Nominal Terms, Existential Variables, and Mathematics

Murdoch J. Gabbay

May 4, 2004

Nominal Terms

To manipulate syntax, e.g. Logic, Unification, or Rewriting, it is useful to have abstract syntax with names and binding. E voilà Nominal Terms:

Terms	$t ::= * \mid a \mid \pi \cdot X \mid \langle t, t \rangle \mid \mathsf{c}t \mid [a]t.$
Permutations	$\pi ::= \mathbf{Id} \mid (a\,b) \circ \pi$

$a, b, \ldots \in \mathbb{A}$ are atoms, they behave (almost) like constant symbols of ground type. c are constructors. Swappings act $(a\ b)(n)$ as

$$(a\ b)(a) \stackrel{\text{def}}{=} b \qquad (a\ b)(b) = a \quad \text{and} \quad (a\ b)(c) = c\ (c \neq a, b).$$

and this action extends elementwise to permutations π.

$$(a\ b) \cdot n = (a\ b)(n) \quad (a\ b) \cdot \mathsf{c}t = \mathsf{c}(a\ b) \cdot t \quad (a\ b) \cdot * = *$$
$$(a\ b) \cdot \langle s, t \rangle = \langle (a\ b) \cdot s, (a\ b) \cdot t \rangle \quad (a\ b) \cdot [n]t = [(a\ b)(n)](a\ b)t$$
$$(a\ b) \cdot (\pi \cdot X) = (a\ b) \circ \pi \cdot X.$$

Substitution

$$(\pi \cdot X)[X \mapsto s] \;=\; \pi \cdot s \qquad (\pi \cdot Y)[X \mapsto s] \;=\; \pi \cdot Y$$

$$\langle t, t' \rangle[X \mapsto s] \;=\; \langle t[X \mapsto s], t'[X \mapsto s] \rangle$$

$$(\mathsf{c}t)[X \mapsto s] \;=\; \mathsf{c}(t[X \mapsto s])$$

$$([a]t)[X \mapsto s] \;=\; [a](t[X \mapsto s]) \qquad a[X \mapsto s] \;=\; a$$

For example,

$$\langle (a\,b) \cdot X, X \rangle[X \mapsto a] \equiv \langle b, a \rangle \qquad ([a]X)[X \mapsto a] \equiv a.$$

$\equiv$ denotes syntactic identity.

Expressivity

1. Programming: $\texttt{Lambda}[a]t$, $\texttt{App}\langle t, t'\rangle$. Write $\lambda a.t$ and tt'.

2. Logic: $\texttt{All}[a]t$, $\texttt{Exist}[a]t$, $\texttt{Imp}\langle t, t'\rangle$, ... Similarly, $\forall a.\, t$.

Proof that $\lambda a.\lambda b.ab =_\alpha \lambda b.\lambda a.ba$:

$$\dfrac{\dfrac{}{a\#\lambda a.ba} \qquad \dfrac{\dfrac{a =_\alpha a \quad b =_\alpha b}{ab =_\alpha ab}}{\lambda b.ab =_\alpha (b\ a)\cdot(\lambda a.ba) \equiv \lambda b.ab}}{\lambda a.\lambda b.ab =_\alpha \lambda b.\lambda a.ba}$$

What's this?

α-equality and freshness

$$\frac{a\#s_1 \quad \cdots \quad a\#s_n}{a\#\langle s_1,\ldots,s_n\rangle} \quad \frac{a\#s}{a\#\mathsf{c}s} \quad \frac{a\#s}{a\#[b]s} \quad \frac{}{a\#b} \quad \frac{}{a\#[a]s} \quad \frac{\pi^{\text{-1}}(a)\#X}{a\#\pi\cdot X}$$

$$\frac{s_1 \underset{\alpha}{=} t_1 \quad \cdots \quad s_n \underset{\alpha}{=} t_n}{\langle s_1,\ldots,s_n\rangle \underset{\alpha}{=} \langle t_1,\ldots,t_n\rangle} \qquad \frac{s \underset{\alpha}{=} t}{\mathsf{c}\,s \underset{\alpha}{=} \mathsf{c}\,t} \qquad \frac{}{a \underset{\alpha}{=} a} \qquad \frac{t \underset{\alpha}{=} t'}{t' \underset{\alpha}{=} t}$$

$$\frac{s \underset{\alpha}{=} t}{[a]s \underset{\alpha}{=} [a]t} \qquad \frac{a\#t \qquad s \underset{\alpha}{=} (a\ b)\cdot t}{[a]s \underset{\alpha}{=} [b]t} \qquad \frac{ds(\pi,\pi')\#X}{\pi\cdot X \underset{\alpha}{=} \pi'\cdot X}$$

$$ds(\pi,\pi') \stackrel{\text{def}}{=} \{n \mid \pi(n) \neq \pi'(n)\}.$$

For example, $ds((a\ b),\mathbf{Id}) = \{a,b\}$. Compare with same-variable flex-flex case in Higher-Order Patterns $X\,as = X\,bs$.

Simple logic

We have a simple logic of freshness and α-equality.

Let a freshness context be a (possibly empty) list of assertions of the form $a\#X$. Write $\Gamma \vdash a\#t$ when $a\#t$ may be deduced using elements of Γ as assumptions.

Let a equality problem be $s \underset{\alpha}{=} t$. Similarly write $\Gamma \vdash s \underset{\alpha}{=} t$.

Lemma: $\Gamma \vdash a\#t$ and $\Gamma \vdash s \underset{\alpha}{=} t$ is decidable.

Proof: By the structural nature of the rules.

Simple algorithm for the logic

Let a unification problem $\mathcal{U}$ be a list of freshness and equality problems. Logically simplify problems according to the rules described, $\mathcal{U} \rightsquigarrow \mathcal{U}'$. If no simplification is possible say the problem is stuck.

Lemma: Problem reduction $\rightsquigarrow$ is strongly normalising and confluent.
Proof: By the purely structural nature of the rules.

Lemma: The only problems in a stuck unification problem are of the form $a\#X$, $\pi \cdot X \underset{\alpha}{=} t$, and $t \underset{\alpha}{=} \pi \cdot X$, where X does not appear in t.
Proof: By consideration of the rules.

Of course a stuck problem is precisely the context necessary to deduce the original problem.

Matching, Unification, MGUs

- Freshness simplification: $a\#X,\ \mathcal{U} \overset{a\#X}{\rightsquigarrow} \mathcal{U}$.
- Matching simplification:
  $\pi \cdot X \underset{\alpha}{=} t,\ \mathcal{U} \overset{X \mapsto \pi^{-1} \cdot t}{\rightsquigarrow} \mathcal{U}[X \mapsto \pi^{-1} \cdot t]$.
- Unification simplification:
  $t \underset{\alpha}{=} \pi \cdot X,\ \mathcal{U} \overset{X \mapsto \pi^{-1} \cdot t}{\rightsquigarrow} \mathcal{U}[X \mapsto \pi^{-1} \cdot t]$.

A solution to $\mathcal{U}$ is a context Γ of $a\#X$ and θ a substitution, such that $\Gamma \vdash P\theta$ for every $P \in \mathcal{U}$.

Theorem: The algorithm implicit above gives most general solutions (MGUs). (Matching, Unification)

Proof: In [“Nominal Unification”, with Urban and Pitts].

For example

1. $[a]X \underset{\alpha}{=} [b][a]ba$ logically simplifies to $X \underset{\alpha}{=} [b]ab$, then matching simplifies to the empty problem emitting the substitution $X \mapsto [b]ab$.

2. $[a]X \underset{\alpha}{=} [b]X$ logically simplifies to $a\#X$ and $X \underset{\alpha}{=} (a\,b) \cdot X$ and logically simplifies further to $a\#X$ and $b\#X$. This freshness reduces to the empty problem emitting the freshness context $a\#X$ and $b\#X$.

3. *More examples. . .*

Extensions of nominal terms

Let's build a logic from these pieces. Terms are as before. Formulae are:

$$F ::= \bot \mid F \wedge F \mid F \vee F \mid F \Rightarrow F \mid \exists a.\, F \mid \forall a.\, F$$
$$\mid s \underset{\alpha}{=} t \mid a\#t \mid p\, t$$

Here p are predicate atoms.

We can express:

- $\forall a.\, a\#X \Rightarrow p\, X$ "p holds of X if it is closed".
- $\forall n.\, ((n \underset{\alpha}{=} a \vee n \underset{\alpha}{=} b) \Rightarrow \bot) \Rightarrow n\#X$ "$fv(X) \subseteq \{a, b\}$".
- $\forall a.\, a\#X \Rightarrow a\#Y$ "$fv(Y) \subseteq fv(X)$".
- $\forall a.\, a\#X \Rightarrow \text{rewrites}(\langle X, Y\rangle, \langle Y, Y\rangle)$ "if the first element of the pair is closed, rewrite as shown".

rewrites is a predicate atom.

Extensions of nominal terms

We would expect some theorems to hold:

- Weakening. Admissible rule: $\dfrac{\Gamma \vdash C}{\Gamma, P \vdash C}$
- Equality. $s \underset{\alpha}{=} t \wedge a\#s \Rightarrow a\#t$ should succeed for any a, s, t.
- Equality again. $X \underset{\alpha}{=} Y \wedge a\#X \Rightarrow a\#Y$ should be a theorem.
- Substitution. Admissible rule: $\dfrac{\Gamma \vdash C}{\Gamma[X \mapsto t] \vdash C[X \mapsto t]}$

  X is not a variable symbol! It is a term.

  E.g. admissibility of this rule is a corollary of weakening and equalities, since we can weaken with $X \underset{\alpha}{=} t$.
- $\forall a.\, \exists b.\, p\langle a, b\rangle \Rightarrow \exists b.\, \forall a.\, t\langle a, b\rangle$ should fail.
- Cut-elimination, . . .

First-Order Logic rules

$$\frac{\Gamma, P, Q \vdash C}{\Gamma, P \wedge Q \vdash C} \qquad \frac{\Gamma \vdash P \qquad \Gamma \vdash Q}{\Gamma \vdash P \wedge Q}$$

$$\frac{\Gamma, P \vdash Q}{\Gamma \vdash P \Rightarrow Q} \qquad \frac{\Gamma \vdash P \quad \Gamma, Q \vdash C}{\Gamma, P \Rightarrow Q \vdash C} \qquad \frac{}{\Gamma, P \vdash P} \qquad \frac{}{\Gamma, \bot \vdash C}$$

$$\frac{\bigwedge_{a \in S} \Big(\Gamma \vdash P[n \mapsto a] \Big)}{\Gamma \vdash \forall n.\, P} \qquad \frac{\Gamma, P \vdash C}{\Gamma, \forall a.\, P \vdash C}$$

$$\frac{\Gamma \vdash P \qquad \Gamma, P \vdash Q}{\Gamma \vdash Q} \qquad \frac{\Gamma, P, P \vdash C}{\Gamma, P \vdash C}$$

Freshness rules

$$\frac{\Gamma, a\#t, a\#t' \vdash C}{\Gamma, a\#\langle t, t'\rangle \vdash C} \quad \frac{\Gamma \vdash a\#t, a\#t'}{\Gamma \vdash a\#\langle t, t'\rangle} \quad \frac{\Gamma, a\#t \vdash C}{\Gamma, a\#\mathsf{c}\, t \vdash C} \quad \frac{\Gamma \vdash a\#t}{\Gamma \vdash a\#\mathsf{c}\, t}$$

$$\frac{\Gamma, a\#[a]t \vdash C}{\Gamma \vdash C} \qquad \frac{\Gamma, a\#t \vdash C}{\Gamma, a\#[b]t \vdash C} \qquad \frac{\Gamma \vdash a\#t}{\Gamma \vdash a\#[b]t}$$

$$\frac{\Gamma, \pi^{-1}(a)\#X \vdash C}{\Gamma, a\#\pi \cdot X \vdash C} \qquad \frac{\Gamma \vdash \pi^{-1}(a)\#X}{\Gamma \vdash a\#\pi \cdot X}$$

α-equality rules   I of II

$$\frac{\Gamma, * \underset{\alpha}{=} * \vdash C}{\Gamma \vdash C} \qquad \frac{\Gamma, a \underset{\alpha}{=} a \vdash C}{\Gamma \vdash C} \qquad \frac{}{\Gamma, a \underset{\alpha}{=} b \vdash C}$$

$$\frac{\Gamma, s \underset{\alpha}{=} t \vdash C}{\Gamma, [a]s \underset{\alpha}{=} [a]t \vdash C} \qquad \frac{\Gamma \vdash s \underset{\alpha}{=} t}{\Gamma \vdash [a]s \underset{\alpha}{=} [a]t}$$

$$\frac{\Gamma, a\#t, s \underset{\alpha}{=} (a\ b) \cdot t \vdash C}{\Gamma, [a]s \underset{\alpha}{=} [b]t \vdash C} \qquad \frac{\Gamma \vdash a\#t \qquad \Gamma \vdash s \underset{\alpha}{=} (a\ b) \cdot t}{\Gamma \vdash [a]s \underset{\alpha}{=} [b]t}$$

$$\frac{\Gamma \vdash ds(\pi,\pi')\#X}{\Gamma \vdash \pi\cdot X =_{\alpha} \pi'\cdot X} \qquad \frac{\Gamma, ds(\pi,\pi')\#X \vdash C}{\Gamma, \pi\cdot X =_{\alpha} \pi'\cdot X \vdash C}$$

$$\frac{\Gamma[X\mapsto\pi^{-1}\cdot t] \vdash C[X\mapsto\pi^{-1}\cdot t]}{\Gamma, t =_{\alpha} \pi\cdot X \vdash C} \qquad (X \notin t)$$

$$\frac{\Gamma[X\mapsto\pi^{-1}\cdot t] \vdash C[X\mapsto\pi^{-1}\cdot t]}{\Gamma, \pi\cdot X =_{\alpha} t \vdash C} \qquad (X \notin t)$$

($\langle t, t'\rangle$ rules omitted to save space)

Compact reformulation of $\underset{\alpha}{=}$ and $\#$ rules; definitions style

$$a\#\langle t,t'\rangle \equiv a\#t \wedge a\#t' \qquad a\#\mathsf{c}t \equiv a\#t \qquad a\#[a]t \equiv \top$$

$$a\#[b]t \equiv a\#t \qquad a\#\pi\cdot X \equiv \pi^{-1}(a)\#X$$

$$* \underset{\alpha}{=} * \equiv \top \qquad a \underset{\alpha}{=} a \equiv \top \qquad a \underset{\alpha}{=} b \equiv \bot$$

$$[a]s \underset{\alpha}{=} [b]t \equiv a\#t \wedge s \underset{\alpha}{=} (a\,b)\cdot t \qquad a \underset{\alpha}{=} \langle t,t'\rangle \equiv a \underset{\alpha}{=} t \wedge a \underset{\alpha}{=} t'$$

$$\frac{\Gamma[X\mapsto\pi^{-1}\cdot t] \vdash C[X\mapsto\pi^{-1}\cdot t]}{\Gamma, t \underset{\alpha}{=} \pi\cdot X \vdash C} \qquad (X \notin t)$$

$$\frac{\Gamma[X\mapsto\pi^{-1}\cdot t] \vdash C[X\mapsto\pi^{-1}\cdot t]}{\Gamma, \pi\cdot X \underset{\alpha}{=} t \vdash C} \qquad (X \notin t)$$

Cut elimination

Theorem: Cut is admissible in the system without it.

Proof: By lots of lemmas.

The spirit of the underlying technical is that the equality rules together implement a Miller-Tiu-style equality 'rule':

$$\frac{\bigwedge_{\theta \,:\, s\theta =_\alpha t\theta} \left(\Gamma\theta \vdash C\theta\right)}{\Gamma,\ s =_\alpha t \vdash C}$$

Here θ varies over closing substitutions so $s\theta =_\alpha t\theta$ is a proof in the simple logic of equality and freshness.

Expressivity

1. Closure and explicit control of free variables: As already commented, e.g. $\forall n.\,(n \underset{\alpha}{=} a \Rightarrow \bot) \Rightarrow n\#a$, or $\forall n.\, n\#X \Rightarrow n\#Y$.

2. Predicate atoms: Add binary predicate atom $?$ and definitions

$$a?\langle t, t'\rangle \equiv (a?t \wedge a\#t') \vee (a\#t \wedge a?t')$$

$$a?\mathsf{c}\ t \equiv a?t \qquad a?[a]t \equiv \bot$$

$$a?[b]t \equiv a?t \qquad a?\pi \cdot X \equiv \pi^{-1}(a)?X$$

This expresses 'occurs exactly once in'; a form of linearity.

Logical simplifications

A problem $\mathcal{U}$ is a set of sequents $\Gamma \vdash C$. Logical simplifications $\mathcal{U} \leadsto \mathcal{U}'$ are given by the sequent system.

Lemma: Logical simplifications are strongly normalising.

Proof: By the structural nature of the rules.

Logical simplifications are not confluent, because of $\vee$ and $\exists$. However in their absence I *believe* this is true.

From now on, everything is blue sky.

Other simplifications

- Freshness. $\Gamma \vdash a\#X,\ \mathcal{U} \ \overset{a\#X}{\rightsquigarrow}\ (\Gamma, a\#X) \cup \mathcal{U}$.
- Matching. $\Gamma \vdash \pi \cdot X =_\alpha t,\ \mathcal{U} \ \overset{X \mapsto \pi^{-1} \cdot X}{\rightsquigarrow}\ \Gamma \cup \mathcal{U}$.
- Unification. $\Gamma \vdash t =_\alpha \pi \cdot X,\ \mathcal{U} \ \overset{X \mapsto \pi^{-1} \cdot X}{\rightsquigarrow}\ \Gamma \cup \mathcal{U}$.

Here $\Gamma \cup \mathcal{U}$ denotes the problem containing $\Gamma,\ \Delta \vdash C$ for every $\Delta \vdash C$ in $\mathcal{U}$.

We seem to need to add Γ to get confluence.

Directions

Hypothesis: Simplifications are strongly normalising and confluent.
We can consider some cases on the board.

Hypothesis: Solving an ordinary nominal unification problem $(a\#t,\ \ s \underset{\alpha}{=} t)$ is equivalent to solving $(\emptyset \vdash a\#t,\ \ \emptyset \vdash s \underset{\alpha}{=} t)$ in this new sense.

Hypothesis: Add a binary atomic predicate $\longrightarrow$; do axioms exist that hijack the theory of equality to do matching, giving rewriting for free?

Conclusions

This logic is expressive and unknowns are first-class terms. Equality on the left is first-class substitution. Equality on the right may fail logically, but 'forcing' it gives unification.

We express relations between universal variables a, b, c and existential variables X, Y, Z. This enables us to write $\forall$-right rules *and also* the $=_\alpha$-left rules.

Miller and Tiu have ∇-quantified variables for a and ordinary variables for X. The substitution $[X \mapsto t]$ gives some flavour of Higher-Order techniques. Note we have *explicit* atoms a for which $a \neq_\alpha b$ when a and b are syntactically non-identical (c.f. definitions).

Limitations and future work: (On the board: no λ-abstraction, quantification only over atoms.)

Mathematics (set theory?)

I propose a flavour of ZFA with two sorts of urelemente; atoms a, b, c and (moderated) unknowns $\pi \cdot X, \pi \cdot Y, \pi \cdot Z$.

Substitution action is as for terms but distributes over set-$\{-\}$.

We have the following additional axioms:

$$\forall x.\, \text{И}a.\, v(x) = \emptyset \;\Rightarrow\; a\#x$$
$$\forall x.\, \text{И}X.\, X\#x$$

Mathematics (algebra)

Algebraic version: a set with a permutation action $(a\,b)$ and substitution action $[X \mapsto x]$. Properties (axioms?) include:

1. $(a\,b) \cdot (y[X \mapsto x]) = ((a\,b) \cdot y)[X \mapsto x]$
2. $y[X \mapsto X] = y$.
3. $X\#y \;\vdash\; y[X \mapsto x] = x$.
4. $X\#x' \;\vdash\; y[X{\mapsto}x][X'{\mapsto}x'] = y[X'{\mapsto}x'][X{\mapsto}x[X'{\mapsto}x']]$.

Types (briefly)

Sort(s) of atoms ν.

Base sorts s.

Data sorts $\delta ::= s \mid \delta \times \delta$.

Compound sorts $\tau ::= \nu \mid \delta \mid 1 \mid \tau \times \tau \mid [\nu]\tau$.

Nominal Terms, this time with types:

$$t ::= a_\nu, b_\nu, c_\nu, \ldots \mid (\pi \cdot (X_\delta))_\delta \mid *_1$$
$$\mid \langle t_\tau, t'_{\tau'} \rangle_{\tau \times \tau'} \mid ([a_\nu]t_\tau)_{[\nu]\tau} \mid (f_{\tau \to \delta} t_\tau)_\delta$$