Equational logic for names and binders

Ranald Clouston

University of Cambridge
Computer Laboratory
Churchill College

June 2009

This dissertation is submitted for the degree of Doctor of Philosophy
Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except where specifically indicated in the text.

This dissertation does not exceed the regulation length of 60000 words, including tables and footnotes.
Equational logic for names and binders

Ranald Clouston

Summary

Equational logic is a fundamental topic in mathematics with many applications throughout computer science. However, many basic logical systems cannot be described equationally. Rather, they require equations that are modulated by side conditions involving the freshness of names. The Gabbay-Pitts FM-sets model, and the related concept of nominal set, provides a context in which names, freshness and $\alpha$-conversion can be treated in a manner that is entirely formal while still close to ‘pen and paper’ practice.

This dissertation’s key contribution is to develop the system of Nominal Equational Logic (NEL), in which the intended interpretation of terms lie within FM-sets rather than the sets of standard equational logic. NEL is developed by analogy with standard equational logic, with notions of signature, structure, term, theory and algebra introduced. Equality and freshness judgements over the terms of NEL are made in the presence of freshness contexts, which capture the notion of side conditions asserting the freshness of names. A sound and complete set of proof rules is provided, along with an alternative set of proof rules for which freshness assertions are only made on variables within the freshness context, rather than on terms as first class judgements. These two notions of judgement are shown to be equally expressive.

The key concept of name binding, without which $\alpha$-conversion is trivial, is not explicitly supported by the signatures of NEL. However, various notions of binding can be defined through the axioms of a theory of the logic. Once this fact is established we provide theories for various logical systems that involve names, such as the untyped $\lambda$-calculus, Fiore-Staton nominal substitution and concretion.

The analogy between Nominal Equational Logic and standard equational logic is exploited by presenting notions of nominal universal algebra and nominal categorial algebra. In the former case, we prove that FM-sets induce free algebras, and that the free functor is monadic. In the latter case, we provide an analogue of Lawvere’s result identifying equational theories with small categories equipped with finite products, introducing the extra categorial structure needed to define nominal Lawvere theories.
Acknowledgments

I would like to thank Nigel Evans and the Woolf Fisher Trust, whose funding and support made this research possible.

Thank you to Andy Pitts, whose supervision and advice have been invaluable throughout my time in Cambridge.

Most of all, thank you so much to Francesca. I cannot imagine how I could have done all this without you by my side.
The glory and the nothing of a name.

Lord Byron, *Churchill’s Grave* (1816).
# Contents

1 Introduction .................................................................................................................. 8
   1.1 Names, freshness and binding .................................................................................... 8
   1.2 FM-sets and nominal sets ......................................................................................... 10
   1.3 From equational logic to Nominal Equational Logic .................................................. 11
   1.4 Overview .................................................................................................................. 12

2 Nominal sets and FM-sets .............................................................................................. 14
   2.1 Atoms and permutations ......................................................................................... 14
   2.2 Nominal sets ........................................................................................................... 15
   2.3 Freshness ............................................................................................................... 17
   2.4 The category of nominal sets .................................................................................. 19
   2.5 FM-Sets ............................................................................................................... 21

3 Nominal Equational Logic .............................................................................................. 23
   3.1 Signatures and structures ....................................................................................... 23
   3.2 Terms and values .................................................................................................... 25
   3.3 Substitution ............................................................................................................ 28
   3.4 Theories and algebras ............................................................................................. 30

4 Binding ........................................................................................................................... 34
   4.1 Abstraction sets ....................................................................................................... 34
   4.2 NEL with binders ..................................................................................................... 37
   4.3 Translation to NEL without binders ....................................................................... 38
   4.4 Other notions of binding ......................................................................................... 43
   4.5 Concretion .............................................................................................................. 46
## 5 Proof theory 51
- 5.1 Proof rules for Nominal Equational Logic 51
- 5.2 Equivariance 55
- 5.3 Ground term algebras 56
- 5.4 Completeness 58
- 5.5 Proof rules for equality 64

## 6 Universal algebra 69
- 6.1 The category of $T$-algebras 69
- 6.2 Free $T$-algebras 70
- 6.3 Monadicity 73

## 7 Nominal Lawvere theories 76
- 7.1 Internal permutation actions 76
- 7.2 FM-categories 79
- 7.3 Categorial algebra 83
- 7.4 The classifying category 88
- 7.5 The category-theory correspondence 90

## 8 Conclusions 94
- 8.1 Summary of contributions 94
- 8.2 Related work – Nominal Algebra 96
- 8.3 Related work – equational systems 97
- 8.4 Related and further work – nominal rewriting 98
- 8.5 Further work – universal and categorial algebra 98
- 8.6 Further work – partial equational logic 98
- 8.7 Further work – order-sorted algebra 99
Chapter 1

Introduction

1.1 Names, freshness and binding

Many formal systems in computer science and logic, including many that are fundamental and well known, make use of variables in their object language, and have some notion of a formula in that language depending on a finite set of such variables. In the terminology of this dissertation, this finite set of variables is called the support of the formula. This notion of support can then be used to state the laws of the system, by requiring that variables that are used are not in the support of certain formulae. In this dissertation, such variables will be said to be fresh for that formula.

For example, it is a standard axiom of first order logic that

\[ \Phi \supset (\forall a. \Psi) = \forall a. (\Phi \supset \Psi) \quad \text{if } a \text{ does not appear free in } \Phi . \]  

(1.1)

Here the symbols \( \Phi, \Psi \) are meta-level variables, ranging over the formulae of first order logic. The symbol \( a \), on the other hand, is not a meta-level construct, but rather a variable in the object language. To avoid confusion between these different types of variable, we will call variables such as \( a \) names.

Further examples of this style of rule, with an equality assertion modulated by a freshness assertion, are ubiquitous in computer science and logic. For example, \( \eta \)-equivalence in the \( \lambda \)-calculus:

\[ \lambda \xi \cdot f \xi =_\xi f \quad \text{if } a \text{ does not appear free in } f , \]

(1.2)

the name restriction construct in the \( \pi \)-calculus [40]:

\[ (\nu a \cdot x) \mid y = (\nu a \cdot (x \mid y)) \quad \text{if } a \text{ is not in the free names of } y \]

(1.3)

and so forth.

It is the presence of binding operators that makes these side conditions non-trivial. In the absence of such operators, the support of a formula would merely be the names that it contains.
For example, if $P$ is a unary predicate symbol of first order logic then the formula $P \, x$ is supported by the set of names $\{x\}$. This means that the formula $P \, x$ can be understood to be making a logical statement about a specific name $x$. However, where a name is bound by an operator, that name is then understood to be anonymous placeholder, so the formula $\forall x. P \, x$ is supported by the empty set $\emptyset$. This formula is not making a statement about any specific name, despite the presence of the specific name $x$ in its concrete syntax.

The most fundamental property that arises in the presence of binding operators is $\alpha$-equivalence. Informally, two formulae of a logical system are equivalent if they differ only in their bound names in a uniform manner. For example, the formulae $\forall x. P \, x$ and $\forall y. P \, y$ are $\alpha$-equivalent.

These issues of support, binding and $\alpha$-equivalence may seem simple enough in the examples provided thus far. Indeed, in pen-and-paper mathematics, one can usually navigate past these issues without problems by carefully choosing one’s names so that they do not clash. But working with these concepts in a completely formal manner, as may be necessary for example if we are to mechanise any of these standard logical systems, has proved to be surprisingly fraught. A variety of different approaches have been suggested to deal with these issues.

The first major approach is to do with away with bound names entirely; the most notable approach in this school are de Bruijn indices [13]. After all, if the formula $\forall x. P \, x$ has no names in its support, why should it mention a specific name $x$? This is all very well for the purposes of mechanisation, but formulae of any complexity written with de Bruijn indices, let alone proofs involving them, rapidly become extremely difficult for humans to read and understand. Essentially, the reason that the formula $\forall x. P \, x$ mentions a specific name $x$ is because this is the way that people prefer to reason. In some circumstances this may be an irrelevant consideration, but in others, such as a machine-assisted proof that a user is developing, cleaving more closely to informal practice is desirable. [5] has a further discussion of these issues.

The second major approach is higher order abstract syntax [42], where questions of binding and $\alpha$-equivalence are dealt with at a meta-level rather than as first class aspects of the logic in question. This approach also sacrifices the ability to manipulate bound names directly in a proof, and thus also departs from pen-and-paper practice [8, Section 2.3], though the extent to which these deviations are problematic remains a subject of debate [11].

More broadly, the question arises of whether it is possible to formalise names in a rigorous and natural way that does not involve treating them as second class citizens, or banishing them entirely, but rather affords them the same first class status they enjoy in pen-and-paper mathematics. Not only would such an approach be useful for applications where closeness to informal practice is desired, but it would also allow the ubiquitous concepts of names and binding to be studied as subjects of interest in their own right. Such an approach is discussed in the next section.
1.2 FM-sets and nominal sets

Pitts and Gabbay introduced \[24, 25\] a model for names, based on a notion of nonstandard set theory, called \textit{Fraenkel-Mostowski sets}, or \textit{FM-sets}. This model starts with a countably infinite set of \textit{atoms}, which are used to represent names. The notions of freshness, binding and $\alpha$-equivalence that were discussed in the previous section can then by defined by the action on a set of a finite permutation of these atoms. Given such a permutation $\pi$ and an element $x$ of the model, we can apply $\pi$ to $x$ to get a new element $\pi \cdot x$. Intuitively, this corresponds to permuting the names of a given formula, making no distinction between bound and free names as we go. A pair of atoms is said to be fresh for an element of the FM-sets model if applying their swapping to the element leaves it unchanged.

For example, in the last section we asserted that the first order formula $P \, x$ has support $\{x\}$, while $\forall x. \, P \, x$ has support $\emptyset$. Choosing some other atom $y$ and applying the swapping $(x \ y)$ to these formulae gives us

\[
(x \ y) \cdot (P \, x) = P \, y \neq_\alpha P \, x
\]
\[
(x \ y) \cdot (\forall x. \, P \, x) = \forall y. \, P \, y =_\alpha \forall x. \, P \, x.
\]

\hfill (1.4)

So $x$ is in the support set of $P \, x$ but not in the support of $\forall x. \, P \, x$.

The FM-sets model was subsequently refined to the \textit{nominal sets} model of \[44\]. In fact, this concept is logically prior to the notion of FM-set, and as such was implicitly present in the original papers on FM-sets. Indeed, some early papers on FM-sets \[47, \text{Definition A.2}\] simply defined FM-sets as what are now known as nominal sets. Intuitively, nominal sets differ from FM-sets in that they are closed under their permutation action. That is, applying a permutation $\pi$ to an element $x$ of a nominal set $X$ will return an element $\pi \cdot x$ of that same nominal set $X$. On the other hand, in the FM-sets model the permutation action on elements extends to a permutation action on the FM-sets themselves, so that if $x \in X$ then $\pi \cdot x$ is a member of an FM-set $\pi \cdot X$, which may not be equal to the original FM-set $X$. Not only are the elements of the FM-sets model supported by finite sets of atoms, but so are the FM-sets themselves.

The move from FM-sets to nominal sets represents a useful shift for three reasons. First, the definition of nominal sets does not require any nonstandard set theory, which makes it clearer to present and digest, though the set theory required to use FM-sets turns out not to be particularly difficult. Second, having to worry about which set an element $\pi \cdot x$ is a member of in the FM-sets model does create some overhead in terms of proof checking and side conditions on rules. Third, the category of nominal sets is equivalent to a well known sheaf category, the Schanuel topos \[25, \text{Section 7}\]. However, while nominal sets are expressive enough for a great many purposes, the work in this dissertation will require the ability to construct sets with non-trivial permutation actions defined upon them. As such the bulk of this thesis will concern itself with the original FM-sets model.

Working with names in the FM-sets and nominal sets models turns out to be both computationally reasonable and close to intuitive practice, as rules such as (1.1), (1.2) and (1.3) can be
represented formally in a manner very similar to informal practice, while proofs involving such rules closely resemble pen-and-paper proofs. It also provides a setting in which names are first class entities and can be studied in their own right. This has led to the FM-sets and nominal sets models being applied in a variety of settings, including but not limited to functional programming [56, 34], logic programming [9], unification [61], rewriting [15], type theory [53], games [1, 58] and automated reasoning [59, 60].

1.3 From equational logic to Nominal Equational Logic

Equational logic is a simple system that forms the foundation of a vast array of mathematical work in universal algebra and categorial logic. Many deep results in mathematics fit within this area, from Birkhoff’s HSP theorem [6] to the relationship of equational theories and monads [36] to Lawvere’s correspondence between equational theories and small categories with finite products [35]. Some of the history of these developments is surveyed in [31].

Equational logic, and the mathematics that has been built up around it, also forms the basis of many computer science applications, including formal specification languages [62] and programming languages [27]. But as we discussed in Section 1.1, the mere equality of logical terms is not expressive enough for many standard logical systems in computer science. A large number of such systems can, however, be adequately described by equations modulated by assertions concerning the freshness of names. This is in some senses a modest extension of standard equational logic, and this provides the motivation to investigate a shift from equational logic to the system we will call Nominal Equational Logic, where terms and theories are interpreted within FM-sets.

The central question of this dissertation can therefore be summed up in the following sentence:

*Can the standard developments of equational logic be recreated in an FM-sets setting?*

At each stage we will endeavour to retain a close analogy between equational logic and Nominal Equational Logic, so that we may where possible apply standard techniques to the questions that arise. Some results, such as the development of nominal universal algebra in Chapter 6, cleave surprisingly closely to this standard account. Others, such as the Completeness Theorem of Chapter 5, depart from standard practice in ways that produce interesting insights into the nature of names and freshness.

The most closely related work to this generalisation of equational logic to names is the Nominal Algebra of [23], which is also based on the nominal sets model. An early version of Nominal Equational Logic has also been shown to be a special case of a very general notion of equational system in [17]. Both of these works will be discussed in the concluding Chapter 8.
1.4 Overview

Chapter 2 introduces the basic definitions of the thesis, in particular those of nominal set, freshness and FM-set. The results of this chapter are standard in the literature on nominal sets, and most can be found in [25], [44] or [45]. This chapter provides an adequate grounding in these concepts for the remainder of the dissertation.

Chapter 3 introduces the signatures and terms of Nominal Equational Logic (NEL), along with the semantics describing how these terms are to be interpreted within the FM-sets model. This development culminates in the definition of NEL-theory, and of algebra for such theories.

The concept of NEL-signature does not include explicit support for binding operators, unlike most other systems designed to deal with binding [18, 61, 48]. Chapter 4 describes how binding structure can in fact be defined by a NEL-theory, and so need not be built in to the logic at a lower level. Because binding structure is not elementary in the logic the theoretical development of NEL need not refer to such structure, and as such the remainder of the dissertation can be understood without reference to this chapter.

Chapter 5 introduces proof rules for freshness and equality assertions, and shows that they are sound and complete for the FM-set semantics of Chapter 3. In the final section of this chapter it is proved that freshness can be defined in terms of equality modulo a freshness environment, and therefore sound and complete proof rules can be given for the equally expressive, though less convenient, NEL with equality only.

Chapter 6 provides a short introduction to universal algebra based on NEL. It shows that free algebras exist and that the forgetful functor is monadic, so that algebras for a theory correspond to algebras for the monad generated by that theory.

Chapter 7 generalises the results of the previous chapters from the category of FM-sets to FM-categories, isolating the categorial structure needed in the category of FM-sets. This involves an internal permutation action representing the action of permutations of names, finite products that are equivariant with respect to the internal permutation action, and fresh-inclusions allowing discussion of freshness in a categorial context. This general setting produces a new, and very simple, proof of completeness and a result analogous to the standard account of Lawvere theories [35], identifying NEL-theories with small FM-categories.

Chapter 8 provides concluding remarks, looking at related work and future research directions that are suggested by the results of this dissertation.

An early version of the results of Chapter 3 and 5, apart from Section 5.5, has been published previously as [10], co-written with Andrew Pitts. The results of [10] differ from the results of these chapters, in that the earlier work was set within the nominal sets model, while this dissertation is largely set within the FM-sets model. This allows us to generalise the notion of NEL-signature to produce a nominal set of sorts, rather than a set of sorts. That is, in [10] sorts
have empty support, while in this dissertation applying a permutation action to them may move
them around. This generalisation simplifies the categorial results of Chapter 7, as is discussed
in Remark 7.5.4, while the practical benefits that may be opened up are discussed in Section
8.7.
Chapter 2

Nominal sets and FM-sets

This chapter will introduce the basic concepts of this thesis, in particular those of nominal set and FM-set. The results of this chapter are standard and largely well known aspects of the nominal sets and FM-sets models [25, 44, 45]. Note, however, that while we define the notion of nominal set in the standard way in Definition 2.2.1, we will redefine it for the remainder of the dissertation in Definition 2.5.4 so that we may work entirely within the FM-sets model.

2.1 Atoms and permutations

We start by fixing a countably infinite set \( \mathbb{A} \), which will capture the notion of names, or object level variables, in applications of Nominal Equational Logic. We will call the members of \( \mathbb{A} \) atoms, in keeping with the origins of nominal sets in models of Zermelo-Fraenkel set theory with atoms [32, Section 6].

The set \( \text{Perm} \) of (finite) permutations of atoms consists of all bijections \( \pi : \mathbb{A} \to \mathbb{A} \) such that

\[
\text{supp}(\pi) \triangleq \{ a \in \mathbb{A} \mid \pi(a) \neq a \}
\]

is finite. \( \text{Perm} \) has group structure by taking group multiplication as functional composition, \( \pi' \pi(a) = \pi'(\pi(a)) \), and the identity as the permutation \( \iota \), where \( \iota(a) = a \) for all \( a \). Inverses are defined because permutations are bijections. \( \text{Perm} \) is generated by transpositions \((a \ a')\) that map \( a \) to \( a' \), \( a' \) to \( a \) and leave all other atoms fixed.

**Example 2.1.1.** Given an integer \( n \), \( \mathbb{A}^n \) is defined as usual as the set of \( n \)-tuples of atoms. Define \( \mathbb{A}^{(n)} \) to be the set of distinct \( n \)-tuples:

\[
\mathbb{A}^{(n)} \triangleq \{ (a_1, \ldots, a_n) \mid a_i \neq a_j \text{ for all } 1 \leq i < j \leq n \}.
\]

Then, given \( \vec{a} = (a_1, \ldots, a_n) \), \( \vec{a}' = (a'_1, \ldots, a'_n) \in \mathbb{A}^{(n)} \) we define the generalised transposition \( (\vec{a} \ \vec{a}') \in \text{Perm} \) by
(i) $\langle \vec{a} \vec{a}' \rangle(a) \triangleq a$ if $a \notin \{a_1, \ldots, a_n, a'_1, \ldots, a'_n\}$;

(ii) $\langle \vec{a} \vec{a}' \rangle(a_i) \triangleq a'_i$ for $1 \leq i \leq n$;

(iii) Where $(b_1, \ldots, b_m)$ is the sublist of $\vec{a}$ of atoms not in $\{a'_1, \ldots, a'_n\}$, and $(b'_1, \ldots, b'_m)$ is the sublist of $\vec{a}'$ of atoms not in $\{a_1, \ldots, a_n\}$, which has the same length, $\langle \vec{a} \vec{a}' \rangle(b'_i) \triangleq b_i$ for $1 \leq i \leq m$.

Where the underlying sets of $\vec{a}, \vec{a}'$ are disjoint it is the case that
$$\langle \vec{a} \vec{a}' \rangle = \langle \vec{a}' \vec{a} \rangle = \langle a_1 a'_1 \rangle \cdots \langle a_n a'_n \rangle .$$ (2.3)

Remark 2.1.2. In the literature on nominal sets (e.g. [61]) $\mathbb{A}$ is often divided into infinitely many sets, each infinite themselves, representing the atom sorts. Perm is then restricted to the finite permutations of atoms that are also sort-respecting, so that for any $\pi \in \text{Perm}$, $\pi(a)$ has the same atom sort as $a$.

For some work [41] this atom sorting is central to the theoretical development; however, in our case while this addition can be useful for applications (as in Example 3.1.3), it adds no particular difficulty to the theoretical developments of this dissertation, and as such we will consider only the case where there is a single sort of atoms.

As usual, a Perm-set is a set $X$ equipped with a permutation action $\text{Perm} \times X \to X$ mapping $(\pi, x) \mapsto \pi \cdot_X x$ so that
$$\nu \cdot_X x = x \quad \pi' \cdot (\pi \cdot_X x) = \pi'\pi \cdot_X x .$$ (2.4)

Where the set $X$ is clear will write $\pi \cdot_X x$ as $\pi \cdot x$.

Our next definition defines the notion of a homomorphism of Perm-sets.

Definition 2.1.3. A function $f$ between Perm-sets $X \to Y$ is equivariant if, for all $\pi \in \text{Perm}$ and $x \in X$,
$$\pi \cdot (f(x)) = f(\pi \cdot x) .$$

2.2 Nominal sets

Given a Perm-set $X$, a set of atoms $\vec{\pi} \subseteq \mathbb{A}$ is said to support $x \in X$ if for all $a, a' \notin \vec{\pi}$, $(aa') \cdot x = x$. $x$ is finitely supported if there is a finite set $\vec{\pi}$ supporting $x$.

Definition 2.2.1. A Perm-set $X$ is a nominal set if every $x \in X$ is finitely supported.

Remark 2.2.2. Note that we will have cause to redefine the notion of nominal set in Definition 2.5.4; in particular, we will usually only be interested in a certain subclass of all nominal sets. For the moment, however, it is convenient to develop the theory of nominal sets for this more general definition; all the results we will prove for this definition can be easily seen to hold for the later redefinition.
The notion of finite support is central to the Gabbay-Pitts model of permutative renaming; the following is a lemma that is essential to making this model work.

**Lemma 2.2.3.** [25, Proposition 3.4] Given a Perm-set $X$, if $x \in X$ is finitely supported then there is a least finite support of $x$. Call this unique set the support of $x$ and write it $\text{supp}(x)$.

**Proof.** We need only show that finite sets that support $x$ are closed under intersection. If $\overline{\pi}$ and $\overline{\pi}'$ support $x$, we need to show that $(a a') \cdot x = x$ for any $a, a' \not\in \overline{\pi} \cup \overline{\pi}'$. Take some $b \not\in a \cup a'$. $(a a') = (a b)(a' b)(a b)$, but $(a b)(a' b)(a b) \cdot x = x$ because $a$ and $b$ are either both not in the supporting set $\overline{\pi}$ or both not in the supporting set $\overline{\pi}'$, and likewise for $a'$ and $b$. \hfill $\square$

**Example 2.2.4.** Any set $X$ is a nominal set under the trivial permutation action $\pi \cdot x \triangleq x$ for all $x \in X$; $\text{supp}(x) = \emptyset$. Two such sets we will use in particular in this dissertation are the empty set $\emptyset$ and the singleton $1 = \{\ast\}$.

**Example 2.2.5.** The set of atoms $\mathbb{A}$ is a nominal set under the action $\pi \cdot a = \pi(a)$; $\text{supp}(a) = \{a\}$. Similarly, $\mathbb{A}^n$ and $\mathbb{A}^{(n)}$ (Example 2.1.1) are nominal sets under the term-wise action

$$\pi \cdot (a_1, \ldots, a_n) \triangleq (\pi(a_1), \ldots, \pi(a_n))$$

(2.5)

and so $\text{supp}(a_1, \ldots, a_n) = \{a_1, \ldots, a_n\}$ in both cases.

**Example 2.2.6.** Perm is a nominal set under the *conjugation* action

$$\pi \cdot \pi' \triangleq \pi \pi' \pi^{-1}.$$ 

(2.6)

This definition accords with our intuition of permutation actions as renamings; if $\pi'$ maps $a \mapsto b$ then $\pi \cdot \pi'$ maps $\pi(a) \mapsto \pi(b)$. We can see that $\text{supp}(\pi)$ is as (2.1). There is another, perhaps more obvious, notion of a permutation action on a permutation, that of *left multiplication*:

$$\left(\pi, \pi'\right) \mapsto \pi \pi'.$$ 

(2.7)

But while this defines a permutation action on Perm, permutations are not finitely supported under this action, so we use (2.6) to make Perm a nominal set.

**Example 2.2.7.** Given nominal sets $X_0, X_1$ we can form their disjoint union $X_0 + X_1$ in the normal manner via injections $i_n : X_i \to X_0 + X_1$ mapping $x \mapsto (x, i)$ for $x \in X_i$ and $i = 0, 1$. The permutation action is $\pi \cdot X_0 + X_1 (x, i) = (\pi \cdot X, x, i)$.

**Example 2.2.8.** Given a nominal set $X$ we can define a permutation action on $\mathcal{P}(X)$ by

$$\pi \cdot S \triangleq \{\pi \cdot X \mid x \in X\}.$$ 

(2.8)

But $\mathcal{P}(X)$ is not necessarily a nominal set under this action; for example, a subset $\overline{\pi} \subseteq \mathbb{A}$ is finitely supported if and only if it is finite or cofinite (so that $\mathbb{A} - \overline{\pi}$ is finite). We can, however, define the *finitely supported powerset*

$$\mathcal{P}_{fs}(X) \triangleq \{S \subseteq X \mid S \text{ is finitely supported with respect to (2.8)}\}.$$ 

(2.9)
We call such an \( S \in \mathcal{P}_f(\mathcal{X}) \) a \textit{finitely supported subset of} \( \mathcal{X} \). If such an \( S \) has empty support then it is itself a nominal set, with \( \pi \cdot_S x = \pi \cdot x \) for each \( x \in S \), and we call \( S \) a \textit{nominal subset of} \( \mathcal{X} \). For example, the finite powerset \( \mathcal{P}_f(\mathcal{A}) \) is a nominal subset of \( \mathcal{P}_f(\mathcal{A}) \).

If \( \mathcal{X} \) is a family of nominal sets linearly ordered by this nominal subset relation then its union is also a nominal set; each \( x \in \bigcup \mathcal{X} \) has permutation action and support defined as for any \( X \in \bigcup \mathcal{X} \) containing \( x \).

### 2.3 Freshness

This section will define what it means for an atom to be fresh for an element of a nominal set and explore some of the properties of freshness. First, we will note a theorem that will help prove some of these properties.

**Theorem 2.3.1.** (Some/Any Theorem [45, Theorem 3.8]) Let \( A \in \mathcal{P}_f(\mathcal{A}) \) be supported by some finite set of atoms \( \overline{a} \). Then the following are equivalent:

\[
\forall a \in \mathcal{A}. (a \notin \overline{a} \implies a \in A) ; \quad \forall a \in \mathcal{A}. (a \notin \overline{a} \land a \in A) .
\]

**Proof.** \( \mathcal{A} - \overline{a} \) is infinite and therefore non-empty, so (2.10) implies (2.11). Conversely, take \( a \notin \overline{a} \) such that \( a \in A \). Then for any other \( a' \notin \overline{a} \), \( (a a') \cdot A = A \) because \( \overline{a} \) supports \( A \). Therefore \( a' = (a a') \cdot a \in (a a') \cdot A = A \).

**Definition 2.3.2.** If \( a \notin \text{supp}(x) \) then we say that \( a \) is \textit{fresh for} \( x \) and write \( a \# x \). If \( a \# x \) for all \( a \) in some set of atoms \( \overline{a} \) then we write \( \overline{a} \# x \).

The next lemma demonstrates that a property that holds for some atom fresh for that property in fact holds for any such atom. This \textit{Some/Any Theorem} will assist in proving some of the basic properties of freshness.

**Lemma 2.3.3.** Given an element \( x \) of a nominal set \( X \), \( a' \# x \) if and only if

\[
\exists a. (a \neq a' \land a \# x \land (a a') \cdot x = x) .
\]

**Proof.** Set \( A = \{ a \mid (a a') \cdot x = x \} \) and \( \overline{a} = \text{supp}(x) \cup \{ a' \} \) in Theorem 2.3.1.

**Lemma 2.3.4.** [45, Lemma 3.7] Given a nominal set \( X \), \( x \in X \), \( \pi \in \mathcal{P}_f(\mathcal{A}) \) and \( \pi \in \text{Perm} \),

\[
\overline{a} \# x \implies \pi \cdot \overline{a} \# \pi \cdot x .
\]

**Proof.** Take any \( a \in \overline{a} \# x \) and, following Lemma 2.3.3, take some other atom \( a' \# (a, \pi, x, \pi \cdot x) \). Then \( (\pi(a) a') \pi \cdot x = \pi(a a') \cdot x \), which is \( \pi \cdot x \) as \( a, a' \# x \).
An immediate corollary of this lemma is that
\[ \pi \cdot \text{supp}(x) = \text{supp}(\pi \cdot x) \, . \] (2.12)

This allows us to generalise Lemma 2.3.3 to a general test for freshness on a set of atoms, using the generalised transpositions of Example 2.1.1.

**Lemma 2.3.5.** Let \( \bar{a} \in \mathbb{A}^{(n)} \) be a tuple of atoms, and \( x \) be an element of a nominal set \( X \). Then the following are equivalent:

(i) \( \text{supp}(\bar{a}) \not\# x \);

(ii) \( (\bar{a} \bar{a}') \cdot x = x \), where \( \bar{a}' \in \mathbb{A}^{(n)} \) is a tuple of the same size as \( \bar{a} \) such that \( \text{supp}(\bar{a}') \not\# (\bar{a}, x) \).

**Proof.** (i) \( \implies \) (ii): \( \text{supp}(\bar{a}) \) and \( \text{supp}(\bar{a}') \) are disjoint, so by (2.3) \( (\bar{a} \bar{a}') \) is a series of transpositions of atoms fresh for \( x \).

(ii) \( \implies \) (i): Apply \( (\bar{a} \bar{a}') \) and Lemma 2.3.4 to \( \text{supp}(\bar{a}') \not\# x \) to get \( \text{supp}(\bar{a}) \not\# (\bar{a} \bar{a}') \cdot x = x \).

Further, we have two results linking permutations in general with the freshness relation:

**Lemma 2.3.6.** Take some nominal set \( X \), element \( x \in X \) and permutation \( \pi \in \text{Perm} \). Then
\[ \text{supp}(\pi) \not\# x \implies \pi \cdot x = x \, . \]

**Proof.** We prove by induction on the size of \( \text{supp}(\pi) \) (for all \( \pi \) simultaneously); If \( \text{dom}(\pi) = \emptyset \), \( \pi = \iota \) so the result follows by (2.4).

So suppose \( \text{dom}(\pi) \) is non-empty and pick some \( a \in \text{supp}(\pi) \). It is simple to confirm that \( \text{supp}((\pi(a) a) \pi) \subseteq \text{supp}(\pi) - \{a\} \), so \( \text{supp}((\pi(a) a) \pi) \) is strictly smaller than \( \text{supp}(\pi) \) and therefore by induction we have
\[ (\pi(a) a) \pi \cdot x = x \, . \] (2.13)

Apply \( (\pi(a) a) \) to each side to get \( \pi \cdot x = (\pi(a) a) \cdot x \). But \( a, \pi(a) \in \text{supp}(\pi) \), so \( a, \pi(a) \not\# x \) and therefore \( (\pi(a) a) \cdot x = x \) by Lemma 2.3.3.

**Corollary 2.3.7.** Given \( \pi, \pi' \in \text{Perm} \), define the disagreement set as the finite set of atoms
\[ ds(\pi, \pi') \triangleq \{ a \in \mathbb{A} \mid \pi(a) \neq \pi'(a) \} \, . \] (2.14)

Then
\[ ds(\pi, \pi') \not\# x \implies \pi \cdot x = \pi' \cdot x \, . \]

**Proof.** \( ds(\pi, \pi') = \text{supp}(\pi^{-1} \pi') \), so apply Lemma 2.3.6, then apply \( \pi \) to both sides.

Finally, the results of this chapter allow us to define another important example of nominal sets.
Example 2.3.8. Given nominal sets $X, Y$ there are two non-trivial notions of a tensor product between them. One is the standard Cartesian product $X \times Y$, and the other is the *separated tensor* $X \otimes Y \triangleq \{ (x, y) \in X \times Y \mid \text{supp}(x) \cap \text{supp}(y) = \emptyset \}$.

(2.15) The permutation action on both $X \times Y$ and $X \otimes Y$ is $\pi \cdot (x, y) = (\pi \cdot x, \pi \cdot y)$. This action preserves membership of $X \otimes Y$ by (2.12), so $X \otimes Y$ is a nominal subset of $X \times Y$. The unit for both tensor products is the singleton nominal set $1$ (Example 2.2.4).

2.4 The category of nominal sets

Definition 2.4.1. Let $\mathcal{Nom}$ be the category whose objects are nominal sets (Definition 2.2.1) and arrows are equivariant functions (Definition 2.1.3) between them.

It follows from Definition 2.1.3 that if we have an equivariant function between nominal sets $f : X \rightarrow Y$, and any $x \in X$ and $a \in \mathcal{A}$,

$$a \# x \implies a \# f(x).$$

(2.16) Now the terminal object in $\mathcal{Nom}$ is the singleton $1 = \{ * \}$. Its element $*$ has empty support in $1$, so by (2.16) the global elements of a nominal set $X$ that may be picked out by an equivariant function $1 \rightarrow X$ can only include the emptily supported members of $X$. Therefore $\mathcal{Nom}$ is not *well-pointed*, as a parallel pair of arrows $f, g : X \rightarrow Y$ may be unequal even if they have equal composition with all global elements $1 \rightarrow X$.

Example 2.4.2. We have already seen several examples of equivariant functions between nominal sets:

(i) The generalised transpositions of Example 2.1.1 arise from an equivariant function $\mathcal{A}^{(n)} \times \mathcal{A}^{(n)} \rightarrow \text{Perm}$, as it is not hard to see that

$$ (\pi \cdot \bar{a} \cdot \bar{a'}) = \pi(\bar{a} \cdot \bar{a'}) = \pi \cdot (\bar{a} \cdot \bar{a'}). $$

(2.17)

(ii) The permutation action $\text{Perm} \times X \rightarrow X$ on any nominal set $X$ is equivariant as $(\pi \cdot \pi') \cdot (\pi \cdot x) = \pi \cdot \pi' \cdot \pi^{-1} \cdot x = \pi \cdot (\pi' \cdot x)$.

(iii) The function $\text{supp} : X \rightarrow \mathcal{P}_{\text{fin}}(\mathcal{A})$ on any nominal set $X$ is equivariant by (2.12).

The next lemma shows another way in which $\mathcal{Nom}$ differs from the category of sets $\text{Set}$; the exponential object of two objects is not the same as the set of morphisms between them. The lemma also introduces the important concept of *finitely supported function*.

Lemma 2.4.3. [45, Section 3.2] $\mathcal{Nom}$ is Cartesian closed.
Proof. Finite products are defined as with $\text{Set}$ (Example 2.3.8). Given nominal sets $X, Y$, the exponential object $Y^X$ is defined to be the set of all functions $f : X \to Y$ finitely supported with respect to the permutation action

$$(\pi \cdot_X f)(x) \triangleq \pi \cdot_Y f(\pi^{-1} \cdot_X x) .$$

(2.18)

Then we define the evaluation function $ev : Y^X \times X \to Y$ as usual as $ev(f, x) = f(x)$. Checking the equivariance of $ev$ and that the exponential object has the universal property in $\text{Nom}$ is routine.

It follows immediately from (2.18) that for finitely supported $f : X \to Y$, $x \in X$ and $\pi \in \text{Perm}$,

$$\pi \cdot (f(x)) = (\pi \cdot f)(\pi \cdot x) .$$

(2.19)

It follows from this and Definition 2.1.3 that the equivariant functions are those that have empty support with respect to the action (2.18). In other words, the global elements of $Y^X$ are precisely the equivariant functions $X \to Y$.

Example 2.4.4. An example of a function that is finitely supported for the action (2.18), but not emptily supported and hence equivariant, is the constant function $X \to A$ from any nominal set $X$ mapping all $x \in X$ to a given atom $a$. This function is supported by $\{a\}$.

Indeed, a function between nominal sets need not even be finitely supported; we now provide an interesting example of a function that cannot have finite support.

Lemma 2.4.5. [20, Theorem 11.4.1] There is no finitely supported choice function on $\mathcal{P}_{fs}(A)$; that is, there is no function $c : \mathcal{P}_{fs}(A) \to A$ such that $c(A) \in A$ for all non-empty and finitely supported $A \subseteq A$.

Proof. Suppose $c$ is finitely supported. If $c(A - \text{supp}(c)) = a$ then $a \not\equiv c$ by definition. Take any $a' \neq a$ also fresh for $c$. Then $a' = (a a') \cdot a = (a a') \cdot (c(A - \text{supp}(c)))$, which equals $c(A - \text{supp}(c))$ by (2.19) and the fact that $a, a' \not\equiv c$. But this equals $a$, contradicting the assumption that $a$ and $a'$ are different atoms.

It is a fact that, as discussed in [25, Section 7], $\text{Nom}$ is equivalent to a category known as the Schanuel topos; if $\mathbb{I}$ is the category whose objects are finite sets of atoms and arrows are injective functions between them, the Schanuel topos is the full subcategory of $\text{Set}^\mathbb{I}$ consisting of functors $\mathbb{I} \to \text{Set}$ that preserve pullbacks. However we will continue to present our results in a set-theoretic, rather than topos-theoretic, manner in this dissertation. In particular, this set-theoretic approach will provide a more convenient setting to introduce the key concept of FM-set in the next section.
2.5 FM-Sets

The usual von Neumann hierarchy of sets [57] is defined by

\[ V_0 \triangleq \emptyset \]
\[ V_{\alpha+1} \triangleq P(V_\alpha) \]
\[ V_\lambda \triangleq \bigcup_{\alpha<\lambda} V_\alpha \quad (\lambda \text{ a limit ordinal}) \]  

(2.20)

We then define the *von Neumann universe* as

\[ V \triangleq \bigcup_{\alpha} V_\alpha \]  

(2.21)

where \( \alpha \) ranges over the ordinals.

Similarly, we define the *Fraenkel-Mostowski hierarchy*, or *FM-hierarchy*, of nominal sets by

\[ \mathcal{F}M_0 \triangleq \emptyset \]
\[ \mathcal{F}M_{\alpha+1} \triangleq \mathcal{A} + P_{fs}(\mathcal{F}M_\alpha) \]
\[ \mathcal{F}M_\lambda \triangleq \bigcup_{\alpha<\lambda} \mathcal{F}M_\alpha \quad (\lambda \text{ a limit ordinal}) \ . \]  

(2.22)

\( \emptyset \) is a nominal set by Example 2.2.4. \( \mathcal{A} \) is a nominal set also (Example 2.2.5), and disjoint unions (Example 2.2.7) and finitely supported powersets (Example 2.2.8) of nominal sets are themselves nominal sets. Each \( \mathcal{F}M_\alpha \) is a nominal subset of \( \mathcal{F}M_{\alpha+1} \), so the union over all \( \alpha < \lambda \) is a nominal set, so this indeed defines a hierarchy of nominal sets.

The *Fraenkel-Mostowski universe*, or *FM-universe*, is defined to be

\[ \mathcal{F}M \triangleq \bigcup_{\alpha} \mathcal{F}M_\alpha \]  

(2.23)

where \( \alpha \) ranges over the ordinals.

**Definition 2.5.1.** \( \mathcal{F}M \) is divided by disjoint union into atoms \((a, 0)\) for \( a \in \mathcal{A} \) and sets \((X, 1)\). Call such an \((X, 1)\) an *FM-set*. We may leave off the 0 and 1 arguments of the disjoint union when discussing the atoms and FM-sets of \( \mathcal{F}M \).

The permutation action on \( \mathcal{F}M \) is defined by Example 2.2.5 and (2.8) as

\[ \pi \cdot a = \pi(a) \]
\[ \pi \cdot X = \{ \pi \cdot x \mid x \in X \} \ . \]  

(2.24)

We can formulate most of the usual set theoretic constructions with FM-sets. In particular, given FM-sets \( X, Y \), an *FM-function* \( X \to Y \) is an FM-set that contains only ordered pairs, where each \( x \in X \) appears exactly once as the first argument of a pair and the second arguments are drawn from \( Y \). The permutation action on an FM-function can be seen to be (2.18), and we will follow Section 2.4 in calling FM-functions with empty support *equivariant*. 

Definition 2.5.2. The FM-sets and FM-functions between them form a category $\mathcal{FM}$-Set. Similarly, for any limit ordinal $\lambda$, the FM-sets and FM-functions between them that are contained in $\mathcal{FM}_\lambda$ form a small category $\mathcal{FM}_\lambda$-Set.

Remark 2.5.3. Because we have access to most set-theoretic constructions with FM-sets, we will sometimes refer to an FM-set without specifying its explicit transfinite inductive foundations, just as we generally ignore the von Neumann hierarchy when dealing with sets. In particular, we will sometimes wish to form FM-sets whose members are labelled, such as

$$\{ V_a \mid a \in A \} .$$

This is easily seen to be isomorphic to $A$, and is a convenient way to present this FM-set for certain applications.

The exception to this rule is that we do not have choice functions by Lemma 2.4.5. In fact this exception is no accident, as FM-sets come from an attempt to prove the independence of the Axiom of Choice from the other axioms of Zermelo-Fraenkel set theory with atoms [32, Section 6].

An FM-set with empty support (that is, one that is closed under the permutation action) can be seen to be a nominal set in the sense of Definition 2.2.1. Throughout this dissertation we will want the ability to refer to FM-sets with non-empty support, but many important concepts will give rise to emptily supported FM-sets. Conversely, for much of the dissertation we will not need to refer to any nominal set that could not be considered as an FM-set, though following Remark 2.5.3 we will not always provide their exhaustive definition within the FM-hierarchy. Therefore for the rest of the dissertation it is convenient for us to redefine the notion of nominal set:

Definition 2.5.4. A nominal set is an FM-set with empty support. To remove ambiguity, when we wish to refer to a nominal set as defined in Definition 2.2.1, as we will in Section 7.1, we will call it a Nom-object.
Chapter 3

Nominal Equational Logic

This chapter will introduce the syntax and semantics of Nominal Equational Logic (NEL), the logic that is the central topic of study in this dissertation.

3.1 Signatures and structures

We are going to consider a simple generalisation of the usual notion of many-sorted algebraic signature [39, Section 3.1] in which the sorts and operation symbols form nominal sets\(^1\) rather than sets, and hence may have non-empty support. Note that this is also a generalisation of the notion of NEL-signature introduced in [10], where the sorts formed only a set. This generalisation will prove useful for the categorial results of Chapter 7, as we discuss in Remark 7.5.4, and will be discussed in Section 8.7 also.

**Definition 3.1.1.** A NEL-signature \(\Sigma\) is specified by

- a nominal set \(\text{Sort}_\Sigma\), whose elements are called the *sorts* of \(\Sigma\);
- a nominal set \(\text{Op}_\Sigma\), whose elements are called the *operation symbols* of \(\Sigma\); and
- an equivariant *typing function* assigning to each \(op \in \text{Op}_\Sigma\) a *type* consisting of a finite (possibly empty) list \(\vec{s}\) of sorts of \(\Sigma\) and a sort \(s\) of \(\Sigma\). As usual, the list \(\vec{s} = [s_1, \ldots, s_n]\) indicates the number and sort of arguments that \(op\) accepts and \(s\) indicates the sort of result it returns. We write

\[
\text{op} : \vec{s} \rightarrow s
\]

(3.1)

to indicate this typing information. Where \(\vec{s}\) is the empty list [ ] we call \(op\) a constant and write \(\text{op} : s\).

\(^1\)Recalling from Definition 2.5.4 that ‘nominal set’ refers to an FM-set with empty support.
Equivariance of the typing function means that for all \( \pi \in \text{Perm} \), if (3.1) holds, then 
\( \pi \cdot \text{op} : \pi \cdot \vec{s} \rightarrow \pi \cdot s \) also holds, where \( \pi \cdot \vec{s} \) is defined by

\[
\pi \cdot [s_1, \ldots, s_n] = [\pi \cdot s_1, \ldots, \pi \cdot s_n].
\] (3.2)

Unlike other notions of signature developed to discuss names and binding [18, 61], the definition above lacks explicit support for name-binding operators. This turns out not to be needed, as an operation symbol can be made to bind an atom or list of atoms via the axioms of a NEL-theory. This subject is discussed more fully in the next chapter.

**Example 3.1.2.** A NEL-signature \( \Sigma \) for the untyped \( \lambda \)-calculus [4] can be defined by setting 
\( \text{Sort}_\Sigma = \{ \text{tm} \} \) with the trivial permutation action, where the only sort \( \text{tm} \) represents \( \lambda \)-terms, and \( \text{Op}_\Sigma \) as the nominal set

\[
\{ V_a \mid a \in A \} \cup \{ L_a \mid a \in A \} \cup \{ A \}
\] (3.3)

whose members represent variables, \( \lambda \)-abstractions and application respectively. The permutation action on \( \text{Op}_\Sigma \) is

\[
\pi \cdot V_a = V_{\pi(a)}, \quad \pi \cdot L_a = L_{\pi(a)}, \quad \pi \cdot A = A.
\] (3.4)

In other words, \( \text{Op}_\Sigma \) is isomorphic to \( A + A + 1 \), where \( A \) is the nominal set of atoms (Example 2.2.5) and 1 is the singleton nominal set (Example 2.2.4). The typing function is defined by

\[
V_a : \text{tm}, \quad L_a : \text{tm} \rightarrow \text{tm}, \quad A : \text{tm, tm} \rightarrow \text{tm}.
\] (3.5)

**Example 3.1.3.** As discussed in Remark 2.1.2, the definition of the set of atoms \( A \) can be naturally extended to label atoms by atom sorts. This modest extension allows us to extend the previous example to the simply typed \( \lambda \)-calculus, typed \( \text{à la} \) Church [3, Section 3.2].

Fix a set \( \text{TyVar} \) of type variables, and define the nominal set \( \text{Sort}_\Sigma \) inductively by

\[
s ::= b \mid s \Rightarrow s. \quad (b \in \text{TyVar})
\] (3.6)

Let \( A \) be divided into countably infinite subsets \( A_s \) for all \( s \in \text{Sort}_\Sigma \). Then the \( \text{Op}_\Sigma \) is defined by

\[
\{ V^s_a \} \cup \{ L^s_{a, s'} \} \cup \{ A^{s, s'} \}
\] (3.7)

as \( s, s' \) range over \( \text{Sort}_\Sigma \) and \( a \) ranges over \( A_s \). The permutation is defined in the evident way, as with (3.4), and is well defined because permutations respect atom sorts. The typing function is defined by

\[
V^s_a : s
\]
\[
L^s_{a, s'} : [s'] \rightarrow (s \Rightarrow s')
\]
\[
A^{s, s'} : [s \Rightarrow s', s] \rightarrow s'.
\] (3.8)

The simply typed \( \lambda \)-calculus can also be typed \( \text{à la} \) Curry [3, Section 3.1], with typing contexts assigning types to atoms rather than explicitly sorting atoms as we have here. This approach to typing cannot be satisfactorily portrayed by Nominal Equational Logic, for reasons that we will discuss in Chapter 8.7.
 CHAPTER 3. NOMINAL EQUATIONAL LOGIC

Definition 3.1.4. Given a NEL-signature $\Sigma$, a $\Sigma$-structure $M$ in the category $\mathcal{FM}$-Set is specified by

- an equivariant map from each sort $s \in \text{Sort}_\Sigma$ to an FM-set $M[\llbracket s \rrbracket]$;

- an equivariant map from each operation symbol $op \in \text{Op}_\Sigma$ with type $\vec{s} \rightarrow s$ to an FM-function

$$M[\llbracket op \rrbracket] : M[\llbracket \vec{s} \rrbracket] \rightarrow M[\llbracket s \rrbracket]$$

where if $\vec{s} = [s_1, \ldots, s_n]$ then $M[\llbracket \vec{s} \rrbracket] \triangleq M[\llbracket s_1 \rrbracket] \times \cdots \times M[\llbracket s_n \rrbracket]$, while if $\vec{s}$ is the empty list then $M[\llbracket \vec{s} \rrbracket] \triangleq 1$.

Note that by (2.19),

$$\pi \cdot (M[\llbracket op \rrbracket](x_1, \ldots, x_n)) = M[\llbracket \pi \cdot op \rrbracket](\pi \cdot x_1, \ldots, \pi \cdot x_n).$$

3.2 Terms and values

The terms over a conventional algebraic signature are built up from variables by applying operation symbols. Given a structure in the category of sets for the signature and a valuation of the variables as elements of the structure, each term denotes an element of the structure. We wish to extend this to NEL-signatures and structures for them in the category $\mathcal{FM}$-Set of FM-sets. Doing so involves an extension of the usual notion of algebraic term to take account of the atom-permutation action. Since operations in a NEL-signature denote FM-functions (3.9), the action of a permutation on a compound term can distribute through the term to act on the operator and on its arguments, as in (3.10). Thus the only trace of the permutation action on terms that it is really necessary to incorporate into their structure is in the case that a permutation acts on a variable. Following [61] we use suspensions $\pi x$ consisting of a permutation $\pi$ waiting to be applied once more is known about the unknown element of an FM-set represented by the variable $x$.

Definition 3.2.1. Fixing a countably infinite set $\text{Var}$ of variables, the grammar of terms over a NEL-signature $\Sigma$ is given by

\[
\begin{align*}
\text{Variables} & \quad x & \in & \text{Var} \\
\text{Permutations} & \quad \pi & \in & \text{Perm} \\
\text{Operation symbols} & \quad op & \in & \text{Op}_\Sigma \\
\text{Terms} & \quad t & ::= & \pi x \mid op t \cdots t. \\
\end{align*}
\]

We call $\pi x$ a suspension and $op t_1 \cdots t_n$ a constructed term.

\(^2\)The term “moderated variable” is also used for what we call suspensions: see [15].
Note that all occurrences of variables $x$ in terms are preceded by a suspended permutation $\pi$. However, when $\pi$ is the identity permutation $\iota$, we shall usually abbreviate the term $\iota x$ just to $x$.

We define the set of variable occurrences $\text{Var}(t) \subseteq \text{Var}$ for a term $t$ in the obvious way:

$$\text{Var}(\pi x) \triangleq \{x\}$$

$$\text{Var}(\text{op } t_1 \cdots t_n) \triangleq \text{Var}(t_1) \cup \cdots \cup \text{Var}(t_n).$$  \hspace{1cm} (3.12)

**Definition 3.2.2.** A sorting environment over a NEL-signature $\Sigma$ is a partial function $\Gamma : \text{Var} \rightarrow \text{Sort}_\Sigma$ with finite domain $\text{dom}(\Gamma) \subseteq \text{Var}$. If $\text{dom}(\Gamma)$ is the set of variables $\{x_1, \ldots, x_n\}$, and $\Gamma(x_i) = s_i$ for $1 \leq i \leq n$, then we can write $\Gamma$ as

$$[x_1 : s_1, \ldots, x_n : s_n].$$  \hspace{1cm} (3.13)

We can define a permutation action on sorting environments by

$$(\pi \cdot \Gamma)(x) \triangleq \pi(\Gamma(x))$$  \hspace{1cm} (3.14)

and hence the set of sorting environments over $\Sigma$ defines a nominal set, which we call $\text{SE}_\Sigma$, where

$$\text{supp}(\Gamma) = \bigcup_{x \in \text{dom}(\Gamma)} \text{supp}(\Gamma(x)).$$  \hspace{1cm} (3.15)

**Definition 3.2.3.** The sets $\Sigma_s(\Gamma)$ of terms of sort $s \in \text{Sort}_\Sigma$ in a sorting environment $\Gamma$ are inductively defined by:

- if $\pi \in \text{Perm}$, $x \in \text{dom}(\Gamma)$ and $\Gamma(x) = s$, then $\pi x \in \Sigma_{\pi \cdot s}(\Gamma)$;
- if $\text{op} \in \text{Op}_\Sigma$ has type $[s_1, \ldots, s_n] \rightarrow s$ and $t_i \in \Sigma_{s_i}(\Gamma)$ for $i = 1..n$, then $\text{op } t_1 \cdots t_n \in \Sigma_{s}(\Gamma)$.

Let $\Sigma(\Gamma)$ be the union of all $\Sigma_s(\Gamma)$ as $s$ ranges over $\text{Sort}_\Sigma$; this is equivalent to the set of terms $t$ such that $\text{Var}(t) \subseteq \text{dom}(\Gamma)$. For all such terms there is exactly one sort $s$ such that $t \in \Sigma_s(\Gamma)$.

Let $\text{Term}_\Sigma$ be the union of all $\Sigma(\Gamma)$ as $\Gamma$ ranges over $\text{SE}_\Sigma$; that is, the set of all terms that can be well-sorted.

**Definition 3.2.4.** Given a signature $\Sigma$, sort $s \in \text{Sort}_\Sigma$, sorting environment $\Gamma \in \text{SE}_\Sigma$, term $t \in \Sigma_s(\Gamma)$ and permutation $\pi \in \text{Perm}$, the meta-level permutation action of $\pi$ on $t$ is defined by:

$$\pi \cdot (\pi' x) \triangleq \pi(\pi' x)$$

$$\pi \cdot (\text{op } t_1 \cdots t_n) \triangleq (\pi \cdot \text{op})(\pi \cdot t_1) \cdots (\pi \cdot t_n).$$  \hspace{1cm} (3.16)

In the suspension case, if $\pi' x \in \Sigma_s(\Gamma)$ then $\Gamma(x) = \pi'^{-1} \cdot s$, so $(\pi \cdot \Gamma)(x) = \pi(\pi'^{-1} \cdot s)$ by (3.14). Therefore $\pi \cdot (\pi' x) \in \Sigma_{\pi \cdot \pi'^{-1}}(\pi \cdot \Gamma) = \Sigma_{\pi \cdot s}(\pi \cdot \Gamma)$. An easy induction shows that $\pi \cdot (\text{op } t_1 \cdots t_n) \in \Sigma_{\pi \cdot s}(\pi \cdot \Gamma)$ similarly, so

$$t \in \Sigma_s(\Gamma) \Rightarrow \pi \cdot t \in \Sigma_{\pi \cdot s}(\pi \cdot \Gamma).$$  \hspace{1cm} (3.17)
As noted in Example 2.2.6, permutations are finitely supported with respect to the conjugation action; operation symbols are also finitely supported because they are elements of the given nominal set $\Omega_\Sigma$. Further, the sets $\Sigma_\pi(\Gamma)$, $\Sigma(\Gamma)$ and $\text{Term}_\Sigma$ can be seen to be FM-sets because they draw their components from the FM-sets $\Omega_\Sigma$ and $\text{Perm}$ (we are following Remark 2.5.3 and not giving the complete description of how terms would be defined as members of the FM-hierarchy, but it is clear that this could be done). The support sets for terms are given by:

$$\text{supp}(\pi x) = \text{supp}(\pi)$$

(3.18)

$$\text{supp}(op t_1 \cdots t_n) = \text{supp}(op) \cup \text{supp}(t_1) \cup \cdots \cup \text{supp}(t_n).$$

The supports of $\Sigma_\pi(\Gamma)$, $\Sigma(\Gamma)$ and $\text{Term}_\Sigma$ are $\text{supp}(\Gamma) \cup \text{supp}(s)$, $\text{supp}(\Gamma)$ and $\emptyset$ respectively, so, in particular, $\text{Term}_\Sigma$ is a nominal set.

**Lemma 3.2.5.** Given a signature $\Sigma$, the map from $\{((\Gamma, t) \in \text{SE}_\Sigma \times \text{Term}_\Sigma \mid t \in \Sigma(\Gamma)\}$ to $\text{Sort}_\Sigma$ sending each pair $(\Gamma, t)$ to its unique sort (c.f. Definition 3.2.3) is equivariant.

**Proof.** $(\Gamma, \pi' \cdot x)$ maps to $\pi' \cdot \Gamma(x)$; applying $\pi$ to this gives us $\pi \pi' \cdot \Gamma(x)$. Conversely, $\pi \cdot (\Gamma, \pi' x) = (\pi \cdot \Gamma, \pi \pi' \cdot x)$ maps to $\pi \pi' \cdot \Gamma(\pi \cdot \cdot \Gamma(x) = \pi \pi' \cdot \Gamma(x)$. The constructed term case is trivial.

The above lemma is useful because if we have a term $t \in \Sigma_\pi(\Gamma)$ and atom $s \# (t, \Gamma)$ then by (2.16) we get $s \# s$ ‘for free’. This will simplify some of our later definitions.

We next describe the intended interpretation of terms as elements of FM-sets.

**Definition 3.2.6.** Given a NEL-signature $\Sigma$, let $M$ be a $\Sigma$-structure and take $\Gamma \in \text{SE}_\Sigma$. The finite product of FM-sets $M[\Gamma(x)]$ as $x$ ranges over $\text{dom}(\Gamma)$ is itself an FM-set, for which we will write $M[\Gamma]$.

We call the elements of this FM-set $\Gamma$-valuations in $M$. They are functions $\rho$ defined on the finite set of variables $\text{dom}(\Gamma)$ and mapping each $x \in \text{dom}(\Gamma)$ to an element $\rho(x)$ of $M[\Gamma(x)]$. Since $M[\Gamma]$ is a finite product of FM-sets, the action of a permutation $\pi \in \text{Perm}$ on $\rho \in M[\Gamma]$ is given by:

$$(\pi \cdot \rho)(x) = \pi \cdot \rho(x).$$

(3.19)

So $(\pi \cdot \rho)(x) \in \pi \cdot M[s]$, which is $M[\pi \cdot s]$ by Definition 3.1.4. Therefore $\pi \cdot \rho \in M[\pi \cdot \Gamma]$, and the support of a valuation under the action (3.19) is

$$\text{supp}(\rho) = \bigcup_{x \in \text{dom}(\rho)} \text{supp}(\rho(x)).$$

(3.20)

**Definition 3.2.7.** The value $M[t]\rho$ of a well-sorted term $t \in \Sigma_\pi(\Gamma)$ with respect to a valuation $\rho \in M[\Gamma]$ is an element of $M[s]$. Values are defined by recursion on the structure of terms:

$$M[\pi x]\rho \triangleq \pi \cdot \rho(x)$$

$$M[op \ t_1 \cdots t_n]\rho \triangleq M[op](M[t_1]\rho, \ldots, M[t_n]\rho).$$

(3.21)
CHAPTER 3. NOMINAL EQUATIONAL LOGIC

Notation 3.2.8. Where the name of the structure is not important we will leave it out; that is, we will write \( M[s] \) as \([s]\), \( M[t] \rho \) as \([t] \rho\), and so forth.

Lemma 3.2.9. Given a \( \Sigma \)-structure \( M \) and \( \Gamma \in \text{SE}_\Sigma \), the map \((t, \rho) \mapsto M[t] \rho\) is an equivariant function from \( \Sigma(\Gamma) \times M[\Gamma] \) to the disjoint union of all \( M[s] \) as \( s \) ranges over \( \text{Sort}_\Sigma \).

Proof. We will follow Notation 3.2.8 and leave out the name of the algebra. The lemma asks us to show that
\[
\llbracket \pi \cdot t \rrbracket (\pi \cdot \rho) = \pi \cdot (\llbracket t \rrbracket \rho) .
\]
by induction on the structure of \( t \). \( \llbracket \pi \cdot (\pi' x) \rrbracket (\pi \cdot \rho) = \pi \cdot (\llbracket \pi' x \rrbracket \rho) \) by (3.16), (3.21) and (2.4), while \( \llbracket \pi \cdot (\text{op} t_1 \cdots) \rrbracket (\pi \cdot \rho) = \pi \cdot (\llbracket \text{op} t_1 \cdots \rrbracket \rho) \) by (3.16), (3.21) and (3.10).

3.3 Substitution

Definition 3.3.1. Given a permutation \( \pi \) and term \( t \), the object-level permutation action of \( \pi \) on \( t \), \((\pi, t) \mapsto \pi \ast t \) is defined by
\[
\pi \ast (\pi' x) \triangleq \pi \pi' x,
\]
\[
\pi \ast (\text{op} t_1 \cdots t_n) \triangleq (\pi \cdot \text{op}) (\pi \ast t_1) \cdots (\pi \ast t_n) .
\]
(3.23)

Compare this to the meta-level action of Definition 3.2.4. It is easily verified by induction on the structure of \( t \) that
\[
t \in \Sigma_s(\Gamma) \Rightarrow \pi \ast t \in \Sigma_{\pi \cdot s}(\Gamma) .
\]
(3.24)

Another easy induction confirms that this defines a permutation action:
\[
\pi \ast (\pi' \ast t) = \pi \pi' \ast t .
\]
(3.25)

However terms are not finitely supported with respect to this action, as they are by the meta-level permutation action, because permutations are not finitely supported under the left multiplication action (Example 2.2.6).

Lemma 3.3.2. Given a NEL-signature \( \Sigma \) and \( \Sigma \)-structure whose name we shall as usual omit, and for all \( \pi \in \text{Perm} \), \( t \in \Sigma_s(\Gamma) \), and \( \rho \in \llbracket \Gamma \rrbracket \),
\[
\llbracket \pi \ast t \rrbracket \rho = \pi \cdot (\llbracket t \rrbracket \rho) .
\]

Proof. We perform induction on the structure of \( t \); the suspension case follows by the definitions (3.21) and (3.23) along with (2.4), while the constructed term case uses the definitions and (3.10).

This lemma shows that finding the value of a term with a permutation applied to it is equivalent to applying that permutation to the term’s value. In other words, when a term \( t \) is interpreted as a member of an FM-set then \( \pi \ast t \) is the correct object level way to apply the permutation action of that FM-set. Therefore when we substitute some term for \( x \) in a suspension \( \pi \cdot x \) this notion of permutation action is the correct one to use:
**Definition 3.3.3.** Given a NEL-signature $\Sigma$ and sorting environments $\Gamma = [x_1 : s_1, \ldots, x_n : s_n]$ and $\Gamma'$ over $\Sigma$, the set $\Sigma(\Gamma, \Gamma')$ of substitutions from $\Gamma$ to $\Gamma'$ consists of functions $\sigma$ mapping each variable $x_i$ in $dom(\Gamma)$ to a term $\sigma(x_i) \in \Sigma_a(\Gamma')$. Given a term $t \in \Sigma_a(\Gamma)$ and a substitution $\sigma \in \Sigma(\Gamma, \Gamma')$, we define a term $t\{\sigma\}$ by

\[
\begin{align*}
(op\ t_1 \cdots t_n)\{\sigma\} & \triangleq op\ t_1\{\sigma\} \cdots t_n\{\sigma\} \\
(\pi\ x)\{\sigma\} & \triangleq \pi\ *\ \sigma(x)
\end{align*}
\] (3.26)

If $\pi\ x \in \Sigma_a(\Gamma)$ then $\Gamma(x) = \pi^{-1} \cdot s$, so $\sigma(x) \in \Sigma_{\pi^{-1}\cdot s}(\Gamma')$. Then $\pi\ *\ \sigma(x) \in \Sigma_a(\Gamma')$ by (3.24). A routine induction shows that $(op\ t_1 \cdots t_n)\{\sigma\} \in \Sigma_a(\Gamma')$ similarly. So

\[
t \in \Sigma_a(\Gamma) \land \sigma \in \Sigma(\Gamma, \Gamma') \Rightarrow t\{\sigma\} \in \Sigma_a(\Gamma') .
\] (3.27)

**Lemma 3.3.4.** The following standard properties of a notion of substitution hold for the definition above:

\[
t\{id\} = t
\] (3.28)

where $id \in \Sigma(\Gamma, \Gamma)$ is the identity substitution, $x \mapsto \iota\ x$; and

\[
(t\{\sigma\})\{\sigma'\} = t\{\sigma;\sigma'\}
\] (3.29)

where $(\sigma;\sigma') \in \Sigma(\Gamma, \Gamma'')$ is the composition of $\sigma \in \Sigma(\Gamma, \Gamma')$ and $\sigma' \in \Sigma(\Gamma', \Gamma'')$, given by

\[
(\sigma;\sigma')(x) \triangleq \sigma(x)\{\sigma'\} .
\] (3.30)

**Proof.** The proof of (3.28) is trivial. To prove (3.29) we first prove

\[
(\pi\ *\ t)\{\sigma\} = \pi\ *\ (t\{\sigma\})
\] (3.31)

by induction on the structure of $t$. The suspension case $(\pi\ *\ \pi'\ x)\{\sigma\} = \pi\ *\ (\pi'\ x\{\sigma\})$ follows by (3.23), (3.26) and (3.25), while the constructed term case is routine. The suspension case of (3.29), $\pi\ x\{\sigma;\sigma'\} = (\pi\ x\{\sigma\})\{\sigma'\}$, then follows by (3.26), (3.30) and (3.31), while the constructed term case is, again, routine.

**Lemma 3.3.5.** Given a NEL-signature $\Sigma$, sorting environments $\Gamma, \Gamma' \in SE_\Sigma$ and some $\Sigma$-structure, for all $t \in \Sigma_a(\Gamma)$, $\sigma \in \Sigma(\Gamma, \Gamma')$ and $\rho \in [\Gamma']$

\[
[[t\{\sigma\}]]\rho = [[t]]([\sigma]\rho)
\]

where by definition $[\sigma]\rho \in [\Gamma]$ is the valuation mapping each $x \in dom(\Gamma)$ to $[[\sigma(x)]]\rho$.

**Proof.** Proof is by induction on the structure of $t$. $[[\pi\ x\{\sigma\}]]\rho = [[\pi\ *\ \sigma(x)]]\rho$ by (3.26), while $[[\pi\ x]]([\sigma]\rho) = \pi\cdot([\sigma(x)]\rho)$ by (3.21). These are equal by Lemma 3.3.2. The constructed term case is routine.
We can now express the meta-level action \((\pi, t) \mapsto \pi \cdot t\) in terms of the object-level action \((\pi, t) \mapsto \pi^* t\) (c.f. [22, Lemma 2.3]).

**Lemma 3.3.6.** Given a NEL-signature \(\Sigma\), a sorting environment \(\Gamma\) and a term \(t \in \Sigma_s(\Gamma)\), for any \(\pi \in \text{Perm}\)

\[\pi \cdot t = (\pi^* t)^{\pi^{-1}}\]

where \((\pi^{-1})\) is defined to be the substitution mapping each \(x \in \text{dom}(\Gamma)\) to \(\pi^{-1} x\).

*Proof.* Proof is by induction on the structure of \(t\). \((\pi^* (\pi' x))^ {\pi^{-1}}\) by (3.23) and (3.26), which is \(\pi \cdot (\pi' x)\) by (3.16). The constructed term case is routine.

As a corollary of this we have that \((t, \sigma) \mapsto t\{\sigma}\) is equivariant:

**Corollary 3.3.7.** Given a NEL-signature \(\Sigma\), a substitution \(\sigma \in \Sigma(\Gamma, \Gamma')\) and a term \(t \in \Sigma_s(\Gamma)\), for any \(\pi \in \text{Perm}\)

\[\pi \cdot (t\{\sigma\}) = (\pi \cdot t)^{\pi \cdot \sigma}\]

where \(\pi \cdot \sigma \in \Sigma(\pi \cdot \Gamma, \pi \cdot \Gamma')\) is defined to be the substitution mapping each \(x \in \text{dom}(\Gamma)\) to \(\pi \cdot \sigma(x)\).

*Proof.* Proof proceeds by induction on the structure of \(t\). The suspension case \((\pi \cdot (\pi' x))^{\pi \cdot \sigma}\) follows by the definitions (3.16) and (3.26) along with the results Lemma 3.3.6, (3.31) and (3.25). The constructed term case is routine.

Note that under the action \((\pi, \sigma) \mapsto \pi \cdot \sigma\), the set of all substitutions forms a nominal set: if \(\sigma \in \Sigma(\Gamma, \Gamma')\) then

\[\text{supp}(\sigma) = \bigcup_{x \in \text{dom}(\Gamma)} \text{supp}(\sigma(x))\]

(3.32)

Each \(\Sigma(\Gamma, \Gamma')\) is an FM-set supported by \(\text{supp}(\Gamma) \cup \text{supp}(\Gamma')\).

### 3.4 Theories and algebras

Ordinary equational logic formalises reasoning about equations between algebraic terms. We wish to formalise reasoning both about equality and about the freshness relation of Definition 2.3.2. In the formal system we will use the symbols “\(\approx\)” and “\(\#\)” for the logical relations intended to represent equality and freshness, and continue to use “\(=\)” and “\(\#\)” for their interpretation in FM-sets as the actual equality and “not-in-the-support-of” relations.
Definition 3.4.1. Equational logic presents terms in the context of sorting environments such as those introduced by Definition 3.2.2. We wish to augment this notion to include freshness assertions about variables. We do this with freshness environments, which are partial functions \( \nabla : \text{Var} \to \text{Sort}_\Sigma \otimes \mathcal{P}_{\text{fin}}(\text{Atom}) \) with finite domain \( \text{dom}(\nabla) \subseteq \text{Var} \). The use of the separated tensor (Example 2.3.8) means that if \( \nabla(x) = (s, \overline{a}) \) then \( \overline{a} \neq s \). If \( \text{dom}(\nabla) = \{x_1, \ldots, x_n\} \) and \( \nabla(x_i) = (s_i, \overline{a}_i) \) for \( 1 \leq i \leq n \) then we can write \( \nabla \) as
\[
[\overline{a}_1 \neq x_1 : s_1, \ldots, \overline{a}_n \neq x_n : s_n].
\] (3.33)
We can define a permutation action on sorting environments by
\[
(\pi \cdot \nabla)(x) \triangleq \pi \cdot (\nabla(x)).
\] (3.34)
This action is well defined because \( \overline{a}_i \neq s_i \) implies \( \pi \cdot \overline{a}_i \neq \pi \cdot s_i \) by Lemma 2.3.4. The set of freshness environments over \( \Sigma \) defines a nominal set, which we call \( \text{FE}_\Sigma \), where
\[
\text{supp}(\nabla) = \bigcup_{x \in \text{dom}(\nabla)} \text{supp}(\nabla(x)).
\] (3.35)
Given \( \nabla \in \text{FE}_\Sigma \) we derive the sorting environment \( \nabla^\downarrow \in \text{SE}_\Sigma \) by composing the first projection with \( \nabla \).

Remark 3.4.2. One aspect of the above definition that may not immediately be clear is why freshness environments are defined to send each variable to a member of the nominal set \( \text{Sort}_\Sigma \otimes \mathcal{P}_{\text{fin}}(\text{Atom}) \), rather than the nominal set of all pairs \( \text{Sort}_\Sigma \times \mathcal{P}_{\text{fin}}(\text{Atom}) \). The justification for this is that we want freshness to be definable in terms of equality, as with Lemma 2.3.5.

Suppose that \( \nabla(x) = (s, \overline{a}) \). By Definition 3.1.4 there is an equivariant map \( s \mapsto [s] \). We intend \( \overline{a} \) to be fresh for all \( x \in [s] \); by Lemma 2.3.5 this is equivalent (in the FM-universe) to saying that \( (\overline{a} \, \overline{a}') \cdot x = x \), where \( \overline{a} \) is an ordering of \( \overline{a} \) and \( \overline{a}' \) is a fresh tuple of the same size. But for them to be equal in the FM-set specified by the given structure we need to know that \( (\overline{a} \, \overline{a}') \cdot [s] = [s] \); this justifies the restriction \( \overline{a} \neq s \), which by (2.16) implies \( \overline{a} \neq [s] \).

This identification of freshness with equality will be a recurring idea in this dissertation, explored particularly in Section 5.5 and Chapter 7.

Definition 3.4.3. We wish NEL to be expressive enough to reason about both equality between terms and the freshness of atoms for terms. Rather than use separate judgements for equality and freshness, it is convenient to roll both into a single judgement form. So we define a \( \text{NEL}-\)judgement over a signature \( \Sigma \) to have the form
\[
\nabla \vdash \overline{a} \neq t \approx t' : s
\] (3.36)
where \( \nabla \in \text{FE}_\Sigma \) (Definition 3.4.1), \( \overline{a} \in \mathcal{P}_{\text{fin}}(A) \), \( s \in \text{Sort}_\Sigma \), \( \overline{a} \neq s \) (following Remark 3.4.2) and \( t, t' \in \Sigma_\delta(\nabla^\downarrow) \) (Definition 3.2.3), where \( \nabla^\downarrow \) is the sorting environment derived from a freshness environment as in Definition 3.4.1.
CHAPTER 3. NOMINAL EQUATIONAL LOGIC

Notation 3.4.4. Although the single form of judgement (3.36) combining equality and freshness is useful for stating the general rules of nominal equational logic, in particular cases it is clearer to use the following abbreviations.

- $t \approx t' : s$ means $\emptyset \not\equiv t \approx t' : s$; similarly, $x : s$ in a freshness environment means $\emptyset \not\equiv x : s$.
- $\overline{a} \not\equiv t : s$ means $\overline{a} \not\equiv t : s$.
- $a \not\equiv t \approx t' : s$ means $\{a\} \not\equiv t \approx t' : s$; similarly, $a \not\equiv x : s$ in a freshness environment means $\{a\} \not\equiv x : s$.

Definition 3.4.5. A NEL-theory $\mathcal{T}$ consists of a NEL-signature $\Sigma$ together with a set of NEL-judgements over $\Sigma$, which we call axioms.

Example 3.4.6. Fig. 3.1 defines a NEL-theory for $\alpha\beta\eta$-equivalence on the untyped $\lambda$-calculus of Example 3.1.2. The axiom ($\eta$) for $\eta$-equivalence is straightforward. We will see in the next chapter that the axiom ($\alpha$) captures the notion of $\alpha$-equivalence modulo the rules of nominal equational logic. The first four axioms for $\beta$-equivalence involve unwinding the usual definition

$$A(L_a x)x' \equiv_\beta x[x'/a] \quad (3.37)$$

through the various possible values of $x$. The axiom ($\beta$-5) tells us that substituting a fresh $a'$ for any $a$ in a term is equivalent to transposing those atoms in the term. These axioms are adapted from the theory of capture avoiding substitution in [22, Fig. 4]; we could alternatively follow that approach more directly by defining a nominal set of operation symbols for substitution

$$S_a : [tm, tm] \rightarrow tm \quad (3.38)$$

for all $a \in \mathcal{A}$, with the evident $Perm$-action $\pi \cdot S_a = S_{\pi(a)}$, then use the axioms of that paper directly, along with an axiom for substitution’s binding structure (following [22, Lemma 4.9]).

Turning to the interpretation of NEL-theories in $\mathcal{FM}$-Set, first note that the intended meaning of a freshness environment (3.33) is to assert not only that each variable $x_i$ has sort $s_i$, but also that it stands for an element of the corresponding FM-set whose support is disjoint from $\overline{x}_i$. Accordingly, we take the meaning of $\nabla$ in a $\Sigma$-structure $M$ to be the subset $M[\nabla] \subseteq M[\nabla^+]$ of the FM-set of $\nabla^+$-valuations (Definition 3.2.6) given by

$$M[\nabla] \overset{\Delta}{=} \{ \rho \in M[\nabla^+] \mid \overline{\pi}_1 \not\equiv \rho(x_1) \land \cdots \land \overline{\pi}_n \not\equiv \rho(x_n) \} \quad (3.39)$$

where $\not\equiv$ is the freshness relation in each $M[s_i]$ and $\nabla^+$ is the sorting environment associated with $\nabla$ as in Definition 3.4.1. As usual we will write $[\nabla]$ for $M[\nabla]$ where this does not cause confusion.
Remark 3.4.9. Judgements, such as the axioms of the theory given in Fig. 3.1 are not schematic over their atoms. That is, the atom \( a \) in the axiom \((\alpha)\) is not a variable that will be instantiated as an atom, but rather is literally a certain atom from the nominal set \( \mathbb{A} \). However Lemma 3.4.8 tells us that \((\alpha)\) holds if and only if the judgement produced by renaming \( a \) in \((\alpha)\) by some other atom \( a' \) also holds, so we may treat it as if it were schematic.

However, this is only true up to permutative renaming. The axiom \((\beta-3)\) uses two atoms \( a, a' \), and no permutation can map these atoms to the same atom. Therefore we may only regard \( a \) and \( a' \) as schematic variables for atoms if we remember that they must be instantiated to different atoms.
Chapter 4

Binding

For many practical applications explicit support for the crucial concept of binding at the lowest level may be desirable. However, the signatures of Definition 3.1.1 do not provide such support. The motivation for this is that the results of this dissertation will require dozens of proofs by induction on the structure of a term, and it is therefore useful to keep this structure as simple as possible. However, the ability to express binding is clearly essential, and this chapter will show that NEL is indeed expressive enough to do so.

4.1 Abstraction sets

To prove that Nominal Equational Logic is expressive enough to describe binding operators, we first need the correct notion of name binding and $\alpha$-equivalence in the FM-sets model. This was defined in [25, Section 5], generalising the equational rules for $\alpha$-equivalence over $\lambda$-operators presented in [28, Chapter 2]:

**Definition 4.1.1.** Given an FM-set $X$ we define a relation on $A \times X$, $(a, x) \sim (a', x')$, by

$$(a, x) \sim (a', x') \iff (a \ b \cdot x = (a' \ b) \cdot x'$$

(4.1)

for some\(^1\) atom $b \neq (a, a', x, x')$. This clearly defines an equivalence relation on $A \times X$; write the equivalence class containing $(a, x)$ as $\langle a \rangle x$ and call such a class the *atom abstraction of $a$ in $x$*.

A useful special case of (4.1) is where $a = a'$, so $(a \ b) \cdot x = (a \ b) \cdot x'$ and applying the bijection $(a \ b)$ to both sides gives us

$$(a, x) \sim (a, x') \iff x = x'$$

(4.2)

Another useful fact about these abstractions is that

$$\langle a \rangle x = \langle a' \rangle ((a \ a') \cdot x)$$

(4.3)

---

\(^1\)Or, equivalently, any - see Theorem 2.3.1.
for \( a' \# (a, x, X) \), as for suitably fresh \( b, (a, b) \cdot x = (a' b)(a a') \cdot x \) by Corollary 2.3.7.

**Lemma 4.1.2.** The equivalence relation of Definition 4.1.1 is equivariant; that is,

\[
(a, x) \sim (a', x') \implies (\pi(a), \pi \cdot x) \sim (\pi(a'), \pi \cdot x') .
\]

**Proof.** Take some \( b \# (a, a', x, x', \pi) \). \( (\pi(a) b) \pi = \pi(a b) \) and \( (\pi(a') b) \pi = \pi(a' b) \), so \( (\pi(a) b) \cdot (\pi \cdot x) = (\pi(a') b) \cdot (\pi \cdot x') \).

**Definition 4.1.3.** Define the set of *atom-abstractions* on an FM-set \( X \) by

\[
[A]X \triangleq \{ \langle a \rangle x \mid x \in X \land a \# X \} .
\]

We attach the condition ‘\( a \# X \)’ to this set for reasons we will discuss in Section 4.5. To keep the relation defined in Definition 4.1.1 total we can consider it as a relation on \((A - \text{supp}(X)) \times X\) rather than \( A \times X \).

In fact \([A]X\) is itself an FM-set, with the \( \text{Perm-action} \)

\[
\pi \cdot (\langle a \rangle x) \triangleq \langle \pi(a) \rangle (\pi \cdot x) \tag{4.4}
\]

which is well-defined by Lemma 4.1.2. It is clear that \( \text{supp}([A]X) = \text{supp}(X) \). Here we are following Remark 2.5.3 and omitting the precise definition of this FM-set in the FM-hierarchy; [25, Definition 5.4] provides these details.

**Lemma 4.1.4.** Given an FM-set \( X \), element \( x \in X \) and atom \( a \# X \),

\[
\text{supp}(\langle a \rangle x) = \text{supp}(x) - \{a\} .
\]

**Proof.** Suppose \( a' \notin \text{supp}(x) - \{a\} \). If \( a' = a \) then for fresh \( b \# (a, x, X) \) we have \( (a b) \cdot (\langle a \rangle x) = \langle b \rangle ((a b) \cdot x) = \langle a \rangle x \) by (4.3). If \( a' \# (a, x) \) then for fresh \( b \# (a, a', x, X) \) we have \( (a' b) \cdot (\langle a \rangle x) = \langle a \rangle x \) by (4.4). Either way, \( a' \# \langle a \rangle x \).

Conversely, suppose \( a' \# (a \langle a \rangle x \). We wish to prove that \( a' = a \) or \( a' \notin X \). Say \( a' \neq a \). Then for fresh \( b \# (a, a', x) \) we have \( (a b) \cdot (\langle a \rangle x) = (a' b) \cdot (\langle a \rangle x) = \langle a \rangle ((a' b) \cdot x) \). Therefore by (4.2) we have \( x = (a' b) \cdot x \), so \( a' \# x \).

**Corollary 4.1.5.** Given FM-sets \( X, Y \), let \( f : A \times X \to Y \) be an equivariant function. Then \( f \) induces a well-defined function \( \hat{f} : [A]X \to Y \), defined by \( \hat{f}(\langle a \rangle x) = f(a, x) \), if and only if \( a \# f(a, x) \) for all \( x \in X \) and \( a \# X \).

**Proof.** Left-to-right: \( a \# \langle a \rangle x \) by Lemma 4.1.4. But an equivariant function cannot increase the support set by (2.16), so \( a \# f(a, x) \).

Right-to-left: Suppose \( \langle a \rangle x = \langle a' \rangle x' \), so \( (a b) \cdot x = (a' b) \cdot x' \) for \( b \# (a, a', x, x') \). Then \( f(a, x) = (a b) \cdot f(a, x) \) because \( a, b \# f(a, x) \). This equals \( f(b, (a b) \cdot x) \) by equivariance, which is \( f(b, (a' b) \cdot x') = (a' b) \cdot f(a', x') = f(a', x') \).
It will be necessary to consider situations when a (not necessarily distinct) finite list of atoms, rather than a single atom, is abstracted:

**Lemma 4.1.6.** Suppose we have a list of atoms \( \vec{a} = (a_1, \ldots, a_n) \in \mathbb{A}^n \). Then we write \( \langle a_1 \rangle (\langle a_2 \rangle (\cdots (\langle a_n \rangle x) \cdots)) \in [\mathbb{A}] (\cdots ([\mathbb{A}] X)) \) as \( \langle \vec{a} \rangle x \in [\mathbb{A}^n] X \).

(i) \( \pi \cdot (\langle \vec{a} \rangle x) = \langle \pi \cdot \vec{a} \rangle (\pi \cdot x) \), where \( \pi \cdot \vec{a} \) is defined pointwise (2.5);

(ii) Given some other list of equal length \( \vec{a}' = (a'_1, \ldots, a'_n) \in \mathbb{A}^n \),
\[
\langle \vec{a} \rangle x = \langle \vec{a}' \rangle x' \iff (a_1 b_1) \cdots (a_n b_n) \cdot x = (a'_1 b_1) \cdots (a'_n b_n) \cdot x'
\]
for distinct and suitably fresh \( b, b_1, \ldots, b_k \). By (4.1) and (i) the left hand side of the equivalence (4.6) holds if and only if
\[
\langle (a b) \cdot \vec{a} \rangle (\langle a b \rangle \cdot x) = \langle (a' b) \cdot \vec{a}' \rangle (\langle a' b \rangle \cdot x')
\]
which by induction holds if and only if
\[
\langle (a b)(a_1 b_1) \cdots (a b)(a_k b_k) (a b) \cdot x \rangle = \langle (a' b)(a'_1 b_1) \cdots (a' b)(a'_k b_k) (a' b) \cdot x' \rangle
\]
The final step is to show that
\[
\langle (a b)(a_1 b_1) \cdots (a b)(a_k b_k) (a b) \cdot x \rangle = \langle (a b)(a_1 b_1) \cdots (a_k b_k) \cdot x \rangle
\]
and similarly for the right hand side of (4.8). By Corollary 2.3.7 we need only consider the permutations’ actions on the atoms \( a, a_1, \ldots, a_k \). For \( a_j \in \{a_1, \ldots, a_k\} \) not equal to \( a \), both sides map \( a_j \mapsto b_j \), so long as \( j > i \) for any other \( a_i = a_j \). If \( a \neq \vec{a} \) then both permutations map \( a \mapsto b \). Otherwise if \( a = a_j \), where \( j > i \) for any other \( a = a_i \), then both sides map \( a \mapsto b \). By this and the symmetric result for the right hand side of (4.8) we have an implication each way to the right hand side of (4.6).

We therefore have a generalisation of (4.3), that
\[
\langle a_1 \cdots a_n \rangle x = \langle b_1 \cdots b_n \rangle ((a_1 b_1) \cdots (a_n b_n) \cdot x)
\]
where \( b_1, \ldots, b_n \in \mathbb{A}^{(n)} \) is a list of distinct atoms each fresh for \((a_1, \ldots, a_n, x, X)\).
4.2 NEL with binders

In this section we will modify the definitions of a NEL-signature $\Sigma$ (Definition 3.1.1), $\Sigma$-structure (Definition 3.1.4), well-sorted terms (Definition 3.2.3) and their values (Definition 3.2.7) to include operation symbols that bind atoms in their arguments. The next section will show that this addition does not in fact add expressivity.

**Definition 4.2.1.** A binding NEL-signature $\Sigma$ is specified by

- a nominal set $\text{Sort}_\Sigma$, whose elements are called the *sorts of $\Sigma$*;
- a nominal set $\text{Op}_\Sigma$, whose elements are called the *operation symbols of $\Sigma$*;
- an equivariant *typing function* assigning to each $\text{op} \in \text{Op}_\Sigma$ a type consisting of a finite (possibly empty) list of pairs $(m, s)$, where $m \in \mathbb{N}$ and $s \in \text{Sort}_\Sigma$, and a result sort $s$. We will write this
  \[ \text{op} : [\langle m_1 \rangle s_1, \ldots, \langle m_n \rangle s_n] \to s . \] (4.11)

  The intended meaning is that $\text{op}$ binds $m_i$ atoms in its $i$'th argument for $1 \leq i \leq n$.

In practice we would often want this binding structure to itself be sorted, so that rather than using integers $m_i$ we use lists of sorts. To do this we would use atom sorts. However in this section we will continue to follow Remark 2.1.2 and leave atom sorts out of our account, as their addition would not materially effect the underlying mathematics.

**Example 4.2.2.** Example 3.1.2 defined the untyped $\lambda$-calculus with a nominal set of operation symbols $\{ L_a \mid a \in A \}$ representing $\lambda$-abstraction, each having the type $[\text{tm}] \to \text{tm}$. Using the above definition we could instead define one operation symbol, $L : [\langle 1 \rangle \text{tm}] \to \text{tm}$. Application would have the type $[\langle 0 \rangle \text{tm}, \langle 0 \rangle \text{tm}] \to \text{tm}$, while the definition for variables need not be changed.

**Definition 4.2.3.** Given a binding NEL-signature $\Sigma$, a *$\Sigma$-structure* $M$ in the category $\mathcal{FM}$-$\text{Set}$ is specified by

- an equivariant map from each sort $s \in \text{Sort}_\Sigma$ to an FM-set $M[s]$;
- an equivariant map from each operation symbol $\text{op} \in \text{Op}_\Sigma$ with type (4.11) to an FM-function
  \[ M[\text{op}] : [A^{m_1}]M[s_1] \times \cdots \times [A^{m_n}]M[s_n] \to M[s] . \] (4.12)

**Definition 4.2.4.** Given a binding NEL-signature $\Sigma$, the sets $\Sigma_\pi(\Gamma)$ of *terms of sort* $s \in \text{Sort}_\Sigma$ *in a sorting environment* $\Gamma$ are inductively defined by:

- if $\pi \in \text{Perm}$, $x \in \text{dom}(\Gamma)$ and $\Gamma(x) = s$, then $\pi x \in \Sigma_{\pi-s}(\Gamma)$;
The Definition 4.2.5. This section defines a translation $4.3$ Translation to NEL without binders

This section defines a translation $T$ from binding NEL-signatures, and structures, terms and theories for such signatures, to the signatures, structures, terms and theories of Chapter 3. It then provides an extended example of this translation for name-for-name substitution.

Definition 4.3.1. Given a binding NEL-signature $\Sigma$, define the NEL-signature $T(\Sigma)$ by

- $\text{Sort}_{T(\Sigma)} \triangleq \text{Sort}_\Sigma$;

- $\text{Op}_{T(\Sigma)}$ replaces each $\text{op} : \langle m_1 \rangle s_1, \ldots, \langle m_n \rangle s_n \rightarrow s \in \text{Op}_\Sigma$ with the set

\[
\{ \text{op}_{\vec{a}_1, \ldots, \vec{a}_n} \mid \vec{a}_i \in A_1, \text{supp}(\vec{a}_i) \neq s_i \text{ for } 1 \leq i \leq n \}.
\]  

All operation symbols in this set have type $[s_1, \ldots, s_n] \rightarrow s$. Defining the $\text{Perm}$-action on $\text{Op}_{T(\Sigma)}$ as

\[
\pi \cdot (\text{op}_{\vec{a}_1, \ldots, \vec{a}_n}) \triangleq (\pi \cdot \text{op})_{\pi \vec{a}_1, \ldots, \pi \vec{a}_n}
\]  

ensures that $\text{Op}_{T(\Sigma)}$ is a nominal set and the typing function is equivariant.

Given a $\Sigma$-structure $M$, where $\Sigma$ is a binding NEL-signature, define the $T(\Sigma)$-structure $T(M)$ by

- $T(M)[s] \triangleq M[s]$ for all $s \in \text{Sort}_\Sigma = \text{Sort}_{T(\Sigma)}$;

- Given $\text{op}_{\vec{a}_1, \ldots, \vec{a}_n} : [s_1, \ldots, s_n] \rightarrow s$ as in (4.14) and $x_i \in T(M)[s_i] = M[s_i]$ for $1 \leq i \leq n$,

\[
T(M)[\text{op}_{\vec{a}_1, \ldots, \vec{a}_n}](x_1, \ldots, x_n) \triangleq M[\text{op}](\langle \vec{a}_1 \rangle x_1, \ldots, \langle \vec{a}_n \rangle x_n).
\]  

The map $\text{op}_{\vec{a}_1, \ldots, \vec{a}_n} \mapsto T(M)[\text{op}_{\vec{a}_1, \ldots, \vec{a}_n}]$ is equivariant because the map $\text{op} \mapsto M[\text{op}]$ is by Definition 4.2.3.

Given a term $t \in \Sigma_\ell(\Gamma)$, where $\Sigma$ is a binding NEL-signature, define $T(t) \in T(\Sigma)_s(\Gamma)$ by

\[
T(\pi x) \triangleq \pi x \\
T(\text{op}_{\vec{a}_1} t_1 \cdots \langle \vec{a}_n \rangle t_n) \triangleq \text{op}_{\vec{a}_1, \ldots, \vec{a}_n} T(t_1) \cdots T(t_n).
\]  

(4.17)
Lemma 4.3.2. Take a binding signature $\Sigma$, sorting environment $\Gamma \in \mathsf{SE}_\Sigma$, term $t \in \Sigma(\Gamma)$, $\Sigma$-structure $M$ and valuation $\rho \in M[\Gamma]$. Then

$$T(M)[[T(t)]]\rho = M[[t][\rho]].$$

Proof. Follows by induction on the structure of $t$ and the definitions of value (Definitions 3.2.7 and 4.2.5). \qed

Definition 4.3.3. Given a theory $T$ for a binding signature $\Sigma$ we define a theory $T(T)$ for $T(\Sigma)$ as follows. First we replace each axiom $\nabla \vdash \exists$ $t \approx t'$ of $T$ with the axiom

$$\nabla \vdash \exists t \approx t': s \quad (4.18)$$

Then we add a binding axiom for each $op_{\bar{a}_1, \ldots, \bar{a}_n} : [s_1, \ldots, s_n] \to s \in Op_\Sigma$, where $\bar{a}_i = (a_i^1, \ldots, a_i^n)$ for $i \leq 1 \leq n$, as follows. Take a list of fresh tuples $\bar{b}_1, \ldots, \bar{b}_n$, where $\bar{b}_i = (b_i^1, \ldots, b_i^m) \in A^{(m_i)}$ and $supp(\bar{b}_i) \neq \langle op, \bar{a}_i, s_i \rangle$ for $1 \leq i \leq n$. Then the binding axiom is

$$supp(\bar{b}_1) \neq \# x_1 : s_1, \ldots, supp(\bar{b}_n) \neq \# x_n : s_n \vdash op_{\bar{a}_1, \ldots, \bar{a}_n} x_1 \cdots x_n \approx op_{\bar{b}_1, \ldots, \bar{b}_n} ((a_1^1 \ b_1^1) \cdots (a_1^n \ b_1^n) \ x_1) \cdots ((a_n^1 \ b_n^1) \cdots (a_n^n \ b_n^n) \ x_n) : s \quad (4.19)$$

(4.19) looks extremely complicated, but that is only because it is robust enough to handle any binding structure in the arguments of an operation symbol. In practice binding structures will tend to be simpler, as in the following example.

Example 4.3.4. In Example 4.2.2 we defined a binding NEL-signature $\Sigma$ for the untyped $\lambda$-calculus, in particular representing $\lambda$-abstractions by the operation symbol $L : [\langle 1 \rangle \text{tm}] \to \text{tm}$. This translates to a NEL-signature $T(\Sigma)$ with the nominal set of operation symbols $\{L_a : a \in A\}$ each having type $[\text{tm}] \to \text{tm}$, as with Example 3.1.2. The binding axioms for $L$ are, following (4.19),

$$b \neq \# x : \text{tm} \vdash L_a x \approx L_b (a \ b) x : \text{tm} \quad (4.20)$$

for each $a \in A$ and picking any $b \neq a$. But any structure satisfies (4.20) if and only if it satisfies

$$x : \text{tm} \vdash a \neq \# L_a x : \text{tm}. \quad (4.21)$$

Let $\rho$ be any member of the nominal set $[[\text{tm}]]$. (4.21) is satisfied if and only if $a \neq [[L_a]](\rho)$. By Lemma 2.3.3 this holds if and only if $[[L_a]](\rho) = (a \ b) \cdot ([[L_a]](\rho))$, which equals $[[L_a]]((a \ b) \cdot \rho)$ by (3.10), for $b \neq (a, \rho)$. But $[[L_a]](\rho) = [[L_a x]](\rho)$ and $[[L_a]]((a \ b) \cdot \rho) = [L_b (a \ b) x] \rho$, so this is exactly the test for satisfaction of (4.20).

By Lemma 3.4.8, if (4.21) is satisfied for some atom $a$ then it is satisfied for any atom via equivariance, so we only need that one axiom to capture the concept of $\alpha$-equivalence for $\lambda$-abstractions. Indeed, this is the axiom $(\alpha)$ we used in Example 3.4.6. We need no new binding axioms for variables and applications as they do not bind any atoms in their arguments.
Lemma 4.3.5. If $\mathcal{T}$ is a theory for a binding NEL-signature $\Sigma$ and $M$ is a $\mathcal{T}$-algebra, then $T(M)$ is a $T(\mathcal{T})$-algebra.

Proof: If $\nabla \vdash \alpha \not\equiv t \approx t' : s$ is an axiom of $\mathcal{T}$ then (4.18) is an axiom of $T(\mathcal{T})$. Take $\rho \in T(M)[\nabla] = M[\nabla], \alpha \# M[t]\rho = M[t']\rho$ because $M$ is a $\mathcal{T}$-algebra, so by Lemma 4.3.2 $\alpha \# T(M)[T(t)]\rho = T(M)[T(t')]\rho$.

We now turn to the binding axioms (4.19). Given appropriate $\rho$ the value of the left hand side is $T(M)[\langle \alpha \rangle](\rho(x_1),\ldots) = M[\langle \alpha \rangle](\rho(x_1),\ldots)$, while the right hand side is equal to $M[\langle \alpha \rangle](\rho(x_1),\ldots)$. These are equal by (4.10). ☐

So that our translation $T$ is sound and complete, we also need to be able to translate algebras in the other direction, from a $T(\mathcal{T})$-algebra $M$ to a $\mathcal{T}$-algebra $U(M)$:

Lemma 4.3.6. Suppose that $\Sigma$ is a binding NEL-signature and that $M$ is a $T(\Sigma)$-structure. Then we can define a $\Sigma$-structure $U(M)$ by

- $U(M)[s] \triangleq M[s]$
- Given $\alpha : [(m_1)s_1,\ldots,(m_n)s_n] \rightarrow s$, $x_i \in U(M)[s_i] = M[s_i]$ and $\vec{a}_i \in A^{m_i}$ such that $\text{supp}(\vec{a}_i) \# s_i$ for $1 \leq i \leq n$,

$$U(M)[\langle \alpha \rangle](x_1,\ldots,x_n) \triangleq M[\langle \vec{a}_1 \rangle](x_1,\ldots,x_n) .$$  \hspace{1cm} (4.22)

Proof. The step that requires proof is that (4.22) defines a function. So suppose we have $\vec{a}_i = (a_1,\ldots,a_{m_i})$ and $\vec{b}_i = (b_1,\ldots,b_{m_i})$ so that $\langle \vec{a}_i \rangle x_i = \langle \vec{b}_i \rangle y_i$ for $1 \leq i \leq n$. We need to prove that $M[\langle \vec{a}_1 \rangle,\ldots,\vec{a}_n](x_1,\ldots,x_n) = M[\langle \vec{b}_1 \rangle,\ldots,\vec{b}_n](y_1,\ldots,y_n)$.

By Lemma 4.1.6(ii)

$$(a_1^1 c_1^1) \cdots (a_{m_i}^1 c_{m_i}^1) \cdot x_i = (b_1^1 c_1^1) \cdots (b_{m_i}^1 c_{m_i}^1) \cdot y_i$$ \hspace{1cm} (4.23)

for suitably fresh $\vec{c}_i = [c_1^1,\ldots,c_{m_i}^1]$. Now

$$M[\langle \vec{a}_1 \rangle,\ldots,\vec{a}_n](x_1,\ldots) = M[\langle \vec{c}_1 \rangle,\ldots,\vec{c}_n]((a_1^1 c_1^1) \cdots (a_{m_i}^1 c_{m_i}^1) \cdot x_1,\ldots)$$ \hspace{1cm} (4.24)

because $M$ satisfies the relevant binding axiom. This equals

$$M[\langle \vec{c}_1 \rangle,\ldots,\vec{c}_n]((b_1^1 c_1^1) \cdots (b_{m_i}^1 c_{m_i}^1) \cdot y_1,\ldots)$$ \hspace{1cm} (4.25)

by (4.23), which equals $M[\langle \vec{b}_1 \rangle,\ldots,\vec{b}_n](y_1,\ldots)$ by satisfaction of another binding axiom. ☐

Lemma 4.3.7. [ref. Lemmas. 4.3.2 and 4.3.5] Take a binding signature $\Sigma$. Then

(i) Given a sorting environment $\Gamma \in \text{SE}_\Sigma$, term $t \in \Sigma(\Gamma)$, $T(\Sigma)$-structure $M$ and valuation $\rho \in M[\Gamma]$, then

$$U(M)[t]\rho = M[T(t)]\rho .$$
(ii) If $T$ is a theory for $\Sigma$ and $M$ is a $T(\Sigma)$-algebra then $U(M)$ is a $\Sigma$-algebra.

Proof. (i) follows by induction on the structure of $t$, and (ii) follows by (4.18) and (i). \qed

The final theorem of this section shows that given any theory for a binding NEL-signature, our translation $T$ to a standard NEL-signature respects the semantic consequence relation of Definition 3.4.7.

**Theorem 4.3.8.** Given any theory $\Pi$ for a binding signature $\Sigma$ and any NEL-judgement $\nabla \vdash_\Pi \bar{x} \not\equiv t \approx t' : s$ over $\Sigma$,

$$\nabla \vdash_{T(\Pi)} \bar{x} \not\equiv T(t) \equiv T(t') : s.$$ 

Proof. Right-to-left: Take any $\Pi$-algebra $M$ and $\rho \in M[\llbracket \nabla \rrbracket]$. $T(M)$ is a $T(\Pi)$-algebra by Lemma 4.3.5, so $\bar{x} \not\equiv T(M)[T(t)]\rho = T(M)[T(t')]\rho$, which gives our result by Lemma 4.3.2. The converse holds similarly, by Lemma 4.3.7. \qed

**Example 4.3.9.** We will now demonstrate this translation from binding to non-binding signatures on the theory of Nominal Substitution [19, Section 1.4], which provides axioms for name-for-name substitution. Using the notation of Section 4.2, this theory has one sort $tm$ and a nominal set of operation symbols

$$\{sub_a : [\langle 1 \rangle tm] \rightarrow tm \mid a \in A\}$$

with the evident Perm-action $\pi \cdot sub_a = sub_{\pi(a)}$. We read the term $sub_b \langle a \rangle t$ as ‘map $a$ to $b$ in $t$’, so $a$ is bound in $t$ while $b$ need not be. Translating this operation symbol to NEL without binders gives us the nominal set

$$\{sub_{b/a} : [tm] \rightarrow tm \mid (b, a) \in A \times A\}.$$ 

Note that $b$ and $a$ may be equal or distinct atoms. These operation symbols have the binding axioms

(i) **Binding-1:** $c \not\equiv x \vdash sub_{b/a} x \approx sub_{b/c} (a \ c) x$;

(ii) **Binding-2:** $c \not\equiv x \vdash sub_{a/a} x \approx sub_{a/c} (a \ c) x$.

for all distinct $a$ and $b$, choosing some fresh $c \notin \{a, b\}$ for each axiom. Note that the Binding-1 axiom is equivalent to

$$x \vdash a \not\equiv sub_{b/a} x$$

by reasoning identical to the proof that (4.20) is equivalent to (4.21) in Example 4.3.4. No such simplification is possible for the second binding axiom at this stage.

The theory presented in [19] has four axioms, but these are presented in a schematic manner rather than in terms of literal atoms (see Remark 3.4.9 for a discussion of this). For example the
Contraction axiom \( x \vdash \text{sub}_{c/b} \text{sub}_{b/a} x \approx \text{sub}_{c/b} \text{sub}_{c/a} x \) is in fact five axioms, covering the cases where \( a, b \) and \( c \) are all distinct atoms, where they are all the same atom, and the three cases where two are the same and one different. However many of these axioms are consequences of other axioms. For example, the case where \( a = b = c \) is a trivial consequence of the Identity axiom. In fact only the cases where all three atoms are distinct and where \( a = c \neq b \) are needed.

The entire theory of Nominal Substitution can be presented in seven axioms:

(i) Binding: \( x \vdash a \approx \text{sslash}_{b/a} x \);

(ii) Identity: \( x \vdash \text{sub}_{a/a} x \approx x \);

(iii) Weakening: \( a \neq x \vdash \text{sub}_{b/a} x \approx x \);

(iv) Contraction-1: \( x \vdash \text{sub}_{c/b} \text{sub}_{b/a} x \approx \text{sub}_{c/b} \text{sub}_{c/a} x \);

(v) Contraction-2: \( x \vdash \text{sub}_{a/b} \text{sub}_{b/a} x \approx \text{sub}_{a/b} x \);

(vi) Permutation-1: \( x \vdash \text{sub}_{d/b} \text{sub}_{c/a} x \approx \text{sub}_{c/a} \text{sub}_{d/b} x \);

(vii) Permutation-2: \( x \vdash \text{sub}_{c/b} \text{sub}_{c/a} x \approx \text{sub}_{c/a} \text{sub}_{c/b} x \).

The axiom Contraction-2 is derived by setting \( a = c \) in the Contraction axiom, while the axiom Permutation-2 is derived by setting \( c = d \) in the Permutation axiom.

But what has happened to the Binding-2 axiom? In fact, it too is a consequence of the other axioms. To see this, first we will prove that any structure that satisfies the seven axioms above also satisfies

\[
b \neq x \vdash \text{sub}_{b/a} x \approx (b a) x . \tag{4.29}
\]

Take some \( \rho \) in the nominal set \( \llbracket \text{tm} \rrbracket \) such that \( b \neq \rho \), and take same other atom \( c \neq (a, b, \rho) \).

Note that \( c \neq (\llbracket \text{sub}_{b/a} \rrbracket(\rho), (b a) \cdot \rho) \), a fact we will use to apply the Weakening axiom. Then

\[
\llbracket \text{sub}_{b/a} \rrbracket(\rho) = \llbracket \text{sub}_{b/c} \rrbracket(\llbracket \text{sub}_{b/a} \rrbracket(\rho)) \quad \text{(Weakening)}
\]
\[
= \llbracket \text{sub}_{b/c} \rrbracket(\llbracket \text{sub}_{c/a} \rrbracket(\rho)) \quad \text{(Contraction-1)}
\]
\[
= \llbracket \text{sub}_{b/c} \rrbracket(\llbracket \text{sub}_{c/b} \rrbracket((b a) \cdot \rho)) \quad \text{(Binding-1)}
\]
\[
= \llbracket \text{sub}_{b/c} \rrbracket((b a) \cdot \rho) \quad \text{(Contraction-2)}
\]
\[
= (b a) \cdot \rho . \quad \text{(Weakening)}
\]

This shows that (4.29) is satisfied. Now take \( \rho \in \llbracket \text{tm} \rrbracket \) where \( c \neq \rho \).

\[
\llbracket \text{sub}_{a/a} \rrbracket(\rho) = \rho \quad \text{(Identity)}
\]
\[
= \llbracket \text{sub}_{a/c} \rrbracket(\rho) \quad \text{(Weakening)}
\]
\[
= \llbracket \text{sub}_{a/c} \rrbracket(\llbracket \text{sub}_{c/a} \rrbracket(\rho)) \quad \text{(Contraction-2)}
\]
\[
= \llbracket \text{sub}_{a/c} \rrbracket((a c) \cdot \rho) . \tag{4.29}
\]

So any structure satisfying the seven axioms also satisfies Binding-2, so it is not needed as an axiom.
4.4 Other notions of binding

The previous section demonstrated that adding explicit support for binding to the Nominal Equational Logic framework of Chapter 3 does not increase its expressivity. But is adding binding operation symbols to NEL and then demonstrating that they are not needed sufficient to show that NEL can deal with the variations on binding that exist ‘in the wild’? To answer that question, this section will investigate some of the formalisms for explicitly representing binding that have been used elsewhere in the literature.

Example 4.4.1. The binding signatures of [18] consist of a set of operation symbols $O$ and a typing function $O \rightarrow \mathbb{N}^*$ sending each operation symbol to a finite list of numbers. An operation symbol of type $(m_1, \ldots, m_n)$ has $n$ arguments and binds $m_i$ atoms in its $i$'th argument. Terms have the form

$$a \mid op\langle \bar{a}_1 \rangle t_1 \cdots \langle \bar{a}_n \rangle t_n$$

(4.30)

for all $a \in A$, operation symbols $op$ with type $(m_1, \ldots, m_n)$ and lists of atoms $\bar{a}_i \in \mathbb{A}^{m_i}$ and terms $t_i$ for $1 \leq i \leq n$.

Over the constructed terms alone the binding signature terms are a special case of NEL with binders, where there is only one sort and the operation symbols form a set, rather than a nominal set. But binding signatures also use atoms directly as terms. If we were to give values to these terms in a nominal set $X$ then we would interpret an atom term by an equivariant function $\mathbb{A} \rightarrow X$. But this is the same way we would interpret the nominal set $\{V_a \mid a \in \mathbb{A}\}$ of constants, so replacing each subterm $a$ by the constant $V_a$ gives a sound and complete translation from binding signatures to NEL signatures.

Example 4.4.2. The nominal signatures of [61] differ from both binding NEL-signatures and the previous example in that binding is described through the sort structure and with explicit atom abstraction terms, rather than through the operation symbols. The sorts of a nominal signature start with a set of base sorts with typical member $b$, and are defined by

$$s ::= b \mid 1 \mid s \times s \mid A \mid [A]s \mid .$$

(4.31)

A nominal signature specifies the base sorts and a set of operation symbols, each with type $s \rightarrow b$, where $s$ is as in (4.31) and $b$ is a base sort. The terms then have the following form and sort:

- **Unit:** $(): 1$;
- **Pair:** $(t_1, t_2) : s_1 \times s_2$ if $t_1 : s_1$ and $t_2 : s_2$;
- **Constructed term:** $op t : b$ if $op : s \rightarrow b$ and $t : s$;
- **Atom:** $a : \mathbb{A}$ for $a \in \mathbb{A}$. In fact [61] divides $\mathbb{A}$ into different atom sorts, but as discussed in Remark 2.1.2 this extension does not materially change the account we give here, so is left out in the interests of conciseness;
CHAPTER 4. BINDING

- **Atom abstraction**: \( \langle a \rangle t : [A]s \) if \( a \in A \) and \( t : s \);

- **Suspension**: \( \pi x : s \) if \( \pi \in \text{Perm} \) and \( x \) is given sort \( s \) by a sorting environment, where \( s \) is a base sort or \( A \).

The condition on suspensions requires that variables be of atom or base sort. In fact, while variables of atom sort may be convenient they do not add expressivity; a judgement whose freshness environment contains \( \pi \not\equiv x : A \) can be replaced in a theory by judgements without that assertion in the freshness environment, applying the substitution \( \{a/x\} \) to the judgement’s terms for each \( a \in A - \pi \). So from now on we will disregard variables of sort \( A \).

An algebra \( M \) for such a nominal signature involves first assigning a nominal set \( M[b] \) to each base sort \( b \), allowing us to define \([s]\) (we as usual omit the algebra’s name where possible) to each sort \( s \):

\[
[1] \triangleq \{\ast\}, \quad [s \times s'] \triangleq [s] \times [s'], \quad [A] \triangleq A, \quad [[A]s] \triangleq [A][s]. \tag{4.32}
\]

We also assign an equivariant function \( M[op] : M[s] \to M[b] \) to each \( op : s \to b \). Now, given a valuation \( \rho \) of variables of sort \( s \) to members of \([s]\) we define the values of terms by

\[
\begin{align*}
[()] & \triangleq \ast, \quad [(t_1, t_2)] \triangleq ([t_1] \rho, [t_2] \rho), \quad [op \ t] \rho \triangleq [op]([t] \rho), \\
[\langle a \rangle t] \rho & \triangleq [op]([t] \rho), \quad [(\pi x) \ t] \rho \triangleq \pi \cdot (\rho(x)). \tag{4.33}
\end{align*}
\]

We may now follow Section 4.3 and define a translation \( T \) from the world of nominal signatures to Nominal Equational Logic; we will omit the details that recapitulate that section and focus on the differences.

Given a nominal signature \( \Sigma \) we define the NEL-signature \( T(\Sigma) \) as follows: set \( \text{Sort}_{T(\Sigma)} \) to be the sorts enumerated by (4.31) and \( \text{Op}_{T(\Sigma)} \) to be the set of operation symbols of \( \Sigma \) along with the new symbols

- **unit** : \( 1 \);
- **pair** : \([s, s'] \to s \times s'\) for all sorts \( s, s' \);
- \( \{\text{atm}_a \mid a \in A\} : A \) with the evident permutation action;
- \( \{\text{abs}_a \mid a \in A\} : [s] \to [A]s \) for all sorts \( s \), with the evident permutation action.

Hence given any \( \Sigma \)-term \( t \) we produce the \( T(\Sigma) \)-term \( T(t) \) by the obvious substitutions, replacing each \( () \) by \( \text{unit} \) and so forth.

Given an \( \Sigma \)-structure \( M \) for a nominal signature \( \Sigma \) we may define a \( T(\Sigma) \)-structure \( T(M) \) by following (4.32) for its sorts and setting \( T(M)[op] = M[op] \) for its operation symbols. Now given a \( \Sigma \)-theory \( \mathbb{T} \) for a nominal signature \( \Sigma \) we define the \( T(\Sigma) \)-theory \( T(\mathbb{T}) \) by replacing each axiom \( \nabla \vdash \pi \not\equiv t \approx t' : s \) by \( \nabla \vdash \pi \not\equiv T(t) \approx T(t') : s \) and adding the binding axioms

\[
x : s \vdash a \not\equiv \langle a \rangle x : [A]s \tag{4.34}
\]
for every sort $s$. Versions of Lemmas 4.3.2 and 4.3.5 for this translation $T$ follow.

Still following Section 4.3, we define a translation $U$ from $T(\Sigma)$-structures to $\Sigma$-structures by setting $U(M)[b] \triangleq M[b]$. We then define a family of equivariant functions $f_s : U(M)[s] \to M[s]$ as $s$ ranges over Sort$_T(\Sigma)$ as follows:

- $f_b$ is the identity on $U(M)[b] = M[b]$;
- $f_1(*) = M[\text{unit}]$;
- $f_{s_1 \times s_2}(x_1, x_2) = M[\text{pair}](f_{s_1}(x_1), f_{s_2}(x_2))$ for $x_i \in U(M)[s_i]$ and $i = 1, 2$;
- $f_{\text{atm}_a}(a) = M[\text{atm}_a] \in M[A]$ for all $a \in A$;
- $f_{[\text{abs}_a]}((a) : x) = M[\text{abs}_a](f(x))$ for all $x \in U(M)[s]$ and $a \in A$.

To see that $f_{[\text{abs}_a]}$ is a well-defined function requires the binding axiom (4.34). Using all this we can now fully define the structure $U(M)$ by

$$U(M)[op] \triangleq M[op] \circ f_s$$  \hspace{1cm} (4.35)

for all operation symbols $op : s \to b$ of $\Sigma$. We can then show by a routine induction that

$$M[T(t)]\rho = f_s(U(M)[t]\rho)$$  \hspace{1cm} (4.36)

where $t$ has sort $s$ according to some freshness environment and $\rho$ is a valuation for that environment. In the case where $s = b$, $f_b$ is the identity, so $M[T(t)]\rho = U(M)[t]\rho$, following Lemma 4.3.6. Therefore if $M$ is a $T(\mathbb{T})$-algebra then $U(M)$ is a $\mathbb{T}$-algebra so long as all the axioms of $\mathbb{T}$ have base sort. We can then prove a version of Theorem 4.3.8 provided again that the sorts of the axioms and the judgement in question are of base sort $b$.

Is this a reasonable restriction on the class of judgements over a nominal signature that we wish to consider? After all, our binding axioms (4.34) do not have base sort. Nonetheless, it is in keeping with the restrictions made in the original paper [61] on the sorts that variables may take and the result sorts of operation symbols. The justification for all these restrictions is that the base sorts represent the sorts that terms could take in some formal system that we are defining with our theory, while the other sorts are used only in constructing a term. Access to these sorts - in particular, the atom-abstraction sort - is useful in stating the metatheory of binding, but not for stating a theory directly. Therefore constraining judgements to have base sort is a reasonable way to define the notion of judgements in this setting and hence prove that the translation $T$ is sound and complete.

**Example 4.4.3.** A number of more exotic notions of binding exist, such as the binding specifications of Coq [48] and Ott [54], but these are designed to extend practical, rather than theoretical expressivity so can be translated into NEL-signatures at the cost of a loss of elegance. For example, Coq allows subterms within the scope of a binding operator to be exempted...
from that binding by tagging them with the \textit{scope specifier} outer (those that are not exempted are tagged with inner). So with a \textit{let} statement of the form

$$\text{let } a = t \text{ in } t'$$

(4.37)

where $a$ is bound in $t'$ but not in $t$, $t$ is tagged with outer while $t'$ is tagged with inner. This can be expressed, albeit less naturally, by pulling $t$ out of the abstraction:

$$\text{let } t \langle a \rangle t' .$$

(4.38)

Indeed this can always be done with expressions of Co\textsc{ml}, so NEL is indeed sufficiently expressive to deal with such ‘syntactic sugaring’.

### 4.5 Concretion

Recall that in Definition 4.1.3 we defined the FM-set of atom abstractions of elements of an FM-set $X$ as

$$[\mathbb{A}]X \triangleq \{ \langle a \rangle x \mid x \in X \land a \# X \}$$

but postponed the explanation of why we require that $a \# X$. The reason stems from our desire to have a \textit{concretion} operation on $[\mathbb{A}]X$ and atoms, sending atom abstractions back to members of their underlying set. However we will not in general be able to express concretion in Nominal Equational Logic, as its definition is not defined on all pairs of abstractions and atoms, but only those pairs given by the separated tensor of Example 2.3.8, and we have no way of expressing such a restriction within our current notion of signature. In Section 8.6 we will discuss how functions such as concretion might be accommodated within a NEL framework, but in this section we will show how far we can go in describing concretion with the logic we have.

\textbf{Definition 4.5.1.} [25, Definition 5.3] \textit{Concretion} is an equivariant function $[\mathbb{A}]X \otimes (\mathbb{A} - \text{supp}(X)) \rightarrow X$, whose action on $(y, b)$ is written $y \circ b$. In other words, by Definition 4.1.3, Example 2.3.8 and Lemma 4.1.4, $\langle a \rangle x \circ b$ is defined if $a, b \# X$ and either $a = b$ or $a \# x$. This action is defined by

$$\langle a \rangle x \circ b \triangleq (a b) \cdot x .$$

In particular, in the case that $a = b$, $\langle a \rangle x \circ a = x$.

The proof of next lemma finally motivates the restriction $a \# X$ of Definition 4.1.3.

\textbf{Lemma 4.5.2.} The concretion function $(\langle a \rangle x, b) \mapsto \langle a \rangle x \circ b$ is well-defined.

\textit{Proof.} There are three things that require proof: that concretion is equivariant, that the function respects the equivalence classes of $\langle a \rangle x$, and that it has as codomain the original FM-set $X$. 
(\pi \cdot (a) x @ \pi(b) = (\pi(a)) (\pi \cdot x) @ \pi(b) \) by (4.4), which is \((\pi(a) \pi(b)) \cdot \pi \cdot x = \pi \cdot (a b) \cdot (x)\), so concretion is equivariant.

By Corollary 4.1.5 we need to check that \(b \neq (a, x)\) implies \(a \neq (a) x @ b = (a b) \cdot x\); this follows by Lemma 2.3.4.

Finally, if \(a = b\) then \((a) x @ a = x \in X\). Otherwise \((a) x @ b = (a b) \cdot x \in (a b) \cdot X\), which equals \(X\) because \(a, b \neq X\). \(\Box\)

Now suppose we wished to define a NEL-theory for atom-abstraction and concretion over some nominal set of sorts. For each sort \(s\) we would define a new sort \(\llbracket A \rrbracket s\), along with operation symbols representing atom abstraction, \(\text{abs}_a : [s] \to [A]s\) for all \(a \neq s\). We could then define operation symbols for concretion likewise, \(\text{con}_a : \llbracket [A]s \rrbracket \to s\). But we would have to interpret \(\text{con}_a\) as a total function \(\llbracket [A]s \rrbracket \to \llbracket s \rrbracket\), when in fact it should only be partial. That is, we should only have terms of the sort \(\text{con}_a x\) when we can guarantee that \(a \neq x\); but NEL as presented in this dissertation offers no way of making such a restriction.

However, it turns out there are a number of examples of interesting FM-sets for which we can sensibly define \(\text{con}_a\) to be total.

**Definition 4.5.3.** A total concretion on an FM-set \(X\) is an equivariant function \(\llbracket A \rrbracket X \times (\mathbb{A} - \text{supp}(X)) \to X\) that agrees with Definition 4.5.1 on the separated tensor. So if we write the map on \((y, b)\) as \(y \mathbb{@} b\), then

\[
\begin{align*}
\text{if } b \neq & \langle a \rangle x \Rightarrow \langle a \rangle x \mathbb{@} b = (a b) \cdot x .
\tag{4.39}
\end{align*}
\]

**Example 4.5.4.** We now provide several examples of total concretions.

(i) We can define a total concretion on the nominal set of atoms \(\mathbb{A}\) by

\[
\langle b \rangle b \mathbb{@} a \triangleq a \
\text{if } b \neq c \Rightarrow \langle b \rangle c \mathbb{@} a \triangleq c .
\tag{4.40}
\]

The second equation includes the case that \(c = a\), which would not be defined by concretion in general as \(a \in \text{supp}((b)a) = \{a\}\).

(ii) If we have total concretions \(\mathbb{@} X, \mathbb{@} Y\) defined on FM-sets \(X, Y\) then we can define a total concretion on \(X \times Y\) by

\[
\langle a \rangle (x, y) \mathbb{@} b \triangleq (\langle a \rangle x \mathbb{@} X b, \langle a \rangle y \mathbb{@} Y b) .
\tag{4.41}
\]

(iii) An FM-set has a restriction structure [43] if there is an equivariant function \(\nu : \llbracket \mathbb{A} \rrbracket X \to X\) such that, for all \(x \in X\),

\[
\begin{align*}
\text{if } a \neq x \Rightarrow \nu\langle a \rangle x &= x \\
\nu\langle a \rangle \nu\langle a' \rangle x &= \nu\langle a' \rangle \nu\langle a \rangle x .
\end{align*}
\tag{4.42}
\]
Pitts has proved [46] that such an FM-set has a total concretion defined by

\[
(a)x \flow{a} x \triangleq x
\]

\[
a \neq b \Rightarrow \langle a \rangle x \flow{a b} \triangleq \nu(a)(a b) \cdot x .
\]  

We can use Corollary 4.1.5 to prove that this is well-defined on the equivalence classes of \( [\mathcal{A}] X \), as \( a \# (a)x \flow{a b} \neq \nu\langle a \rangle(a b) \cdot x \). To see that we have defined a total concretion, observe that if \( b \# (a, x) \) then \( \langle a \rangle x \flow{a b} = \nu\langle a \rangle(a b) \cdot x = (a b) \cdot x \) by (4.42) because \( a \# (a b) \cdot x \).

Note that this notion of restriction structure is not general enough to capture the above examples, as \( \mathcal{A} \) has no restriction structure: if \( \nu \) is equivariant then \( \nu\langle a \rangle a \) must have empty support, but no atom has empty support. However there is one example that is interesting for our purposes that does support a restriction structure:

(iv) The nominal set of permutations \( \text{Perm} \) has a restriction structure defined by

\[
\nu\langle a \rangle \pi \triangleq (a \pi(a))\pi .
\]  

This is well-defined by Corollary 4.1.5 because \( (a \pi(a))\pi(a) = a \), so \( a \# \nu\langle a \rangle \pi \). The first equation of (4.42) clearly holds, while the second equation follows by verifying that

\[
(a (a' \pi(a'))\pi(a))(a' \pi(a')) = (a' (a \pi(a))\pi(a'))(a \pi(a)) .
\]

This follows by a case analysis. If \( a = a' \) then (4.45) is \( (a \pi(a)) \). Otherwise, if \( \pi(a) = a' \) and \( \pi(a') = a \) then (4.45) is \( (a \pi(a'))\pi(a') = (a' \pi(a'))(a a') \). A similar equality holds if \( \pi(a') = a \) but \( \pi(a) \neq a' \). Finally, if \( a \neq a' \), \( \pi(a) \neq a' \) and \( \pi(a') \neq a \) then (4.45) is \( (a \pi(a))(a' \pi(a')) = (a' \pi(a'))(a \pi(a)) \).

We can hence follow (4.43) to define a total concretion on \( \text{Perm} \):

\[
\langle a \rangle \pi \flow{a} \pi
\]

\[
a \neq b \Rightarrow \langle a \rangle \pi \flow{a b} \triangleq (a (a b)\pi(b))(a b)\pi(a b) .
\]  

The bottom equation simply means we swap \( a \) and \( b \) in \( \pi \), then restrict \( a \) in the resulting permutation. Note that (4.44) is not unique; for example, we can define a restriction structure on \( \text{Perm} \) by

\[
a \# \pi \Rightarrow \nu\langle a \rangle \pi = \pi
\]

\[
a \in \text{supp}(\pi) \Rightarrow \nu\langle a \rangle \pi = \iota .
\]

This in turn induces a different total concretion on \( \text{Perm} \).

(v) By (ii) and (iv), if the nominal set of operation symbols \( \text{Op}_\Sigma \) for a signature \( \Sigma \) has a total concretion defined for it, then so does the nominal set of terms \( \text{Term}_\Sigma \) defined by Definition 3.2.3.

However, not all FM-sets can have total concretions defined for them:
Example 4.5.5. It is not possible to define a total concretion for the nominal set $\mathbb{A} \otimes \mathbb{A}$. To see this, consider the pair that would be defined by

$$\langle a \rangle (a, b) \# b$$  \hfill (4.48)

where $a \neq b$ by definition. $\langle a \rangle (a, b)$ has support $\{b\}$, so because concretion is equivariant, (4.48) must have support contained in $\{b\}$. But no member of $\mathbb{A} \otimes \mathbb{A}$ has a support set of less than two members.

We will now define a NEL-theory that will capture the notions of abstraction and total concretion, and hence characterise the FM-set of atom abstractions on any FM-set for which total concretion can be defined.

Definition 4.5.6. Given a nominal set of base sorts $B$, define a nominal set of sorts $\text{Sort}_\Sigma$ by

$$s ::= b \mid [A]s \quad (b \in B)$$  \hfill (4.49)

with the evident permutation action. Let $\text{Op}_\Sigma$ contain the operation symbols

$$\{\text{abs}_a \mid a \# s\} : [s] \rightarrow [A]s$$

$$\{\text{con}_a \mid a \# s\} : [A]s \rightarrow s$$

for all $s \in \text{Sort}_\Sigma$, with the evident permutation action. Then the theory of abstraction and total concretion, $\mathbb{T}$, has axioms

$$x : s \vdash a \# \text{abs}_a x : [A]s \quad \text{(Ax1)}$$

$$x : s \vdash \text{con}_a \text{abs}_a x \approx x : s \quad \text{(Ax2)}$$

$$b \# x : s \vdash \text{con}_b \text{abs}_a x \approx (a b) x : s \quad \text{(Ax3)}$$

$$a \# x : [A]s \vdash \text{abs}_a \text{con}_a x \approx x : [A]s \quad \text{(Ax4)}$$

Theorem 4.5.7. Given any $\mathbb{T}$-algebra for the theory of abstraction and total concretion defined above,

$$\llbracket [A]s \rrbracket \cong [A] \llbracket s \rrbracket$$

for all $s \in \text{Sort}_\Sigma$, and $\llbracket \text{abs}_a \rrbracket$ and $\llbracket \text{con}_a \rrbracket$ define abstraction and a total concretion respectively up to that isomorphism.

Proof. For each $\langle a \rangle x \in [A] \llbracket s \rrbracket$ map

$$\langle a \rangle x \mapsto \llbracket \text{abs}_a \rrbracket (x)$$  \hfill (4.50)

in $\llbracket [A]s \rrbracket$. This is well defined on the equivalence classes of $\langle a \rangle x$ by (Ax1). We can assume that $a \# s$, and hence that $\text{abs}_a$ is defined, because if it were not we could pick some $a' \# (a, s)$ and use the fact that $\langle a \rangle x = \langle a' \rangle (a a') \cdot x$ by (4.3).

Conversely, for each $x \in \llbracket [A]s \rrbracket$ take $a \# (x, s)$ and map

$$x \mapsto \langle a \rangle \llbracket \text{con}_a \rrbracket (x)$$  \hfill (4.51)
in \([A][s]\).

Now \(a \not\# \llbracket\text{abs}_a\rrbracket(x)\) by (Ax1), so we can apply the map (4.51) to get \(\langle a\rangle\llbracket\text{con}_a\rrbracket(\llbracket\text{abs}_a\rrbracket(x))\), which is \(\langle a\rangle x\) by (Ax2). Conversely, \(\llbracket\text{abs}_a\rrbracket(\llbracket\text{con}_a\rrbracket(x)) = x\) for \(a \not\# (x, s)\) by (Ax4).

The isomorphism sends abstraction by \(a\) to \(\llbracket\text{abs}_a\rrbracket\) by (4.50); \(\llbracket\text{con}_a\rrbracket\) defines a total concretion by (Ax2) and (Ax3).

**Theorem 4.5.8.** Suppose we have an equivariant map from each base sort \(s \in B\) to an FM-set \(\llbracket s \rrbracket\) equipped with a total concretion \(\oplus\). Then we can define a \(T\)-algebra for the theory of abstraction and total concretion by setting \(\llbracket [A]s \rrbracket \triangleq [A][s]\), \(\llbracket \text{abs}_a\rrbracket(x) \triangleq \langle a\rangle x\) and \(\llbracket \text{con}_b\rrbracket(y) = y \oplus b\).

**Proof.** (Ax1) asks that \(a \not\# \langle a\rangle x\); this holds by Lemma 4.1.4. (Ax2) and (Ax3) hold by (4.39). (Ax4) asks us to show that if \(a\) is fresh for some abstraction \(y\) then \(\langle a\rangle(\langle a\rangle x \oplus a) = y\). Because of the freshness condition this is a property of concretion, and indeed is proved for concretion in [25, Proposition 5.5]: if \(y = \langle a\rangle x\) then \(\langle a\rangle(\langle a\rangle x \oplus a) = \langle a\rangle x\), while if \(y = \langle a'\rangle x\) and \(a \not\# x\) then \(\langle a\rangle(\langle a'\rangle x \oplus a) = \langle a\rangle(a a') \cdot x\), which is \(\langle a'\rangle x\) by (4.3).
Chapter 5

Proof theory

This chapter will provide proof rules for NEL that are sound and complete with respect to the semantic consequence relation of Definition 3.4.7. Section 5.5 will then show that allowing freshness judgements on a term does not in fact increase the expressivity of the logic, though such judgements are certainly convenient to have, and will provide sound and complete rules for NEL that use equality judgements only.

5.1 Proof rules for Nominal Equational Logic

Figure 5.1 gives a collection of rule schemes for inductively generating judgements of the form \( \nabla \vdash \sigma \# t \approx t' : s \) for a NEL-signature \( \Sigma \).

**Notation 5.1.1.** Figure 5.1 makes use of the following notation introduced in earlier chapters.

- Rules (REFL), (#-EQUIVAR) and (SUSP) make use of the abbreviations for judgements introduced in Notation 3.4.4.

- Rules (REFL) and (WEAK) refer to the nominal set of freshness environments \( FE_\Sigma \) defined by Definition 3.4.1. That definition also introduced the sorting environment \( \nabla \), which is referred to by (REFL), (SUBST) and (ATM-INTRO).

- The rule (REFL) refers to the set of terms \( \Sigma_\sigma(\nabla) \) defined by Definition 3.2.3.

- (SUBST) makes the use of the set of substitutions \( \Sigma(\nabla', (\nabla')', \sigma) \) and term \( t\{\sigma\} \) defined by Definition 3.3.3.

- Rules (ATM-INTRO), (ATM-ELIM), (#-EQUIVAR) and (SUSP) make use of side-conditions concerning the semantic freshness relation of Definition 2.3.2. For example, (ATM-INTRO) asks that \( a \# (\nabla', \sigma, t, t') \), asserting freshness for the nominal set \( SE_\Sigma \times P_{fin}(\Lambda) \times Term_\Sigma \times Term_\Sigma \). As discussed in Example 2.3.8, this asks that \( a \) be fresh for each of the
(REFL)  \[ \nabla \vdash t \approx t : s \]

(TRANS)  \[ \nabla \vdash \pi_1 \not\equiv t \approx t' : s \quad \nabla \vdash \pi_2 \not\equiv t'' \approx t' : s \]

(SYM)  \[ \nabla \vdash \pi \not\equiv t \approx t' : s \]

(SUBL)  \[ \nabla' \vdash \sigma \approx \sigma' : \nabla \quad \nabla \vdash \pi \not\equiv t \approx t' : s \]

\[ \nabla' \vdash \pi \not\equiv t\{\sigma\} \approx t'\{\sigma'\} : s \quad \sigma, \sigma' \in \Sigma(\nabla, (\nabla')) \]

(WEA)  \[ \nabla \vdash \pi \not\equiv t \approx t' : s \]

\[ \nabla' \vdash \pi \not\equiv t \approx t' : s \quad \nabla' \in \ FE_\Sigma, \nabla \leq \nabla' \]

(ATM-INTRO)  \[ \nabla \vdash \pi \not\equiv t \approx t' : s \]

\[ \nabla^{\#a} \vdash \pi \not\equiv t \approx t' : s \quad a \# (\nabla', \pi, t, t') \]

(ATM-ELIM)  \[ \nabla^{\#a} \vdash \pi \not\equiv t \approx t' : s \]

\[ \nabla \vdash \pi \not\equiv t \approx t' : s \quad a \# (\nabla, \pi, t, t') \]

(#-EQUIVAR)  \[ \pi \not\equiv x : s \vdash \pi \cdot \pi \not\equiv x : \pi : s \quad a \# s \]

(SUSP)  \[ ds(\pi, \pi') \not\equiv x : s \vdash \pi x \approx \pi' x : \pi : s \]

\[ \nabla' \vdash \sigma \approx \sigma' : \nabla \quad (5.1) \]

Figure 5.1: The Rules of Nominal Equational Logic

members of the tuple individually; freshness for \( SE_\Sigma \) is defined by (3.15), for \( P_{fin}(\\check{A}) \) by Examples 2.2.5 and 2.2.8 (explicitly, \( a \# \pi \) if and only if \( a \notin \pi \)) and for \( \text{Term}_\Sigma \) by (3.18). (ATM-ELIM) uses freshness for \( FE_\Sigma \), which is defined by (3.35), while (#-EQUIVAR) and (SUSP) use freshness for \( \text{Sort}_\Sigma \), which could be any nominal set.

- The rule (SUSP) refers to the disagreement set \( ds(\pi, \pi') \) defined by (2.14).

Notation 5.1.2. Figure 5.1 makes use of the following new pieces of notation.

- In rule (SUBL)

\[ \nabla' \vdash \sigma \approx \sigma' : \nabla \quad (5.1) \]

stands for the finite number of hypotheses \( \nabla' \vdash \pi_i \not\equiv \sigma(x_i) \approx \sigma'(x_i) : s_i \) for \( 1 \leq i \leq n \), where \( \nabla = [\pi_1 \not\equiv x_1 : s_1, \ldots, \pi_n \not\equiv x_n : s_n] \).
CHAPTER 5. PROOF THEORY

53

The relation

\[ \nabla \leq \nabla' \]  \hspace{1cm} (5.2)

of *weakening* between freshness environments, which is used as a side condition in the rule (\textsc{weak}), is defined to hold if \( \nabla, \nabla' \) are freshness environments such that \( \text{dom}(\nabla) \subseteq \text{dom}(\nabla') \), and for all \( x \in \text{dom}(\nabla) \) if \( \nabla(x) = (s, \overline{a}) \) then \( \nabla'(x) = (s, \overline{a}') \) for some \( \overline{a}' \supseteq \overline{a} \).

- In rules (\textsc{atm-intro}) and (\textsc{atm-elim}), if \( \nabla = [\overline{a}_1 \not\approx x_1 : s_1, \ldots, \overline{a}_n \not\approx x_n : s_n] \) then

  \[ \nabla \not\approx a \triangleq [\overline{a}_1 \cup \{a\} \not\approx x_1 : s_1, \ldots, \overline{a}_n \cup \{a\} \not\approx x_n : s_n] . \]  \hspace{1cm} (5.3)

  More generally, if \( \overline{a} \) is a set of atoms and \( \overline{a} \not\approx \nabla \) then

  \[ \nabla \not\approx \overline{a} \triangleq [\overline{a}_1 \cup \overline{a} \not\approx x_1 : s_1, \ldots, \overline{a}_n \cup \overline{a} \not\approx x_n : s_n] . \]  \hspace{1cm} (5.4)

**Lemma 5.1.3.** The proof rules of Figure 5.1 preserve the well-formedness condition we placed on judgements in Definition 3.4.3, namely that we have a freshness environment in \( \text{FE}_\Sigma \) (Definition 3.4.1) to the left of the turnstile \( \vdash \), that both terms are assigned the given sort by the sorting environment associated with that freshness environment (following Definition 3.2.3) and that the set of atoms to the right of the turnstile is fresh for that sort.

**Proof.** The rules (\textsc{refl}), (\textsc{symm}), (\textsc{trans}) and (\textsc{atm-elim}) and (\textsc{#{equivar}}) clearly preserve well-formedness. (\textsc{subst}) preserves well-formedness by (3.27). We can see that (\textsc{weak}) preserves well-formedness by the observation that the sort a term is assigned can vary only if the sorting of the variables in the term varies, so \( \Gamma \leq \Gamma' \) implies \( \Sigma_s(\Gamma) \subseteq \Sigma_s(\Gamma') \).

For (\textsc{atm-intro}) we need to confirm that \( \nabla \not\approx a \in \text{FE}_\Sigma \) in that if \( \nabla \not\approx a(x_i) = (s, \overline{a}_i \cup \{a\}) \) then \( \overline{a}_i \cup \{a\} \not\approx s_i \). We know that \( \overline{a}_i \not\approx s_i \) because \( \nabla \) is a freshness environment, and we know that \( a \not\approx s_i \), because \( a \not\approx \nabla \). Also, we observe that \( a \not\approx s \) by \( a \not\approx (\nabla, t) \), Lemma 3.2.5 and (2.16).

For (\textsc{susp}) we need to confirm that \( \pi' \cdot x \in \Sigma_{\pi' \cdot s}(x : s) \); that is, that \( \pi \cdot s = \pi' \cdot s \). This follows by Corollary 2.3.7 and the side condition \( ds(\pi, \pi') \not\approx s \). \hfill \Box

**Definition 5.1.4.** (Logical Consequence) The set of *theorems* of a NEL-theory \( T \) is the least set of judgements containing the axioms of \( T \) and closed under the rules in Figure 5.1. We write

\[ \nabla \vdash_T \overline{a} \not\approx t \approx t' : s \]  \hspace{1cm} (5.5)

to indicate that the judgement is a theorem of \( T \) and call (5.5) the *logical consequence relation*.

We are going to show that the rules in Figure 5.1 are both sound and complete for the interpretation of judgements in \( \mathcal{F}\mathcal{M}\text{-Set} \). In other words, we will show that the logical consequence relation coincides with the semantic consequence relation of Definition 3.4.7. We may immediately proceed to the simpler property of soundness.
CHAPTER 5. PROOF THEORY

Theorem 5.1.5. (Soundness) If a judgement $\nabla \vdash \pi \not\approx t \approx t' : s$ is a theorem of a NEL-theory $T$, then it is satisfied by any $T$-algebra in $\mathcal{FM}$-Set:

$\nabla \vdash \pi \not\approx t \approx t' : s \Rightarrow \nabla \vdash \pi \not\approx t \approx t' : s$.

Proof. Given any $T$-algebra we must show that the collection of judgements satisfied (Definition 3.4.7) is closed under each of the rules in Figure 5.1.

Closure of satisfaction under rules (REFL), (SYMM) and (TRANS) is immediate from Definition 3.4.7.

For (SUBST) take a valuation $\rho \in [\nabla']$. Then, following Lemma 3.3.5, $[[\sigma]]\rho \in [[\nabla']]$ is the substitution mapping each $x \in \text{dom}(\nabla)$ to $[[\sigma(x)]]\rho$. Now if $\nabla = [\overline{a}_1 \neq x_1 : s_1, \ldots, \overline{a}_n \neq x_n : s_n]$ then for $1 \leq i \leq n$ we have $\overline{a}_i \neq [[\sigma(x_i)]]\rho = [[\sigma'(x_i)]]\rho$ by applying the induction hypothesis to the left hand premise. Therefore $[[\sigma]]\rho \in [[\nabla]]$, so $\overline{a} \not\approx [[t]]([[[\sigma]]\rho]) = [[t']]([[[\sigma]]\rho])$ by applying the induction hypothesis to the right hand premise. But, by Lemma 3.3.5, $[[t]]([[[\sigma]]\rho]) = [[t\{\sigma]\}]\rho$ so we have $\overline{a} \not\approx [[t\{\sigma]\}]\rho$ as required.

For (WEAK) we need only observe that if $\rho \in [[\nabla']]$ and $\nabla \leq \nabla'$ then $\rho \in [[\nabla]]$, so $\overline{a} \not\approx [[t]]\rho = [[t']]\rho$ by applying the induction hypothesis.

For (ATM-INTRO), if $\rho \in [[\nabla^{\#a}]$ then $\rho \in [[\nabla]]$, so we have $\overline{a} \not\approx [[t]]\rho = [[t']]\rho$ and all we need to prove is that $a \not\approx [[t]]\rho$. For all $x \in \text{dom}(\rho)$ we have $a \not\approx \rho(x)$, and hence by (3.20), $a \not\approx \rho$. If we also have $a \not\approx t$, then by (2.16) applied to the function $(t, \rho) \mapsto [[t]]\rho$ (which is equivariant by Lemma 3.2.9), we have $a \not\approx [[t]]\rho$ as required.\footnote{Rule (ATM-INTRO) also includes the inessential side-condition $a \notin \pi$, since if $a \notin \pi$ then the rule becomes an instance of (WEAK).}

For (ATM-ELIM), if $\rho \in [[\nabla]]$ then take some atom $a'$ such that $a' \not\in (\nabla, \rho, a, \overline{a}, t', t')$. Now if $\nabla = [\overline{a}_1 \neq x_1 : s_1, \ldots, \overline{a}_n \neq x_n : s_n]$ and $\nabla^{\#a} \in \mathcal{FE}_\Sigma$ then for $1 \leq i \leq n$ we have $a \not\approx s_i$. Now $(a \cdot a') \cdot \rho(x_i) \in (a' \cdot [s_i])$, where $(a \cdot a') \cdot [s_i] = [[a' \cdot a]]$ by the equivariance of the map $s \mapsto [s]$ defined by Definition 3.1.4. This equals $[s_i]$ because $a, a' \not\approx s_i$. Hence $(a \cdot a') \cdot \rho \in [[\nabla]]$. Further, $a, a' \not\approx \nabla$ implies $a, a' \not\approx \overline{a}_i$, so $\overline{a}_i \not\approx \rho(x_i)$ implies that $\overline{a}_i = (a' \cdot \overline{a}_i) \neq (a \cdot a') \cdot \rho(x_i)$. Therefore $(a \cdot a') \cdot \rho \in [[\nabla]]$. In fact $(a' \cdot a) \not\approx \nabla^{\#a}$ since $a \not\approx (a' \cdot a) \cdot \rho$ (by Lemma 2.3.4 applied to $a' \not\approx \rho$). So if $\nabla^{\#a} \vdash \pi \cup \{a\} \not\approx t \approx t' : s$ is satisfied then

$\overline{a} \not\approx [[t]]((a' \cdot a) \cdot \rho) = [[t']]((a') \cdot \rho)$.

By Lemma 3.2.9 the function $(t, \rho) \mapsto [[t]]\rho$ is equivariant, so we can apply $(a' \cdot a)$ to (5.6) and use Lemma 2.3.4 to get $\overline{a} \not\approx [[t]]\rho = [[t']]\rho$, as $a, a' \not\approx (\overline{a}, t, t')$.

Finally, closure under the rules (#-EQUIVAR) and (SUSP) follow directly from Lemma 2.3.4 and Corollary 2.3.7 respectively, along with the definition of $[[\pi x]]\rho$ given by (3.21).
5.2 Equivariance

Lemma 3.4.8 showed that the semantic consequence relation is equivariant. We wish logical consequence to coincide with semantic consequence; in order to prove this we must first show that it too is equivariant under the meta-level permutation action of Definition 3.2.4. To reach this conclusion, we first prove a lemma concerning the object-level permutation action of Definition 3.3.1:

**Lemma 5.2.1.** For any NEL-theories T, if

\[ \triangledown \vdash_{T} \overline{\pi} \not\equiv t \approx t' : s \]  

(5.7)

then for all \( \pi \in \text{Perm} \)

\[ \triangledown \vdash_{T} \pi \cdot \overline{\pi} \not\equiv \pi \ast t \approx \pi \ast t' : \pi \cdot s . \]  

(5.8)

**Proof.** Let \( \sigma, \sigma' \in \Sigma([x : s], \triangledown^\circ) \) be substitutions mapping \( x \) to \( t \) and \( t' \) respectively. Then apply the rule (\textsc{subst}):

\[
\begin{array}{c}
\triangledown \vdash \sigma \approx \sigma' : [\overline{\pi} \not\equiv x : s] \\
\overline{\pi} \not\equiv x : s \vdash \pi \cdot \overline{\pi} \not\equiv \pi x : \pi \cdot s \\
\hline
\triangledown \vdash \pi \cdot \overline{\pi} \not\equiv (\pi x)\{\sigma\} \approx (\pi x)\{\sigma'\} : \pi \cdot s
\end{array}
\]

(5.9)

The left hand premise is simply a different notation for (5.7) and the right hand premise is (\not\equiv\text{-equivar}). By (3.26) we have \( (\pi x)\{\sigma\} = \pi \ast \sigma(x) = \pi \ast t \), and similarly \( (\pi x)\{\sigma'\} = \pi \ast t' \).

**Theorem 5.2.2.** (Equivariance of Logical Consequence) For any NEL-theory T, if \( \triangledown \vdash_{T} \overline{\pi} \not\equiv t \approx t' : s \), then for all \( \pi \in \text{Perm} \), \( \pi \cdot \triangledown \vdash_{T} \pi \cdot \overline{\pi} \not\equiv \pi \ast t \approx \pi \ast t' : \pi \cdot s \).

**Proof.** Suppose that \( \triangledown = [\overline{a_1} \not\equiv x_1 : s_1, \ldots , \overline{a_n} \not\equiv x_n : s_n] \). Then \( \pi \cdot \overline{a_i} \not\equiv x_i : \pi \cdot s_i \vdash \overline{a_i} \not\equiv \pi^{-1} x_i : s_i \) for \( 1 \leq i \leq n \) by (\not\equiv\text{-equivar}), and we can apply (\textsc{weak}) to this to get \( \pi \cdot \triangledown \vdash \overline{a_i} \not\equiv \pi^{-1} x_i : s_i \). Therefore \( \pi \cdot \triangledown \vdash (\pi^{-1} -) \approx (\pi^{-1} -) : \triangledown \), where \( (\pi^{-1} -) \in \Sigma(\triangledown^\circ, \pi \cdot \triangledown^\circ) \) is the substitution introduced in Lemma 3.3.6 mapping each \( x \in \text{dom}(\triangledown) \) to \( \pi^{-1} x \).

Now \( \triangledown \vdash_{T} \overline{\pi} \not\equiv t \approx t' : s \) implies \( \triangledown \vdash_{T} \pi \cdot \overline{\pi} \not\equiv \pi \ast t \approx \pi \ast t' : \pi \cdot s \) by Lemma 5.2.1, so we can apply (\textsc{subst}:)

\[
\pi \cdot \triangledown \vdash (\pi^{-1} -) \approx (\pi^{-1} -) : \triangledown \vdash \pi \cdot \overline{\pi} \not\equiv \pi \ast t \approx \pi \ast t' : \pi \cdot s
\]

(5.10)

\((\pi \ast t)\{\pi^{-1} -\} = \pi \ast t \) and \((\pi \ast t')\{\pi^{-1} -\} = \pi \ast t' \) by Lemma 3.3.6.

**Remark 5.2.3.** (Theorems of T form a Nominal Set) Note that the set of judgements of the form (3.36) over a NEL-signature \( \Sigma \), once equipped with the atom-permutation action

\[
\pi \cdot (\triangledown \vdash_{T} \overline{\pi} \not\equiv t \approx t' : s) \triangleq (\pi \cdot \triangledown \vdash \pi \cdot \overline{\pi} \not\equiv \pi \cdot t \approx \pi \cdot t' : \pi \cdot s)
\]

(5.11)
forms a nominal set. The supports for such judgements are
\[
\text{supp}(\nabla \vdash \alpha \not\approx t' : s) = \text{supp}(\nabla) \cup \alpha \cup \text{supp}(t) \cup \text{supp}(t') \tag{5.12}
\]
using the support sets for freshness environments (3.35) and terms (3.18). (5.12) need not explicitly include \(\text{supp}(s)\), as it is contained within \(\text{supp}(\nabla) \cup \text{supp}(t)\) by Lemma 3.2.5 and (2.16).

The set of theorems for a NEL-theory \(\mathcal{T}\) is closed under the permutation action (5.11) by Theorem 5.2.2, so is itself a nominal set.

5.3 Ground term algebras

In this section we show how to form a \(\mathcal{T}\)-algebra \(\mathcal{M}_\mathcal{T}\) from the terms of a NEL-theory \(\mathcal{T}\) that do not involve any variables. The construction provides a stepping stone towards the completeness result of the next section.

Definition 5.3.1. (Ground Terms) Let \(\Sigma\) be a NEL-signature. The set of ground terms of sort \(s \in \text{Sort}_\Sigma\) over \(\Sigma\) is defined to be \(\Sigma_s(\emptyset)\), that is, the set of terms of sort \(s\) in the empty sorting environment \(\emptyset\). If \(t\) is a ground term then \(\text{Var}(t) = \emptyset\) (3.12), so \(t\) does not contain any suspensions \(\pi x\). The union of all \(\Sigma_s(\emptyset)\) where \(s\) ranges over \(\text{Sort}_\Sigma\) is denoted \(\Sigma(\emptyset)\).

Now let \(\mathcal{T}\) be a NEL-theory with signature \(\Sigma\). By virtue of the rules \((\text{REFL})\), \((\text{SYMM})\) and \((\text{TRANS})\) in Figure 5.1, the logical consequence relation of Definition 5.1.4 gives rise to an equivalence relation on \(\Sigma_s(\emptyset)\) that relates \(t\) and \(t'\) if \(\emptyset \vdash \mathcal{T} t \approx t' : s\). Let \(\mathcal{M}_\mathcal{T}[s]\) denote the quotient of \(\Sigma_s(\emptyset)\) by this equivalence relation. We write the equivalence class of \(t\) as \([t]\).

Recall from Definition 3.2.4 that the set of well-sorted \(\Sigma\)-terms \(\text{Term}_\Sigma\) is a nominal set once we endow it with the \(\text{Perm}\)-action \((\pi, t) \mapsto \pi \cdot t\) of (3.16). In the case of ground terms Lemma 3.3.6 implies that this action coincides with the object-level permutation action of (3.23):
\[
\forall t \in \Sigma_s(\emptyset). \; \pi \cdot t = \pi \cdot [t] \tag{5.13}
\]

Note that by Theorem 5.2.2, this \(\text{Perm}\)-action on \(\Sigma_s(\emptyset)\) preserves the equivalence relation \(\emptyset \vdash \mathcal{T} t \approx t' : s\). Hence we get a well-defined action on the quotient set \(\mathcal{M}_\mathcal{T}[s]\), defined by
\[
\pi \cdot [t] \triangleq [\pi \cdot t] = [\pi \cdot [t]] \tag{5.14}
\]

In fact, under this quotient \(\mathcal{M}_\mathcal{T}[s]\) forms an FM-set: if a finite set \(\overline{\alpha} \in \mathcal{P}_{\text{fin}}(\mathcal{A})\) supports \(t\) in \(\Sigma_s(\emptyset)\), then it also supports \([t]\) in \(\mathcal{M}_\mathcal{T}[s]\), because for any \(a, a' \notin \overline{\alpha}\), \((a a') \cdot [t] = [(a a') \cdot t] = [t]\). Thus
\[
a \not\approx t \Rightarrow a \not\approx [t] \tag{5.15}
\]

However, we can be more precise about the freshness relation for \(\mathcal{M}_\mathcal{T}[s]\). As the following lemma shows, when one restricts to ground terms the semantic notion of freshness coincides with the logical one determined by the rules in Figure 5.1.
Lemma 5.3.2. [Semantic Freshness = Ground Logical Freshness] For all $t \in \Sigma_\Delta(\emptyset)$ and $a \neq s$,
\[
a \neq [t] \in M_T[s] \iff \emptyset \vdash_T a \neq t : s.
\]

Proof. Take an atom $a'$ such that $a' \neq (a, t)$. By (5.15) we also have $a' \neq [t]$. Now by (REFL) and (ATM-INTRO) we have
\[
\emptyset \vdash_T a' \neq t : s
\]
and hence by ($\neq$-EQUIVAR) and (SUBST)
\[
\emptyset \vdash_T a \neq (a') : t : s.
\]

Suppose $a \neq [t]$ holds, so $(a') \cdot [t] = [t]$. By (5.14) $(a') \cdot [t] = [(a') \cdot t]$, so $[(a') \cdot t] = [t]$, so $\emptyset \vdash_T (a') \cdot t \approx t : s$. Applying (TRANS) and (SYMM) to this and (5.17) yields $\emptyset \vdash_T a \neq t : s$.

Conversely, if $\emptyset \vdash_T a \neq t : s$ holds, then by (ATM-INTRO) we have $\emptyset \vdash_T \{a, a'\} \neq t : s$; and hence by (SUSP) and (SUBST), $\emptyset \vdash_T (a) \cdot t \approx t : s$. In other words $[(a) \cdot t] = [t]$ and thus, as above, $(a') \cdot [t] = [t]$ so $a \neq [t]$.

Definition 5.3.3. Given a NEL-theory $T$ over a signature $\Sigma$, the ground term algebra $M_T$ is a $\Sigma$-structure defined by

- Mapping each $s \in \text{Sort}_\Sigma$ to $M_T[s]$. This is equivariant as required given the permutation action $\pi \cdot M_T[s] = M_T[\pi \cdot s]$;

- Given $op : s' \to s$ in $\text{Op}_\Sigma$, define $M_T[op] : M_T[s'] \to M_T[s]$ by
  \[
  M_T[op]([t_1], \ldots, [t_n]) \triangleq [op \, t_1 \cdots t_n].
  \] (5.18)

The fact that the function in (5.18) is well-defined (that is, $[op \, t_1 \cdots t_n]$ does not depend on the representative we choose from the equivalence classes $[t_1], \ldots, [t_n]$) follows by applying the rule (SUBST). The map $op \mapsto M_T[op]$ is equivariant by:

\[
(\pi \cdot M_T[op])([t_1], \ldots) = \pi \cdot (M_T[op](\pi^{-1} \cdot [t_1], \ldots)) = [\pi \cdot (op \, \pi^{-1} \cdot t_1) \cdots] \quad (2.18)
\]

\[
= [\pi \cdot (op \, t_1) \cdots] \quad (5.18) \text{and} \quad (5.14)
\]

\[
= [\pi \cdot op \, t_1 \cdots] \quad (3.16)
\]

\[
= M_T[\pi \cdot op]([t_1], \ldots) \quad (5.18).
\]

Lemma 5.3.4. Given a term $t \in \Sigma_\Delta(\Gamma)$ and a valuation $\rho \in M_T[\Gamma]$, let $\sigma \in \Delta(\Gamma, \emptyset)$ be a substitution that represents $\rho$ in the sense that $\rho(x) = [\sigma(x)]$ for all $x \in \text{dom}(\Gamma)$. Then
\[
M_T[t]\rho = [t\{\sigma\}] \quad .
\] (5.19)
Proof. By induction on the structure of $t$. First, the suspension case:

$$M_T \llbracket \pi \, x \rrbracket \rho = \pi \cdot \rho(x) \quad \text{by (3.21)}$$

$$= \pi \cdot [\sigma(x)]$$

$$= [\pi \ast \sigma(x)] \quad \text{by (5.14)}$$

$$= [\pi \, x \{\sigma\}] \quad \text{by (3.26)}.$$  

The constructed term case follows by (5.18).  

Theorem 5.3.5. (Ground Completeness) $M_T$ is a $\mathbb{T}$-algebra, that is, it satisfies all the axioms of $\mathbb{T}$ (and hence by the Soundness Theorem 5.1.5, all the theorems of $\mathbb{T}$). Furthermore, for ground terms, a judgement $\emptyset \vdash \bar{\pi} \neq t \approx t' : s$ is satisfied by $M_T$ only if it is a theorem of $\mathbb{T}$.

Proof. Suppose $\bar{\pi} \neq t \approx t' : s$ is an axiom of $\mathbb{T}$ where $\bar{\pi} = [\bar{\pi}_1 \neq x_1 : s_1, \ldots, \bar{\pi}_n \neq x_n : s_n]$. Given any valuation $\rho \in M_T[\bar{\pi}]$, for each $1 \leq i \leq n$ we have $\bar{\pi} \neq \rho(x_i) \in M_T[s_i]$. Choosing a representative term $t_i$ for each equivalence class $\rho(x_i)$, by Lemma 5.3.2 we have $\emptyset \vdash t_i \neq t : s_i$. Therefore the function $\sigma$ mapping each $x_i$ to $t_i$ is a substitution in $\Sigma(\bar{\pi}, \emptyset)$ that satisfies $\emptyset \vdash t \approx \sigma : \bar{\pi}$. Applying (subst) to this and $\emptyset \vdash t \approx t' : s$ gives $\emptyset \vdash t \approx t \{\sigma\} \approx t' \{\sigma\} : s$ and hence $\bar{\pi} \neq [t \{\sigma\}] = [t' \{\sigma\}] \in M_T[s]$ by Lemma 5.3.2 again. Lemma 5.3.4 and the definition of $\sigma$ gives $\bar{\pi} \neq M_T[t] \rho = M_T[t'] \rho$ as required.

So $M_T$ is a $\mathbb{T}$-algebra and it just remains to check that it satisfies a ground judgement $\emptyset \vdash \bar{\pi} \neq t \approx t' : s$ only if that judgement is a theorem of $\mathbb{T}$. If it satisfies the judgement $\emptyset \vdash \bar{\pi} \neq t \approx t' : s$, then $\bar{\pi} \neq M_T[t] \rho = M_T[t'] \rho$ holds for the unique valuation $\rho$ in $M_T[\emptyset]$. By Lemma 5.3.4 this means $\bar{\pi} \neq [t \{\sigma\}] = [t' \{\sigma\}] \in M_T[s]$ for $\sigma$ the unique substitution in $\Sigma(\emptyset, \emptyset)$. Since this is necessarily the identity substitution for the empty sorting environment, from (3.28) we get $\bar{\pi} \neq [t] = [t'] \in M_T[s]$. Thus by Lemma 5.3.2 and the definition of the equivalence classes of $M_T[s]$, $\emptyset \vdash \bar{\pi} \neq t \approx t' : s$ holds.

5.4 Completeness

In this section we prove the main result of the chapter, namely that for any NEL-theory the logical consequence relation (Definition 5.1.4) and the semantic consequence relation (Definition 3.4.7) coincide. For conventional algebra, completeness of equational logic for the usual interpretation of terms in algebras in the category of sets is a simple result: Given an equational theory, the collection of terms is quotiented by provable equality to get an algebra for which satisfaction coincides with theorem-hood. The role of variables in this term-algebra construction is to act as indeterminates - constants that do not occur in the signature of the original theory. Indeed, instead of working with all terms, it comes to the same thing if one extends the signature with countably many new constants and forms the term-algebra from ground-terms, as in the previous section. This interchangeability of variables and fresh constants in conventional equational logic is not so straightforward for nominal equational logic. In the interpretation of
our language of terms in $\mathcal{FM}$-Set, variables stand for indeterminate elements of an FM-set that therefore have indeterminate finite support, whereas constants have fixed finite supports in the nominal set of operation symbols. To prove the completeness theorem, we have to show that provability of a judgement involving variables can be recovered from provability of ground instantiations of the judgement, where the variables are replaced by constants with suitably fresh supports. These constants will be defined using the generalised transpositions of Example 2.1.1.

**Definition 5.4.1.** (Atom-Parameterised Constants) Take a tuple of distinct atoms $\vec{a} \in A^{(n)}$ and a sort $s \in \text{Sort}_\Sigma$ of a NEL-signature $\Sigma$ with the property

$$\supp(s) \subseteq \supp(\vec{a}).$$ (5.20)

Then we define a signature $\Sigma[c_{\vec{a}} : s]$ by adding new operation symbols $\{c_{\vec{a}'} \mid \vec{a}' \in A^{(n)}\}$, which we will call atom-parameterised constants, to $\text{Op}_\Sigma$. $\Sigma[c_{\vec{a}} : s]$ has the same nominal set of sorts as $\Sigma$ and the nominal set of operators given by the disjoint union $\text{Op}_\Sigma \cup \{c_{\vec{a}'} \mid \vec{a}' \in A^{(n)}\}$, where we assume each operation symbol $c_{\vec{a}'}$ is not already an element of $\text{Op}_\Sigma$. Therefore the Perm-action on the new operation symbols is $\pi \cdot c_{\vec{a}'} = c_{\pi \cdot \vec{a}'}$. The type for the new constants is given by

$$c_{\vec{a}'} : (\vec{a} \vec{a}') \cdot s$$ (5.21)

where $(\vec{a} \vec{a}')$ is the generalised transposition of Example 2.1.1. $\pi \cdot c_{\vec{a}'} = c_{\pi \cdot \vec{a}'}$ has sort $(\vec{a} \pi \cdot \vec{a}') \cdot s$. This is equal to $\pi(\vec{a} \vec{a}') \cdot s$ by Corollary 2.3.7 and (5.20), so the typing assignment is equivariant.

Note that for any permutation $\pi$ we have

$$\Sigma[c_{\vec{a}} : s] = \Sigma[c_{\pi \cdot \vec{a}} : \pi \cdot s].$$ (5.22)

Both signatures add all $\vec{a}' \in A^{(n)}$ to the original nominal set of operation symbols $\text{Op}_\Sigma$, so we need only check the typing assignment. $c_{\vec{a}}$ has type $(\vec{a} \vec{a}') \cdot s$ in $\Sigma[c_{\vec{a}} : s]$, and type $(\pi \cdot \vec{a} \vec{a}') \pi \cdot s$ in $\Sigma[c_{\pi \cdot \vec{a}} : \pi \cdot s]$. But by Corollary 2.3.7 and (5.20) we only need to consider the action of the permutation on the atoms of $\vec{a}$, so $(\vec{a} \vec{a}') \cdot s = (\pi \cdot \vec{a} \vec{a}') \pi \cdot s$.

If $T$ is a NEL-theory with underlying signature $\Sigma$, then $T[c_{\vec{a}} : s]$ denotes the theory with signature $\Sigma[c_{\vec{a}} : s]$ and the same axioms as $T$.

We will use atom-parameterised constants $c_{(a_1, \ldots, a_n)}$ as indeterminates in the proof of the completeness theorem given below. Of course $c_{(a_1, \ldots, a_n)}$ is not as indeterminate as is a variable $x$: The former represents an element of an FM-set for which a support set is known, namely $\{a_1, \ldots, a_n\}$; whereas the latter represents an element whose support only has to avoid at most finitely many atoms $\bar{a}$, supposing an assumption $\bar{a} \not\models x : s$ occurs in the freshness environment. Nevertheless, as the following proposition shows, one can recover a $T$-theorem involving a variable from an instance of it obtained by substituting a new atom-parameterised constant for the
(π x)\{c_\vec{a} := x_1\} \triangleq π x

(op t_1 \cdots t_n)\{c_\vec{a} := x_1\} \triangleq \begin{cases} (\vec{a} \vec{a}') x_1 & \text{if } \op = c_{\vec{a}'} \\ op \left((t_1\{c_\vec{a} := x_1\}) \cdots (t_n\{c_\vec{a} := x_1\})\right) & \text{otherwise.} \end{cases}

Figure 5.2: Replacing Atom-Parameterised Constants by Variables

variable. The proposition makes use of single term substitution: The term \( t\{t'/x'\} \) is defined by recursion on the structure of \( t \) by:

\[
(\text{op } t_1 \cdots t_n)\{t'/x'\} \triangleq \text{op } t_1\{t'/x'\} \cdots t_n\{t'/x'\}
\]

\[
(π x)\{t'/x'\} \triangleq \begin{cases} π x & \text{if } x \neq x' \\ π \ast t' & \text{if } x = x' \end{cases}
\]

where \( π \ast t' \) is as the object-level permutation action (3.23). This is a special case of the kind of simultaneous substitution \( t \mapsto t\{σ\} \) considered in Section 3.3, in the sense that if \( t \in Σ_\vec{a}(Γ, x' : s') \) (with \( x' \notin \text{dom}(Γ) \)) and \( t' \in Σ_\vec{a}(Γ) \), then \( t\{t'/x'\} \in Σ_\vec{a}(Γ) \) where \( \{t'/x'\} \in Σ((Γ, x' : s'), Γ) \) is the substitution mapping \( x' \) to \( t' \) and mapping each \( x \in \text{dom}(Γ) \) to itself.

**Proposition 5.4.2.** Suppose \( \triangledown, \overline{α}_1 \not\# x_1 : s_1 \vdash \overline{α} \not\# t \approx t' : s \) is a well-formed judgement (with \( x_1 \notin \text{dom}(\triangledown) \)) over the signature \( Σ \) of a NEL-theory \( T \). Given any finite set of atoms \( \overline{α}' \) supporting \((\overline{α}, t, t', s_1)\), let \( \vec{a} \in A\{n\} \) be an ordered list of the distinct atoms in \( \overline{α}' − \overline{α}_1 \). Then because \( \overline{α}_1 \not\# s_1 \) we know that \( \text{supp}(s_1) \subseteq \text{supp}(\vec{a}) \), so we can define \( T[c_\vec{a} : s_1] \) as in Definition 5.4.1. Then

\[
\triangledown \vdash_{T[c_\vec{a} : s_1]} \overline{α} \not\# t\{c_\vec{a}/x_1\} \approx t'\{c_\vec{a}/x_1\} : s \Rightarrow \triangledown, \overline{α}_1 \not\# x_1 : s_1 \vdash_{T} \overline{α} \not\# t \approx t' : s \ .
\]

To prove this proposition, we need an operation on terms that replaces atom-parameterised constants by variables: Given \( t \in Σ[c_\vec{a} \mid \vec{a} \in A\{n\}] \), Figure 5.2 defines a term \( t\{c_\vec{a} := x_1\} \in Σ_\vec{a}(Γ, x_1 : s_1) \), obtained by replacing each \( c_\vec{a} \) by the suspension \((\vec{a} \vec{a}') x_1 \).

The following lemmas give the properties of the operation in Figure 5.2 that we need.

**Lemma 5.4.3.** Suppose we have \( t \in Σ[c_\vec{a} : s_1]\{n\}(Γ) \).

(i) If \( t \) does not contain any of the operation symbols in \( \{c_\vec{a} \mid \vec{a} \in A\{n\}\} \), then \( t\{c_\vec{a} := x_1\} = t \).

(ii) If \( x_1 \) and \( x'_1 \) do not occur in \( t \) and if \( σ \) is the substitution that swaps \( x_1 \) and \( x'_1 \), then \( t\{c_\vec{a} := x_1\}\{σ\} = t\{c_\vec{a} := x'_1\} \).

(iii) \( π \cdot (t\{c_\vec{a} := x_1\}) = (π \cdot t)\{c_{π\vec{a}} := x_1\} \).
\textit{Proof.} (i) and (ii) follow easily by induction on the structure of $t$. For (iii) note that the equation is well-defined because $\pi \cdot t \in \Sigma[c_{\pi}, t] : s_1(\pi \cdot \Gamma)$ by (3.17) and (5.22), so both sides of the equation are terms in $\Sigma_{\pi, s}(\pi \cdot \Gamma)$. The result then follows by induction on the structure of $t$, using (2.17) in the atom-parameterised constant case.

\textbf{Lemma 5.4.4.} Let $\Sigma[c_{\pi} : s_1]$ and $\mathbb{T}[c_{\pi} : s_1]$ be as defined as in Definition 5.4.1.

(i) Take $\nabla \in \mathbb{F}E_{\Sigma}, x_1 \notin \text{dom}(\nabla), \pi \in \text{Perm}, t \in \Sigma[c_{\pi} : s_1]|s(\nabla')$ and $\overline{\pi}_1$ supporting $\pi$. If $\overline{\pi}_1 \not\equiv \overline{a}$ (so $\overline{\pi}_1 \neq s_1$) then

$$\nabla, \overline{\pi}_1 \not\equiv x_1 : s_1 \vdash_{\mathbb{T}} (\pi \ast t)\{c_{\pi} := x_1\} \approx \pi \ast (t\{c_{\pi} := x_1\}) : \pi \cdot s.$$  

(ii) Take $\nabla_1, \nabla_2 \in \mathbb{F}E_{\Sigma}, \sigma \in \Sigma[c_{\pi} : s_1]|s((\nabla_1)^{\overline{a}}, (\nabla_2)^{\overline{a}}), x_1 \notin \text{dom}(\nabla_1) \cup \text{dom}(\nabla_2), t \in \Sigma[c_{\pi} : s_1]|s((\nabla_1)^{\overline{a}})$ and $\overline{\pi}_1$ supporting $t$. If $\overline{\pi}_1 \not\equiv \overline{a}$ (so $\overline{\pi}_1 \neq s_1$) then

$$\nabla_2, \overline{\pi}_1 \not\equiv x_1 : s_1 \vdash_{\mathbb{T}} t\{\sigma\}\{c_{\pi} := x_1\} \approx t\{c_{\pi} := x_1\}\{\sigma\{c_{\pi} := x_1\}\} : s$$

where $\sigma\{c_{\pi} := x_1\} \in \Sigma(((\nabla_1)^{\overline{a}}, x_1 : s_1), ((\nabla_2)^{\overline{a}}, x_1 : s_1))$ is the substitution mapping each $x \in \text{dom}(\nabla_1)$ to $\sigma(x)\{c_{\pi} := x_1\}$ and mapping $x_1$ to itself.

\textit{Proof.} Both parts are proved by induction on the structure of $t$. For (i) the only non-trivial case is where $t = c_{\pi}^{\overline{a}}$, and we must show that

$$\nabla, \overline{\pi}_1 \not\equiv x_1 : s_1 \vdash_{\mathbb{T}} (\pi \ast (\overline{a} \cdot \overline{a}')) x_1 \approx \pi(\overline{a} \overline{a}') x_1 : \pi \cdot s.$$  

This holds by (susp) and (weak) using

$$\overline{\pi}_1 \supseteq \{a \mid \pi^{-1}(a) \neq a\}$$

(since $\text{supp}(\pi^{-1}) = \text{supp}(\pi) \subseteq \overline{\pi}_1$)

$$= \{a \mid (\pi \cdot \overline{a} \pi) \pi^{-1}(a) \neq \pi(\overline{a} \overline{a}')(a)\}$$

(by (2.17))

$$= \{a \mid (\pi \cdot \overline{a} \cdot \overline{a}')(a) \neq \pi(\overline{a} \overline{a}')(a)\}$$

(since $\text{dom}(\pi) \subseteq \overline{\pi}_1 \not\equiv \overline{a}$).

For (ii) the only non-trivial case is where $t = \pi x$, which follows immediately from (i).

The next lemma shows that we can safely replace atom-parameterised constants with suspensions over a new variable, so long as we assume the freshness of certain atoms for that variable.

\textbf{Lemma 5.4.5.} Let $\mathbb{T}[c_{\pi} : s_1]$ be a theory as in Definition 5.4.1. Suppose $\nabla \vdash_{\mathbb{T}[c_{\pi} : s_1]} \overline{\pi} \not\equiv t \approx t' : s$, that $x_1 \notin \text{dom}(\nabla)$, that $\overline{\pi}_1$ is a set of atoms supporting $(\overline{\pi}, t, t')$ and that $\text{supp}(\overline{a}) \not\equiv (\overline{\pi}_1, \nabla)$ (and therefore $\overline{\pi}_1 \neq s_1$). Then

$$\nabla, \overline{\pi}_1 \not\equiv x_1 : s_1 \vdash_{\mathbb{T}} \overline{\pi} \not\equiv t\{c_{\pi} := x_1\} \approx t'\{c_{\pi} := x_1\} : s$$  

(5.26)
Proof. This is proved by induction on the derivation of $\nabla \vdash T[c_a : s_1] \not\vdash t \approx t' : s$. Let $IH$ be the set of well-formed judgements
\[
\nabla \vdash \not\vdash t \approx t' : s \quad (5.27)
\]
over $\Sigma[c_a : s_1]$ such that for all $\bar{a}_1$ supporting $(\bar{a}, t, t')$, if $\bar{a} \not\vDash (\bar{a}_1, \nabla)$ and $\nabla \vdash T[c_a : s_1] \not\vdash t \approx t' : s$ hold then so does (5.26).

If (5.27) is an axiom of $T[c_a : s_1]$ then it is by definition an axiom of $T$ and hence does not contain any atom-parameterised constants. So by Lemma 5.4.3(i), (5.26) is $\nabla \not\vdash x_1 : s \vdash \not\vdash t \approx t' : s$, which is a theorem of $T$ by applying (weak) to the axiom (5.27). So $IH$ contains the axioms of $T[c_a : s_1]$ and to prove the lemma we just have to show that it is closed under each of the rules in Figure 5.1. Closure under rules (Refl), (Symm), (Weak), (Atm-intro), (≠-equivar) and (Susp) is straightforward. However with (Trans), (Subst) and (Atm-elim) the support of the hypotheses of each rule may be bigger than that of its conclusion, so the induction hypothesis cannot be applied without some work. We give the argument for (Trans). The proof for the other two rules is similar, using Lemma 5.4.4(ii) in the case of (Subst).

To prove closure under (Trans), suppose
\[
(\nabla \not\vdash \not\vdash t \approx t' : s), (\nabla \not\vdash \not\vdash t \approx t'' : s) \in IH . \quad (5.28)
\]
We wish to show that $(\nabla \not\vdash \not\vdash t' \approx t'' : s) \in IH$ for any $\bar{a}_1$ supporting $(\bar{a}, \bar{a}', t, t'')$. That is, if $\text{supp}(\bar{a}) \not\vDash (\bar{a}_1, \nabla)$ and $\nabla \not\vdash T[c_a : s_1] \not\vdash \not\vdash t \approx t' : s$ then $\nabla, \bar{a}_1 \not\vdash x_1 : s_1 \vdash \not\vdash \bar{a} \cup \bar{a}' \not\vdash t\{c_a := x_1\} \approx t'\{c_a := x_1\} : s$.

Let $\bar{a}_2$ be a set of atoms supporting $(\bar{a}_1, t')$ and let $\bar{a}'$ be a tuple of atoms of the same size as $\bar{a}$ such that $\text{supp}(\bar{a}') \not\vDash (\bar{a}, \bar{a}_2, \nabla)$. Now $\Sigma[c_a : s_1] = \Sigma[c_a' : (\bar{a}' \bar{a}) \cdot s_1]$ by (5.22), so by (5.28) we have
\[
\nabla, \bar{a}_2 \not\vdash x_1 : (\bar{a}' \bar{a}) \cdot s_1 \vdash \not\vdash t\{c_a := x_1\} \approx t'\{c_a := x_1\} : s \quad (5.29)
\]
and similarly for $\bar{a}' \not\vDash t' \approx t''$. We apply (Trans) and (Weak) to get
\[
(\nabla, \bar{a}_1 \not\vdash x_1 : (\bar{a}' \bar{a}) \cdot s_1) \not\vdash (\bar{a}_2 - \bar{a}_1) \vdash \not\vdash t\{c_a := x_1\} \approx t''\{c_a := x_1\} : s \quad . \quad (5.30)
\]
$\bar{a}_1$ supports $(\bar{a}, \bar{a}', t', t'')$, so $(\bar{a}_2 - \bar{a}_1) \not\vDash (\bar{a}, \bar{a}', t', t'')$. $(\bar{a}_2 - \bar{a}_1) \not\vDash \bar{a}'$ by construction. So by the equivariance of the ‘:=’ operation (Lemma 5.4.3(iii)) we have $(\bar{a}_2 - \bar{a}_1) \not\vDash (\bar{a} \cup \bar{a}', t\{c_a := x_1\}), t'\{c_a := x_1\})$ and can apply (Atm-elim) to get
\[
\nabla, \bar{a}_1 \not\vdash x_1 : (\bar{a}' \bar{a}) \cdot s_1 \vdash \not\vdash t\{c_a := x_1\} \approx t''\{c_a := x_1\} : s \quad . \quad (5.31)
\]
Now Theorem 5.2.2 told us that logical consequence is equivariant, so we can apply $(\bar{a}' \bar{a})$ to (5.31) to get
\[
\nabla, \bar{a}_1 \not\vdash x_1 : s_1 \vdash \not\vdash t\{c_a := x_1\} \approx t''\{c_a := x_1\} : s \quad . \quad (5.32)
\]
as required, using Lemma 5.4.3(iii) and $\text{supp}(\bar{a}' \bar{a}) \not\vDash (\nabla, \bar{a}_1, \bar{a}, t, t'', s)$.

We may now proceed to the proof of the main technical result of this section.
Proof of Proposition 5.4.2. First note that since substitution is equivariant (Corollary 3.3.7) and \( \text{supp}(\bar{a}) \subseteq \bar{\pi} \) we have that \( \bar{\pi} \) supports \( t\{c_{\bar{a}}/x_1\} \) and \( t'\{c_{\bar{a}}/x_1\} \). Therefore, picking any \( x'_1 \notin \text{dom}(\nabla) \cup \{x_1\} \) and \( \bar{a}' \in A^{(n)} \) with \( \text{supp}(\bar{a}') \# (\bar{\pi}, \nabla) \), and noting that \( \Sigma[c_{\bar{a}} : s_1] = \Sigma[c_{\bar{a}'} : (\bar{a} \bar{a}') \cdot s_1] \) by (5.22), we can apply Lemma 5.4.5 to get

\[
\nabla, \bar{\pi} \# x'_1 : (\bar{a} \bar{a}') \cdot s_1 \vdash_T \bar{\pi} \# t\{c_{\bar{a}}/x_1\} \{c_{\bar{a}'} := x'_1\} \approx t'\{c_{\bar{a}}/x_1\} \{c_{\bar{a}'} := x'_1\} : s .
\]

(5.33)

From this, using Lemma 5.4.4(ii), Lemma 5.4.3(i) and the definition in Figure 5.2, we get

\[
\nabla, \bar{\pi}' \# x'_1 : (\bar{a} \bar{a}') \cdot s_1 \vdash_T \bar{\pi}' \# t\{c_{\bar{a}}/x_1\} \approx t'\{(\bar{a} \bar{a}') x'_1/x_1\} : s .
\]

(5.34)

Now (\( t\{(\bar{a} \bar{a}') x'_1/x_1\}\)){\((\bar{a} \bar{a}') x'_1/x_1\)} = \( t \), so by (\text{SUBST}), (5.34) and (5.35) we have

\[
\nabla, \bar{\pi}_1 \cup \text{supp}(\bar{a}') \# x_1 : s_1 \vdash_T \bar{\pi} \# (\bar{a} \bar{a}') x_1 : (\bar{a} \bar{a}') \cdot s_1 .
\]

(5.35)

Now \( (t\{(\bar{a} \bar{a}') x'_1/x_1\})\{(\bar{a} \bar{a}') x'_1/x_1\}) = t \), so by (\text{ATM-ELIM}) we have \( \nabla, \bar{\pi}_1 \# x_1 : s_1 \vdash_T \bar{\pi} \# t \approx t' : s \) as required.

\( \square \)

Proposition 5.4.6. With the same assumptions as in the statement of Proposition 5.4.2, it is the case that

\[
\nabla, \bar{\pi}_1 \# x_1 : s_1 \vdash_T \bar{\pi} \# t \approx t' : s \Rightarrow \nabla \vdash_{T[\Sigma[s_{\bar{a}}]]} \bar{\pi} \# t\{c_{\bar{a}}/x_1\} \approx t'\{c_{\bar{a}}/x_1\} : s .
\]

(5.38)

Proof. Given \( \rho \in \llbracket \nabla \rrbracket \) we must prove that the left hand side of (5.38) implies

\[
\bar{\pi} \# \llbracket t\{c_{\bar{a}}/x_1\} \rrbracket \rho = \llbracket t'\{c_{\bar{a}}/x_1\} \rrbracket \rho
\]

(5.39)

which by Lemma 3.3.5 is equivalent to

\[
\bar{\pi} \# \llbracket t\{c_{\bar{a}}/x_1\} \rrbracket \rho = \llbracket t'\{c_{\bar{a}}/x_1\} \rrbracket \rho
\]

(5.40)

where \( \{c_{\bar{a}}/x_1\} \rho \in \llbracket \nabla, x_1 : s_1 \rrbracket \) is the valuation mapping each \( x \in \text{dom}(\nabla) \) to \( \rho(x) \) that also maps \( x_1 \) to \( \llbracket c_{\bar{a}} \rrbracket \).

Now a structure for \( \Sigma[c_{\bar{a}} : s_1] \) is a structure for \( \Sigma \) if we ignore the assignments \( c_{\bar{a}} \mapsto \llbracket c_{\bar{a}} \rrbracket \). But \( T \) and \( T[c_{\bar{a}} : s_1] \) have the same axioms, so an algebra for \( T[c_{\bar{a}} : s_1] \) is an algebra for \( T \) similarly. \( \bar{\pi}_1 \# \bar{a} \) by construction, so by the equivariance of the map defined by Definition 3.1.4 we have \( \bar{\pi}_1 \# \llbracket c_{\bar{a}} \rrbracket \). Therefore \( \{c_{\bar{a}}/x_1\} \rho \in \llbracket \nabla, \bar{\pi}_1 \# x_1 : s_1 \rrbracket \), so if the left hand side of (5.38) holds then (5.40) follows. \( \square \)
Using Propositions 5.4.2 and 5.4.6, we can now prove the desired completeness result.

**Theorem 5.4.7.** (Completeness) A judgement $\nabla \vdash \bar{\alpha} \not\equiv t \approx t' : s$ is a theorem of a NEL-theory $T$ if it is satisfied by all $T$-algebras:

$$\nabla \vdash_T \bar{\alpha} \not\equiv t \approx t' : s \Rightarrow \nabla \vdash_T \bar{\alpha} \not\equiv t \approx t' : s .$$

**Proof.** The proof proceeds by induction on the length of the freshness environment $\nabla$, for all $T$, $\bar{\alpha}$, $t$ and $t'$ simultaneously. The base case when the length is zero is a consequence of the Ground Completeness Theorem 5.3.5. For the induction step, if $\nabla, \bar{\alpha}_1 \not\equiv x_1 : s_1 \vdash_T \bar{\alpha} \not\equiv t \approx t' : s$ (with $x_1 \notin \text{dom}(\nabla)$), then let $\bar{\alpha}'$ be some finite set of atoms supporting $(\bar{\alpha}, t, t', s_1)$ and $\bar{a}$ be an ordering of $\bar{\alpha}' - \bar{\alpha}_1$. Then by Proposition 5.4.6 we have

$$\nabla \vdash_{T[\bar{c};s_1]} \bar{\alpha} \not\equiv t\{c_{\bar{a}}/x_1\} \approx t'\{c_{\bar{a}}/x_1\} : s$$

and hence by the induction hypothesis

$$\nabla \vdash_{T[\bar{c};s_1]} \bar{\alpha} \not\equiv t\{c_{\bar{a}}/x_1\} \approx t'\{c_{\bar{a}}/x_1\} : s .$$

Now we can apply Proposition 5.4.2 to deduce $\nabla, \bar{\alpha}_1 \not\equiv x_1 : s_1 \vdash_T \bar{\alpha} \not\equiv t \approx t' : s$, as required. \qed

### 5.5 Proof rules for equality

The three part judgements of Definition 3.4.5, expressing both freshness and equality assertions, are generally the most convenient way to express theories of Nominal Equational Logic. For example, the discussion of the untyped $\lambda$-calculus in Example 4.3.4 showed that binding patterns are often more naturally expressed in terms of freshness than equality. However, it turns out that freshness assertions are not necessary on theoretical grounds, as freshness can always be defined equationally modulo certain freshness assumptions. This should not be surprising given Lemma 2.3.5, which gave an equational definition for freshness in the category $\text{Nom}$. A similar result was proved independently for the system of Nominal Algebra in [23], and some of the consequences of this result were explored in [17].

Using this result, we will provide proof rules for NEL with equality only. Although this system is arguably less intuitive and elegant than the proof rules for NEL presented in Figure 5.1, we will find them useful in Chapter 7 where we will be working with categorial logic, where there is a clear definition of equality (that is, the equality of two arrows in a category) but freshness is a more complicated concept that must be defined.

**Lemma 5.5.1.** [c.f. Lemma 2.3.5] Suppose that $T$ is a NEL-theory for a signature $\Sigma$, and that we have a sort $s \in \text{Sort}_\Sigma$, a freshness environment $\nabla \in \text{FE}_\Sigma$, a term $t \in \Sigma_s(\nabla^\uparrow)$ and a finite set of atoms $\bar{\alpha} \in \mathcal{P}_{\text{fin}}(\mathbb{A})$ such that $\bar{\alpha} \not\equiv s$. Then

$$\nabla \vdash_T \bar{\alpha} \not\equiv t : s \iff \nabla \not\equiv \text{supp}(\bar{a}) \vdash_T t \approx (\bar{a} \bar{a}^* \bar{a}) * s$$
where \( \vec{a} \in \mathbb{A}^{(n)} \) is an ordering of \( \vec{a} \) and \( \vec{a}' \in \mathbb{A}^{(n)} \) is a tuple of the same size such that \( \text{supp}(\vec{a}') \neq (\nabla, \vec{a}, t) \).

**Proof.** Left-to-right: Applying the rule (ATM-INTRO) to the left hand side gives us \( \nabla \# \text{supp}(\vec{a}') \vdash \vec{a} \cup \text{supp}(\vec{a}') \neq t : s \); (SUSP) gives us \( \vec{a} \cup \text{supp}(\vec{a}') \neq x : s \vdash x \approx (\vec{a} \vec{a}') x : s \). We then apply (SUBST) for our result.

Right-to-left: Applying (REFL) and (ATM-INTRO) gives us \( \nabla \# \text{supp}(\vec{a}') \vdash \text{supp}(\vec{a}') \neq t : s \). By (\#-EQUIVAR) we have \( \text{supp}(\vec{a}') \neq x : s \vdash \vec{a} \neq (\vec{a} \vec{a}') x : s \). Applying (SUBST), \( \nabla \# \text{supp}(\vec{a}') \vdash \vec{a} \neq (\vec{a} \vec{a}') \neq t : s \). Therefore by our hypothesis, \( \nabla \# \text{supp}(\vec{a}') \vdash \vec{a} \neq t : s \), and we apply (ATM-ELIM) to \( \text{supp}(\vec{a}') \) to get our result. \( \square \)

Note that by the results the previous section the above Lemma applies also to the equivalent semantic consequence relation, \( \models \).

**Definition 5.5.2.** Let a \( \text{NEL}^{E} \)-theory \( \mathbb{T} \) consist of a NEL-signature \( \Sigma \) together with a collection of axioms of the form

\[
\nabla \vdash t \approx t' : s
\]

where \( \nabla \in \text{FE}_{\Sigma} \) and \( t, t' \in \Sigma_{s}(\nabla') \). Note that any \( \text{NEL}^{E} \)-theory is also a NEL-theory.

The set of theorems of a \( \text{NEL}^{E} \)-theory \( \mathbb{T} \) is the least set of judgements of the form (5.41) containing the axioms of \( \mathbb{T} \) and closed under the rules in Figure 5.3. The rules other than (PERM) have clear analogues in the rules of Figure 5.1, so we have used the superscript \( E \) to distinguish them. Additionally, the rule (SUBST\(^{E}\)) uses the following new pieces of notation:

- In the rule (SUBST\(^{E}\)), \( \nabla \vdash \sigma \approx \sigma' \) means that for each \( x \in \text{dom}(\nabla) \) where \( \nabla'(x) = s \), we have \( \nabla' \vdash \sigma(x) \approx \sigma'(x) : s \);

- \( \nabla' \vdash \sigma : \nabla \) means that for each \( x \in \text{dom}(\nabla) \) where \( \nabla'(x) = (s, \vec{a}) \), if \( \vec{a} \in \mathbb{A}^{(n)} \) is an ordering of \( \vec{a} \) and \( \vec{a}' \in \mathbb{A}^{(n)} \) is a tuple of the same size as \( \vec{a} \) such that \( \text{supp}(\vec{a}') \neq (\nabla', \vec{a}, \sigma(x)) \) then

\[
(\nabla')\# \text{supp}(\vec{a}') \vdash \sigma(x) \approx (\vec{a} \vec{a}') * \sigma(x) : s .
\]

We write

\[
\nabla \vdash \vec{E}_{\mathbb{T}} t \approx t' : s
\]

to indicate that a judgement (5.41) is a theorem of \( \mathbb{T} \).

**Lemma 5.5.3.** [c.f. Lemma 5.2.1] For any \( \text{NEL}^{E} \)-theory \( \mathbb{T} \), \( \nabla \vdash \vec{E}_{\mathbb{T}} t \approx t' : s \) implies \( \nabla \vdash \vec{E}_{\mathbb{T}} \pi * t \approx \pi * t' : \pi \cdot s \).

**Proof.**

\[
\frac{(\text{SUBST}^{E}) \quad \nabla \vdash t \approx t' : s \quad x : s \vdash \pi x \approx \pi x : \pi \cdot s}{\nabla \vdash \pi x\{t/x\} \approx \pi x\{t'/x\} : \pi \cdot s}
\]

\( \square \)
Lemma 5.5.4. [c.f. Theorem 5.2.2] For any $\text{NEL}^E$-theory $\mathcal{T}$, $\nabla \vdash E \tau \approx t' : s$ implies $\pi \cdot \nabla \vdash E \tau \pi \cdot s$.

Proof. Suppose that $\nabla = [\bar{a}_1 \# x_1 : s_1, \ldots, \bar{a}_n \# x_n : s_n]$. Then for $1 \leq i \leq n$ let $\bar{a}_i$ be an ordering of $\pi_i$ and $\bar{a}_i'$ be a suitably fresh tuple of the same size. Then by (PERM) we have $\pi \cdot \pi_i \cup \text{supp}(\bar{a}_i') \not\approx x_i : \pi \cdot s_i \vdash \pi^{-1} x_i \approx (\bar{a}_i, \bar{a}_i') \pi^{-1} x_i : s_i$, and by (WEAK$^E$) we have $(\pi \cdot \nabla)^{\# \text{supp}(\bar{a}_i')} \vdash \pi^{-1} x_i \approx (\bar{a}_i, \bar{a}_i') \pi^{-1} x_i : s_i$. Therefore $\pi \cdot \nabla \vdash (\pi^{-1} -) : \nabla$, following the notation introduced in Definition 5.5.2.

The rest of the proof follows as with Theorem 5.2.2, using Lemma 5.5.3 and (SUBST):

\[
\pi \cdot \nabla \vdash (\pi^{-1} -) : \nabla \quad \nabla \vdash \pi \cdot s \quad \pi \cdot s \approx t \quad \pi \cdot t \approx \pi \cdot t' : \pi \cdot s
\]

$(\pi \cdot t)\{\pi^{-1} -\} = \pi \cdot t$ and $(\pi \cdot t')\{\pi^{-1} -\} = \pi \cdot t'$ by Lemma 3.3.6. \hfill \Box

Lemma 5.5.5. Suppose we have an $\text{NEL}^E$-theory $\mathcal{T}$, a freshness environment $\nabla$ and term $t \in \Sigma_s(\nabla')$. Suppose we have $\bar{a}, \bar{b} \in A^{(m)}$ such that $\text{supp}(\bar{a}) \not\approx s$ and $\text{supp}(\bar{b}) \not\approx (\nabla', t, t')$. Now suppose that $\bar{a}', \bar{b}' \in A^{(m)}$ for some $m \leq n$, with $\text{supp}(\bar{a}') \subseteq \text{supp}(\bar{a})$ and $\text{supp}(\bar{b}') \subseteq \text{supp}(\bar{b})$. Then

\[
\nabla^{\# \text{supp}(\bar{b})} \vdash E \tau \approx (\bar{a} \bar{b}) \ast t : s \Rightarrow \nabla^{\# \text{supp}(\bar{b}')} \vdash E \tau \approx (\bar{a}' \bar{b}') \ast t : s
\]

Proof. By (PERM) we have $\text{supp}(\bar{a}) \cup \text{supp}(\bar{b}') \not\approx x : s \vdash x \approx (\bar{a}' \bar{b}') x : s$. We wish to use (SUBST$^E$) to conclude that $\nabla^{\# \text{supp}(\bar{b}')} \vdash E x\{t/x\} \approx ((\bar{a}' \bar{b}') x)\{t/x\} : s$; for this substitution to
work we must prove that
\[ \nabla \vdash \{ t/x \} \cdot [\text{supp}(\vec{a}) \cup \text{supp}(\vec{b})] : x:s \]  \hspace{1cm} (5.44)

Applying Lemma 5.5.4 and (\textsc{weak}^E) to our hypothesis gives us
\[ \nabla \vdash \{ t/x \} \cdot [\text{supp}(\vec{b})] : t:s \]  \hspace{1cm} (5.45)

Now take fresh \( \vec{c} \in \mathbb{A}^{(n)}, \vec{c}' \in \mathbb{A}^{(m)} \). By (\textsc{perm}) we have
\[ \nabla \vdash \{ t/x \} \cdot [\text{supp}(\vec{b})] : t:s \]  \hspace{1cm} (5.46)

We can use (\textsc{trans}^E) on (5.46) and (5.45) to get
\[ \nabla \vdash \{ t/x \} \cdot [\text{supp}(\vec{b})] : t:s \]  \hspace{1cm} (5.47)

which is equivalent to (5.44), so we are done. \( \square \)

\textbf{Definition 5.5.6.} Given a NEL-theory \( \mathbb{T} \), let \( \mathbb{T}^E \) be the NEL\(^E\)-theory produced by replacing each axiom
\[ \nabla \vdash \vec{a} \not\approx t : s \]  \hspace{1cm} (5.48)

by the axioms
\[ \nabla \vdash t \approx t' : s \text{ and } \nabla \vdash \{ \vec{a} \} : t \approx (\vec{a} \vec{a}) : t : s \]  \hspace{1cm} (5.49)

where \( \vec{a} \in \mathbb{A}^{(n)} \) is an ordering of \( \vec{a} \) and \( \vec{a}' \in \mathbb{A}^{(n)} \) is a tuple of the same size such that \( \text{supp}(\vec{a}') \not\approx (\nabla, \vec{a}, t) \). By Lemma 5.5.1 the axioms of \( \mathbb{T} \) and \( \mathbb{T}^E \) are equivalent.

\textbf{Theorem 5.5.7.} Let \( \mathbb{T} \) be a NEL-theory. Then \( \nabla \vdash \vec{a} \not\approx t : s \) implies that \( \nabla \vdash \vec{a}' \not\approx t' : s \) and \( \nabla \vdash \{ \vec{a} \} \vdash \vec{a}' \not\approx t \vdash \vec{a}' \approx t' : s \), where \( \vec{a} \in \mathbb{A}^{(n)} \) is an ordering of \( \vec{a} \) and \( \vec{a}' \in \mathbb{A}^{(n)} \) is a tuple of the same size such that \( \text{supp}(\vec{a}') \not\approx (\nabla, \vec{a}, t) \).

\textit{Proof.} If \( \nabla \vdash \vec{a} \not\approx t : s \) is an axiom of \( \mathbb{T} \) then the corresponding axioms hold for \( \mathbb{T}^E \) by Definition 5.5.6. The proof then proceeds by induction on the derivation of \( \nabla \vdash \vec{a} \not\approx t : s \) by the rules of Figure 5.1.

The rule (\textsc{refl}) is identical to (\textsc{refl}^E). (\textsc{weak}) and (\textsc{atm-elim}) follow immediately from applications of (\textsc{weak}^E) and (\textsc{atm-elim}^E), while (\textsc{susp}) is a special case of (\textsc{perm}).

(\textsc{symm}): Applying the induction hypothesis to the premise of the rule gives us \( \nabla \vdash t \approx t' : s \) and \( \nabla \vdash \{ \vec{a} \} \vdash t \approx (\vec{a} \vec{a}) : s \). \( \nabla \vdash t' \approx t : s \) follows by (\textsc{symm}^E); \( \nabla \vdash (\vec{a} \vec{a}) : t \approx (\vec{a} \vec{a}) : t' : s \) by Lemma 5.5.3, so we use (\textsc{weak}^E) and (\textsc{trans}^E) to conclude that \( \nabla \vdash \{ \vec{a} \} \vdash t' \approx (\vec{a} \vec{a}) : t' : s \).

(\textsc{trans}): The induction hypothesis gives us \( \nabla \vdash t \approx t' : s \), \( \nabla \vdash \{ \vec{a} \} \vdash t \approx (\vec{a} \vec{a}) : t : s \), \( \nabla \vdash t' \approx t' : s \) and \( \nabla \vdash \{ \vec{a} \} \vdash t' \approx (\vec{a} \vec{a}) : t' : s \), where \( \vec{a}_1, \vec{a}_2 \) are orderings of \( \vec{a}_1 \) and \( \vec{a}_2 \), while \( \vec{a}_1, \vec{a}_2 \) are fresh tuples of the same size. \( \nabla \vdash t \approx t' : s \) by (\textsc{trans}^E). Now suppose
\(\overline{a}_1 \not\equiv \overline{a}_2\) (if not the case we can use Lemma 5.5.5 to make it so), and use (weakE), successive applications of (transE) and Lemma 5.5.3:

\[
\nabla \#\supp(\overline{a}_1)\cup\supp(\overline{a}_2) \vdash^E \overline{t} \approx (\overline{a}_1 \overline{a}_1) * \overline{t} \\
\approx (\overline{a}_1 \overline{a}_1) * \overline{t}' \\
\approx (\overline{a}_1 \overline{a}_1)(\overline{a}_2 \overline{a}_2) * \overline{t}' \\
\approx (\overline{a}_1 \overline{a}_1)(\overline{a}_2 \overline{a}_2) * \overline{t} .
\]

(atm-intro): Given \(\nabla \#\supp(\overline{a}') \vdash \overline{t} \approx (\overline{a} \overline{a}') * \overline{t} \), we need to prove that \(\nabla \#\supp(\overline{a}')\cup\{a,a'\} \vdash \overline{t} \approx (a a')(\overline{a} \overline{a}') * \overline{t} \) for suitably fresh \(a,a'\). This follows by (substE):

\[
\nabla \#\supp(\overline{a}')\cup\{a,a'\} \vdash \overline{t} \approx (\overline{a} \overline{a}') * \overline{t} : \overline{s} \\
\nabla \#\supp(\overline{a}')\cup\{a,a'\} \vdash \{t/x\} : \{\{a,a'\} \not\equiv x : s\} \vdash \overline{t} \approx (\overline{a} \overline{a}') x : s \\
\nabla \#\supp(\overline{a}') \vdash x\{t/x\} \approx ((a a') x)\{((a a') * t)\{\sigma\} \} \text{ by (3.31), so we wish to use (substE)}:
\]

\[
(\nabla') \#\supp(\overline{a}') \vdash \sigma : \nabla \#\supp(\overline{a}') \\
(\nabla') \#\supp(\overline{a}') \vdash t\{\sigma\} \approx ((\overline{a} \overline{a}') * t)\{\sigma\} : s
\]

The second premise is among our hypotheses, while the first follows from the hypothesis (\(\nabla') \#\supp(\overline{a}') \vdash \sigma(x_i) \approx (\overline{a}_i \overline{a}_i') * \sigma(x_i) : s\), along with introducing the fresh set of atoms \(\overline{a}\), as we showed we could in our discussion of (atm-intro) above.

(#-equivar): \(\overline{a} \not\equiv x : s \vdash \pi x \approx (\pi \cdot \overline{a} \overline{a}') x : s \) by (perm).

\[\square\]

**Theorem 5.5.8.** If \(\mathcal{T}\) is a NELE-theory then \(\nabla \vdash^E \overline{t} \approx \overline{t}' : s\) implies \(\nabla \vdash^E \overline{t} \approx t' : s\), so by Lemma 5.5.1 and Theorem 5.5.7 NEL and NELE are equivalent.

**Proof.** We need only check that each of the rules of Figure 5.3 can be derived by the rules of Figure 5.1. (reflE), (symmE), (transE), (weakE) and (atm-elime) are special cases of the corresponding rules of Figure 5.1. (substE) is a special case of (subst), as (5.42) is equivalent to the usual condition \(\nabla' \vdash x \equiv (\pi \cdot \overline{a} \overline{a}') x : s\) by Lemma 5.5.1.

(perm) follows by (subst):

\[
\nabla \#\ds(x,\pi) \vdash ds(x,\pi) \equiv x : s \\
\nabla \#\ds(x,\pi) \vdash ds(x,\pi) \equiv x : s \vdash x \equiv \pi x : s
\]

The first premise follows by (atm-intro), while the second is (susp), \(\pi x\{t/x\} = \pi * t\) and \(\pi x\{t'/x\} = \pi * t'\) by (5.23). \[\square\]
Chapter 6

Universal algebra

This chapter will provide an account of universal algebra over Nominal Equational Logic. The results of this chapter have been proved independently in a very general setting by [17]; that approach will be discussed in Section 8.3. Nonetheless, it is interesting that tackling these results in a more traditional manner produces a picture that very closely mirrors the standard account over equational logic (e.g. [37, Chapter VI]), in contrast to the extra effort required for the completeness proof of the last section and Lawvere theories of the next.

6.1 The category of $T$-algebras

**Definition 6.1.1.** Given a NEL-theory $T$ over a signature $\Sigma$, recall the definition of $T$-algebra from Definition 3.4.7. Given two such $T$-algebras $M, M'$, a $T$-homomorphism $f : M \rightarrow M'$ is an equivariant map $s \mapsto f_s$ from the sorts of $\text{Sort}_{\Sigma}$ to FM-functions, such that

$$f_s : M[s] \rightarrow M'[s] \quad (6.1)$$

and $f$ commutes with the operation symbols of $\text{Op}_{\Sigma}$; that is, for each $op : \vec{s} \rightarrow s$, where $\vec{s} = [s_1, \ldots, s_n]$,

$$M[\vec{s}] \xrightarrow{f} M'[\vec{s}] \quad \text{(6.2)}$$

commutes, where, following Definition 3.1.4, $M[\vec{s}] = M[s_1] \times \cdots \times M[s_n]$ (and likewise for $M'[\vec{s}]$), and $f_s = f_{s_1} \times \cdots \times f_{s_n}$.

**Definition 6.1.2.** The **category of $T$-algebras**, $T\text{-Alg}$, has as objects $T$-algebras and as arrows $T$-homomorphisms.

**Lemma 6.1.3.** The ground term algebra (Definition 5.3.3) $M_T$ is initial in the category $T\text{-Alg}$. 
Given a NEL-signature

Definition 6.2.1.

6.2 Free

T -algebras

CHAPTER 6. UNIVERSAL ALGEBRA

70

is unique. For each \( s \in \text{Sort} \) and ground term \( t \in \Sigma_s(\emptyset) \) we have \([t] \in M\llbracket s\rrbracket\), so we will define \( !_s : M\llbracket s\rrbracket \to M\llbracket s\rrbracket \) by

\[
!_s([t]) \triangleq M[[t]] \rho
\]

(6.3)

where \( \rho \) is the empty valuation. \( s \mapsto !_s \) is equivariant because for any \( t \in \Sigma_{\pi_s}(\emptyset) \), \((\pi \cdot !_s)([t]) = !_s(\pi([t]))\) by (2.18) along with the equivariance results (5.14) and (3.22) and the observation that the empty valuation has empty support.

(6.3) is well defined because if \([t] = [t']\) then by definition \( \emptyset \vdash_T t \approx t' : s \), so by Soundness \( \emptyset \vdash_T t \approx t' : s \). But \( M \) is a \( T \)-algebra, so \( M[[t]] \rho = M[[t']] \rho \).

\( ! \) satisfies (6.2) because \( M[[op]] \circ (\pi \cdot !_s([t_1]), \ldots) = M[[op]](M[[t_1]] \rho, \ldots) = M[[op \cdot t_1 \cdots]] \rho = !_s([op \cdot t_1 \cdots]) = !_s(M[[op]]([t_1], \ldots))\).

Now suppose we have any \( T \)-homomorphism \( f : M_T \to M \). We will prove that \( f \) must equal \( ! \) by induction on the structure of each ground term \( t \in \Sigma_s(\emptyset) \). If \( t \) is a constant \( c \) then (6.2) requires that \( f_s : M_T[c] = M[c] \), so \( f_s([c]) = M[c] \rho = !_s([c]) \). If \( t = op \cdot t_1 \cdots t_n \) then \( f_s([op \cdot t_1 \cdots]) = f_s \circ M_T[op]([t_1], \ldots) = M[op] \circ (f_s([t_1]), \ldots) \) by (6.2); we then apply our induction hypothesis and (6.2) to get our result.

\[\square\]

6.2 Free \( T \)-algebras

Definition 6.2.1. Given a NEL-signature \( \Sigma \), let \( \mathcal{FM}_{\Sigma} \) be the category whose

- Objects \( X \) are equivariant maps \( s \mapsto X_s \) from the sorts of \( \text{Sort}_\Sigma \) to FM-sets;

- Arrows \( f : X \to X' \) are equivariant maps \( s \mapsto f_s \) from the sorts of \( \text{Sort}_\Sigma \) to FM-functions, so that each \( f_s \) is an FM-function \( X_s \to X'_s \).

If \( \Sigma \) is a NEL-theory over \( \Sigma \), we define the forgetful functor \( U : \mathcal{T}_{\text{Alg}} \to \mathcal{FM}_{\Sigma} \) by \((UM)_s = M\llbracket s\rrbracket\) and \((Uf)_s = f_s \) for each \( T \)-homomorphism \( f : M \to M' \).

Definition 6.2.2. Given a \( \mathcal{FM}_{\Sigma} \)-object \( X \), we define the comma category \( X \downarrow U \) as usual [37, Section II.6]:

- As objects \( X \downarrow U \) has pairs \((M, f)\) where \( M \) is a \( T \)-algebra and \( f \) is a \( \mathcal{FM}_{\Sigma} \)-arrow \( X \to UM \);

- As arrows \( g : (M, f) \to (M', f') \), \( X \downarrow U \) has \( T \)-homomorphisms \( g : M \to M' \) such that \( Ug \circ f = f' \):

\[
f' \Rightarrow f \Rightarrow f'

de Uf Uf

\[\text{(6.4)}\]
As with our proof of completeness in Section 5.4, the proof of the existence of free algebras will involve defining a new signature by adding new constants.

**Definition 6.2.3.** Given a $\mathcal{F}M_{\Sigma}$-object $X$ and a NEL-signature $\Sigma$, define a NEL-signature $\Sigma[X]$ by setting $\text{Sort}_{\Sigma[X]} \triangleq \text{Sort}_{\Sigma}$ and

$$\text{Op}_{\Sigma[X]} \triangleq \text{Op}_{\Sigma} + \bigoplus_{s \in \text{Sort}_{\Sigma}} X_s$$

(6.5)

where each $x \in X_s$ is a constant of type $x : s$. Let $\mathcal{T}[X]$ be the theory with signature $\Sigma[X]$ and the same axioms as $\mathcal{T}$.

**Lemma 6.2.4.** Take an $\mathcal{F}M_{\Sigma}$-object $X$ and theory $\mathcal{T}$. Then, following Definitions 6.2.2 and 6.2.3,

$$X \downarrow U \cong \mathcal{T}[X]_{\text{Alg}}.$$

**Proof.** We define our first functor $G : X \downarrow U \to \mathcal{T}[X]_{\text{Alg}}$ by

- $G(M, f)[s] = M[s]$;
- $G(M, f)[\text{op}] = \begin{cases} f_s(x) & \text{op} = x \in X_s \\ M[\text{op}] & \text{otherwise}; \end{cases}$
- $G$ is the identity on arrows of $X \downarrow U$. That is, given $g : (M, f) \to (M', f')$, $Gg = g$ is defined by FM-functions $g_s : M[s] \to M'[s]$.

Because $G$ is the identity on arrows it clearly preserves identities and compositions. The map $\text{op} \mapsto G(M, f)[\text{op}]$ is equivariant because $\pi \cdot M[\text{op}] = M[\pi \cdot \text{op}]$ and $\pi \cdot f_s(x) = f_s(\pi(x))$, so $G(M, f)$ is a $\Sigma[X]$-structure. Now for any $t \in \text{Term}_{\Sigma}$ and valuation $\rho$ it is clear that $M[t] \rho = G(M, f)[t] \rho$. But the axioms of $\mathcal{T}[X]$ only contain such terms in $\text{Term}_{\Sigma}$ (that is, they do not contain any of the new constants) so because $M$ is a $\mathcal{T}$-algebra, $G(M, f)$ is also. Finally, we need to show that given $g : (M, f) \to (M', f')$, $g$ is a $\mathcal{T}[X]$-homomorphism $G(M, f) \to G(M', f')$. (6.2) holds for $\text{op} \in \text{Op}_{\Sigma}$ because $g$ is a $\mathcal{T}$-homomorphism, and for $x \in X_s$ because $g_s \circ f_s(x) = f'_s(x)$ by (6.4).

We define our second functor $H : \mathcal{T}[X]_{\text{Alg}} \to X \downarrow U$ by $HM = (H^M, h^M)$, where

- $H^M[s] = M[s]$ and $H^M[\text{op}] = M[\text{op}]$ for all $s \in \text{Sort}_{\Sigma}, \text{op} \in \text{Op}_{\Sigma}$;
- $h^M : X \to UH^M$ is defined by FM-functions $h^M_s : X_s \to M[s]$ mapping $x \mapsto M[x]$;
- $H$ is the identity on $\mathcal{T}[X]$-homomorphisms.

$H^M$ is a $\mathcal{T}$-algebra because $M$ is a $\mathcal{T}[X]$-algebra. The map $s \mapsto h^M_s$ is equivariant because for all $x \in X_{\pi,s}$, $(\pi \cdot h^M_s)(x) = \pi \cdot (h^M_s(\pi^{-1} \cdot x)) = \pi \cdot M[\pi^{-1} \cdot x] = M[x] = h^M_{\pi s}(x)$. Finally, if
We can unpack the details of this theorem as follows:

**Theorem 6.2.6.** The functor $\mathbb{T}[X]$ has a left adjoint.

**Proof.** It suffices to show that for each $\mathcal{F}_\Sigma$-object $X$ there is a universal arrow $(FX, \eta_X)$ from $X$ to $U$, where $FX$ is a $\mathbb{T}$-algebra and $\eta_X : X \to UFX$ is a $\mathcal{F}_\Sigma$-arrow.

$M_{\mathbb{T}[X]}$ is the initial $\mathbb{T}[X]$-algebra by Lemma 6.1.3. Therefore $X \downarrow U$ has an initial object by Lemma 6.2.4, namely $H\mathbb{T}[X] = \langle H^{M_{\mathbb{T}[X]}}, h^{M_{\mathbb{T}[X]}}, \rangle$. By Lemma 6.2.5 this is universal from $X$ to $U$.

We can unpack the details of this theorem as follows:

- Given a NEL-theory $\mathbb{T}$ and $\mathcal{F}_\Sigma$-object $X$, the free $\mathbb{T}$-algebra on $X$, $FX$, is $H^{M_{\mathbb{T}[X]}}$; that is, the ground term $\mathbb{T}[X]$-algebra considered as a $\mathbb{T}$-algebra.

- The unit $\eta : 1 \to UF$ (where 1 is the identity functor on $\mathcal{F}_\Sigma$) of the adjunction is defined by components $\eta_X = h^{M_{\mathbb{T}[X]}}, X \to UFX$ mapping $x \in X_s$ to $[x] \in FX[s]$.

- The adjunction

$$\mathbb{T}_{\mathcal{Alg}}(FX, M) \cong \mathcal{F}_\Sigma(X, U M) \quad (6.6)$$

is defined by sending $\mathbb{T}$-homomorphisms $f : FX \to M$ to $f \circ \eta_X : X \to UFX \to UM$, and by sending $\mathcal{F}_\Sigma$-arrows $f : X \to UM$ to the unique $\mathbb{T}$-homomorphism $\hat{f} : FX \to M$ such that $U \hat{f} \circ \eta_X = Uf$. This is equivalent (by Lemmas 6.2.5 and 6.2.4) to the unique $\mathbb{T}[X]$-homomorphism $! : M_{\mathbb{T}[X]} \to G(M, f)$, which is defined concretely by Lemma 6.1.3 and 6.2.4 as

$$\hat{f}_s([t]) = \begin{cases} f_s(x) & t = x \in X_s \\ M[[op]](\hat{f}_{s_1}([t_1]), \ldots, \hat{f}_{s_n}([t_n])) & t = op t_1 \cdots t_n \end{cases} \quad (6.7)$$
• The free functor \( F : \mathcal{F}_\Sigma \to \mathbb{T}_{\text{Alg}} \) sends \( \mathcal{F}_\Sigma \)-objects \( X \) to the free \( \mathbb{T} \)-algebra on \( X \) defined above, and \( \mathcal{F}_\Sigma \)-arrows \( f : X \to X' \) to \( Ff = \eta_{X'} \circ f : FX \to FX' \);

• The counit \( \varepsilon : FU \to 1 \) (where 1 is the identity on \( \mathbb{T}_{\text{Alg}} \)) is defined by components \( \varepsilon_M = \eta U M \).

6.3 Monadicity

Definition 6.3.1. [37, Chapter VI] A monad on a category \( C \) is a triple \( (T, \eta, \mu) \), where \( T \) is an endofunctor \( C \to C \), \( \eta \) is a natural transformation \( 1 \to T \) and \( \mu \) is a natural transformation \( T^2 \to T \), such that

\[
\begin{array}{ccc}
T^3 & \xrightarrow{T\mu} & T^2 \\
\mu T & \downarrow & \mu \\
T^2 & \xrightarrow{\mu} & T \\
\end{array}
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (6.8)
\]

commutes, where \( T\mu \) has components \( (T\mu)_X = T(\mu_X) \), \( \mu T \) has components \( (\mu T)_X = \mu_{TX} \), and similarly for \( T\eta \) and \( \eta T \).

Every adjunction gives rise to a monad. In particular, we are interested in the monad defined by the adjunction of the previous section:

Definition 6.3.2. Given a NEL-theory \( T \), the \( T \)-monad on \( \mathcal{F}_\Sigma \) is specified by a triple \( T = (UF, \eta, U\varepsilon F) \), where \( U, F, \eta \) and \( \varepsilon \) are as defined in Section 6.2.

Definition 6.3.3. A \( T \)-algebra is a pair \( (X, \alpha_X) \) where \( X \) is a \( \mathcal{F}_\Sigma \)-object and \( \alpha_X \) is an \( \mathcal{F}_\Sigma \)-arrow such that

\[
\begin{array}{ccc}
UF & \xrightarrow{UF \alpha_X} & UX \\
\varepsilon F_X & \downarrow & \alpha_X \\
UFX & \xrightarrow{\alpha_X} & X \\
\end{array}
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (6.9)
\]

commutes. A \( T \)-homomorphism \( f : (X, \alpha_X) \to (Y, \alpha_Y) \) is a \( \mathcal{F}_\Sigma \)-arrow \( f : X \to Y \) such that

\[
\begin{array}{ccc}
UF & \xrightarrow{UF f} & UFY \\
\alpha_X & \downarrow & \alpha_Y \\
X & \xrightarrow{f} & Y \\
\end{array}
\quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad (6.10)
\]

commutes. The \( T \)-algebras and \( T \)-homomorphisms form the Eilenberg-Moore category of \( T \)-algebras, which we will call \( T_{\text{Alg}} \).

We wish to define an isomorphism between the category of \( \mathbb{T} \)-algebras introduced in Section 6.1 and the category of \( T \)-algebras for the \( \mathbb{T} \)-monad \( T \).
**Definition 6.3.4.** The comparison functor $C : \mathbb{T}_{\text{Alg}} \to \mathbb{T}_{\text{Alg}}$ is defined by setting
\[ CM \triangleq (UM, U\varepsilon_M); \quad Cf \triangleq Uf \]
for all $\mathbb{T}$-algebras $M$ and $\mathbb{T}$-homomorphisms $f$.

We will use some well-known category theory [37, Section VI.7] to show that $C$ is an isomorphism:

**Definition 6.3.5.** Parallel arrows $f, g : X \to Y$ in a category $C$ have a split coequaliser $q : Y \to Q$ if we have arrows
\[ X \leftarrow w Y \overline{\leftarrow} q \rightarrow Q \]
(6.11)
such that $q \circ f = q \circ g$, $q \circ v = \text{id}_Q$, $f \circ w = \text{id}_Y$ and $g \circ w = v \circ q$. It is the case that $q$ coequalises $f, g$: if we have some $r : Y \to R$ such that $r \circ f = r \circ g$ then the unique arrow induced is $r \circ v : Q \to R$.

**Theorem 6.3.6** (Beck’s Precise Tripleability Theorem). The comparison functor of Definition 6.3.4 is an isomorphism if and only if $U : \mathbb{T}_{\text{Alg}} \to \mathcal{FM}_\Sigma$ creates coequalisers for those parallel pairs of $\mathbb{T}$-homomorphisms $f, g$ for which $Uf, Ug$ have a split coequaliser in $\mathcal{FM}_\Sigma$.

With these preliminaries out the way, we may proceed with the parts of the proof that are specific to Nominal Equational Logic.

**Lemma 6.3.7.** Suppose we have a NEL-theory $\mathbb{T}$ over a signature $\Sigma$, a sorting environment $\Gamma \in \mathbb{SE}_\Sigma$ and a term $t \in \Sigma_\delta(\Gamma)$. Suppose further that we have $\mathbb{T}$-algebras $M, M'$, a valuation $\rho \in M'[[\Gamma]]$ and a $\mathbb{T}$-homomorphism $f : M \to M'$. Then if $Uf$ has a right inverse $g : UM' \to UM$ in $\mathcal{FM}_\Sigma$ (that is, $Uf \circ g = \text{id}_{UM'}$) then
\[ f_s(M[[t]](g \circ \rho)) = M'[[t]]\rho \]
where $(g \circ \rho)(x) = g' \circ \rho(x)$ for $\Gamma(x) = s'$.

**Proof.** The proof proceeds by induction on the structure of $t$:

In the suspension case, $f_s(M[[\pi x]](g \circ \rho)) = f_s(\pi \cdot (g_{\pi^{-1} \cdot \pi} \circ \rho(x))) = f_s \circ g_\pi(\pi \cdot \rho(x)) = \pi \cdot \rho(x) = M'[[\pi x]]\rho$.

In the constructed term case, $f_s(M[[op t_1 \cdots]](g \circ \rho)) = f_s \circ M[[op]](M[[t_1]](g \circ \rho), \ldots) = M'[[op]](f_{s_1}(M[[t_1]](g \circ \rho)), \ldots)$ by (6.2), with the rest of the proof following by induction.

**Theorem 6.3.8.** The comparison functor $C : \mathbb{T}_{\text{Alg}} \to \mathbb{T}_{\text{Alg}}$ of Definition 6.3.4 is an isomorphism, so
\[ \mathbb{T}_{\text{Alg}} \cong \mathbb{T}_{\text{Alg}} \]
Proof. The result follows by Beck’s Theorem 6.3.6. Take parallel \( \mathbb{T} \)-homomorphisms \( f, g : M \rightarrow M' \) such that \( U f, U g \) have a split coequaliser (Definition 6.3.5) \( q : U M' \rightarrow Q \) with arrows \( v : Q \rightarrow U M', w : U M' \rightarrow UM \) in \( \mathcal{F}_M \Sigma \):

\[
\begin{array}{ccc}
UM & \xrightarrow{w} & U M' \\
\downarrow{U f} & \\ \\
UM & \xrightarrow{v} & Q
\end{array}
\]  \hspace{1cm} (6.12)

We make \( Q \) a \( \Sigma \)-structure by setting \( Q[s] \triangleq Qs \) and, given \( op : s \rightarrow t \),

\[
Q[op] \triangleq q_s \circ M'[op] \circ v_t . \hspace{1cm} (6.13)
\]

where if \( s = [s_1, \ldots, s_n] \) then \( v_t = v_{s_1} \times v_{s_n} \). \( q \) is therefore a \( \mathbb{T} \)-homomorphism \( M' \rightarrow Q \) because

\[
\begin{array}{ccc}
M'[s] & \xrightarrow{id} & M'[s] \\
\downarrow{q_s} & \\
Q[s] & \xrightarrow{v_t} & Q[s]
\end{array}
\]

commutes by (6.2) and Definition 6.3.5. In fact, for \( q \) to be a \( \mathbb{T} \)-homomorphism \( Q[op] \) must be defined as (6.13): given any such definition, \( Q[op] = Q[op] \circ q_s \circ v_t = q \circ M'[op] \circ v_t \).

Now to see that \( Q \) is a \( \mathbb{T} \)-algebra, take a \( \mathbb{T} \)-axiom \( \nabla \vdash \bar{\pi} \# t \approx t' : s \) and \( \rho \in Q[\nabla] \). The map \( s \mapsto v_s \) is equivariant for all \( s \in \text{Sort}_\Sigma \), so \( v \circ \rho \in M'[\nabla] \). But \( M' \) is a \( \mathbb{T} \)-algebra, so

\[
\bar{\pi} \# M'[t](v \circ \rho) = M'[t'](v \circ \rho) . \hspace{1cm} (6.15)
\]

\( \bar{\pi} \# s \) and \( \text{supp}(q_s) \subseteq \text{supp}(s) \), so we can apply \( q_s \) to get \( \bar{\pi} \# q_s(M'[t](v \circ \rho)) = q_s(M'[t'](v \circ \rho)) \). We then apply Lemma 6.3.7 to conclude that \( \bar{\pi} \# Q[t]\rho = Q[t']\rho \).

Finally, we must show that \( q \) is a coequaliser in \( T_{\text{Alg}} \). We know from Definition 6.3.5 that \( q \) coequalises \( U f, U g \) in \( \mathcal{F}_M \Sigma \), so any \( \mathbb{T} \)-homomorphism \( r : M' \rightarrow R \) such that \( r \circ f = r \circ g \) induces the unique \( \mathcal{F}_M \Sigma \)-arrow \( U r \circ v : Q \rightarrow UR \). In fact \( U r \circ v \) is a \( \mathbb{T} \)-homomorphism because

\[
R[op] \circ r_t \circ v_t = r_s \circ M[op] \circ v_t = r_s \circ f_s \circ w_s \circ M[op] \circ v_t = r_s \circ g_s \circ w_s \circ M[op] \circ v_t = r_s \circ v_s \circ q_s \circ M[op] \circ v_t = r_s \circ v_s \circ Q[op] .
\]

\( \Box \)
Chapter 7

Nominal Lawvere theories

This chapter will generalise the results of the previous chapters from the category of FM-sets to any category with certain structure. We will refer to such a category as an FM-category. Given a NEL-theory we will show that a small FM-category called the classifying category for that theory can be defined. This will lead to a new completeness proof and two key correspondences, analogous to those proved by Lawvere [35] for standard equational logic: the first is between algebras for a theory in a small FM-category and structure preserving functors from the classifying category to that category, and the second is between NEL-theories and small FM-categories. [12, Chapter 3] gives an accessible introduction to these constructions in the case of equational logic.

7.1 Internal permutation actions

**Notation 7.1.1.** Given a category $C$ we will refer to the collection of $C$-objects as $ob\ C$ and $C$-arrows as $ar\ C$. The categories we are most interested in will be small, so it is safe to think of $ob\ C$ and $ar\ C$ as sets.

Lawvere’s result identifies equational theories with small categories with finite products. This section will define the first piece of additional categorial structure required to obtain a Lawvere-style result for NEL; the remaining structure will be defined in the next section. The paradigmatic example of a category with such structure will be the category $\mathcal{F.M}$-$\text{Set}$, as defined by Definition 2.5.2, so we will first describe the internal permutation action on FM-sets and then give the categorial generalisation.

**Example 7.1.2.** The internal permutation action on $\mathcal{F.M}$-$\text{Set}$ for each FM-set $X$ and permutation $\pi$ is the FM-arrow $\pi_X : X \rightarrow \pi \cdot X$ defined by

\[
\pi_X(x) \triangleq \pi \cdot x .
\]
In fact the usual permutation actions on \( \text{ob} \mathcal{FM}-\text{Set} \) (2.24) and \( \text{ar} \mathcal{FM}-\text{Set} \) (2.18) can be defined in terms of the internal permutation action as follows: let \( \pi \cdot X \) be the codomain of \( \pi_X \) and, given \( f : X \to Y \), let \( \pi \cdot f : \pi \cdot X \to \pi \cdot Y \) be

\[
\pi \cdot f \triangleq \pi_Y \circ f \circ (\pi^{-1})_{\pi \cdot X}.
\]

(7.2)

We may now proceed to the general definition.

**Definition 7.1.3.** A category \( \mathcal{C} \) has an *internal permutation action* if for each \( \pi \in \text{Perm} \) and \( C \in \text{ob} \mathcal{C} \) there is a \( \mathcal{C} \)-arrow \( \pi_C \) with domain \( C \) such that

(i) \( \iota_C \) is the identity \( \text{id}_C \);

(ii) \( (\pi' \pi)_C = \pi'_C \circ \pi_C \), where \( \pi \cdot C \) is defined to be the codomain of \( \pi_C \).

Where the object \( C \) is clear from context we will write \( \pi_C \) simply as \( \pi \).

**Remark 7.1.4.** As an aside, Definition 7.1.3 is a special case of the concept of *cofunctor*\(^1\) [2, Chapter 4], which is itself a generalisation of the *comorphisms* of [29]. Given categories \( \mathcal{C}, \mathcal{D} \), a cofunctor \( F : \mathcal{C} \to \mathcal{D} \) is a pair \( (F_0, F_1) \) where

- \( F_0 \) is a map \( \text{ob} \mathcal{D} \to \text{ob} \mathcal{C} \) (note that this map is backwards with respect to the direction of the cofunctor);

- \( F_1 \) is a map from pairs \((f, d)\) where \( f \in \text{ar} \mathcal{C}, d \in \text{ob} \mathcal{D} \) and the domain of \( f \) is \( F_0 d \), to \( \mathcal{D} \)-arrows such that

\[
(1) \quad \text{domain}(F_1(f, d)) = d;
\]

\[
(2) \quad F_0(\text{codomain}(F_1(f, d))) = \text{codomain}(f);
\]

\[
(3) \quad \text{The usual functorial conditions apply to } F_1, \text{ preserving identities and compositions.}
\]

This is dual to the definition of functor in the sense described in [2, Section 4.3]. Now if we take \( \text{Perm} \) as the category with one object whose arrows are permutations then we can see that an internal permutation action on a category \( \mathcal{C} \) is equivalent to defining a cofunctor \( \text{Perm} \to \mathcal{C} \): \( F_0 \) and condition (2) are trivial, while (1) ensures that \( F_1(\pi, C) \) has domain \( C \) and (3) follows directly from the conditions of Definition 7.1.3.\(^2\)

We now prove some results regarding the internal permutation action.

**Lemma 7.1.5.** Suppose \( \mathcal{C} \) is a small category with an internal permutation action \( (\pi, C) \mapsto \pi_C \). Then

---

\(^1\)The term *cofunctor* is sometimes also used as an abbreviation for *contravariant functor*.

\(^2\)We are grateful to Richard Garner for bringing the concept of cofunctor to our attention.
(i) \((\pi, C) \mapsto \pi \cdot C\), where \(\pi \cdot C\) is defined to be the codomain of \(\pi_C\), defines a permutation action on \(\text{ob} \, C\);

(ii) Given a \(C\)-arrow \(f : C \to D\), define \(\pi \cdot f : \pi \cdot C \to \pi \cdot D\) to be

\[
\pi \cdot C \xrightarrow{(\pi^{-1}) \pi_C} C \xrightarrow{f} D \xrightarrow{\pi_D} \pi \cdot D.
\]

Then \((\pi, f) \mapsto \pi \cdot f\) defines a permutation action on \(\text{ar} \, C\).

(iii) If every arrow in \(\text{ar} \, C\) is finitely supported with respect to the permutation action defined by (7.3) then every object in \(\text{ob} \, C\) is also finitely supported.

Proof. (i) and (ii) follow because the conditions of Definition 7.1.3 satisfy (2.4). (iii) follows because if \(a \neq \text{id}_X\) then \(a \neq X\).

Lemma 7.1.6. The following results hold for the internal permutation action:

(i) The map \((\pi, C) \mapsto \pi_C\) is equivariant, in that \(\pi \cdot (\pi'_C) = (\pi \pi' \pi^{-1}) \pi_C\).

(ii) Each \(\pi_C\) is iso, with inverse \((\pi^{-1}) \pi_C : \pi \cdot C \to C\);

(iii) The maps \(\pi \cdot ()\) on \(\text{ob} \, C\) and \(\text{ar} \, C\) define an endofunctor \(C \to C\);

(iv) The \(C\)-arrows \(\pi_C : C \to \pi \cdot C\) are the components of a natural transformation \(\pi : \text{id}_C \to \pi \cdot ()\), where \(\text{id}_C\) is the identity functor on \(C\).

Proof. (i) follows from (7.3) and Definition 7.1.3(ii). (ii) follows immediately from Definition 7.1.3, while (iii) and (iv) follow from (ii).

Definition 7.1.7. An internal permutation action on a category \(C\) is finitely supported if every arrow in \(\text{ar} \, C\) is finitely supported with respect to the permutation action defined by (7.3).

We provide a counter-example to show that Definition 7.1.7 is not trivial:

Example 7.1.8. The category whose objects are nominal sets and arrows are functions between them has an internal permutation action, defined by (7.1), but that action is not in general finitely supported; see Lemma 2.4.5 for an example of such a function that cannot be finitely supported.

Lemma 7.1.9. A small category \(C\) with a finitely supported permutation action is an internal category in the category of nominal sets \(\text{Nom}\) defined by Definition 2.4.1.

Proof. Lemma 7.1.5 establishes that \(\text{ob} \, C\) and \(\text{ar} \, C\) are \(\text{Nom}\)-objects. Following [33, Chapter B2], we need to show that the domain, codomain, identity and composition maps are equivariant functions. Domain and codomain follow immediately by (7.3). Identity follows by Definition 7.1.3(i). Composition

\[
\pi \cdot (g \circ f) = (\pi \cdot g) \circ (\pi \cdot f)
\]

follows by Definition 7.1.3(ii).
While any small category with a finitely supported permutation action is internal to $\mathcal{N}om$, the converse need not apply. What is key here is that the category is ‘aware’ of its own permutation action, in the sense that it is represented by arrows within the category.

**Lemma 7.1.10.** Suppose we have categories $C, C'$ equipped with internal permutation actions, and a functor $F : C \to C'$ that strictly preserves that action. That is,

$$F(\pi_C) = \pi_{FC}$$

for all $C \in \text{ob } C$. Then

(i) $F$ is a functor in $\mathcal{N}om$;

(ii) Given two such functors $F, F'$, any natural transformation $\phi : F \Rightarrow F'$ is a natural transformation in $\mathcal{N}om$.

**Proof.** (i): We need to show that $\pi \cdot FC = F(\pi \cdot C)$ and, for all $f : C \in C'$ in $\text{ar } C$, $\pi \cdot Ff = F(\pi \cdot f)$. $\pi \cdot C$ is defined to be the codomain of $\pi_C$, so $F(\pi \cdot C)$ is the codomain of $F(\pi_C) = \pi_{FC}$, which is indeed $\pi \cdot FC$. Now $\pi \cdot Ff \circ \pi_{FC} = \pi_{FC'} \circ Ff$ by Lemma 7.1.6(iv), which is $F(\pi_{C'}) \circ Ff = F(\pi \cdot f) \circ F(\pi_C) = F(\pi \cdot f) \circ \pi_{FC}$. But $\pi_{FC}$ is iso by Lemma 7.1.6(ii) and therefore epi.

(ii): $\pi \cdot (\phi_C) = \phi_{\pi \cdot C}$ follows by Lemma 7.1.6(iv). 

### 7.2 FM-categories

This section will introduce the remaining categorial structure required to obtain a Lawvere-style result for NEL. Again, we will describe this structure for $\mathcal{F}\mathcal{M}\text{-Set}$ first to provide intuition for the more general definition.

**Example 7.2.1.** Just as Lawvere’s classical result requires the existence of finite products, so shall the results of this chapter. But we will further require that these finite products be *equivariant* with respect to the category’s permutation action, in a sense that will be defined formally in Definition 7.2.2. It will be clear that the standard Cartesian products in $\mathcal{F}\mathcal{M}\text{-Set}$ define equivariant finite products.

Given any FM-set $X$ and finite set of atoms $\overline{\pi} \# X$ we can define\(^3\) a new FM-set by

$$X^{\overline{\pi}} \triangleq \{ x \in X \mid \overline{\pi} \# x \} .$$

We call the injection $i_X^{\overline{\pi}} : X^{\overline{\pi}} \hookrightarrow X$ a *fresh-inclusion*.

\(^3\)In fact the definition (7.6) does not require that $\overline{\pi} \# X$, but this restriction will prove useful in developing the results of this section.
**Definition 7.2.2.** Take a category \( C \) with an internal permutation action and finite products defined by a terminal object \( 1 \) and, for each pair of objects \( C_1, C_2 \), a product \( C_1 \times C_2 \) with projections \( pr_i(C_1, C_2) : C_1 \times C_2 \to C_i \) for \( i = 1, 2 \). Then \( C \) has **equivariant finite products** if for all \( \pi \in \text{Perm} \),

1. \( \pi \cdot 1 = 1 \);
2. \( \pi \cdot pr_i(C_1, C_2) = pr_i(\pi \cdot C_1, \pi \cdot C_2) \) for \( i = 1, 2 \).

Note that (i) implies that \( \pi \cdot !_C = !_{\pi \cdot C} \), where \( !_C, !_{\pi \cdot C} \) are the unique arrows from \( C, \pi \cdot C \) respectively, and that (ii) implies that \( \pi \cdot (C_1 \times C_2) = (\pi \cdot C_1, \pi \cdot C_2) \).

Our next example shows that requiring our finite products be equivariant is not trivial:

**Example 7.2.3.** Consider the category whose objects are finite sets of atoms and arrows are functions between them. We can define finitely supported permutation actions on the objects and arrows of this category in the usual way:

\[
\pi \cdot a \triangleq \{ \pi(a) \mid a \in a \} ; \\
(\pi \cdot f)(a) \triangleq \pi \cdot (f(\pi^{-1}(a))) .
\] (7.7)

This category has a terminal object; any singleton \( \{a\} \) will do. But it does not have an equivariant terminal object, as no singleton has empty support. Following this example, a category with all finite products but not equivariant finite products can readily be imagined.

The next definition is a generalisation of the fresh-inclusions that were defined by Example 7.2.1. The concept defines freshness in a category in terms of the equality of certain arrows in that category, and as such can be seen to be related to our previous results defining freshness in terms of equality, Lemmas 2.3.5 and 5.5.1.

**Definition 7.2.4.** Let \( C \) have a finitely supported internal permutation action and equivariant finite products. Then \( C \) has **fresh-inclusions** if for every finite set of atoms \( \pi \subseteq A \) and \( C \)-object \( C \) such that \( \pi \# C \) we have a \( C \)-arrow \( i_\pi^C \) with codomain \( C \) for which the following properties hold:

1. **(Equivariance):** \( \pi \cdot i_\pi^C = i_{\pi \cdot C}^\pi \);
2. **(Sets of Atoms):** \( i_\emptyset^C = id_C \) and \( i_\pi^C \circ i_{\pi \cdot C}^C = i_{\pi \cup \pi'}^C \), where \( C^{\#\pi} \) is defined to be the domain of \( i_\pi^C \);
3. **(Products):** \( i_{C_1 \times C_2}^\pi = i_{C_1}^\pi \times i_{C_2}^\pi \);
4. **(Internal permutation action):** If \( \text{supp}(\pi) \# C \) then \( \pi_{C^{\#\text{supp}(\pi)}} \) is equal to the identity \( id_{C^{\#\text{supp}(\pi)}} \),
(v) \((Epi \text{ When Fresh})\): If we have parallel \(C\)-arrows \(f, g : C \to D\) such that \(f \circ i_C^\pi = g \circ i_C^\pi\) and \(\pi \neq (f, g)\), then \(f = g\);

(vi) \((Freshness)\): Given a \(C\)-arrow \(f : C \to D\), a permutation \(\pi\) such that \(supp(\pi) \neq D\) and set of atoms \(\pi \neq (\pi \cdot \pi, f)\), if \(\pi_D \circ f \circ i_C^\pi = f \circ i_C^\pi\), then we have a unique \(\hat{f} : C \to D\) such that \(i_D^\pi \circ \hat{f} = f\):

\[
\begin{array}{c}
\xrightarrow{i_C^\pi} C & \xrightarrow{f} & D \xleftarrow{i_D^\pi} \pi_D \\
\end{array}
\]  

(7.8)

The fresh-inclusions in (7.8) are well-defined because \(\pi \neq f\) implies \(\pi \neq C\) because the domain map is equivariant (Lemma 7.1.9), and \(\pi \neq \pi \cdot \pi\) implies that \(\pi \cdot \pi \subseteq supp(\pi) \neq D\).

We will write \(i_C^\pi\) as \(\overline{i}\) or \(i\) where the object or set of atoms is clear from context.

Remark 7.2.5. The properties of Definition 7.2.4 are defined by the arrows \(i_C^\pi\) rather than their domain objects \(C^\pi\). However, it should be noted that conditions (i)-(iii) have the immediate consequences

(i) \(\pi \cdot (C^\pi) = (\pi \cdot C)^\pi\);

(ii) \((C^\emptyset) = C\) and \((C^\pi) \cup \pi = C^\pi\);

(iii) \((C_1 \times C_2)^\pi = C_1^\pi \times C_2^\pi\).

Lemma 7.2.6. The \(FM\)-sets \(X^\pi\) defined by (7.6) and the inclusion \(FM\)-functions \(i_X : X^\pi \to X\) define fresh-inclusions in \(FM\)-Set.

Proof. Conditions (i)-(iii) of Definition 7.2.4 are trivial consequences of the fresh-inclusions being inclusion functions. (iv) follows because for all \(x \in X^{supp(\pi)}\), \(\pi \cdot x = x\), so \(\pi_X^{supp(\pi)} = id_X^{\pi \cdot supp(\pi)}\).

(v): Say \(f(x) = g(x)\) for all \(x \in X^\pi\). Then for any \(x \in X\) let \(\bar{a} \in A^n\) be an ordering of \(\pi\) and \(\bar{a}' \in A^n\) be a fresh tuple with \(supp(\bar{a}') \neq (\bar{a}, f, g)\). Then \(\pi \neq (\bar{a} \cdot \bar{a}) \cdot x\), so \((\bar{a} \cdot f) = f((\bar{a} \cdot x) = g((\bar{a} \cdot g(x)\cdot x = (\bar{a} \cdot g(x)\cdot x\cdot x)\cdot x\cdot x). Apply (\bar{a} \cdot \bar{a}') to both sides to get \(f(x) = g(x)\).

(vi): The top line of (7.8) means that \(\pi \neq \pi\) implies \(\pi \cdot f(x) = f(x)\), so \(\pi \cdot \pi \neq f(x)\). Let \(\bar{a} \in A^n\) be an ordering of \(\pi \cdot \pi\) and \(\bar{a}' \in A^n\) be a fresh tuple with \(supp(\bar{a}') \neq (\pi \cdot \bar{a}', f)\). Then for any \(x, \pi \neq (\bar{a} \cdot \bar{a}') \cdot x\), so \(\pi \cdot \pi \neq f((\bar{a} \cdot \bar{a}') \cdot x) = (\bar{a} \cdot \bar{a}') \cdot f(x)\). Applying (\bar{a} \cdot \bar{a}') to both sides yields \(\pi \cdot \pi \neq f(x)\), so we can define our unique arrow \(\hat{f}\) as the restriction of \(f\).

Remark 7.2.7. The most complex part of Definition 7.2.4 is the ‘Freshness’ condition (vi). To provide some intuition into this condition, consider the special case where \(C = FM\)-Set,
\[ \overline{a} = \{a'\} \text{ and } \pi = (a' a'), \text{ so we have} \]
\[ X^{\#(a')} \xrightarrow{i} X \xrightarrow{f} Y^{\#(a')} \]
\[ \xymatrix{ X^{\#(a')} \ar[r]^i \ar[d]_f & X \ar[r]^f \ar@{~}[d] & Y^{\#(a')} \ar@/^/[l]_i } \]  

(7.9)

The top line of (7.9), along with the original definition, tell us that given \( x \in X \) and \( a' \neq (f, x) \) we have \((a' a') \cdot (f(x)) = f(x)\). The fact that \( i \circ \hat{f} = f \), where \( i \) is an injection, means that for all \( x \in X \), \( f(x) \in Y^{\#(a')} \) and so by definition \( a \neq f(x) \). These are equivalent by Lemma 2.3.3. In other words, the Freshness condition gives rise to an arrow-theoretic description of freshness. This fact will be exploited in the coming sections, in particular by Definition 7.3.5.

We now prove some results regarding fresh-inclusions.

**Lemma 7.2.8.** Suppose a category \( C \) has fresh-inclusions. Then

(i) If \( \overline{a} \neq (f : C \to D) \) there is a unique arrow \( f^{\#\pi} : C^{\#\pi} \to D^{\#\pi} \) such that \( f \circ i = i \circ f^{\#\pi} \);

(ii) This assignment is functorial, in the sense that \( (id_C)^{\#\pi} = id_{C^{\#\pi}} \) and \( (g \circ f)^{\#\pi} = g^{\#\pi} \circ f^{\#\pi} \).

**Proof.** (i): Let \( \tilde{a} \in \mathbb{A}^{(n)} \) be an ordering of \( \overline{a} \) and let \( \tilde{a}' \in \mathbb{A}^{(n)} \) be a fresh tuple of the same size. Then

\[ C^{\#\pi \cup \text{supp}(\tilde{a}')} \xrightarrow{i} C \xrightarrow{f} D \]

(7.10)

\[ \xymatrix{ C^{\#\pi \cup \text{supp}(\tilde{a}')} \ar[r]^i \ar[d]_{\tilde{a}'} & C \ar[r]^f \ar[d]_{\tilde{a}'} & D } \]

commutes by the naturality of \((\tilde{a}, \tilde{a}')\) (Lemma 7.1.6(iv)), the fact that \( \overline{a} \neq f \), and Definition 7.2.4(i). But \((\tilde{a}, \tilde{a}')_{C^{\#\pi \cup \text{supp}(\tilde{a}')}}\) is the identity by Definition 7.2.4(iv), so rearranging by Definition 7.2.4(ii) we have \((\tilde{a}, \tilde{a}') \circ (f \circ i^{\#\pi}) = (f \circ i^{\#\pi}) \circ i^{\#\pi}(\tilde{a})\). By Definition 7.2.4(iii) this induces a unique arrow \( C^{\#\pi} \to D^{\#\pi} \) with the required property. That is,

\[ f^{\#\pi} \triangleq f \circ i^{\#\pi} \tag{7.11} \]

in (7.8). (ii) then follows by the uniqueness property of (7.11). \( \square \)

**Lemma 7.2.9.** All fresh-inclusions \( i : C^{\#\pi} \to C \) in a category \( C \) are mono.

**Proof.** Say we have \( i^{\#\pi} \circ g = i^{\#\pi} \circ g' \) for \( g, g' : D \subseteq C^{\#\pi} \), and take \( \tilde{a} \) as an ordering of \( \overline{a} \) and \( \tilde{a}' \) as a fresh tuple of the same size. Then by reasoning similarly to that for Lemma 7.2.8(i) we can show that \((\tilde{a}, \tilde{a}') \circ i^{\#\pi} \circ g \circ i^{\#\pi}(\tilde{a}') = i^{\#\pi} \circ g \circ i^{\#\pi}(\tilde{a}')\), so by (7.8) we have a unique \((i \circ g)\) such that \( i \circ (i \circ g) = i \circ g = i \circ g' \). By uniqueness, \((i \circ g) = g = g'\). \( \square \)

**Definition 7.2.10.** A category with a finitely supported internal permutation action (Definition 7.1.3 and 7.1.7), equivariant finite products (Definition 7.2.2) and fresh-inclusions (Definition 7.2.4) is said to be an FM-category.

In particular, if the objects and arrows of this category form sets then it is a small FM-category.
Example 7.2.11. From Examples 7.1.2 and 7.2.1 and Lemma 7.2.6 it can be seen that $\mathcal{FM}$-Set is an FM-category. Further, these constructions can be made within the category $\mathcal{FM}_\lambda$-Set for any limit ordinal $\lambda$ (Definition 2.5.2), so $\mathcal{FM}_\lambda$-Set is a small FM-category.

Other examples of FM-categories that are known at this stage are derived by adding some extra structure to $\mathcal{FM}$-Set, such as FM-sets equipped with a restriction structure (Example 4.5.4(iii)) and the FM-cpos of [55] which can be applied to denotational semantics. Section 7.4 will introduce another important example of an FM-category.

Definition 7.2.12. Given FM-categories $\mathcal{C}, \mathcal{C}'$, an FM-functor between them is a functor $F : \mathcal{C} \to \mathcal{C}'$ that strictly preserves

(i) the internal permutation action: $F(\pi_\mathcal{C}) = \pi_{\mathcal{F}C}$;

(ii) finite products: $F(1_{\mathcal{C}}) = 1_{\mathcal{F}C}$ and $F(pr_i(C_1, C_2)) = pr_i(\mathcal{F}C_1, \mathcal{F}C_2)$ for $i = 1, 2$;

(iii) fresh-inclusions: $F(i_{\mathcal{C}}) = i_{\mathcal{F}C}$.

The category of FM-functors $\mathcal{C} \rightarrow \mathcal{C}'$ and natural transformations between them is called $\mathcal{FM}(\mathcal{C}, \mathcal{C}')$.

In fact, it is not necessary for the preservation of finite products to be strict; if we relax this condition then the results of the next sections can still be constructed, with the isomorphism of Theorem 7.5.1 becoming an equivalence. However, we will continue to present the results in terms of strict preservation of finite products simply for the sake of convenience.

7.3 Categorial algebra

This section will generalise the constructions of Chapters 3, 5 and 6 to any small FM-category, and prove that this generalisation is sound with respect to the proof rules for equality of Figure 5.3.

Definition 7.3.1 (c.f. Definition 3.1.4). Given a NEL-signature $\Sigma$ and small FM-category $\mathcal{C}$, a $\Sigma$-structure $M$ in $\mathcal{C}$ is defined by

(i) An equivariant function $M[\_] : \text{Sort}_\Sigma \to ob \, \mathcal{C}$,

(ii) An equivariant function $M[\_] : \text{Op}_\Sigma \to ar \, \mathcal{C}$, where if $op$ has type $[s_1, \ldots, s_n] \rightarrow s$ then

$$M[[op]] : M[[s_1]] \times \cdots \times M[[s_n]] \rightarrow M[[s]]. \quad (7.12)$$

As usual we follow Notation 3.2.8 and write $M[[s]]$ as $[s]$, and so forth where the structure in question is clear.
Lemma 7.3.4. The following hold for the above constructions:

(i) The map \( \nabla \mapsto [\nabla] \) is equivariant;

(ii) The map \((\nabla, t \in \Sigma_\nabla(\nabla')) \mapsto [\nabla \vdash t : s] \) is equivariant;

(iii) If \( \nabla \) is as (7.13) then \([\nabla \vdash t : s] = [x_1 : s_1, \ldots, x_n : s_n \vdash t : s] \circ (t_{[s_1]}^\nabla \times \cdots \times t_{[s_n]}^\nabla);\)

(iv) \([\nabla \vdash \pi \ast t : \pi \cdot s] = \pi_{[s]} \circ [\nabla \vdash t : s];\)

(v) Given \( \Gamma \in \text{SE}_\Sigma \) with domain \( \{x_1, \ldots, x_n\} \), a substitution \( \sigma \in \Sigma(\Gamma, \nabla') \) and a term \( t \in \Sigma_\nabla(\Gamma), \)

\([\nabla \vdash t(\sigma) : s] = [\Gamma \vdash t : s] \circ \langle \ldots, [\nabla \vdash \sigma(x_i) : \Gamma(x_i)], \ldots \rangle .\]

where \( \Gamma \) is defined to be a freshness environment in \( \text{FE}_\Sigma \) by \( \Gamma(x) = (\Gamma(x), \emptyset) \).

(vi) Given \( \nabla \leq \nabla' \) there exists an arrow \( \text{weak} : [\nabla'] \to [\nabla] \) such that for any \( t \in \Sigma_\nabla(\nabla'), \)

\([\nabla' \vdash t : s] = [\nabla \vdash t : s] \circ \text{weak};\)

(vii) \([\nabla^\ast \vdash t : s] = [\nabla \vdash t : s] \circ t^\ast.\)

Proof. (i) follows by the equivariance properties of Definitions 7.2.2, 7.2.4 and 7.3.1. (ii) follows by induction on the structure of \( t \), using the aforementioned equivariance properties along with the equivariance of composition (Lemma 7.1.9) and, for the suspension case, the internal permutation action (Lemma 7.1.6(i)).

(iii) follows easily by induction on the structure of \( t \). (iv) follows by another such induction, with the suspension case using Definition 7.1.3(ii) and the constructed term case using Lemma.

Definition 7.3.2 (c.f. (3.39) and Definition 3.2.7). Given a freshness environment \( \nabla \in \text{FE}_\Sigma \) defined as

\[ \nabla = [\tau_1 \not\equiv x_1 : s_1, \ldots, \tau_n \not\equiv x_n : s_n] \]  

(7.13)

and a \( \Sigma \)-structure \( M \) in a small FM-category \( C \) we define the \( C \)-object \( M[\nabla] \) (abbreviated to \([\nabla]\)) by

\[ [\nabla] \triangleq [s_1]^\nabla_1 \times \cdots \times [s_n]^\nabla_n . \]  

(7.14)

Given a term \( t \in \Sigma_\nabla(\nabla) \) we define the value arrow \([\nabla \vdash t : s] \) as a \( C \)-arrow \([\nabla] \to [s] \) by

\[ [\nabla \vdash \pi \ast x_i : s_i] \triangleq \pi_{[s_i]} \circ t_{[s_i]}^\nabla \circ \text{pr}_i \]

\[ [\nabla \vdash op \cdot t_1 \cdots t_n : s] \triangleq [op] \circ ([\nabla \vdash t_1 : s_1], \ldots, [\nabla \vdash t_n : s_n]) \]  

(7.15)

where \( op : [s_1, \ldots, s_n] \to s.\)

Definition 7.3.3 (c.f. Definition 3.4.7). A structure \( M \) in a small FM-category \( C \) satisfies the equality judgement \( \nabla \vdash t \approx t' : s \) if \( M[\nabla \vdash t : s] = M[\nabla \vdash t' : s] \). If \( M \) satisfies all axioms of a theory \( \mathbb{T} \) then it is a \( \mathbb{T} \)-algebra in \( C \).
7.1.6(iv) and Definitions 7.3.1 and 7.2.2. (v) also follows by such an induction, using (iv) and Definition 7.2.4(ii) in the constructed term case.

For (vi), take $\nabla$ as (7.13) and $\nabla'(x_i) = (\overline{\alpha}, \overline{\pi}_i, s_i)$ for $1 \leq i \leq n$. Then set

$$weak \triangleq \langle i^{\overline{\pi}_1} \circ pr_1, \ldots, i^{\overline{\pi}_n} \circ pr_n \rangle.$$  \hspace{1cm} (7.16)

The result then follows by induction on the structure of $t$, using Definition 7.2.4(ii) in the suspension case. (vii) is a corollary of (vi) by Definition 7.2.4(iii).

Definition 7.3.3 defined satisfaction in terms of equality judgements; by the results of Section 5.5 we know that these are as expressive as the three part judgements of Definition 3.4.3. Nonetheless, it will be convenient to discuss freshness judgements in the categorial setting also. To do so we will use the fresh-inclusions of Definition 7.2.4, and in particular the ‘Freshness’ condition (vi).

Suppose we have a freshness environment $\nabla \in FE_{\Sigma}$, term $t \in \Sigma_s(\nabla')$ and set of atoms $\overline{\alpha}$ such that $\overline{\alpha} \neq s$. What would it mean for a structure in a small FM-category to satisfy the judgement $\nabla \vdash \overline{\alpha} \neq t : s$? One answer might be to ask that $\overline{\alpha} \neq [\nabla \vdash t : s]$. However, this is too strong a requirement, as by the equivariance of the domain map (Lemma 7.1.9) this judgement requires that $\overline{\alpha} \neq [\nabla]$.

Rather, we must follow Lemma 5.5.1 and convert the freshness judgement into the equality judgement $\nabla^{\neq supp(\overline{\alpha})} \vdash (\overline{\alpha}, \overline{\pi}) \triangleright t \approx t : s$, where $\overline{\alpha}$ is an ordering of $\overline{\pi}$ and $\overline{\pi}$ is a tuple of the same size such that $supp(\overline{\alpha}) \neq (\nabla, \overline{\alpha}, t)$. So satisfaction asks that

$$[\nabla^{\neq supp(\overline{\pi})} \vdash (\overline{\alpha}, \overline{\pi}) \triangleright t : s] = [\nabla^{\neq supp(\overline{\pi})} \vdash t : s].$$ \hspace{1cm} (7.17)

By Lemma 7.3.4(iv) and (vii) this is equivalent to

$$(\overline{\alpha} \overline{\pi}) \circ [\nabla \vdash t : s] \circ i^{supp(\overline{\pi})} = [\nabla \vdash t : s] \circ i^{supp(\overline{\pi})}.$$ \hspace{1cm} (7.18)

**Definition 7.3.5.** If (7.18) holds then by Definition 7.2.4(vi) a unique arrow is induced. Call this arrow $[\nabla \vdash \overline{\alpha} \neq t : s]$:

$$[\nabla]^{\neq supp(\overline{\pi})} \xrightarrow{i} [\nabla]^{\nabla \vdash \overline{\alpha} \neq t : s} \xrightarrow{i^{supp(\overline{\pi})}} [\nabla \vdash t : s].$$ \hspace{1cm} (7.19)

**Lemma 7.3.6.** Given a freshness environment $\nabla = [\overline{\alpha}_1 \neq x_1 : s_1, \ldots, \overline{\alpha}_n \neq x_n : s_n]$,

(i) $id[\nabla] = \langle [\nabla \vdash \overline{\alpha}_1 \neq x_1 : s_1], \ldots, [\nabla \vdash \overline{\alpha}_n \neq x_n : s_n] \rangle$;

(ii) $\pi[\nabla] = \langle [\nabla^{\neq \pi} \vdash \overline{\alpha}_1 \neq x_1 : s_1], \ldots, [\nabla^{\neq \pi} \vdash \overline{\alpha}_n \neq x_n : s_n] \rangle$;

(iii) $\pi[\nabla] = \langle [\nabla \vdash \pi \cdot \overline{\alpha}_1 \neq \pi \cdot x_1 : \pi \cdot s_1], \ldots, [\nabla \vdash \pi \cdot \overline{\alpha}_n \neq \pi \cdot x_n : \pi \cdot s_n] \rangle$;
(iv) Take $\nabla_1, \nabla_2 \in FE_\Sigma$ with disjoint domains and say that $\nabla_j = \nabla$ for $j = 1$ or $2$. Then $pr_j([\nabla_1], [\nabla_2]) = \langle [\nabla_1 \cup \nabla_2 \vdash \sigma_1 \neq x_1 : s_1], \ldots, [\nabla_1 \cup \nabla_2 \vdash \sigma_n \neq x_n : s_n] \rangle$;

(v) Suppose we have a term $t \in \Sigma_\delta(\nabla)$ and substitution $\sigma \in \Sigma(\nabla, (\nabla'))$ for some other $\nabla' \in FE_\Sigma$. If the arrows $[\nabla' \vdash \sigma(1) : s_1], \ldots, [\nabla' \vdash \sigma(n) : s_n]$ are defined for $1 \leq i \leq n$ then $[\nabla' \vdash t\{\sigma\} : s]$ equals

$$[\nabla \vdash t : s] \circ \langle [\nabla' \vdash \sigma(1) : s_1], \ldots, [\nabla' \vdash \sigma(n) : s_n] \rangle.$$  

\textbf{Proof.} We will first prove that the arrows used in (i)-(iv) are well-defined according to Definition 7.3.5 and (7.18).

(i): $[\nabla \vdash \sigma_i \neq x_i : s_i]$ is defined if (7.18) holds for $t = x_i$; that is,

$$\langle \vec{a}_i, \vec{a}_i' \rangle \circ i_\vec{a}_i \circ pr_i \circ i_\text{supp}(\vec{a}_i') = i_{\vec{a}_i} \circ pr_i \circ i_\text{supp}(\vec{a}_i') \quad (7.20)$$

where $\vec{a}_i$ is an ordering of $\sigma_i$ and $\vec{a}_i'$ is a fresh tuple of the same size. This is proved as follows:

\begin{align*}
(\vec{a}_i, \vec{a}_i') \circ i_{\vec{a}_i} \circ pr_i \circ i_\text{supp}(\vec{a}_i') \\
= (\vec{a}_i, \vec{a}_i') \circ i_{\vec{a}_i} \circ i_\text{supp}(\vec{a}_i') \circ pr_i \quad \text{(Definition 7.2.4(iii))} \\
= i_{\vec{a}_i} \circ i_\text{supp}(\vec{a}_i') \circ (\vec{a}_i, \vec{a}_i')_t \circ pr_i \quad \text{(Lemma 7.1.6(iv) and Definition 7.2.4(i))} \\
= i_{\vec{a}_i} \circ i_\text{supp}(\vec{a}_i') \circ pr_i \\
= i_{\vec{a}_i} \circ pr_i \circ i_\text{supp}(\vec{a}_i') \quad \text{(Definition 7.2.4(iv)).}
\end{align*}

(ii): $[\nabla^\# \vdash \sigma_i \neq x_i : s_i]$ is defined by Lemma 7.3.4(vii) and (7.20).

(iii): Set $t = \pi \cdot x_i$ and $\vec{a}' = \pi \cdot \vec{a}_i$ in (7.18). Then $[\nabla \vdash \pi \cdot \sigma_i \neq \pi \cdot x_i : \pi \cdot s_i]$ is defined as a corollary of (7.20).

(iv): The existence of $[\nabla_1 \cup \nabla_2 \vdash \sigma_i \neq x_i : s_i]$ follows similarly as above.

The results (i), (ii) and (iv) then follow by applying $i_{\vec{a}_i} \circ pr_i$ to each side, as $i_{\vec{a}_i}$ is mono by Lemma 7.2.9 and projections are jointly mono. (iii) follows similarly by applying $i\pi_{\vec{a}_i} \circ pr_i$ to each side. Finally, (v) follows by Lemma 7.3.4(iii) and (v).

\textbf{Theorem 7.3.7} (Soundness Theorem, c.f. Theorem 5.1.5). If $M$ is $T$-algebra and $\nabla \vdash \pi \vdash t \approx t'$ : $s$ then $M$ satisfies that equality judgement.

\textbf{Proof.} We need to show closure under the proof rules for equality of Figure 5.3. (\textsc{refl}$_E$), (\textsc{symm}$_E$) and (\textsc{trans}$_E$) are trivial, while (\textsc{weak}$_E$) follows by Lemma 7.3.4(vi).

\textsc{subst}$_E$: If $\nabla$ is as (7.13) then the arrows $[\nabla' \vdash \sigma(1) : s_1]$ are defined for $1 \leq i \leq n$, so by Lemma 7.3.6(v)

$$[\nabla' \vdash t\{\sigma\} : s] = [\nabla \vdash t : s] \circ \langle [\nabla' \vdash \sigma(1) : s_1], \ldots \rangle \quad (7.21)$$

and similarly for $t'\{\sigma'\}$. We have $[\nabla \vdash t : s] = [\nabla \vdash t' : s]$, while $i_{\vec{a}_i} \circ [\nabla' \vdash \sigma(1) : s_1] = [\nabla' \vdash \sigma(x_i) : s_i] = [\nabla' \vdash \sigma'(x_i) : s_i] = i_{\vec{a}_i} \circ [\nabla' \vdash \sigma(x_i) : s_i]$. But $i_{\vec{a}_i}$ is mono by Lemma 7.2.8(ii), so we are done.
Definition 7.3.9 (c.f. Definition 6.1.2). A $\text{T}$-homomorphism $M \rightarrow M'$ in $\mathcal{C}$ is an equivariant function $h : \text{Sort}_\Sigma \rightarrow ar\mathcal{C}$, with $h_s : M[s] \rightarrow M'[s]$, such that

$$h_s \circ M[[op]] = M'[[op]] \circ (h_{s_1} \times \cdots \times h_{s_n})$$

(7.22)

for $op : [s_1, \ldots, s_n] \rightarrow s$.

Definition 7.3.9 (c.f. Definition 6.1.2). Given a NEL-theory $\text{T}$ and small FM-category $\mathcal{C}$, the category of $\text{T}$-algebras in $\mathcal{C}$, $(\text{T}, \mathcal{C})_{\text{Alg}}$, has as object $\text{T}$-algebras in $\mathcal{C}$ and as arrows $\text{T}$-homomorphisms between them.

Lemma 7.3.10. Given a freshness environment $\nabla = [\overline{a}_1 \neq x_1 : s_1, \ldots, \overline{a}_n \neq x_n : s_n]$ and a $\text{T}$-homomorphism $h : M \rightarrow M'$ in a small FM-category $\mathcal{C}$, then for all $t \in \Sigma_\nabla(\nabla)$,

$$h_s \circ M[[\nabla \vdash t : s]] = M'[[\nabla \vdash t : s]] \circ (h_{s_1}^{\overline{a}_1} \times \cdots \times h_{s_n}^{\overline{a}_n})$$

where $h_{s_i}^{\overline{a}_i}$ is as defined by Lemma 7.2.8.

Proof. The proof proceeds by induction on the structure of $t$. First consider $t = \pi x_i$. Then the result follows by

$$M[[s_1]]^{\pi_1} \times \cdots \times M[[s_n]]^{\pi_n} \xrightarrow{pr_i} M[[s_i]]^{\pi_i} \xrightarrow{i} M[[\pi \cdot s_i]] \xrightarrow{\pi} M[[\pi \cdot s_i]]$$

(7.23)

The leftmost square commutes by a standard property of projections, the middle by Lemma 7.2.8(i) and the rightmost by Lemma 7.1.6(iv) and the equivariance property of Definition 7.3.8. The constructed term case then follows by (7.22).

Lemma 7.3.11. Given FM-categories $\mathcal{C}, \mathcal{C}'$ and an algebra $M \in (\text{T}, \mathcal{C})_{\text{Alg}}$, we can define a functor, called the modelling functor, $M(-) : FM(\mathcal{C}, \mathcal{C}') \rightarrow (\text{T}, \mathcal{C}')_{\text{Alg}}$ by

(i) $M(F)[s] = F(M[s])$ and $M(F)[op] = F(M[op])$;

(ii) $M(\phi)_s = \phi_{M[s]}$. 

Proof. The maps \( s \mapsto M(F)[s] \) and \( op \mapsto M(F)[op] \) are equivariant by Lemma 7.1.10(i), so \( M(F) \) is a \( \Sigma \)-structure. Now for any \( t \in \Sigma_s(\nabla') \),

\[
M(F)[\nabla \vdash t : s] = F(M[\nabla \vdash t : s])
\]

(7.24)

by induction on the structure of \( t \) and the properties of Definition 7.2.12, so if \( M \in (\mathbb{T}, \mathbb{C}, \mathbb{A})_{\text{Alg}} \) then \( M(F) \in (\mathbb{T}, \mathbb{C}', \mathbb{A})_{\text{Alg}} \).

The map \( s \mapsto M(\phi)s \) is equivariant by Lemma 7.1.10(ii), while (7.22) holds because natural transformations between finite product preserving functors commute with those products. \( \square \)

### 7.4 The classifying category

Given a certain theory \( \mathbb{T} \), this section will define a small FM-category called the classifying category, \( Cl(\mathbb{T}) \). This construction will give rise to a simple completeness proof, as well as the correspondences of the next section.

**Definition 7.4.1.** Fix an ordering \( v_1, v_2, \ldots \) on the set of variables \( \text{Var} \) and let \( \mathbb{T} \) be a theory over a signature \( \Sigma \). Then the classifying category \( Cl(\mathbb{T}) \) is defined by

\[
\text{ob} Cl(\mathbb{T}) \triangleq \{ \nabla \in \text{FE}_\Sigma \mid \dom(\nabla) = \{ v_1, \ldots, v_n \} \text{ for some } n \}.
\]

(7.25)

Given \( \nabla, \nabla' \in \text{ob} Cl(\mathbb{T}) \) where

\[
\nabla = [\overline{a}_1 \# v_1 : s_1, \ldots, \overline{a}_n \# v_n : s_n]
\]

(7.26)

a \( Cl(\mathbb{T}) \)-arrow \( f : \nabla' \to \nabla \) is defined by

\[
\nabla' \vdash [\overline{a}_1 \# [t_1] : s_1, \ldots, \overline{a}_n \# [t_n] : s_n]
\]

(7.27)

where, for \( 1 \leq i \leq n \),

(i) \( t_i \in \Sigma_s((\nabla')^i) \);

(ii) \( \nabla' \vdash_\mathbb{T} \overline{a}_i \# t_i : s_i \);

(iii) \( [t_i] \) is the equivalence class of terms \( u \) such that \( \nabla' \vdash_\mathbb{T} t_i \approx u : s_i \).

We will follow Notation 3.4.4 and write \( \emptyset \# [t_1] : s_1 \) as \( [t_1] : s \) in a \( Cl(\mathbb{T}) \)-arrow (7.27).

**Lemma 7.4.2.** With the appropriate definitions \( Cl(\mathbb{T}) \) is a small FM-category.

**Proof.** First note that the equivalence classes of condition (iii) in Definition 7.4.1 are well defined with respect to the freshness judgement of (ii) by application of the rule \( (\text{trans}) \). We can then work through the definition of an FM-category requirement by requirement.
Identity: Given $\nabla \in ob \, Cl(\mathbb{T})$ as (7.26), the identity on $\nabla$ is

$$id_\nabla \triangleq \nabla \vdash [\vec{a}_1] \neq [v_1] : s_1, \ldots, [\vec{a}_n] \neq [v_n] : s_n \, .$$

(7.28)

$\nabla \vdash \vec{a}_i \neq v_i : s_i$ by (ATM-INTRO) and (WEAK).

Composition: Taking $f$ as (7.27) and $g = \nabla \vdash [\vec{a}_1] \neq [t'_1] : s'_1, \ldots, [\vec{a}_m] \neq [t'_m] : s'_m],$

$$g \circ f \triangleq \nabla' \vdash [\vec{a}_1] \neq [t'_1] \cdot [\sigma] : s'_1, \ldots, [\vec{a}_m] \neq [t'_m] \cdot [\sigma] : s'_m]$$

(7.29)

where $\sigma(v_i) = t_i$ for $1 \leq i \leq n$. $\nabla' \vdash \vec{a}_i \neq t'_i \cdot [\sigma] : s'_i$, and the substitutions are well defined on the equivalence classes, by (SUBST). The composition and identity laws hold by Lemma 3.3.4, so $Cl(\mathbb{T})$ is indeed a category.

Finitely supported internal permutation action: Given $\nabla$ as above,

$$\pi_\nabla \triangleq \nabla \vdash [\pi \cdot \vec{a}_1] \neq [\pi \cdot v_1] : \pi \cdot s_1, \ldots, [\pi \cdot \vec{a}_n] \neq [\pi \cdot v_n] : \pi \cdot s_n \, .$$

(7.30)

$\nabla \vdash \pi \cdot \vec{a}_i \neq \pi \cdot v_i : \pi \cdot s_i$ by ($\#\text{-EQUIVAR}$) and (WEAK). The properties of Definition 7.1.3 clearly hold. $\pi \cdot \nabla$, the codomain of $\pi_\nabla$, is therefore defined as (3.34). Taking $f$ as above and applying (7.3),

$$\pi \cdot f = \pi \cdot \nabla' \vdash [\pi \cdot \vec{a}_1] \neq [\pi \cdot t_1] : \pi \cdot s_1, \ldots, [\pi \cdot \vec{a}_n] \neq [\pi \cdot t_n] : \pi \cdot s_n$$

(7.31)

by Lemma 3.3.6. or $Cl(\mathbb{T})$ is clearly finitely supported under this action.

Equivariant finite products: The terminal object in $Cl(\mathbb{T})$ is the empty freshness environment.

Given $\nabla$ as above and $\nabla' = [\vec{a}_1] \neq v_1 : s'_1, \ldots, [\vec{a}_m] \neq v_m : s_m]$, their product $\nabla \times \nabla'$ is defined by

$$[\vec{a}_1] \neq v_1 : s_1, \ldots, [\vec{a}_n] \neq v_n : s_n, [\vec{a}_1] \neq v_{n+1} : s'_1, \ldots, [\vec{a}_m] \neq v_{n+m} : s'_m]$$

(7.32)

with projections

$$pr_1 \triangleq \nabla \times \nabla' \vdash [\vec{a}_1] \neq [v_1] : s_1, \ldots, [\vec{a}_n] \neq [v_n] : s_n]$$

$$pr_2 \triangleq \nabla \times \nabla' \vdash [\vec{a}_1] \neq [v_{n+1}] : s'_1, \ldots, [\vec{a}_m] \neq [v_{n+m}] : s'_m]$$

(7.33)

It is clear that this indeed defines a product in $Cl(\mathbb{T})$, and further that this product is equivariant.

Fresh-inclusions: Take $\nabla^{\#\pi}$ as in (5.4). Then

$$i_\nabla^{\#\pi} \triangleq \nabla^{\#\pi} \vdash [\vec{a}_1] \neq [v_1] : s_1, \ldots, [\vec{a}_n] \neq [v_n] : s_n]$$

(7.34)

Definition 7.2.4(i)–(iii) are trivial. (iv) asks that $\pi_\nabla^{\#\text{supp}(\pi)} \equiv id_{\nabla^{\#\text{supp}(\pi)}}$; this follows because $\nabla^{\#\text{supp}(\pi)} \vdash \pi \cdot v_i \approx v_i : s$ by (SUSP). (v) follows by (ATM-ELIM). For (vi), consider (7.10) where $C = \nabla'$, $D = \nabla$ and $f$ is as above. Then the unique arrow $\nabla' \to \nabla^{\#\pi}$ can only be defined as

$$\hat{f} = \nabla' \vdash [\vec{a}_1 \cup \pi \cdot \vec{a}] \neq [t_1] : s_1, \ldots, [\vec{a}_n \cup \pi \cdot \vec{a}] \neq [t_n] : s_n]$$

(7.35)

We therefore need to prove that $\nabla' \vdash \pi \cdot \vec{a} \neq t_i : s_i$ for $1 \leq i \leq n$. Applying Lemma 5.2.1 to $f$ gives us $\nabla' \vdash \pi \cdot \vec{a} \neq \pi \cdot t_i : s_i$. The top line of (7.10) commuting tells us that $(\nabla')^{\#\pi} \vdash \pi \cdot \vec{a} \neq \pi \cdot t_i : s_i$. But $\vec{a} \neq (\pi \cdot \vec{a}, f)$, so we can apply (ATM-ELIM) to get our result. \qed
\textbf{Definition 7.4.3.} Define the \textit{generic algebra} $G$ by

\begin{align*}
G[s] & \triangleq [v_1 : s] \\
G[op] & \triangleq [v_1 : s_1, \ldots, v_n : s_n] \vdash [[op \, v_1 \, \cdots \, v_n] : s]
\end{align*}

for $op : [s_1, \ldots, s_n] \rightarrow s$.

\textbf{Lemma 7.4.4.} The generic algebra $G$ is a $\mathbb{T}$-algebra in $\text{Cl}(\mathbb{T})$.

\textbf{Proof.} Let $\nabla = [\overline{a}_1 \# x_1 : s_1, \ldots, \overline{a}_1 \# x_n : s_n]$ be a freshness environment in $\text{FE}_\Sigma$. Let $\sigma$ be the substitution mapping $x_i \mapsto v_i$ for $1 \leq i \leq n$, and $\nabla\{\sigma\}$ be the matching $\text{Cl}(\mathbb{T})$-object $[\overline{a}_1 \# v_1 : s_1, \ldots, \overline{a}_1 \# v_n : s_n]$. Then $G[\nabla] = \nabla\{\sigma\}$ and, given $t \in \Sigma(\nabla)$,

\begin{equation}
G[\nabla \vdash t : s] = \nabla\{\sigma\} \vdash [[t\{\sigma\}] : s] \tag{7.36}
\end{equation}

by a routine induction on the structure of $t$.

Now if $\nabla \vdash t \approx t' : s$ is a $\mathbb{T}$-axiom then by (7.36) we must show that $\nabla\{\sigma\} \vdash [[t\{\sigma\}] : s] = \nabla\{\sigma\} \vdash [[t'\{\sigma\}] : s]$. That is, $\nabla\{\sigma\} \vdash t\{\sigma\} \approx t'\{\sigma\} : s$, which follows by an application of (\text{SUBST}). \hfill \square

\textbf{Theorem 7.4.5} (Completeness Theorem, c.f. Theorem 5.4.7). \textit{If} $\nabla \vdash t \approx t' : s$ \textit{is satisfied by any $\mathbb{T}$-model in any small FM-category then} $\nabla \vdash_{\mathbb{T}} t \approx t' : s$.

\textbf{Proof.} $\nabla \vdash t \approx t' : s$ is satisfied by the generic model $G$ in $\text{Cl}(\mathbb{T})$, so by (7.36) $\nabla\{\sigma\} \vdash [[t\{\sigma\}] : s] = \nabla\{\sigma\} \vdash [[t'\{\sigma\}] : s]$, so $\nabla \vdash_{\mathbb{T}} t \approx t' : s$ by the definition of $\text{Cl}(\mathbb{T})$. \hfill \square

### 7.5 The category-theory correspondence

The final section of this chapter will demonstrate two key relationships. First, there is an isomorphism between $\mathbb{T}$-algebras in a small FM-category $\mathcal{C}$ and FM-functors from the classifying category of $\mathbb{T}$ into $\mathcal{C}$. Second, there is a correspondence between NEL-theories and small FM-categories.

\textbf{Theorem 7.5.1.} \textit{Given a small FM-category $\mathcal{C}$ and the generic algebra $G$ of Definition 7.4.3, the modelling functor of Lemma 7.3.11, $G(-)$, defines an isomorphism}

$$
\text{FM}(\text{Cl}(\mathbb{T}), \mathcal{C}) \cong (\mathbb{T}, \mathcal{C})_{\text{Alg}}
$$

\textit{from the category of FM-functors $\text{Cl}(\mathbb{T}) \rightarrow \mathcal{C}$ (Definition 7.2.12) to the category of $\mathbb{T}$-algebras in $\mathcal{C}$ (Definition 7.3.9).}
Finally, given a natural transformation \( G^{-1}(\_): (\mathbb{T}, \mathbb{C})_{Alg} \to FM(Cl(\mathbb{T}), \mathbb{C}) \) by:

\[
\begin{align*}
G^{-1}(M)(\nabla) &= M[\nabla] \\
G^{-1}(M)(f) &= (M[\nabla' \vdash a_1 \# t_1 : s_1], \ldots, M[\nabla' \vdash a_n \# t_n : s_n]) \\
G^{-1}(h)_\nabla &= h_{\nabla_1} \times \cdots \times h_{\nabla_n}
\end{align*}
\]

for \( \nabla \) as in (7.26) and \( f: \nabla' \to \nabla \) as in (7.27).

\( G^{-1}(\_\nabla) \) preserves identities and compositions by Lemma 7.2.8(ii), so is indeed a functor. For any \( \mathbb{T} \)-algebra \( M \) in \( \mathbb{C} \), \( G^{-1}(M) \) is an FM-functor \( Cl(\mathbb{T}) \to \mathbb{C} \) by the results of Lemma 7.3.6.

Now if we have a \( \mathbb{T} \)-homomorphism \( h: M \to M' \), we need to show that \( G^{-1}(h) \) is a natural transformation \( G^{-1}(M) \to G^{-1}(M') \). Take \( f: \nabla' \to \nabla \) as in (7.27). Then, for all \( 1 \leq i \leq n \), \( i_\nabla \circ pr_i \circ G^{-1}(M')(f) \circ G^{-1}(h)_\nabla = M[\nabla' \vdash t_i : s_i] \circ G^{-1}(h)_\nabla \) by Definition 7.3.5. This is \( h_{t_i} \circ M[\nabla' \vdash t_i : s_i] \) by Lemma 7.3.10, which is \( h_{t_i} \circ i_\nabla \circ M[\nabla' \vdash a_i \# t_i : s_i] = i_\nabla \circ h_{a_i} \circ M[\nabla' \vdash a_i \# t_i : s_i] \) by Lemma 7.2.8(i). This is \( i_\nabla \circ pr_i \circ G^{-1}(h)_\nabla \circ G^{-1}(M)(f) \), but \( i_\nabla \) is mono by Lemma 7.2.9, and the projections are jointly mono, so \( G^{-1}(M')(f) \circ G^{-1}(h)_\nabla = G^{-1}(h)_\nabla \circ G^{-1}(M)(f) \).

\( GG^{-1} \) is the identity on \( (\mathbb{T}, \mathbb{C})_{Alg} \) by working through the relevant definitions; the converse requires more work. Given an FM-functor \( F: Cl(\mathbb{T}) \to \mathbb{C} \) and \( Cl(\mathbb{T}) \)-object \( \nabla \) we have \( G^{-1}(G(F))_\nabla = G(F)_\nabla = F(G[\nabla]) \) because \( F \) preserves finite products, and this equals \( F\nabla \). Given a \( Cl(\mathbb{T}) \)-arrow \( f: \nabla' \to \nabla \), we will again apply the jointly mono \( i_\nabla \circ pr_i :\)

\[
\begin{align*}
i_\nabla \circ pr_i \circ G^{-1}(G(F))(f) &= i_\nabla \circ (G(F)[\nabla' \vdash a_1 \# t_1 : s_1], \ldots) \\
&= i_\nabla \circ (G(F)[\nabla' \vdash t_i : s_i]) \quad \text{by (7.19)} \\
&= F(G[\nabla' \vdash t_i : s_i]) \quad \text{by (7.24)} \\
&= F(\nabla \vdash [t_i : s_i]) \quad \text{by (7.36)}
\end{align*}
\]

Conversely, \( i_\nabla \circ pr_i \circ f = F(\nabla \vdash [t_i : s_i]) \) because \( F \) preserves finite products (so in particular, commutes with the arrow \( f = (\nabla' \vdash [a_1 \# [t_i : s_i], \ldots, \nabla' \vdash [a_n \# [t_n : s_n]]) \) and fresh-inclusions.

Finally, given a natural transformation \( \phi: F \to F' \) in \( FM(Cl(\mathbb{T}), \mathbb{C}) \), we must show that \( G^{-1}(G(\phi))_\nabla = \phi_\nabla \) for all \( Cl(\mathbb{T}) \)-objects \( \nabla \). Applying \( i_\nabla \circ pr_i \) as above we have \( i_\nabla \circ pr_i \circ G^{-1}(G(\phi))_\nabla = \phi_{[t_i:s_i]} \circ pr_i \). This equals \( i_\nabla \circ pr_i \circ \phi_\nabla \) by the naturality of \( \phi \) along with the fact that \( F \) preserves finite products and fresh-inclusions.

\[\square\]

**Definition 7.5.2.** Given a small FM-category \( \mathbb{C} \), we define the signature \( Sg(\mathbb{C}) \) by

\[
\begin{align*}
\text{Sort}_{Sg(\mathbb{C})} &\triangleq ob \mathbb{C} \\
\text{Op}_{Sg(\mathbb{C})} &\triangleq \{ f: [C_1, \ldots, C_n] \to C \mid f: C_1 \times \cdots \times C_n \to C \in ar \mathbb{C} \}
\end{align*}
\]

with permutation actions defined by the internal permutation on \( \mathbb{C} \) in the normal way. Note that one arrow can give rise to multiple operation symbols. For example, \( f: C_1 \times C_2 \to C \) induces operation symbols \([C_1 \times C_2] \to C \) and \([C_1, C_2] \to C \).
Let $M(C)$ be the $Sg(C)$-structure in $C$ which we define by $M(C)[C] = C$ and $M(C)[f] = f$. Then define $Th(C)$ as the $Sg(C)$-theory whose axioms are all the judgements that are satisfied by $M(C)$, so $M(C)$ is trivially an algebra in $(Th(C), C)_{Alg}$.

We will use the above definition, along with the notion of classifying category from Definition 7.4.1, to define a correspondence between small FM-categories and NEL-theories.

**Theorem 7.5.3.** For any small FM-category $C$ there is an equivalence

$$C \simeq Cl(Th(C)).$$

**Proof.** The equivalence functor is $G^{-1}(M(C)) : Cl(Th(C)) \to C$, as defined by (7.37) and Definition 7.5.2. The proof proceeds by showing that this functor is full, faithful and essentially surjective. Unpacking the definition explicitly, given a $Cl(Th(C))$-object $\nabla = [\overline{a}_1 \neq v_1 : C_1, \ldots, \overline{a}_n \neq v_n : C_n]$ and $Cl(Th(C))$-arrow $f = \nabla' : [\overline{a}_1 \neq [t_1] : C_1, \ldots, \overline{a}_n \neq [t_n] : C_n]$,

$$G^{-1}(M(C))(\nabla) = C_1^{\overline{a}_1} \times \cdots \times C_n^{\overline{a}_n};$$

$$G^{-1}(M(C))(f) = \{M(C)[\forall \overline{a}_1 \neq t_1 : C_1], \ldots\}. \quad (7.38)$$

$G^{-1}(M(C))$ is full: take any $\nabla, \nabla' \in \text{ob } Cl(T(C))$ and $C$-arrow $f : G^{-1}(M(C))(\nabla) \to G^{-1}(M(C))(\nabla')$. Then $[v_1 : G^{-1}(M(C))(\nabla)] \vdash [[f v_1] : G^{-1}(M(C))(\nabla')]$ is a $Cl(Th(C))$-arrow. Applying the functor $G^{-1}(M(C))$ to this arrow gives us $M(C)[\forall v_1 : G^{-1}(M(C)) \vdash f v_1 : G^{-1}(M(C))(\nabla')] = M(C)[f] = f$.

$G^{-1}(M(C))$ is faithful: say we have $f : \nabla' \to \nabla$ as above and $f' : \nabla' \to \nabla$ defined by $\nabla' \vdash [\overline{a}_1 \neq [t'_1] : C_1, \ldots, \overline{a}_n \neq [t'_n] : C_n]$, and that $G^{-1}(M(C))(f) = G^{-1}(M(C))(f')$. We wish to prove that $f = f'$, so $\nabla' \vdash_{Th(C)} t_i \approx t'_i : C_i$ for $1 \leq i \leq n$. This holds if $M(C)[\forall \overline{a}_1 \neq t_i] = M(C)[\forall \overline{a}_1 \neq t'_i]$. But if $G^{-1}(M(C))(f) = G^{-1}(M(C))(f')$ then applying $i^\overline{a}_1 \circ pr_i$ to both sides, noting the explicit definition of (7.38), gives us that result exactly.

$G^{-1}(M(C))$ is surjective, and hence essentially surjective: for any $C \in \text{ob } C$, $G^{-1}(M(C))[v_1 : C] = C$. \hfill $\square$

**Remark 7.5.4.** Theorem 7.5.3 has, finally, justified the generalisation of the signatures of [10] introduced back in Definition 3.1.1, making the set of sorts into a nominal set. In order to correctly represent the freshness environments of NEL as objects in a category it was necessary to have a non-trivial notion of permutation action on the objects of this category, as freshness environments contain atoms that may be permuted. But if the sorts of NEL all had empty support then an equivariant interpretation into categorial algebra could only send them to the emptyly supported members of an FM-category. We would therefore lose the full equivalence that was proved above. Section 8.7 will go on to discuss some practical advantages that this generalisation might offer for a certain extension of NEL.

We wish to prove the converse of Theorem 7.5.3, but to do this we need the correct notion of a homomorphism of theories. It turns out that the most reasonable way to define this is in terms of their classifying categories:
Definition 7.5.5. Given NEL-theories $\mathbb{T}, \mathbb{T}'$, a translation of NEL-theories $T : \mathbb{T} \to \mathbb{T}'$ is an FM-functor $Cl(\mathbb{T}) \to Cl(\mathbb{T}')$.

The converse to Theorem 7.5.3, that for any $\mathbb{T}$ we have an equivalence $\mathbb{T} \simeq Th(Cl(\mathbb{T}))$, is therefore equivalent to asking that $Cl(\mathbb{T}) \simeq Cl(Th(Cl(\mathbb{T})))$. This is therefore an immediate corollary of Theorem 7.5.3, so we have our final correspondence:

Theorem 7.5.6. The equivalences $\mathcal{C} \simeq Cl(Th(\mathcal{C}))$ and $\mathbb{T} \simeq Th(Cl(\mathbb{T}))$ define a correspondence between NEL-theories and small FM-categories.
Chapter 8

Conclusions

This chapter will summarise the contributions made by the dissertation and outline and discuss areas of related and potential further work.

8.1 Summary of contributions

This dissertation has presented Nominal Equational Logic (NEL), a logic for judgements of equality between terms, and the freshness of names for terms, modulated by side conditions concerning the freshness of names for variables that appear in those terms. Judgements of this type are ubiquitous in logic, and particularly in computer science. The intended interpretation of these terms and judgements is in the universe of FM-sets. Where possible, NEL was developed by analogy with equational logic, allowing algebraic accounts of the logic to be developed.

Chapter 3 introduced the syntax and semantics of NEL. In keeping with the desire to keep NEL as close to equational logic as possible, a simple notion of signature was used, differing from equational logic only in requiring that the sorts and operations symbols of a signature form nominal sets, and that the typing function be equivariant. The notion of term was also kept as close as possible to equational logic, with the exception that permutations can be ‘suspended’ over a variable. The sorting environments of equational logic were extended to freshness environments guaranteeing that certain atoms were fresh for certain variables, and judgements were extended to include assertions of the freshness of names for terms. The semantics for NEL was given in the category $\mathcal{F}_M$-Set of FM-sets, in which the suspensions, freshness environments and freshness judgements can be interpreted. The notion of an algebra for a NEL-theory was hence defined.

Chapter 4 looked at the issue of binding, asking if the very simple notion of NEL-signature was sufficiently expressive to capture the crucial notion of an operation symbol that binds names in its arguments. This was proved by extending NEL to ‘NEL with binders’, explicitly supporting binding patterns and interpreting them in $\mathcal{F}_M$-Set via the established notion of atom abstraction. It was then shown that this extension of NEL does not in fact add expressivity, as theories
with binders can be translated into theories without binders by adding certain binding axioms. An extended example of this translation was given for nominal substitution, where atoms are substituted for atoms in a term. It was then shown that NEL was expressive enough to handle certain binding structures that exist in the literature. Finally, the issue of concretion, the converse in $\mathcal{FM}$-Set to atom-abstraction, was discussed. It was shown that the partial nature of concretion makes it unsuitable for axiomatisation in NEL, but that it can be axiomatised for those FM-sets that support total concretion.

Chapter 5 returned to the main thrust of the dissertation to offer proof rules for NEL. The proof that these rules are sound followed by the application of standard results concerning FM-sets, but the completeness proof required more work. The cause of this difficulty lay in the difference between variables and constants, which in equational logic can essentially be treated the same. In NEL, however, constants have a known finite support, while variables will be interpreted as elements that may have any finite support, bar a finite number of atoms asserted not to be in that support. The ground term case therefore follows as with equational logic. This then supplied the base case of an induction proof that involved replacing variables with constants whose support is ‘large enough’, covering all the atoms of a judgement apart from those asserted to be fresh for the original variable. The proof, while conceptually simple, involved quite subtle manipulation of terms and judgements, including an operation to convert the new constants back into suspensions. The final section of the chapter then proved that, given the presence of freshness environments, NEL is indeed equational, as freshness judgements can be converted into equality judgements. Proof rules were given for this equality-only NEL, and their equivalence to the standard proof rules was proved.

Chapter 6 developed an account of universal algebra for NEL. Given a NEL-theory the category of algebras for that theory was described, and initial and free algebras were found. A monadicity result then followed, showing that the category of algebras for a theory is isomorphic to the Eilenberg-Moore category of algebras for the monad generated by that theory. These results and their proofs were surprisingly close to the standard account of universal algebra for equational theories.

Finally Chapter 7 gave an account of Nominal Lawvere Theories, culminating in a correspondence between NEL-theories and categories with certain structure, called small FM-categories. The first structural requirement was the existence of internal permutation actions, which are arrows of the category that allow a permutation action on its objects and arrows to be defined. Second, the category needed finite products (as is standard in Lawvere’s account) that are equivariant with respect to the internal permutation action. Third, the category needed fresh-inclusions. These arrows captured the notion of creating the subset of an FM-set of elements whose support is disjoint from some finite set of atoms. This third condition was the most complex, but it was crucial as it allowed freshness environments, and freshness in general, to be correctly represented. Having defined the notion of FM-category, an account of semantics in an FM-category was given generalising that of earlier chapters, and it was shown the proof rules of NEL were sound for this account. The notion of classifying category was defined, and a
completeness proof followed. Finally, two key correspondences were given. First, that algebras in a small FM-category are equivalent to structure preserving functors from the classifying category to that category. Following on from that, the final result of the dissertation demonstrated a correspondence between NEL-theories and small FM-categories.

8.2 Related work – Nominal Algebra

The most closely related work to that presented in this dissertation is that of Nominal Algebra (NA) [23, 38], an independently produced account of equational logic in the context of nominal sets. This section will discuss the differences between these accounts, but it should be noted that their similarities are also important, and indeed the literature on NA has influenced the presentation of some of the results of this dissertation, such as the use of the phrases ‘object-level’ and ‘meta-level’ to describe the permutation actions on terms. As such, these two strands of research should continue to inform each other.

There are three key differences between NEL and NA. The first is that NA uses the nominal signatures of [61], with built-in support for binding and other structure. Example 4.4.2 demonstrated that such structure does not add expressivity. As discussed in the introduction of Chapter 4, such structure may have practical benefit in applying nominal techniques, but in developing theoretical and algebraic results it is a hindrance. This dissertation contains dozens of proofs by induction of the structure of a term, some of them technical and complex. Complicating the structure of a term any more than absolutely necessary is therefore to be avoided.

The second difference is that NEL uses nominal sets of sorts and operation symbols, while NA uses only sets. In the case of sorts the extension to nominal sets is new to this dissertation. This generalisation may have produced marginally more work throughout this dissertation as we had to concern ourselves with the action of permutations on sorts as well as terms, but it paid dividends in the categorial results of Chapter 7, as discussed in Remark 7.5.4. Further, Section 8.7 will argue that this generalisation may prove necessary for at least one commonsense extension of NEL.

The third difference, and the one that has received the most discussion in the literature, is that NA is not complete for freshness. The deduction

\[ \nabla \vdash \pi \not\# t : s \quad \nabla \vdash t \approx t' : s \]

\[ \nabla \vdash \pi \not\# t' : s \]  \hspace{1cm} (8.1)

is not valid in NA, while it follows by the application of the rule (\textsc{trans}) in NEL. This is because NA treats freshness as a decidable syntactic side condition, while NEL treats it as a first-class element of the logic. In [38, Section 2.6.1] Mathijssen claims that this can cause ‘incorrect’ derivations to be made in NEL, giving the example from the \(\lambda\)-calculus

\[ \lambda a.((\lambda a.b)a) \approx_\eta (\lambda a.b)a \]  \hspace{1cm} (8.2)
which follows because \( a \) is fresh for \( b \), which is \( \beta \)-equivalent to the subterm \( (\lambda a. b)a \). But if this is an incorrect derivation, then NA fares no better:

\[
\lambda a.((\lambda a. b)a) \approx_\beta \lambda a. (ba) \approx_\eta b \approx_\beta (\lambda a. b)a.
\] (8.3)

Neither NEL nor NA make any distinctions between different types of equality; both equate the \( \lambda \)-terms \( \lambda a.((\lambda a. b)a) \) and \( (\lambda a. b)a \). As such, it is misleading to call the derivation in one system correct and the other incorrect, as both produce the same (correct) equality. Indeed, in the light of the results of Section 5.5 which show that freshness can be expressed in terms of equality, this difference between the models amounts to different design choices, rather than any deeper distinction. NA’s choice may prove appropriate for certain practical applications, while NEL’s choice is clearly the appropriate one for approaching freshness as a first-class subject of study.

### 8.3 Related work – equational systems

Nominal Equational Logic and its universal algebra have been shown to be a special case of the more general categorial construction of an Equational System [17, Section 7.3]. While this result was proved for the earlier version of NEL in [10], the generalisation of the set of sorts to a nominal set in this dissertation could also be accommodated within the framework.

Briefly, an Equational System in a category \( \mathcal{C} \) involves two endofunctors, \( \Sigma, \Gamma : \mathcal{C} \to \mathcal{C} \), known as the functorial signature and functorial context respectively. The categories \( \Sigma_{\text{Alg}}, \Gamma_{\text{Alg}} \) are the categories for algebras for that functor, and a functorial term is a functor \( \Sigma_{\text{Alg}} \to \Gamma_{\text{Alg}} \). An equational system is then defined by the quintuple \( (\mathcal{C}, \Sigma, \Gamma, L, R) \) where \( L, R \) are functorial terms - intuitively, the left and right sides of an equation. The category of algebras for this system is then simply the equaliser of \( L, R : \Sigma_{\text{Alg}} \to \Gamma_{\text{Alg}} \).

The advantage of this presentation is that important properties of an equational system, such as the existence of free algebras, can be be proved via the properties of the category and the functors on it in a very general way. In the case of NEL, while many of the algebraic properties in question followed fairly easily by more conventional means in Chapter 6, the proof [17, Corollary 7.5] that the category of algebras for a NEL theory is cocomplete is a genuine advance that has been made thanks to this treatment.

This work was extended by the more concrete constructions of Term Equational Systems and Logics [16], which in an enriched setting can give rise to sound and complete proof rules for an equational logic. This was used to produce Synthetic Nominal Equational Logic (SNEL), which is strongly related to both Nominal Algebra and the earlier version of NEL. The main difference in the presentation of SNEL is that terms and judgements exist within an \( \text{atom context} \) as well as a freshness context, so that, for example, a generalised term of SNEL is a pair \( (\overline{a}, t) \) where \( \text{supp}(t) \subseteq \overline{a} \).
8.4 Related and further work – nominal rewriting

This dissertation has been occupied with the theoretical aspects of extending equational logic to deal with names and freshness. However, there is also the question of whether there exists a decision procedure for NEL, and what properties such a procedure might have. The most promising approach to this are term rewriting systems [14], which are algorithms based upon directed equations. Indeed, there exists a notion of nominal term rewriting [15], which is based upon terms over nominal signatures with explicit support for binding. No relationship has yet been demonstrated between this work and either NEL or Nominal Algebra; it is an interesting open question whether such a relationship could be demonstrated, and whether nominal rewriting can be efficiently defined in terms of the simpler notion of signature used in this dissertation.

8.5 Further work – universal and categorial algebra

Chapters 6 and 7 introduced the basic concepts and properties of nominal universal algebra and Lawvere theories, but there are a wealth of results in the literature for universal and categorial algebra for which we have not yet attempted to provide nominal analogues. For example, are there nominal versions of the correspondence between monads induced by equational theories and monads of finite rank [51], or Birkhoff’s HSP theorem concerning the closure properties of equationally defined classes of algebra [6] (for which an analogue has been demonstrated [21] for Nominal Algebra), or the identification of most general unifiers of terms with coequalisers in the classifying category [52]?

Focusing particularly on the categorial results of Chapter 7, the generalisation of NEL from FM-sets to FM-categories opens the way for nominal equational logic in other settings, such as FM-cppos [55]. At this stage no examples are known of FM-categories that are not defined by adding extra structure to $\mathcal{F}M$-Set, and the discovery of such categories would be desirable. Finally, the categorial presentation of NEL may allow us to perform certain constructions on NEL-theories that may be hard to define directly, such as sums and tensor products [30].

As noted in Section 8.3, Equational Systems provide a general categorial context for nominal universal algebra, but at this stage no such result is known for nominal Lawvere theories. For example, there is no obvious relationship between nominal Lawvere theories and present work on generalised Lawvere theories such as [49, 50]. Putting the results of Chapter 7 into a more general framework would be desirable.

8.6 Further work – partial equational logic

Nominal Equational Logic makes no claim, of course, to be expressive enough to handle every syntactic system that a computer scientist or logician might want to axiomatise. As such, exten-
8.7 Further work – order-sorted algebra

In Example 3.1.3 a signature was defined for the simply typed λ-calculus typed à la Church. But this system can also be typed à la Curry [3, Section 3.1]. Here we have a partial function with finite domain mapping atoms to types, known as a typing context. The typing rules are then given by Figure 8.1.

The sorts of a NEL-signature are defined to be a nominal set, but although the results of Chapter 7 used this to our advantage, as discussed in Remark 7.5.4, we have yet to see practical examples of its usefulness. But perhaps here is an example; there is an obvious permutation action on the set of typing contexts, where if \( T(a) = s \) then \( \pi \cdot T(a) = s \). The sort of each term could therefore be a pair \((T, s)\) where \( T \) is a typing context, \( s \) is a type, and \( \pi \cdot (T, s) = (\pi \cdot T, s) \).

Operation symbols are then defined according to Figure 8.1. For example, \( V_a^T : (T(T(a))) \) is defined where \( a \in \text{dom}(T) \).

Unfortunately this does not quite work in practice. Consider \( \eta \)-equivalence:

\[
\begin{align*}
a \not\in \text{dom}(T) & \\
T \vdash t : s & \Rightarrow \text{ s' } T \vdash t' : \text{ s' } \\
T \cup \{ a \mapsto \text{ s} \} & \vdash t : \text{ s' } \\
T \vdash L_a t : s & \Rightarrow \text{ s' }
\end{align*}
\]

Figure 8.1: The Simply Typed λ-calculus typed à la Curry

sions to NEL might be desired in all manner of directions where that extra expressivity is what is required to deal with a given system. Nonetheless, there are areas where extensions to NEL would seem particularly apposite given the nature of names and binding. One of these is partial equational logic.

Section 4.5 showed that NEL was not expressive enough to axiomatise concretion, except for the case where a total concretion existed on an FM-set. For the more general case, we would need the ability to stipulate that the term \( \text{con}_a x \) can only be formed where \( a \not\approx x \). That is, \( \text{con}_a \) should be a partial operation symbol. The fresh-inclusions of Section 7.2 are another example of such partiality, as \( i_{\pi C} \) is defined only where \( \pi \not\approx C \). Both these examples seem to call for operation symbols that are partial modulo a freshness context. In order to permit such an extension to NEL, existing approaches to partial equational logic and algebra [7] may prove fruitful.
The question is this: is $a \in \text{dom}(T)$? If the freshness environment $a \not\# x : (T, s \Rightarrow s')$ is to be well-defined according to Definition 3.4.1 then it must be the case that $a \not\# (T, s \Rightarrow s')$, and so $a \notin \text{dom}(T)$. But the formation of the term $A x V_a$ tells us that $x$ and $T_a$ have the same typing context, so $a \in \text{dom}(T)$ as otherwise $V_a$ would not be defined, so we have a contradiction.

A natural answer to this problem would be to allow the typing contexts to be weakened, in the sense that if $x$ has sort $(T, s \Rightarrow s')$ where $a \notin \text{dom}(T)$ then it also has sort $(T \cup \{a \mapsto s\}, s \Rightarrow s')$, as the former is a subsort of the latter. The area of order-sorted algebra [26] provides an approach to extending equational logic in this manner, with a partial order defined on the sorts of a signature. In the FM-sets model this partial order should be interpreted as the finitely supported subset relation of Example 2.2.8.

This provides a practical argument, alongside the theoretical advances of Chapter 7, for developing the results of this dissertation with the sorts of a signature forming a nominal set, as such generality would be needed in order to make this extension to Nominal Equational Logic.
Bibliography


