Hilber-type calculus

Definition 1.3.7. Hilbert-type calculus for propositional logic (denoted *HPC*) consists of axioms:

1.1. $F \supset (G \supset F)$;

1.2. $\big(F \supset (G \supset H)\big) \supset \big((F \supset G) \supset (F \supset H)\big)$;

2.1. $(F \wedge G) \supset F$;

2.2. $(F \wedge G) \supset G$;

2.3. $(F \supset G) \supset \Big((F \supset H) \supset \big(F \supset (G \wedge H)\big)\Big)$;

3.1. $F \supset (F \vee G)$;

3.2. $G \supset (F \vee G)$;

3.3. $(F \supset H) \supset \Big((G \supset H) \supset \big((F \vee G) \supset H\big)\Big)$;

4.1. $(F \supset G) \supset (\neg G \supset \neg F)$;

4.2. $F \supset \neg\neg F$;

4.3. $\neg\neg F \supset F$;

Definition 2.4.3. Hilbert-type calculus for predicate logic (denoted *HPR*) consists of axioms of *HPC* and:

H

5.1 $\forall x F(x) \supset F(t)$,

5.2 $F(t) \supset \exists x F(x)$.

Here, t is a term, which is free with respect to variable x in formula $F(x)$.

The rules of *HPR* are:

$$\frac{F \qquad F \supset G}{G}\,MP \qquad\qquad \frac{G \supset F(y)}{G \supset \forall x F(x)}\,\forall \qquad\qquad \frac{F(y) \supset G}{\exists x F(x) \supset G}\,\exists$$

Example 1.3.9. A derivation of $p \supset p$ in *HPC* is as follows:

1.	$\big(p \supset ((p \supset p) \supset p)\big) \supset \big((p \supset (p \supset p)) \supset (p \supset p)\big)$	Axiom *1.2*, $\{p/F, p\supset p/G, p/H\}$.
2.	$p \supset \big((p \supset p) \supset p\big)$	Axiom *1.1*, $\{p/F, p\supset p/G\}$.
3.	$\big(p \supset (p \supset p)\big) \supset (p \supset p)$	*MP* rule from 2 and 1.
4.	$p \supset (p \supset p)$	Axiom *1.1*, $\{p/F, p/G\}$.
5.	$p \supset p$	*MP* rule from 4 and 3.

Example 2.4.4. Let's show, that from formula $\forall x \forall y P(x,y)$ another formula $\forall y \forall x P(x,y)$ is derivable. Let's construct a derivation:

1.	$\forall x \forall y P(x,y)$	An assumption.
2.	$\forall x \forall y P(x,y) \supset \forall y P(a,y)$	Axiom *5.1*.
3.	$\forall y P(a,y)$	*MP* rule from 1 and 2.
4.	$\forall y P(a,y) \supset P(a,b)$	Axiom *5.1*.
5.	$P(a,b)$	*MP* rule from 3 and 4.
6.	$\forall x P(x,b)$	$\forall$ rule from 5.
7.	$\forall y \forall x P(x,y)$	$\forall$ rule from 6.

Sequential calculus

Definition 1.3.13. The original Gentzen-type calculus for propositional logic (GPC_o) consists of an axiom $F \to F$, structural rules:

Weakening:

$$\frac{\Gamma \to \Delta}{F, \Gamma \to \Delta}\,(w \to) \qquad \frac{\Gamma \to \Delta}{\Gamma \to \Delta, F}\,(\to w)$$

Contraction:

$$\frac{F, F, \Gamma \to \Delta}{F, \Gamma \to \Delta}\,(c \to) \qquad \frac{\Gamma \to \Delta, F, F}{\Gamma \to \Delta, F}\,(\to c)$$

Exchange:

$$\frac{\Gamma_1, G, F, \Gamma_2 \to \Delta}{\Gamma_1, F, G, \Gamma_2 \to \Delta}\,(e \to) \qquad \frac{\Gamma \to \Delta_1, G, F, \Delta_2}{\Gamma \to \Delta_1, F, G, \Delta_2}\,(\to e)$$

logical rules:

Negation:

$$\frac{\Gamma \to \Delta, F}{\neg F, \Gamma \to \Delta}\,(\neg \to) \qquad \frac{F, \Gamma \to \Delta}{\Gamma \to \Delta, \neg F}\,(\to \neg)$$

Conjunction:

$$\frac{F, \Gamma \to \Delta}{F \wedge G, \Gamma \to \Delta}\,(\wedge \to)_1 \qquad \frac{G, \Gamma \to \Delta}{F \wedge G, \Gamma \to \Delta}\,(\wedge \to)_2$$

$$\frac{\Gamma \to \Delta, F \qquad \Gamma \to \Delta, G}{\Gamma \to \Delta, F \wedge G}\,(\to \wedge)$$

Disjunction:

$$\frac{F, \Gamma \to \Delta \qquad G, \Gamma \to \Delta}{F \vee G, \Gamma \to \Delta}\,(\vee \to)$$

$$\frac{\Gamma \to \Delta, F}{\Gamma \to \Delta, F \vee G}\,(\to \vee)_1 \qquad \frac{\Gamma \to \Delta, G}{\Gamma \to \Delta, F \vee G}\,(\to \vee)_2$$

Implication:

$$\frac{\Gamma \to \Delta, F \qquad G, \Gamma \to \Delta}{F \supset G, \Gamma \to \Delta}\,(\supset \to) \qquad \frac{F, \Gamma \to \Delta, G}{\Gamma \to \Delta, F \supset G}\,(\to \supset)$$

and the cut rule:

$$\frac{\Gamma_1 \to \Delta_1, F \qquad F, \Gamma_2 \to \Delta_2}{\Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2}\,(\text{cut } F)$$

$$\frac{F(z), \Gamma \to \Delta}{\exists x F(x), \Gamma \to \Delta}\,(\exists \to) \qquad \frac{\Gamma \to \Delta, \exists x F(x), F(t)}{\Gamma \to \Delta, \exists x F(x)}\,(\to \exists)$$

$$\frac{F(t), \forall x F(x), \Gamma \to \Delta}{\forall x F(x), \Gamma \to \Delta}\,(\forall \to) \qquad \frac{\Gamma \to \Delta, F(z)}{\Gamma \to \Delta, \forall x F(x)}\,(\to \forall)$$

Here z is a new variable, which doesn't belong to Γ, Δ, $\exists x F(x)$ nor $\forall x F(x)$ and t is a term, which is free with respect to x in formula $F(x)$.

Example 1.3.14. A derivation in *GPC* of the formula used in Example 1.3.9 is obvious, so a derivation tree of axiom *2.3* of *HPC* is provided instead.

$$
\begin{prooftree}
\AxiomC{$F \to F$}
\RightLabel{$(\to w)$}
\UnaryInfC{$F \to F, G \wedge H$}
\RightLabel{$(\to e)$}
\UnaryInfC{$F \to G \wedge H, F$}
\RightLabel{$(w \to)$}
\UnaryInfC{$F \supset G, F \to G \wedge H, F$}
\RightLabel{$(e \to)$}
\UnaryInfC{$F, F \supset G \to G \wedge H, F$}
\AxiomC{$F \to F$}
\RightLabel{$(\to w)$}
\UnaryInfC{$F \to F, G \wedge H$}
\RightLabel{$(\to e)$}
\UnaryInfC{$F \to G \wedge H, F$}
\RightLabel{$(w \to)$}
\UnaryInfC{$H, F \to G \wedge H, F$}
\AxiomC{$G \to G$}
\RightLabel{$(w \to)$}
\UnaryInfC{$F, G \to G$}
\RightLabel{$(e \to)$}
\UnaryInfC{$G, F \to G$}
\RightLabel{$(w \to)$}
\UnaryInfC{$H, G, F \to G$}
\RightLabel{$(e \to)$}
\UnaryInfC{$G, H, F \to G$}
\AxiomC{$H \to H$}
\RightLabel{$(w \to)$}
\UnaryInfC{$F, H \to H$}
\RightLabel{$(e \to)$}
\UnaryInfC{$H, F \to H$}
\RightLabel{$(w \to)$}
\UnaryInfC{$G, H, F \to H$}
\RightLabel{$(\to \wedge)$}
\BinaryInfC{$G, H, F \to G \wedge H$}
\RightLabel{$(\supset \to)$}
\BinaryInfC{$F \supset G, H, F \to G \wedge H$}
\RightLabel{$(e \to)$}
\UnaryInfC{$H, F \supset G, F \to G \wedge H$}
\RightLabel{$(e \to)$}
\UnaryInfC{$H, F, F \supset G \to G \wedge H$}
\RightLabel{$(\supset \to)$}
\BinaryInfC{$F \supset H, F, F \supset G \to G \wedge H$}
\RightLabel{$(e \to)$}
\UnaryInfC{$F, F \supset H, F \supset G \to G \wedge H$}
\RightLabel{$(\to \supset)$}
\UnaryInfC{$F \supset H, F \supset G \to F \supset (G \wedge H)$}
\RightLabel{$(\to \supset)$}
\UnaryInfC{$F \supset G \to (F \supset H) \supset \big(F \supset (G \wedge H)\big)$}
\RightLabel{$(\to \supset)$}
\UnaryInfC{$\to (F \supset G) \supset \Big((F \supset H) \supset \big(F \supset (G \wedge H)\big)\Big)$}
\end{prooftree}
$$

However such calculus has some drawbacks too. First of all, it is not obvious how to chose the main formula in the cut rule.

Example 1.3.15. The choice of main formula in the cut rule is not obvious:

$$
\begin{prooftree}
\AxiomC{$p \to p$}
\RightLabel{$(\to w)$}
\UnaryInfC{$p \to p, q$}
\RightLabel{$(\to e)$}
\UnaryInfC{$p \to q, p$}
\AxiomC{$q \to q$}
\RightLabel{$(w \to)$}
\UnaryInfC{$p, q \to q$}
\RightLabel{$(e \to)$}
\UnaryInfC{$q, p \to q$}
\RightLabel{$(\supset \to)$}
\BinaryInfC{$p \supset q, p \to q$}
\RightLabel{$(e \to)$}
\UnaryInfC{$p, p \supset q \to q$}
\AxiomC{$q \to q$}
\RightLabel{$(\to w)$}
\UnaryInfC{$q \to q, r$}
\RightLabel{$(\to e)$}
\UnaryInfC{$q \to r, q$}
\AxiomC{$r \to r$}
\RightLabel{$(w \to)$}
\UnaryInfC{$q, r \to r$}
\RightLabel{$(e \to)$}
\UnaryInfC{$r, q \to r$}
\RightLabel{$(\supset \to)$}
\BinaryInfC{$q \supset r, q \to r$}
\RightLabel{$(e \to)$}
\UnaryInfC{$q, q \supset r \to r$}
\RightLabel{(cut q)}
\BinaryInfC{$p, p \supset q, q \supset r \to r$}
\end{prooftree}
$$

Next, there are a lot of applications of different structural rules:

Example 1.3.16. Consider the following derivation:

$$
\begin{prooftree}
\AxiomC{$p \to p$}
\RightLabel{$(w \to)$}
\UnaryInfC{$q, p \to p$}
\AxiomC{$q \to q$}
\RightLabel{$(w \to)$}
\UnaryInfC{$p, q \to q$}
\RightLabel{$(e \to)$}
\UnaryInfC{$q, p \to q$}
\RightLabel{$(\to \wedge)$}
\BinaryInfC{$q, p \to p \wedge q$}
\RightLabel{$(\to \vee)_1$}
\UnaryInfC{$q, p \to (p \wedge q) \vee (p \wedge r)$}
\AxiomC{$p \to p$}
\RightLabel{$(w \to)$}
\UnaryInfC{$r, p \to p$}
\AxiomC{$r \to r$}
\RightLabel{$(w \to)$}
\UnaryInfC{$p, r \to r$}
\RightLabel{$(e \to)$}
\UnaryInfC{$r, p \to r$}
\RightLabel{$(\to \wedge)$}
\BinaryInfC{$r, p \to p \wedge r$}
\RightLabel{$(\to \vee)_2$}
\UnaryInfC{$r, p \to (p \wedge q) \vee (p \wedge r)$}
\RightLabel{$(\vee \to)$}
\BinaryInfC{$q \vee r, p \to (p \wedge q) \vee (p \wedge r)$}
\RightLabel{$(\wedge \to)_2$}
\UnaryInfC{$p \wedge (q \vee r), p \to (p \wedge q) \vee (p \wedge r)$}
\RightLabel{$(e \to)$}
\UnaryInfC{$p, p \wedge (q \vee r) \to (p \wedge q) \vee (p \wedge r)$}
\RightLabel{$(\wedge \to)_1$}
\UnaryInfC{$p \wedge (q \vee r), p \wedge (q \vee r) \to (p \wedge q) \vee (p \wedge r)$}
\RightLabel{$(e \to)$}
\UnaryInfC{$p \wedge (q \vee r) \to (p \wedge q) \vee (p \wedge r)$}
\end{prooftree}
$$

Gentzen-type calculus

Aksiomos: $F \vdash F$.

Struktūrinės taisyklės:

(silpninimas) $$\frac{\Gamma \vdash \Delta}{F, \Gamma \vdash \Delta}, \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, F},$$

(prastinimas) $$\frac{F, F, \Gamma \vdash \Delta}{F, \Gamma \vdash \Delta}, \qquad \frac{\Gamma \vdash \Delta, F, F}{\Gamma \vdash \Delta, F},$$

(perstatymas) $$\frac{\Gamma_1, F, G, \Gamma_2 \vdash \Delta}{\Gamma_1, G, F, \Gamma_2 \vdash \Delta}, \qquad \frac{\Gamma \vdash \Delta_1, F, G, \Delta_2}{\Gamma \vdash \Delta_1, G, F, \Delta_2}.$$

Loginių operacijų taisyklės:

$$(\neg \vdash) \quad \frac{\Gamma \vdash \Delta, F}{\neg F, \Gamma \vdash \Delta}, \qquad (\vdash \neg) \quad \frac{F, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg F},$$

$$(\& \vdash) \quad \frac{F, G, \Gamma \vdash \Delta}{F\&G, \Gamma \vdash \Delta}, \qquad (\vdash \&) \quad \frac{\Gamma \vdash \Delta, F \qquad \Gamma \vdash \Delta, G}{\Gamma \vdash \Delta, F\&G},$$

$$(\vee \vdash) \quad \frac{F, \Gamma \vdash \Delta \qquad G, \Gamma \vdash \Delta}{F \vee G, \Gamma \vdash \Delta}, \qquad (\vdash \vee) \quad \frac{\Gamma \vdash \Delta, F, G}{\Gamma \vdash \Delta, F \vee G},$$

$$(\to\vdash) \quad \frac{\Gamma \vdash \Delta, F \qquad G, \Gamma \vdash \Delta}{F \to G, \Gamma \vdash \Delta}, \qquad (\vdash\to) \quad \frac{F, \Gamma \vdash \Delta, G}{\Gamma \vdash \Delta, F \to G}.$$

Pjūvio taisyklė:

$$\frac{\Gamma_1 \vdash \Delta_1, F \qquad F, \Gamma_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2}.$$

Kvantorinės taisyklės:

$$(\exists \vdash) \quad \frac{F(z), \Gamma \vdash \Delta}{\exists x F(x), \Gamma \vdash \Delta}, \qquad (\vdash \exists) \quad \frac{\Gamma \vdash \Delta, F(t), \exists x F(x)}{\Gamma \vdash \Delta \exists x F(x)},$$

$$(\forall \vdash) \quad \frac{F(t), \forall x F(x), \Gamma \vdash \Delta}{\forall x F(x), \Gamma \vdash \Delta}, \qquad (\vdash \forall) \quad \frac{\Gamma \vdash \Delta, F(z)}{\Gamma \vdash \Delta \forall x F(x)}.$$

Čia z yra naujas kintamasis, neįeinantis į $\Gamma, \Delta, \exists x F(x)$ arba $\forall x F(x)$, t — termas, laisvas kintamojo x atžvilgiu formulėje $F(x)$.

Primename, kad sekvencijoje $\Gamma \vdash \Delta$ seka Γ vadinama ***antecedentu***, o Δ — ***sukcedentu***. Sakoma: formulė F priklauso sekvencijos antecedentui, jei ji yra sekoje Γ; formulė priklauso sukcedentui, jei ji yra sekoje Δ.

Gentzen-type calculus

Pavyzdžiai:

1. Parodykime, kad sekvencija $F\&G, \neg H \vdash (\neg F \vee \neg G) \to H$ išvedama nagrinėjamajame skaičiavime:

$$
\begin{array}{c}
\begin{array}{c}
F \vdash F \\ \hline
F \vdash F, H \\ \hline
F \vdash H, F \\ \hline
\neg H, F \vdash H, F \\ \hline
F, \neg H \vdash H, F \\ \hline
G, F, \neg H \vdash H, F \\ \hline
F, G, \neg H \vdash H, F \\ \hline
\neg F, F, G, \neg H \vdash H
\end{array}
\qquad
\begin{array}{c}
G \vdash G \\ \hline
G \vdash G, H \\ \hline
G \vdash H, G \\ \hline
\neg H, G \vdash H, G \\ \hline
G, \neg H \vdash H, G \\ \hline
F, G, \neg H \vdash H, G \\ \hline
\neg G, F, G, \neg H \vdash H
\end{array}
\\ \hline
\neg F \vee \neg G, F, G, \neg H \vdash H \\ \hline
F, G, \neg H \vdash (\neg F \vee \neg G) \to H \\ \hline
F\&G, \neg H \vdash (\neg F \vee \neg G) \to H
\end{array}
$$

2. Sekvencija $\vdash \exists x \forall y A(x, y) \to \forall y \exists x A(x, y)$ taip pat išvedama:

$$
\begin{array}{c}
A(a, b) \vdash A(a, b) \\ \hline
A(a, b) \vdash A(a, b), \exists x A(x, b) \\ \hline
\forall y A(a, y), A(a, b) \vdash A(a, b), \exists x A(x, b) \\ \hline
A(a, b), \forall y A(a, y) \vdash A(a, b), \exists x A(x, b) \\ \hline
A(a, b), \forall y A(a, y) \vdash \exists x A(x, b) \\ \hline
\forall y A(a, y) \vdash \exists x A(x, b) \\ \hline
\forall y A(a, y) \vdash \forall y \exists x A(x, y) \\ \hline
\exists x \forall y A(x, y) \vdash \forall y \exists x A(x, y) \\ \hline
\vdash \exists x \forall y A(x, y) \to \forall y \exists x A(x, y)
\end{array}
$$

Example 2.5.4. Let's show that sequent $\to \forall x \forall y P(x, y) \supset \forall y \forall x P(x, y)$ is derivable in calculus GPR. Recall, that this sequent is similar to the case analysed in Example 2.4.4.

$$
\begin{array}{cl}
P(x_1, y_1), \forall y P(x_1, y), \forall x \forall y P(x, y) \to P(x_1, y_1) & \\ \hline
\forall y P(x_1, y), \forall x \forall y P(x, y) \to P(x_1, y_1) & (\forall \to) \\ \hline
\forall x \forall y P(x, y) \to P(x_1, y_1) & (\forall \to) \\ \hline
\forall x \forall y P(x, y) \to \forall x P(x, y_1) & (\to \forall) \\ \hline
\forall x \forall y P(x, y) \to \forall y \forall x P(x, y) & (\to \forall) \\ \hline
\to \forall x \forall y P(x, y) \supset \forall y \forall x P(x, y) & (\to \supset)
\end{array}
$$

Example 2.5.5. Let's derive sequent: $\to \exists x \forall y P(x, y) \supset \forall y \exists x P(x, y)$ in calculus GPR.

$$
\begin{array}{cl}
P(x_1, y_1), \forall y P(x_1, y) \to \exists x P(x, y_1), P(x_1, y_1) & \\ \hline
P(x_1, y_1), \forall y P(x_1, y) \to \exists x P(x, y_1) & (\to \exists) \\ \hline
\forall y P(x_1, y) \to \exists x P(x, y_1) & (\forall \to) \\ \hline
\forall y P(x_1, y) \to \forall y \exists x P(x, y) & (\to \forall) \\ \hline
\exists x \forall y P(x, y) \to \forall y \exists x P(x, y) & (\exists \to) \\ \hline
\to \exists x \forall y P(x, y) \supset \forall y \exists x P(x, y) & (\to \supset)
\end{array}
$$

Example 2.5.6. Now a derivation of a well known tautology of predicate logic is presented: $\to \big(\neg \forall x P(x) \supset \exists x \neg P(x)\big) \wedge \big(\exists x \neg P(x) \supset \neg \forall x P(x)\big)$.

$$
\begin{array}{cl}
\begin{array}{cl}
P(z) \to \exists x \neg P(x), P(z) & \\ \hline
\to \exists x \neg P(x), \neg P(z), P(z) & (\to \neg) \\ \hline
\to \exists x \neg P(x), P(z) & (\to \exists) \\ \hline
\to \exists x \neg P(x), \forall x P(x) & (\to \forall) \\ \hline
\neg \forall x P(x) \to \exists x \neg P(x) & (\neg \to) \\ \hline
\to \neg \forall x P(x) \supset \exists x \neg P(x) & (\to \supset)
\end{array}
\qquad
\begin{array}{cl}
P(z), \forall x P(x) \to P(z) & \\ \hline
\forall x P(x) \to P(z) & (\forall \to) \\ \hline
\forall x P(x), \neg P(z) \to & (\neg \to) \\ \hline
\forall x P(x), \exists x \neg P(x) \to & (\exists \to) \\ \hline
\exists x \neg P(x) \to \neg \forall x P(x) & (\to \neg) \\ \hline
\to \exists x \neg P(x) \supset \neg \forall x P(x) & (\to \supset)
\end{array}
& \\ \hline
\to \big(\neg \forall x P(x) \supset \exists x \neg P(x)\big) \wedge \big(\exists x \neg P(x) \supset \neg \forall x P(x)\big) & (\to \wedge)
\end{array}
$$

Gentzen-type calculus
For intuitionistic logic

Definition 2.10.6. Gentzen-type calculus for intuitionistic logic (denoted *GIN*) is composed of axiom $F \to F$, structural rules:

Weakening:

$$\frac{\Gamma \to \Delta}{F, \Gamma \to \Delta}\,(w \to) \qquad \frac{\Gamma \to}{\Gamma \to F}\,(\to w)$$

Contraction:

$$\frac{F, F, \Gamma \to \Delta}{F, \Gamma \to \Delta}\,(c \to)$$

Exchange:

$$\frac{\Gamma_1, G, F, \Gamma_2 \to \Delta}{\Gamma_1, F, G, \Gamma_2 \to \Delta}\,(e \to)$$

Logical rules:

Negation:

$$\frac{\Gamma \to F}{\neg F, \Gamma \to \Delta}\,(\neg \to) \qquad \frac{F, \Gamma \to}{\Gamma \to \neg F}\,(\to \neg)$$

Conjunction:

$$\frac{F, G, \Gamma \to \Delta}{F \wedge G, \Gamma \to \Delta}\,(\wedge \to) \qquad \frac{\Gamma \to F \qquad \Gamma \to G}{\Gamma \to F \wedge G}\,(\to \wedge)$$

Disjunction:

$$\frac{F, \Gamma \to \Delta \qquad G, \Gamma \to \Delta}{F \vee G, \Gamma \to \Delta}\,(\vee \to)$$

$$\frac{\Gamma \to F}{\Gamma \to F \vee G}\,(\to \vee)_1 \qquad \frac{\Gamma \to G}{\Gamma \to F \vee G}\,(\to \vee)_2$$

Implication:

$$\frac{\Gamma \to F \qquad G, \Gamma \to \Delta}{F \supset G, \Gamma \to \Delta}\,(\supset \to) \qquad \frac{F, \Gamma \to G}{\Gamma \to F \supset G}\,(\to \supset)$$

Quantifier rules:

Existence:

$$\frac{F(z), \Gamma \to \Delta}{\exists x F(x), \Gamma \to \Delta}\,(\exists \to) \qquad \frac{\Gamma \to F(t)}{\Gamma \to \Delta, \exists x F(x)}\,(\to \exists)$$

Universality:

$$\frac{F(t), \forall x F(x), \Gamma \to \Delta}{\forall x F(x), \Gamma \to \Delta}\,(\forall \to) \qquad \frac{\Gamma \to F(z)}{\Gamma \to \forall x F(x)}\,(\to \forall)$$

Variable z and term t must satisfy the same requirements as in the case of GPR_a.

And the cut rule:

$$\frac{\Gamma_1 \to F \qquad F, \Gamma_2 \to \Delta}{\Gamma_1, \Gamma_2 \to \Delta}\,(\text{cut } F)$$

Note that set Δ in this calculus represents one formula or an empty set. Succedent is not necessary, however it must contain no more than one formula.

Equivalency

Lemma 1.3.29. *These formulas are equivalent in propositional logic:*

- $\neg\neg p \equiv p$,
- $\neg(p \wedge q) \equiv \neg p \vee \neg q$,
- $\neg(p \vee q) \equiv \neg p \wedge \neg q$,
- $p \supset q \equiv \neg p \vee q$,
- $p \leftrightarrow q \equiv (p \supset q) \wedge (q \supset p)$.

Lemma 2.1.19. *These formulas are equivalent in predicate logic:*

1. $\forall x \forall y F \equiv \forall y \forall x F$,
2. $\exists x \exists y F \equiv \exists y \exists x F$,
3. $\forall x F(x) \equiv \forall y F(y)$, *here y is not part of $F(x)$ and x is not part of $F(y)$,*
4. $\exists x F(x) \equiv \exists y F(y)$, *here y is not part of $F(x)$ and x is not part of $F(y)$,*
5. $\neg \forall x F \equiv \exists x \neg F$,
6. $\neg \exists x F \equiv \forall x \neg H$.

1. $\neg \forall x F(x) \equiv \exists x \neg F(x)$,
2. $\neg \exists x F(x) \equiv \forall x \neg F(x)$,
3. $\exists x F(x) \equiv \exists y F(y)$, where y is a new variable, not part of $F(x)$,
4. $\forall x F(x) \equiv \forall y F(y)$, where y is a new variable, not part of $F(x)$,
5. $\forall x F(x) \wedge G \equiv \forall x \big(F(x) \wedge G\big)$, where x is not part of G,
6. $\exists x F(x) \wedge G \equiv \exists x \big(F(x) \wedge G\big)$, where x is not part of G,
7. $\forall x F(x) \vee G \equiv \forall x \big(F(x) \vee G\big)$, where x is not part of G,
8. $\exists x F(x) \vee G \equiv \exists x \big(F(x) \vee G\big)$, where x is not part of G,
   1. $\forall x F(x) \wedge \forall x G(x) \equiv \forall x \big(F(x) \wedge G(x)\big)$,
   2. $\exists x F(x) \vee \exists x G(x) \equiv \exists x \big(F(x) \vee G(x)\big)$,

Function symbols / Compactness

Definition 2.3.1. A *term* is defined as follows:

- Individual constant is a term.
- Individual variable is a term.
- If f^n is function symbol and $t_1, t_2, \ldots, t_n$ are terms, then $f(t_1, t_2, \ldots, t_n)$ is also a term.

Example 2.3.2. Let $\mathcal{M} = \{a, b, c\}$, $f(x)$ is one-place function symbol. Then the following expressions are terms: a, b, c, x, y, z, $f(x)$, $f(a)$, $f(c)$, $f\big(f(x)\big)$, $f\big(f(a)\big)$, $f\Big(f\big(f(y)\big)\Big)$,....

Definition 2.3.3. *Formula* is defined recursively as follows:

1. If P is n-place predicate variable and $t_1, t_2, \ldots, t_n$ are terms, then $P(t_1, t_2, \ldots, t_n)$ is a formula.
2. If p is a propositional variable, then p is a formula.
3. If F is a formula, then $\neg F$ is also a formula.
4. If F and G are formulas, then $(F \wedge G)$, $(F \vee G)$, $(F \supset G)$ and $(F \leftrightarrow G)$ are formulas.
5. if F is a formula and x is individual variable, then $\exists x F$ and $\forall x F$ are also formulas.

2.7 Compactness

In this section only propositional formulas are analysed. However, the results presented here are needed for the next section.

Definition 2.7.1. A set of formulas is *finite satisfiable*, if every finite subset of the set is satisfiable. A set of formulas $\{F_1, F_2, \ldots\}$ is *satisfiable* if there exists an interpretation ν such that $\nu \models F_i$ for every $i \in [1, \infty)$.

If a set of formulas is not satisfiable, then it is called *contradictory*. I.e. a set of propositional formulas $\mathcal{F}$ is contradictory if for any interpretation ν there is a formula $F \in \mathcal{F}$ such that $\nu \not\models F$.

Definition 2.7.2. A set of formulas $\mathcal{F}$ is *maximal*, if:

- $\mathcal{F}$ is finite satisfiable and
- for any formula F either $F \in \mathcal{F}$ or $\neg F \in \mathcal{F}$.

2.6 Semantic tableaux method

Suppose, sequent $F_1, F_2, \ldots, F_n \to G_1, G_2, \ldots, G_m$ is derivable in sequent calculus for predicate logic GPR and the derivation is $\mathcal{D}$. Then sequent $F_1, F_2, \ldots, F_n, \neg G_1, \neg G_2, \ldots, \neg G_m \to$ is also derivable and the derivation is:

$$\cfrac{\cfrac{\cfrac{\mathcal{D}}{F_1, F_2, \ldots, F_n \to G_1, G_2, \ldots, G_m}}{F_1, F_2, \ldots, F_n, \neg G_1 \to G_2, G_3, \ldots, G_m}\,(\neg\to)}{F_1, F_2, \ldots, F_n, \neg G_1, \neg G_2 \to G_3, G_4, \ldots, G_m}\,(\neg\to)$$

$$\ldots$$

$$F_1, F_2, \ldots, F_n, \neg G_1, \neg G_2, \ldots, \neg G_m \to$$

$$\frac{F, \Gamma \to}{\neg\neg F, \Gamma \to} \qquad \frac{\neg F, \Gamma \to \qquad \neg G, \Gamma \to}{\neg(F \wedge G), \Gamma \to} \qquad \frac{\neg F, \neg G, \Gamma \to}{\neg(F \vee G), \Gamma \to}$$

$$\frac{F, \neg G, \Gamma \to}{\neg(F \supset G), \Gamma \to} \qquad \frac{\neg F(t), \neg\exists x F(x), \Gamma \to}{\neg\exists x F(x), \Gamma \to} \qquad \frac{\neg F(z), \Gamma \to}{\neg\forall x F(x), \Gamma \to}$$

The change to rule $(\supset\to)$ is obvious:

$$\frac{\neg F, \Gamma \to \qquad G, \Gamma \to}{F \supset G, \Gamma \to}\,(\supset\to)$$

Conjunction rules:

$$\frac{F \wedge G}{\begin{array}{c}F\\G\end{array}} \qquad \frac{\neg(F \vee G)}{\begin{array}{c}\neg F\\\neg G\end{array}} \qquad \frac{\neg(F \supset G)}{\begin{array}{c}F\\\neg G\end{array}}$$

Disjunction rules:

$$\frac{\neg(F \wedge G)}{\neg F \mid \neg G} \qquad \frac{F \vee G}{F \mid G} \qquad \frac{F \supset G}{\neg F \mid G}$$

Double negation rule:

$$\frac{\neg\neg F}{F}$$

Quantifier rules:

$$\frac{\forall x F(x)}{F(t)} \qquad \frac{\neg\forall x F(x)}{\neg F(z)} \qquad \frac{\exists x F(x)}{F(z)} \qquad \frac{\neg\exists x F(x)}{\neg F(t)}$$

Here z is a new free variable, which is not part of any formula of the branch, and t is some main term.

2.8 Semantic trees

Now let's return to predicate logic. In this section predicate formulas with function symbols but without free variables are analysed. Moreover, formulas must be transformed into normal prenex form and skolemized.

Definition 2.8.1. A *Herbrand universe* $\mathcal{H}$ of formula F is defined as follows:

- Every constant of formula F is part of $\mathcal{H}$. If F does not contain any constants, then $a \in \mathcal{H}$.
- If f^n is n-place function symbol in formula F and $t_1, t_2, \dots, t_n \in \mathcal{H}$, then $f(t_1, t_2, \dots, t_n) \in \mathcal{H}$

It must be noted that according to the definition Herbrand universe of any formula is never empty. It is finite, if the formula does not contain function symbols, or infinite but countable.

Definition 2.8.2. A *Herbrand base* $\mathcal{B}$ of formula F is the set of all the atomic formulas $P(t_1, t_2, \dots, t_n)$, where P^n is n-place predicate variable of F and $t_i, i \in [1, n]$ are some elements of Herbrand universe of F.

Definition 2.8.3. *H-interpretation* (or *Herbrand interpretation*) of formula F is a set $\{\alpha_1 P_1, \alpha_2 P_2, \dots\}$, where $\alpha_i \in \{\neg, \varnothing\}$ and $P_i \in \mathcal{B}, i \in [1, \infty)$. If $\alpha_i = \neg$, then it is understood that P_i is false with the H-interpretation, otherwise P_i is true.

Example 2.8.5. Let's start by finding a finite semantic tree of predicate formula $\forall x\Big(P\big(x, f(a)\big) \wedge \neg P\big(x, x\big)\Big)$.

$$\mathcal{H} = \{a, f(a), f\big(f(a)\big), \dots\}, \mathcal{B} = \{P\big(a, a\big), P\big(a, f(a)\big), P\big(f(a), f(a)\big), \dots\}$$

$$\dfrac{\dfrac{\oplus}{P(a,a)} \qquad \dfrac{\dfrac{\dfrac{\oplus}{P\big(f(a),f(a)\big)} \quad \dfrac{\oplus}{\neg P\big(f(a),f(a)\big)}}{P\big(a,f(a)\big)} \qquad \dfrac{\oplus}{\neg P\big(a,f(a)\big)}}{\neg P(a,a)}}{\forall x\Big(P\big(x,f(a)\big) \wedge \neg P\big(x,x\big)\Big)}$$

In this case set $\mathcal{T}$ is equal to:

$$\{F_1 = P\big(a, f(a)\big) \wedge \neg P\big(a, a\big),$$

$$F_2 = P\big(f(a), f(a)\big) \wedge \neg P\big(f(a), f(a)\big),$$

$$F_3 = P\Big(f\big(f(a)\big), f(a)\Big) \wedge \neg P\Big(f\big(f(a)\big), f\big(f(a)\big)\Big), \dots\}$$

Let's analyse branches from left to right. With interpretation $\{P(a, a)\}$ formula F_1 is false. With interpretations $\{\neg P\big(a, a\big), P\big(a, f(a)\big), P\big(f(a), f(a)\big)\}$ and $\{\neg P\big(a, a\big), P\big(a, f(a)\big), \neg P\big(f(a), f(a)\big)\}$ formula F_2 is false. With interpretation $\{\neg P\big(a, a\big), \neg P\big(a, f(a)\big)\}$ once again formula F_1 is false. Thus the contradictory subset consists of two formulas: F_1 and F_2.

2.9 Resolution method

2.9.1 The description of resolution calculus

Once again, in this section only skolemized normal prenex form predicate formulas are analysed. In order to apply resolution method a set of disjuncts must be obtained. As mentioned in the previous section, all the analysed formulas are of the form $\forall x_1 \forall x_2 \dots \forall x_n G(x_1, x_2, \dots, x_n)$. This formula is not satisfiable iff a set $\mathcal{T} = \{G(t_1, t_2 \dots, t_n) : t_i \in \mathcal{H}, i \in [1, n]\}$, where $\mathcal{H}$ is Herbrand universe of the formula, is contradictory. From the fact that $\mathcal{H}$ is either finite or countable and there are only finite number (n) of individual variables in the formula it follows that $\mathcal{T}$ is either finite or countable.

Now let's transform formula $G(x_1, x_2, \dots, x_n)$ into NCF. Suppose that NCF of $G(x_1, x_2, \dots, x_n)$ is $\bigwedge\limits_{j=1}^{m} H_j(x_1, x_2, \dots, x_n)$, where $H_j, j \in [1, m]$ are disjuncts. Let's analyse a set of disjuncts

Definition 2.9.1. A substitution α is called a *unifier* of formulas (terms) F_1 and F_2, if $F_1\alpha = F_2\alpha$.

Definition 2.9.2. A unifier α is the *most general* one for formulas (terms) F_1 and F_2, if for any unifier β of F_1 and F_2 there is a substitution γ, such that $\beta = \alpha \circ \gamma$ (i.e. β is a composition of α and γ).

Substitution α is a *unifier* of set $\{F_1, F_2, \dots, F_n\}$, if $F_1\alpha = F_2\alpha = \dots = F_n\alpha$. Substitution α is the *most general unifier* of the set, if it is a unifier of the set and for any unifier of the set β there is a substitution γ such that $\beta = \alpha \circ \gamma$.

Example 2.9.4. Atomic formulas $P\Big(x, f\big(f(y)\big)\Big)$ and $P\Big(f(z), f\Big(f\big(g(a)\big)\Big)\Big)$ are unifiable. The unifiers are, for example, $\big\{f(a)/x, g(a)/y, a/z\big\}$ and $\Big\{f\big(f(a)\big)/x, g(a)/y, f(a)/z\Big\}$. The most general unifier is $\big\{f(z)/x, g(a)/y\big\}$.

Of course, not all the expressions are unifiable. E.g. formulas $F(t_1)$ and $F(t_2)$ are not unifiable, if terms t_1 and t_2 are:

- two different constants,
- a variable x and term t which contains x, $t \neq x$,
- a constant and function symbol,
- terms starting with different function symbols.

Now a rule of resolution for predicate formulas is:

$$\frac{F \vee l_1 \qquad l_2 \vee G}{(F \vee G)\alpha}\,\alpha$$

Here l_1 and l_2 are literals, one of which is $P(t_1^1, t_2^1, \dots, t_n^1)$ and another one is $\neg P(t_1^2, t_2^2, \dots, t_n^2)$. Substitution α is a unifier of $P(t_1^1, t_2^1, \dots, t_n^1)$ and $P(t_1^2, t_2^2, \dots, t_n^2)$.

In order to derive formula in resolution calculus the following steps must be taken:

1. it must be transformed into normal prenex form;
2. it must be skolemized, $\forall$ quantifiers must be removed;
3. NCF of skolemized formula must be found;
4. a set composed of the disjuncts of the NCF must be created;
5. the set of disjuncts must be derived in resolution calculus using resolution rule for predicate logic.

Example 2.9.5. Using resolution method let's prove that if some binary relation is irreflexive and transitive, then it is asymmetric.

Let's denote a binary relation between x and y as predicate $P(x,y)$. Irreflexivity is defined as $F_1 = \forall x \neg P(x,x)$ and transitivity is defined as $F_2 = \forall x \forall y \forall z \big(\big(P(x,y) \wedge P(y,z)\big) \supset P(x,z)\big)$. Asymmetry is defined as $F_3 = \forall x \forall y \big(P(x,y) \supset \neg P(y,x)\big)$.

The aim is to prove that from F_1 and F_2 follows F_3. This can be done by showing that set $\{F_1, F_2, \neg F_3\}$ is contradictory.

Let's transform the set into set of disjuncts. After removing $\forall$ quantifier from F_1, a disjunct $\neg P(x,x)$ is obtained. Formula F_2 is also skolemized, thus let's remove all the $\forall$ quantifiers and transform it into NCF:

$$\big(P(x,y) \wedge P(y,z)\big) \supset P(x,z) = \neg\big(P(x,y) \wedge P(y,z)\big) \vee P(x,z) =$$

$$= \neg P(x,y) \vee \neg P(y,z) \vee P(x,z)$$

Thus formula F_2 results in one disjunct: $\neg P(x,y) \vee \neg P(y,z) \vee P(x,z)$. Finally with F_3 every step of the algorithm must be performed. First it must be transformed into normal prenex form:

$$\neg\forall x \forall y \big(P(x,y) \supset \neg P(y,x)\big) = \exists x \exists y \neg\big(P(x,y) \supset \neg P(y,x)\big) =$$

$$= \exists x \exists y \neg\big(\neg P(x,y) \vee \neg P(y,x)\big) = \exists x \exists y \big(P(x,y) \wedge P(y,x)\big)$$

After skolemization formula $P(a,b) \wedge P(b,a)$ is obtained and this results in two disjuncts $P(a,b)$ and $P(b,a)$.

Therefore, $\mathcal{S} = \{\neg P(x,x), \neg P(x,y) \vee \neg P(y,z) \vee P(x,z), P(a,b), P(b,a)\}$. Now the following derivation can be obtained:

$$\dfrac{\neg P(x,x) \qquad \dfrac{P(b,a) \qquad \dfrac{P(a,b) \qquad \neg P(x,y) \vee \neg P(y,z) \vee P(x,z)}{\neg P(b,z) \vee P(a,z)}\{^a/_x, {}^b/_y\}}{P(a,a)}\{^a/_z\}}{\square}\{^a/_x\}$$

2.9.2 Linear tactics

Suppose formula F is derivable from set $\mathcal{S}$ and $T_1, T_2, \ldots T_n$ is a sequence of applications of resolution rule in the derivation. If a conclusion of T_n is F and for every $i \in [2,n]$ one of the premisses of T_i is a conclusion of T_{i-1}, then it is said that the derivation follows the linear tactics.

Example 2.9.6. Let's transform the following derivation into the one that follows the linear tactics:

$$\dfrac{\dfrac{\mathcal{D}}{F_3 \vee p} \qquad \dfrac{F_1 \vee \neg p \vee q \qquad F_2 \vee \neg q}{F_1 \vee F_2 \vee \neg p}}{F_1 \vee F_2 \vee F_3}$$

Here $\mathcal{D}$ denotes a derivation of $F_3 \vee p$, which follows linear tactics. The solution is as follows:

$$\dfrac{\dfrac{\dfrac{\mathcal{D}}{F_3 \vee p} \qquad F_1 \vee \neg p \vee q}{F_1 \vee F_3 \vee q} \qquad F_2 \vee \neg q}{F_1 \vee F_2 \vee F_3}$$

Example 2.9.7. Let's derive the set used in Example 1.3.25 in resolution calculus by following the linear tactics:

$$\dfrac{F \qquad \dfrac{\neg F \vee H \qquad \dfrac{\neg F \vee G \qquad \neg G \vee \neg H}{\neg F \vee \neg H}}{\neg F}}{\square}$$

2.9.3 Tactics of semantic resolution

Suppose, $\mathcal{S}$ is a set of disjuncts and $\mathcal{I}$ is some interpretation, which divides $\mathcal{S}$ into two subsets $\mathcal{S}_+$ and $\mathcal{S}_-$. Here $\mathcal{S}_+$ consists of formulas from $\mathcal{S}$ that are true with interpretation $\mathcal{I}$ and $\mathcal{S}_-$ consists of formulas from $\mathcal{S}$ that are false with interpretation $\mathcal{I}$.

According to the tactics, premisses of the application of resolution rule must be part of different subsets. After the application, the conclusion is also assigned to $\mathcal{S}_+$ or $\mathcal{S}_-$ depending on if it is true with $\mathcal{I}$ or not.

Example 2.9.8. Let $\mathcal{S} = \{p \vee q \vee \neg r, \neg p \vee q, \neg q \vee \neg r, r\}$ and interpretation $\mathcal{I}$ defined as follows: $\mathcal{I}(p) = \top$, $\mathcal{I}(q) = \top$ and $\mathcal{I}(r) = \bot$.

In this case $\mathcal{S}_+ = \{p \vee q \vee \neg r, \neg p \vee q, \neg q \vee \neg r\}$ and $\mathcal{S}_- = \{r\}$.

The derivation is as follows:

$$\dfrac{\dfrac{\dfrac{p \vee q \vee \neg r \quad r}{p \vee q} \quad \dfrac{\neg q \vee \neg r \quad r}{\neg q}}{p} \qquad \dfrac{\neg p \vee q \quad \dfrac{\neg q \vee \neg r \quad r}{\neg q}}{\neg p}}{\square}$$

It should be noted that the conclusions are included into subsets as soon as they are obtained. At the end $\mathcal{S}_+ = \{p \vee q \vee \neg r, \neg p \vee q, \neg q \vee \neg r, p \vee q, p\}$ and $\mathcal{S}_- = \{r, \neg q, \neg p\}$.

Example 2.9.9. Let's find the derivation of

$$\mathcal{S} = \{p_1 \vee p_2, p_1 \vee \neg p_2, \neg p_1 \vee p_2, \neg p_1 \vee \neg p_2\}$$

using semantic resolution tactics, when the interpretation $\mathcal{I}$ is $\mathcal{I}(p_1) = \top$, $\mathcal{I}(p_2) = \bot$.

Based on the given information $\mathcal{S}_+ = \{p_1 \vee p_2, p_1 \vee \neg p_2, \neg p_1 \vee \neg p_2\}$ and $\mathcal{S}_- = \{\neg p_1 \vee p_2\}$. Thus the derivation would be as follows:

$$\dfrac{\dfrac{p_1 \vee \neg p_2 \quad \dfrac{p_1 \vee p_2 \quad \neg p_1 \vee p_2}{p_2}}{p_1} \qquad \dfrac{\neg p_1 \vee \neg p_2 \quad \neg p_1 \vee p_2}{\neg p_1}}{\square}$$

The subsets are:

$$\mathcal{S}_+ = \{p_1 \vee p_2, p_1 \vee \neg p_2, \neg p_1 \vee \neg p_2, p_1\}, \mathcal{S}_- = \{\neg p_1 \vee p_2, p_2, \neg p_1\}$$

It is said that disjunct $F_1 = l \vee G$ is absorbed by another disjunct F_2, if $F_2 = \neg l \vee G \vee G'$. Here l is a literal, $\neg\neg l = l$, G and G' are disjuncts and G' may be empty.

According to absorption tactics, a resolution rule can be applied only if one of the disjuncts absorbs the other one. More precisely, the rule can be applied to F_1 and F_2 with unifier α is either $F_1\alpha$ absorbs or is absorbed by $F_2\alpha$.

Example 2.9.12. Suppose, it is possible to transform any derivation tree to the one which follows the absorption tactics, if there are no more than n applications of resolution rule (induction hypothesis). A derivation tree of $F \vee G$ with $n+1$ application of resolution rule is as follows:

$$\dfrac{\dfrac{\dfrac{\mathcal{D}_1}{F \vee p \vee q} \quad \dfrac{\mathcal{D}_2}{F \vee p \vee \neg q}}{F \vee p} \qquad \dfrac{\mathcal{D}_3}{G \vee \neg p}}{F \vee G}\,(*)$$

The $(*)$ denotes an application of resolution rule which doesn't follow the absorption tactics. Let's prove, that it is possible to derive disjunct $F \vee G$ in resolution method using absorption tactics.

Let's analyse the two derivations:

$$\dfrac{\dfrac{\mathcal{D}_3}{G \vee \neg p} \quad \dfrac{\mathcal{D}_1}{F \vee p \vee q}}{F \vee G \vee q} \quad \text{and} \quad \dfrac{\dfrac{\mathcal{D}_3}{G \vee \neg p} \quad \dfrac{\mathcal{D}_2}{F \vee p \vee \neg q}}{F \vee G \vee \neg q}$$

Both of them have no more than n applications of resolution rule. Therefore, according to the induction hypothesis, it is possible to derive both disjuncts $F \vee G \vee q$ and $F \vee G \vee \neg q$ using absorption tactics. Let's denote the resulting derivations $\mathcal{D}_1'$ and $\mathcal{D}_2'$. Then the derivation of $F \vee G$ that follows the absorption tactics is as follows:

$$\dfrac{\dfrac{\mathcal{D}_1'}{F \vee G \vee q} \quad \dfrac{\mathcal{D}_2'}{F \vee G \vee \neg q}}{F \vee G}$$

that $\mathcal{S} = \langle \mathcal{M}, \mathcal{R}, \mathcal{V} \rangle$ is a Kripke structure, then:

- relation $\mathcal{R}$ is reflexive, iff formula $\Box F \supset F$ is valid in structure $\mathcal{S}$ for any formula F;
- relation $\mathcal{R}$ is complete, iff formula $\Box F \supset \Diamond F$ is valid in structure $\mathcal{S}$ for any formula F;
- relation $\mathcal{R}$ is symmetric, iff formula $F \supset \Box\Diamond F$ is valid in structure $\mathcal{S}$ for any formula F;
- relation $\mathcal{R}$ is transitive, iff formula $\Box F \supset \Box\Box F$ is valid in structure $\mathcal{S}$ for any formula F;
- relation $\mathcal{R}$ is euclidean, iff formula $\Box F \supset \Box\Diamond F$ is valid in structure $\mathcal{S}$ for any formula F;

Theorem 4.1.9. *The following formulas are valid in any Kripke structure.*

1. $\Diamond F \supset \neg\Box\neg F$;
2. $\neg\Box\neg F \supset \Diamond F$;
3. $\Box F \supset \neg\Diamond\neg F$;
4. $\neg\Diamond\neg F \supset \Box F$;
5. $\Box(F \supset G) \supset (\Box F \supset \Box G)$.

Definition 4.1.5. Suppose $\mathcal{S} = \langle \mathcal{W}, \mathcal{R}, \nu \rangle$ is a Kripke structure of formula F and $w \in \mathcal{W}$, then:

- if F is a propositional variable, then $\mathcal{S}, w \vDash F$ iff $\nu(F, w) = \top$.
- if $F = \neg G$, then $\mathcal{S}, w \vDash F$ iff $\mathcal{S}, w \nvDash G$.
- if $F = G \wedge H$, then $\mathcal{S}, w \vDash F$ iff $\mathcal{S}, w \vDash G$ and $\mathcal{S}, w \vDash H$.
- if $F = G \vee H$, then $\mathcal{S}, w \nvDash F$ iff $\mathcal{S}, w \nvDash G$ and $\mathcal{S}, w \nvDash H$.
- if $F = G \supset H$, then $\mathcal{S}, w \nvDash F$ iff $\mathcal{S}, w \vDash G$ and $\mathcal{S}, w \nvDash H$.
- if $F = G \leftrightarrow H$, then $\mathcal{S}, w \vDash F$ iff $\mathcal{S}, w \vDash G$ and $\mathcal{S}, w \vDash H$ or $\mathcal{S}, w \nvDash G$ and $\mathcal{S}, w \nvDash H$.
- if $F = \Box G$, then $\mathcal{S}, w \vDash F$ iff for every world $w_1 \in \mathcal{W}$ such that $w\mathcal{R}w_1$ it is true that $\mathcal{S}, w_1 \vDash G$.
- if $F = \Diamond G$, then $\mathcal{S}, w \vDash F$ iff there is a world $w_1 \in \mathcal{W}$ such that $w\mathcal{R}w_1$ and $\mathcal{S}, w_1 \vDash G$.

Modal Logic Hilber Type

Definition 4.2.1. Hilbert-type calculus for modal logics consists of the same axioms and rules as *HPC*, Necessity Generalisation rule (NG):

$$\frac{F}{\Box F}$$

And axioms that depend on modal logic in question:

(K): $\Box(F \supset G) \supset (\Box F \supset \Box G)$;

(T): $\Box F \supset F$;

(D): $\Box F \supset \Diamond F$;

(B): $F \supset \Box\Diamond F$;

(4): $\Box F \supset \Box\Box F$;

(5): $\Box F \supset \Box\Diamond F$.

Different modal logics are obtained by choosing a different set of modal axioms. Propositional axioms, *MP* and *NG* rules are included in all the calculi.

The different modal logics are summarised in the following table:

Logics	Hilbert-type calculus	Modal axioms
K	*HK*	*(K)*
T	*HT*	*(K)*, *(T)*
S4	*HS4*	*(K)*, *(T)*, *(4)*
S5	*HS5*	*(K)*, *(T)*, *(4)*, *(5)*
K4	*HK4*	*(K)*, *(4)*
B	*HB*	*(K)*, *(B)*
D	*HD*	*(K)*, *(D)*
K45	*HK45*	*(K)*, *(4)*, *(5)*
KD45	*HKD45*	*(K)*, *(D)*, *(4)*, *(5)*

Definition 1.3.7. Hilbert-type calculus for propositional logic (denoted *HPC*) consists of axioms:

1.1. $F \supset (G \supset F)$;

1.2. $\big(F \supset (G \supset H)\big) \supset \big((F \supset G) \supset (F \supset H)\big)$;

2.1. $(F \wedge G) \supset F$;

2.2. $(F \wedge G) \supset G$;

2.3. $(F \supset G) \supset \Big((F \supset H) \supset \big(F \supset (G \wedge H)\big)\Big)$;

3.1. $F \supset (F \vee G)$;

3.2. $G \supset (F \vee G)$;

3.3. $(F \supset H) \supset \Big((G \supset H) \supset \big((F \vee G) \supset H\big)\Big)$;

4.1. $(F \supset G) \supset (\neg G \supset \neg F)$;

4.2. $F \supset \neg\neg F$;

4.3. $\neg\neg F \supset F$;

4.3 Sequent calculi

Definition 4.3.1. Gentzen-type calculus for modal logics is obtained from *GPC* by adding modal rules, which depend on logic:

Calculus *GK* for modal logic *K*:

$$\frac{\Gamma_2 \to F}{\Gamma_1, \Box\Gamma_2 \to \Delta, \Box F}\,(\to\Box)$$

Calculus *GK4* for modal logic *K4*:

$$\frac{\Gamma_2, \Box\Gamma_2 \to F}{\Gamma_1, \Box\Gamma_2 \to \Delta, \Box F}\,(\to\Box)$$

Calculus *GT* for modal logic *T*:

$$\frac{F, \Box F, \Gamma \to \Delta}{\Box F, \Gamma \to \Delta}\,(\Box\to) \qquad \frac{\Gamma_2 \to F}{\Gamma_1, \Box\Gamma_2 \to \Delta, \Box F}\,(\to\Box)$$

Calculus *GS4* for modal logic *S4*:

$$\frac{F, \Box F, \Gamma \to \Delta}{\Box F, \Gamma \to \Delta}\,(\Box\to) \qquad \frac{\Box\Gamma_2 \to F}{\Gamma_1, \Box\Gamma_2 \to \Delta, \Box F}\,(\to\Box)$$

Here $\Box\Gamma_2$ is a set of formulas, starting with $\Box$. In that case Γ_2 is a set of formulas obtained from $\Box\Gamma_2$ by removing outermost $\Box$ occurrence in each formula.

Modal Logic Example

Example 4.3.2. Let's derive sequent $\to \neg\Box\neg(p \vee \Box\neg p)$ in sequent calculus *GS4*.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{p, \Box\neg(p \vee \Box\neg p) \to p, \Box\neg p}{p, \Box\neg(p \vee \Box\neg p) \to p \vee \Box\neg p}\,(\to\vee)}{p, \neg(p \vee \Box\neg p), \Box\neg(p \vee \Box\neg p) \to}\,(\neg\to)}{p, \Box\neg(p \vee \Box\neg p) \to}\,(\Box\to)}{\Box\neg(p \vee \Box\neg p) \to \neg p}\,(\to\neg)}{\Box\neg(p \vee \Box\neg p) \to p, \Box\neg p}\,(\to\Box)}{\Box\neg(p \vee \Box\neg p) \to p \vee \Box\neg p}\,(\to\vee)}{\neg(p \vee \Box\neg p), \Box\neg(p \vee \Box\neg p) \to}\,(\neg\to)}{\Box\neg(p \vee \Box\neg p) \to}\,(\Box\to)}{\to \neg\Box\neg(p \vee \Box\neg p)}\,(\to\neg)$$

The main formula of the application of $(\Box\to)$ rule is repeated in the premiss too. This repetition cannot be omitted, because without it the sequent wouldn't be derivable:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{p \to}{\to \neg p}\,(\to\neg)}{\to p, \Box\neg p}\,(\to\Box)}{\to p \vee \Box\neg p}\,(\to\vee)}{\neg(p \vee \Box\neg p) \to}\,(\neg\to)}{\Box\neg(p \vee \Box\neg p) \to}}{\to \neg\Box\neg(p \vee \Box\neg p)}\,(\to\neg)$$

Example 4.3.3. Let's derive sequent $\Box(p \wedge q) \to \Box p \wedge \Box q$ in sequent calculus of logic *T*.

$$\dfrac{\dfrac{\dfrac{p, q \to p}{p \wedge q \to p}\,(\wedge\to)}{\Box(p \wedge q) \to \Box p}\,(\to\Box) \qquad \dfrac{\dfrac{p, q \to q}{p \wedge q \to q}\,(\wedge\to)}{\Box(p \wedge q) \to \Box q}\,(\to\Box)}{\Box(p \wedge q) \to \Box p \wedge \Box q}\,(\to\wedge)$$

Sometimes it is not enough to analyse only modal operator $\Box$. One such case is when it is required that negation be only in front of propositional variables. Every modal formula can be transformed into such form, however, operator $\Diamond$ is necessary. It is easy to extend a calculus to deal with modal operator $\Diamond$ (it must be kept in mind that $\Diamond F$ can be changed by $\neg\Box\neg F$). E.g. modal rules of sequent calculus for logic *S4* are as follows:

rules for operator $\Box$:

$$\dfrac{F, \Box F, \Gamma \to \Delta}{\Box F, \Gamma \to \Delta}\,(\Box\to) \qquad \dfrac{\Box\Gamma_2 \to \Diamond\Delta_2, F}{\Gamma_1, \Box\Gamma_2 \to \Delta_1, \Diamond\Delta_2, \Box F}\,(\to\Box)$$

rules for operator $\Diamond$:

$$\dfrac{\Box\Gamma_2, F \to \Diamond\Delta_2}{\Gamma_1, \Box\Gamma_2, \Diamond F \to \Delta_1, \Diamond\Delta_2}\,(\Diamond\to) \qquad \dfrac{\Gamma \to \Delta, F, \Diamond F}{\Gamma \to \Delta, \Diamond F}\,(\to\Diamond)$$

Here $\Diamond\Delta_2$ is a set of formulas, starting with $\Diamond$.

Example 4.2.2. Let's find a derivation of formula $p \supset \Diamond p$ in calculus *HT*. First of all, $\Diamond$ is not part of the calculus, therefore it must be replaced and a derivation of formula $p \supset \neg\Box\neg p$ must be found:

1. $\Box\neg p \supset \neg p$ — Axiom *(T)*, $\{\neg p/F\}$.
2. $(\Box\neg p \supset \neg p) \supset (\neg\neg p \supset \neg\Box\neg p)$ — Axiom *4.1*, $\{\Box\neg p/F, \neg p/G\}$.
3. $\neg\neg p \supset \neg\Box\neg p$ — *MP* rule from 1 and 2.
4. $p \supset \neg\neg p$ — Axiom *4.2*, $\{p/F\}$.
5. $(\neg\neg p \supset \neg\Box\neg p) \supset (p \supset (\neg\neg p \supset \neg\Box\neg p))$ — Axiom *1.1*, $\{\neg\neg p \supset \neg\Box\neg p/F, p/G\}$.
6. $p \supset (\neg\neg p \supset \neg\Box\neg p)$ — *MP* rule from 3 and 5.
7. $(p \supset (\neg\neg p \supset \neg\Box\neg p)) \supset ((p \supset \neg\neg p) \supset (p \supset \neg\Box\neg p))$ — Axiom *1.2*, $\{p/F, \neg\neg p/G, \neg\Box\neg p/H\}$.
8. $(p \supset \neg\neg p) \supset (p \supset \neg\Box\neg p)$ — *MP* rule from 6 and 7.
9. $p \supset \neg\Box\neg p$ — *MP* rule from 4 and 8.

Example 4.4.2. Let's derive formula $F = (\Box\Diamond p \wedge \Box\Diamond q) \supset \Box\Diamond(\Box\Diamond p \wedge \Box\Diamond q)$ in tableaux calculus *TS4*:

$$1\ \neg\big((\Box\Diamond p \wedge \Box\Diamond q) \supset \Box\Diamond(\Box\Diamond p \wedge \Box\Diamond q)\big) = F_1$$
$$1\ \Box\Diamond p \wedge \Box\Diamond q = F_2$$
$$1\ \neg\Box\Diamond(\Box\Diamond p \wedge \Box\Diamond q) = F_3$$
$$1\ \Box\Diamond p = F_4$$
$$1\ \Box\Diamond q = F_5$$
$$1.1\ \neg\Diamond(\Box\Diamond p \wedge \Box\Diamond q) = F_6$$
$$1.1\ \neg(\Box\Diamond p \wedge \Box\Diamond q) = F_7$$

$$\begin{array}{cc} 1.1\ \neg\Box\Diamond p = F_8 & 1.1\ \neg\Box\Diamond q = F_9 \\ 1.1\ \Box\Diamond p = F_{10} & 1.1\ \Box\Diamond q = F_{11} \end{array}$$

Formulas F_2 and F_3 are obtained by applying conjunction rule to formula F_1. In the same way formulas F_4 and F_5 are obtained from F_2. To obtain formula F_6, possibility rule is applied to F_3. Formula F_7 results from the application of rule *(T)* to F_6. Formulas F_8 and F_9 are the result of application of disjunction rule to formula F_7. Finally, F_{10} is obtained from F_4 and F_{11} is obtained from F_5 after application of *(4)* rule.

One branch of derivation tree includes prefixed formulas $1.1\ \neg\Box\Diamond p$ and $1.1\ \Box\Diamond p$, the other one — $1.1\ \neg\Box\Diamond q$ and $1.1\ \Box\Diamond q$. Thus the tree is closed and it is a derivation of formula F.

4.4 Tableaux calculus

Tableaux calculus for modal logics is similar to the one for predicate logic defined in Section 2.6. However instead of regular formulas, *prefixed formulas* are used. A prefixed formula is an expression of the form $\sigma\ F$, where σ is a finite sequence of natural numbers, called a *prefix*, and F is some formula. To derive formula F using tableaux calculus, the derivation search must start with prefixed formula $1\ \neg F$.

Definition 4.4.1. A tableaux calculus for modal logics consists of the following rules:

Conjunction rules:

$$\frac{\sigma\ F \wedge G}{\begin{array}{c}\sigma\ F\\ \sigma\ G\end{array}} \qquad \frac{\sigma\ \neg(F \vee G)}{\begin{array}{c}\sigma\ \neg F\\ \sigma\ \neg G\end{array}} \qquad \frac{\sigma\ \neg(F \supset G)}{\begin{array}{c}\sigma\ F\\ \sigma\ \neg G\end{array}}$$

Disjunction rules:

$$\frac{\sigma\ \neg(F \wedge G)}{\sigma\ \neg F \mid \sigma\ \neg G} \qquad \frac{\sigma\ F \vee G}{\sigma\ F \mid \sigma\ G} \qquad \frac{\sigma\ F \supset G}{\sigma\ \neg F \mid \sigma\ G}$$

Double negation rule:

$$\frac{\sigma\ \neg\neg F}{\sigma\ F}$$

Necessity rules:

$$\frac{\sigma\ \Box F}{\sigma.n\ F} \qquad \frac{\sigma\ \neg\Diamond F}{\sigma.n\ \neg F}$$

where $\sigma.n$ is any prefix, $n \in \mathbb{N}$.

Possibility rules:

$$\frac{\sigma\ \Diamond F}{\sigma.k\ F} \qquad \frac{\sigma\ \neg\Box F}{\sigma.k\ \neg F}$$

where $\sigma.k$ is a new prefix, that does not appear in the branch, $k \in \mathbb{N}$.

And additional rules for different modal logics:

Rules *(T)*:

$$\frac{\sigma\ \Box F}{\sigma\ F} \qquad \frac{\sigma\ \neg\Diamond F}{\sigma\ \neg F}$$

Rules *(D)*:

$$\frac{\sigma\ \Box F}{\sigma\ \Diamond F} \qquad \frac{\sigma\ \neg\Diamond F}{\sigma\ \neg\Box F}$$

Rules *(B)*:

$$\frac{\sigma.n\ \Box F}{\sigma\ F} \qquad \frac{\sigma.n\ \neg\Diamond F}{\sigma\ \neg F}$$

where $\sigma.n$ is any prefix, $n \in \mathbb{N}$.

Rules *(4)*:

$$\frac{\sigma\ \Box F}{\sigma.n\ \Box F} \qquad \frac{\sigma\ \neg\Diamond F}{\sigma.n\ \neg\Diamond F}$$

where $\sigma.n$ is any prefix, $n \in \mathbb{N}$.

4.5 Relation to predicate logic

For any modal formula F it is possible to find a predicate formula $\mathrm{Tr}(F)_x$ with one free variable x such that F is satisfiable in logic K iff $\mathrm{Tr}(F)_x$ is satisfiable in predicate logic. Let's demonstrate how formula $\mathrm{Tr}(F)_x$ can be obtained:

- $\mathrm{Tr}(G)_x = P(x)$, if G is a propositional variable. Here P is a predicate variable into which propositional variable G is transformed. The predicate variable is different for different propositional variables of modal formula, however it is the same for different occurrences of the same propositional variable.
- $\mathrm{Tr}(\neg G)_x = \neg\,\mathrm{Tr}(G)_x$;
- $\mathrm{Tr}(G \wedge H)_x = \mathrm{Tr}(G)_x \wedge \mathrm{Tr}(H)_x$;
- $\mathrm{Tr}(G \vee H)_x = \mathrm{Tr}(G)_x \vee \mathrm{Tr}(H)_x$;
- $\mathrm{Tr}(G \supset H)_x = \mathrm{Tr}(G)_x \supset \mathrm{Tr}(H)_x$;
- $\mathrm{Tr}(\Box G)_x = \forall y\big(R(x,y) \supset \mathrm{Tr}(G)_y\big)$, here y is a new variable and R is a predicate variable, which represents the relation of Kripke structure.
- $\mathrm{Tr}(\Diamond G)_x = \exists y\big(R(x,y) \wedge \mathrm{Tr}(G)_y\big)$, here y is a new variable.

Property	Formula
Reflexivity	$\forall x R(x,x)$
Completeness	$\forall x \exists y R(x,y)$
Symmetricity	$\forall x \forall y\big(R(x,y) \supset R(y,x)\big)$
Transitivity	$\forall x \forall y \forall z\Big(\big(R(x,y) \wedge R(y,z)\big) \supset R(x,z)\Big)$
Euclidicity	$\forall x \forall y \forall z\Big(\big(R(x,y) \wedge R(x,z)\big) \supset R(y,z)\Big)$

E.g. if $\sigma = S4$, which contains axioms *(T)* and *(4)*, then formulas for reflexivity and transitivity must be included into F_σ, thus

$$F_\sigma = \forall x R(x,x) \wedge \forall x \forall y \forall z\Big(\big(R(x,y) \wedge R(y,z)\big) \supset R(x,z)\Big)$$

Example 4.5.1. Let's find $\mathrm{Tr}\left(\Box\Diamond(p \supset q)\right)_x$:

$$\mathrm{Tr}\left(\Box\Diamond(p \supset q)\right)_x = \forall y\Big(R(x,y) \supset \mathrm{Tr}\left(\Diamond(p \supset q)\right)_y\Big) =$$

$$= \forall y\Big(R(x,y) \supset \exists z\big(R(y,z) \wedge \mathrm{Tr}(p \supset q)_z\big)\Big) =$$

$$= \forall y\Big(R(x,y) \supset \exists z\Big(R(y,z) \wedge \big(\mathrm{Tr}(p)_z \supset \mathrm{Tr}(q)_z\big)\Big)\Big) =$$

$$= \forall y\Big(R(x,y) \supset \exists z\Big(R(y,z) \wedge \big(P(z) \supset Q(z)\big)\Big)\Big)$$

Datalog

Example 3.1.1. A deductive database can be defined as follows.
Facts:

Father(John, Eve)	Mother(Christine, Eve)
Father(Peter, Monica)	Mother(Eve, Amy)

Rules:

$$\begin{aligned}
&\text{Relative}(x, y) :- \text{Father}(x, y)\\
&\text{Relative}(x, y) :- \text{Mother}(x, y)\\
&\text{Relative}(x, z) :- \text{Mother}(x, y) \wedge \text{Relative}(y, z)\\
&\text{Relative}(x, z) :- \text{Father}(x, y) \wedge \text{Relative}(y, z)
\end{aligned}$$

Restrictions:

$$\begin{aligned}
&:- \text{Father}(x, x)\\
&:- \text{Mother}(x, x)\\
&:- \text{Father}(x, y) \wedge \text{Mother}(x, z)
\end{aligned}$$

Herbrand universe is {John, Eve, Christine, Peter, Monica, Amy}. Herbrand model is:

Father(John, Eve)	Mother(Christine, Eve)
Father(Peter, Monica)	Mother(Eve, Amy)
Relative(John, Eve)	Relative(Christine, Eve)
Relative(Peter, Monica)	Relative(Eve, Amy)
Relative(Christine, Amy)	Relative(John, Amy)

A set of facts can also be presented as tables:

Father	
John	Eve
Peter	Monica

Mother	
Christine	Eve
Eve	Amy

There are three types of predicates in a database of Datalog:

1. *Extensional.* They are defined using only facts. The set of extensional predicates is called *extensional database* (denoted EDB).
2. *Intentional.* They are defined by rules. Every predicate, which occurs in a head of at least one rule is intentional. The set of intentional predicates is called *intentional database* (denoted IDB). IDB is also called a *Datalog program.*
3. Equality predicate = and other predicates already included in database syntax.

3.2 Programs with negation operator

In standard Datalog if formula F is not derivable from database, then it is assumed that $\neg F$ is derivable.

Example 3.2.1. Suppose the database describes the flight schedule. The EDB is:

$$\begin{aligned}
&\text{Flight}(\text{Vilnius}, \text{Prague})\\
&\text{Flight}(\text{Vilnius}, \text{Oslo})\\
&\text{Flight}(\text{Vilnius}, \text{Tallinn})\\
&\text{Flight}(\text{Vilnius}, \text{Helsinki})
\end{aligned}$$

The IDB is empty and query is $?-\text{Flight}(\text{Vilnius}, \text{Beijing})$. Because predicate Flight(Vilnius, Beijing) is not derivable, it is assumed that formula $\neg\text{Flight}(\text{Vilnius}, \text{Beijing})$ is true.

Such interpretation of negation is called *closed world semantics*. It is assumed that $\neg F$ is true if it is not clear (not derivable, impossible to prove) that F is true.

If sets $\{l_1, \ldots, l_{i-1}, F, l_{i+1}, \ldots, l_n\}$ and $\{l_1, \ldots, l_{i-1}, \neg F, l_{i+1}, \ldots, l_n\}$ are Herbrand models of one database, then it is said that they are *independent* from atomic formula F and instead of two models, only one is used: $\{l_1, \ldots, l_{i-1}, l_{i+1}, \ldots, l_n\}$. Let's analyse only Herbrand models, in which negative atomic formulas are omitted and there are no atomic formulas from which they are independent. Suppose that $\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_m$ are Herbrand models. Model $\mathcal{I}_j$ is called *minimal* if it is a subset of every model, i.e. $\mathcal{I}_j \subseteq \mathcal{I}_k$ for every $k \in [1, m]$.

If rules that are not Horn disjuncts are analysed, then it is possible to have several minimal models for one database.

Example 3.2.2. Suppose database is:

$$\begin{aligned}
&P(a)\\
&Q(x) :- P(x), \neg R(x)\\
&R(x) :- P(x), \neg Q(x)
\end{aligned}$$

This database has two models: $\{P(a), Q(a)\}$ and $\{P(a), R(a)\}$. Therefore, it is impossible to find a single minimal model.