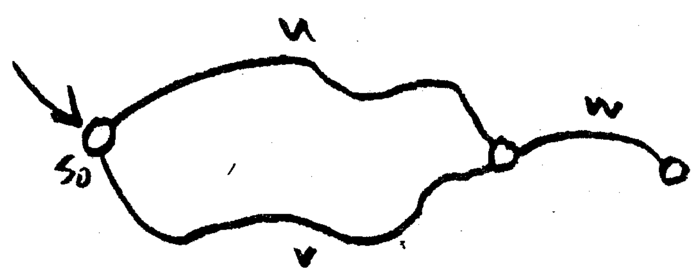
Myhill-Nerode Handout

Definition. An equivalence relation E on strings is *right invariant* iff concatenating a string w onto two equivalent strings u and v produces two strings (uw and vw) that are also equivalent; i.e., for all strings u, v, and w, we have $u\,E\,v \;\Rightarrow\; uw\,E\,vw$.

Theorem 1. A language L is accepted by a DFA iff L is the union of some equivalence classes of a right-invariant equivalence relation of finite index.

Proof, Part A. Suppose that a language L is accepted by a DFA $M = \langle S, \Sigma, \delta, s_0, F\rangle$. Define an equivalence relation E so that two strings u and w are equivalent iff the DFA (starting in state s_0) would transition to the same state by reading either u or w, i.e.,



$$u\,E\,v \text{ iff } \hat{\delta}(s_0, u) = \hat{\delta}(s_0, v)$$

Let "$EC(s_i)$" denote the equivalence class $\{w \mid \hat{\delta}(s_0, w) = s_i\}$ (i.e., the set of strings that transition the DFA to state s_i). It is easy to verify that all members of $EC(s_i)$ are indeed equivalent to each other. E is of finite index, because the set of equivalence classes is $\{EC(s_i) \mid s_i \in S\}$, which is finite. Since a string w is in language L iff w transitions the DFA to an accepting state, L is the union of the equivalence classes for accepting states: $L = \bigcup_{t\in F} EC(t)$. Now we need to show that E is right-invariant. This is simple; if $u\,E\,v$, then, as illustrated in the diagram,

$$\hat{\delta}(s_0, uw) \;=\; \hat{\delta}(\hat{\delta}(s_0, u), w) \;=\; \hat{\delta}(\hat{\delta}(s_0, v), w) \;=\; \hat{\delta}(s_0, vw)$$

Proof, Part B. Let us write "$[w]$" to denote the equivalence class to which w belongs. Consider a right-invariant equivalence relation E (on Σ^*, for a given Σ) of finite index. Let S be the (finite) set of equivalence classes. Suppose there is a subset F of these equivalence classes whose union is L. Define a DFA $M = \langle S, \Sigma, \delta, s_0, F\rangle$, where the start state s_0 is $[\epsilon]$ and the transition function is $\delta([u], \alpha) = [u\alpha]$.

Note that the transition function is well-defined; if u and v are in the same equivalence class, then $[u\alpha] = [v\alpha]$ because E is right-invariant.

To show that the DFA M recognizes L, we will show inductively that reading a string w puts the DFA into the state $[w]$; i.e., we will show that $\hat{\delta}(s_0, w) = [w]$.

- Base Case: If $w = \epsilon$, then $\hat{\delta}(s_0, \epsilon) = s_0 = [\epsilon]$.
- Recursive Case: If $w = u\alpha$, where α is a single symbol of the alphabet Σ, then $\hat{\delta}(s_0, u\alpha) = \delta(\hat{\delta}(s_0, u), \alpha) = \delta([u], \alpha) = [u\alpha]$.

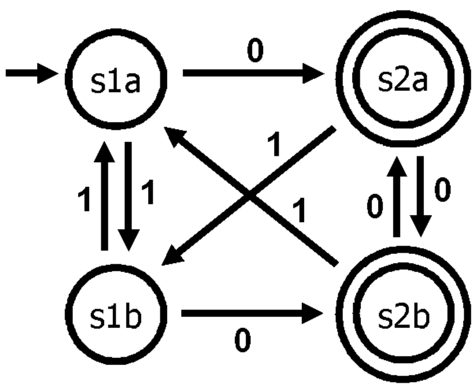
Theorem 2. Let L be a regular language, and let the equivalence relation R be defined by

$$u\,R\,v \quad \text{iff, for all } z \in \Sigma^*, \quad (u\,z \in L) \Leftrightarrow (v\,z \in L)$$

Then R is of finite index and the DFA constructed by the method of Theorem 1 using R is minimal.

Proof that R is of finite index. Let E be an equivalence relation satisfying Theorem 1. Then R must either equal E or be a *consolidation* of E; i.e., each equivalence class of R must either be an equivalence class of E or be formed by consolidating several equivalence classes of E into a single equivalence class. (Proof: Suppose, to the contrary, that there are two strings u and v that are equivalent under E but not under R. Then there must exist a string z such that $(uz \in L) \neq (vz \in L)$. But since u and v are equivalent under E, then uz and vz must also be equivalent under E (because E is right-invariant), even though only one of them is in L, so E would not satisfy the requirements of Theorem 1.) Since E has finite index, then *a fortiori* so does R.

Example. Consider the following DFA:



What is the equivalence relation E constructed by the method of Theorem 1 Part 1?

What is the equivalence relation R defined by Theorem 2 for the language recognized by this DFA?