EQUIVARIANT ZFA AND THE FOUNDATIONS OF NOMINAL TECHNIQUES

MURDOCH GABBAY

ABSTRACT. We give an accessible presentation to the foundations of nominal techniques, lying between Zermelo-Fraenkel set theory and Fraenkel-Mostowski set theory, and which has several nice properties including being consistent with the Axiom of Choice. We give two presentations of equivariance, accompanied by detailed yet user-friendly discussions of its theory and application.

CONTENTS

Key words and phrases: Nominal techniques, equivariance, Zermelo-Fraenkel set theory with atoms (ZFA), names and binding.

LOGICAL METHODS
IN COMPUTER SCIENCE
DOI:10.2168/LMCS-???

1. Introduction

Nominal techniques are based on positing the existence of a set of *atoms* $a, b, c, \ldots \in \mathbb{A}$. These are atomic elements which can be compared for equality but which have few if any other properties. The applications of this deceptively simple idea, are collectively called *nominal techniques*, and they are surprisingly rich, varied, and numerous. We list some of them in Subsection 1.1; just enough to give the reader some flavour of the scope of this field.

Nominal techniques are a success story in the fruitful interaction of logical foundations, mathematics, and computing. Yet precisely this interaction means that even experienced readers sometimes struggle to understand what is going on: what does it mean when we write 'assume a set of atoms', and what does this assumption really buy us?

We aim to clarify such questions for three types of reader:

- Readers who may have seen nominal techniques in action but have not given much thought to the subtleties involved in making their foundation precise, and who might appreciate an exposition.
- The fellow writer of a paper using nominal techniques, looking for ideas and suggestions on how to set up the foundations.
- Experts in mathematical foundations who (perhaps inspired by Section 1.1) might be interested in equivariance as an interesting new foundational principle for its own sake.

Looking at the instruction 'assume a set of atoms', an expert in foundations may take this to mean that we work in Zermelo-Fraenkel set theory with atoms (**ZFA**), instead of in Zermelo-Fraenkel set theory (**ZF**). This intuition is correct, albeit not a full picture, so let us start there.

Now ZFA is unnecessary because atoms could be modelled by $\mathbb{N}$ in ZF (or by $pset(\mathbb{N})$ if we want more atoms, and so forth). In fact ZFA and ZF are equivalent and biinterpretable in the sense that any model of ZFA can be embedded in a model of ZF, and vice-versa, and anything that we express in ZFA can be translated (quite easily) to an assertion about ZF, and vice-versa. So in terms of expressivity, ZF and ZFA are the same. And yet:

- if the translation from ZFA to ZF leads to, say, a quadratic increase in proof-size, or
- if ZFA provides an environment in which it is easier to express ourselves, or
- if ZFA is an environment which naturally lets us perceive *native ZFA concepts*[1]

then the net gain from working in ZFA can be significant *even if* everything could in principle be compiled back down to ZF.

An analogy: Roman numerals I, II, III, IV, can express numbers just as arabic numerals 0, 1, 2, 3, 4, but nobody seriously asserts they are functionally equivalent. There is such a thing as 'a good foundation' and 'a poor foundation' for a given task; foundations matter.

So in this paper we will explore specifically and in detail what it means when we write 'assume a set of atoms'. Specifically, I propose—perhaps a little provocatively—that it means we are working in **equivariant Zermelo-Fraenkel set theory** or *EZFA* for short.

Now a foundation for nominal techniques has been proposed before, in the paper which started the topic [GP01]: Fraenkel-Mostowski set theory (**FM**), which is an elaboration of ZFA with a *finite support* axiom (see Subsection 4.3). We will discuss FM in detail, but we can note in this Introduction that FM is inconsistent with the Axiom of Choice, whereas EZFA has the advantage that EZFA plus Choice (EZFAC) is consistent.

[1] ...meaning concepts that are hard to address in full generality in ZF, where we do not give ourselves atoms, but easy to see in ZFA, where we do. We will see two examples in this paper: equivariance (see Remark 3.11); and freshness and support (see Subsection 4.3.2); and Subsection 1.1 is, in a sense, a longer list of native ZFA concepts either combined with and extending familiar concepts from ZF, or leading to entirely new ways constructs and ways of thinking.

1.1. **Motivations for considering the foundations of equivariance.** This list, which is nonexhaustive and in no particular order, indicates some of the applications of nominal ideas and why working in an environment in which atoms are explicitly available, can be helpful:

(1) In EZFA, the notion of finiteness generalises naturally to *orbit-finiteness*; the property a set can have of having finitely many orbits under permutations of atoms (see Definition 2.16 and Subsection 4.2).

We can then work with orbit-finite computational structures, such as orbit-finite automata. See [BKL14, KKOT15].

(2) In a similar vein we can consider nominal generalisations of Kleene Algebras [GC11, KMPS15, KMS17], thus, classes of languages that include orbits under permutations of atoms.

(3) Homotopy Type Theory [Uni13] is a dependent type theory whose types include *paths*; simplifying somewhat, two elements are 'equal' when there is a path from one to the other. There are many semantics for these paths; a nominal semantics provides one of the simplest and cleanest [CCM17].

(4) The native universal algebra[2] of EZFA significantly and interesting generalises that of ZF.

We can give axiomatisations of theories with binders, including substitution, first-order logic, and the λ-calculus [GM08, Gab14, GG17].[3]

These axiomatisations lead to Stone Dualities between: algebras for first-order logic and the λ-calculus; and topological spaces with points naturally constructed in EZFA [Gab14, GG17].

Furthermore the general theory of nominal algebra models has a rich structure. This includes familiar constructs, such as a nice generalisation of the HSP theorem [Gab09], as well as constructions for which mathematics based on ZF has no correspondent [Gab12b, Gab12a].

(5) The original motivation for nominal techniques was *nominal abstract syntax*, essentially a generalisation of tree-structured data to include name-binding [GP01, Gab11, Pit13]. Extensive implementations exist, most notably perhaps Nominal Isabelle [Urb08]. These implementations are designed to allow us to specify and reason about all the applications listed here, and more—starting of course with the syntax and operational semantics of logic and programming.

(6) Nominal rewriting [FG07] is a theory of rewriting (directed equality) that lets us reason on theories with binding and has good properties, such as most general unifiers.

The theory of syntax and meta-syntax in EZFA is itself rich; aside from nominal abstract syntax we have nominal terms and their unification [UPG04], nominal rewriting as mentioned above, connections to the simply-typed λ-calculus and higher-order logic [LV12, DG12b], and an equation between infinite streams and meta-languages [Gab12b].

(7) The equivariance properties described in Figure 2 and Theorem 3.6 are generalisations of α-equivalence. Conversely, the variable-renaming that unobtrusively converts $\forall x.(x{=}x)$ into $\forall y.(y{=}y)$ is a special case of the general set-theoretic principles discussed in this paper. So the equivariance principle of this paper is already known to the reader, much as ring theory is already known to anybody who has added and multiplied numbers.

(8) An EZFA model using (and considerably developing on) the tools in this paper was used to prove the consistency of Quine's NF [Gab16], solving a longstanding open consistency problem in set theory.

(9) More references are e.g. in the bibliography of [Pit13].

[2] Logic and models of equational theories, i.e. of sets of equalities between terms.

[3] The informed reader will wonder whether these theories look a bit like cylindric algebras [HMT85]. The answer is: yes and no, in the same way that the natural numbers do look a bit like atoms; but only from a distance, and the closer we look the less similar they become.

(**AtmEmpty**)	$t \in s \Rightarrow s \notin \mathbb{A}$
(**EmptySet**)	$t \notin \varnothing$
(**Extensionality**)	$s, s' \notin \mathbb{A} \Rightarrow (\forall \mathtt{b}.(\mathtt{b} \in s \Leftrightarrow \mathtt{b} \in s')) \Rightarrow s = s'$
(**Comprehension**)	$s \in \{\mathtt{a} \in t \mid \phi\} \Leftrightarrow (s \in t \wedge \phi[\mathtt{a}{:=}s])$
(**Pair**)	$t \in \{s, s'\} \Leftrightarrow (t = s \vee t = s')$
(**Union**)	$t \in \bigcup s \Leftrightarrow \exists \mathtt{a}.(t \in \mathtt{a} \wedge \mathtt{a} \in s)$
(**Powerset**)	$t \in pset(s) \Leftrightarrow t \subseteq s$
(**Induction**)	$(\forall \mathtt{a}.(\forall \mathtt{b} \in \mathtt{a}.\phi[\mathtt{a}{:=}\mathtt{b}]) \Rightarrow \phi) \Rightarrow \forall \mathtt{a}(\phi)$
(**Infinity**)	$\exists \mathtt{c}.\varnothing \in \mathtt{c} \wedge \forall \mathtt{a}.\mathtt{a} \in \mathtt{c} \Rightarrow \mathtt{a} \cup \{\mathtt{a}\} \in \mathtt{c}$
(**AtmInf**)	$\neg(\mathbb{A} \subseteq_{fin} \mathbb{A})$
(**Replacement**)	$\exists \mathtt{b}.\forall \mathtt{a}.\mathtt{a} \in \mathtt{b} \Leftrightarrow \exists \mathtt{a}'.\mathtt{a}' \in u \wedge \mathtt{a} = F(\mathtt{a}')$
(**Choice**)	$\varnothing \neq (pset^*(s) \to s)$ $\quad$ $pset^*$ is nonempty powerset

Figure 1: Axioms of ZFA(C)

(**Equivar**) $\quad \forall \mathtt{a} \in Perm.(\phi \Leftrightarrow \phi[\mathtt{a}_1{:=}\mathtt{a}\cdot\mathtt{a}_1, \ldots, \mathtt{a}_n{:=}\mathtt{a}\cdot\mathtt{a}_n]) \quad fv(\phi) = \{\mathtt{a}_1, \ldots, \mathtt{a}_n\}$

Figure 2: Equivariance axiom of EZFA

2. The language of sets with atoms

We establish the syntax and semantics of ZFA set theory. The reader already familiar with this might prefer to skip straight to Definition 2.14, which is the first item that is not a standard ZFA definition.

2.1. Equivariant ZFA.

Definition 2.1. Assume a countably infinite collection of **variable symbols** $\mathtt{a}, \mathtt{b}, \mathtt{c}, \ldots$. Let the **language of sets with atoms** be defined by:

$$s, t ::= \mathtt{a} \mid \{s, t\} \mid pset(s) \mid \bigcup s \mid \{\mathtt{a} \in t \mid \phi\} \mid \varnothing \mid \mathbb{A}$$
$$\phi ::= s = t \mid t \in s \mid \bot \mid \phi \wedge \phi \mid \forall \mathtt{a}.\phi$$

Remark 2.2. Definition 2.1 defines a language of first-order logic with:

- equality $=$,
- sets membership $\in$,
- terms for pairset $\{s, t\}$, union $\bigcup s$, powerset $pset(s)$, bounded comprehension $\{s \in t \mid \phi\}$, and
- constant symbols for the empty set $\varnothing$ and the set of atoms $\mathbb{A}$.

This is a sufficient foundation for mathematics, and we can define:

Definition 2.3. (1) Write **ZFA** for axioms (**AtmEmpty**) to (**Replacement**) from Figure 1.
(2) Write **ZFAC** for ZFA plus (**Choice**).
(3) Write **EZFA** for ZFA plus (**Equivar**) from Figure 2.
(4) Write **EZFAC** for ZFA plus (**Choice**) plus (**Equivar**).
In full, EZFAC stands for *Equivariant Zermelo-Fraenkel Set Theory with Atoms and Choice*.

Notation 2.4. In Figure 1 and elsewhere we may use standard syntactic sugar. For example:

- In (**Powerset**) we write $t \subseteq s$ and this is shorthand for $\forall \mathtt{a}.(\mathtt{a} \in t \Rightarrow \mathtt{a} \in s)$.

- In (**AtmInf**) we write $\subseteq_{\mathit{fin}}$. For a given y, the relation $x \subseteq_{\mathit{fin}} y$ is inductively defined to be least such that $\varnothing \subseteq_{\mathit{fin}} y$ and $x \subseteq_{\mathit{fin}} y \wedge y' \in y$ implies $x \cup \{y'\} \subseteq_{\mathit{fin}} x$.
- In (**Choice**) we write $pset^*(x)$ and this is shorthand for the set of nonempty subsets $\{\mathtt{a} \in pset(x) \mid \mathtt{a} \neq \varnothing\}$.
- We will write $\hookrightarrow$ for injective function-sets, where a function-set is an element representing the graph of a function. So $x \hookrightarrow y$ is the set of graphs of injective functions from x to y.

The interested reader can find detailed explanations of how these standard constructs are built up from first principles in e.g. Section 5 of [Joh87]. We will assume these basic set-theoretic conventions henceforth. See also the encyclopaedic treatment of set theory in [Jec06] and in particular see page 250 where ZFA is discussed.

Remark 2.5. (1) There is redundancy in Definition 2.1:
- We could restrict terms to just be variable symbols $\mathtt{a}$ (losing term-formers for pairset, powerset, union, comprehension, emptyset, and the set of atoms).
- We could restrict predicates by losing equality (just keeping $\in$, $\bot$, $\wedge$, and $\forall$).

This restricted language would be just as expressive, but more verbose.

(2) The axioms in Figures 1 and 2 are actually axiom-schemes (for all s, t, s', ϕ, and F). We could reduce the number of axiom-schemes, in some cases very easily: for instance by rewriting (**Emptyset**) as $\forall \mathtt{a}.\mathtt{a} \notin \varnothing$. We will not be too concerned about this and we just optimise for what seems to be the most readable form.

Remark 2.6. We briefly spell out intuitions for the axioms:
- (**AtmEmpty**) Atoms have no elements (so they are extensionally equal to the empty set $\varnothing$).
- (**EmptySet**) $\varnothing$ has no elements.
- (**Extensionality**) If x and x' are sets and have the same elements, then $x = x'$.
- (**Comprehension**) If y is an element then $\{x{\in}y \mid \phi(x)\}$ is a set.
- (**Pair**) If x and x' are elements then so is $\{x, x'\}$.
- (**Union**) If x is an element then so is $\bigcup x = \{x'' \mid x'' \in x' \in x\}$.
- (**Powerset**) If x is an element then so is $pset(x)$ its collection of subsets.
- (**Induction**) $\in$ is well-founded.
- (**Infinity**) A set exists that contains $\mathbb{N}$, so that (by comprehension) $\mathbb{N}$ is a set.
- (**AtmInf**) This expresses that $\mathbb{A}$ is an infinite set.[4]
- (**Replacement**) If F is a function-class and z is a set then $\{F(x) \mid x \in z\}$ is a set.[5]
- (**Choice**) If x is a set then there exists a **choice function**-set mapping $pset^*(x)$ to x.
- We discuss (**Equivar**) from Figure 2 from Remark 2.17 onwards, after we have constructed some necessary machinery.

Remark 2.7 (A design alternative: Quine atoms)**.** An alternative to (**AtmEmpty**) is to use an axiom (**AtmQuine**) that $a = \{a\}$ (these are called *Quine atoms*). In the implementation of Fraenkel-Mostowski set theory in [GP01] it was convenient to use Quine atoms because doing so removed the condition $s, s' \notin \mathbb{A}$ from (**Extensionality**). The axiom (**Induction**) needs to be slightly adjusted instead but this is used far less often.

[4]There are other ways of saying that $\mathbb{A}$ is infinite; for instance that $\varnothing \neq \mathbb{N} \hookrightarrow \mathbb{A}$ (in words: there exists a sets injection of natural numbers into $\mathbb{A}$). But that axiom would also imply that an infinite subset of $\mathbb{A}$ can be well-ordered, which FM teaches us is a stronger and possibly undesired assumption. (**AtmInf**) as written is elementary, in the sense that it does not smuggle in this assumption.

[5]The power here is that F could be a proper class. If F is a function-set then it is easy to prove that $\{F(x) \mid x \in z\}$ is a set using comprehension, because then $img(F)$ the image of F is a set and we can write $\{F(x) \mid x \in z\}$ as $\{x' \in img(F) \mid x' = F(x) \wedge x \in z\}$.

Some people are suspicious of Quine atoms because they make the universe non-wellfounded. This is unnecessary: the non-wellfoundedness that Quine atoms introduce is so mild as to be negligible, and even were this not the case, fears of non-wellfounded sets may sometimes be overblown.

This concludes the presentation of the syntax and axioms of ZFAC. We will discuss $(\mathbf{Equivar})$ in Remark 2.17 and Section 3. First, we need to develop some terminology and a theory of denotation:

2.2. **Atoms, sets, and denotations.**

Notation 2.8. • If $a \in \mathbb{A}$ then call a an **atom**.
• If $x \notin \mathbb{A}$ then call x **a set**.
• We call an x that is either an atom or a set, an **element**.

Remark 2.9. So:
• $(\mathbf{AtmInf})$ from Figure 1 states that atoms are empty, and
• $(\mathbf{EmptySet})$ and $(\mathbf{Extensionality})$ imply that the only empty *set* is $\varnothing$.

Definition 2.10. Define **free variables** $fv(s)$ and $fv(\phi)$ as usual. For example, $fv(\forall \mathtt{a}.(\mathtt{a} = \mathtt{b})) = \{\mathtt{b}\}$.

If $fv(s) = \varnothing$ or $fv(\phi) = \varnothing$ (so s or ϕ have no free variables) then call them **closed**.

We introduce a standard abuse of notation.

Definition 2.11. Suppose $\mathfrak{M}$ is a model of ZFA. Enrich the term and predicate language from Definition 2.1 by admitting elements of $\mathfrak{M}$ as constants. Call terms in this new language **$\mathfrak{M}$-terms** and **$\mathfrak{M}$-predicates**.

If s or ϕ do *not* mention any of these additional contants from $\mathfrak{M}$, then call them **pure**.

Definition 2.12. Suppose $\mathfrak{M}$ is a model of ZFA and suppose s is a closed $\mathfrak{M}$-term and ϕ is a closed $\mathfrak{M}$-predicate. Then define **denotations**

$$[\![s]\!]_{\mathfrak{M}} \quad \text{and} \quad \mathfrak{M} \vDash \phi$$

by the usual inductive definition. We give a relevant selection of cases:
• $\mathfrak{M} \vDash y \in x$ when $y \in_{\mathfrak{M}} x$; that is, when (y, x) is in the $\in_{\mathfrak{M}}$ relation that we assumed when we wrote 'suppose $\mathfrak{M}$ is a model of ZFA' at the start of this Definition.
• $\mathfrak{M} \vDash x = x'$ when $x = x'$.[6]
• $\mathfrak{M} \vDash \forall \mathtt{a}.\phi$ when $\mathfrak{M} \vDash \phi[\mathtt{a}{:=}x]$ for every $x \in \mathfrak{M}$.
• $[\![x]\!]_{\mathfrak{M}} = x$ so that our extra constants denote themselves in $\mathfrak{M}$.
• $[\![\{\mathtt{a} \in s \mid \phi\}]\!] = \{x \in [\![s]\!]_{\mathfrak{M}} \mid \mathfrak{M} \vDash \phi[\mathtt{a}{:=}x]\}_{\mathfrak{M}}$; such an element exists in $\mathfrak{M}$, by $(\mathbf{Comprehension})$.
• ...and so forth.

[6] So we interpret equality by literal identity. This is a standard design choice but not a necessary one. We could introduce an 'equality relation' $=_{\mathfrak{M}}$ instead, but then we would need axioms saying that $\mathfrak{M}$ cannot distinguish $\mathfrak{M}$-equal elements.

2.3. **Pairs and permutations.** We recall some standard terminology:

Definition 2.13. (1) The Kuratowski **ordered pair** of $x, y \in \mathfrak{M}$ is defined by

$$(x, y) = \{\{x, y\}, \{x\}\} \in \mathfrak{M}.$$

(2) A **function-set** in $\mathfrak{M}$ is an element $f \in \mathfrak{M}$ that is a set of ordered pairs that define the graph of a function in $\mathfrak{M}$. Thus we can write that

$$f = \{(x, y) \in f \mid y = f(x)\} \in \mathfrak{M}.$$

Definition 2.14. (1) Let a(n $\mathfrak{M}$**-)permutation** π be a bijection on atoms.
(2) Write $Perm$ for the **set of permutations**. In symbols:

$$Perm = \{\pi \in \mathbb{A} \cong \mathbb{A}\} \in \mathfrak{M}.$$

(3) Write id for the **identity** permutation such that $\mathrm{id}(a) = a$ for all a.
(4) Write $\pi' \circ \pi$ for composition, so that $(\pi' \circ \pi)(a) = \pi'(\pi(a))$.
(5) Write π^{-1} for the inverse of π, so that $\pi^{-1} \circ \pi = \mathrm{id} = \pi \circ \pi^{-1}$.
(6) If $a, b \in \mathbb{A}$ then write $(a\ b)$ for the **swapping** (terminology from [GP01]) mapping a to b, b to a, and all other c to themselves, and take $(a\ a) = \mathrm{id}$.

Remark 2.15. $Perm$ is an element in $\mathfrak{M}$; a permutation $\pi \in \mathfrak{M}$ is a bijection of $\mathbb{A}$ represented as its function-set-graph in $\mathfrak{M}$. There may be bijections of $\mathbb{A}$ that cannot be represented as elements of $\mathfrak{M}$. These are simply not function-sets inside the model. This is a standard distinction. We go into more detail about the possible design choices for $Perm$ in Subsection 4.2.

Definition 2.16 (The pointwise action)**.** Given a permutation $\pi \in Perm$ we define a **pointwise (atoms-)permutation action** $\pi{\cdot}x$ by $\in$-induction in $\mathfrak{M}$ as follows:

$$\begin{array}{ll} \pi{\cdot}a = \pi(a) & a \in \mathbb{A} \\ \pi{\cdot}X = \{\pi{\cdot}x \mid x \in X\} & X \text{ a set} \end{array}$$

Remark 2.17. We now have the machinery needed to read axiom $(\mathbf{Equivar})$ from Figure 2: given any predicate in the language from Definition 2.1, $(\mathbf{Equivar})$ asserts that validity is unaffected by uniformly permuting atoms in the free parameters of that predicate.

In other words, $(\mathbf{Equivar})$ states that atoms can be permuted provided we do so consistently in all parameters. For instance if we have proved $\phi(a, b, c)$, then

- taking $\pi = (a\ c)$ we also know by $(\mathbf{Equivar})$ that $\phi(c, b, a)$ and
- taking $\pi = (a\ a')(b\ b')(c\ c')$ we also know by $(\mathbf{Equivar})$ that $\phi(a', b', c')$, but
- $(\mathbf{Equivar})$ does not tell us that $\phi(a, b, a)$; this may still hold, just not by equivariance because no permutation takes (a, b, c) to (a, b, a).

We now come to a rather subtle observation:

Remark 2.18 (The dual nature of atoms)**.** $(\mathbf{Equivar})$ expresses that atoms have a dual nature:

- individually, atoms behave like pointers to themselves,[7] but
- collectively, atoms have the flavour of variables ranging permutatively over the set of all atoms.[8]

We will continue this discussion in Remark 3.10.

[7] If atoms are Quine atoms as per Remark 2.7 then this is literally true, in the sense that $a = \{a\}$.

[8] This too can be made precise: see Subsection 2.6 and Lemma 4.17 of [DG12b].

2.4. **Permutation is a group action.** We take a moment to note that the permutation action is a group action on the sets universe, and then we will study equivariance in more detail in Section 3.

Lemma 2.19. *Suppose $\pi, \pi' \in \mathfrak{M}$ are permutations and $x \in \mathfrak{M}$ is any element. Then we have:*

$$\begin{array}{c}\mathrm{id}{\cdot}x = x\\ \pi{\cdot}(\pi'{\cdot}x) = (\pi \circ \pi'){\cdot}x.\end{array}$$

Proof. By a routine $\in$-induction on $\mathfrak{M}$. □

Corollary 2.20. *Suppose $\pi \in Perm \in \mathfrak{M}$ is a permutation. Then the π-action on $\mathfrak{M}$*

$$x \in \mathfrak{M} \longmapsto \pi{\cdot}x \in \mathfrak{M}$$

is a bijection.

Proof. From Lemma 2.19 noting that $\pi{\cdot}(\pi^{-1}{\cdot}x) = x$. □

3. Equivariance

3.1. **Some helpful notation.**

Notation 3.1. Suppose $\mathfrak{M}$ is a model of ZFA and suppose ϕ is a $\mathfrak{M}$-predicate and s and π are $\mathfrak{M}$-terms (writing 's and t' would be more principled, but the reader will soon see why we write the second term π). Then define

$$\pi{\cdot\cdot}\phi \quad \text{and} \quad \pi{\cdot\cdot}s$$

by:

(1) $\pi{\cdot\cdot}\phi$ is that predicate obtained by replacing every free variable $\mathtt{a}$ in ϕ with $\pi{\cdot}\mathtt{a}$, and every $x \in \mathfrak{M}$ in ϕ with $\pi{\cdot}x$.
(2) $\pi{\cdot\cdot}s$ is that term obtained by replacing every free variable $\mathtt{a}$ in s with $\pi{\cdot}\mathtt{a}$, and every $x \in \mathfrak{M}$ in s with $\pi{\cdot}x$.

An inductive definition would be routine to write out.

Example 3.2. For example, if $\mathfrak{M}$ is a model of ZFA and $\pi \in Perm \in \mathfrak{M}$ is a permutation, then

$$\pi{\cdot\cdot}\forall\mathtt{a}.(x \in \mathtt{a} \wedge x \notin \mathtt{b}) \quad \text{is} \quad \forall\mathtt{a}.(\pi{\cdot}x \in \mathtt{a} \wedge \pi{\cdot}x \notin \pi{\cdot}\mathtt{b}).$$

We link Notation 3.1 to (**Equivar**) from Figure 2:

Lemma 3.3. *Suppose ϕ is a pure predicate (so mentions no elements of $\mathfrak{M}$) and s is a pure term, with free variables $\mathtt{a}_1, \ldots, \mathtt{a}_n$. Then $\pi{\cdot\cdot}\phi$ and $\pi{\cdot\cdot}s$ can be rewritten more explicitly as follows:*

$$\begin{array}{lll}\pi{\cdot\cdot}\phi & \textit{is} & \phi[\mathtt{a}_1{:=}\pi{\cdot}\mathtt{a}_1, \ldots, \mathtt{a}_n{:=}\pi{\cdot}\mathtt{a}_n]\\ \pi{\cdot\cdot}s & \textit{is} & s[\mathtt{a}_1{:=}\pi{\cdot}\mathtt{a}_1, \ldots, \mathtt{a}_n{:=}\pi{\cdot}\mathtt{a}_n]\end{array}$$

As a corollary, (**Equivar**) *from Figure 2 can be rewritten in the following more compact form:*

(**Equivar**) $\quad \forall\mathtt{a}{\in}Perm.(\phi \Leftrightarrow \mathtt{a}{\cdot\cdot}\phi).$

Proof. A fact of syntax. □

Remark 3.4. The (**Equivar**) from Lemma 3.3 has two advantages over the (**Equivar**) in Figure 2: it is more compact, and if we extend our language we can cleanly extend the axiom-scheme by fine-tuning how $\cdot\cdot$ behaves on the new terms. See Remarks 3.16 and 3.17.

The only disadvantage to the (**Equivar**) from Lemma 3.3 is that we do need to define $\cdot\cdot$ first.

3.2. **Equivariance was there all along.**

Lemma 3.5. *Suppose $\mathfrak{M}$ is a model of ZFA(C). Then:*
(1) $\mathfrak{M} \models \pi{\cdot}y \in \pi{\cdot}x$ *if and only if* $\mathfrak{M} \models y \in x$.
(2) $\mathfrak{M} \models \pi{\cdot}x = \pi{\cdot}y$ *if and only if* $\mathfrak{M} \models x = y$.
(3) $\mathfrak{M} \models \pi{\cdot}x \subseteq \pi{\cdot}y$ *if and only if* $\mathfrak{M} \models x \subseteq y$.

Proof. By a routine induction on $\mathfrak{M}$ using the fact that:

- By Corollary 2.20 π acts bijectively on atoms, and
- by construction in Definition 2.16 π acts pointwise on sets. □

We can now observe that every model of ZFA or ZFAC *already* satisfies the equivariance axiom-scheme (**Equivar**) from Figure 2. The result goes back to [Gab01, Theorem 8.1.10]:

Theorem 3.6. *If $\mathfrak{M}$ is a model of ZFA(C) then $\mathfrak{M}$ is also a model of EZFA(C).*

Proof. Examining (**Equivar**) from Figure 2 we see that it suffices to show that for every closed predicate ϕ and closed term s (possibly mentioning elements of $\mathfrak{M}$):

$$\mathfrak{M} \models \phi \Leftrightarrow \pi{\cdot\cdot}\phi \quad \text{and}$$
$$\mathfrak{M} \models \pi{\cdot}s = \pi{\cdot\cdot}s.$$

We reason by induction on syntax:

- *The cases of $t \in s$ and $s = t$.* From Lemma 3.5.
- *The case of $\forall \mathtt{a}.\phi$.* Suppose $\mathfrak{M} \models \forall \mathtt{a}.\phi$. By Corollary 2.20 the action of π on $\mathfrak{M}$ is a bijection, so that $\mathfrak{M} \models \phi[\mathtt{a}{:=}\pi^{-1}{\cdot}x]$ for every $x \in \mathfrak{M}$. By inductive hypothesis it follows that $\mathfrak{M} \models \pi{\cdot}(\phi[\mathtt{a}{:=}\pi^{-1}{\cdot}x])$ and thus $\mathfrak{M} \models (\pi{\cdot\cdot}\phi)[\mathtt{a}{:=}x]$ for every $x \in \mathfrak{M}$. Thus $\mathfrak{M} \models \forall \mathtt{a}.(\pi{\cdot\cdot}\phi)$ and so $\mathfrak{N} \models \pi{\cdot\cdot}\forall \mathtt{a}.\phi$ as required.
- *The cases of $\varnothing$ and $\mathbb{A}$.* It is clear from the pointwise action in Definition 2.16 that $\mathfrak{M} \models \pi{\cdot}\varnothing = \varnothing$ and by assumption π is a bijection on $\mathbb{A}$ so that again from Definition 2.16 that $\mathfrak{M} \models \pi{\cdot}\mathbb{A} = \mathbb{A}$.
- *The cases of $\bot$ and $\wedge$.* Routine.
- Other cases are no harder; and we know they must work because everything else is definable in terms of $\in$, $=$, $\forall$, $\wedge$, and $\bot$. □

A detailed discussion of the significance of Theorem 3.6 is in Subsection 3.5.

3.3. **An example.**

When we define a function in nominal techniques, we almost always want to know that it commutes with the permutation action. Broadly speaking we have three options:

- Check this by explicit calculations for every function that we define.
- Use Theorem 3.6.
- Use the axiom-scheme (**Equivar**).

We illustrate these three methods on a trio of simple but characteristic examples:

Lemma 3.7. (1) *If $x, y \in \mathfrak{M}$ then*

$$\pi{\cdot}(x,y) = (\pi{\cdot}x, \pi{\cdot}y).$$

(2) *If $x \in \mathfrak{M}$ then*

$$\pi{\cdot}pset(x) = pset(\pi{\cdot}x).$$

(3) *If $f \in (X \hookrightarrow Y) \in \mathfrak{M}$ then $\pi{\cdot}f \in (\pi{\cdot}X \hookrightarrow \pi{\cdot}Y) \in \mathfrak{M}$.*

In words: if f is an injective function-set in $\mathfrak{M}$ from X to Y then $\pi{\cdot}f$ is an injective function-set in $\mathfrak{M}$ from $\pi{\cdot}X$ to $\pi{\cdot}Y$.

Proof. All immediate from Theorem 3.6 or (**Equivar**), as we prefer.

For the reader's convenience we now sketch the explicit calculations that this abstract result corresponds to for the three specific cases of this result. We reason using the pointwise action from Definition 2.16, and Lemmas 2.19 and 3.5, and Definition 2.13, as follows:

$$\begin{aligned}\pi{\cdot}(x,y) &= \pi{\cdot}\{\{x,y\},\{x\}\}\\ &= \{\{\pi{\cdot}x, \pi{\cdot}y\},\{\pi{\cdot}x\}\}\\ &= (\pi{\cdot}x, \pi{\cdot}y)\end{aligned}$$

$$\begin{aligned}\pi{\cdot}pset(x) &= \pi{\cdot}\{x' \mid x' \subseteq x\}\\ &= \{\pi{\cdot}x' \mid x' \subseteq x\}\\ &= \{x' \mid \pi^{-1}{\cdot}x' \subseteq x\}\\ &= \{x' \mid x' \subseteq \pi{\cdot}x\}\\ &= pset(\pi{\cdot}x)\end{aligned}$$

$$\begin{aligned}\pi{\cdot}f &= \{\pi{\cdot}(x,y) \mid (x,y) \in f\}\\ &= \{(\pi{\cdot}x, \pi{\cdot}y) \mid (x,y) \in f\}\end{aligned}$$

□

3.4. **Equivariance and Choice.**

Remark 3.8. (**Choice**) is not mentioned in Theorem 3.6. How can Choice be compatible with equivariance; surely making arbitrary choices is inherently non-equivariant?

Not if the choices are made inside $\mathfrak{M}$: from Lemma 3.7 (or direct from Theorem 3.6)

$$f \in (pset^*(x) \hookrightarrow x) \quad \text{implies} \quad \pi{\cdot}f \in pset^*(\pi{\cdot}x) \hookrightarrow \pi{\cdot}x.$$

In words: if f is a choice function for x in $\mathfrak{M}$ then $\pi{\cdot}f$ is a choice function for $\pi{\cdot}x$ in $\mathfrak{M}$. We just permute atoms pointwise in the choice functions.

Remark 3.9. Continuing Remark 3.8, the following statements are consistent with EZFA (and are derivable in EZFAC):

(1) "There exists a total ordering on $\mathbb{A}$".
(2) "Every set can be well-ordered (even if the set mention atoms)".

More on this in Remark 4.10.

3.5. **What is equivariance.** We continue the discussion from Remarks 2.17 and 2.18:

Remark 3.10. Equivariance appears in this paper twice:

- as an axiom-scheme (**Equivar**) in Figure 2 and
- as a ZFA theorem in Theorem 3.6.

Each can be derived from the other. So which should we take as primitive: the axiom-scheme or the Theorem?

(**Equivar**)-the-axiom-scheme is derivable by Theorem 3.6, just like explicit pairset, powerset, union, comprehension, and emptyset terms in Definition 2.1 are derivable (see Remark 2.5). But being derivable is not the same as being useless. Consider:

Remark 3.11 (Equivariance is native to ZFA)**.** Equivariance is what we might call a *native ZFA concept*: it clearly naturally inhabits a ZFA universe.

What this means in practice is that some readers find equivariance hard to understand, and pointing to equivariance-the-theorem in Theorem 3.6 does not help as much as it should because the difficulty is not the mathematics so much as in the background assumptions which the reader brings to the paper.

It seems unintuitive; surely there must be a mistake somewhere: "We can't just permute elements. Suppose atoms are numbers: then are you claiming $1 < 2$ if and only if $2 < 1$?". This is a silly objection from inside ZFA—where "Suppose atoms are numbers" makes no sense, because they are just not—but from inside ZF, in which atoms *have* to *be* something else that is not atoms, we can see why such questions are asked.[9] The reader's experience may differ but for me the background assumption that atoms cannot just *be* atoms has been, to this day, a significant impediment to explaining nominal techniques.

EZFA can serve as a practical device with which to get talking about nominal material. When asked "What are atoms?" we can simply answer "elements of $\mathbb{A}$ in EZFA"; and when asked "Why equivariance?" we can simply answer "Because we are in EZFA, and it is an axiom.". And when asked "What else?" we can simply answer "That's it.".

Remark 3.12 (Equivariance is a natural axiom)**.** If we are working in a theorem-prover, the equivariance axiom-scheme assumes a particular importance.

The problem is that while in principle every instance of the axiom-scheme $(\mathbf{Equivar})$ can be derived, in practice the cost of proving all those instances from first principles à la Theorem 3.6 scales up with the complexity of ϕ, and it is not trivial to automate.

My PhD thesis contained an implementation of FM set theory inside Isabelle, and in the end this issue of proving countless instances of Theorem 3.6 caused the development to stall.

The irony of an implementation of nominal techniques stalling due to the cost of proving renaming lemmas was not lost on me at the time, and I concluded that it was very important inside a nominal theorem-prover that equivariance have constant cost to the user, regardless of the size of the predicate ϕ.

Assuming an axiom $(\mathbf{Equivar})$ is an effective way to reduce the cost of renaming to *constant* effort, namely, the cost of invoking the axiom. If instead implementation requires explicit proofs of equivariance properties in the style of Theorem 3.6, then our implementation needs to make sure that all instances of this Theorem are proved in a fully automated manner, achieving the same effect *as if* the Theorem were an Axiom. In this sense, we can say that from the point of view of implementation Equivariance is a natural axiom.

---

[9] Case in point: while describing nominal techniques to a category-theorist colleague I mentioned $\mathbb{A}$ and he asked 'but what *are* atoms?'. I said atoms were atoms; he refused to accept this, and I was beaten back to setting $\mathbb{A}$ to $\mathbb{N}$ just so the conversation could make progress.

From this and the rest of the discussion I was disconcerted to conclude that my category-theoretic colleague, even though he was happy to switch toposes in mid-sentence, was a ZF fundamentalist. For him, if it was not translated all the way down to ZF, then it was incomprehensible.

This paper might not cure him of this mindset but it does provide a fuller answer to his original question.

Another case in point: in a paper on permissive-nominal techniques, in which we assume a set of atoms $\mathbb{A}$ and a partition of $\mathbb{A}$ into two infinite halves $\mathbb{A}^<$ and $\mathbb{A}^>$, a referee refused to allow the paper to be published. I finally understood that the sticking point was whether the partition was computable. Since after all, in ZF atoms have to be something else, so it could be that $\mathbb{A} = \mathbb{N}$ and $\mathbb{A}^<$ is 'accidentally' an uncomputable set. We could have asserted that $\mathbb{A}^<$ was computable, but this would validate the ZF-style premise that there must necessarily be internal structure to atoms; that is, that atoms are not *really* atoms. The problem was solved by assuming $\mathbb{A}^<$ and $\mathbb{A}^>$ *first* and then defining $\mathbb{A} = \mathbb{A}^< \cup \mathbb{A}^>$. This seemed to work: the referee was satisfied and the paper published.

Remark 3.13. In view of the difficulties discussed in Remark 3.12 arising from the practical cost of implementing Theorem 3.6 in Isabelle, after my PhD I initiated a 'mark 2' axiomatisation of nominal techniques in which equivariance was an axiom-scheme (technically: an Isabelle Oracle).

This contrasts for example with the first-order axiomatisation spun off by Pitts [Pit01, Pit03] at the same time, which following [GP99, GP01] treated equivariance as a theorem in the style of Theorem 3.6.

So it is an interesting aspect of the mark 2 implementation that equivariance was an axiom-scheme in the style of $(\mathbf{Equivar})$. Where other work has treated equivariance, it has treated it as a theorem in the style of Theorem 3.6 (see for example Theorem 8.1.10 of [Gab01], Lemma 4.7 of [GP01], and Proposition 2 of [Pit03]). A message of the mark 2 implementation is that in practical engineering terms, it may be useful for equivariance to be an axiom.

Remark 3.14 (Equivariance the practical time-saver)**.** We take a step back and consider the practical application of equivariance to writing mathematics papers. Provided the reader accepts Theorem 3.6—and alas, as outlined in Remark 3.11 not all readers do—it costs the same to write 'from $(\mathbf{Equivar})$' as it does to write 'from Theorem 3.6': it is all just 'equivariance'.

Consider our example Lemma 3.7. This has a one-line proof from equivariance which we give first, and a longer proof by explicit calculations which we also write out (though not in full).

In practice, in nominal techniques if we write a definition we will most probably want to prove it equivariant. Without an equivariance principle like $(\mathbf{Equivar})$/Theorem 3.6, the cost of these proofs increases roughly linearly with complexity for each definition, and so can rise roughly quadratically for the paper overall.[10]

Nominal techniques were developed to cut development time of formal proofs in theorem-provers, but they are also effective in rigorous but informal proofs—i.e. for ordinary mathematics. Equivariance creates value because it clears tangles of lemmas into a single unifying principle and distils their long proofs by calculation down to crisp one-liners.

This is clear even from simple examples: consider the first line of the proof of Lemma 3.7, and the proof of Lemma 4.9.

The first practical application of equivariance in rigorous but informal mathematics was in [Gab07]; see Section 7.4. For more recent examples see uses of Theorem 2.13 of [Gab14] and of Theorem 2.3.1 of [GG17]. The reader can find practical examples of how equivariance can be usefully applied in normal mathematics, in those papers.

Remark 3.15. In conclusion:

- When we introduced nominal techniques in [GP01], we could have marketed EZFA as a foundation, and not FM.[11]
- When we write maths papers using nominal techniques we might do well to be explicit that we are working in ZFA or EZFA.
- Equivariance can be usefully applied in the background of normal maths papers, and when so used it converts collections of long renaming lemmas into crisp one-liners (just like it can do in a theorem-prover).
- To be most successful, a theorem-prover implementation of names should be based on axioms modelled on EZFA, and not on ZF.

[10] This being a mathematics paper we need make no apology for slipping in an idealised model: a paper containing n definitions where the ith definition has complexity i for $1 \leq i \leq n$ will require n equivariance proofs by concrete calculations each of length roughly i for $1 \leq i \leq n$. We observe that $\Sigma_{1 \leq i \leq n} i$ is order n^2.

[11] This is not a critisism of FM *per se*. And since the paper was written in collaboration, this is also not a critisism of my coauthor.

If the implementation is a set theory it should probably *be* EZFA; if it is a (simple) type theory then it will need to assume a primitive type of atoms along with suitable axioms including (**Equivar**), or an oracle, or at least a universally applicable automated tactic.[12]

3.6. **Five ways to not understand equivariance.** We now try to head off some of the ways in which equivariance has been misunderstood:

Remark 3.16. (1) $\mathtt{a}_1, \ldots, \mathtt{a}_n$ from Figure 2 must mention *all* the variables mentioned in the predicate ϕ.

It is not the case that $x = y$ if and only if $\pi{\cdot}x = y$ in general; however, it is the case that $x = y$ if and only if $\pi{\cdot}x = \pi{\cdot}y$, and similarly for $y \in x$.

(2) If we extend our term language with constants, then we need to extend (**Equivar**) sensibly.

For instance, if we introduce a constant $c \in \mathbb{A}$ and blindly extend the axiom-scheme (**Equivar**) then we will create a contradiction, because it is not the case that $x = c$ if and only if $\pi(x) = c$.

Instead, we should extend the axiom-scheme (**Equivar**) so that—using the notation from Notation 3.1 and 3.3—$\pi{\cdot\cdot}c = \pi{\cdot}c$. This is essentially how Notation 3.1(2) works.

In other instances we might consider using a variable instead of a constant, or making our constant be a function over the nonequivariant choice of atoms concerned ...or perhaps this is all a sign that really c belongs in a different datatype, like $\mathbb{N}$.

(3) *Equivariant* constants where $\pi{\cdot}c = c$ is natural and desired, such as $\mathbb{A}$, $\mathbb{N}$, or 0, are fine.

Most constants of practical interest are equivariant, the only exception being when for convenience we want to add all elements of a model to our language, as we do in Definition 2.11.

(4) Theorem 3.6 does not imply that if $\mathfrak{M}$ is a model of ZFA then every $x \in \mathfrak{M}$ is equivariant.

It implies that every $x \in \mathfrak{M}$ that we can reference explicitly using a closed pure term, is equivariant. That is a very different assertion.

(5) Theorem 3.6 does not imply that if $\mathfrak{M}$ is a model of ZFA then every $x \in \mathfrak{M}$ has finite support (see [GP01] or Subsection 4.3).

Indeed, $\mathfrak{M}$ can and in general will contain elements with finite support, infinite support, or no sensible notion of support at all. For examples see Exercise 4.6.

Remark 3.17. We can extend our term language with a *unique choice* term-former $\iota\mathtt{a}.\phi$ with an axiom-scheme

$$(\exists!\mathtt{a}.\phi) \Rightarrow \phi(\iota\mathtt{a}.\phi).$$

The (**Equivar**) axiom-scheme extends smoothly in this case.

However, we do need to be sensible if we extend our term-language with arbitrary choice $\epsilon\mathtt{a}.\phi$ (also called *Hilbert's epsilon*) with axiom-scheme

$$(\exists\mathtt{a}.\phi) \Rightarrow \phi(\epsilon\mathtt{a}.\phi)$$

because we can then write a closed term such as $\epsilon\mathtt{a}.(\mathtt{a} \in \mathbb{A}) \in \mathbb{A}$.

In this case we need to extend (**Equivar**) to the language with ϵ in such a way that any nonequivariant choices made by ϵ are respected, for instance by setting $\pi{\cdot\cdot}\epsilon\mathtt{a}.\phi = \pi{\cdot}\epsilon\mathtt{a}.\phi$.[13]

[12] Seasoned with axioms to taste of course, such as Replacement or Choice. The critical point is that equivariance must have constant cost: if we use Theorem 3.6 instead of (**Equivar**) then we *must* include an automated tactic such that use of the Theorem becomes functionally equivalent to invoking an axiom (**Equivar**). Cost must *not* scale up with the complexity of the predicate ϕ.

[13] The version of Isabelle/ZF used during my PhD wielded ϵ with the joy of a toddler with a biro and a white sofa. The modern version seems more refined, and uses of ϵ seem to have been minimised.

4. Relative consistency, and freshness

4.1. **Relative consistency of EZFAC.** We prove that EZFAC is consistent relative to ZF. This is a known result but aside from a sketch in [Gab01] this has not been spelled out in the nominal literature:

Definition 4.1. Suppose $\mathfrak{M}$ is a model of ZFA. Write $\mathsf{ON}_{\mathfrak{M}}$ for the class of $\mathfrak{M}$-ordinals, or (assuming $\mathfrak{M}$ is fixed) just ON for short. This can be taken to be the least collection in $\mathfrak{M}$ such that:

(1) ON is **transitive**, meaning that if $\alpha \in \mathsf{ON}$ and $\alpha' \in \alpha$ then $\alpha' \in \mathsf{ON}$.
(2) If U is a transitive subset of ON, then $U \in \mathsf{ON}$.

Theorem 4.2. (1) *ZFAC is consistent relative to ZFC.*
(2) *As a corollary, EZFAC is consistent relative to ZFC.*

Proof. The corollary follows from part 1 of this result using Theorem 3.6, so we now prove part 1 of this result.

Assume some model $\mathfrak{M}$ of ZFC. We will use this to build a model $\mathfrak{N}$ of ZFAC. Note that by Theorem 3.6 $\mathfrak{N}$ will also be a model of EZFAC.

The elements of $\mathfrak{N}$ will be a proper class in $\mathfrak{M}$, so the rest of this proof will be conducted inside $\mathfrak{M}$.

Let x, y, and z range over elements of $\mathfrak{M}$, and let $i, j \in \{0,1\} \in \mathfrak{M}$, and recall the construction of ordered pairs $(x,y) \in \mathfrak{M}$ from Definition 2.13.

Define relations $\dot\in$ and $\dot\subseteq$ on $\mathfrak{M}$ as follows:

- $(x,1) \mathrel{\dot\in} (y,0)$ is false.
- $(x,0) \mathrel{\dot\in} (y,1)$ when $(x,0) \in y$.
- $(x,1) \mathrel{\dot\in} (y,1)$ when $(x,1) \in y$.
- $(x,i) \mathrel{\dot\subseteq} (y,j)$ when for every (z,k) with $z \in \mathfrak{M}$ and $k \in \{0,1\}$, if $(z,k) \mathrel{\dot\in} (x,i)$ then $(z,k) \mathrel{\dot\in} (y,j)$.

Then we define a class $\mathfrak{N} \subseteq \mathfrak{M}$ inductively as follows:

$$\begin{array}{l}\mathfrak{N}_0 = \{(n,0) \mid n \in \mathbb{N}\}\\ \mathfrak{N}_\alpha = \{(x,1) \mid (x,1) \mathrel{\dot\subseteq} \bigcup_{\alpha'<\alpha} \mathfrak{N}_\alpha\} \quad \alpha \in ON\\ \mathfrak{N} = \bigcup_{\alpha \in ON} \mathfrak{N}_\alpha\end{array}$$

Intuitively every element in $\mathfrak{N}$ is tagged with 0 or 1, where 0 indicates '$\mathfrak{N}$ believes that I am an atom' and 1 indicates '$\mathfrak{N}$ believes that I am a set'.

$\mathfrak{N}$ with $\dot\in$ extends to a model of the syntax from Definition 2.1 as follows:

- $\mathfrak{N} \vDash x \doteq y$ when $x = y$.
- $\mathfrak{N} \vDash \forall \mathtt{a}.\phi$ when $\mathfrak{N} \vDash \phi[\mathtt{a}{:=}x]$ for every $x \in \mathfrak{N}$.
- $\{x,y\}_{\mathfrak{N}} = (\{x,y\},1)$.
- $pset_{\mathfrak{N}}(y) = (\{x \in \mathfrak{N} \mid x \mathrel{\dot\subseteq} y\},1)$.
- $\bigcup_{\mathfrak{N}} x = (\{x' \in \mathfrak{N} \mid x' \mathrel{\dot\in} x\},1)$.
- $\{\mathtt{a} \in y \mid \phi\}_{\mathfrak{N}} = (\{x \mathrel{\dot\in} y \mid \mathfrak{N} \vDash \phi\},1)$.
- $\varnothing_{\mathfrak{N}} = (\varnothing,1)$.
- $\mathbb{A}_{\mathfrak{N}} = (\{(n,0) \mid n \in \mathbb{N}\},1)$.

It is routine to check that this satisfies the axioms of ZFAC from Figure 1. In particular, $\mathfrak{N}$ satisfies (**Choice**) because the ambient universe $\mathfrak{N}$ does—we just make choices on tagged sets. □

4.2. **On the group of permutations.** In Definition 2.14 we took a *permutation* to be any bijection on $\mathbb{A}$. But in fact, choosing what should and should not be a permutation is a non-trivial decision. Here is a sample of options:

(1) A permutation π is *any* bijection on atoms.
    This is the simplest option. Note that 'any' should be interpreted within the model, so it is perfectly compatible for a permutation to be any bijection on atoms, and also all permutations are finitely supported (see next point).
(2) A permutation π is a bijection on atoms such that $nontriv(\pi) = \{a{\in}\mathbb{A} \mid \pi(a) \neq a\}$ is finite (...or countable, and so forth; choose your favourite cardinality restriction). We call such a π **finitely supported**.
(3) Fix some total ordering $\leq$ on $\mathbb{A}$. Then a permutation is a bijection on atoms that *respects* $\leq$, meaning that $a \leq b$ if and only if $\pi(a) \leq \pi(b)$.
    This notion is particularly important in a computational context, where we may wish to assume that atoms are orderable.
(4) Fix a partition $\mathbb{A} = \mathbb{A}^{<} \cup \mathbb{A}^{>}$ where $\mathbb{A}^{<}$ and $\mathbb{A}^{>}$ are larger than finite (we can fill in our favourite cardinality constraint here; e.g. countable). Then a permutation is a bijection π on atoms such that $nontriv(\pi) \setminus \mathbb{A}^{>}$ is finite (countable). This is the basis of *permissive-nominal* techniques as used for instance in [DG12a] and is helpful for quantifying over nominal terms style unknowns.
(5) Fix a $\mathbb{Z}$-indexed family $(S_k \mid k \in \mathbb{Z})$ where $S_k \subseteq \mathbb{A}$ and $S_k \subseteq S_{k+1}$ for every $k \in \mathbb{Z}$, and all of the following sets have equal cardinality: S_k, $\mathbb{A} \setminus S_k$, and $S_{k+j} \setminus S_k$ for every $j \geq 1$.
    Then a permutation is a bijection π on atoms such that $nontriv(\pi) \subseteq S_k$ for some $k \in \mathbb{Z}$.
    This notion of permutation has many useful properties, for instance there exist π such that $\#nontriv(\pi) = \#\mathbb{A}$ and for any π, π' we have that $\#(\mathit{fix}(\pi) \cap \mathit{fix}(\pi')) = \#\mathbb{A}$. More on this in [Gab16].

So there is no one right answer to the question of 'What is a permutation?'. It depends on what we wish to accomplish.

4.3. **Support and freshness.** Aside from equivariance, another notable feature of nominal techniques is *support* and *freshness*. We give a concise account tailored to the 'classic' nominal case where all permutations are finite:

4.3.1. *The basic definition.*

Notation 4.3. If $A \subseteq \mathbb{A}$ write

$$\mathit{fix}(A) = \{\pi \in Perm \mid \forall a{\in}A.\pi(a) = a\}.$$

Recall the pointwise atoms-permutation action from Definition 2.16:

Definition 4.4. (1) Say that $K \subseteq \mathbb{A}$ **supports** an element x when $\forall \pi{\in}\mathit{fix}(K).\pi{\cdot}x = x$.
    We may write

$$K\$x \quad \text{for} \quad \text{'}K \text{ supports } x\text{'}.$$

(2) If x is K-supported for some finite $K \subseteq \mathbb{A}$ then call x **(finitely) supported** and say that x has **finite support**.
(3) Finally we define $K\#x$—in words: K is **fresh for** x—when

$$K\#x \quad \text{for} \quad \exists K'{\subseteq}\mathbb{A}.(K'\$x \wedge K \cap K' = \varnothing).$$

Taking $K = \{a\}$, we might write $\{a\}\#x$ as $a\#x$.

(4) If x is supported by $\varnothing$, so that $\pi{\cdot}x = x$ for every permutation π, then call x **equivariant**. Otherwise call x **non-equivariant**.

Remark 4.5. • Equivariance does not mean that for every model $\mathfrak{M}$ of (E)ZFAC and every $x \in \mathfrak{M}$, x must be equivariant in the sense of Definition 4.4(4); this is simply a false inference.
• Furthermore, just because $X \in \mathfrak{M}$ is equivariant in the sense of Definition 4.4(4) does not imply that every $x \in X$ is equivariant. We have:

Exercise 4.6. (1) Using Definition 2.16, prove that $\mathbb{A}$ is equivariant (that is, $\pi{\cdot}\mathbb{A} = \mathbb{A}$), but every $x \in \mathbb{A}$ is non-equivariant.
(2) Find an X such that X is equivariant, but every $x \in X$ does not have finite support.[14]

Lemmas 4.7 collects the most important corollaries of Definition 4.4:

Lemma 4.7. *Suppose X is an element and $K \subseteq \mathbb{A}$ is a set of atoms. Then:*
(1) *X is equivariant if and only if $\forall x.\forall \pi{\in}Perm.\big(x{\in}X \Leftrightarrow \pi{\cdot}x \in X\big)$.*
(2) *$K\$X$ if and only if $\forall x.\forall \pi{\in}fix(K).\big(x{\in}X \Leftrightarrow \pi{\cdot}x \in X\big)$.*
Suppose $A \subseteq \mathbb{A}$ supports some element x. Then:
(3) *If $\pi \in fix(A)$ then $\pi{\cdot}x = x$.*
(4) *If $\pi(a) = \pi'(a)$ for every $a{\in}A$ then $\pi{\cdot}x = \pi'{\cdot}x$.*

Proof. By routine calculations from Definition 4.4. □

Proposition 4.8. *Suppose $\mathfrak{M}$ is a model of ZFA and $x \in \mathfrak{M}$ has finite support.*
(1) *If $C, C' \subseteq_{fin} \mathbb{A}$ both support x, then $C \cap C'$ supports x.*
(2) *The* **support** *of x defined by*

$$supp(x) = \bigcap\{C \subseteq_{fin} \mathbb{A} \mid C\$x\}$$

is the unique least finite set of atoms supporting x.

Proof. See e.g. Theorem 2.21 of [Gab11]. □

We can now give a nice illustration of equivariance:

Lemma 4.9. • *$\pi{\cdot}supp(x) = supp(\pi{\cdot}x)$.*
• *$K\$x$ if and only if $(\pi{\cdot}K)\$(\pi{\cdot}x)$.*
• *$a\#x$ if and only if $\pi(a)\#(\pi{\cdot}x)$.*

Proof. From (**Equivar**)/Theorem 3.6. □

Remark 4.10. We discussed in Subsection 3.4 how (**Choice**) from Figure 1 is compatible with (**Equivar**) from Figure 2. There is an incompatibility, but it is not with (**Equivar**).

Consider the nominal **finite support axiom** that every element has small support, which we can write in a natural notation as follows:

(**Fresh**) $\qquad \forall \mathtt{a}.\exists \mathtt{b} \subseteq_{fin} \mathbb{A}.\mathtt{b}\$\mathtt{a}.$

If we introduce (**Fresh**) as an axiom to be satisfied by all elements then this conflicts with (**Choice**). For example, a choice function from $pset(\mathbb{A})$ to $\mathbb{A}$ will not have finite support. We discuss this next:

[14] *Hint:* order.

4.3.2. ($\mathbf{Fresh}$) *as a well-behavedness property.* ($\mathbf{Fresh}$) is a *native ZFA* property: it is best approached from within a ZFA universe. ($\mathbf{Fresh}$) is typically presented in the literature as an axiom, for example:

- axiom ($\mathbf{Fresh}$) just before Definition 4.4 in [GP01],
- the *finite support* axiom in Definition 1(ii) in [Pit03],
- the axiom ($\mathbf{Fresh}$) in Figure 2 of [Gab11], and
- Definition 2.2 of [Pit13]

(stated for finite support in [GP01, Pit03, Pit13] and for possibly infinite support in [Gab11]).

However, it is sometimes preferable to present ($\mathbf{Fresh}$) not as an axiom of Fraenkel-Mostowski set theory but instead as a *well-behavedness* or *niceness* property of the larger (E)ZFA universe, alongside other well-behavedness properties such as 'being countable', 'being computable', 'being a closed set in a topological space', and so forth. Thus, from the point of view of EZFA, being supported is just a property that some elements have and others do not.

There are two reasons for this:

(1) Even if every element we intend to work with will satisfy ($\mathbf{Fresh}$), presenting ($\mathbf{Fresh}$) as a well-behavedness property does no harm and may be beneficial. To see why, consider that no paper on number theory has ever been rejected because

> "number theory assumes that everything has to be a number and therefore countable, which is easily contradicted by an easy Gödel diagonalisation argument".

But this is only because we can rely on readers having a background understanding that "assume arithmetic" does not mean "...and reject any mention of infinite sets". Yet by presenting ($\mathbf{Fresh}$) as an axiom we open ourselves to giving a precisely analogous impression, and if by chance we talk momentarily about a non-supported set (meaning no more by this than if a number theorist mentions 'the set of even numbers'), then we risk giving the appearance of inconsistency.

This is not necessary: we can start with EZFA, treat ($\mathbf{Fresh}$) as a well-behavedness property, and then everything can proceed rigorously, in a single foundation, and at little to no cost. Had we done this in [GP01] it might have saved some trouble.[15]

(2) We sometimes require non-supported elements, and furthermore being supported is not a hereditary sets property

Two recent papers [Gab14, GG17] are mostly concerned with certain collections of supported sets whose elements do not have support. These are 'nominal-flavoured' papers, but their foundation is definitely EZFAC and not FM. They are native EZFAC papers, not native FM papers.

There are also pertinent sets in the ZFA universe that have finite support even though their elements do not have finite support, and conversely there are sets that do not have finite support even though all their elements do. For example:

- 'The set of all well-orderings of atoms' is supported by $\varnothing$, since if O is a well-ordering on atoms then so is $\pi{\cdot}O$. However, no well-ordering of $\mathbb{A}$ has finite support.
- A well-ordering of $\mathbb{A}$, write it $\leq$, encoded as a set of ordered pairs so that $\leq = \{(m,n) \in \mathbb{A}^2 \mid m \leq n\}$, does not have finite support even though every element of it does ((m,n) is supported by $\{m,n\}$).

[15] Larry Paulson once told me that in the early days of Isabelle he had trouble getting papers on theorem-provers published because referees said "Computers are finite, so mechanised mathematics can only ever talk about finite sets and this is obviously too limited to be useful". Critisisms of nominal techniques in general and ($\mathbf{Fresh}$) in particular have often had a similar nature, but this does not mean we should walk into the trap and keep asking for more.

Even if the reader's next paper uses an FM sets foundation, which would be perfectly reasonable, then this paper may serve as a reference to which to send the reader to explain why FM sets is a sensible foundation, what larger universe it naturally embeds in, and how using nominal techniques is compatible with the Axiom of Choice.

4.4. **PNL to HOL.** Nominal techniques model names using ZFA-style atoms. Another model of names is functional arguments in higher-order logic (**HOL**). An extended study of the relationship between nominal-style names and HOL-style functional arguments is in [DG12b]. See specifically Subsection 2.6, Lemma 4.17, and the Conclusion (Section 9) of [DG12b]. In the Conclusions to that paper we wrote:

> *The translation ... can only raise n variables, in order [so] we lose the equivariance (name symmetry) ... This matters because in losing symmetry we lose what makes nominal techniques so distinctive. So although we show how to translate a complete 'nominal' proof to a complete 'HOL' proof, we also see how the way in which nominal and HOL proofs are manipulated and combined, are different.*

In very brief overview, a conclusion of [DG12b] is that a key difference between modelling names as ZFA atoms, and modelling names as functional arguments, is *precisely* that the nominal representation gives us equivariance.

The mathematics in [DG12b] is extensive and outside the scope of this paper, but the sufficiently dedicated reader might find parts of that paper interesting as further reading.

5. Conclusions

We have introduced EZFAC (Definition 2.3), explored its relationship and relative consistency with ZFC sets and FM sets (Section 4), and given our time and consideration to a key item of the nominal toolbox: equivariance. Definitions and extended discussion are in the body of the paper.

Equivariance formalises that

> *atoms are distinguishable, but interchangable*.

As noted in Remark 2.18 this gives atoms a dual character: individually atoms refer to themselves, but collectively they behave like variables, via the action of permutations. This is a subtle notion which readers frequently struggle with.

In summary:

(1) We can present equivariance as a theorem of ZFA or FM (as in Theorem 3.6), or as an axiom-scheme (as in $(\mathbf{Equivar})$ in Figure 2). This is what the 'E' in EZFA represents.

In rigorous but informal mathematics—that is, in the written arguments which appear in a publication such as this paper—there is no practical difference between writing "by equivariance (Theorem 3.6)" or "by $(\mathbf{Equivar})$" *except* from the point of view of reaching out to readers whose intuitions are ZF-shaped.

If a reader's starting intuitions are informed only by ZF, then this can make Theorem 3.6 difficult to perceive. There may be benefits in clarity to taking atoms by explicit definition to be elements that populate $\mathbb{A}$ and that satisfy $(\mathbf{Equivar})$. See Remarks 3.11 and 3.14.

(2) In a theorem-prover, the cost of writing "by equivariance (Theorem 3.6)" scales at least linearly with the complexity of the predicate, and though this can be ameliorated using automated tactics it is hard to make that cost fully disappear for the user. Using the axiom-scheme reduces the cost of equivariance to 1. See Remarks 3.12 and 3.13.

(3) Nominal techniques introduce several tools, including nominal atoms-abstraction, support conditions, and equivariance.

At least for my style of nominal techniques, this list is in increasing order of practical importance: equivariance is the most important and often-used, followed by support conditions, followed by atoms-abstraction and related constructs.

(4) Fraenkel-Mostowski set theory (FM) is often proposed as a foundation for nominal techniques (this may be done implicitly, via the Schanuel Topos, the category of Nominal Sets, or similar). Comparing and contrasting them:
- FM does not have (**Choice**) and assumes (**Fresh**) as an axiom.
- EZFAC has (**Choice**) and treats (**Fresh**) as a well-behavedness condition.

FM is a useful and important environment and, to be clear, this paper should not be read as an argument against it. Yet anything that can be done in FM can be done easily in EZFAC, and as a sets foundation for nominal techniques, EZFAC merits serious consideration. It is compatible with (**Choice**), and it is relevantly and usefully more general: for instance [Gab14, DG12b, Gab16] explicitly require an EZFAC, not an FM, foundation. See Subsection 4.3.2. And even if we use FM, the fact that it embeds in EZFAC is an important observation.

References

[BKL14] Mikołaj Bojańczyk, Bartek Klin, and Sławomir Lasota, *Automata theory in nominal sets*, Logical Methods in Computer Science **10** (2014).

[CCM17] Simon Huber Cyril Cohen, Thierry Coquand and Anders Mörtberg, *Cubical type theory: A constructive interpretation of the univalence axiom*, IFCoLog Journal of Logic and its Applications **4** (2017), no. 10.

[DG12a] Gilles Dowek and Murdoch J. Gabbay, *Permissive Nominal Logic (journal version)*, Transactions on Computational Logic **13** (2012), no. 3.

[DG12b] _____, *PNL to HOL: from the logic of nominal sets to the logic of higher-order functions*, Theoretical Computer Science **451** (2012), 38–69.

[FG07] Maribel Fernández and Murdoch J. Gabbay, *Nominal rewriting (journal version)*, Information and Computation **205** (2007), no. 6, 917–965.

[Gab01] Murdoch J. Gabbay, *A Theory of Inductive Definitions with alpha-Equivalence*, Ph.D. thesis, University of Cambridge, UK, March 2001.

[Gab07] _____, *Fresh Logic*, Journal of Applied Logic **5** (2007), no. 2, 356–387.

[Gab09] _____, *Nominal Algebra and the HSP Theorem*, Journal of Logic and Computation **19** (2009), no. 2, 341–367.

[Gab11] _____, *Foundations of nominal techniques: logic and semantics of variables in abstract syntax*, Bulletin of Symbolic Logic **17** (2011), no. 2, 161–229.

[Gab12a] _____, *Finite and infinite support in nominal algebra and logic: nominal completeness theorems for free*, Journal of Symbolic Logic **77** (2012), no. 3.

[Gab12b] _____, *Meta-variables as infinite lists in nominal terms unification and rewriting*, Logic Journal of the IGPL **20** (2012), no. 6, 967–1000.

[Gab14] _____, *Stone duality for First-Order Logic: a nominal approach*, HOWARD-60. A Festschrift on the Occasion of Howard Barringer's 60th Birthday, Easychair books, 2014, pp. 178–209.

[Gab16] _____, *Consistency of Quine's New Foundations using nominal techniques*, Submitted. See arXiv preprint arxiv.org/abs/1406.4060, 2016.

[GC11] Murdoch J. Gabbay and Vincenzo Ciancia, *Freshness and name-restriction in sets of traces with names*, Foundations of software science and computation structures, 14th International Conference (FOSSACS 2011), Lecture Notes in Computer Science, vol. 6604, Springer, 2011, pp. 365–380.

[GG17] Murdoch J. Gabbay and Michael J. Gabbay, *Representation and duality of the untyped lambda-calculus in nominal lattice and topological semantics, with a proof of topological completeness*, Annals of Pure and Applied Logic **168** (2017), 501–621.

[GM08] Murdoch J. Gabbay and Aad Mathijssen, *Capture-Avoiding Substitution as a Nominal Algebra*, Formal Aspects of Computing **20** (2008), no. 4-5, 451–479.

[GP99] Murdoch J. Gabbay and Andrew M. Pitts, *A New Approach to Abstract Syntax Involving Binders*, Proceedings of the 14th Annual Symposium on Logic in Computer Science (LICS 1999), IEEE Computer Society Press, July 1999, pp. 214–224.

[GP01] _____, *A New Approach to Abstract Syntax with Variable Binding*, Formal Aspects of Computing **13** (2001), no. 3–5, 341–363.

[HMT85] Leon Henkin, J. Donald Monk, and Alfred Tarski, *Cylindric algebras*, North Holland, 1971 and 1985, Parts I and II.

[Jec06] Thomas Jech, *Set theory*, Springer, 2006, Third edition.

[Joh87] Peter T. Johnstone, *Notes on logic and set theory*, Cambridge University Press, 1987.

[KKOT15] Bartek Klin, Eryk Kopczynski, Joanna Ochremiak, and Szymon Torunczyk, *Locally finite constraint satisfaction problems*, Proceedings of the 30th Annual IEEE Symposium on Logic in Computer Science (LICS 2015), 2015, pp. 475–486.

[KMPS15] Dexter Kozen, Konstantinos Mamouras, Daniela Petrişan, and Alexandra Silva, *Nominal Kleene coalgebra*, International Colloquium on Automata, Languages, and Programming, Springer, 2015, pp. 286–298.

[KMS17] Dexter Kozen, Konstantinos Mamouras, and Alexandra Silva, *Completeness and incompleteness in nominal Kleene algebra*, Journal of Logical and Algebraic Methods in Programming (2017).

[LV12] Jordi Levy and Mateu Villaret, *Nominal unification from a higher-order perspective*, Transactions on Computational logic (TOCL) **13** (2012), no. 2.

[Pit01] Andrew M. Pitts, *Nominal logic: A first order theory of names and binding*, Proc. 4th Int'l Symposium on Theoretical Aspects of Computer Software (TACS 2001) (N. Kobayashi and B. C. Pierce, eds.), Lecture Notes in Computer Science, vol. 2215, Springer, 2001, pp. 219–242.

[Pit03] _____, *Nominal logic, a first order theory of names and binding*, Information and Computation **186** (2003), no. 2, 165–193.

[Pit13] _____, *Nominal sets: Names and symmetry in computer science*, Cambridge University Press, May 2013.

[Uni13] The Univalent Foundations Program, *Homotopy type theory: Univalent foundations of mathematics*, `https://homotopytypetheory.org/book`, Institute for Advanced Study, 2013.

[UPG04] Christian Urban, Andrew M. Pitts, and Murdoch J. Gabbay, *Nominal Unification*, Theoretical Computer Science **323** (2004), no. 1–3, 473–497.

[Urb08] Christian Urban, *Nominal reasoning techniques in Isabelle/HOL*, Journal of Automatic Reasoning **40** (2008), no. 4, 327–356.