

# Chapter 1

## Myhill-Nerode

The last characterization we consider is given by the Myhill-Nerode theorem.

### 1.1 Definition

The following definitions (taken from ?) will lead us to the statement of the Myhill-Nerode theorem. We assume  $\equiv$  to be an equivalence relation on  $\Sigma^*$ , and  $L$  to be a language over  $\Sigma$ .

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

$$x \equiv y \Rightarrow x \cdot a \equiv y \cdot a.$$

- (ii)  $\equiv$  **refines**  $L$  if and only if for all  $x, y \in \Sigma^*$ ,

$$x \equiv y \Rightarrow (x \in L \Leftrightarrow y \in L).$$

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

$$\{[x] | x \in \Sigma^*\} \text{ is finite}$$

**Definition 1.1.1.** A relation is Myhill-Nerode if and only if it satisfies properties ??, ?? and ??.

Fix everything below this line

Given a language  $L$ , the Myhill-Nerode relation  $\approx_L$  is defined such that

$$\forall u, v \in \Sigma^*. u \approx_L v \iff \forall w \in \Sigma^*. u \cdot w \in L \Leftrightarrow v \cdot w \in L.$$

Listing 1.1: Myhill-Nerode relation

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

**Theorem 1.1.1.** Myhill-Nerode Theorem. A language  $L$  is regular if and only if  $\approx_L$  is of finite index.

## 1.2 Finite Partitionings and Equivalence Classes

CoQ does not have quotient types. We pair up functions and proofs for certain properties of those functions 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.,

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

Listing 1.2: Finite partitioning of the Myhill-Nerode relation

**Definition** `MN_rel (f: Fin_eq_cls) := forall w1 w2, f w1 == f w2 <-> MN w1 w2.`

**Theorem 1.2.1.**  *$\approx_L$  is of finite index if and only if there exists a finite partitioning of the Myhill-Nerode relation.*

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

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

$f$  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  $\approx_L$  must be of finite index since  $f$ 's values directly correspond to equivalence classes. The image of  $f$  is finite. Therefore,  $\approx_L$  is of finite index.  $\square$

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

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

Listing 1.3: Refining finite partitioning of the Myhill-Nerode relation

**Definition** `MN_ref (f: Fin_eq_cls) := forall w1 w2, f w1 == f w2 -> 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  $f$ .

### 1.3 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 (?). 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 1.3.1.** 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 1.3.2.**

$$unnamed(dist) := dist_0 \cup dist \cup \{(x, y) \mid \exists a. ext_a(x, y) \in dist\}$$

**Lemma 1.3.1.** *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. □

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

**Theorem 1.3.1.**  *$f_{min}$  is a finite partitioning of the Myhill-Nerode relation on  $L$ .*

Finish  
construc-  
tion

Add for-  
maliza-  
tion

## 1.4 Finite Automata and Myhill-Nerode

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

### 1.4.1 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 1.4.1.**  *$f$  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 1.4.1.** *If  $L$  is accepted by DFA  $A$ , then there exists a finite partitioning of the Myhill-Nerode relation on  $L$ .*

*Proof.* From lemma ?? 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 ?? there also exists a finite partition of the Myhill-Nerode relation on  $L$ .  $\square$

### 1.4.2 Myhill-Nerode to Finite Automata

**Chapter 2**

**Conclusion**

## Chapter 3

## References