FOPL and FOPLN Backward Proofs of Equivalence Laws

Chunyan Mu, Bill Stoddart, and Steve Dunne
Teesside University
c.mu@tees.ac.uk

This report presents First Order Predicate Logic (FOPL) and First Order Predicate Logic with NULL (FOPLN) tree proofs of a set of equivalence laws, and is a supplementary document to [SDM22].

Equivalence Laws

deMorgan
1. $(\neg\forall\ x \bullet P) \Leftrightarrow (\exists\ x \bullet \neg P)$
2. $(\forall\ x \bullet \neg\ P) \Leftrightarrow \neg(\exists\ x \bullet P)$

Change of order of bound variables
3. $(\forall\ x \bullet \forall\ y \bullet P) \Leftrightarrow (\forall\ y \bullet \forall\ x \bullet P)$
4. $(\exists\ x \bullet \exists\ y \bullet P) \Leftrightarrow (\exists y \bullet \exists\ x \bullet P)$

Splitting
5. $(\forall\ x \bullet P \land Q) \Leftrightarrow (\forall\ x \bullet P) \land (\forall\ x \bullet Q)$
6. $(\exists\ x \bullet P \lor Q) \Leftrightarrow (\exists\ x \bullet P) \lor (\exists\ x \bullet Q)$
7. $(\exists\ x \bullet P \land Q) \Rightarrow (\exists\ x \bullet P) \land (\exists\ x \bullet Q)$

Assuming x is not free in Q:
8. $(\forall x \bullet P) \land Q \Leftrightarrow \forall x \bullet (P \land Q)$
9. $(\forall x \bullet P) \lor Q \Leftrightarrow \forall x \bullet (P \lor Q)$
10. $(\exists x \bullet P) \land Q \Leftrightarrow \exists x \bullet (P \land Q)$
11. $(\exists x \bullet P) \lor Q \Leftrightarrow \exists x \bullet (P \lor Q)$
12. $\forall x \bullet (Q \Rightarrow P) \Leftrightarrow (Q \Rightarrow \forall x \bullet P)$
13. $\exists x \bullet (P \Rightarrow Q) \Leftrightarrow ((\forall x \bullet P) \Rightarrow Q)$
14. $\forall x \bullet (P \Rightarrow Q) \Leftrightarrow ((\exists x \bullet P) \Rightarrow Q)$
15. $\exists x \bullet (Q \Rightarrow P) \Leftrightarrow (Q \Rightarrow \exists x \bullet P)$

Change of bound variable name, assuming P admits y for x.*
16. $(\forall\ x \bullet P) \Leftrightarrow (\forall\ y \bullet P[y/x])$
17. $(\exists\ x \bullet P) \Leftrightarrow (\exists\ y \bullet P[y/x])$

Monotonicity.
18 $(\forall\ x \bullet P \Rightarrow Q) \Rightarrow ((\forall\ x \bullet P) \Rightarrow (\forall\ x \bullet Q))$
19 $(\forall\ x \bullet P \Rightarrow Q) \Rightarrow ((\exists\ x \bullet P) \Rightarrow (\exists\ x \bullet Q))$

Equivalence
20 $(\forall\ x \bullet P \Leftrightarrow Q) \Rightarrow ((\forall\ x \bullet P) \Leftrightarrow (\forall\ x \bullet Q))$
21 $(\forall\ x \bullet P \Leftrightarrow Q) \Rightarrow ((\exists\ x \bullet P) \Leftrightarrow (\exists\ x \bullet Q))$
22 $(\exists\ x \bullet P \Rightarrow Q) \Leftrightarrow ((\forall\ x \bullet P) \Rightarrow (\exists\ x \bullet Q))$
One point rules, given premise $\delta(E)$

23. $(\forall x \bullet x=E \Rightarrow P) \Leftrightarrow P[E/x]$
24. $(\exists x \bullet x=E \wedge P) \Leftrightarrow P[E/x]$

* P admits y for x if every free occurrence of x in P becomes a free occurrence of y in P[y/x]

First Order Predicate Logic with Null (FOPLN): proof rules

$$\frac{HYP \vdash \delta(E)}{HYP \vdash E \neq null} \quad \text{NULL}$$

$$\frac{HYP \vdash \forall x \bullet P \qquad HYP \vdash \delta(E)}{HYP \vdash P(E/x)} \quad \text{where E is an arbitrary expression.} \quad \forall\text{-elim}$$

$$\frac{HYP, \delta(\alpha) \vdash P[\alpha/x]}{HYP \vdash \forall x \bullet P} \quad \text{where } \alpha \text{ is fresh.} \quad \forall\text{-intro}$$

$$\frac{HYP, \delta(\alpha) \vdash \exists x \bullet P \qquad HYP, \delta(\alpha) \vdash P[\alpha/x] \Rightarrow Q}{HYP \vdash Q} \quad \text{where } \alpha \text{ is fresh, } x \backslash Q. \quad \exists\text{-elim}$$

$$\frac{HYP \vdash P[E/x] \qquad HYP \vdash E \neq null}{HYP \vdash \exists x \bullet P} \quad \exists\text{-intro}$$

A derived law to simplify some of the proofs:

$$\frac{HYP \vdash \exists x \bullet P}{HYP, \delta(\alpha) \vdash \exists x \bullet P} \quad \text{Mu}$$

FOPL and FOPLN Proof of the equivalence laws

1. deMorgan I : (¬∀ x • P) ⇔ (∃ x • ¬P)

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>(¬∀ x • P) ⇒ (∃ x • ¬P)
DED
{ ¬∀ x • P}
|
∃ x • ¬P
CONTRA
{¬ (∃ x • ¬P) }
/ \
¬∀ x • P   ∀ x • P
INHYP   ∀-intro
|
P[α/x]
CONTRA
{¬ P[α/x]}
/ \
¬ (∃ x • ¬P)   ∃x • ¬P
INHYP   ∃-intro
|
¬ P[α/x]
INHYP</td>
<td>(¬∀ x • P) ⇒ (∃ x • ¬P)
DED
{¬∀ x • P}
|
∃ x • ¬P
CONTRA
{¬ (∃ x • ¬P)}
/ \
¬∀ x • P   ∀ x • P
INHYP   ∀-intro
{δ(α)}
|
P[α/x]
CONTRA
{¬ P[α/x]}
/ \
¬ (∃ x • ¬P)   ∃x • ¬P
INHYP   ∃-intro
{δ(α)}
/ \
¬ P[α/x]   α≠ null
INHYP   NULL
|
δ(α)
INHYP</td>
</tr>
<tr>
<td>(∃ x • ¬P) ⇒ (¬∀ x • P)
DED
{∃ x • ¬P)
|
¬∀ x • P
∃-elim
/ \
∃ x•¬P   ¬P[α/x] ⇒ (¬∀ x • P)
INHYP   DED
{¬P[α/x]}
|
¬∀ x • P
CONTRA
{∀ x • P}
/ \
¬P[α/x]   P[α/x]
INHYP   ∀-elim
|
∀ x • P
INHYP</td>
<td>(∃ x • ¬P) ⇒ (¬∀ x • P)
DED
{∃ x • ¬P)
|
¬∀ x • P
∃elim
{δ(α)}
/ \
∃ x•¬P   ¬P[α/x] ⇒ (¬∀ x • P)
INHYP   DED
{¬P[α/x]}
|
¬∀ x • P
CONTRA
{∀ x • P}
/ \
¬P[α/x]   P[α/x]
INHYP   ∀-elim
/ \
∀ x • P   δ(α)
INHYP   INHYP</td>
</tr>
</table>

2. deMorgan II: (∀ x • ¬ P) ⇔ ¬(∃ x • P)

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr><td>(∀ x • ¬ P) ⇒ ¬(∃ x • P)
DED
{ ∀ x • ¬ P }
|
¬(∃ x • P)
CONTRA
{∃ x • P}
/ \
∀ x • ¬P   ¬(∀ x • ¬P)
INHYP   ∃-elim
/ \
∃ x • P   P[α/x]⇒¬(∀ x • ¬P)
INHYP   DED
{P[α/x]}
|
¬(∀ x • ¬P)
CONTRA
{∀ x • ¬P}
/ \
P[α/x]   ¬P[α/x]}
INHYP   ∀-elim
|
∀ x • ¬P
INHYP</td><td>(∀ x • ¬ P) ⇒ ¬(∃ x • P)
DED
{ ∀ x • ¬ P }
|
¬(∃ x • P)
CONTRA
{∃ x • P}
/ \
∀ x • ¬P   ¬(∀ x • ¬P)
INHYP   ∃-elim
{δ(α)}
/ \
∃ x • P   P[α/x]⇒¬(∀ x • ¬P)
INHYP   DED
{P[α/x]}
|
¬(∀ x • ¬P)
CONTRA
{∀ x • ¬P}
/ \
P[α/x]}.   ¬P[α/x]}
INHYP   ∀-elim
/ \
∀ x • ¬P   δ(α)
INHYP   INHYP</td></tr>
<tr><td>¬(∃ x • P) ⇒ (∀ x • ¬ P)
DED
{¬(∃ x • P)}
|
∀ x • ¬ P
∀-intro
|
¬P[α/x]
CONTRA
{P[α/x]}
/ \
¬(∃ x • P)   ∃ x • P
INHYP   ∃-intro
|
P[α/x]
INHYP</td><td>¬(∃ x • P) ⇒ (∀ x • ¬ P)
DED
{¬(∃ x • P)}
|
∀ x • ¬ P
∀-intro
{δ(α)}
|
¬P[α/x]
CONTRA
{P[α/x]}
/ \
¬(∃ x • P)   ∃ x • P
INHYP   ∃-intro
{δ(α)}
/ \
P[α/x]   α≠null
INHYP   NULL
|
δ(α)
INHYP</td></tr>
</table>

3. Change of order of bound variables I: (∀ x • ∀ y • P) ⇔ (∀ y • ∀ x • P)

FOPL	FOPLN
(∀ x • ∀ y • P) ⇒ (∀ y • ∀ x • P) DED {∀ x • ∀ y • P } | ∀ y • ∀ x • P ∀-intro | ∀ x • P [α/y] ∀-intro | P[α/y][α’/x] i.e. P[α’/x][α/y] ∀-elim | ∀ y • P [α’/x] ∀-elim | ∀ x • ∀ y • P INHYP	(∀ x • ∀ y • P) ⇒ (∀ y • ∀ x • P) DED {∀ x • ∀ y • P } | ∀ y • ∀ x • P ∀-intro { δ(α)} | ∀ x • P [α/y] ∀-intro {δ(α’)} | P[α/y][α’/x] i.e. P[α’/x][α/y] ∀-elim / \\ ∀ y • P [α’/x] δ(α) ∀-elim INHYP / \\ ∀ x • ∀ y • P δ(α’) INHYP INHYP
(∀ y • ∀ x • P) ⇒ (∀ x • ∀ y • P) Similar to the above	(∀ y • ∀ x • P) ⇒ (∀ x • ∀ y • P) Similar to the above

4. Change of order of bound variables II: (∃ x • ∃ y • P) ⇔ (∃y • ∃ x • P)

FOPL	**FOPLN**
(∃ x • ∃ y • P) ⇒ (∃y • ∃ x • P) DED { ∃ x • ∃ y • P } | ∃y • ∃ x • P ∃-elim / \ ∃x•∃y•P ∃y•P[α/x] ⇒ ∃y•∃x•P INHYP DED {∃y•P[α/x]} | ∃y•∃x•P ∃-elim / \ ∃y•P[α/x] P[α/x][α'/y] ⇒ ∃y•∃x•P INHYP DED { P[α/x][α'/y] } i.e. { P[α'/y][α/x] } | ∃y•∃x•P ∃-intro | ∃x•P [α'/y] ∃-intro | P[α'/y][α/x] INHYP	(∃ x • ∃ y • P) ⇒ (∃y • ∃ x • P) DED { ∃ x • ∃ y • P } | ∃y • ∃ x • P ∃-elim {δ(α)} / \ ∃x•∃y•P ∃y•P[α/x] ⇒ ∃y•∃x•P INHYP DED {∃y•P[α/x]} | ∃y•∃x•P ∃-elim {δ(α')} / \ ∃y•P[α/x] P[α/x][α'/y] ⇒ ∃y•∃x•P INHYP DED { P[α/x][α'/y] } i.e. { P[α'/y][α/x] } | ∃y•∃x•P ∃-intro {δ(α')} / \ ∃x•P [α'/y] α'≠ null ∃-intro NULL {δ(α)} / \ | P[α'/y][α/x] α≠ null δ(α') INHYP NULL INHYP | δ(α) INHYP
(∃y • ∃ x • P) ⇒ (∃ x • ∃ y • P) Similar to the above	(∃y • ∃ x • P) ⇒ (∃ x • ∃ y • P) Similar to the above

5. Splitting I: (∀ x • P ∧ Q) ⇔ (∀ x • P) ∧ (∀ x • Q)

FOPL	**FOPLN**
(∀ x • P ∧ Q) ⇒ (∀ x • P) ∧ (∀ x • Q) DED { ∀ x • P ∧ Q } | (∀ x • P) ∧ (∀ x • Q) ∧-intro / \\ ∀ x • P ∀ x • Q ∀ intro similar to the left | P[α/x] ∧-elim | P[α/x] ∧ Q[α/x] i.e. (P∧Q)[α/x] ∀-elim | ∀ x • P ∧ Q INHYP	(∀ x • P ∧ Q) ⇒ (∀ x • P) ∧ (∀ x • Q) DED { ∀ x • P ∧ Q } | (∀ x • P) ∧ (∀ x • Q) ∧-intro / \\ ∀ x • P ∀ x • Q ∀ intro similar to the left {δ(α)} | P[α/x] ∧elim | P[α/x] ∧ Q[α/x] i.e. (P∧Q)[α/x] ∀-elim / \\ ∀ x • P ∧ Q α≠ null INHYP NULL | δ(α) INHYP
(∀ x • P) ∧ (∀ x • Q) ⇒ (∀ x • P ∧ Q) DED { (∀ x • P) ∧ (∀ x • Q) } | ∀ x • P ∧ Q ∀-intro | (P ∧ Q) [α/x] i.e. P[α/x] ∧ Q[α/x] ∧-intro / \\ P[α/x] Q[α/x] ∀-elim similar to the left | ∀ x • P ∧-elim | (∀ x • P) ∧ (∀ x • Q) INHYP	(∀ x • P) ∧ (∀ x • Q) ⇒ (∀ x • P ∧ Q) DED { (∀ x • P) ∧ (∀ x • Q) } | ∀ x • P ∧ Q ∀-intro {δ(α)} | (P ∧ Q) [α/x] i.e. P[α/x] ∧ Q[α/x] ∧-intro / \\ P[α/x] Q[α/x] ∀elim similar to the left {δ(α)} / \\ ∀ x • P α≠ null ∧-elim NULL | | (∀ x • P) ∧ (∀ x • Q) δ(α) INHYP INHYP

6. Splitting II: (∃ x • P ∨ Q) ⇔ (∃ x • P) ∨ (∃ x • Q)

FOPL	FOPLN
(∃ x • P ∨ Q) ⇒ (∃ x • P) ∨ (∃ x • Q) DED { ∃ x • P ∨ Q } | (∃ x • P) ∨ (∃ x • Q) ∃-elim / \\ ∃ x•P∨Q (PVQ)[α/x] ⇒ (∃x•P)∨(∃x•Q) INHYP DED { (P∨Q)[α/x] } i.e. {P[α/x] ∨ Q[α/x] } | (∃x•P)∨(∃x•Q) ∨-elim / | \\ P[α/x]∨Q[α/x] P[α/x]⇒(∃x•P)∨(∃x•Q) Q[α/x]⇒... INHYP DED similar { P[α/x] } | (∃x•P) ∨ (∃x•Q) ∨-intro | ∃x•P ∃-intro | P [α/x] INHYP	(∃ x • P ∨ Q) ⇒ (∃ x • P) ∨ (∃ x • Q) DED { ∃ x • P ∨ Q } | (∃ x • P) ∨ (∃ x • Q) ∃-elim {δ(α)} / \\ ∃ x•P∨Q (P∨Q)[α/x] ⇒ (∃x•P)∨(∃x•Q) INHYP DED { (P∨Q)[α/x] } i.e. {P[α/x] ∨ Q[α/x] } | (∃x•P)∨(∃x•Q) ∨-elim / | \\ P[α/x]∨Q[α/x] P[α/x]⇒(∃x•P)∨(∃x•Q). Q[α/x]⇒... INHYP DED similar { P[α/x] } | (∃x•P) ∨ (∃x•Q) ∨-intro | ∃x•P ∃-intro {δ(α)} / \\ P [α/x] α≠ null INHYP NULL | δ(α) INHYP
(∃ x • P) ∨ (∃ x • Q) ⇒ (∃ x • P ∨ Q) DED { (∃ x • P) ∨ (∃ x • Q) } | ∃ x • P ∨ Q ∨-elim / | \\ (∃x•P)∨(∃x•Q). ∃x•P⇒(∃x•P∨Q) ∃x•Q⇒(∃x•P∨Q) INHYP DED similar { ∃x • P } | ∃x • P∨Q ∃-elim / \\ ∃x•P P[α/x] ⇒ ∃x • P∨Q INHYP DED { P[α/x] } | ∃x • P∨Q ∃-intro | (P∨Q)[α/x] i.e. P[α/x]∨Q[α/x] ∨-intro | P[α/x] INHYP	(∃ x • P) ∨ (∃ x • Q) ⇒ (∃ x • P ∨ Q) DED { (∃ x • P) ∨ (∃ x • Q) } | ∃ x • P ∨ Q ∨-elim / | \\ (∃x•P)∨(∃x•Q) ∃x•P⇒(∃x•P∨Q) ∃x•Q⇒(∃x•P∨Q) INHYP DED similar { ∃x • P } | ∃x • P∨Q ∃-elim {δ(α)} / \\ ∃x•P P[α/x] ⇒ ∃x • P∨Q INHYP DED { P[α/x] } | ∃x • P∨Q ∃-intro / \\ (P∨Q)[α/x] α≠ null i.e. P[α/x]∨Q[α/x] NULL ∨-intro | | δ(α) P[α/x] INHYP INHYP

7. Splitting III: (∃ x • P ∧ Q) ⇒ (∃ x • P) ∧ (∃ x • Q)

FOPL	**FOPLN**
(∃ x • P ∧ Q) ⇒ (∃ x • P) ∧ (∃ x • Q) { ∃ x • P ∧ Q } | (∃ x • P) ∧ (∃ x • Q) ∧-intro / \\ ∃ x • P ∃ x • Q ∃-elim similar / \\ ∃ x•P∧Q (P∧Q)[α/x] ⇒ P[α/x] INHYP DED { P∧Q)[α/x] } i.e. { P[α/x] ∧ Q[α/x] } | ∃x•P ∃-intro | P[α/x] ∧-elim | P[α/x] ∧ Q[α/x] INHYP	(∃ x • P ∧ Q) ⇒ (∃ x • P) ∧ (∃ x • Q) { ∃ x • P ∧ Q } | (∃ x • P) ∧ (∃ x • Q) ∧-intro / \\ ∃ x • P ∃ x • Q ∃-elim similar {δ(α)} / \\ ∃ x•P∧Q (P∧Q)[α/x] ⇒ P[α/x] INHYP DED { P∧Q)[α/x] } i.e. { P[α/x] ∧ Q[α/x] } | ∃x•P ∃-intro / \\ P[α/x] α≠ null ∧-elim NULL | | P[α/x]∧Q[α/x] δ(α) INHYP INHYP

8. Assuming x is not free in Q: ((∀x•P) ∧ Q) ⇔ ∀x• (P ∧ Q)

FOPL	**FOPLN**
((∀x•P) ∧ Q) ⇒ ∀x• (P ∧ Q) DED { (∀x•P) ∧ Q } | ∀x• (P ∧ Q) ∀-intro | (P ∧ Q) [α/x] i.e. P[α/x] ∧ Q ∧-intro / \\ P[α/x] Q ∀-elim ∧-elim | | ∀x•P (∀x•P) ∧ Q ∧-elim INHYP | (∀x•P) ∧ Q INHYP	((∀x•P) ∧ Q) ⇒ ∀x• (P ∧ Q) DED { (∀x•P) ∧ Q } | ∀x• (P ∧ Q) ∀-intro { δ(α)} | (P ∧ Q) [α/x] i.e. P[α/x] ∧ Q ∧-intro / \\ P[α/x] Q ∀-elim ∧-elim / \\ | ∀x•P δ(α) (∀x•P) ∧ Q ∧-elim INHYP INHYP (∀x•P) ∧ Q INHYP
∀x• (P ∧ Q) ⇒ ((∀x•P) ∧ Q) DED { ∀x• (P ∧ Q) } | (∀x•P) ∧ Q ∧-intro / \\ ∀x•P Q ∀-intro ∧-elim | | P[α/x] P[α/x] ∧ Q ∧-elim i.e. (P∧ Q) [α/x] | ∀-elim P[α/x] ∧ Q[α/x] | i.e. (P ∧ Q) [α/x] ∀x• (P ∧ Q) ∀-elim INHYP | ∀x• (P ∧ Q) INHYP	∀x• (P ∧ Q) ⇒ ((∀x•P) ∧ Q) DED { ∀x• (P ∧ Q) } | (∀x•P) ∧ Q ∧-intro / \\ ∀x•P Q i.e. ∀x•Q ∀-intro ∀-intro { δ(α)} { δ(α)} | | P[α/x] Q [α/x] ∧-elim ∧-elim | | P[α/x] ∧ Q[α/x] P[α/x] ∧ Q[α/x] i.e. (P ∧ Q) [α/x] i.e. (P ∧ Q) [α/x] ∀-elim ∀-elim / \\ / \\ ∀x• (P ∧ Q) δ(α) ∀x• (P ∧ Q) δ(α) INHYP INHYP INHYP INHYP

9. Assuming x is not free in Q: (∀x•P) ∨ Q ⇔ ∀x•(P ∨ Q)

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>((∀x•P) ∨ Q) ⇒ (∀x•(P ∨ Q))
DED
{(∀x•P) ∨ Q}
|
∀x•(P ∨ Q)
∀-intro
|
(P ∨ Q) [α/x]
i.e. P[α/x] ∨ Q
∨-elim
/ | \
(∀x•P)∨Q ∀x•P⇒ (P[α/x]∨Q) Q⇒ (P[α/x] ∨ Q)
INHYP DED DED
{∀x•P } {Q}
| |
P[α/x]∨Q P[α/x]∨Q
∨-intro ∨-intro
| |
P[α/x] Q
∀-elim INHYP
|
∀x•P
INHYP</td>
<td>((∀x•P) ∨ Q) ⇒ (∀x•(P ∨ Q))
DED
{(∀x•P) ∨ Q}
|
∀x•(P ∨ Q)
∀-intro
{δ(α)}
|
(P ∨ Q) [α/x]
i.e. P[α/x] ∨ Q
∨-elim
/ | \
(∀x•P)∨Q ∀x•P⇒ (P[α/x]∨Q) Q⇒ (P[α/x] ∨ Q)
INHYP DED DED
{∀x•P } {Q}
| |
P[α/x]∨Q P[α/x]∨Q
∨-intro ∨-intro
| |
P[α/x] Q
∀-elim INHYP
/ \
∀x•P δ(α)
INHYP INHYP</td>
</tr>
<tr>
<td>(∀x•(P ∨ Q)) ⇒ ((∀x•P) ∨ Q)
DED
{ ∀x•(P ∨ Q) }
|
(∀x•P) ∨ Q
∨-elim
/ | \
Q ∨¬Q Q⇒ ((∀x•P) ∨ Q) ¬Q⇒((∀x•P) ∨ Q)
INLAW DED DED
{Q} {¬Q}
| |
(∀x•P) ∨ Q (∀x•P) ∨ Q
∨-intro ∨-intro
| |
Q ∀x•P
INHYP ∀-intro
|
P[α/x]
∨-elim
/ | \
P[α/x]⇒P[α/x] Q⇒P[α/x] P[α/x] ∨ Q
DED DED i.e.
(P∨ Q)
[α/x]
{P[α/x]} {Q} ∀-elim
| | |
P[α/x] P[α/x] ∀x•(P∨Q)
INHYP CONTRA. INHYP
{¬P[α/x]}
/ \
Q ¬Q
INHYP INHYP</td>
<td>(∀x•(P ∨ Q)) ⇒ ((∀x•P) ∨ Q)
DED
{ ∀x•(P ∨ Q) }
|
(∀x•P) ∨ Q
∨-elim
/ | \
Q ∨¬Q Q⇒ ((∀x•P) ∨ Q) ¬Q⇒((∀x•P) ∨ Q)
INLAW DED DED
{Q} {¬Q}
| |
(∀x•P) ∨ Q (∀x•P) ∨ Q
∨-intro ∨-intro
| |
Q ∀x•P
INHYP ∀-intro
{δ(α)}
|
P[α/x]
∨-elim
/ | \
P[α/x]⇒P[α/x] Q⇒P[α/x] P[α/x] ∨ Q
DED DED i.e.
(P∨ Q) [α/x]
{P[α/x]} {Q} ∀-elim
| | / \
P[α/x] P[α/x] ∀x•(P∨Q) δ(α)}
INHYP CONTRA. INHYP INHYP
{¬P[α/x]}
/ \
Q ¬Q
INHYP INHYP</td>
</tr>
</table>

10. Assuming x is not free in Q: **((∃x•P) ∧ Q) ⇔ ∃x•(P ∧ Q)**

FOPL

```
   ((∃x•P) ∧ Q) ⇒ ∃x•(P ∧ Q)
               DED
          {(∃x•P) ∧ Q }
               |
          ∃x•(P ∧ Q)
            ∃-intro
               |
          (P∧Q) [α/x]
        i.e. P[α/x] ∧Q
              ∧-intro
         /                    \
     P[α/x]                    Q
     ∃-elim                  ∧-elim
    /       \                  |
∃x•P    P[α/x]⇒P[α/x]     (∃x•P) ∧ Q
∧-elim       DED             INHYP
  |        {P[α/x]}
(∃x•P)∧Q       |
INHYP        P[α/x]
             INHYP
```

FOPLN

```
        ((∃x•P) ∧ Q) ⇒ ∃x•(P ∧ Q)
                   DED
              {(∃x•P) ∧ Q }
                   |
              ∃x•(P ∧ Q)
                   Mu
                 {δ(α)}
                   |
              ∃x•(P ∧ Q)
                 ∃-intro
          /                         \
   (P∧Q) [α/x]                    α≠ null
  i.e. P[α/x] ∧Q                   NULL
      ∧-intro                        |
   /                 \             δ(α)
 P[α/x]               Q            INHYP
 ∃-elim             ∧-elim
  /       \           |
∃x•P  P[α/x]⇒P[α/x] (∃x•P) ∧ Q
∧elim       DED       INHYP
  |       {P[α/x]}
(∃x•P)∧Q      |
INHYP       P[α/x]
            INHYP
```

FOPL

```
              ∃x•(P ∧ Q) ⇒ ((∃x•P) ∧ Q)
                         DED
                    {∃x•(P ∧ Q)}
                         |
                    (∃x•P) ∧ Q
                       ∧-intro
      /                                   \
   ∃x•P                                    Q
  ∃-intro                                ∃elim
     |                              /           \
   P[α/x]                    ∃x•(P∧Q)   (P∧Q)[α/x]⇒Q
   ∃-elim                      INHYP          DED
  /         \                            {P∧Q)[α/x]}
∃x•(P∧Q)  (P∧Q)[α/x]⇒P[α/x]           i.e. {P[α/x]∧Q }
INHYP            DED                          |
            {P∧Q)[α/x]}                       Q
          i.e. {P[α/x]∧Q}                  ∧-elim
                 |                            |
               P[α/x]                    P[α/x]∧Q
               ∧-elim                      INHYP
                 |
              P[α/x]∧Q
               INHYP
```

FOPLN

```
              ∃x•(P ∧ Q) ⇒ ((∃x•P) ∧ Q)
                         DED
                    {∃x•(P ∧ Q)}
                         |
                    (∃x•P) ∧ Q
                       ∧-intro
          /                               \
        ∃x•P                               Q
       ∃-intro                           ∃-elim
       {δ(α)}                            {δ(α)}
      /        \                      /          \
   P[α/x]     α≠ null   ∃x•(P∧Q)   (P∧Q)[α/x]⇒Q
   ∃elim       NULL      INHYP          DED
   /        \    |                  {P∧Q)[α/x]}
  /          \  δ(α)             i.e. {P[α/x]∧Q }
 /            \ INHYP                    |
/              \
∃x•(P∧Q)  (P∧Q)[α/x]⇒P[α/x]              Q
INHYP            DED                  ∧-elim
            {P∧Q)[α/x]}                  |
          i.e. {P[α/x]∧Q}            P[α/x]∧Q
                 |                     INHYP
               P[α/x]
               ∧-elim
                 |
              P[α/x]∧Q
               INHYP
```

11. Assuming x is not free in Q: **(∃x• P) ∨ Q ⇔ ∃x• (P ∨ Q)**

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>((∃x• P) ∨ Q) ⇒ ∃x• (P ∨ Q)
DED
{(∃x• P) ∨ Q }
|
∃x• (P ∨ Q)
∨-elim
/ | \
(∃x• P)⇒∃x•(P∨Q)   Q⇒∃x•(P∨Q)   (∃x• P) ∨ Q
DED   DED   INHYP
{∃x• P}   {Q}
| |
∃x•(P∨Q)   ∃x•(P∨Q)
∃-elim   ∃-intro
/   \ |
∃x•P   P[α/x]⇒∃x•(P∨Q)   (P∨Q)[α/x]
INHYP.   DED   i.e. P[α/x]∨Q
{P[α/x]}   ∨-intro
| |
∃x•(P∨Q)   Q
∃-intro   INHYP
(P∨Q)[α/x]
i.e. P[α/x]∨Q[α/x]
∨-intro
|
P[α/x]}
INHYP</td>
<td>((∃x• P) ∨ Q) ⇒ ∃x• (P ∨ Q)
DED
{ (∃x• P) ∨ Q }
|
∃x• (P ∨ Q)
∨-elim
/ | \
(∃x• P)⇒∃x•(P∨Q)   Q⇒∃x•(P∨Q)   (∃x• P) ∨ Q
DED   DED   INHYP
{∃x• P}   {Q}
| |
∃x•(P∨Q)   ∃x•(P∨Q)
∃-elim   Mu
{δ(α)}   {δ(α)}
/   \ |
∃x• P   P[α/x]⇒∃x•(P∨Q)   ∃x•(P∨Q)
INHYP.   DED   ∃-intro
{P[α/x]}   /   \
| (P∨Q)[α/x]   α≠ null
∃x•(P∨Q)   i.e. P[α/x]∨Q   NULL
∃-intro   ∨-intro |
/   \ | δ(α)
(P∨Q)[α/x]   α≠ null   Q   INHYP
i.e. P[α/x]∨Q[α/x]   NULL   INHYP
∨-intro |
| δ(α)
P[α/x]}   INHYP
INHYP</td>
</tr>
<tr>
<td>∃x• (P ∨ Q) ⇒ ((∃x• P) ∨ Q)
DED
{∃x• (P ∨ Q)}
|
(∃x• P) ∨ Q
∃elim
/   \
∃x• (P ∨ Q)   (P∨Q)[α/x] ⇒((∃x• P) ∨ Q)
DED
{(P∨Q)[α/x]} i.e. {P[α/x] ∨Q}
|
(∃x• P) ∨ Q
∨elim
/ | \
P[α/x]⇒((∃x•P)∨Q)   Q⇒((∃x•P)∨Q)   P[α/x]∨Q
DED   DED   INHYP
{P[α/x]}   {Q}
| |
(∃x• P) ∨ Q   (∃x• P) ∨ Q
∨intro   ∨intro
| |
∃x• P   Q
∃intro   INHYP
|
P[α/x]
INHYP</td>
<td>∃x• (P ∨ Q) ⇒ ((∃x• P) ∨ Q)
DED
{∃x• (P ∨ Q)}
|
(∃x• P) ∨ Q
∃elim
{δ(α)}
/   \
∃x• (P ∨ Q)   (P∨Q)[α/x] ⇒((∃x• P) ∨ Q)
DED
{(P∨Q)[α/x]} i.e. {P[α/x] ∨Q}
|
(∃x• P) ∨ Q
∨elim
/ | \
P[α/x]⇒((∃x•P)∨Q)   Q⇒((∃x•P)∨Q)   P[α/x]∨Q
DED   DED   INHYP
{P[α/x]}   {Q}
| |
(∃x• P) ∨ Q   (∃x• P) ∨ Q
∨intro   ∨intro
| |
∃x• P   Q
∃intro   INHYP
{δ(α)}
/   \
P[α/x].   α≠ null
INHYP   NULL
|
δ(α)
INHYP</td>
</tr>
</table>

12. Assuming x is not free in Q: (∀x • (Q ⇒ P)) ⇔ (Q ⇒ (∀x•P))

FOPL	**FOPLN**
(∀ x • Q⇒P) ⇒ (Q⇒(∀ x • P)) DED {∀ x • Q⇒P} | Q⇒(∀ x • P) DED {Q} | ∀ x • P ∀-intro | P[α/x] MP / \\ Q Q⇒P[α/x] INHYP i.e. (Q⇒P)[α/x] ∀-elim | ∀x• Q⇒P INHYP	(∀ x • Q⇒P) ⇒ (Q⇒(∀ x • P)) DED {∀ x • Q⇒P} | Q⇒(∀ x • P) DED {Q} | ∀ x • P ∀-intro {δ(α)} | P[α/x] MP / \\ Q Q⇒P[α/x] INHYP i.e. (Q⇒P)[α/x] ∀-elim / \\ ∀x• Q⇒P δ(α) INHYP INHYP
(Q⇒(∀ x • P)) ⇒ (∀ x • Q⇒P) DED {Q⇒(∀ x • P)} | ∀ x • Q⇒P ∀-intro | (Q⇒P)[α/x] i.e. Q ⇒ P[α/x] DED {Q} | P[α/x] ∀-elim | ∀x.P MP / \\ Q Q⇒(∀ x • P) INHYP INHYP	(Q⇒(∀ x • P)) ⇒ (∀ x • Q⇒P) DED {Q⇒(∀ x • P)} | ∀ x • Q⇒P ∀-intro {δ(α)} | (Q⇒P)[α/x] i.e. Q ⇒ P[α/x] DED {Q} | P[α/x] ∀-elim / \\ ∀x.P δ(α) MP INHYP / \\ Q Q⇒(∀ x • P) INHYP INHYP

13. Assuming x is not free in Q: (∃x•(P⇒Q)) ⇔ ((∀x•P)⇒Q)

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>(∃x•(P⇒Q)) ⇒ ((∀x•P)⇒Q)
DED
{∃x•(P⇒Q)}
|
(∀x•P)⇒Q
DED
{∀x•P}
|
Q
∃-elim
/ \
∃x•(P⇒Q) (P⇒Q)[α/x] ⇒ Q
INHYP DED
{(P⇒Q)[α/x]}
i.e. {P[α/x]⇒Q}
|
Q
MP
/ \
P[α/x] P[α/x]⇒Q
∀-elim INHYP
|
∀x•P
INHYP</td>
<td>(∃x•(P⇒Q)) ⇒ ((∀x•P)⇒Q)
DED
{∃x•(P⇒Q)}
|
(∀x•P)⇒Q
DED
{∀x•P}
|
Q
∃-elim
{δ(α)}
/ \
∃x•(P⇒Q) (P⇒Q)[α/x] ⇒ Q
INHYP DED
{(P⇒Q)[α/x]} i.e. {P[α/x]⇒Q}
|
Q
MP
/ \
P[α/x] P[α/x]⇒Q
∀-elim INHYP
/ \
∀x•P δ(α)
INHYP INHYP</td>
</tr>
<tr>
<td>((∀x•P)⇒Q) ⇒ (∃x•(P⇒Q))
DED
{(∀x•P)⇒Q }
|
∃x•(P⇒Q)
∃-intro
|
(P⇒Q) [α/x] i.e. P[α/x] ⇒Q
DED
{ P[α/x] }
|
Q
MP
/ \
(∀x•P)⇒Q ∀x•P
INHYP ∀-intro
|
P[α/x]
INHYP</td>
<td>((∀x•P)⇒Q) ⇒ (∃x•(P⇒Q))
DED
{(∀x•P)⇒Q }
|
∃x•(P⇒Q)
Mu
{δ(α)}
|
∃x•(P⇒Q)
∃-intro
/ \
(P⇒Q) [α/x] i.e. P[α/x] ⇒Q α≠ null
DED NULL
{ P[α/x] } |
| δ(α)}
Q INHYP
MP
/ \
(∀x•P)⇒Q ∀x•P
INHYP ∀-intro
{δ(α)}
|
P[α/x]
INHYP</td>
</tr>
</table>

14. Assuming x is not free in Q: (∀x• (P ⇒ Q)) ⇔ ((∃x•P) ⇒ Q)

FOPL	**FOPLN**
(∀x• (P ⇒ Q)) ⇒ ((∃x•P) ⇒ Q) DED {∀x• (P ⇒ Q)} | (∃x•P) ⇒ Q DED {∃x• P} | Q ∃-elim / \ ∃x•P INHYP P[α/x] ⇒ Q i.e. (P⇒ Q) [α/x] ∀-elim | ∀x• (P ⇒ Q) INHYP	(∀x• (P ⇒ Q)) ⇒ ((∃x•P) ⇒ Q) DED {∀x• (P ⇒ Q)} | (∃x•P) ⇒ Q DED {∃x• P} | Q ∃-elim {δ(α)} / \ ∃x•P INHYP P[α/x] ⇒ Q i.e. (P⇒ Q) [α/x] ∀-elim / \ δ(α) INHYP ∀x• (P ⇒ Q) INHYP
((∃x•P) ⇒ Q) ⇒ (∀x• (P ⇒ Q)) DED {(∃x•P) ⇒ Q } | ∀x• (P ⇒ Q) ∀-intro | (P ⇒ Q) [α/x] i.e. P[α/x] ⇒ Q DED {P[α/x]} | Q MP / \ ∃x•P ∃-intro | P[α/x] INHYP (∃x•P) ⇒ Q INHYP	((∃x•P) ⇒ Q) ⇒ (∀x• (P ⇒ Q)) DED {(∃x•P) ⇒ Q } | ∀x• (P ⇒ Q) ∀-intro {δ(α)} | (P ⇒ Q) [α/x] i.e. P[α/x] ⇒ Q DED {P[α/x]} | Q MP / \ ∃x•P ∃-intro / \ P[α/x] INHYP α≠ null NULL | δ(α) INHYP (∃x•P) ⇒ Q INHYP

15. Assuming x is not free in Q: ∃x• (Q ⇒ P) ⇔ (Q ⇒ (∃x•P))

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>∃x• (Q ⇒ P) ⇒ (Q ⇒ (∃x•P))
DED
{∃x• (Q ⇒ P)}
|
Q ⇒ (∃x•P)
DED
{Q}
|
∃x•P
∃-intro
|
P[α/x]
∃-elim
/ \
{∃x• (Q ⇒ P)} (Q⇒P) P[α/x]⇒P[α/x]
INHYP DED
{ Q⇒P) P[α/x]}
i.e. { Q⇒P[α/x] }
MP
/ \
Q Q⇒P[α/x]
INHYP INHYP</td>
<td>∃x• (Q ⇒ P) ⇒ (Q ⇒ (∃x•P))
DED
{∃x• (Q ⇒ P)}
|
Q ⇒ (∃x•P)
DED
{Q}
|
∃x•P
∃-elim
{δ(α)}
/ \
{∃x• (Q ⇒ P)} (Q⇒P)[α/x]⇒∃x•P
DED
{(Q⇒P)[α/x]}
i.e. { Q⇒P[α/x] }
|
∃x•P
MP
/ \
Q Q⇒∃x•P
INHYP DED
{Q}
|
∃x•P
∃-intro
/ \
P[α/x] α≠ null
MP NULL
/ \ |
Q Q⇒P[α/x] δ(α)
INHYP INHYP INHYP</td>
</tr>
<tr>
<td>(Q ⇒ (∃x•P)) ⇒ (∃x• (Q ⇒ P))
DED
{Q ⇒ (∃x•P)}
|
∃x• (Q ⇒ P)
∃-intro
(Q ⇒ P) [α/x]
i.e. Q ⇒ P[α/x]
DED
{Q}
|
P[α/x]
∃-elim
/ \
P[α/x] ⇒ P[α/x] ∃x•P
DED MP
{P[α/x]} / \
| Q Q ⇒ (∃x•P)
P[α/x] INHYP INHYP
INHYP</td>
<td>(Q ⇒ (∃x•P)) ⇒ (∃x• (Q ⇒ P))
DED
{Q ⇒ (∃x•P)}
|
∃x• (Q ⇒ P)
Mu
{δ(α)}
|
∃x• (Q ⇒ P)
∃-intro
/ \
(Q ⇒ P) [α/x] α≠ null
i.e. Q ⇒ P[α/x] NULL
DED |
{Q} δ(α)
| INHYP
P[α/x]
∃-elim
/ \
P[α/x] ⇒ P[α/x] ∃x•P
DED MP
{P[α/x]} / \
| Q Q ⇒ ∃x•P
P[α/x] INHYP. INHYP
INHYP</td>
</tr>
</table>

16. Change of bound variable name I: $(\forall x \bullet P) \Leftrightarrow (\forall y \bullet P[y/x])$

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr><td>$(\forall x \bullet P) \Rightarrow (\forall y \bullet P[y/x])$
DED
$\{ \forall x \bullet P \}$
|
$\forall y \bullet P[y/x]$
∀-intro
|
$P[y/x][\alpha/y]$
i.e. $P[\alpha/x]$
∀-elim
|
$\forall x \bullet P$
INHYP</td><td>$(\forall x \bullet P) \Rightarrow (\forall y \bullet P[y/x])$
DED
$\{ \forall x \bullet P \}$
|
$\forall y \bullet P[y/x]$
∀-intro
$\{\delta(\alpha)\}$
|
$P[y/x][\alpha/y]$
i.e. $P[\alpha/x]$
∀-elim
/ \
$\forall x \bullet P$ $\delta(\alpha)$
INHYP INHYP</td></tr>
<tr><td>$(\forall y \bullet P[y/x]) \Rightarrow (\forall x \bullet P)$
DED
$\{ \forall y \bullet P[y/x] \}$
|
$\forall x \bullet P$
∀-intro
|
$P[\alpha/x]$
i.e. $P[y/x][\alpha/y]$
∀-elim
|
$\forall y \bullet P[y/x]$
INHYP</td><td>$(\forall y \bullet P[y/x]) \Rightarrow (\forall x \bullet P)$
DED
$\{ \forall y \bullet P[y/x] \}$
|
$\forall x \bullet P$
∀-intro
$\{\delta(\alpha)\}$
|
$P[\alpha/x]$
i.e. $P[y/x][\alpha/y]$
∀-elim
/ \
$\forall y \bullet P[y/x]$ $\delta(\alpha)$
INHYP INHYP</td></tr>
</table>

17. Change of bound variable name II: (∃ x • P) ⇔ (∃ y • P[y/x])

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>(∃ x • P) ⇒ (∃ y • P[y/x])
DED
{ ∃ x • P }
|
∃ y • P[y/x]
∃-elim
/ \
∃ x • P   P[α/x] ⇒ (∃ y • P[y/x])
INHYP   DED
{ P[α/x] }
|
∃ y • P[y/x]
∃-intro
|
P[y/x][α/y]
i.e. P[α/x]
INHYP</td>
<td>(∃ x • P) ⇒ (∃ y • P[y/x])
DED
{ ∃ x • P }
|
∃ y • P[y/x]
∃-elim
{δ(α)}
/ \
∃ x • P   P[α/x] ⇒ (∃ y • P[y/x])
INHYP   DED
{ P[α/x] }
|
∃ y • P[y/x]
∃-intro
{δ(α)}
/ \
P[y/x][α/y]   α≠ null
i.e. P[α/x]   NULL
INHYP |
δ(α)
INHYP</td>
</tr>
<tr>
<td>(∃ y • P[y/x]) ⇒ (∃ x • P)
DED
{ ∃ y • P[y/x] }
|
∃ x • P
∃-intro
|
P[α/x]
∃-elim
/ \
∃ y • P[y/x]   P[y/x][α/y] ⇒ P[α/x]
INHYP   DED
{ P[y/x][α/y] }
i.e. { P[α/x] }
|
P[α/x]
INHYP</td>
<td>(∃ y • P[y/x]) ⇒ (∃ x • P)
DED
{ ∃ y • P[y/x] }
|
∃ x • P
Mu
{δ(α)}
|
∃ x • P
∃-intro
/ \
P[α/x]   α≠ null
∃-elim   NULL
/ \ |
∃ y • P[y/x]   P[y/x][α/y] ⇒ P[α/x]   δ(α)
INHYP   DED   INHYP
{ P[y/x][α/y] }
i.e. { P[α/x] }
|
P[α/x]
INHYP</td>
</tr>
</table>

18. Monotonicity I : (∀ x • P ⇒ Q) ⇒ ((∀ x • P) ⇒ (∀ x • Q))

FOPL	**FOPLN**
(∀ x • P ⇒ Q) ⇒ ((∀ x • P) ⇒ (∀ x • Q)) DED { ∀ x • P ⇒ Q } | (∀ x • P) ⇒ (∀ x • Q) DED { ∀ x • P } | ∀ x • Q ∀-intro | Q[α/x] MP / \ P[α/x] P[α/x] ⇒ Q[α/x] ∀-elim i.e. (P ⇒ Q) [α/x] | ∀-elim ∀ x • P | INHYP ∀ x • P ⇒ Q INHYP	(∀ x • P ⇒ Q) ⇒ ((∀ x • P) ⇒ (∀ x • Q)) DED { ∀ x • P ⇒ Q } | (∀ x • P) ⇒ (∀ x • Q) DED { ∀ x • P } | ∀ x • Q ∀-intro { δ(α) } | Q[α/x] MP / \ P[α/x] P[α/x] ⇒ Q[α/x] ∀-elim i.e. (P ⇒ Q) [α/x] / \ ∀-elim ∀ x • P δ(α) / \ INHYP INHYP ∀ x • P ⇒ Q δ(α) INHYP INHYP

19. Monotonicity II : $(\forall x \bullet P \Rightarrow Q) \Rightarrow ((\exists x \bullet P) \Rightarrow (\exists x \bullet Q))$

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>
$(\forall x \bullet P \Rightarrow Q) \Rightarrow ((\exists x \bullet P) \Rightarrow (\exists x \bullet Q))$
DED
$\{ \forall x \bullet P \Rightarrow Q \}$
|
$(\exists x \bullet P) \Rightarrow (\exists x \bullet Q)$
DED
$\{ \exists x \bullet P \}$
|
$\exists x \bullet Q$
∃-intro
|
$Q[\alpha/x]$
∃-elim
/ \
$\exists x \bullet P$ $P[\alpha/x] \Rightarrow Q[\alpha/x]$
INHYP   i.e. $(P \Rightarrow Q) [\alpha/x]$
∀-elim
|
$\forall x \bullet P \Rightarrow Q$
INHYP
</td>
<td>
$(\forall x \bullet P \Rightarrow Q) \Rightarrow ((\exists x \bullet P) \Rightarrow (\exists x \bullet Q))$
DED
$\{ \forall x \bullet P \Rightarrow Q \}$
|
$(\exists x \bullet P) \Rightarrow (\exists x \bullet Q)$
DED
$\{ \exists x \bullet P \}$
|
$\exists x \bullet Q$
Mu
$\{ \delta(\alpha) \}$
|
$\exists x \bullet Q$
∃-intro
/ \
$Q[\alpha/x]$ $\alpha \neq null$
∃-elim   NULL
/ \ |
$\exists x \bullet P$ $P[\alpha/x] \Rightarrow Q[\alpha/x]$ $\delta(\alpha)$
INHYP   i.e. $(P \Rightarrow Q) [\alpha/x]$ INHYP
∀-elim
/ \
$\forall x \bullet P \Rightarrow Q$ $\delta(\alpha)$
INHYP   INHYP
</td>
</tr>
</table>

20. Equivalence I: (∀ x • P ⇔ Q) ⇒ ((∀ x• P) ⇔ (∀ x • Q))

FOPL	**FOPLN**
(∀ x • P ⇔ Q) ⇒ ((∀ x• P) ⇔ (∀ x • Q)) DED { ∀ x • P ⇔ Q } | (∀ x• P) ⇔ (∀ x • Q) REP / \\ P[α/x] ⇔ Q[α/x] ((∀x•P)⇔(∀x•Q)) i.e. (P ⇔ Q)[α/x] ∀-elim ⇔-intro | / \\ ∀ x•P⇔Q (∀x•P)⇒(∀x•P) (∀x•P)⇒(∀x•P) INHYP DED DED {∀x•P} {∀x•P} | | ∀x•P ∀x•P INHYP INHYP	(∀ x • P ⇔ Q) ⇒ ((∀ x• P) ⇔ (∀ x • Q)) DED { ∀ x • P ⇔ Q } | (∀ x• P) ⇔ (∀ x • Q) REP / \\ P[α/x] ⇔ Q[α/x] ((∀x•P)⇔(∀x•Q) i.e. (P ⇔ Q)[α/x] ⇔-intro ∀-elim / \\ {δ(α)} (∀x•P)⇒(∀x•P) (∀x•P)⇒(∀x•P) / \\ DED DED ∀ x•P⇔Q δ(α) {∀x•P} {∀x•P} INHYP INHYP | | ∀x•P ∀x•P INHYP INHYP

21. Equivalence II: (∀ x • P ⇔ Q) ⇒ ((∃ x• P) ⇔ (∃ x • Q))

FOPL	**FOPLN**
(∀ x • P ⇔ Q) ⇒ ((∃ x• P) ⇔ (∃ x • Q)) DED { ∀ x • P ⇔ Q } | (∃ x• P) ⇔ (∃ x • Q) i.e. ((∃ x• P) ⇔ (∃ x • Q)) [P/Q] REP / \\ P[α/x] ⇔ Q[α/x] ∃x•P⇔∃•P i.e. (P ⇔ Q)[α/x] ∀-elim ⇔-intro | / \\ ∀ x•P⇔Q (∃x•P)⇒(∃x•P) (∃x•P)⇒ (∃x•P) INHYP DED DED {∃x•P} {∃x•P} | | ∃x•P ∃x•P INHYP INHYP	(∀ x • P ⇔ Q) ⇒ ((∃ x• P) ⇔ (∃ x • Q)) DED { ∀ x • P ⇔ Q } | (∃ x• P) ⇔ (∃ x • Q) REP / \\ P[α/x] ⇔ Q[α/x] ∃x•P⇔∃•P i.e. (P ⇔ Q)[α/x] ⇔-intro ∀-elim / \\ {δ(α)} (∀x•P)⇒(∀x•P) (∀x•P)⇒(∀x•P) / \\ DED DED ∀ x•P⇔Q δ(α) {∀x•P} {∀x•P} INHYP INHYP | | ∀x•P ∀x•P INHYP INHYP

22. Equivalence III: (∃ x • P ⇒ Q) ⇔ ((∀ x • P) ⇒ (∃ x • Q))

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>(x • P⇒Q) ⇒ ((∀ x • P) ⇒ (∃ x • Q))
DED
{∃ x • P⇒Q }
|
(∀ x• P) ⇒ (∃ x • Q)
DED
{∀ x• P}
|
∃ x • Q
∃-elim
/ \
∃x•P⇒Q (P⇒Q)[α/x]⇒(∃ x • Q)
INHYP DED
{(P⇒Q)[α/x]} i.e. {P[α/x]⇒Q[α/x]}
|
∃ x • Q
∃-intro
|
Q[α/x]
MP
/ \
P[α/x] P[α/x]⇒Q[α/x]
∀-elim INHYP
|
∀ x• P
INHYP</td>
<td>(x • P⇒Q) ⇒ ((∀ x • P) ⇒ (∃ x • Q))
DED
{∃ x • P⇒Q }
|
(∀ x• P) ⇒ (∃ x • Q)
DED
{∀ x• P}
|
∃ x • Q
∃-elim
/ \
∃x•P⇒Q (P⇒Q)[α/x]⇒(∃ x • Q)
INHYP DED
{(P⇒Q)[α/x]} i.e. {P[α/x]⇒Q[α/x]}
|
∃ x • Q
∃-intro
/ \
Q[α/x] α≠ null
MP NULL
/ \ |
P[α/x] P[α/x]⇒Q[α/x] δ(α)
∀-elim INHYP INHYP
|
∀ x• P
INHYP</td>
</tr>
<tr>
<td>((∀ x • P) ⇒ (∃ x • Q)) ⇒ (∃ x • P ⇒ Q)
DED
{ (∀ x • P) ⇒ (∃ x • Q) }
|
∃ x • P ⇒ Q
∃-intro
|
(P⇒Q)[α/x]}
i.e. P[α/x]⇒Q[α/x]
DED
{P[α/x]}
|
Q[α/x]
∃-elim
/ \
Q[α/x]⇒Q[α/x] ∃ x • Q
DED MP
{Q[α/x] } / \
| ∀ x • P (∀ x • P) ⇒ (∃ x • Q)
Q[α/x] ∀-intro INHYP
INHYP |
P[α/x]
INHYP</td>
<td>((∀ x • P) ⇒ (∃ x • Q)) ⇒ (∃ x • P ⇒ Q)
DED
{ (∀ x • P) ⇒ (∃ x • Q) }
|
∃ x • P ⇒ Q
Mu
{δ(α)}
|
∃ x • P ⇒ Q
∃-intro
/ \
(P⇒Q)[α/x]} α≠ null
i.e. P[α/x]⇒Q[α/x] NULL
DED |
{P[α/x]} δ(α)
| INHYP
Q[α/x]
∃-elim
/ \
Q[α/x]⇒Q[α/x] ∃ x • Q
DED MP
{Q[α/x] } / \
| ∀ x • P (∀x•P) ⇒ (∃ x • Q)
Q[α/x] ∀-intro INHYP
INHYP |
P[α/x]
INHYP</td>
</tr>
</table>

23. One point rule I, given premise δ(E): (∀ x • x=E ⇒ P) ⇔ P[E/x]

<table>
<tr><th>FOPL</th><th>FOPLN</th></tr>
<tr>
<td>(∀ x • x=E ⇒ P) ⇒ P[E/x]
DED
{ ∀ x • x=E ⇒ P }
|
P[E/x]
MP
/ \
E=E (E=E) ⇒ P[E/x]
EQUALS i.e. (x=E ⇒ P)[E/x]
∀-elim
|
∀ x • x=E ⇒ P
INHYP</td>
<td>{δ(E)}
((∀ x • x=E ⇒ P) ⇒ P[E/x])
DED
{ ∀ x • x=E ⇒ P }
|
P[E/x]
MP
/ \
E=E (E=E) ⇒ P[E/x]
EQUALS i.e. (x=E ⇒ P)[E/x]
∀-elim
/ \
∀ x • x=E ⇒ P δ(E)
INHYP INHYP</td>
</tr>
<tr>
<td>P[E/x] ⇒ (∀ x • x=E ⇒ P)
DED
{ P[E/x }
|
∀ x • x=E ⇒ P
∀-intro
|
(x=E ⇒ P)[E/x]
i.e. (x=E)[E/x] ⇒ P[E/x]
i.e. (E=E) ⇒ P[E/x]
DED
{E=E}
|
P[E/x]
INHYP</td>
<td>Same to the left.</td>
</tr>
</table>

24. One point rule II, given premise δ(E): (∃ x • x=E ∧ P) ⇔ P[E/x]

FOPL	**FOPLN**
(∃ x • x=E ∧ P) ⇒ P[E/x] DED { ∃ x • x=E ∧ P } | P[E/x] ∃-elim / \\ ∃ x • x=E ∧ P (x=E ∧ P)[α/x] ⇒ P[E/x] INHYP DED { (x=E ∧ P)[α/x] } i.e. {α=E ∧ P[α/x]} | P[E/x] LEIBNIZ / \\ α=E P[α/x] ∧-elim ∧-elim | | α=E ∧ P[α/x] α=E ∧ P[α/x] INHYP INHYP	Same to the left
P[E/x] ⇒ (∃ x • x=E ∧ P) DED { P[E/x] } | ∃ x • x=E ∧ P ∃-intro | (x=E ∧ P) [E/x] i.e. E=E ∧ P[E/x] ∧-intro / \\ E=E P[E/x] EQUALS INHYP	{δ(E)} P[E/x] ⇒ (∃ x • x=E ∧ P) DED { P[E/x] } | ∃ x • x=E ∧ P ∃-intro {δ(E)} / \\ (x=E ∧ P) [E/x] E≠ null i.e. E=E ∧ P[E/x] NULL ∧-intro | / \\ δ(E) E=E P[E/x] INHYP EQUALS INHYP

References

[SDM22] B. Stoddart, S. Dunne, and C. Mu. Much ado about nothing. *Formal Aspects of Computing Science (FACS) FACTS Newsletter*, July 2022.