TU/

# One-and-a-halfth-order Logic

Aad Mathijssen Murdoch J. Gabbay

22th June 2006

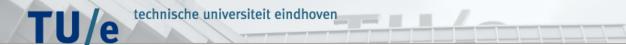
/department of mathematics and computer science



### Motivation

Consider the following valid assertions in first-order logic:

- $\bullet \ \phi \supset \psi \supset \phi$
- if  $a \not\in fn(\phi)$  then  $\phi \supset \forall a.\phi$
- if  $a \not\in fn(\phi)$  then  $\phi \supset \phi[\![a \mapsto t]\!]$
- if  $b \not\in fn(\phi)$  then  $\forall a.\phi \supset \forall b.\phi[\![a \mapsto b]\!]$



### Motivation

Consider the following valid assertions in first-order logic:

- $\bullet \ \phi \supset \psi \supset \phi$
- $\bullet \text{ if } a \not\in \mathit{fn}(\phi) \text{ then } \phi \supset \forall a.\phi$
- if  $a \not\in fn(\phi)$  then  $\phi \supset \phi[\![a \mapsto t]\!]$
- if  $b \not\in fn(\phi)$  then  $\forall a.\phi \supset \forall b.\phi \llbracket a \mapsto b \rrbracket$

These are not valid syntax in first-order logic, because of meta-level concepts:

- meta-variables *varying* over syntax:  $\phi$ ,  $\psi$ , a, b, t
- properties of syntax:  $a \not\in fn(\phi)$ ,  $\phi[\![a \mapsto t]\!]$ ,  $\alpha$ -equivalence



### Motivation

Consider the following valid assertions in first-order logic:

- $\bullet \ \phi \supset \psi \supset \phi$
- if  $a \not\in fn(\phi)$  then  $\phi \supset \forall a.\phi$
- if  $a \not\in fn(\phi)$  then  $\phi \supset \phi[\![a \mapsto t]\!]$
- if  $b \not\in fn(\phi)$  then  $\forall a.\phi \supset \forall b.\phi \llbracket a \mapsto b \rrbracket$

These are not valid syntax in first-order logic, because of meta-level concepts:

- meta-variables *varying* over syntax:  $\phi$ ,  $\psi$ , a, b, t
- properties of syntax:  $a \not\in fn(\phi)$ ,  $\phi[\![a \mapsto t]\!]$ ,  $\alpha$ -equivalence

Is there a logic in which the above assertions can be expressed directly in the syntax?



### Motivation (2)

Consider the following derivations in Gentzen's sequent calculus:

$$\frac{\overline{\psi,\phi\vdash\phi}\left(\mathbf{A}\mathbf{x}\right)}{\overline{\phi\vdash\psi\supset\phi}\left(\supset\mathbf{R}\right)}_{\vdash\phi\supset\psi\supset\phi}\left(\supset\mathbf{R}\right)$$

$$\begin{array}{c} \frac{\overline{\mathsf{p}(d),\mathsf{p}(c)\vdash\mathsf{p}(c)}\left(\mathbf{A}\mathbf{x}\right)}{\mathbf{p}(c)\vdash\mathsf{p}(d)\supset\mathsf{p}(c)}\left(\supset\mathbf{R}\right)\\ \overline{\mathsf{p}(c)\succ\mathsf{p}(d)\supset\mathsf{p}(c)}\left(\supset\mathbf{R}\right) \end{array} \\ \end{array}$$

And for  $b \not\in fn(\phi)$ :

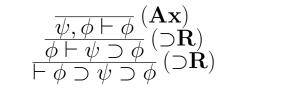
$$\frac{\overline{\forall a.\phi \vdash \forall b.\phi \llbracket a \mapsto b \rrbracket} (\mathbf{A}\mathbf{x})}{\vdash \forall a.\phi \supset \forall b.\phi \llbracket a \mapsto b \rrbracket} (\supset \mathbf{R})$$

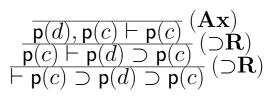
$$\frac{\forall c.\mathbf{p}(c) \vdash \forall d.\mathbf{p}(d)}{\vdash \forall c.\mathbf{p}(c) \supset \forall d.\mathbf{p}(d)} (\supset \mathbf{R})$$



### Motivation (2)

Consider the following derivations in Gentzen's sequent calculus:





And for  $b \not\in fn(\phi)$ :

$$\frac{\overline{\forall a.\phi \vdash \forall b.\phi \llbracket a \mapsto b \rrbracket} (\mathbf{A}\mathbf{x})}{\vdash \forall a.\phi \supset \forall b.\phi \llbracket a \mapsto b \rrbracket} (\supset \mathbf{R})$$

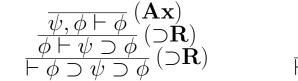
$$\frac{\forall c.\mathbf{p}(c) \vdash \forall d.\mathbf{p}(d)}{\vdash \forall c.\mathbf{p}(c) \supset \forall d.\mathbf{p}(d)} (\mathbf{A}\mathbf{x}) \\ (\supset \mathbf{R})$$

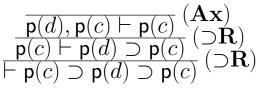
The left ones are not derivations, they are *schemas* of derivations. When p is a *specific* atomic predicate and c and d are *specific* variables, the right ones are derivations; they are *instances* of the schemas on the left.



### Motivation (2)

Consider the following derivations in Gentzen's sequent calculus:





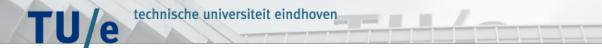
And for  $b \not\in fn(\phi)$ :

$$\frac{\overline{\forall a.\phi \vdash \forall b.\phi \llbracket a \mapsto b \rrbracket} (\mathbf{A}\mathbf{x})}{\vdash \forall a.\phi \supset \forall b.\phi \llbracket a \mapsto b \rrbracket} (\supset \mathbf{R})$$

$$\frac{\forall c.\mathbf{p}(c) \vdash \forall d.\mathbf{p}(d)}{\vdash \forall c.\mathbf{p}(c) \supset \forall d.\mathbf{p}(d)} (\mathbf{A}\mathbf{x}) \\ (\supset \mathbf{R})$$

The left ones are not derivations, they are *schemas* of derivations. When p is a *specific* atomic predicate and c and d are *specific* variables, the right ones are derivations; they are *instances* of the schemas on the left.

Is there a logic in which the derivation on the left is a derivation too?

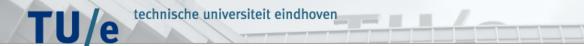


### Motivation (3)

First-order logic and its proof systems formalise *reasoning*.

But also a lot of reasoning is *about* first-order logic.

So why shouldn't that be formalised?



### Motivation (3)

First-order logic and its proof systems formalise *reasoning*.

But also a lot of reasoning is *about* first-order logic.

So why shouldn't that be formalised?

**One-and-a-halfth-order logic** does this by means of formalising:

- meta-variables
- properties of syntax

### Overview

TU/e

• Introduction to one-and-a-halfth-order logic

technische universiteit eindhoven

- Syntax of one-and-a-halfth-order logic
- Sequent calculus for one-and-a-halfth-order logic
- Axiomatisation of one-and-a-halfth-order logic
- Relation to first-order logic
- Semantics of one-and-a-halfth-order logic
- Conclusions, related and future work



### Introduction

In the syntax of one-and-a-halfth-order logic:

- Unknowns P , Q and T represent meta-variables  $\phi$  ,  $\psi$  and t.
- Atoms *a* and *b* represent meta-variables *a* and *b*.
- Freshness a # P represents  $a \notin fn(\phi)$ .
- Explicit substitution  $P[a \mapsto T]$  represents  $\phi[\![a \mapsto t]\!]$ .

# Introduction (2)

The meta-level assertions in first-order logic

 $\bullet \ \phi \supset \psi \supset \phi$ 

TU

- if  $a \not\in fn(\phi)$  then  $\phi \supset \forall a.\phi$
- if  $a \not\in fn(\phi)$  then  $\phi \supset \phi[\![a \mapsto t]\!]$
- if  $b \not\in fn(\phi)$  then  $\forall a.\phi \supset \forall b.\phi \llbracket a \mapsto b \rrbracket$

correspond to valid assertions in the syntax of one-and-a-halfth-order logic:

- $\bullet \ P \supset Q \supset P$
- $\bullet \ a \# P \to P \supset \forall [a] P$
- $a \# P \to P \supset P[a \mapsto T]$
- $b \# P \to \forall [a] P \supset \forall [b] P[a \mapsto b]$

# Introduction (3)

TU/

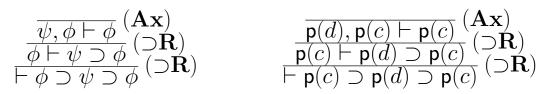
In sequent derivations of one-and-a-halfth-order logic:

- Contexts of freshnesses are added to the sequents.
- *Derivability of freshnesses* are added as side-conditions.
- Substitutional equivalence on terms is added as two derivation rules, taking care of  $\alpha$ -equivalence and substitution.

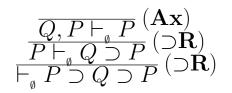
# Introduction (4)

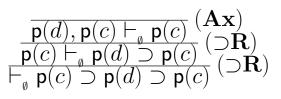
TU/e

The (schematic) derivations in first-order logic



correspond to valid derivations in one-and-a-halfth-order logic:





### Introduction (5)

TU/e

The (schematic) derivations in first-order logic, where  $b \not\in f\!n(\phi)$  ,

$$\frac{\overline{\forall a.\phi \vdash \forall b.\phi \llbracket a \mapsto b \rrbracket}}{\neg \forall a.\phi \supset \forall b.\phi \llbracket a \mapsto b \rrbracket} (\mathbf{A}\mathbf{x}) \qquad \qquad \frac{\overline{\forall c.\mathbf{p}(c) \vdash \forall d.\mathbf{p}(d)} (\mathbf{A}\mathbf{x})}{\vdash \forall c.\mathbf{p}(c) \supset \forall d.\mathbf{p}(d)} (\mathbf{\Box}\mathbf{R})$$

correspond to valid derivations in one-and-a-halfth-order logic:

$$\begin{array}{l} & \frac{\forall [a]P \vdash_{_{b\#P}} \forall [a]P}{\forall [a]P \vdash_{_{b\#P}} \forall [b]P[a \mapsto b]} \left( \mathbf{StructR} \right) \\ \hline \forall [a]P \vdash_{_{b\#P}} \forall [b]P[a \mapsto b] \left( \supset \mathbf{R} \right) \end{array} & (b\#P \vdash_{_{\mathsf{SUB}}} \forall [a]P = \forall [b]P[a \mapsto b] \right) \\ \hline \vdash_{_{b\#P}} \forall [a]P \supset \forall [b]P[a \mapsto b] \left( \supset \mathbf{R} \right) \end{aligned} \\ & \frac{\forall [c]\mathbf{p}(c) \vdash_{_{\emptyset}} \forall [c]\mathbf{p}(c)}{\forall [c]\mathbf{p}(c) \vdash_{_{\emptyset}} \forall [d]\mathbf{p}(d)} \left( \mathbf{StructR} \right) \\ \hline \forall [c]\mathbf{p}(c) \vdash_{_{\emptyset}} \forall [d]\mathbf{p}(d)} \left( (\supset \mathbf{R}) \right) \end{aligned} \quad (\emptyset \vdash_{_{\mathsf{SUB}}} \forall [c]\mathbf{p}(c) = \forall [d]\mathbf{p}(d)) \end{aligned}$$



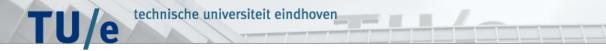
### Syntax of one-and-a-halfth-order logic

We use **Nominal Terms** to specify the syntax, since they have built-in support for:

- meta-variables
- binding
- freshness

Nominal terms allow for a *direct* and *natural* representation of systems with binding.

Nominal terms are *first-order*, not higher-order.



# Sorts

Base sorts  $\mathbb P$  for 'predicates' and  $\mathbb T$  for 'terms'.

Atomic sort  $\mathbb{A}$  for the object-level variables.

Sorts  $\tau$ :

$$\tau ::= \mathbb{P} \mid \mathbb{T} \mid \mathbb{A} \mid [\mathbb{A}]\tau$$

### Terms

TU

Atoms  $a, b, c, \ldots$  have sort  $\mathbb{A}$ ; they represent *object-level* variable symbols.

**Unknowns**  $X, Y, Z, \ldots$  have sort  $\tau$ ; they represent *meta-level* variable symbols. Let P, Q, R be unknowns of sort  $\mathbb{P}$ , and T, U of sort  $\mathbb{T}$ .

We call  $\pi \cdot X$  a **moderated unknown**. This represents the **permutation of atoms**  $\pi$  acting on an unknown term.

**Term-formers**  $f_{\rho}$  have an associated **arity**  $\rho = (\tau_1, \ldots, \tau_n)\tau$ . f :  $\rho$  means 'f with arity  $\rho$ '.

**Terms** *t*, subscripts indicate sorting rules:

$$t ::= a_{\mathbb{A}} \mid (\pi \cdot X_{\tau})_{\tau} \mid ([a_{\mathbb{A}}]t_{\tau})_{[\mathbb{A}]\tau} \mid (\mathsf{f}_{(\tau_{1},...,\tau_{n})\tau}(t_{\tau_{1}}^{1},\ldots,t_{\tau_{n}}^{n}))_{\tau}$$

Write f for f() if n = 0.

# Terms (2)

TU

Term-formers for one-and-a-halfth-order logic:

- $\bot$  : () $\mathbb{P}$  represents *falsity*
- $\supset$ :  $(\mathbb{P}, \mathbb{P})\mathbb{P}$  represents *implication*, write  $\phi \supset \psi$  for  $\supset (\phi, \psi)$ ;
- $\forall$  :  $([\mathbb{A}]\mathbb{P})\mathbb{P}$  represents universal quantification, write  $\forall [a]\phi$  for  $\forall ([a]\phi)$
- $\approx:(\mathbb{T},\mathbb{T})\mathbb{P}$  represents object-level equality, write  $t\approx u$  for  $\approx\!\!(t,u)$
- var : (A)T is *variable casting*, forced upon us by the sort system, write a for var(a)
- sub :  $([\mathbb{A}]\tau, \mathbb{T})\tau$ , where  $\tau \in \{\mathbb{T}, [\mathbb{A}]\mathbb{T}, \mathbb{P}, [\mathbb{A}]\mathbb{P}\}$ , is explicit substitution, write  $v[a \mapsto t]$  for sub([a]v, t)
- $\bullet$   $p_1,\ldots,p_n:(\mathbb{T},\ldots,\mathbb{T})\mathbb{P}$  are object-level predicate term-formers
- $\bullet~f_1,\ldots,f_m:(\mathbb{T},\ldots,\mathbb{T})\mathbb{T}$  are object-level term-formers

# Terms (3)

We may call terms  $\phi$  and  $\psi$  of sort  $\mathbb{P}$  **predicates**.

Sugar:

Descending order of operator precedence:

$$[a]\_, \ \_[\_ \mapsto \_], \ \approx, \ \{\neg, \forall, \exists\}, \ \{\land, \lor\}, \ \supset, \Leftrightarrow$$

 $\land$ ,  $\lor$ ,  $\supset$  and  $\Leftrightarrow$  associate to the right.

# Terms (3)

We may call terms  $\phi$  and  $\psi$  of sort  $\mathbb{P}$  **predicates**.

Sugar:

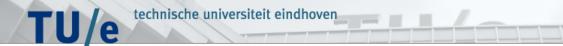
Descending order of operator precedence:

$$[a]\_, \ \_[\_ \mapsto \_], \ \approx, \ \{\neg, \forall, \exists\}, \ \{\land, \lor\}, \ \supset, \Leftrightarrow$$

 $\land$ ,  $\lor$ ,  $\supset$  and  $\Leftrightarrow$  associate to the right.

Example terms of sort  $\mathbb{P}$ :

 $P\supset Q\supset P \qquad P\supset \forall [a]P \qquad P\supset P[a\mapsto T] \qquad \forall [a]P\supset \forall [b]P[a\mapsto b]$ 



### Freshness

**Freshness (assertions)** a # t, which means 'a is fresh for t. If t is an unknown X, the freshness is called **primitive**.

A **freshness context**  $\Delta$  is a set of *primitive* freshnesses.

Example freshness contexts:

$$\emptyset \quad a \# X \quad a \# P, b \# Q$$

### Freshness

**Freshness (assertions)** a # t, which means 'a is fresh for t. If t is an unknown X, the freshness is called **primitive**.

A **freshness context**  $\Delta$  is a set of *primitive* freshnesses.

Example freshness contexts:

$$\emptyset \qquad a \# X \qquad a \# P, b \# Q$$

We call  $\Delta \rightarrow t$  a **term-in-context**. We may write t if  $\Delta = \emptyset$ .

Example terms-in-context of sort  $\mathbb{P}$ :

$$\begin{split} P \supset Q \supset P & a \# P \to P \supset \forall [a] P \\ a \# P \to P \supset P[a \mapsto T] & b \# P \to \forall [a] P \supset \forall [b] P[a \mapsto b] \end{split}$$

### Derivability of freshness

TU/e

$$\frac{\overline{a\#b}}{a\#a} (\#\mathbf{ab}) \quad \frac{\pi^{-1}(a)\#X}{a\#\pi \cdot X} (\#\mathbf{X})$$

$$\frac{\overline{a\#b}}{a\#[a]t} (\#[]\mathbf{a}) \quad \frac{a\#t}{a\#[b]t} (\#[]\mathbf{b}) \quad \frac{a\#t_1 \cdots a\#t_n}{a\#f(t_1, \dots, t_n)} (\#\mathbf{f})$$

 $\boldsymbol{a}$  and  $\boldsymbol{b}$  range over distinct atoms.

Write  $\Delta \vdash a \# t$  when there exists a derivation of a # t using the elements of  $\Delta$  as assumptions. Say that a # t is derivable from  $\Delta$ .

### Derivability of freshness

$$\overline{a\#b} (\#\mathbf{ab}) \quad \frac{\pi^{-1}(a)\#X}{a\#\pi \cdot X} (\#\mathbf{X})$$
$$\overline{a\#[a]t} (\#[]\mathbf{a}) \quad \frac{a\#t}{a\#[b]t} (\#[]\mathbf{b}) \quad \frac{a\#t_1 \cdots a\#t_n}{a\#f(t_1, \dots, t_n)} (\#\mathbf{f})$$

 $\boldsymbol{a}$  and  $\boldsymbol{b}$  range over distinct atoms.

Write  $\Delta \vdash a \# t$  when there exists a derivation of a # t using the elements of  $\Delta$  as assumptions. Say that a # t is derivable from  $\Delta$ .

Examples:

TU/

e

$$\vdash a \# b \qquad \vdash a \# \forall [a] P \qquad a \# P \vdash a \# \forall [b] P$$

### Derivability of equality

**Equality (assertions)** t = u, where t and u are of the same sort.

Nominal Algebra is the logic of equality between nominal terms.

# Derivability of equality

**Equality (assertions)** t = u, where t and u are of the same sort.

Nominal Algebra is the logic of equality between nominal terms.

Derivability:

TU/e

$$\begin{split} \overline{t = t} & (\mathbf{refl}) \quad \frac{t = u}{u = t} (\mathbf{symm}) \quad \frac{t = u \quad u = v}{t = v} (\mathbf{tran}) \\ \frac{t = u}{C[t] = C[u]} (\mathbf{cong}) \quad \frac{a \# t \quad b \# t}{(a \ b) \cdot t = t} (\mathbf{perm}) \\ \frac{\Delta^{\pi} \sigma}{t^{\pi} \sigma = u^{\pi} \sigma} (\mathbf{ax_A}) A \text{ is } \Delta \to t = u \quad \begin{bmatrix} a \# X_1, \dots, a \# X_n \end{bmatrix} \quad \Delta \\ \vdots \\ \frac{t = u}{t = u} (\mathbf{fr}) \quad (a \notin t, u, \Delta) \end{split}$$

Write  $\Delta \vdash_{\tau} t = u$  when t = u is derivable from  $\Delta$  using axioms A from T only.

### Derivability of equality (2)

TU/e

Nominal algebraic theory SUB of explicit substitution:

$$\begin{array}{ll} (\mathbf{var} \mapsto) & a[a \mapsto T] = T \\ (\# \mapsto) & a\#X \to X[a \mapsto T] = X \\ (\mathbf{f} \mapsto) & \mathbf{f}(X_1, \dots, X_n)[a \mapsto T] = \mathbf{f}(X_1[a \mapsto T], \dots, X_n[a \mapsto T]) \\ (\mathbf{abs} \mapsto) & b\#T \to ([b]X)[a \mapsto T] = [b](X[a \mapsto T]) \\ (\mathbf{ren} \mapsto) & b\#X \to X[a \mapsto b] = (b \ a) \cdot X \end{array}$$

### Derivability of equality (2)

Nominal algebraic theory SUB of explicit substitution:

$$\begin{array}{ll} (\mathbf{var} \mapsto) & a[a \mapsto T] = T \\ (\# \mapsto) & a\#X \to X[a \mapsto T] = X \\ (\mathbf{f} \mapsto) & \mathbf{f}(X_1, \dots, X_n)[a \mapsto T] = \mathbf{f}(X_1[a \mapsto T], \dots, X_n[a \mapsto T]) \\ (\mathbf{abs} \mapsto) & b\#T \to ([b]X)[a \mapsto T] = [b](X[a \mapsto T]) \\ (\mathbf{ren} \mapsto) & b\#X \to X[a \mapsto b] = (b \ a) \cdot X \end{array}$$

Examples:

TU/e

$$\begin{split} b \# P \vdash_{\text{sub}} \forall [a] P = \forall [b] P[a \mapsto b] \\ \vdash_{\text{sub}} X[a \mapsto a] = X \\ a \# Y \vdash_{\text{sub}} Z[a \mapsto X][b \mapsto Y] = Z[b \mapsto Y][a \mapsto X[b \mapsto Y]] \end{split}$$

technische universiteit eindhoven

### Sequent calculus for one-and-a-halfth-order logic

Let **(predicate) contexts**  $\Phi, \Psi$  be finite sets of predicates. Examples:

$$\emptyset \hspace{0.4cm} \phi \hspace{0.4cm} \phi, \Phi \hspace{0.4cm} \Phi, \Phi'$$

A **sequent** is a triple  $\Phi \vdash_{\Delta} \Psi$ . We may omit empty predicate contexts, e.g. writing  $\vdash_{\Delta}$  for  $\emptyset \vdash_{\Delta} \emptyset$ .

Define derivability on sequents...

TU/

# Sequent calculus (2)

TU/e

Rules resembling Gentzen's sequent calculus for first-order logic:

$$\begin{split} \overline{\phi, \Phi \vdash_{\Delta} \Psi, \phi} (\mathbf{A}\mathbf{x}) & \overline{\perp, \Phi \vdash_{\Delta} \Psi} (\bot \mathbf{L}) \\ \frac{\Phi \vdash_{\Delta} \Psi, \phi - \psi, \Phi \vdash_{\Delta} \Psi}{\phi \supset \psi, \Phi \vdash_{\Delta} \Psi} (\supset \mathbf{L}) & \frac{\phi, \Phi \vdash_{\Delta} \Psi, \psi}{\Phi \vdash_{\Delta} \Psi, \phi \supset \psi} (\supset \mathbf{R}) \\ \frac{\phi[a \mapsto t], \Phi \vdash_{\Delta} \Psi}{\forall [a] \phi, \Phi \vdash_{\Delta} \Psi} (\forall \mathbf{L}) & \frac{\Phi \vdash_{\Delta} \Psi, \psi}{\Phi \vdash_{\Delta} \Psi, \forall [a] \psi} (\forall \mathbf{R}) \quad (\Delta \vdash a \# \Phi, \Psi) \\ \frac{\phi[a \mapsto t'], \Phi \vdash_{\Delta} \Psi}{t' \approx t, \phi[a \mapsto t], \Phi \vdash_{\Delta} \Psi} (\approx \mathbf{L}) & \overline{\Phi \vdash_{\Delta} \Psi, t \approx t} (\approx \mathbf{R}) \end{split}$$

### Sequent calculus (3)

Other rules:

TU/e

$$\begin{split} \frac{\phi', \Phi \vdash_{\Delta} \Psi}{\phi, \Phi \vdash_{\Delta} \Psi} \left( \mathbf{StructL} \right) & \left( \Delta \vdash_{\mathsf{SUB}} \phi' = \phi \right) \\ \frac{\Phi \vdash_{\Delta} \Psi, \psi'}{\Phi \vdash_{\Delta} \Psi, \psi} \left( \mathbf{StructR} \right) & \left( \Delta \vdash_{\mathsf{SUB}} \psi' = \psi \right) \\ \frac{\Phi \vdash_{\Delta \sqcup \{a \# X_1, \dots, a \# X_n\}} \Psi}{\Phi \vdash_{\Delta} \Psi} \left( \mathbf{Fresh} \right) & \left( a \not\in \Phi, \Psi, \Delta \right) \\ \frac{\Phi \vdash_{\Delta} \Psi, \phi - \phi', \Phi \vdash_{\Delta} \Psi}{\Phi \vdash_{\Delta} \Psi} \left( \mathbf{Cut} \right) & \left( \Delta \vdash_{\mathsf{SUB}} \phi = \phi' \right) \end{split}$$

### Example derivations

TU/e

Derivation of  $a \# P \to P \supset \forall [a] P$ :

$$\frac{\overline{P \vdash_{a \# P} P}(\mathbf{A}\mathbf{x})}{\frac{P \vdash_{a \# P} \forall [a] P}{\vdash_{a \# P} \forall [a] P} (\forall \mathbf{R})} (a \# P \vdash a \# P) \xrightarrow{P \vdash_{a \# P} \forall [a] P} (\supset \mathbf{R})$$

Derivation of  $a \# P \to P \supset P[a \mapsto T]$ :

$$\frac{\overline{P \vdash_{a \# P} P}\left(\mathbf{Ax}\right)}{\frac{P \vdash_{a \# P} P\left[a \mapsto T\right]}{\vdash_{a \# P} P\left[a \mapsto T\right]}\left(\mathbf{StructR}\right) \quad (a \# P \vdash_{\mathsf{sub}} P = P[a \mapsto T])} \xrightarrow{\left[P \vdash_{a \# P} P \supset P[a \mapsto T\right]} (\supset \mathbf{R})$$



### Properties of the sequent calculus

We may *permute* atoms and *instantiate* unknowns in derivations.

**Theorem 1** If  $\Pi$  is a valid derivation of  $\Phi \vdash_{\Delta} \Psi$ , then  $\Pi^{\pi}$  is a valid derivation of  $\Phi^{\pi} \vdash_{\Delta^{\pi}} \Psi^{\pi}$ .

**Theorem 2** If  $\Pi$  is a valid derivation of  $\Phi \vdash_{\Delta} \Psi$  and  $\Delta' \vdash \Delta \sigma$ , then  $\Pi(\sigma, \Delta')$  is a valid derivation of  $\Phi \sigma \vdash_{\Delta'} \Psi \sigma$ .

 $\Pi(\sigma,\Delta')$  is  $\Pi$  in which:

- $\bullet$  each unknown X is replaced by  $\sigma(X)$
- $\bullet$  each freshness context  $\Delta$  is replaced by  $\Delta'$

### Properties of the sequent calculus (2)

For example,  $\Pi$  is the derivation of  $a \# P \to P \supset P[a \mapsto T]$ :

$$\frac{\overline{P \vdash_{a \# P} P} (\mathbf{Ax})}{\frac{P \vdash_{a \# P} P[a \mapsto T]}{-_{a \# P} P \supset P[a \mapsto T]} (\mathbf{StructR}) \quad (a \# P \vdash_{\mathsf{sub}} P = P[a \mapsto T])$$

Take  $\sigma = [{\bf p}(c)/P, d/T]$  and  $\Delta' = \emptyset$  , then:

technische universiteit eindhoven

•  $\Delta' \vdash \Delta \sigma$ , i.e.  $\emptyset \vdash a \# \mathbf{p}(c)$ 

TU/

•  $\Pi(\sigma, \Delta')$  is the following valid derivation of  $\mathbf{p}(c) \supset \mathbf{p}(c)[a \mapsto d]$ :

$$\begin{array}{c} \displaystyle \frac{\overline{\mathsf{p}(c) \vdash_{\scriptscriptstyle \emptyset} \mathsf{p}(c)} \left( \mathbf{A} \mathbf{x} \right) }{ \frac{\mathsf{p}(c) \vdash_{\scriptscriptstyle \emptyset} \mathsf{p}(c) [a \mapsto d]}{\mathsf{p}(c) \vdash_{\scriptscriptstyle \emptyset} \mathsf{p}(c) [a \mapsto d]} \left( \mathbf{StructR} \right) & (\emptyset \vdash_{\mathsf{sub}} \mathsf{p}(c) = \mathsf{p}(c)[a \mapsto d]) \\ \displaystyle \vdash_{\scriptscriptstyle \emptyset} \mathsf{p}(c) \supset \mathsf{p}(c)[a \mapsto d] \left( \supset \mathbf{R} \right) \end{array}$$

TU/e

# Properties of the sequent calculus (3)

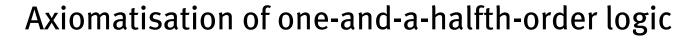
**Theorem 3** [Cut elimination] The (Cut) rule is admissible in the system without it.

TU/

## Properties of the sequent calculus (3)

**Theorem 3** [Cut elimination] The (Cut) rule is admissible in the system without it.

**Corollary 4** The sequent calculus is **consistent**, i.e.  $\vdash_{\Delta}$  can never be derived.



Theory FOL extends theory SUB with the following axioms:

technische universiteit eindhoven

TU/e

$$P \supset Q \supset P = \top \quad \neg \neg P \supset P = \top \quad \text{(Props)}$$

$$(P \supset Q) \supset (Q \supset R) \supset (P \supset R) = \top \quad \bot \supset P = \top$$

$$\forall [a] P \supset P[a \mapsto T] = \top \quad \text{(Quants)}$$

$$\forall [a] (P \land Q) \Leftrightarrow \forall [a] P \land \forall [a] Q = \top$$

$$a \# P \rightarrow \forall [a] (P \supset Q) \Leftrightarrow P \supset \forall [a] Q = \top$$

$$T \approx T = \top \quad U \approx T \land P[a \mapsto T] \supset P[a \mapsto U] = \top \quad \text{(Eq)}$$

Axioms are all of the form  $\phi = \top$ , which intuitively means ' $\phi$  is true'.

Note that this is a *finite* number of axioms.

Axiomatisation of one-and-a-halfth-order logic (2)

technische universiteit eindhoven

TU

The **conjunctive form**  $\Phi^{\wedge}$  of a predicate contexts  $\Phi$  is  $\Phi$  where we put  $\wedge$  between its elements. Analogously, define its **disjunctive form** by putting  $\vee$  between its elements. For example:

$$\emptyset^{\wedge} = \top \qquad \{\phi,\psi\}^{\wedge} = \phi \wedge \psi \qquad \emptyset^{\vee} = \bot \qquad \{\phi,\psi\}^{\vee} = \phi \vee \psi$$

technische universiteit eindhoven

TU

## Axiomatisation of one-and-a-halfth-order logic (2)

The **conjunctive form**  $\Phi^{\wedge}$  of a predicate contexts  $\Phi$  is  $\Phi$  where we put  $\wedge$  between its elements. Analogously, define its **disjunctive form** by putting  $\vee$  between its elements. For example:

$$\emptyset^{\wedge} = \top \qquad \{\phi, \psi\}^{\wedge} = \phi \wedge \psi \qquad \emptyset^{\vee} = \bot \qquad \{\phi, \psi\}^{\vee} = \phi \vee \psi$$

**Theorem 5** For all predicate contexts  $\Phi$ ,  $\Psi$  and freshness contexts  $\Delta$ :

 $\Phi \vdash_{\scriptscriptstyle \Delta} \Psi \text{ is derivable } \quad \text{iff } \quad \Delta \vdash_{\scriptscriptstyle \mathsf{FOL}} \Phi^{\wedge} \,\supset\, \Psi^{\vee} \,=\, \top.$ 

So sequent and equational derivability are equivalent.

technische universiteit eindhoven

TU

## Axiomatisation of one-and-a-halfth-order logic (2)

The **conjunctive form**  $\Phi^{\wedge}$  of a predicate contexts  $\Phi$  is  $\Phi$  where we put  $\wedge$  between its elements. Analogously, define its **disjunctive form** by putting  $\vee$  between its elements. For example:

$$\emptyset^{\wedge} = \top \qquad \{\phi,\psi\}^{\wedge} = \phi \wedge \psi \qquad \emptyset^{\vee} = \bot \qquad \{\phi,\psi\}^{\vee} = \phi \vee \psi$$

**Theorem 5** For all predicate contexts  $\Phi$ ,  $\Psi$  and freshness contexts  $\Delta$ :

 $\Phi \vdash_{\scriptscriptstyle \Delta} \Psi \text{ is derivable } \quad \text{iff } \quad \Delta \vdash_{\scriptscriptstyle \mathsf{FOL}} \Phi^{\wedge} \supset \Psi^{\vee} = \top.$ 

So sequent and equational derivability are equivalent.

**Corollary 6** Theory FOL is consistent, i.e.  $\Delta \vdash_{FOL} \top = \bot$  does not hold.

## Relation to First-order Logic

TU/

Call a term or a predicate context **ground** if it does not contain unknowns or explicit substitutions.

Call  $\Phi \vdash \Psi$  a **first-order sequent**, when  $\Phi$  and  $\Psi$  are ground predicate contexts.

Gentzen's sequent calculus for first-order logic:

$$\begin{array}{ccc} \overline{\phi, \ \Phi \vdash \Psi, \ \phi} \ (\mathbf{A}\mathbf{x}) & \overline{\perp, \ \Phi \vdash \Psi} \ (\bot \mathbf{L}) \\ \\ \underline{\Phi \vdash \Psi, \ \phi} \ \psi, \ \Phi \vdash \Psi \ (\supset \mathbf{L}) & \underline{\phi, \ \Phi \vdash \Psi, \ \psi} \\ \overline{\phi \supset \psi, \ \Phi \vdash \Psi} \ (\supset \mathbf{L}) & \underline{\phi, \ \Phi \vdash \Psi, \ \psi} \\ \overline{\Phi \vdash \Psi, \ \phi \supset \psi} \ (\supset \mathbf{R}) \\ \\ \underline{\phi \llbracket a \mapsto t \rrbracket, \ \Phi \vdash \Psi} \ (\forall \mathbf{L}) & \underline{\Phi \vdash \Psi, \ \phi} \\ \overline{\Phi \vdash \Psi, \ \forall a.\phi} \ (\forall \mathbf{R}) & (a \not\in fn(\Phi, \Psi)) \\ \\ \\ \underline{\phi \llbracket a \mapsto t' \rrbracket, \ \Phi \vdash \Psi} \\ \hline t' \approx t, \ \phi \llbracket a \mapsto t \rrbracket, \ \Phi \vdash \Psi \\ (\approx \mathbf{L}) & \overline{\Phi \vdash \Psi, \ t \approx t} \ (\approx \mathbf{R}) \end{array}$$

# Relation to First-order Logic (2)

Note that:

TU/

e

- $\bullet$  we write  $\forall a.\phi$  for  $\forall [a]\phi$
- $[\![a \mapsto t]\!]$  is capture-avoiding substitution
- $a \not\in fn(\phi)$  is 'a does not occur in the free names of  $\phi$  '
- $\bullet$  we take predicates up to  $\alpha\text{-equivalence}$

# Relation to First-order Logic (2)

Note that:

ΤU

- $\bullet$  we write  $\forall a.\phi$  for  $\forall [a]\phi$
- $[\![a \mapsto t]\!]$  is capture-avoiding substitution
- $a \not\in fn(\phi)$  is 'a does not occur in the free names of  $\phi$ '
- $\bullet$  we take predicates up to  $\alpha\text{-equivalence}$

**Theorem 7**  $\Phi \vdash \Psi$  is derivable in the sequent calculus for first-order logic, iff  $\Phi \vdash_{\theta} \Psi$  is derivable in the sequent calculus for one-and-a-halfth-order logic.

So on ground terms, one-and-a-halfth-order logic *is* first-order logic.

TU

For closed terms t, its **ground form** t[[]] is t in which each explicit substitution  $v[a \mapsto u]$  is replaced by  $v[[a \mapsto u]]$  bottom-up in the syntax.

**Theorem 8** For closed terms t,  $\vdash_{SUB} t = t$ 

technische universiteit eindhoven

Call a substitution  $\sigma$  closing for a term t if  $t\sigma$  is closed.

A term-in-context  $\Delta \to \phi$  is **valid** iff for all closing substitutions  $\sigma$  (for  $\phi$ ) for which  $\vdash \Delta \sigma$  holds,  $\phi \sigma$  [] is valid in the semantics of first-order logic.

TU

For closed terms t, its **ground form** t[[]] is t in which each explicit substitution  $v[a \mapsto u]$  is replaced by  $v[[a \mapsto u]]$  bottom-up in the syntax.

**Theorem 8** For closed terms t,  $\vdash_{SUB} t = t[[]]$ 

technische universiteit eindhoven

Call a substitution  $\sigma$  closing for a term t if  $t\sigma$  is closed.

A term-in-context  $\Delta \to \phi$  is **valid** iff for all closing substitutions  $\sigma$  (for  $\phi$ ) for which  $\vdash \Delta \sigma$  holds,  $\phi \sigma$  [] is valid in the semantics of first-order logic.

The sequent calculus for one-and-a-halfth-order logic is **sound** for this semantics:

**Theorem 9** If  $\vdash_{\Delta} \phi$  is derivable then  $\Delta \rightarrow \phi$  is valid.



TU

Using nominal terms, we can:

• *accurately* represent systems with binding: e.g. explicit substitution and first-order logic

technische universiteit eindhoven

• specify *novel* systems with their own mathematical interest: e.g. one-and-a-halfth-order logic

One-and-a-halfth-order logic:

- makes meta-level concepts of first-order logic *explicit*
- has a sequent calculus with *syntax-directed* rules
- has a *semantics* in first-order logic
- has a *finite* equational axiomatisation
- is the *result* of axiomatising first-order logic in nominal algebra

#### Related work

In **second-order logic (SOL)** we can quantify over predicates *anywhere*, which makes it more expressive than one-and-a-halfh-order logic.

On the other hand, we can easily extend theory FOL with *one* axiom to express the principle of induction on natural numbers:

$$P[a \mapsto 0] \land \forall [a] (P \supset P[a \mapsto succ(a)]) \supset \forall [a] P = \top A$$

**Higher-order logic (HOL)** is type raising, while our logic is *not*:

- $P[a \mapsto t]$  corresponds to f(t) in HOL, where  $f: \mathbb{T} \to \mathbb{P}$
- $P[a \mapsto t][a' \mapsto t']$  corresponds to f'(t)(t') where  $f' : \mathbb{T} \to \mathbb{T} \to \mathbb{P}$

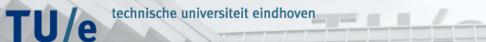
• ...

One-and-a-halfth-order logic is not a subset of SOL or HOL because of freshnesses.



#### Future work

- Completeness of the sequent calculus with respect to the semantics.
- Let unknowns range over *sequent derivations*, and establish a Curry-Howard correspondence (term-in-contexts as types, derivations as terms).
- Two-and-a-halfth-order logic (where you can abstract X)?
- Implementation and automation?



#### Future work

- Completeness of the sequent calculus with respect to the semantics.
- Let unknowns range over *sequent derivations*, and establish a Curry-Howard correspondence (term-in-contexts as types, derivations as terms).
- Two-and-a-halfth-order logic (where you can abstract X)?
- Implementation and automation?

### **Current status**

- M.J. Gabbay, A.H.J. Mathijssen, Nominal Algebra, submitted STACS'07.
- M.J. Gabbay, A.H.J. Mathijssen, Capture-avoiding Substitution as a Nominal Algebra, submitted ICTAC'06.
- M.J. Gabbay, A.H.J. Mathijssen, One-and-a-halfth-order Logic, PPDP'06.

#### Just to scare you

TU/e

$$\frac{P[b \mapsto c][a \mapsto c] \vdash_{c \# P} P[b \mapsto c][a \mapsto c]}{\forall [a]P[b \mapsto c] \vdash_{c \# P} P[b \mapsto c][a \mapsto c]} (\mathsf{Ax}) \\
\frac{\forall [a]P[b \mapsto c] \vdash_{c \# P} P[b \mapsto c][a \mapsto c]}{(\forall Ia]P)[b \mapsto c] \vdash_{c \# P} P[b \mapsto a][a \mapsto c]} (\mathsf{StructL}) \quad (I.) \\
\frac{\forall [b]\forall [a]P \vdash_{c \# P} P[b \mapsto c][a \mapsto c]}{\forall [b]\forall [a]P \vdash_{c \# P} \forall [c]P[b \mapsto c][a \mapsto c]} (\forall R) \quad (2.) \\
\frac{\forall [b]\forall [a]P \vdash_{c \# P} \forall [a]P[b \mapsto a]}{\forall [b]\forall [a]P \vdash_{c \# P} \forall [a]P[b \mapsto a]} (\mathsf{Fresh}) \quad (4.)$$

a

Side-conditions:

I. 
$$c \# P \vdash_{\mathsf{SUB}} \forall [a] P[b \mapsto c] = (\forall [a] P)[b \mapsto c]$$
  
2.  $c \# P \vdash c \# \forall [b] \forall [a] P$   
3.  $c \# P \vdash_{\mathsf{SUB}} \forall [c] P[b \mapsto c][a \mapsto c] = \forall [a] P[b \mapsto c]$   
4.  $c \notin \forall [b] \forall [a] P, \forall [a] P[b \mapsto a]$