

Miscellaneous FOL Examples

April 19, 2009

Contents

1	A simple formulation of First-Order Logic	2
1.1	Syntax	2
1.2	Propositional logic	3
1.3	Equality	4
1.4	Quantifiers	4
2	Natural numbers	4
3	Examples for the manual “Introduction to Isabelle”	6
3.0.1	Some simple backward proofs	6
3.0.2	Demonstration of <i>fast</i>	6
3.0.3	Derivation of conjunction elimination rule	6
3.1	Derived rules involving definitions	6
4	Theory of the natural numbers: Peano’s axioms, primitive recursion	7
4.1	Proofs about the natural numbers	7
5	Intuitionistic FOL: Examples from The Foundation of a Generic Theorem Prover	9
5.1	Examples with quantifiers	10
6	First-Order Logic: PROLOG examples	10
7	Intuitionistic First-Order Logic	12
7.1	de Bruijn formulae	13
7.2	Intuitionistic FOL: propositional problems based on Pelletier.	14
7.3	11. Proved in each direction (incorrectly, says Pelletier!!)	15
7.4	****Examples with quantifiers****	15
7.5	The converse is classical in the following implications...	15
7.6	The following are not constructively valid!	16
7.7	Hard examples with quantifiers	16

8 First-Order Logic: propositional examples (intuitionistic version)	19
9 First-Order Logic: quantifier examples (intuitionistic version)	22
10 Classical Predicate Calculus Problems	23
10.1 Pelletier's examples	24
10.2 Classical Logic: examples with quantifiers	25
10.3 Problems requiring quantifier duplication	26
10.4 Hard examples with quantifiers	26
10.5 Problems (mainly) involving equality or functions	30
11 First-Order Logic: propositional examples (classical version)	33
12 First-Order Logic: quantifier examples (classical version)	35
12.1 Negation Normal Form	37
12.1.1 de Morgan laws	37
12.1.2 Pushing in the existential quantifiers	37
12.1.3 Pushing in the universal quantifiers	38
13 First-Order Logic: the 'if' example	38
14 Example of Declaring an Oracle	39
14.1 Oracle declaration	39
14.2 Oracle as low-level rule	39
14.3 Oracle as proof method	39

1 A simple formulation of First-Order Logic

theory *First-Order-Logic* **imports** *Pure* **begin**

The subsequent theory development illustrates single-sorted intuitionistic first-order logic with equality, formulated within the Pure framework. Actually this is not an example of Isabelle/FOL, but of Isabelle/Pure.

1.1 Syntax

typedecl *i*
typedecl *o*

judgment
Trueprop :: *o* \Rightarrow *prop* (- 5)

1.2 Propositional logic

axiomatization

false :: o (\perp) **and**

imp :: $o \Rightarrow o \Rightarrow o$ (**infixr** \longrightarrow 25) **and**

conj :: $o \Rightarrow o \Rightarrow o$ (**infixr** \wedge 35) **and**

disj :: $o \Rightarrow o \Rightarrow o$ (**infixr** \vee 30)

where

falseE [*elim*]: $\perp \Longrightarrow A$ **and**

impI [*intro*]: $(A \Longrightarrow B) \Longrightarrow A \longrightarrow B$ **and**

mp [*dest*]: $A \longrightarrow B \Longrightarrow A \Longrightarrow B$ **and**

conjI [*intro*]: $A \Longrightarrow B \Longrightarrow A \wedge B$ **and**

conjD1: $A \wedge B \Longrightarrow A$ **and**

conjD2: $A \wedge B \Longrightarrow B$ **and**

disjE [*elim*]: $A \vee B \Longrightarrow (A \Longrightarrow C) \Longrightarrow (B \Longrightarrow C) \Longrightarrow C$ **and**

disjI1 [*intro*]: $A \Longrightarrow A \vee B$ **and**

disjI2 [*intro*]: $B \Longrightarrow A \vee B$

theorem *conjE* [*elim*]:

assumes $A \wedge B$

obtains A **and** B

$\langle proof \rangle$

definition

true :: o (\top) **where**

$\top \equiv \perp \longrightarrow \perp$

definition

not :: $o \Rightarrow o$ (\neg - [40] 40) **where**

$\neg A \equiv A \longrightarrow \perp$

definition

iff :: $o \Rightarrow o \Rightarrow o$ (**infixr** \longleftrightarrow 25) **where**

$A \longleftrightarrow B \equiv (A \longrightarrow B) \wedge (B \longrightarrow A)$

theorem *trueI* [*intro*]: \top

$\langle proof \rangle$

theorem *notI* [*intro*]: $(A \Longrightarrow \perp) \Longrightarrow \neg A$

$\langle proof \rangle$

theorem *notE* [*elim*]: $\neg A \Longrightarrow A \Longrightarrow B$

$\langle proof \rangle$

theorem *iffI* [*intro*]: $(A \Longrightarrow B) \Longrightarrow (B \Longrightarrow A) \Longrightarrow A \longleftrightarrow B$

$\langle proof \rangle$

theorem *iff1* [*elim*]: $A \longleftrightarrow B \implies A \implies B$
 $\langle proof \rangle$

theorem *iff2* [*elim*]: $A \longleftrightarrow B \implies B \implies A$
 $\langle proof \rangle$

1.3 Equality

axiomatization

equal :: $i \Rightarrow i \Rightarrow o$ (**infixl** = 50)

where

refl [*intro*]: $x = x$ **and**

subst: $x = y \implies P\ x \implies P\ y$

theorem *trans* [*trans*]: $x = y \implies y = z \implies x = z$
 $\langle proof \rangle$

theorem *sym* [*sym*]: $x = y \implies y = x$
 $\langle proof \rangle$

1.4 Quantifiers

axiomatization

All :: $(i \Rightarrow o) \Rightarrow o$ (**binder** \forall 10) **and**

Ex :: $(i \Rightarrow o) \Rightarrow o$ (**binder** \exists 10)

where

allI [*intro*]: $(\bigwedge x. P\ x) \implies \forall x. P\ x$ **and**

allD [*dest*]: $\forall x. P\ x \implies P\ a$ **and**

exI [*intro*]: $P\ a \implies \exists x. P\ x$ **and**

exE [*elim*]: $\exists x. P\ x \implies (\bigwedge x. P\ x \implies C) \implies C$

lemma $(\exists x. P\ (f\ x)) \longrightarrow (\exists y. P\ y)$
 $\langle proof \rangle$

lemma $(\exists x. \forall y. R\ x\ y) \longrightarrow (\forall y. \exists x. R\ x\ y)$
 $\langle proof \rangle$

end

2 Natural numbers

theory *Natural-Numbers* **imports** *FOL* **begin**

Theory of the natural numbers: Peano's axioms, primitive recursion. (Modernized version of Larry Paulson's theory "Nat".)

typedecl *nat*

arities $nat :: term$

axiomatization

$Zero :: nat \quad (0)$ **and**

$Suc :: nat \Rightarrow nat$ **and**

$rec :: [nat, 'a, [nat, 'a] \Rightarrow 'a] \Rightarrow 'a$

where

$induct \ [case-names \ 0 \ Suc, \ induct \ type: \ nat]:$

$P(0) \Rightarrow (!x. P(x) \Rightarrow P(Suc(x))) \Rightarrow P(n)$ **and**

$Suc-inject: Suc(m) = Suc(n) \Rightarrow m = n$ **and**

$Suc-neq-0: Suc(m) = 0 \Rightarrow R$ **and**

$rec-0: rec(0, a, f) = a$ **and**

$rec-Suc: rec(Suc(m), a, f) = f(m, rec(m, a, f))$

lemma $Suc-n-not-n: Suc(k) \neq k$

$\langle proof \rangle$

definition

$add :: [nat, nat] \Rightarrow nat \quad (\mathbf{infixl} \ + \ 60)$ **where**

$m + n = rec(m, n, \lambda x y. Suc(y))$

lemma $add-0 \ [simp]: 0 + n = n$

$\langle proof \rangle$

lemma $add-Suc \ [simp]: Suc(m) + n = Suc(m + n)$

$\langle proof \rangle$

lemma $add-assoc: (k + m) + n = k + (m + n)$

$\langle proof \rangle$

lemma $add-0-right: m + 0 = m$

$\langle proof \rangle$

lemma $add-Suc-right: m + Suc(n) = Suc(m + n)$

$\langle proof \rangle$

lemma

assumes $!!n. f(Suc(n)) = Suc(f(n))$

shows $f(i + j) = i + f(j)$

$\langle proof \rangle$

end

3 Examples for the manual “Introduction to Isabelle”

```
theory Intro
imports FOL
begin
```

3.0.1 Some simple backward proofs

```
lemma mythm:  $P \mid P \dashv\vdash P$ 
<proof>
```

```
lemma  $(P \ \& \ Q) \mid R \dashv\vdash (P \mid R)$ 
<proof>
```

```
lemma  $(\text{ALL } x \ y. \ P(x,y)) \dashv\vdash (\text{ALL } z \ w. \ P(w,z))$ 
<proof>
```

3.0.2 Demonstration of *fast*

```
lemma  $(\text{EX } y. \ \text{ALL } x. \ J(y,x) \ <-> \ \sim J(x,x))$ 
 $\dashv\vdash \ \sim (\text{ALL } x. \ \text{EX } y. \ \text{ALL } z. \ J(z,y) \ <-> \ \sim J(z,x))$ 
<proof>
```

```
lemma  $\text{ALL } x. \ P(x,f(x)) \ <->$ 
 $(\text{EX } y. \ (\text{ALL } z. \ P(z,y) \dashv\vdash P(z,f(x))) \ \& \ P(x,y))$ 
<proof>
```

3.0.3 Derivation of conjunction elimination rule

```
lemma
  assumes major:  $P \ \& \ Q$ 
  and minor:  $[\mid P; \ Q \mid] \implies R$ 
  shows  $R$ 
<proof>
```

3.1 Derived rules involving definitions

Derivation of negation introduction

```
lemma
  assumes  $P \implies \text{False}$ 
  shows  $\sim P$ 
<proof>
```

```
lemma
  assumes major:  $\sim P$ 
  and minor:  $P$ 
```

shows R
 $\langle proof \rangle$

Alternative proof of the result above

lemma
 assumes $major: \sim P$
 and $minor: P$
 shows R
 $\langle proof \rangle$

end

4 Theory of the natural numbers: Peano's axioms, primitive recursion

theory *Nat*
imports *FOL*
begin

typedecl *nat*
arities *nat* :: *term*

consts
 $0 :: nat \quad (0)$
 $Suc :: nat \Rightarrow nat$
 $rec :: [nat, 'a, [nat, 'a] \Rightarrow 'a] \Rightarrow 'a$
 $add :: [nat, nat] \Rightarrow nat \quad (\mathbf{infixl} + 60)$

axioms
 $induct: \quad [| P(0); !!x. P(x) \implies P(Suc(x)) |] \implies P(n)$
 $Suc-inject: \quad Suc(m)=Suc(n) \implies m=n$
 $Suc-neq-0: \quad Suc(m)=0 \implies R$
 $rec-0: \quad rec(0,a,f) = a$
 $rec-Suc: \quad rec(Suc(m), a, f) = f(m, rec(m,a,f))$

defs
 $add-def: \quad m+n == rec(m, n, \%x y. Suc(y))$

4.1 Proofs about the natural numbers

lemma *Suc-n-not-n*: $Suc(k) \sim = k$
 $\langle proof \rangle$

lemma $(k+m)+n = k+(m+n)$
 $\langle proof \rangle$

lemma *add-0* [*simp*]: $0+n = n$

$\langle proof \rangle$

lemma *add-Suc* [*simp*]: $Suc(m)+n = Suc(m+n)$
 $\langle proof \rangle$

lemma *add-assoc*: $(k+m)+n = k+(m+n)$
 $\langle proof \rangle$

lemma *add-0-right*: $m+0 = m$
 $\langle proof \rangle$

lemma *add-Suc-right*: $m+Suc(n) = Suc(m+n)$
 $\langle proof \rangle$

lemma
 assumes *prem*: $!!n. f(Suc(n)) = Suc(f(n))$
 shows $f(i+j) = i+f(j)$
 $\langle proof \rangle$

end

theory *Nat-Class*
imports *FOL*
begin

This is an abstract version of theory *Nat*. Instead of axiomatizing a single type *nat* we define the class of all these types (up to isomorphism).

Note: The *rec* operator had to be made *monomorphic*, because class axioms may not contain more than one type variable.

class *nat* =
 fixes *Zero* :: 'a (0)
 and *Suc* :: 'a \Rightarrow 'a
 and *rec* :: 'a \Rightarrow 'a \Rightarrow ('a \Rightarrow 'a \Rightarrow 'a) \Rightarrow 'a
 assumes *induct*: $P(0) \Longrightarrow (\bigwedge x. P(x) \Longrightarrow P(Suc(x))) \Longrightarrow P(n)$
 and *Suc-inject*: $Suc(m) = Suc(n) \Longrightarrow m = n$
 and *Suc-neq-Zero*: $Suc(m) = 0 \Longrightarrow R$
 and *rec-Zero*: $rec(0, a, f) = a$
 and *rec-Suc*: $rec(Suc(m), a, f) = f(m, rec(m, a, f))$
begin

definition
 add :: 'a \Rightarrow 'a \Rightarrow 'a (**infixl** + 60) **where**
 $m + n = rec(m, n, \lambda x y. Suc(y))$

lemma *Suc-n-not-n*: $Suc(k) \neq (k::'a)$
 $\langle proof \rangle$


```

lemma  $(k + m) + n = k + (m + n)$ 
   $\langle proof \rangle$ 

lemma add-Zero [simp]:  $0 + n = n$ 
   $\langle proof \rangle$ 

lemma add-Suc [simp]:  $Suc(m) + n = Suc(m + n)$ 
   $\langle proof \rangle$ 

lemma add-assoc:  $(k + m) + n = k + (m + n)$ 
   $\langle proof \rangle$ 

lemma add-Zero-right:  $m + 0 = m$ 
   $\langle proof \rangle$ 

lemma add-Suc-right:  $m + Suc(n) = Suc(m + n)$ 
   $\langle proof \rangle$ 

lemma
  assumes prem:  $\bigwedge n. f(Suc(n)) = Suc(f(n))$ 
  shows  $f(i + j) = i + f(j)$ 
   $\langle proof \rangle$ 

end

end

```

5 Intuitionistic FOL: Examples from The Foundation of a Generic Theorem Prover

```

theory Foundation
imports IFOL
begin

lemma  $A \& B \dashv\vdash (C \dashv\vdash A \& C)$ 
   $\langle proof \rangle$ 

  A form of conj-elimination

lemma
  assumes  $A \& B$ 
  and  $A \implies B \implies C$ 
  shows  $C$ 
   $\langle proof \rangle$ 

lemma
  assumes  $!!A. \sim \sim A \implies A$ 
  shows  $B \mid \sim B$ 

```

$\langle proof \rangle$

lemma

assumes $\neg\neg A. \neg\neg A \implies A$

shows $B \mid \neg B$

$\langle proof \rangle$

lemma

assumes $A \mid \neg A$

and $\neg\neg A$

shows A

$\langle proof \rangle$

5.1 Examples with quantifiers

lemma

assumes $\forall z. G(z)$

shows $\forall z. G(z) \mid H(z)$

$\langle proof \rangle$

lemma $\forall x. \exists y. x=y$

$\langle proof \rangle$

lemma $\exists y. \forall x. x=y$

$\langle proof \rangle$

Parallel lifting example.

lemma $\exists u. \forall x. \exists v. \forall y. \exists w. P(u,x,v,y,w)$

$\langle proof \rangle$

lemma

assumes $(\exists z. F(z)) \ \& \ B$

shows $\exists z. F(z) \ \& \ B$

$\langle proof \rangle$

A bigger demonstration of quantifiers – not in the paper.

lemma $(\exists y. \forall x. Q(x,y)) \implies (\forall x. \exists y. Q(x,y))$

$\langle proof \rangle$

end

6 First-Order Logic: PROLOG examples

theory *Prolog*

imports *FOL*

begin

```

typeddecl 'a list
arities list :: (term) term
consts
  Nil    :: 'a list
  Cons   :: ['a, 'a list] => 'a list  (infixr : 60)
  app    :: ['a list, 'a list, 'a list] => o
  rev    :: ['a list, 'a list] => o
axioms
  appNil: app(Nil,ys,ys)
  appCons: app(xs,ys,zs) ==> app(x:xs, ys, x:zs)
  revNil: rev(Nil,Nil)
  revCons: [| rev(xs,ys); app(ys, x:Nil, zs) |] ==> rev(x:xs, zs)

lemma app(a:b:c:Nil, d:e:Nil, ?x)
  <proof>

lemma app(?x, c:d:Nil, a:b:c:d:Nil)
  <proof>

lemma app(?x, ?y, a:b:c:d:Nil)
  <proof>

lemmas rules = appNil appCons revNil revCons

lemma rev(a:b:c:d:Nil, ?x)
  <proof>

lemma rev(a:b:c:d:e:f:g:h:i:j:k:l:m:n:Nil, ?w)
  <proof>

lemma rev(?x, a:b:c:Nil)
  <proof>

  <ML>

lemma rev(?x, a:b:c:Nil)
  <proof>

lemma rev(a:?x:c:?y:Nil, d:?z:b:?u)
  <proof>

lemma rev(a:b:c:d:e:f:g:h:i:j:k:l:m:n:o:p:Nil, ?w)
  <proof>

```

lemma $a:b:c:d:e:f:g:h:i:j:k:l:m:n:o:p:Nil = ?x \ \& \ app(?x,?x,?y) \ \& \ rev(?y,?w)$
 $\langle proof \rangle$

end

7 Intuitionistic First-Order Logic

theory *Intuitionistic* **imports** *IFOL* **begin**

Metatheorem (for *propositional* formulae): P is classically provable iff $\neg\neg P$ is intuitionistically provable. Therefore $\neg P$ is classically provable iff it is intuitionistically provable.

Proof: Let Q be the conjunction of the propositions $A \vee \neg A$, one for each atom A in P . Now $\neg\neg Q$ is intuitionistically provable because $\neg\neg(A \vee \neg A)$ is and because double-negation distributes over conjunction. If P is provable classically, then clearly $Q \rightarrow P$ is provable intuitionistically, so $\neg\neg(Q \rightarrow P)$ is also provable intuitionistically. The latter is intuitionistically equivalent to $\neg\neg Q \rightarrow \neg\neg P$, hence to $\neg\neg P$, since $\neg\neg Q$ is intuitionistically provable. Finally, if P is a negation then $\neg\neg P$ is intuitionistically equivalent to P .
[Andy Pitts]

lemma $\sim\sim(P \ \& \ Q) \ \leftrightarrow \ \sim\sim P \ \& \ \sim\sim Q$
 $\langle proof \rangle$

lemma $\sim\sim((\sim P \ \rightarrow \ Q) \ \rightarrow \ (\sim P \ \rightarrow \ \sim Q) \ \rightarrow \ P)$
 $\langle proof \rangle$

Double-negation does NOT distribute over disjunction

lemma $\sim\sim(P \ \rightarrow \ Q) \ \leftrightarrow \ (\sim\sim P \ \rightarrow \ \sim\sim Q)$
 $\langle proof \rangle$

lemma $\sim\sim\sim P \ \leftrightarrow \ \sim P$
 $\langle proof \rangle$

lemma $\sim\sim((P \ \rightarrow \ Q \mid R) \ \rightarrow \ (P \ \rightarrow \ Q) \mid (P \ \rightarrow \ R))$
 $\langle proof \rangle$

lemma $(P \ \leftrightarrow \ Q) \ \leftrightarrow \ (Q \ \leftrightarrow \ P)$
 $\langle proof \rangle$

lemma $((P \ \rightarrow \ (Q \mid (Q \ \rightarrow \ R))) \ \rightarrow \ R) \ \rightarrow \ R$
 $\langle proof \rangle$

lemma $((((G \ \rightarrow \ A) \ \rightarrow \ J) \ \rightarrow \ D \ \rightarrow \ E) \ \rightarrow \ (((H \ \rightarrow \ B) \ \rightarrow \ I) \ \rightarrow \ C \ \rightarrow \ J) \ \rightarrow \ (A \ \rightarrow \ H) \ \rightarrow \ F \ \rightarrow \ G \ \rightarrow \ (((C \ \rightarrow \ B) \ \rightarrow \ I) \ \rightarrow \ D) \ \rightarrow \ (A \ \rightarrow \ C))$

$$\text{---> } (((F\text{--->}A)\text{--->}B) \text{--->} I) \text{--->} E$$

 $\langle proof \rangle$

Lemmas for the propositional double-negation translation

lemma $P \text{--->} \sim\sim P$
 $\langle proof \rangle$

lemma $\sim\sim(\sim\sim P \text{--->} P)$
 $\langle proof \rangle$

lemma $\sim\sim P \ \& \ \sim\sim(P \text{--->} Q) \text{--->} \sim\sim Q$
 $\langle proof \rangle$

The following are classically but not constructively valid. The attempt to prove them terminates quickly!

lemma $((P\text{--->}Q) \text{--->} P) \text{--->} P$
 $\langle proof \rangle$

lemma $(P\&Q\text{--->}R) \text{--->} (P\text{--->}R) \mid (Q\text{--->}R)$
 $\langle proof \rangle$

7.1 de Bruijn formulae

de Bruijn formula with three predicates

lemma $((P\text{<->}Q) \text{--->} P\&Q\&R) \ \&$
 $((Q\text{<->}R) \text{--->} P\&Q\&R) \ \&$
 $((R\text{<->}P) \text{--->} P\&Q\&R) \text{--->} P\&Q\&R$
 $\langle proof \rangle$

de Bruijn formula with five predicates

lemma $((P\text{<->}Q) \text{--->} P\&Q\&R\&S\&T) \ \&$
 $((Q\text{<->}R) \text{--->} P\&Q\&R\&S\&T) \ \&$
 $((R\text{<->}S) \text{--->} P\&Q\&R\&S\&T) \ \&$
 $((S\text{<->}T) \text{--->} P\&Q\&R\&S\&T) \ \&$
 $((T\text{<->}P) \text{--->} P\&Q\&R\&S\&T) \text{--->} P\&Q\&R\&S\&T$
 $\langle proof \rangle$

Problem 1.1

lemma $(ALL \ x. \ EX \ y. \ ALL \ z. \ p(x) \ \& \ q(y) \ \& \ r(z)) \text{<->}$
 $(ALL \ z. \ EX \ y. \ ALL \ x. \ p(x) \ \& \ q(y) \ \& \ r(z))$
 $\langle proof \rangle$

Problem 3.1

lemma $\sim (EX \ x. \ ALL \ y. \ mem(y,x) \text{<->} \sim mem(x,x))$
 $\langle proof \rangle$

Problem 4.1: hopeless!

lemma $(ALL\ x.\ p(x) \multimap p(h(x)) \mid p(g(x))) \ \&\ (EX\ x.\ p(x)) \ \&\ (ALL\ x.\ \sim p(h(x)))$
 $\multimap (EX\ x.\ p(g(g(g(g(x))))))$
 $\langle proof \rangle$

7.2 Intuitionistic FOL: propositional problems based on Pelletier.

1

lemma $\sim\sim((P \multimap Q) \iff (\sim Q \multimap \sim P))$
 $\langle proof \rangle$

2

lemma $\sim\sim(\sim\sim P \iff P)$
 $\langle proof \rangle$

3

lemma $\sim(P \multimap Q) \multimap (Q \multimap P)$
 $\langle proof \rangle$

4

lemma $\sim\sim((\sim P \multimap Q) \iff (\sim Q \multimap P))$
 $\langle proof \rangle$

5

lemma $\sim\sim((P \mid Q \multimap P \mid R) \multimap P \mid (Q \multimap R))$
 $\langle proof \rangle$

6

lemma $\sim\sim(P \mid \sim P)$
 $\langle proof \rangle$

7

lemma $\sim\sim(P \mid \sim\sim P)$
 $\langle proof \rangle$

8. Peirce's law

lemma $\sim\sim(((P \multimap Q) \multimap P) \multimap P)$
 $\langle proof \rangle$

9

lemma $((P \mid Q) \ \&\ (\sim P \mid Q) \ \&\ (P \mid \sim Q)) \multimap \sim (\sim P \mid \sim Q)$
 $\langle proof \rangle$

10

lemma $(Q \multimap R) \multimap (R \multimap P \ \&\ Q) \multimap (P \multimap (Q \mid R)) \multimap (P \iff Q)$
 $\langle proof \rangle$

7.3 11. Proved in each direction (incorrectly, says Pelletier!!)

lemma $P \leftrightarrow P$

<proof>

12. Dijkstra's law

lemma $\sim\sim((P \leftrightarrow Q) \leftrightarrow R) \leftrightarrow (P \leftrightarrow (Q \leftrightarrow R))$

<proof>

lemma $((P \leftrightarrow Q) \leftrightarrow R) \rightarrow \sim\sim(P \leftrightarrow (Q \leftrightarrow R))$

<proof>

13. Distributive law

lemma $P \mid (Q \ \& \ R) \leftrightarrow (P \mid Q) \ \& \ (P \mid R)$

<proof>

14

lemma $\sim\sim((P \leftrightarrow Q) \leftrightarrow ((Q \mid \sim P) \ \& \ (\sim Q \mid P)))$

<proof>

15

lemma $\sim\sim((P \rightarrow Q) \leftrightarrow (\sim P \mid Q))$

<proof>

16

lemma $\sim\sim((P \rightarrow Q) \mid (Q \rightarrow P))$

<proof>

17

lemma $\sim\sim(((P \ \& \ (Q \rightarrow R)) \rightarrow S) \leftrightarrow ((\sim P \mid Q \mid S) \ \& \ (\sim P \mid \sim R \mid S)))$

<proof>

Dijkstra's "Golden Rule"

lemma $(P \ \& \ Q) \leftrightarrow P \leftrightarrow Q \leftrightarrow (P \mid Q)$

<proof>

7.4 ****Examples with quantifiers****

7.5 The converse is classical in the following implications...

lemma $(EX \ x. \ P(x) \rightarrow Q) \rightarrow (ALL \ x. \ P(x)) \rightarrow Q$

<proof>

lemma $((ALL \ x. \ P(x)) \rightarrow Q) \rightarrow \sim (ALL \ x. \ P(x) \ \& \ \sim Q)$

<proof>

lemma $((ALL \ x. \ \sim P(x)) \rightarrow Q) \rightarrow \sim (ALL \ x. \ \sim (P(x) \mid Q))$

<proof>

lemma $(ALL\ x.\ P(x)) \mid Q \dashv\dashv (ALL\ x.\ P(x) \mid Q)$
 $\langle proof \rangle$

lemma $(EX\ x.\ P \dashv\dashv Q(x)) \dashv\dashv (P \dashv\dashv (EX\ x.\ Q(x)))$
 $\langle proof \rangle$

7.6 The following are not constructively valid!

The attempt to prove them terminates quickly!

lemma $((ALL\ x.\ P(x)) \dashv\dashv Q) \dashv\dashv (EX\ x.\ P(x) \dashv\dashv Q)$
 $\langle proof \rangle$

lemma $(P \dashv\dashv (EX\ x.\ Q(x))) \dashv\dashv (EX\ x.\ P \dashv\dashv Q(x))$
 $\langle proof \rangle$

lemma $(ALL\ x.\ P(x) \mid Q) \dashv\dashv ((ALL\ x.\ P(x)) \mid Q)$
 $\langle proof \rangle$

lemma $(ALL\ x.\ \sim\sim P(x)) \dashv\dashv \sim\sim(ALL\ x.\ P(x))$
 $\langle proof \rangle$

Classically but not intuitionistically valid. Proved by a bug in 1986!

lemma $EX\ x.\ Q(x) \dashv\dashv (ALL\ x.\ Q(x))$
 $\langle proof \rangle$

7.7 Hard examples with quantifiers

The ones that have not been proved are not known to be valid! Some will require quantifier duplication – not currently available

18

lemma $\sim\sim(EX\ y.\ ALL\ x.\ P(y) \dashv\dashv P(x))$
 $\langle proof \rangle$

19

lemma $\sim\sim(EX\ x.\ ALL\ y\ z.\ (P(y) \dashv\dashv Q(z)) \dashv\dashv (P(x) \dashv\dashv Q(x)))$
 $\langle proof \rangle$

20

lemma $(ALL\ x\ y.\ EX\ z.\ ALL\ w.\ (P(x) \& Q(y) \dashv\dashv R(z) \& S(w)))$
 $\dashv\dashv (EX\ x\ y.\ P(x) \& Q(y)) \dashv\dashv (EX\ z.\ R(z))$
 $\langle proof \rangle$

21

lemma $(EX\ x.\ P \dashv\dashv Q(x)) \& (EX\ x.\ Q(x) \dashv\dashv P) \dashv\dashv \sim\sim(EX\ x.\ P \dashv\dashv Q(x))$
 $\langle proof \rangle$

22

lemma $(ALL\ x.\ P \leftrightarrow Q(x)) \dashv\vdash (P \leftrightarrow (ALL\ x.\ Q(x)))$
 $\langle proof \rangle$

23

lemma $\sim\sim((ALL\ x.\ P \mid Q(x)) \leftrightarrow (P \mid (ALL\ x.\ Q(x))))$
 $\langle proof \rangle$

24

lemma $\sim(EX\ x.\ S(x) \& Q(x)) \& (ALL\ x.\ P(x) \dashv\vdash Q(x) \mid R(x)) \&$
 $(\sim(EX\ x.\ P(x)) \dashv\vdash (EX\ x.\ Q(x))) \& (ALL\ x.\ Q(x) \mid R(x) \dashv\vdash S(x))$
 $\dashv\vdash \sim\sim(EX\ x.\ P(x) \& R(x)) \langle proof \rangle$

25

lemma $(EX\ x.\ P(x)) \&$
 $(ALL\ x.\ L(x) \dashv\vdash \sim(M(x) \& R(x))) \&$
 $(ALL\ x.\ P(x) \dashv\vdash (M(x) \& L(x))) \&$
 $((ALL\ x.\ P(x) \dashv\vdash Q(x)) \mid (EX\ x.\ P(x) \& R(x)))$
 $\dashv\vdash (EX\ x.\ Q(x) \& P(x))$
 $\langle proof \rangle$

26

lemma $(\sim\sim(EX\ x.\ p(x)) \leftrightarrow \sim\sim(EX\ x.\ q(x))) \&$
 $(ALL\ x.\ ALL\ y.\ p(x) \& q(y) \dashv\vdash (r(x) \leftrightarrow s(y)))$
 $\dashv\vdash ((ALL\ x.\ p(x) \dashv\vdash r(x)) \leftrightarrow (ALL\ x.\ q(x) \dashv\vdash s(x)))$
 $\langle proof \rangle$

27

lemma $(EX\ x.\ P(x) \& \sim Q(x)) \&$
 $(ALL\ x.\ P(x) \dashv\vdash R(x)) \&$
 $(ALL\ x.\ M(x) \& L(x) \dashv\vdash P(x)) \&$
 $((EX\ x.\ R(x) \& \sim Q(x)) \dashv\vdash (ALL\ x.\ L(x) \dashv\vdash \sim R(x)))$
 $\dashv\vdash (ALL\ x.\ M(x) \dashv\vdash \sim L(x))$
 $\langle proof \rangle$

28. AMENDED

lemma $(ALL\ x.\ P(x) \dashv\vdash (ALL\ x.\ Q(x))) \&$
 $(\sim\sim(ALL\ x.\ Q(x) \mid R(x)) \dashv\vdash (EX\ x.\ Q(x) \& S(x))) \&$
 $(\sim\sim(EX\ x.\ S(x)) \dashv\vdash (ALL\ x.\ L(x) \dashv\vdash M(x)))$
 $\dashv\vdash (ALL\ x.\ P(x) \& L(x) \dashv\vdash M(x))$
 $\langle proof \rangle$

29. Essentially the same as Principia Mathematica *11.71

lemma $(EX\ x.\ P(x)) \& (EX\ y.\ Q(y))$
 $\dashv\vdash ((ALL\ x.\ P(x) \dashv\vdash R(x)) \& (ALL\ y.\ Q(y) \dashv\vdash S(y)) \leftrightarrow$
 $(ALL\ x\ y.\ P(x) \& Q(y) \dashv\vdash R(x) \& S(y)))$
 $\langle proof \rangle$

30

lemma $(ALL\ x. (P(x) \mid Q(x)) \dashv\vdash \sim R(x)) \ \&$
 $(ALL\ x. (Q(x) \dashv\vdash \sim S(x)) \dashv\vdash P(x) \ \& \ R(x))$
 $\dashv\vdash (ALL\ x. \sim\sim S(x))$
 $\langle proof \rangle$

31

lemma $\sim(EX\ x. P(x) \ \& \ (Q(x) \mid R(x))) \ \&$
 $(EX\ x. L(x) \ \& \ P(x)) \ \&$
 $(ALL\ x. \sim R(x) \dashv\vdash M(x))$
 $\dashv\vdash (EX\ x. L(x) \ \& \ M(x))$
 $\langle proof \rangle$

32

lemma $(ALL\ x. P(x) \ \& \ (Q(x) \mid R(x)) \dashv\vdash S(x)) \ \&$
 $(ALL\ x. S(x) \ \& \ R(x) \dashv\vdash L(x)) \ \&$
 $(ALL\ x. M(x) \dashv\vdash R(x))$
 $\dashv\vdash (ALL\ x. P(x) \ \& \ M(x) \dashv\vdash L(x))$
 $\langle proof \rangle$

33

lemma $(ALL\ x. \sim\sim(P(a) \ \& \ (P(x) \dashv\vdash P(b)) \dashv\vdash P(c))) \ \<\dashv\vdash\>$
 $(ALL\ x. \sim\sim((\sim P(a) \mid P(x) \mid P(c)) \ \& \ (\sim P(a) \mid \sim P(b) \mid P(c))))$
 $\langle proof \rangle$

36

lemma $(ALL\ x. EX\ y. J(x,y)) \ \&$
 $(ALL\ x. EX\ y. G(x,y)) \ \&$
 $(ALL\ x\ y. J(x,y) \mid G(x,y) \dashv\vdash (ALL\ z. J(y,z) \mid G(y,z) \dashv\vdash H(x,z)))$
 $\dashv\vdash (ALL\ x. EX\ y. H(x,y))$
 $\langle proof \rangle$

37

lemma $(ALL\ z. EX\ w. ALL\ x. EX\ y.$
 $\sim\sim(P(x,z) \dashv\vdash P(y,w)) \ \& \ P(y,z) \ \& \ (P(y,w) \dashv\vdash (EX\ u. Q(u,w)))) \ \&$
 $(ALL\ x\ z. \sim P(x,z) \dashv\vdash (EX\ y. Q(y,z))) \ \&$
 $(\sim\sim(EX\ x\ y. Q(x,y)) \dashv\vdash (ALL\ x. R(x,x)))$
 $\dashv\vdash \sim\sim(ALL\ x. EX\ y. R(x,y))$
 $\langle proof \rangle$

39

lemma $\sim(EX\ x. ALL\ y. F(y,x) \ \<\dashv\vdash\> \sim F(y,y))$
 $\langle proof \rangle$

40. AMENDED

lemma $(EX\ y. ALL\ x. F(x,y) \ \<\dashv\vdash\> F(x,x)) \dashv\vdash$
 $\sim(ALL\ x. EX\ y. ALL\ z. F(z,y) \ \<\dashv\vdash\> \sim F(z,x))$

$\langle proof \rangle$

44

lemma $(ALL\ x.\ f(x) \dashv\vdash$
 $(EX\ y.\ g(y) \ \&\ h(x,y) \ \&\ (EX\ y.\ g(y) \ \&\ \sim h(x,y)))) \ \&$
 $(EX\ x.\ j(x) \ \&\ (ALL\ y.\ g(y) \dashv\vdash h(x,y)))$
 $\dashv\vdash (EX\ x.\ j(x) \ \&\ \sim f(x))$

$\langle proof \rangle$

48

lemma $(a=b \mid c=d) \ \&\ (a=c \mid b=d) \dashv\vdash a=d \mid b=c$

$\langle proof \rangle$

51

lemma $(EX\ z\ w.\ ALL\ x\ y.\ P(x,y) <-> (x=z \ \&\ y=w)) \dashv\vdash$
 $(EX\ z.\ ALL\ x.\ EX\ w.\ (ALL\ y.\ P(x,y) <-> y=w) <-> x=z)$

$\langle proof \rangle$

52

Almost the same as 51.

lemma $(EX\ z\ w.\ ALL\ x\ y.\ P(x,y) <-> (x=z \ \&\ y=w)) \dashv\vdash$
 $(EX\ w.\ ALL\ y.\ EX\ z.\ (ALL\ x.\ P(x,y) <-> x=z) <-> y=w)$

$\langle proof \rangle$

56

lemma $(ALL\ x.\ (EX\ y.\ P(y) \ \&\ x=f(y)) \dashv\vdash P(x)) <-> (ALL\ x.\ P(x) \dashv\vdash$
 $P(f(x)))$

$\langle proof \rangle$

57

lemma $P(f(a,b), f(b,c)) \ \&\ P(f(b,c), f(a,c)) \ \&$
 $(ALL\ x\ y\ z.\ P(x,y) \ \&\ P(y,z) \dashv\vdash P(x,z)) \dashv\vdash P(f(a,b), f(a,c))$

$\langle proof \rangle$

60

lemma $ALL\ x.\ P(x, f(x)) <-> (EX\ y.\ (ALL\ z.\ P(z,y) \dashv\vdash P(z, f(x))) \ \&\ P(x,y))$

$\langle proof \rangle$

end

8 First-Order Logic: propositional examples (intuitionistic version)

theory *Propositional-Int*

imports *IFOL*
begin

commutative laws of $\&$ and $|$

lemma $P \& Q \dashv\dashv Q \& P$
 $\langle proof \rangle$

lemma $P | Q \dashv\dashv Q | P$
 $\langle proof \rangle$

associative laws of $\&$ and $|$

lemma $(P \& Q) \& R \dashv\dashv P \& (Q \& R)$
 $\langle proof \rangle$

lemma $(P | Q) | R \dashv\dashv P | (Q | R)$
 $\langle proof \rangle$

distributive laws of $\&$ and $|$

lemma $(P \& Q) | R \dashv\dashv (P | R) \& (Q | R)$
 $\langle proof \rangle$

lemma $(P | R) \& (Q | R) \dashv\dashv (P \& Q) | R$
 $\langle proof \rangle$

lemma $(P | Q) \& R \dashv\dashv (P \& R) | (Q \& R)$
 $\langle proof \rangle$

lemma $(P \& R) | (Q \& R) \dashv\dashv (P | Q) \& R$
 $\langle proof \rangle$

Laws involving implication

lemma $(P \dashv\dashv R) \& (Q \dashv\dashv R) \dashv\dashv (P | Q \dashv\dashv R)$
 $\langle proof \rangle$

lemma $(P \& Q \dashv\dashv R) \dashv\dashv (P \dashv\dashv (Q \dashv\dashv R))$
 $\langle proof \rangle$

lemma $((P \dashv\dashv R) \dashv\dashv R) \dashv\dashv ((Q \dashv\dashv R) \dashv\dashv R) \dashv\dashv (P \& Q \dashv\dashv R) \dashv\dashv R$
 $\langle proof \rangle$

lemma $\sim(P \dashv\dashv R) \dashv\dashv \sim(Q \dashv\dashv R) \dashv\dashv \sim(P \& Q \dashv\dashv R)$
 $\langle proof \rangle$

lemma $(P \dashv\dashv Q \& R) \dashv\dashv (P \dashv\dashv Q) \& (P \dashv\dashv R)$
 $\langle proof \rangle$

Propositions-as-types

— The combinator K

lemma $P \multimap (Q \multimap P)$

<proof>

lemma $(P \multimap Q \multimap R) \multimap (P \multimap Q) \multimap (P \multimap R)$

<proof>

lemma $(P \multimap Q) \mid (P \multimap R) \multimap (P \multimap Q \mid R)$

<proof>

lemma $(P \multimap Q) \multimap (\sim Q \multimap \sim P)$

<proof>

Schwichtenberg's examples (via T. Nipkow)

lemma *stab-imp*: $((Q \multimap R) \multimap R) \multimap Q \multimap (((P \multimap Q) \multimap R) \multimap R) \multimap P \multimap Q$

<proof>

lemma *stab-to-peirce*:

$((P \multimap R) \multimap R) \multimap P \multimap (((Q \multimap R) \multimap R) \multimap Q)$
 $\multimap ((P \multimap Q) \multimap P) \multimap P$

<proof>

lemma *peirce-imp1*: $((Q \multimap R) \multimap Q) \multimap Q$

$\multimap ((P \multimap Q) \multimap R) \multimap P \multimap Q \multimap P \multimap Q$

<proof>

lemma *peirce-imp2*: $((P \multimap R) \multimap P) \multimap P \multimap ((P \multimap Q \multimap R) \multimap P) \multimap P$

<proof>

lemma *minits*: $((P \multimap Q) \multimap P) \multimap P \multimap Q \multimap Q$

<proof>

lemma *minits-solovev*: $(P \multimap (Q \multimap R) \multimap Q) \multimap ((P \multimap Q) \multimap R) \multimap R$

<proof>

lemma *tatsuta*: $((P7 \multimap P1) \multimap P10) \multimap P4 \multimap P5$

$\multimap (((P8 \multimap P2) \multimap P9) \multimap P3 \multimap P10)$

$\multimap (P1 \multimap P8) \multimap P6 \multimap P7$

$\multimap (((P3 \multimap P2) \multimap P9) \multimap P4)$

$\multimap (P1 \multimap P3) \multimap (((P6 \multimap P1) \multimap P2) \multimap P9) \multimap P5$

<proof>

lemma *tatsuta1*: $((P8 \multimap P2) \multimap P9) \multimap P3 \multimap P10$

$\multimap (((P3 \multimap P2) \multimap P9) \multimap P4)$

$\multimap (((P6 \multimap P1) \multimap P2) \multimap P9)$

$\multimap (((P7 \multimap P1) \multimap P10) \multimap P4 \multimap P5)$

$\multimap (P1 \multimap P3) \multimap (P1 \multimap P8) \multimap P6 \multimap P7 \multimap P5$

<proof>

end

9 First-Order Logic: quantifier examples (intuitionistic version)

theory *Quantifiers-Int*
imports *IFOL*
begin

lemma $(ALL\ x\ y.\ P(x,y)) \dashv\vdash (ALL\ y\ x.\ P(x,y))$
<proof>

lemma $(EX\ x\ y.\ P(x,y)) \dashv\vdash (EX\ y\ x.\ P(x,y))$
<proof>

lemma $(ALL\ x.\ P(x)) \mid (ALL\ x.\ Q(x)) \dashv\vdash (ALL\ x.\ P(x) \mid Q(x))$
<proof>

lemma $(ALL\ x.\ P \dashv\vdash Q(x)) \iff (P \dashv\vdash (ALL\ x.\ Q(x)))$
<proof>

lemma $(ALL\ x.\ P(x) \dashv\vdash Q) \iff ((EX\ x.\ P(x)) \dashv\vdash Q)$
<proof>

Some harder ones

lemma $(EX\ x.\ P(x) \mid Q(x)) \iff (EX\ x.\ P(x)) \mid (EX\ x.\ Q(x))$
<proof>

lemma $(EX\ x.\ P(x) \& Q(x)) \dashv\vdash (EX\ x.\ P(x)) \ \& \ (EX\ x.\ Q(x))$
<proof>

Basic test of quantifier reasoning

— TRUE

lemma $(EX\ y.\ ALL\ x.\ Q(x,y)) \dashv\vdash (ALL\ x.\ EX\ y.\ Q(x,y))$
<proof>

lemma $(ALL\ x.\ Q(x)) \dashv\vdash (EX\ x.\ Q(x))$
<proof>

The following should fail, as they are false!

lemma $(ALL\ x.\ EX\ y.\ Q(x,y)) \dashv\vdash (EX\ y.\ ALL\ x.\ Q(x,y))$
<proof>

lemma $(EX\ x.\ Q(x)) \dashv\vdash (ALL\ x.\ Q(x))$
<proof>

lemma $P(?a) \dashv\vdash (ALL\ x.\ P(x))$
<proof>

lemma $(P(?a) \multimap (ALL\ x.\ Q(x))) \multimap (ALL\ x.\ P(x) \multimap Q(x))$
 $\langle proof \rangle$

Back to things that are provable ...

lemma $(ALL\ x.\ P(x) \multimap Q(x)) \ \&\ (EX\ x.\ P(x)) \multimap (EX\ x.\ Q(x))$
 $\langle proof \rangle$

lemma $(P \multimap (EX\ x.\ Q(x))) \ \&\ P \multimap (EX\ x.\ Q(x))$
 $\langle proof \rangle$

lemma $(ALL\ x.\ P(x) \multimap Q(f(x))) \ \&\ (ALL\ x.\ Q(x) \multimap R(g(x))) \ \&\ P(d) \multimap R(?a)$
 $\langle proof \rangle$

lemma $(ALL\ x.\ Q(x)) \multimap (EX\ x.\ Q(x))$
 $\langle proof \rangle$

Some slow ones

— Principia Mathematica *11.53

lemma $(ALL\ x\ y.\ P(x) \multimap Q(y)) \ \<->\ ((EX\ x.\ P(x)) \multimap (ALL\ y.\ Q(y)))$
 $\langle proof \rangle$

lemma $(EX\ x\ y.\ P(x) \ \&\ Q(x,y)) \ \<->\ (EX\ x.\ P(x) \ \&\ (EX\ y.\ Q(x,y)))$
 $\langle proof \rangle$

lemma $(EX\ y.\ ALL\ x.\ P(x) \multimap Q(x,y)) \multimap (ALL\ x.\ P(x) \multimap (EX\ y.\ Q(x,y)))$
 $\langle proof \rangle$

end

10 Classical Predicate Calculus Problems

theory *Classical* **imports** *FOL* **begin**

lemma $(P \multimap Q \mid R) \multimap (P \multimap Q) \mid (P \multimap R)$
 $\langle proof \rangle$

If and only if

lemma $(P \ \<->\ Q) \ \<->\ (Q \ \<->\ P)$
 $\langle proof \rangle$

lemma $\sim (P \ \<->\ \sim P)$
 $\langle proof \rangle$

Sample problems from F. J. Pelletier, Seventy-Five Problems for Testing Automatic Theorem Provers, J. Automated Reasoning 2 (1986), 191-216. Errata, JAR 4 (1988), 236-236.

The hardest problems – judging by experience with several theorem provers, including matrix ones – are 34 and 43.

10.1 Pelletier's examples

1

lemma $(P \multimap Q) \leftrightarrow (\sim Q \multimap \sim P)$
<proof>

2

lemma $\sim \sim P \leftrightarrow P$
<proof>

3

lemma $\sim(P \multimap Q) \multimap (Q \multimap P)$
<proof>

4

lemma $(\sim P \multimap Q) \leftrightarrow (\sim Q \multimap P)$
<proof>

5

lemma $((P|Q) \multimap (P|R)) \multimap (P|(Q \multimap R))$
<proof>

6

lemma $P | \sim P$
<proof>

7

lemma $P | \sim \sim \sim P$
<proof>

8. Peirce's law

lemma $((P \multimap Q) \multimap P) \multimap P$
<proof>

9

lemma $((P|Q) \& (\sim P|Q) \& (P|\sim Q)) \multimap \sim(\sim P|\sim Q)$
<proof>

10

lemma $(Q \multimap R) \ \& \ (R \multimap P \ \& \ Q) \ \& \ (P \multimap Q \mid R) \multimap (P \multimap Q)$
 $\langle proof \rangle$

11. Proved in each direction (incorrectly, says Pelletier!!)

lemma $P \multimap P$
 $\langle proof \rangle$

12. "Dijkstra's law"

lemma $((P \multimap Q) \multimap R) \multimap (P \multimap (Q \multimap R))$
 $\langle proof \rangle$

13. Distributive law

lemma $P \mid (Q \ \& \ R) \multimap (P \mid Q) \ \& \ (P \mid R)$
 $\langle proof \rangle$

14

lemma $(P \multimap Q) \multimap ((Q \mid \sim P) \ \& \ (\sim Q \mid P))$
 $\langle proof \rangle$

15

lemma $(P \multimap Q) \multimap (\sim P \mid Q)$
 $\langle proof \rangle$

16

lemma $(P \multimap Q) \mid (Q \multimap P)$
 $\langle proof \rangle$

17

lemma $((P \ \& \ (Q \multimap R)) \multimap S) \multimap ((\sim P \mid Q \mid S) \ \& \ (\sim P \mid \sim R \mid S))$
 $\langle proof \rangle$

10.2 Classical Logic: examples with quantifiers

lemma $(\forall x. P(x) \ \& \ Q(x)) \multimap (\forall x. P(x)) \ \& \ (\forall x. Q(x))$
 $\langle proof \rangle$

lemma $(\exists x. P \multimap Q(x)) \multimap (P \multimap (\exists x. Q(x)))$
 $\langle proof \rangle$

lemma $(\exists x. P(x) \multimap Q) \multimap (\forall x. P(x)) \multimap Q$
 $\langle proof \rangle$

lemma $(\forall x. P(x)) \mid Q \multimap (\forall x. P(x) \mid Q)$
 $\langle proof \rangle$

Discussed in Avron, Gentzen-Type Systems, Resolution and Tableaux, JAR 10 (265-281), 1993. Proof is trivial!

lemma $\sim((\exists x. \sim P(x)) \ \& \ ((\exists x. P(x)) \mid (\exists x. P(x) \ \& \ Q(x))) \ \& \ \sim(\exists x. P(x)))$
 $\langle proof \rangle$

10.3 Problems requiring quantifier duplication

Theorem B of Peter Andrews, Theorem Proving via General Matings, JACM 28 (1981).

lemma $(\exists x. \forall y. P(x) \leftrightarrow P(y)) \dashv\vdash ((\exists x. P(x)) \leftrightarrow (\forall y. P(y)))$
 $\langle proof \rangle$

Needs multiple instantiation of ALL.

lemma $(\forall x. P(x) \dashv\vdash P(f(x))) \ \& \ P(d) \dashv\vdash P(f(f(f(d))))$
 $\langle proof \rangle$

Needs double instantiation of the quantifier

lemma $\exists x. P(x) \dashv\vdash P(a) \ \& \ P(b)$
 $\langle proof \rangle$

lemma $\exists z. P(z) \dashv\vdash (\forall x. P(x))$
 $\langle proof \rangle$

lemma $\exists x. (\exists y. P(y)) \dashv\vdash P(x)$
 $\langle proof \rangle$

V. Lifschitz, What Is the Inverse Method?, JAR 5 (1989), 1–23. NOT PROVED

lemma $\exists x \ x'. \forall y. \exists z \ z'.$
 $(\sim P(y,y) \mid P(x,x) \mid \sim S(z,x)) \ \&$
 $(S(x,y) \mid \sim S(y,z) \mid Q(z',z')) \ \&$
 $(Q(x',y) \mid \sim Q(y,z') \mid S(x',x'))$
 $\langle proof \rangle$

10.4 Hard examples with quantifiers

18

lemma $\exists y. \forall x. P(y) \dashv\vdash P(x)$
 $\langle proof \rangle$

19

lemma $\exists x. \forall y \ z. (P(y) \dashv\vdash Q(z)) \dashv\vdash (P(x) \dashv\vdash Q(x))$
 $\langle proof \rangle$

20

lemma $(\forall x \ y. \exists z. \forall w. (P(x) \ \& \ Q(y) \dashv\vdash R(z) \ \& \ S(w)))$
 $\dashv\vdash (\exists x \ y. P(x) \ \& \ Q(y)) \dashv\vdash (\exists z. R(z))$
 $\langle proof \rangle$

21

lemma $(\exists x. P \dashv\vdash Q(x)) \ \& \ (\exists x. Q(x) \dashv\vdash P) \dashv\vdash (\exists x. P \leftrightarrow Q(x))$

$\langle proof \rangle$

22

lemma $(\forall x. P \leftrightarrow Q(x)) \rightarrow (P \leftrightarrow (\forall x. Q(x)))$
 $\langle proof \rangle$

23

lemma $(\forall x. P \mid Q(x)) \leftrightarrow (P \mid (\forall x. Q(x)))$
 $\langle proof \rangle$

24

lemma $\sim(\exists x. S(x) \& Q(x)) \& (\forall x. P(x) \rightarrow Q(x) \mid R(x)) \&$
 $(\sim(\exists x. P(x)) \rightarrow (\exists x. Q(x))) \& (\forall x. Q(x) \mid R(x) \rightarrow S(x))$
 $\rightarrow (\exists x. P(x) \& R(x))$
 $\langle proof \rangle$

25

lemma $(\exists x. P(x)) \&$
 $(\forall x. L(x) \rightarrow \sim(M(x) \& R(x))) \&$
 $(\forall x. P(x) \rightarrow (M(x) \& L(x))) \&$
 $((\forall x. P(x) \rightarrow Q(x)) \mid (\exists x. P(x) \& R(x)))$
 $\rightarrow (\exists x. Q(x) \& P(x))$
 $\langle proof \rangle$

26

lemma $((\exists x. p(x)) \leftrightarrow (\exists x. q(x))) \&$
 $(\forall x. \forall y. p(x) \& q(y) \rightarrow (r(x) \leftrightarrow s(y)))$
 $\rightarrow ((\forall x. p(x) \rightarrow r(x)) \leftrightarrow (\forall x. q(x) \rightarrow s(x)))$
 $\langle proof \rangle$

27

lemma $(\exists x. P(x) \& \sim Q(x)) \&$
 $(\forall x. P(x) \rightarrow R(x)) \&$
 $(\forall x. M(x) \& L(x) \rightarrow P(x)) \&$
 $((\exists x. R(x) \& \sim Q(x)) \rightarrow (\forall x. L(x) \rightarrow \sim R(x)))$
 $\rightarrow (\forall x. M(x) \rightarrow \sim L(x))$
 $\langle proof \rangle$

28. AMENDED

lemma $(\forall x. P(x) \rightarrow (\forall x. Q(x))) \&$
 $((\forall x. Q(x) \mid R(x)) \rightarrow (\exists x. Q(x) \& S(x))) \&$
 $((\exists x. S(x)) \rightarrow (\forall x. L(x) \rightarrow M(x)))$
 $\rightarrow (\forall x. P(x) \& L(x) \rightarrow M(x))$
 $\langle proof \rangle$

29. Essentially the same as Principia Mathematica *11.71

lemma $(\exists x. P(x)) \& (\exists y. Q(y))$

$$\begin{aligned} & \rightarrow (\forall x. P(x) \rightarrow R(x)) \ \& \ (\forall y. Q(y) \rightarrow S(y)) \quad \leftrightarrow \\ & (\forall x y. P(x) \ \& \ Q(y) \rightarrow R(x) \ \& \ S(y)) \end{aligned}$$
 $\langle proof \rangle$

30

lemma $(\forall x. P(x) \mid Q(x) \rightarrow \sim R(x)) \ \&$
 $(\forall x. (Q(x) \rightarrow \sim S(x)) \rightarrow P(x) \ \& \ R(x))$
 $\rightarrow (\forall x. S(x))$
 $\langle proof \rangle$

31

lemma $\sim(\exists x. P(x) \ \& \ (Q(x) \mid R(x))) \ \&$
 $(\exists x. L(x) \ \& \ P(x)) \ \&$
 $(\forall x. \sim R(x) \rightarrow M(x))$
 $\rightarrow (\exists x. L(x) \ \& \ M(x))$
 $\langle proof \rangle$

32

lemma $(\forall x. P(x) \ \& \ (Q(x) \mid R(x)) \rightarrow S(x)) \ \&$
 $(\forall x. S(x) \ \& \ R(x) \rightarrow L(x)) \ \&$
 $(\forall x. M(x) \rightarrow R(x))$
 $\rightarrow (\forall x. P(x) \ \& \ M(x) \rightarrow L(x))$
 $\langle proof \rangle$

33

lemma $(\forall x. P(a) \ \& \ (P(x) \rightarrow P(b)) \rightarrow P(c)) \quad \leftrightarrow$
 $(\forall x. (\sim P(a) \mid P(x) \mid P(c)) \ \& \ (\sim P(a) \mid \sim P(b) \mid P(c)))$
 $\langle proof \rangle$

34 AMENDED (TWICE!!). Andrews's challenge

lemma $((\exists x. \forall y. p(x) \leftrightarrow p(y)) \quad \leftrightarrow$
 $((\exists x. q(x)) \leftrightarrow (\forall y. p(y)))) \quad \leftrightarrow$
 $((\exists x. \forall y. q(x) \leftrightarrow q(y)) \quad \leftrightarrow$
 $((\exists x. p(x)) \leftrightarrow (\forall y. q(y))))$
 $\langle proof \rangle$

35

lemma $\exists x y. P(x, y) \rightarrow (\forall u v. P(u, v))$
 $\langle proof \rangle$

36

lemma $(\forall x. \exists y. J(x, y)) \ \&$
 $(\forall x. \exists y. G(x, y)) \ \&$
 $(\forall x y. J(x, y) \mid G(x, y) \rightarrow (\forall z. J(y, z) \mid G(y, z) \rightarrow H(x, z)))$
 $\rightarrow (\forall x. \exists y. H(x, y))$
 $\langle proof \rangle$

37

lemma $(\forall z. \exists w. \forall x. \exists y.$
 $(P(x,z) \dashv\vdash P(y,w)) \ \& \ P(y,z) \ \& \ (P(y,w) \dashv\vdash (\exists u. Q(u,w)))) \ \&$
 $(\forall x \ z. \sim P(x,z) \dashv\vdash (\exists y. Q(y,z))) \ \&$
 $((\exists x \ y. Q(x,y)) \dashv\vdash (\forall x. R(x,x)))$
 $\dashv\vdash (\forall x. \exists y. R(x,y))$
 $\langle proof \rangle$

38

lemma $(\forall x. p(a) \ \& \ (p(x) \dashv\vdash (\exists y. p(y) \ \& \ r(x,y))) \dashv\vdash$
 $(\exists z. \exists w. p(z) \ \& \ r(x,w) \ \& \ r(w,z))) \ <\dashv\vdash$
 $(\forall x. (\sim p(a) \mid p(x) \mid (\exists z. \exists w. p(z) \ \& \ r(x,w) \ \& \ r(w,z))) \ \&$
 $(\sim p(a) \mid \sim(\exists y. p(y) \ \& \ r(x,y)) \mid$
 $(\exists z. \exists w. p(z) \ \& \ r(x,w) \ \& \ r(w,z))))$
 $\langle proof \rangle$

39

lemma $\sim (\exists x. \forall y. F(y,x) \ <\dashv\vdash \sim F(y,y))$
 $\langle proof \rangle$

40. AMENDED

lemma $(\exists y. \forall x. F(x,y) \ <\dashv\vdash F(x,x)) \dashv\vdash$
 $\sim(\forall x. \exists y. \forall z. F(z,y) \ <\dashv\vdash \sim F(z,x))$
 $\langle proof \rangle$

41

lemma $(\forall z. \exists y. \forall x. f(x,y) \ <\dashv\vdash f(x,z) \ \& \ \sim f(x,x))$
 $\dashv\vdash \sim (\exists z. \forall x. f(x,z))$
 $\langle proof \rangle$

42

lemma $\sim (\exists y. \forall x. p(x,y) \ <\dashv\vdash \sim (\exists z. p(x,z) \ \& \ p(z,x)))$
 $\langle proof \rangle$

43

lemma $(\forall x. \forall y. q(x,y) \ <\dashv\vdash (\forall z. p(z,x) \ <\dashv\vdash p(z,y)))$
 $\dashv\vdash (\forall x. \forall y. q(x,y) \ <\dashv\vdash q(y,x))$
 $\langle proof \rangle$

44

lemma $(\forall x. f(x) \dashv\vdash (\exists y. g(y) \ \& \ h(x,y) \ \& \ (\exists y. g(y) \ \& \ \sim h(x,y)))) \ \&$
 $(\exists x. j(x) \ \& \ (\forall y. g(y) \dashv\vdash h(x,y)))$
 $\dashv\vdash (\exists x. j(x) \ \& \ \sim f(x))$
 $\langle proof \rangle$

45

lemma $(\forall x. f(x) \ \& \ (\forall y. g(y) \ \& \ h(x,y) \dashv\vdash j(x,y))$
 $\dashv\vdash (\forall y. g(y) \ \& \ h(x,y) \dashv\vdash k(y))) \ \&$

$\sim (\exists y. l(y) \& k(y)) \&$
 $(\exists x. f(x) \& (\forall y. h(x,y) \dashrightarrow l(y))$
 $\& (\forall y. g(y) \& h(x,y) \dashrightarrow j(x,y)))$
 $\dashrightarrow (\exists x. f(x) \& \sim (\exists y. g(y) \& h(x,y)))$
 $\langle proof \rangle$

46

lemma $(\forall x. f(x) \& (\forall y. f(y) \& h(y,x) \dashrightarrow g(y)) \dashrightarrow g(x)) \&$
 $((\exists x. f(x) \& \sim g(x)) \dashrightarrow$
 $(\exists x. f(x) \& \sim g(x) \& (\forall y. f(y) \& \sim g(y) \dashrightarrow j(x,y)))) \&$
 $(\forall x y. f(x) \& f(y) \& h(x,y) \dashrightarrow \sim j(y,x))$
 $\dashrightarrow (\forall x. f(x) \dashrightarrow g(x))$
 $\langle proof \rangle$

10.5 Problems (mainly) involving equality or functions

48

lemma $(a=b \mid c=d) \& (a=c \mid b=d) \dashrightarrow a=d \mid b=c$
 $\langle proof \rangle$

49 NOT PROVED AUTOMATICALLY. Hard because it involves substitution for Vars the type constraint ensures that x,y,z have the same type as a,b,u.

lemma $(\exists x y::'a. \forall z. z=x \mid z=y) \& P(a) \& P(b) \& a \sim b$
 $\dashrightarrow (\forall u::'a. P(u))$
 $\langle proof \rangle$

50. (What has this to do with equality?)

lemma $(\forall x. P(a,x) \mid (\forall y. P(x,y))) \dashrightarrow (\exists x. \forall y. P(x,y))$
 $\langle proof \rangle$

51

lemma $(\exists z w. \forall x y. P(x,y) \<-> (x=z \& y=w)) \dashrightarrow$
 $(\exists z. \forall x. \exists w. (\forall y. P(x,y) \<-> y=w) \<-> x=z)$
 $\langle proof \rangle$

52

Almost the same as 51.

lemma $(\exists z w. \forall x y. P(x,y) \<-> (x=z \& y=w)) \dashrightarrow$
 $(\exists w. \forall y. \exists z. (\forall x. P(x,y) \<-> x=z) \<-> y=w)$
 $\langle proof \rangle$

55

Non-equational version, from Manthey and Bry, CADE-9 (Springer, 1988). fast DISCOVERS who killed Agatha.

lemma $\text{lives}(\text{agatha}) \ \& \ \text{lives}(\text{butler}) \ \& \ \text{lives}(\text{charles}) \ \&$
 $(\text{killed}(\text{agatha}, \text{agatha}) \mid \text{killed}(\text{butler}, \text{agatha}) \mid \text{killed}(\text{charles}, \text{agatha})) \ \&$
 $(\forall x \ y. \text{killed}(x, y) \dashrightarrow \text{hates}(x, y) \ \& \ \sim \text{richer}(x, y)) \ \&$
 $(\forall x. \text{hates}(\text{agatha}, x) \dashrightarrow \sim \text{hates}(\text{charles}, x)) \ \&$
 $(\text{hates}(\text{agatha}, \text{agatha}) \ \& \ \text{hates}(\text{agatha}, \text{charles})) \ \&$
 $(\forall x. \text{lives}(x) \ \& \ \sim \text{richer}(x, \text{agatha}) \dashrightarrow \text{hates}(\text{butler}, x)) \ \&$
 $(\forall x. \text{hates}(\text{agatha}, x) \dashrightarrow \text{hates}(\text{butler}, x)) \ \&$
 $(\forall x. \sim \text{hates}(x, \text{agatha}) \mid \sim \text{hates}(x, \text{butler}) \mid \sim \text{hates}(x, \text{charles})) \dashrightarrow$
 $\text{killed}(\text{?who}, \text{agatha})$
 $\langle \text{proof} \rangle$

56

lemma $(\forall x. (\exists y. P(y) \ \& \ x=f(y)) \dashrightarrow P(x)) <-> (\forall x. P(x) \dashrightarrow P(f(x)))$
 $\langle \text{proof} \rangle$

57

lemma $P(f(a, b), f(b, c)) \ \& \ P(f(b, c), f(a, c)) \ \&$
 $(\forall x \ y \ z. P(x, y) \ \& \ P(y, z) \dashrightarrow P(x, z)) \dashrightarrow P(f(a, b), f(a, c))$
 $\langle \text{proof} \rangle$

58 NOT PROVED AUTOMATICALLY

lemma $(\forall x \ y. f(x)=g(y)) \dashrightarrow (\forall x \ y. f(f(x))=f(g(y)))$
 $\langle \text{proof} \rangle$

59

lemma $(\forall x. P(x) <-> \sim P(f(x))) \dashrightarrow (\exists x. P(x) \ \& \ \sim P(f(x)))$
 $\langle \text{proof} \rangle$

60

lemma $\forall x. P(x, f(x)) <-> (\exists y. (\forall z. P(z, y) \dashrightarrow P(z, f(x))) \ \& \ P(x, y))$
 $\langle \text{proof} \rangle$

62 as corrected in JAR 18 (1997), page 135

lemma $(\forall x. p(a) \ \& \ (p(x) \dashrightarrow p(f(x))) \dashrightarrow p(f(f(x)))) <->$
 $(\forall x. (\sim p(a) \mid p(x) \mid p(f(f(x)))) \ \&$
 $(\sim p(a) \mid \sim p(f(x)) \mid p(f(f(x))))$
 $\langle \text{proof} \rangle$

From Davis, Obvious Logical Inferences, IJCAI-81, 530-531 fast indeed copes!

lemma $(\forall x. F(x) \ \& \ \sim G(x) \dashrightarrow (\exists y. H(x, y) \ \& \ J(y))) \ \&$
 $(\exists x. K(x) \ \& \ F(x) \ \& \ (\forall y. H(x, y) \dashrightarrow K(y))) \ \&$
 $(\forall x. K(x) \dashrightarrow \sim G(x)) \dashrightarrow (\exists x. K(x) \ \& \ J(x))$
 $\langle \text{proof} \rangle$

From Rudnicki, Obvious Inferences, JAR 3 (1987), 383-393. It does seem obvious!

lemma $(\forall x. F(x) \ \& \ \sim G(x) \dashrightarrow (\exists y. H(x,y) \ \& \ J(y))) \ \& \$
 $(\exists x. K(x) \ \& \ F(x) \ \& \ (\forall y. H(x,y) \dashrightarrow K(y))) \ \& \$
 $(\forall x. K(x) \dashrightarrow \sim G(x)) \dashrightarrow (\exists x. K(x) \dashrightarrow \sim G(x))$
 $\langle proof \rangle$

Halting problem: Formulation of Li Dafa (AAR Newsletter 27, Oct 1994.)
author U. Egly

lemma $((\exists x. A(x) \ \& \ (\forall y. C(y) \dashrightarrow (\forall z. D(x,y,z)))) \dashrightarrow \$
 $(\exists w. C(w) \ \& \ (\forall y. C(y) \dashrightarrow (\forall z. D(w,y,z)))) \$
 $\& \$
 $(\forall w. C(w) \ \& \ (\forall u. C(u) \dashrightarrow (\forall v. D(w,u,v))) \dashrightarrow \$
 $(\forall y \ z. \$
 $(C(y) \ \& \ P(y,z) \dashrightarrow Q(w,y,z) \ \& \ OO(w,g)) \ \& \$
 $(C(y) \ \& \ \sim P(y,z) \dashrightarrow Q(w,y,z) \ \& \ OO(w,b))) \$
 $\& \$
 $(\forall w. C(w) \ \& \$
 $(\forall y \ z. \$
 $(C(y) \ \& \ P(y,z) \dashrightarrow Q(w,y,z) \ \& \ OO(w,g)) \ \& \$
 $(C(y) \ \& \ \sim P(y,z) \dashrightarrow Q(w,y,z) \ \& \ OO(w,b))) \dashrightarrow \$
 $(\exists v. C(v) \ \& \$
 $(\forall y. ((C(y) \ \& \ Q(w,y,y)) \ \& \ OO(w,g) \dashrightarrow \sim P(v,y)) \ \& \$
 $((C(y) \ \& \ Q(w,y,y)) \ \& \ OO(w,b) \dashrightarrow P(v,y) \ \& \ OO(v,b)))) \$
 $\dashrightarrow \$
 $\sim (\exists x. A(x) \ \& \ (\forall y. C(y) \dashrightarrow (\forall z. D(x,y,z)))) \$
 $\langle proof \rangle$

Halting problem II: credited to M. Bruschi by Li Dafa in JAR 18(1), p.105

lemma $((\exists x. A(x) \ \& \ (\forall y. C(y) \dashrightarrow (\forall z. D(x,y,z)))) \dashrightarrow \$
 $(\exists w. C(w) \ \& \ (\forall y. C(y) \dashrightarrow (\forall z. D(w,y,z)))) \$
 $\& \$
 $(\forall w. C(w) \ \& \ (\forall u. C(u) \dashrightarrow (\forall v. D(w,u,v))) \dashrightarrow \$
 $(\forall y \ z. \$
 $(C(y) \ \& \ P(y,z) \dashrightarrow Q(w,y,z) \ \& \ OO(w,g)) \ \& \$
 $(C(y) \ \& \ \sim P(y,z) \dashrightarrow Q(w,y,z) \ \& \ OO(w,b))) \$
 $\& \$
 $((\exists w. C(w) \ \& \ (\forall y. (C(y) \ \& \ P(y,y) \dashrightarrow Q(w,y,y) \ \& \ OO(w,g)) \ \& \$
 $(C(y) \ \& \ \sim P(y,y) \dashrightarrow Q(w,y,y) \ \& \ OO(w,b)))) \$
 $\dashrightarrow \$
 $(\exists v. C(v) \ \& \ (\forall y. (C(y) \ \& \ P(y,y) \dashrightarrow P(v,y) \ \& \ OO(v,g)) \ \& \$
 $(C(y) \ \& \ \sim P(y,y) \dashrightarrow P(v,y) \ \& \ OO(v,b)))) \$
 $\dashrightarrow \$
 $((\exists v. C(v) \ \& \ (\forall y. (C(y) \ \& \ P(y,y) \dashrightarrow P(v,y) \ \& \ OO(v,g)) \ \& \$
 $(C(y) \ \& \ \sim P(y,y) \dashrightarrow P(v,y) \ \& \ OO(v,b)))) \$
 $\dashrightarrow \$
 $(\exists u. C(u) \ \& \ (\forall y. (C(y) \ \& \ P(y,y) \dashrightarrow \sim P(u,y)) \ \& \$
 $(C(y) \ \& \ \sim P(y,y) \dashrightarrow P(u,y) \ \& \ OO(u,b)))) \$
 $\dashrightarrow \$
 $\sim (\exists x. A(x) \ \& \ (\forall y. C(y) \dashrightarrow (\forall z. D(x,y,z)))) \$
 $\langle proof \rangle$

Challenge found on info-hol

lemma $\forall x. \exists v w. \forall y z. P(x) \ \& \ Q(y) \dashv\vdash (P(v) \mid R(w)) \ \& \ (R(z) \dashv\vdash Q(v))$
<proof>

Attributed to Lewis Carroll by S. G. Pulman. The first or last assumption can be deleted.

lemma $(\forall x. \text{honest}(x) \ \& \ \text{industrious}(x) \dashv\vdash \text{healthy}(x)) \ \& \sim (\exists x. \text{grocer}(x) \ \& \ \text{healthy}(x)) \ \& \ (\forall x. \text{industrious}(x) \ \& \ \text{grocer}(x) \dashv\vdash \text{honest}(x)) \ \& \ (\forall x. \text{cyclist}(x) \dashv\vdash \text{industrious}(x)) \ \& \ (\forall x. \sim \text{healthy}(x) \ \& \ \text{cyclist}(x) \dashv\vdash \sim \text{honest}(x)) \dashv\vdash (\forall x. \text{grocer}(x) \dashv\vdash \sim \text{cyclist}(x))$
<proof>

end

11 First-Order Logic: propositional examples (classical version)

theory *Propositional-Cla*
imports *FOL*
begin

commutative laws of $\&$ and \mid

lemma $P \ \& \ Q \dashv\vdash Q \ \& \ P$
<proof>

lemma $P \mid Q \dashv\vdash Q \mid P$
<proof>

associative laws of $\&$ and \mid

lemma $(P \ \& \ Q) \ \& \ R \dashv\vdash P \ \& \ (Q \ \& \ R)$
<proof>

lemma $(P \mid Q) \mid R \dashv\vdash P \mid (Q \mid R)$
<proof>

distributive laws of $\&$ and \mid

lemma $(P \ \& \ Q) \mid R \dashv\vdash (P \mid R) \ \& \ (Q \mid R)$
<proof>

lemma $(P \mid R) \ \& \ (Q \mid R) \dashv\vdash (P \ \& \ Q) \mid R$

$\langle proof \rangle$

lemma $(P \mid Q) \ \& \ R \ \multimap (P \ \& \ R) \mid (Q \ \& \ R)$
 $\langle proof \rangle$

lemma $(P \ \& \ R) \mid (Q \ \& \ R) \ \multimap (P \mid Q) \ \& \ R$
 $\langle proof \rangle$

Laws involving implication

lemma $(P \multimap R) \ \& \ (Q \multimap R) \ \leftrightarrow (P \mid Q \multimap R)$
 $\langle proof \rangle$

lemma $(P \ \& \ Q \multimap R) \ \leftrightarrow (P \multimap (Q \multimap R))$
 $\langle proof \rangle$

lemma $((P \multimap R) \multimap R) \multimap ((Q \multimap R) \multimap R) \multimap (P \ \& \ Q \multimap R) \multimap R$
 $\langle proof \rangle$

lemma $\sim(P \multimap R) \multimap \sim(Q \multimap R) \multimap \sim(P \ \& \ Q \multimap R)$
 $\langle proof \rangle$

lemma $(P \multimap Q \ \& \ R) \ \leftrightarrow (P \multimap Q) \ \& \ (P \multimap R)$
 $\langle proof \rangle$

Propositions-as-types

— The combinator K

lemma $P \multimap (Q \multimap P)$
 $\langle proof \rangle$

lemma $(P \multimap Q \multimap R) \multimap (P \multimap Q) \multimap (P \multimap R)$
 $\langle proof \rangle$

lemma $(P \multimap Q) \mid (P \multimap R) \multimap (P \multimap Q \mid R)$
 $\langle proof \rangle$

lemma $(P \multimap Q) \multimap (\sim Q \multimap \sim P)$
 $\langle proof \rangle$

Schwichtenberg's examples (via T. Nipkow)

lemma *stab-imp*: $((Q \multimap R) \multimap R) \multimap Q \multimap (((P \multimap Q) \multimap R) \multimap R) \multimap P \multimap Q$
 $\langle proof \rangle$

lemma *stab-to-peirce*:
 $((P \multimap R) \multimap R) \multimap P \multimap (((Q \multimap R) \multimap R) \multimap Q)$
 $\multimap ((P \multimap Q) \multimap P) \multimap P$
 $\langle proof \rangle$

lemma *peirce-imp1*: $((Q \multimap R) \multimap Q) \multimap Q$
 $\multimap ((P \multimap Q) \multimap R) \multimap P \multimap Q \multimap P \multimap Q$
 $\langle proof \rangle$

lemma *peirce-imp2*: $((P \multimap R) \multimap P) \multimap P \multimap ((P \multimap Q \multimap R) \multimap P) \multimap P$
 $\langle proof \rangle$

lemma *mints*: $((((P \multimap Q) \multimap P) \multimap P) \multimap Q) \multimap Q$
 $\langle proof \rangle$

lemma *mints-solovev*: $(P \multimap (Q \multimap R) \multimap Q) \multimap ((P \multimap Q) \multimap R) \multimap R$
 $\langle proof \rangle$

lemma *tatsuta*: $((P7 \multimap P1) \multimap P10) \multimap P4 \multimap P5$
 $\multimap (((P8 \multimap P2) \multimap P9) \multimap P3 \multimap P10)$
 $\multimap (P1 \multimap P8) \multimap P6 \multimap P7$
 $\multimap (((P3 \multimap P2) \multimap P9) \multimap P4)$
 $\multimap (P1 \multimap P3) \multimap (((P6 \multimap P1) \multimap P2) \multimap P9) \multimap P5$
 $\langle proof \rangle$

lemma *tatsuta1*: $((P8 \multimap P2) \multimap P9) \multimap P3 \multimap P10$
 $\multimap (((P3 \multimap P2) \multimap P9) \multimap P4)$
 $\multimap (((P6 \multimap P1) \multimap P2) \multimap P9)$
 $\multimap (((P7 \multimap P1) \multimap P10) \multimap P4 \multimap P5)$
 $\multimap (P1 \multimap P3) \multimap (P1 \multimap P8) \multimap P6 \multimap P7 \multimap P5$
 $\langle proof \rangle$

end

12 First-Order Logic: quantifier examples (classical version)

theory *Quantifiers-Cla*
imports *FOL*
begin

lemma $(ALL\ x\ y.\ P(x,y)) \multimap (ALL\ y\ x.\ P(x,y))$
 $\langle proof \rangle$

lemma $(EX\ x\ y.\ P(x,y)) \multimap (EX\ y\ x.\ P(x,y))$
 $\langle proof \rangle$

lemma $(ALL\ x.\ P(x)) \mid (ALL\ x.\ Q(x)) \multimap (ALL\ x.\ P(x) \mid Q(x))$
 $\langle proof \rangle$

lemma $(ALL\ x.\ P \multimap Q(x)) \multimap (P \multimap (ALL\ x.\ Q(x)))$
 $\langle proof \rangle$

lemma $(ALL\ x.\ P(x) \dashv\vdash Q) \Leftrightarrow ((EX\ x.\ P(x)) \dashv\vdash Q)$
 $\langle proof \rangle$

Some harder ones

lemma $(EX\ x.\ P(x) \mid Q(x)) \Leftrightarrow (EX\ x.\ P(x)) \mid (EX\ x.\ Q(x))$
 $\langle proof \rangle$

lemma $(EX\ x.\ P(x) \& Q(x)) \dashv\vdash (EX\ x.\ P(x)) \ \& \ (EX\ x.\ Q(x))$
 $\langle proof \rangle$

Basic test of quantifier reasoning

— TRUE

lemma $(EX\ y.\ ALL\ x.\ Q(x,y)) \dashv\vdash (ALL\ x.\ EX\ y.\ Q(x,y))$
 $\langle proof \rangle$

lemma $(ALL\ x.\ Q(x)) \dashv\vdash (EX\ x.\ Q(x))$
 $\langle proof \rangle$

The following should fail, as they are false!

lemma $(ALL\ x.\ EX\ y.\ Q(x,y)) \dashv\vdash (EX\ y.\ ALL\ x.\ Q(x,y))$
 $\langle proof \rangle$

lemma $(EX\ x.\ Q(x)) \dashv\vdash (ALL\ x.\ Q(x))$
 $\langle proof \rangle$

lemma $P(?a) \dashv\vdash (ALL\ x.\ P(x))$
 $\langle proof \rangle$

lemma $(P(?a) \dashv\vdash (ALL\ x.\ Q(x))) \dashv\vdash (ALL\ x.\ P(x) \dashv\vdash Q(x))$
 $\langle proof \rangle$

Back to things that are provable ...

lemma $(ALL\ x.\ P(x) \dashv\vdash Q(x)) \ \& \ (EX\ x.\ P(x)) \dashv\vdash (EX\ x.\ Q(x))$
 $\langle proof \rangle$

lemma $(P \dashv\vdash (EX\ x.\ Q(x))) \ \& \ P \dashv\vdash (EX\ x.\ Q(x))$
 $\langle proof \rangle$

lemma $(ALL\ x.\ P(x) \dashv\vdash Q(f(x))) \ \& \ (ALL\ x.\ Q(x) \dashv\vdash R(g(x))) \ \& \ P(d) \dashv\vdash R(?a)$
 $\langle proof \rangle$

lemma $(ALL\ x.\ Q(x)) \dashv\vdash (EX\ x.\ Q(x))$
 $\langle proof \rangle$

Some slow ones

— Principia Mathematica *11.53

lemma $(ALL\ x\ y.\ P(x) \dashv\vdash Q(y)) \Leftrightarrow ((EX\ x.\ P(x)) \dashv\vdash (ALL\ y.\ Q(y)))$
 $\langle proof \rangle$

lemma $(EX\ x\ y.\ P(x) \ \&\ Q(x,y)) \leftrightarrow (EX\ x.\ P(x) \ \&\ (EX\ y.\ Q(x,y)))$
 $\langle proof \rangle$

lemma $(EX\ y.\ ALL\ x.\ P(x) \dashrightarrow Q(x,y)) \dashrightarrow (ALL\ x.\ P(x) \dashrightarrow (EX\ y.\ Q(x,y)))$
 $\langle proof \rangle$

end

theory *Miniscope*
imports *FOL*
begin

lemmas *ccontr* = *FalseE* [*THEN classical*]

12.1 Negation Normal Form

12.1.1 de Morgan laws

lemma *demorgans*:
 $\sim(P \ \&\ Q) \leftrightarrow \sim P \mid \sim Q$
 $\sim(P \mid Q) \leftrightarrow \sim P \ \&\ \sim Q$
 $\sim\sim P \leftrightarrow P$
 $!!P.\ \sim(ALL\ x.\ P(x)) \leftrightarrow (EX\ x.\ \sim P(x))$
 $!!P.\ \sim(EX\ x.\ P(x)) \leftrightarrow (ALL\ x.\ \sim P(x))$
 $\langle proof \rangle$

lemma *nnf-simps*:
 $(P \dashrightarrow Q) \leftrightarrow (\sim P \mid Q)$
 $\sim(P \dashrightarrow Q) \leftrightarrow (P \ \&\ \sim Q)$
 $(P \leftrightarrow Q) \leftrightarrow (\sim P \mid Q) \ \&\ (\sim Q \mid P)$
 $\sim(P \leftrightarrow Q) \leftrightarrow (P \mid Q) \ \&\ (\sim P \mid \sim Q)$
 $\langle proof \rangle$

12.1.2 Pushing in the existential quantifiers

lemma *ex-simps*:
 $(EX\ x.\ P) \leftrightarrow P$
 $!!P\ Q.\ (EX\ x.\ P(x) \ \&\ Q) \leftrightarrow (EX\ x.\ P(x)) \ \&\ Q$
 $!!P\ Q.\ (EX\ x.\ P \ \&\ Q(x)) \leftrightarrow P \ \&\ (EX\ x.\ Q(x))$
 $!!P\ Q.\ (EX\ x.\ P(x) \mid Q(x)) \leftrightarrow (EX\ x.\ P(x)) \mid (EX\ x.\ Q(x))$
 $!!P\ Q.\ (EX\ x.\ P(x) \mid Q) \leftrightarrow (EX\ x.\ P(x)) \mid Q$
 $!!P\ Q.\ (EX\ x.\ P \mid Q(x)) \leftrightarrow P \mid (EX\ x.\ Q(x))$

$\langle proof \rangle$

12.1.3 Pushing in the universal quantifiers

lemma *all-simps*:

$(ALL\ x.\ P) <-> P$
 $!!P\ Q.\ (ALL\ x.\ P(x) \ \&\ Q(x)) <-> (ALL\ x.\ P(x)) \ \&\ (ALL\ x.\ Q(x))$
 $!!P\ Q.\ (ALL\ x.\ P(x) \ \&\ Q) <-> (ALL\ x.\ P(x)) \ \&\ Q$
 $!!P\ Q.\ (ALL\ x.\ P \ \&\ Q(x)) <-> P \ \&\ (ALL\ x.\ Q(x))$
 $!!P\ Q.\ (ALL\ x.\ P(x) \ | \ Q) <-> (ALL\ x.\ P(x)) \ | \ Q$
 $!!P\ Q.\ (ALL\ x.\ P \ | \ Q(x)) <-> P \ | \ (ALL\ x.\ Q(x))$
 $\langle proof \rangle$

lemmas *mini-simps* = *demorgans* *nnf-simps* *ex-simps* *all-simps*

$\langle ML \rangle$

end

13 First-Order Logic: the 'if' example

theory *If* imports *FOL* begin

constdefs

$if :: [o,o,o] \Rightarrow o$
 $if(P,Q,R) == P \ \&\ Q \ | \ \sim P \ \&\ R$

lemma *ifI*:

$[| P \Rightarrow Q; \sim P \Rightarrow R |] \Rightarrow if(P,Q,R)$
 $\langle proof \rangle$

lemma *ifE*:

$[| if(P,Q,R); [| P; Q |] \Rightarrow S; [| \sim P; R |] \Rightarrow S |] \Rightarrow S$
 $\langle proof \rangle$

lemma *if-commute*: $if(P, if(Q,A,B), if(Q,C,D)) <-> if(Q, if(P,A,C), if(P,B,D))$
 $\langle proof \rangle$

Trying again from the beginning in order to use *blast*

declare *ifI* [*intro!*]

declare *ifE* [*elim!*]

lemma *if-commute*: $if(P, if(Q,A,B), if(Q,C,D)) <-> if(Q, if(P,A,C), if(P,B,D))$
 $\langle proof \rangle$

lemma $if(if(P,Q,R), A, B) <-> if(P, if(Q,A,B), if(R,A,B))$
 $\langle proof \rangle$

Trying again from the beginning in order to prove from the definitions

lemma $if(if(P,Q,R), A, B) <-> if(P, if(Q,A,B), if(R,A,B))$
 $\langle proof \rangle$

An invalid formula. High-level rules permit a simpler diagnosis

lemma $if(if(P,Q,R), A, B) <-> if(P, if(Q,A,B), if(R,B,A))$
 $\langle proof \rangle$

Trying again from the beginning in order to prove from the definitions

lemma $if(if(P,Q,R), A, B) <-> if(P, if(Q,A,B), if(R,B,A))$
 $\langle proof \rangle$

end

14 Example of Declaring an Oracle

theory *Iff-Oracle*
imports *FOL*
begin

14.1 Oracle declaration

This oracle makes tautologies of the form $P <-> P <-> P <-> P$. The length is specified by an integer, which is checked to be even and positive.

$\langle ML \rangle$

14.2 Oracle as low-level rule

$\langle ML \rangle$

These oracle calls had better fail.

$\langle ML \rangle$

14.3 Oracle as proof method

$\langle ML \rangle$

lemma $A <-> A$
 $\langle proof \rangle$

lemma $A <-> A <-> A <-> A <-> A <-> A <-> A <-> A <-> A$
 $\langle proof \rangle$

lemma $A <-> A <-> A <-> A <-> A$

$\langle proof \rangle$

lemma A
 $\langle proof \rangle$

end