

The categorical origin of Non-deterministic automata

Ryuya Hora

April 18, 2022

Abstract

We interpret NFA in terms of category theory and prove two theorems. The first is that NFA is just DFA in the category of truth-value enriched profunctors. The second, inspired by the first theorem, is a category-theoretic proof of the equivalence of NFA and DFA. It follows naturally from the category-theoretic properties of profunctors and, at the same time, gives a category-theoretic interpretation of the Radin-Scott powerset construction.

Contents

1	Category of deterministic automata $\text{DA}(\mathcal{C})$	1
2	Category of non-deterministic automata NA	2
3	Truth value enriched profunctors ΩProf	5
4	Main theorem $\text{NA} \simeq \text{DA}(\Omega\text{Prof})$	7
5	$(\mathcal{C}, T)$-regular language	8
5.1	Definition of $(\mathcal{C}, T)$ -regular language	8
5.2	Comparison of $(\mathcal{C}, T)$ -regularities	9
6	Equivalence of DFA and NFA from the categorical point of view	10

Throughout this paper, we fix a set of alphabets Σ , and assume that the symbol ϵ does not belongs to Σ .

1 Category of deterministic automata $\text{DA}(\mathcal{C})$

At first, we define the notion of deterministic automata in a category $\mathcal{C}$. We choose the simplest definition of them here. We will deal with additional structures such as an initial state and accept states in section 5.

Definition 1.1 (deterministic automaton in $\mathcal{C}$). Let $\mathcal{C}$ be a category. A deterministic automaton in $\mathcal{C}$ is an object X together with a family of endomorphisms $(\delta_a : X \rightarrow X)_{a \in \Sigma}$ indexed by the set of alphabets Σ .

A deterministic automaton in $\mathbf{Set}$ is an usual one, and one in $\mathbf{FinSet}$ is a deterministic finite automaton. Although we will consider automata that are not necessarily finite, the following arguments can be restricted to finite automata.

Definition 1.2 (morphism of deterministic automata). Let $\mathcal{C}$ be a category and $(X, (\delta_a : X \rightarrow X)_{a \in \Sigma}), (X', (\delta'_a : X' \rightarrow X')_{a \in \Sigma})$ be deterministic automata in $\mathcal{C}$. A morphism of deterministic automata $(X, (\delta_a : X \rightarrow X)_{a \in \Sigma}) \rightarrow (X', (\delta'_a : X' \rightarrow X')_{a \in \Sigma})$ is a morphism $f : X \rightarrow X'$ in $\mathcal{C}$ such that for any $a \in \Sigma$ the following diagram commutes:

$$\begin{array}{ccc} X & \xrightarrow{f} & X' \\ \downarrow \delta_a & & \downarrow \delta'_a \\ X & \xrightarrow{f} & X' \end{array}$$

A category $\mathbf{DA}(\mathcal{C})$ has deterministic automata as its objects and morphisms of deterministic automata as its morphisms.

Let Σ^* denote the free monoid of a set Σ , namely the monoid of words of Σ together with their concatenation. The category $\mathbf{DA}(\mathcal{C})$ is just a functor category $\mathcal{C}^{\Sigma^*}$.

Therefore we have

Proposition 1.3 (functoriality of $\mathbf{DA} : \mathbf{CAT} \rightarrow \mathbf{CAT}$). Let $F : \mathcal{C} \rightarrow \mathcal{D}$ be a functor. F induces a functor $\mathbf{DA}(\mathcal{C}) \rightarrow \mathbf{DA}(\mathcal{D})$ by just sending a deterministic automaton $(X, (\delta_a)_{a \in \Sigma})$ to $(FX, (F\delta_a)_{a \in \Sigma})$ and a morphism f to Ff .

This proposition will be used in subsection 5.2 to convert a deterministic automaton in a category to a deterministic automaton in another category.

2 Category of non-deterministic automata $\mathbf{NA}$

Recall the notion of non-deterministic automata. Though we thought of the notion of deterministic automata in an arbitrary category, we consider only usual non-deterministic automata, in $\mathbf{Set}$.

Definition 2.1 (non-deterministic automata). A non-deterministic automaton is a set Q together with a family of functions $(\delta_a : Q \rightarrow \mathcal{P}(Q))_{a \in \Sigma \cup \{\epsilon\}}$ indexed by Σ , where $\mathcal{P}(Q)$ denotes the power set of Q .

Remark 2.2. It is worth emphasizing that this definition has two different points compared with the definition of deterministic automata.

1. For any alphabet $a \in \Sigma$, the codomain of δ_a is replaced by $\mathcal{P}(Q)$.
2. A new symbol ϵ and an associated function δ_ϵ are introduced.

The first point is the obvious one. It means that δ become non-deterministic. The second point is called ϵ -move. Some states are allowed to spontaneously move to other states without reading alphabets. In this sense, ϵ -move is a “free-accessibility” in a non-deterministic automaton.

Next, recall how it works.

Definition 2.3. Let $(Q, (\delta_a : Q \rightarrow \mathcal{P}(Q))_{a \in \Sigma \cup \{\epsilon\}})$ be a non-deterministic automaton. For $q_0, q_1 \in Q$ and $w \in \Sigma^*$, q_1 is *accessible with w* from q_0 iff there exist a non-negative integer $n \in \mathbb{N}$, $a_1, \dots, a_n \in \Sigma \cup \{\epsilon\}$, and $r_0, \dots, r_n \in Q$ that satisfy the following condition

- the concatenation of $a_0, \dots, a_n$ is w , where the symbol ϵ is regarded as a empty word.
- $r_0 = q_0$
- $r_{i+1} \in \delta_{a_{i+1}}(r_i)$ for any $0 \leq i < n$
- $r_n = q_1$

Let $q_0 \overset{w}{\rightsquigarrow} q_1$ mean that q_1 is accessible with w from q_0 , and as a special case, $q_0 \rightsquigarrow q_1$ mean that q_1 is accessible with the empty word from q_0 .

To define the category of non-deterministic automata, the notion of morphisms of non-deterministic automata should be defined. One might naively try to define them as functions that satisfy some conditions. However, our definition is different. Morphisms of non-deterministic automata should also be non-deterministic!

Recall that a morphism of deterministic automata $Q \rightarrow Q'$ is a (deterministic) function $f : Q \rightarrow Q'$ that respects the transition of states by alphabets. In detail, it means that for any $q \in Q$ and $a \in \Sigma$, the following two transitions coincide:

$$\begin{array}{ccc} q & \xrightarrow{f} & f(q) \\ \downarrow a & & \downarrow a \\ \delta_a(q) & \xrightarrow{f} & f(\delta_a(q)) = \delta'_a(f(q)) \end{array}$$

We define morphisms of non-deterministic automata in a similar way. Roughly speaking, a morphism of non-deterministic automata are defined below as a non-deterministic function, which is just a function $f : Q \rightarrow \mathcal{P}(Q')$, that respects accessibility of states by alphabets, instead of deterministic transition.

However, respecting accessibility by alphabets is not enough. From the perspective introduced in Remark 2.2, it should also respect the other accessibility of non-deterministic automata, namely ϵ -move. Once a morphism $f : Q \rightarrow \mathcal{P}(Q')$ is also regarded as some kind of accessibility relation, it is natural to assume that it is compatible with ϵ -move, which is “free-accessibility” in a non-deterministic automaton.

Just for simplicity of notation, for sets X, Y , a function $f : X \rightarrow \mathcal{P}(Y)$ and $x \in X, y \in Y$, let $x \xrightarrow{f} y$ denote $y \in f(x)$.

Definition 2.4 (morphism of non-deterministic automata). A morphism of non-deterministic automata from $(Q, (\delta_a : Q \rightarrow \mathcal{P}(Q))_{a \in \Sigma \cup \{\epsilon\}})$ to $(Q', (\delta'_a : Q' \rightarrow \mathcal{P}(Q'))_{a \in \Sigma \cup \{\epsilon\}})$ is a function $f : Q \rightarrow \mathcal{P}(Q')$ that satisfies the following two conditions:

1. (respecting accessibility by alphabets) For any $q \in Q, q' \in Q'$ and $a \in \Sigma$, the following two conditions are equivalent.
 - there exists $q_0 \in Q$ such that $q \xrightarrow{a} q_0$ and $q_0 \xrightarrow{f} q'$
 - there exists $q'_0 \in Q'$ such that $q \xrightarrow{f} q'_0$ and $q'_0 \xrightarrow{a} q'$

$$\begin{array}{ccc} q & \xrightarrow{\quad f \quad} & q'_0 \\ \downarrow \scriptstyle a & & \downarrow \scriptstyle a \\ q_0 & \xrightarrow{\quad f \quad} & q' \end{array}$$

2. (respecting accessibility with ϵ -move) For any $q_0, q_1 \in Q$ and $q'_0, q'_1 \in Q'$, if

$$q_0 \rightsquigarrow q_1 \xrightarrow{\quad f \quad} q'_0 \rightsquigarrow q'_1$$

then

$$q_0 \xrightarrow{\quad f \quad} q'_1$$

To call this notion morphisms of non-deterministic automata, verification is needed. In other words, we should check that they forms a category NA.

Before checking it, composition of non-deterministic morphisms should be defined. For functions $f : X \rightarrow \mathcal{P}(Y)$ and $g : Y \rightarrow \mathcal{P}(Z)$, $g \bullet f : X \rightarrow \mathcal{P}(Z)$ is defined as

$$(g \bullet f)(x) := \bigcup_{y \in f(x)} g(y)$$

In other words,

$$x \xrightarrow{\quad g \bullet f \quad} z$$

if and only if there exists $y \in Y$ such that

$$x \xrightarrow{f} y \xrightarrow{g} z$$

Proposition 2.5. Non-deterministic automata, their morphisms and compositions given by $\bullet$ form a category.

Proof. It is general fact that the $\bullet$ composition is associative. Therefore, there are essentially two things to prove.

- morphisms of non-deterministic automata are closed under composition.
- For any non-deterministic automaton Q , there exists the identity morphism of Q .

The former property can be proved in a straightforward way. To prove the later one, it is enough to see that the function $I_Q : Q \rightarrow \mathcal{P}(Q)$ defined as

$$I_Q(q_0) := \{q \in Q \mid q_0 \rightsquigarrow q\}$$

is an endomorphism of Q and moreover is the identity. $\square$

Definition 2.6 (The category NA). NA denotes the category that consists of non-deterministic automata and morphisms of them.

3 Truth value enriched profunctors ΩProf

In this section, we describe the category of Ω -enriched profunctors in an elementary way, for readers who are not familiar with profunctor. The reason to recall this category will be explained in section 4. Ω denotes the totally ordered set with two elements $\perp \leq \top$, which can be regarded as the poset of truth values.

Definition 3.1 (The category ΩProf). The category ΩProf has preordered sets (in other words, Ω -enriched categories) as its objects and order preserving map $f : P \times Q^{\text{op}} \rightarrow \Omega$ (in other words, Ω -enriched profunctor $P \rightrightarrows Q$) as its morphism from P to Q . The composition $g \bullet f : P \times R^{\text{op}} \rightarrow \Omega$ of $f : P \times Q^{\text{op}} \rightarrow \Omega$ and $g : Q \times R^{\text{op}} \rightarrow \Omega$ is defined so that $(g \bullet f)(p, r) = \top$ if and only if there exists $q \in Q$ such that $f(p, q) = \top$ and $g(q, r) = \top$.

As a morphism of non-deterministic automata, let $p \xrightarrow{f} q$ denote $f(p, q) = \top$.

Remark 3.2 (2-dimensional structure of ΩProf). For the readers who are familiar with profunctors, we should mention the 2-dimensional structure of ΩProf . In general, profunctors forms bicategory, not (1-)category. In fact, ΩProf becomes strict 2-category, not just bicategory. In this paper, we ignore 2-cells of the 2-category, and regard it as a (1-)category.

The relationship between $\mathbf{Set}$ and $\Omega\mathbf{Prof}$ should be mentioned. First, we give a functor that faithfully embeds $\mathbf{Set}$ into $\Omega\mathbf{Prof}$.

Definition 3.3 ($I : \mathbf{Set} \rightarrow \Omega\mathbf{Prof}$). The functor $I : \mathbf{Set} \rightarrow \Omega\mathbf{Prof}$ sends a set X to a discrete preordered set $I(X)$ whose underlying set is X and a function $f : X \rightarrow Y$ to its graph $I(f) : X \times Y \rightarrow \Omega$, in other words $I(f)(x, y) = \top \iff f(x) = y$.

The standard way to obtain a functor in the reverse direction is to consider adjunctions. In fact, the embedding function I has a right adjoint. Let $\mathbf{1}$ denotes the singleton preordered set.

Proposition 3.4. The hom functor $\Omega\mathbf{Prof}(\mathbf{1}, -) : \Omega\mathbf{Prof} \rightarrow \mathbf{Set}$ is the right adjoint of $I : \mathbf{Set} \rightarrow \Omega\mathbf{Prof}$.

$$\mathbf{Set} \begin{array}{c} \xrightarrow{I} \\ \perp \\ \xleftarrow{\Omega\mathbf{Prof}(\mathbf{1}, -)} \end{array} \Omega\mathbf{Prof}$$

Proof. It is enough to see that $I(X)$ is the X -times coproduct of $\mathbf{1}$. □

The category of finite sets is denoted by $\mathbf{FinSet}$ and the full subcategory of $\Omega\mathbf{Prof}$ spanned by finite preordered sets is denoted by $\mathbf{Fin}\Omega\mathbf{Prof}$. By restricting the above adjunction, we have the finite version of the adjunction.

Corollary 3.5. The hom functor $\mathbf{Fin}\Omega\mathbf{Prof}(\mathbf{1}, -) : \mathbf{Fin}\Omega\mathbf{Prof} \rightarrow \mathbf{FinSet}$ is the right adjoint of $I : \mathbf{FinSet} \rightarrow \mathbf{Fin}\Omega\mathbf{Prof}$.

$$\mathbf{FinSet} \begin{array}{c} \xrightarrow{I} \\ \perp \\ \xleftarrow{\mathbf{Fin}\Omega\mathbf{Prof}(\mathbf{1}, -)} \end{array} \mathbf{Fin}\Omega\mathbf{Prof}$$

This corollary along with the faithfulness of I will be used in section 6.

Remark 3.6 (profunctors as cocontinuous functors between presheaf categories). One of the most fundamental property of profunctors is the correspondence with cocontinuous functor between presheaf category. Let $\mathbf{CLattice}_{\sup}$ denote the category of complete lattices and sup-preserving (but not necessarily inf-preserving) functions. For any preordered set P , $\hat{P}$ denotes the complete lattice of downward subsets of P .

Under these notations, the following claim holds: A functor $\Omega\mathbf{Prof} \rightarrow \mathbf{CLattice}_{\sup}$ that sends $f : P \times Q^{\text{op}} \rightarrow \Omega$ to $\hat{P} \rightarrow \hat{Q} : D \mapsto \{q \in Q \mid \exists p \in D, p \overset{f}{\dashrightarrow} q\}$ is fully faithful. Especially, it gives an equivalence of categories between its essential image and $\Omega\mathbf{Prof}$.

Through this claim, Proposition 3.4 can be seen as a consequence of *the universality of power sets*. In detail, the power set functor $\mathcal{P} : \mathbf{Set} \rightarrow \mathbf{CLattice}_{\sup}$ is the left adjoint of the forgetful functor $U : \mathbf{CLattice}_{\sup} \rightarrow \mathbf{Set}$.

$$\mathbf{Set} \begin{array}{c} \xrightarrow{\mathcal{P}} \\ \perp \\ \xleftarrow{U} \end{array} \mathbf{CLattice}_{\sup}$$

The adjunction in Proposition 3.4 is the restriction of the above adjunction to the full subcategory ΩProf .

4 Main theorem $\text{NA} \simeq \text{DA}(\Omega\text{Prof})$

In this section, we prove that deterministic automata in ΩProf are actually non-deterministic automata! Before doing that, we will give a informal explanation on the reason why ΩProf is related.

As mentioned in Remark 2.2, there are two different points in definitions of non-deterministic automata, compared to deterministic automata. The first point is that transition is non-deterministic. This corresponds with the fact that morphisms in ΩProf are relations. The second point is that the transition in a non-deterministic automaton includes ϵ -move. Basically, when considering state transitions, we are not interested in whether or not ϵ -move was used, or if so, how many times. The only thing that matters is whether a transition can be done using ϵ -move finite times. In this sense, ϵ -move can be understood under the idea of considering a preorder $\rightsquigarrow$ on the state set Q . This corresponds with the fact that objects in ΩProf are preordered sets.

Definition 4.1 (associated non-deterministic automaton). Let $(Q, (\delta_a : Q \times Q^{\text{op}} \rightarrow \Omega))$ be a deterministic automaton in ΩProf . Its *associated non-deterministic automaton* $\mathcal{A}(Q) = (Q, (\delta'_a : Q \rightarrow \mathcal{P}(Q))_{a \in \Sigma \cup \{\epsilon\}})$ has the following data:

- set of states of $\mathcal{A}(Q)$ is (the underlying set of preordered set) Q .
- non-deterministic transition of $\mathcal{A}(Q)$ by an alphabet $a \in \Sigma$ is defined by $\delta'_a(q_0) := \{q \in Q \mid \delta_a(q_0, q) = \top\}$
- ϵ -move of $\mathcal{A}(Q)$ is defined by $\delta'_\epsilon(q_0) := \{q \in Q \mid q_0 \geq q\}$

Lemma 4.2. By sending a morphism $f : P \times Q^{\text{op}} \rightarrow \Omega$ in $\text{DA}(\Omega\text{Prof})$ to a morphism $\mathcal{A}(f) : P \rightarrow \mathcal{P}(Q) : p \mapsto \{q \in Q \mid f(p, q) = \top\}$ in NA , these data define a fully faithful functor $\mathcal{A} : \text{DA}(\Omega\text{Prof}) \rightarrow \text{NA}$.

Proof. Let $(P, (\delta_{P,a} : P \times P^{\text{op}} \rightarrow \Omega)), (Q, (\delta_{Q,a} : Q \times Q^{\text{op}} \rightarrow \Omega))$ be deterministic automata in ΩProf . What we should prove first is that a (a priori, not necessarily order-preserving) relation $f : P \times Q^{\text{op}} \rightarrow \Omega$ is a morphism in $\text{DA}(\Omega\text{Prof})$ if and only if it is a morphism in NA . In fact, f is a morphism in ΩProf if and only if f satisfies the second condition for being a morphism in NA , namely respecting accessibility with ϵ -move. Assuming that f is order-preserving, f is a morphism in $\text{DA}(\Omega\text{Prof})$ if and only if f satisfies the first condition for being a morphism in NA , namely respecting accessibility by alphabets.

Preservation of compositions and identities can be easily proved. $\square$

Theorem 4.3 ($\text{NA} \simeq \text{DA}(\Omega\text{Prof})$). $\mathcal{A} : \text{DA}(\Omega\text{Prof}) \rightarrow \text{NA}$ gives an equivalence of categories.

Proof. By the preceding lemma, it is enough to prove that $\mathcal{A} : \text{DA}(\Omega\text{Prof}) \rightarrow \text{NA}$ is essentially surjective. Take an arbitrary non-deterministic automaton $(Q, (\delta_a : Q \rightarrow \mathcal{P}(Q))_{\Sigma \cup \{\epsilon\}})$. A preorder $\leq$ on Q is defined by $q_0 \geq q_1 \iff q_0 \rightsquigarrow q_1$ and for $a \in \Sigma$, $\delta'_a : (Q, \leq) \times (Q, \leq)^{\text{op}} \rightarrow \Omega$ is defined by $\delta'_a(q_1, q_1) = \top \iff q_0 \xrightarrow{a} q_1$. Then, $((Q, \leq), (\delta'_a)_{a \in \Sigma})$ is a deterministic automaton in ΩProf .

Two non-deterministic automata $(Q, (\delta_a : Q \rightarrow \mathcal{P}(Q))_{\Sigma \cup \{\epsilon\}})$ and $\mathcal{A}((Q, \leq), (\delta'_a)_{a \in \Sigma})$ are isomorphic. In fact, $I_Q : Q \times Q^{\text{op}} \rightarrow \Omega$ defined by $I_Q(q_0, q_1) = \top \iff q_0 \rightsquigarrow q_1$ gives isomorphisms between the two non-deterministic automata in both directions. $\square$

The full subcategory of NA spanned by non-deterministic finite automata is denoted by NFA. By restricting the equivalence given in Theorem 4.3, we have the following corollary.

Corollary 4.4 ($\text{NFA} \simeq \text{DA}(\text{Fin}\Omega\text{Prof})$). The restriction of $\mathcal{A} : \text{DA}(\Omega\text{Prof}) \rightarrow \text{NA}$ gives an equivalence of categories $\text{NFA} \simeq \text{DA}(\text{Fin}\Omega\text{Prof})$

5 $(\mathcal{C}, T)$ -regular language

From this section, we consider the recognition of languages.

5.1 Definition of $(\mathcal{C}, T)$ -regular language

In this subsection, we generalize the definition of regular language in the context of $\text{DA}(\mathcal{C})$.

Definition 5.1 ($(\mathcal{C}, T)$ -regular language). Let $(\mathcal{C}, T)$ be a pointed category, in other words, a locally small category $\mathcal{C}$ together with its object $T \in \text{ob}(\mathcal{C})$. A language $\mathcal{L} \subset \Sigma^*$ is said to be $(\mathcal{C}, T)$ -regular if there exists a deterministic automaton $(X, (\delta_a)_{a \in \Sigma})$ and a morphism, (which is thought of as an initial state) $i : T \rightarrow X$ and a set $F \subset \mathcal{C}(T, X)$ of morphisms from T to X , (which is thought of as a set of accept states) such that

$$\mathcal{L} = \{w = a_1 \dots a_n \in \Sigma^* \mid \delta_{a_n} \circ \dots \circ \delta_{a_1} \circ i \in F\}$$

The set of all $(\mathcal{C}, T)$ -regular languages is denoted by $\text{RegLan}(\mathcal{C}, T)$

Example 5.2 ($(\text{FinSet}, 1)$ -regular languages). $\mathcal{L} \subset \Sigma^*$ is $(\text{FinSet}, 1)$ -regular if and only if it is regular in the usual sense.

Example 5.3 ($(\text{Set}, 1)$ -regular languages). Every language $\mathcal{L} \subset \Sigma^*$ is $(\text{Set}, 1)$ -regular.

Example 5.4 (group-regular language). Let G be a group and $*$ be the unique object of one-object category G . $\mathcal{L} \subset \Sigma^*$ is $(G, *)$ -regular if and only if there exist a Σ -indexed family $(g_a)_{a \in \Sigma}$ of elements of G and $S \subset G$ such that

$$\mathcal{L} = \{w = a_1 \dots a_n \in \Sigma^* \mid g_{a_n} \dots g_{a_1} \in S\}$$

. When S is the singleton $S = \{e\}$ for the unit $e \in G$ and $(g_a)_{a \in \Sigma}$ is a generating family of G , the corresponding language is the set of all relations of $(g_a)_{a \in \Sigma}$. In this case, G has a presentation $G = \langle (g_a)_{a \in \Sigma} \mid \mathcal{L} \rangle$. In this point of view, $(G, *)$ -regularity can be regarded as realizability as a solution of *generalized word problem* of G .

5.2 Comparison of $(\mathcal{C}, T)$ -regularities

In this subsection, we compare two regularities for two given pointed categories. The basic tool to do it is the following comparison lemma, which uses the conversion given by Proposition 1.3 to construct a suitable deterministic automaton.

Lemma 5.5 (comparison lemma). Let $(\mathcal{C}, T_0), (\mathcal{D}, T_1)$ be pointed categories and $G : \mathcal{C} \rightarrow \mathcal{D}$ be a functor. If there exists a morphism $e : T_1 \rightarrow GT_0$ with the property that for any $X \in \text{ob}(\mathcal{C})$ and a morphism $f : T_1 \rightarrow GX$, there exists at most one morphism $g : T_0 \rightarrow X$ such that

$$\begin{array}{ccc} T_1 & \xrightarrow{e} & GT_0 \\ & \searrow f & \downarrow Gg \\ & & GX \end{array}$$

commutes (in other words, the composite function of

$$\mathcal{C}(T_0, X) \xrightarrow{G} \mathcal{D}(GT_0, GX) \xrightarrow{- \circ e} \mathcal{D}(T_1, GX)$$

is injective), then $\text{RegLan}(\mathcal{C}, T_0) \subset \text{RegLan}(\mathcal{D}, T_1)$.

Proof. Let $\mathcal{L}$ be a $(\mathcal{C}, T_0)$ -regular language. By definition of $(\mathcal{C}, T_0)$ -regularity, there exist a deterministic automaton $(X, (\delta_a : X \rightarrow X)_{a \in \Sigma})$, initial state $i : T_0 \rightarrow X$ and $F \subset \mathcal{C}(T_0, X)$ such that

$$\mathcal{L} = \{w = a_1 \dots a_n \in \Sigma^* \mid \delta_{a_n} \circ \dots \circ \delta_{a_1} \circ i \in F\}$$

. Let $F' \subset \mathcal{C}(T_1, GX)$ be the image of the composite function of the following functions:

$$\mathcal{C}(T_0, X) \xrightarrow{G} \mathcal{D}(GT_0, GX) \xrightarrow{- \circ e} \mathcal{D}(T_1, GX)$$

. Since the composite function is injective by the hypothesis, for any $w = a_1 \dots a_n \in \Sigma^*$,

$$\begin{aligned} w \in \mathcal{L} &\iff \delta_{a_n} \circ \dots \circ \delta_{a_1} \circ i \in F \\ &\iff G(\delta_{a_n} \circ \dots \circ \delta_{a_1} \circ i) \circ e \in F' \\ &\iff G\delta_{a_n} \circ \dots \circ G\delta_{a_1} \circ (Gi \circ e) \in F' \end{aligned}$$

. Therefore, a deterministic automaton $(GX, (G\delta_a)_{a \in \Sigma})$ together with an initial state $Gi \circ e$ and a set of accept states $F' \subset \mathcal{D}(T_1, GX)$ recognizes $\mathcal{L}$. $\square$

This lemma has a few corollaries.

Corollary 5.6 ($(\mathcal{C}, T)$ -regularity for a locally finite category $\mathcal{C}$). Let $(\mathcal{C}, T)$ be a pointed category. If $\mathcal{C}$ is locally finite, in other words all hom-sets are finite sets, then $\text{RegLan}(\mathcal{C}, T) \subset \text{RegLan}(\text{FinSet}, 1)$.

Proof. The comparison lemma can be used to the hom-functor $\mathcal{C}(T, -) : \mathcal{C} \rightarrow \text{FinSet}$, since the function $1 \rightarrow \mathcal{C}(T, T)$ that sends the unique element $*$ of 1 to $\text{id}_T \in \mathcal{C}(T, T)$ satisfies the hypothesis of the lemma. $\square$

Corollary 5.7 (faithful functor). Let $G : \mathcal{C} \rightarrow \mathcal{D}$ be a faithful functor and T be an object of $\mathcal{C}$. Then $\text{RegLan}(\mathcal{C}, T) \subset \text{RegLan}(\mathcal{D}, GT)$.

Proof. Take $\text{id}_{GT} : GT \rightarrow GT$ as the morphism e in the comparison lemma. $\square$

Corollary 5.8 (right adjoint functor). Let $G : \mathcal{C} \rightarrow \mathcal{D}$ be a functor and T be an object of $\mathcal{D}$. If G has a left adjoint $L : \mathcal{D} \rightarrow \mathcal{C}$, then $\text{RegLan}(\mathcal{C}, LT) \subset \text{RegLan}(\mathcal{D}, T)$.

Proof. Take the T -component $\eta_T : T \rightarrow LGT$ of the unit η of the adjunction $L \vdash R$ as the morphism e in the comparison lemma. $\square$

By combining the preceding two corollaries, we have the following useful corollaries.

Corollary 5.9 (faithful left adjoints witness the equivalence of regularities.). Let $G : \mathcal{C} \rightarrow \mathcal{D}$ be a faithful left adjoint functor and T be an object of $\mathcal{C}$. Then $\text{RegLan}(\mathcal{C}, T) = \text{RegLan}(\mathcal{D}, GT)$.

6 Equivalence of DFA and NFA from the categorical point of view

In this section, we give a categorical proof of the equivalence of DFA and NFA and explain how it corresponds to the classical proof, which is called the Rabin–Scott powerset construction. Before we get to the proof, one thing should be noted. That is, even though Corollary 4.4 holds, the recognition of a language by NFA in the ordinary sense and the recognition of a language by the DA in $\text{Fin}\Omega\text{Prof}$, i.e. $(\text{Fin}\Omega\text{Prof}, 1)$ -regularity, are defined slightly differently (although they are consequently identical). More specifically, $(\text{Fin}\Omega\text{Prof}, 1)$ -regularity is seemingly a stronger condition than recognition by NFA in the usual sense. In fact, a straightforward translation of the definition through the correspondence given by Theorem 4.4 yields the following lemma.

Lemma 6.1. If $\mathcal{L} \subset \Sigma^*$ is recognized by a non-deterministic automaton Q , then $\mathcal{L}$ is recognized by the deterministic automaton in ΩProf that is associated to Q by Theorem 4.4.

Proof. We can assume that the associated deterministic automaton has the same underlying set Q together with the preorder given by $\rightsquigarrow$. If $\mathcal{L}$ is recognized by a initial state q_0 and the accept states $F \subset Q$, define an initial state $i : \mathbf{1} \times Q^{\text{op}} \rightarrow \Omega$ by $i(*, q) = \top \iff q_0 \rightsquigarrow q$ and define the accept states $F' \subset \text{Fin}\Omega\text{Prof}(\mathbf{1}, Q)$ by $f : \mathbf{1} \times Q^{\text{op}} \rightarrow \Omega \in F' \iff \exists q \in F, f(*, q) = \top$, where $*$ is the unique element of $\mathbf{1}$. $\square$

As we have seen in the discussion so far, the following implication relations are easy to prove.

$$\begin{aligned} (\text{FinSet}, 1)\text{-regularity} &\iff \text{recognition by DFA} \\ &\implies \text{recognition by NFA} \\ &\implies (\text{Fin}\Omega\text{Prof}, 1)\text{-regularity} \end{aligned}$$

The subtle difference between the definitions of recognition by NFA and the definition of $(\text{Fin}\Omega\text{Prof}, 1)$ -regularity makes the inverse of the last implication non-trivial, but this is not a serious problem. This is because what we are about to show is that the seemingly strongest condition of $(\text{Fin}\Omega\text{Prof}, 1)$ -regularity is equivalent to the seemingly weakest condition of $(\text{FinSet}, 1)$ -regularity!

Theorem 6.2 (Equivalence of NFA and DFA).

$$\text{RegLan}(\text{Fin}\Omega\text{Prof}, \mathbf{1}) = \text{RegLan}(\text{FinSet}, 1)$$

Proof. Apply Corollary 5.9 to the adjunction of Proposition 3:

$$\text{FinSet} \begin{array}{c} \xrightarrow{I} \\ \perp \\ \xleftarrow{\text{Fin}\Omega\text{Prof}(\mathbf{1}, -)} \end{array} \text{Fin}\Omega\text{Prof}$$

$\square$

This proof by adjunction is a rediscovery of the classical proof of the equivalence of NFA and DFA. In fact, the conversion of DA by functor $I : \text{FinSet} \rightarrow \text{Fin}\Omega\text{Prof}$ is the standard way to consider DFA as NFA. Furthermore, the conversion of DA by functor $\text{Fin}\Omega\text{Prof}(\mathbf{1}, -) : \text{Fin}\Omega\text{Prof} \rightarrow \text{FinSet}$ transfers NFA Q to DFA on the underlying set $\text{Fin}\Omega\text{Prof}(\mathbf{1}, Q) \cong \hat{Q}$. This is a Rabin-Scott powerset construction!

Remark 6.3. The non-trivial part of Theorem 6.2 is the inclusion relation $\text{RegLan}(\text{Fin}\Omega\text{Prof}, \mathbf{1}) \subset \text{RegLan}(\text{FinSet}, 1)$ and actually this is a special case of more general result, namely Lemma 5.6, since $\text{Fin}\Omega\text{Prof}$ is locally small. However, what is actually done by Lemma 5.6 is the same as Theorem 6.2, which is the conversion of DA by $\text{Fin}\Omega\text{Prof}(\mathbf{1}, -) : \text{Fin}\Omega\text{Prof} \rightarrow \text{FinSet}$.