

## Chapter 5

# Myhill-Nerode

The last characterization of regular languages that we consider is given by the Myhill-Nerode theorem.

### 5.1 Definition

The following definitions (taken from [19]) will lead us to the statement of the Myhill-Nerode theorem.

Let  $\equiv$  be an equivalence relation on  $\Sigma^*$ . Let  $L$  be a language over  $\Sigma$ .

**Definition 5.1.1.** The **equivalence class** of  $u \in \Sigma^*$  w.r.t.  $\equiv$  is the set of all  $v$  such that  $u \equiv v$ . It is denoted by  $[u]_{\equiv}$ .

**Definition 5.1.2.**

(i)  $\equiv$  is **right congruent** if and only if for all  $u, v \in \Sigma^*$  and  $a \in \Sigma$ ,

$$u \equiv v \Rightarrow u \cdot a \equiv v \cdot a.$$

(ii)  $\equiv$  **refines**  $L$  if and only if for all  $u, v \in \Sigma^*$ ,

$$u \equiv v \Rightarrow (u \in L \iff v \in L).$$

(iii)  $\equiv$  is of **finite index** if and only if it has finitely many equivalence classes, i.e.

$$\{[u]_{\equiv} \mid u \in \Sigma^*\} \text{ is finite}$$

**Definition 5.1.3.** A relation is Myhill-Nerode if and only if it satisfies properties (i), (ii) and (iii).

Fix everything below this line

**Definition 5.1.4.** *Given a language  $L$ , the coarsest Myhill-Nerode relation  $\equiv_L$  is the Myhill-Nerode relation that subsumes every other Myhill-Nerode relation, i.e.*

$$\forall u, v. u \equiv v \Rightarrow u \equiv_L v.$$

Listing 5.1: Myhill-Nerode relation

**Definition**  $MN\ w1\ w2 := \text{forall } w3, w1++w3 \setminus \text{in } L == (w2++w3 \setminus \text{in } L).$

**Theorem 5.1.1.** *Myhill-Nerode Theorem. A language  $L$  is regular if and only if  $\equiv_L$  is of finite index.*

## 5.2 Finite Partitionings and Equivalence Classes

Coq does not have quotient types. We pair up functions and proofs of certain properties to emulate quotient types.

A finite partitioning is a function from  $\Sigma^*$  to some finite type  $F$ . We use this concept to model equivalent classes in Coq. A finite partitioning of the Myhill-Nerode relation is a finite partitioning  $f$  that also respects the Myhill-Nerode relation, i.e.,

$$\forall u, v \in \Sigma^*. f(u) = f(v) \Leftrightarrow u \equiv_L v.$$

Listing 5.2: Finite partitioning of the Myhill-Nerode relation

**Definition**  $MN\_rel\ (f: \text{Fin\_eq\_cls}) := \text{forall } w1\ w2, f\ w1 == f\ w2 \Leftrightarrow MN\ w1\ w2.$

**Theorem 5.2.1.**  *$\equiv_L$  is of finite index if and only if there exists a finite partitioning of the Myhill-Nerode relation.*

*Proof.* If  $\equiv_L$  is of finite index, we use the set equivalence classes as a finite type and construct  $f$  such that

$$\forall w. f(w) = [w]_{\equiv}.$$

$f$  is a finite partitioning of the Myhill-Nerode relation by definition.

Conversely, if we have a finite partitioning of the Myhill-Nerode relation, we can easily see that  $\equiv_L$  must be of finite index since  $f$ 's values directly correspond to equivalence classes. The image of  $f$  is finite. Therefore,  $\equiv_L$  is of finite index.  $\square$

A more general concept is that of a refining finite partitioning of the Myhill-Nerode relation:

$$\forall u, v \in \Sigma^*. f(u) = f(v) \Rightarrow u \equiv_L v.$$

Listing 5.3: Refining finite partitioning of the Myhill-Nerode relation

**Definition**  $\text{MN\_ref}(f: \text{Fin\_eq\_cls}) := \text{forall } w1\ w2, f\ w1 == f\ w2 \rightarrow \text{MN } w1\ w2.$

We require all partitionings to be surjective. Therefore, every equivalence class  $x$  has at least one class representative which we denote  $cr(x)$ . Mathematically, this is not a restriction since there are no empty equivalence classes. In our constructive setting we would have to give a procedure that builds a minimal finite type  $F'$  from  $F$  and a corresponding function  $f'$  from  $\Sigma^*$  to  $F'$  such that  $f'$  is surjective and extensionally equal to  $f$ .

### 5.3 Minimizing Equivalence Classes

We will prove that refining finite partitionings can be converted into finite partitionings. For this purpose, we employ the table-filling algorithm to find indistinguishable states under the Myhill-Nerode relation ([14]). However, we do not rely on an automaton. In fact, we use the finite type  $F$ , i.e., the equivalence classes, instead of states.

Given a refining finite partitioning  $f$ , we construct a fixed-point algorithm. The algorithm initially outputs the set of equivalence classes that are distinguishable by the inclusion of their class representative in  $L$ . We denote this initial set  $dist_0$ .

$$dist_0 := \{(x, y) \in F \times F \mid cr(x) \in L \Leftrightarrow cr(y) \notin L\}.$$

To find more distinguishable equivalence classes, we have to identify equivalence classes that lead to distinguishable equivalence classes.

**Definition 5.3.1.** *We say that a pair of equivalence classes  $(x, y)$  **transitions** to  $(x', y')$  with  $a$  if and only if*

$$f(cr(x) \cdot a) = x' \wedge f(cr(y) \cdot a) = y'.$$

*We denote  $(x', y')$  by  $ext_a(x, y)$ .*

The fixed-point algorithm tries to extend the set of distinguishable equivalence classes by looking for a so-far undistinguishable pair of equivalence classes that transitions to a pair of distinguishable equivalence classes.

**Definition 5.3.2.**

$$\text{unnamed}(\text{dist}) := \text{dist}_0 \cup \text{dist} \cup \{(x, y) \mid \exists a. \text{ext}_a(x, y) \in \text{dist}\}$$

**Lemma 5.3.1.** *unnamed is monotone and has a fixed-point.*

*Proof.* Monotonicity follows directly from the monotonicity of  $\cup$ . The number of sets in  $F \times F$  is finite. Therefore, *unnamed* has a fixed point.  $\square$

Let **distinct** be the fixed point of *unnamed*. Let **equiv** be the complement of *distinct*.

Finish  
construc-  
tion

**Theorem 5.3.1.**  *$f_{\text{min}}$  is a finite partitioning of the Myhill-Nerode relation on  $L$ .*

Add for-  
maliza-  
tion

## 5.4 Finite Automata and Myhill-Nerode

We prove theorem 5.1.1 by proving it equivalent to the existence of an automaton that accepts  $L$ .

### 5.4.1 Finite Automata to Myhill-Nerode

Given DFA  $A$ , for all words  $w$  we define  $f(w)$  to be the last state of the run of  $w$  on  $A$ .

**Lemma 5.4.1.**  *$f$  is a refining finite partitioning of the Myhill-Nerode relation on  $\mathcal{L}(A)$ .*

*Proof.* The set of states of  $A$  is finite. For all  $u, v$  and  $w$  we have that if  $f(u) = f(v) = x$ , i.e., the runs of  $u$  and  $v$  on  $A$  end in the exact same state  $x$ . From this, we get that for all  $w$ , runs of  $u \cdot w$  and  $v \cdot w$  on  $A$  also end in the same state. Therefore,  $u \cdot w \in \mathcal{L}(A)$  if and only if  $v \cdot w \in \mathcal{L}(A)$ .  $\square$

**Theorem 5.4.1.** *If  $L$  is accepted by DFA  $A$ , then there exists a finite partitioning of the Myhill-Nerode relation on  $L$ .*

*Proof.* From lemma 5.4.1 we get a refining finite partitioning  $f$  of the Myhill-Nerode relation on  $\mathcal{L}(A)$ . Since  $L$  is accepted by  $A$ ,  $L = \mathcal{L}(A)$ . Therefore,  $f$  is a refining finite partitioning of the Myhill-Nerode relation on  $L$ . By theorem 5.3.1 there also exists a finite partition of the Myhill-Nerode relation on  $L$ .  $\square$

### 5.4.2 Myhill-Nerode to Finite Automata