Consistency of Quine's NF using nominal techniques

Murdoch J. Gabbay

December 9, 2014

Introduction

Many to the Midlands Logic Seminar for inviting me here today. Thank you all for coming.

Naive set theory has one rule; naive sets comprehension:

- If ϕ is a predicate, $\{a \mid \phi(a)\}$ is a set (the a such that ϕ). "Everything is a set, and a sets comprehension is a set."

This is inconsistent. Russell's famous 1901 paradox:

$$\{a \mid a \notin a\} \in \{a \mid a \notin a\} \quad \Leftrightarrow \quad \{a \mid a \notin a\} \notin \{a \mid a \notin a\}$$

Solutions

Various solutions proposed:

- Zermelo-Fraenkel set theory (ZF sets).
  Familiar to me as a PhD student as e.g. the category of sets, or Isabelle/ZF, and so on. 'Proved' consistent by the von Neumann cumulative hierarchy model; $\varnothing$, $\mathit{powerset}(\varnothing)$, ...
- Type Theory.
  Familiar as Higher-Order Logic, ML, and so on. 'Proved' consistent by taking sets and function-sets; ι, ι^ι, $(\iota^\iota)^\iota$, ι^{ι^ι}, ...
- Quine's New Foundations (NF).

Why NF is nice

NF is pretty.

- It admits a universal set: $\{a \mid \top\}$, the set of all sets, is a set.
- My favourite NF trick: model 2 as 'the set of all two-element sets' (due to Frege, 1884).
- Nicer than the standard, brutal, efficient, ZF model: $2 = \{\varnothing, \{\varnothing\}\}$.
- Nicer even than the Church numeral at type α: $2_\alpha = \lambda f{:}\alpha{\to}\alpha.\lambda x{:}\alpha.f(f(x))$.

The specification of NF is almost as concise as that of naive sets:

- If ϕ is a stratifiable predicate, then $\{a \mid \phi(a)\}$ is a set. "Everything is a set, and stratifiable sets comprehension is a set."

Stratifiability

ϕ is stratifiable when there exists an assignment of a level to its variables such that:

- If $a{=}b$ appears in ϕ then $\mathit{level}(a) = \mathit{level}(b)$.
- If $a{\in}b$ appears in ϕ then $\mathit{level}(a){+}1 = \mathit{level}(b)$.

$a \notin a$ is unstratifiable; this blocks the comprehension of Russell's paradox.

How to semantically interpret stratifiability is part of the mystery of proving NF's consistency.

(cf. Thomas Forster's notion of stratimorphism. I will propose my own interpretation later in this talk.)

NF-the-theory versus NF-the-universe

I am not myself an NF-iste. I use Zermelo-Fraenkel set theory and Fraenkel-Mostowksi set theory for my proofs.

To me, NF is a logical theory to prove things about, not a universe to prove things in.

Thomas Forster and Randall Holmes are examples of NF-istes; they live and work in NF. I was exposed to this via Thomas, who taught me logic in university.

NF is important as a foundation with a universal set, and for its mysterious and fascinating stratifiability condition. Inconsistency of NF would tell us interesting things about the viability of having a universal set.

So consistency is an important question.

Proof engineering arguments to take NF seriously

ZF (and type theory) suffer from size issues — classes, hierarchies of universes, cardinality restrictions, and so forth.

Type theory needs polymorphism (e.g. to define 'one' number 2). But polymorphism is like cocaine: you always want more; then life gets really complicated and you easily end up worse than when you began.

In theorem-provers (such as COQ) hierarchies of universes reappear as a proof-engineering issue.

NF has its own peculiarities, but it's a sweet spot.

Note: NFU (NF with urelemente) can be proved consistent. But we lose extensionaliity; not everything is a set.

About the proof

My consistency proof is in two halves:

- Sections 1 to 4 are standard computer science: define datatype of syntax (more on this later); do inductive proofs. The designs may be subtle but it's still just inductions.
- Sections 5 to 8 contain the crazy sets stuff: notably amgis-algebras and points.
  Amgis-algebras are modelled after previous topological duality results. Points are modelled after filters.

High-level tour of the paper

I will not transmit a full technical account of the paper.

I will talk at a very high level about how the proof works.

I urge you to download and read my paper. I'd like to think it's reasonably well-written, and if at some point I failed to be clear then do let me know, and I'll fix it.

The more eyes on this thing, the better.

Induction/coinduction

Consider some sets comprehension $\{a \mid \phi\}$.

- This has inductive structure, since ϕ is syntax.
- It has coinductive structure where $\{a \mid \phi\} \rightarrow \{b \mid \psi\}$ when $\{b \mid \psi\} \in \{a \mid \phi\}$.

So a model of NF sets can be viewed as finding a solution, write it NF, to this inductive/coinductive equation:

$$SyntaxFormers(NF) \longrightarrow NF \longrightarrow Powerset(NF)$$

NF is difficult because the right-hand side loops back to the left-hand side: everything is a set—including the set's behaviour!

This raises size issues which we deal with shortly. But there is another problem, with variables: we may need to dereference a variable, and we have no control over what gets put into it.

See (σ**elta**) of Figure 1 of my paper.

Peek into the paper: internal syntax

$$x ::= \mathtt{atm}(a) \mid [a]X$$
$$X ::= \mathtt{and}(\mathcal{X}) \mid \mathtt{neg}(X) \mid \mathtt{all}[a]X \mid \mathtt{elt}(x, a)$$

An internal set x is either

- a variable symbol $\mathtt{atm}(a)$ (think 'a')
- a comprehension $[a]X$ (think '$\{a \mid \phi\}$').

An internal predicate X is either a conjunction ($\mathcal{X}$ is a finite possibly empty set of X), a negation, a universal quantification, or has the form $x \in a$.

This is a syntax of normal forms under the rewrite rule

$$s \in \{a \mid \phi\} \to \phi[a{:=}s].$$

Define substitution ...

Peek into the paper: the sigma-action on syntax

$$
\begin{array}{lll}
(\sigma\mathtt{and}) & & \mathtt{and}(\mathcal{X})[a{\mapsto}x] = \mathtt{and}(\{X[a{\mapsto}x] \mid X{\in}\mathcal{X}\}) \\
(\sigma\mathtt{neg}) & & \mathtt{neg}(X)[a{\mapsto}x] = \mathtt{neg}(X[a{\mapsto}x]) \\
(\sigma\mathtt{all}) & b\#x \Rightarrow & (\mathtt{all}[b]X)[a{\mapsto}x] = \mathtt{all}[b](X[a{\mapsto}x]) \\
(\sigma\mathtt{eltatm}) & a\#y, x \Rightarrow & \mathtt{elt}(y,a)[a{\mapsto}\mathtt{atm}(n)] = \mathtt{elt}(y[a{\mapsto}\mathtt{atm}(n)], n) \\
(\sigma\mathtt{elt}\mathbf{a}) & & \mathtt{elt}(y,a)[a{\mapsto}[a']X] = X[a'{\mapsto}y[a{\mapsto}[a']X]] \\
(\sigma\mathtt{elt}\mathbf{b}) & & \mathtt{elt}(y,b)[a{\mapsto}x] = \mathtt{elt}(y[a{\mapsto}x], b) \\
(\sigma[]) & c\#x \Rightarrow & ([c]X)[a{\mapsto}x] = [c](X[a{\mapsto}x]) \\
(\sigma\mathbf{a}) & & \mathtt{atm}(a)[a{\mapsto}x] = x \\
(\sigma\mathbf{b}) & & \mathtt{atm}(b)[a{\mapsto}x] = \mathtt{atm}(b)
\end{array}
$$

Permutative convention: a, b, c over distinct atoms.

In $(\sigma\mathtt{elt}\mathbf{a})$, stratification ensures definition of substitution is inductive. X may be larger than y, but a' must have lower level than a.

Intuition of stratifiability

Imagine that each time we dereference a variable, it costs us one stratifiability dollar.

My intuition for stratifiability is that ϕ comes with a number n of stratifiability dollars (equal to the difference between the highest- and lowest-level variables in a stratification of ϕ).

When we dereference a variable, we spend one dollar and n is decremented.

Eventually we run out of dollars and then we are trapped on the left-hand side, in syntax. We can no longer afford to look up variables, and we can reason inductively with variables as a base case.

See this happen in Lemma 4.9 and Proposition 5.12.

The fundamental equation: Sets = [$\mathbb{A}$]Predicates

NF has a universal set, so we might try to construct a model such that $X = \mathit{powerset}(X)$ (so that $X \in X$).

This raises size issues. We exploit nominal atoms-abstraction instead.

Sets comprehension is just a binder:

$$\{a \mid \phi\} \quad \forall a.\phi \quad \lambda a.t \quad \int f(a)\,da$$

So our semantics solves the equation:

$$\begin{array}{ll} \mathsf{NF} & = [\mathbb{A}]\mathsf{SemanticsOfPredicates} \\ [\![\{a \mid \phi\}]\!] & = [a][\![\phi]\!] \end{array}$$

$[a]$- is atoms-abstraction (cf my PhD). No size issues: if $\mathcal{X}$ is an infinite nominal set then $[\mathbb{A}]\mathcal{X}=\{[a]x \mid a\in\mathbb{A},\ x\in\mathcal{X}\}$ has same size as $\mathcal{X}$.

SemanticsOfPredicates

Avoid size issues using atoms-abstraction. Guarantee induction using stratifiability.

What does SemanticsOfPredicates look like?

Digression: a filter is a set of predicates that is consistent and deductively closed. Recall standard completeness proofs by giving first-order logic (FOL) predicates semantics as sets of filters.

We do the same here: $[\![\phi]\!] \in$ SemanticsOfPredicates is a set of filters, for a suitable non-evident notion of filter (this notion also looks a lot like a valuation environment, for a suitable non-evident notation of value).

So:

- $[\![\phi]\!] \in$ SemanticsOfPredicates is a set of filters.
- $[\![\{a \mid \phi\}]\!] = [a][\![\phi]\!] \in$ SemanticsOfSets.

Logic in nominal powersets

How to interpret logic in a nominal powerset?

Conjunction and negation correspond to sets intersection and complement, as usual. Sets membership becomes substitution:

$$[\![\{b \mid \psi\} \in \{a \mid \phi\}]\!] = [\![\phi[a{:=}\{b \mid \psi\}]]\!].$$

What about quantification $[\![\forall a.\phi]\!]$?

It is known from previous work that if $\mathcal{X}$ is a nominal set with a substitution action, then so is $\mathit{powerset}(\mathit{powerset}(\mathcal{X}))$ (cf. topological duality results for FOL and λ-calculus).

Sketch of sigma and amgis

Suppose $\mathcal{X}$ has a σ-action $x[a \mapsto u]$ and suppose $p \in \mathit{powerset}(\mathcal{X})$ and $X \in \mathit{powerset}(\mathit{powerset}(\mathcal{X}))$. Then we write:

$$x \in p[u \leftarrow\!\shortmid a] \Leftrightarrow x[a \mapsto u] \in p$$
$$p \in X[a \mapsto u] \Leftrightarrow p[u \leftarrow\!\shortmid a] \in X$$

(More on amgis in another talk; see references at end.)

$\mathbf{[}\phi\mathbf{]}$ is a set of filters, and filters are (almost) sets of predicates, so $\mathbf{[}\phi\mathbf{]} \in \mathit{powerset}(\mathit{powerset}(\text{Predicates}))$.

Predicates ϕ have a σ-action (substitution), thus so does $\mathbf{[}\phi\mathbf{]}$.

Consistency proof works by unfolding what that σ-action has to be.

Logic in nominal powersets

Thus general nominal abstract nonsense allows us to write the following:

$$[\![\forall a.\phi]\!] = \bigcap_u [\![\phi]\!][a \mapsto u]$$

In fact, an important lemma is that

$$[\![\phi]\!][a \mapsto u] = [\![\phi[a{:=}u]]\!].$$

See Lemma 5.30.

For this to be well-defined (i.e. for u to not be 'larger' than ϕ) requires notions of internal value and quantifier depth. Intuitively, u ranges over quantifier-free internal sets.

See Subsections 5.1 & 5.2.

Logic in nominal sigma-powersets

The technical jargon:

Predicates are a σ-algebra ('set with substitution'). Sets of predicates form a dual amgis-algebra. Sets of sets of predicates restore the original σ-algebra structure.

Fact: σ-powersets of $\mathfrak{o}$-algebras naturally interpret first-order logic with equality.

Furthermore, using nominal lattices and fresh-finite limits we can identify a subset of the full powerset that is necessary for modelling logic—avoids size issues.

We need a notion of quantifier depth to retain inductive reasoning principles.

In summary:

- Sets comprehension is modelled in both syntax and semantics by nominal atoms-abstraction.
- Semantics use the sigma-powerset $powerset(powerset(Syntax))$.
- This set is large, but using nominal lattice theory we can identify a small 'logical' subclass of it.
- Nominal topological duality give this logical structure off-the-shelf: conjunction, negation, substitution, and quantification.
- Stratifiability guarantees we can only 'flip' between syntax and semantics finitely often ('is an element of').
- Quantifier depth guarantees inductive reasoning in the presence of universal quantification ('for all').

Applications:

A hard but elementary proof of the consistency of NF.

The proof is subtle but systematic: inductions, duality/filter construction, and novel but off-the-shelf material on sigma-powersets.

A concrete interpretation of stratifiability. Perhaps more could be made of this.

This is not only about NF. There's a basket of techniques here; what else could it be applied to?

References

- Consistency of Quine's NF:
  Consistency of Quine's NF using nominal techniques.
  Get it at `gabbay.org.uk/papers.html#conqnf`.
- Logic in nominal sigma-powersets:
  Semantics out of context: nominal absolute denotations for first-order logic and computation.
  Get it at `gabbay.org.uk/papers.html#semooc`.

I also recommend two less detailed but more accessible documents:

- **What sequent quantifier rules tell us about nominal semantics for logic.**
  Get it at `gabbay.org.uk/talks.html`.
- **Nominal semantics for predicate logic**.
  Get it at `gabbay.org.uk/papers.html#nomspl`.