# Introduction to Sequent Calculus and Abstract Logic Programming

## Course Notes

These notes refer to an old course, they are not meant for DS 2001!

Alessio Guglielmi

Technische Universität Dresden
Hans-Grundig-Str. 25 - 01062 Dresden - Germany
Alessio.Guglielmi@Inf.TU-Dresden.DE

## 1 Introduction and Aim of the Course

Sequent calculus was introduced by Gentzen in 1935. Inference rules, like in natural deduction, directly model syntactical properties of logical connectives, as opposed to the Hilbert-Tarski tradition of using axioms to this purpose.

In recent years, the logic community started new and broad studies in the syntactical realm, and sequent calculus has become the main logical instrument for the new investigations. In particular, it has been proven a very effective way to study and design new logic programming languages, whose more mature offspring is now $\lambda$-Prolog. Syntax in sequent calculus is much closer to operational semantics than the old methods based on traditional logical semantics, like model theory, so the new sequent calculus perspective permits easier and more natural development of languages.

Purpose of this short course is to present the fundamentals of sequent calculus syntax, in comparison to the more familiar natural deduction syntax. The cut-elimination theorem will be presented. The main goal of the course will be the use of sequent calculus as a way to design logic programming languages. The concept of abstract logic programming will be presented and examples of languages in this setting will be given.

This course is to be considered introductory to a course in $\lambda$-Prolog which should be offered next year.

## 2 Study Material, Final Exam and Exercises

After each lecture a new handout is given to the students covering the last topics. The examination material is exactly what is covered in these notes, except where explicitly indicated. The exercises in the notes are representative of what shall be found in the final exam.

It is expected that students read all the material which is cited in the notes, which are rather sketchy, but what is not in the notes is not required to successfully complete the examination. These notes are not independent, they are just a means to organize and connect material from different sources.

For a discussion about syntactic vs. semantic tradition, read chapter 1 in [2]. Chapters 2 to 5 of the same book are also a very good reading for those who are interested in a better understanding of the first part of the course.

The relation between natural deduction and sequent calculus and the cut-elimination theorem shall be presented following [1]. In these notes you will find just a scheme of what is required from [1].

Abstract Logic Programming is presented following [3].

During the course three assignments shall be given. Each of them can contribute for 33.3% to the final grade. Every student is free to give her solution to me for correction, and if she wish I can evaluate the homework and consider it for the final grade. For example, if a student gives me three times her homework, and two times she asks for grade consideration, 66.6% of her final grade will be determined by the scores she gets from the homework, and the remaining 33.4% will be from the final. The final shall consist in written exercises in the same style of the ones considered during the course.

## 3 Syntax in the Hilbert-Tarski Tradition

Logic in the tradition of Hilbert and Tarski was primarily *semantics oriented*. The central interest was in *model theory*, problems were mainly inspired by *set theory*. In general the emphasis was on *infinite mathematical structures*, like models in general are.

The syntactical counterpart, *i.e.* a formal system in which valid sentences can be derived, was presented in a rather obscure way, from our point of view. Computer science is particularly interested in *finite structures*, and the formal theory of languages is more concerned about the *connectives* of a logical system, and their relations, than in traditional models.

In this section we will show syntax in the Hilbert-Tarski tradition; in the following ones we will see that we can replace it by more suitable formal systems. First of all, some notation.

**3.1** *First order formulae* are denoted by $A$, $B$, $C$.

A formal system following Hilbert and Tarski consists in axioms and inference rules. There are several equivalent ways of presenting such a system, the following is one of the simplest, giving syntax to *propositional classical logic*.

**3.2** Let $\mathsf{HT}$ be the formal system whose axiom schemes are:

$$(\mathsf{HT}_1) \qquad A \supset (B \supset A),$$
$$(\mathsf{HT}_2) \qquad (A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C)),$$
$$(\mathsf{HT}_3) \qquad (\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B),$$

and whose inference rule is *modus ponens*:

$$\mathsf{mp}\ \frac{A \quad A \supset B}{B}\,.$$

**3.3** $\mathsf{HT}$ can be extended to *first order classical logic* by adding the axioms

$$(\mathsf{HT}_4) \qquad \forall x.A \supset A[t/x],$$
$$(\mathsf{HT}_5) \qquad \forall x.(A \supset B) \supset (A \supset \forall x.B),$$

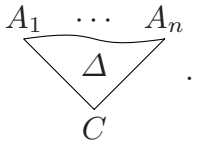
where $t$ is any term and $A[t/x]$ stands for the formula obtained from $A$ by substituting $x$ with $t$. The following inference rule (*generalization*) is also added:

$$\mathsf{gen}\ \frac{A}{\forall x.A}\,.$$

All the relations between connectives are derivable from the axiom schemes provided. In a sense, they are hard-coded inside the axioms, and

there is no intuitive way to reason about them, except by using *semantics* (truth tables, for example). Semantic methods tend to stick to logics whose semantics is well-known, or nice, as classical logic; they are not necessarily suitable to all logics.

In HT, and all the formal systems to come, *deductions* (or *derivations*, or *proofs*) are trees, whose leaves are instances of axioms (*hypotheses*) and whose root is the conclusion of the deduction, *i.e.* a *theorem*. Consider for example the following deduction $\Delta$, whose axioms are $A_1, \ldots, A_n$ and whose conclusion is $C$:




.

There are two ways in which we can look at deductions:

1) *Top-down*. From the top-down point of view, one starts from the axioms and reaches a conclusion. This is the usual, *declarative* way of looking at proofs. The semantics of inference rules is the usual one, for example the mp's semantics is the syllogism.

2) *Bottom-up*. One starts from the conclusion and reaches the axioms. The classical problem in automated deduction is finding a proof for a proposition. It is usually a bottom-up construction of a proof. This point of view corresponds to an *operational* semantics, in particular of the connectives of the logic at hand.

While HT is adequate to a top-down approach, it is completely inadequate to a bottom-up one. In particular the mp rule has two extremely serious drawbacks:

1) Given conclusion $B$, formula $A$ must be invented out of thin air, $A$ has no relation with $B$.

2) Formulae in the search become bigger and bigger: one starts from $B$ and has to look for proofs for $A$ and $A \supset B$.

These reasons are enough to make us look for alternative foundations than HT for syntax-based automated deduction methods. We will see that we can do much better.

**Axiom**

$$\Gamma, x\colon A \vdash A$$

**Introduction Rules** **Elimination Rules**

$$\supset_{\mathsf{I}} \frac{\Gamma, x\colon A \vdash B}{\Gamma \vdash A \supset B} \qquad \supset_{\mathsf{E}} \frac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B}$$

$$\wedge_{\mathsf{I}} \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \qquad \wedge_{\mathsf{EL}} \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \qquad \wedge_{\mathsf{ER}} \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}$$

$$\vee_{\mathsf{IL}} \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \qquad \vee_{\mathsf{IR}} \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \qquad \vee_{\mathsf{E}} \frac{\Gamma \vdash A \vee B \quad \Gamma, x\colon A \vdash C \quad \Gamma, y\colon B \vdash C}{\Gamma \vdash C}$$

$$\bot_{\mathsf{E}} \frac{\Gamma \vdash \bot}{\Gamma \vdash A} \text{ where } A \neq \bot$$

**Fig. 1** *Intuitionistic Propositional Natural Deduction* $\mathsf{N}_{\mathsf{I}}^{\supset,\wedge,\vee,\bot}$

## 4 Natural Deduction

As we have done with system HT, for natural deduction we shall stick to a propositional system. First order shall be introduced later on, with sequent systems.

Following [1], for natural deduction we will use a particular form of *sequents*, to take care in a clean way of hypothesis discarding. As a general remark, we can say that sequents are expressions of the form

$$\Gamma \vdash \Delta,$$

where $\Gamma$ and $\Delta$ are sets, multisets or sequences of formulae. Intuitively, we think of $\Gamma$ as a *conjunction* and of $\Delta$ as a *disjunction*. $\vdash$, called "entailment," corresponds to logical implication: $\Gamma$ *entails* $\Delta$ means that from all the formulae in $\Gamma$ we can conclude some of the formulae in $\Delta$. Please take this as a very general and imprecise statement, because there are numerous exceptions. Nonetheless, it is useful considering this as a guiding idea.

To begin, we have a particular form of sequents, which is enough for natural deduction.

**4.1** A *sequent* is a formal expression of the form

$$\Gamma \vdash A,$$

where $\Gamma$ stands for a finite set of the form $\{x_1\colon A_1, \ldots, x_n\colon A_n\}$, where the $x_i$ are pairwise distinct labels. If $\Gamma = x_1\colon A_1, \ldots, x_n\colon A_n$, the notation $\Gamma, x\colon A$ is well defined only when $x \neq x_i$, for $1 \leqslant i \leqslant n$, in which case it denotes the set $\{x_1\colon A_1, \ldots, x_n\colon A_n, x\colon A\}$.

**4.2** We define $\neg A$ as an abbreviation of $A \supset \bot$.

**4.3** The system $\mathsf{N}_{\mathsf{I}}^{\supset,\wedge,\vee,\bot}$ of *intuitionistic propositional natural deduction* is defined in fig. 1.

$$\supset_{\mathsf{I}} \frac{\wedge_{\mathsf{I}} \dfrac{x\colon A, y\colon B \vdash A \quad x\colon A, y\colon B \vdash B}{x\colon A, y\colon B \vdash A \wedge B}}{\supset_{\mathsf{I}} \dfrac{\wedge_{\mathsf{EL}} \dfrac{x\colon A, y\colon B \vdash A \wedge B}{x\colon A, y\colon B \vdash A}}{x\colon A \vdash B \supset A}} \over \vdash A \supset (B \supset A)$$

**Fig. 2** *Proof of* $\mathsf{HT}_1$ *in* $\mathsf{N}_{\mathsf{I}}^{\supset,\wedge,\vee,\bot}$

$$\supset_{\mathsf{E}} \frac{\supset_{\mathsf{E}} \dfrac{\supset_{\mathsf{I}} \dfrac{\supset_{\mathsf{I}} \dfrac{\supset_{\mathsf{E}} \dfrac{\Gamma \vdash A \supset (B \supset C) \quad \Gamma \vdash A}{\Gamma \vdash B \supset C} \quad \supset_{\mathsf{E}} \dfrac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B}}{x\colon A \supset (B \supset C), y\colon A \supset B, z\colon A \vdash C}}{x\colon A \supset (B \supset C), y\colon A \supset B \vdash A \supset C}}{x\colon A \supset (B \supset C) \vdash (A \supset B) \supset (A \supset C)}}{\vdash (A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))}$$

where $\Gamma = \{x\colon A \supset (B \supset C), y\colon A \supset B, z\colon A\}$

**Fig. 3** *Proof of* $\mathsf{HT}_2$ *in* $\mathsf{N}_{\mathsf{I}}^{\supset,\wedge,\vee,\bot}$

**Axiom**

$$\Gamma, x\colon A \vdash A$$

**Introduction Rules**

$$\supset_{\mathsf{I}} \frac{\Gamma, x\colon A \vdash B}{\Gamma \vdash A \supset B} \qquad \wedge_{\mathsf{I}} \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$$

$$\vee_{\mathsf{IL}} \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \qquad \vee_{\mathsf{IR}} \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}$$

**Elimination Rules**

$$\supset_{\mathsf{E}} \frac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B} \qquad \wedge_{\mathsf{EL}} \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \qquad \wedge_{\mathsf{ER}} \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}$$

$$\vee_{\mathsf{E}} \frac{\Gamma \vdash A \vee B \quad \Gamma, x\colon A \vdash C \quad \Gamma, y\colon B \vdash C}{\Gamma \vdash C}$$

$$\mathsf{bc} \frac{\Gamma, x\colon \neg A \vdash \bot}{\Gamma \vdash A} \text{ where } A \neq \bot$$

**Fig. 4** *Classical Propositional Natural Deduction* $\mathsf{N}_{\mathsf{C}}^{\supset,\wedge,\vee,\bot}$

As the reader can see, connectives are more explicitly defined in inference rules, compared to $\mathsf{HT}$, and axioms are reduced to trivialities. Observe also some symmetries between introduction and elimination rules.

**4.4 Example** In $\mathsf{N}_{\mathsf{I}}^{\supset,\wedge,\vee,\bot}$ we are able to prove the two $\mathsf{HT}$ axioms $\mathsf{HT}_1$ and $\mathsf{HT}_2$. See figs. 2 and 3.

Modifying the $\bot$ elimination rule into the *by-contra* rule, we obtain a system for classical propositional logic.

**4.5** The system $\mathsf{N}_{\mathsf{C}}^{\supset,\wedge,\vee,\bot}$ of *classical propositional natural deduction* is defined in fig. 4.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash \neg B \supset \neg A \quad \Gamma \vdash \neg B}{\Gamma \vdash \neg A}\supset_{\mathsf{E}} \quad \dfrac{\Gamma \vdash \neg B \supset A \quad \Gamma \vdash \neg B}{\Gamma \vdash A}\supset_{\mathsf{E}}}{x\colon \neg B \supset \neg A, y\colon \neg B \supset A, z\colon \neg B \vdash \bot}\,\mathsf{bc}}{x\colon \neg B \supset \neg A, y\colon \neg B \supset A \vdash B}\supset_{\mathsf{I}}}{x\colon \neg B \supset \neg A \vdash (\neg B \supset A) \supset B}\supset_{\mathsf{I}}}{\vdash (\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)}$$

where $\Gamma = \{x\colon \neg B \supset \neg A, y\colon \neg B \supset A, z\colon \neg B\}$

**Fig. 5** *Proof of* $\mathsf{HT}_3$ *in* $\mathsf{N}_{\mathsf{c}}^{\supset,\wedge,\vee,\bot}$

Everything which is provable in $\mathsf{N}_{\mathsf{I}}^{\supset,\wedge,\vee,\bot}$ is provable in $\mathsf{N}_{\mathsf{c}}^{\supset,\wedge,\vee,\bot}$.

**4.6 Exercise** Prove this last statement.

In $\mathsf{N}_{\mathsf{c}}^{\supset,\wedge,\vee,\bot}$ proving axiom $\mathsf{HT}_3$ is possible, as it is shown in the next example.

**4.7 Example** In fig. 5 a proof in $\mathsf{N}_{\mathsf{c}}^{\supset,\wedge,\vee,\bot}$ of $\mathsf{HT}_3$ is shown. Please remember that $\neg A$ is just a shortcut for $A \supset \bot$.

**4.8 Exercise** Prove the following three formulae in $\mathsf{N}_{\mathsf{c}}^{\supset,\wedge,\vee,\bot}$, remembering that to prove $A \equiv B$ one has to prove $A \supset B$ and $B \supset A$:

1) $\neg\neg A \equiv A$,

2) $(A \supset B) \supset (\neg B \supset \neg A)$,

3) $(A \vee \neg B) \equiv \neg(\neg A \wedge B)$.

In $\mathsf{N}_{\mathsf{I}}^{\supset,\wedge,\vee,\bot}$ the so-called *Curry-Howard isomorphism* between natural deduction and simply typed $\lambda$-calculus can be established. We have introduced labeled hypotheses to conform to the notation in [1], where this isomorphism is treated.

**Axiom**

$$A, \Gamma \vdash A$$

**Structural Rule**

$$>_{\mathsf{L}} \frac{A, A, \Gamma \vdash C}{A, \Gamma \vdash C}$$

**Left Rules**

$$\supset_{\mathsf{L}} \frac{\Gamma \vdash A \quad B, \Gamma \vdash C}{A \supset B, \Gamma \vdash C}$$

$$\wedge_{\mathsf{L}} \frac{A, B, \Gamma \vdash C}{A \wedge B, \Gamma \vdash C}$$

$$\vee_{\mathsf{L}} \frac{A, \Gamma \vdash C \quad B, \Gamma \vdash C}{A \vee B, \Gamma \vdash C}$$

**Right Rules**

$$\supset_{\mathsf{R}} \frac{A, \Gamma \vdash B}{\Gamma \vdash A \supset B}$$

$$\wedge_{\mathsf{R}} \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$$

$$\vee_{\mathsf{RL}} \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \qquad \vee_{\mathsf{RR}} \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}$$

$$\bot_{\mathsf{R}} \frac{\Gamma \vdash \bot}{\Gamma \vdash A} \text{ where } A \neq \bot$$

**Fig. 6** *Intuitionistic Propositional Sequent Calculus* $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$

## 5 An Intuitionistic Sequent Calculus

In this section we show a calculus for intuitionistic propositional logic in the style of *Gentzen's sequents*. This calculus is equivalent to $\mathsf{N}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$, and derivations in it can be transformed into equivalent deductions in $\mathsf{N}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$, and vice-versa.

We drop labels for hypotheses in sequents, by employing multisets.

**5.1** A *sequent* is a formal expression of the form

$$\Gamma \vdash A,$$

where $\Gamma$ stands for a finite multiset of formulae.

**5.2** The system $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$ of *intuitionistic propositional sequent calculus* is defined in fig. 6.

There exists an effective procedure $\mathsf{N}$ which transforms any derivation $\Pi$ in $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$ into an equivalent deduction $\mathsf{N}(\Pi)$ in $\mathsf{N}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$. $\mathsf{N}$ is not surjective, since it only generates deductions in normal form. We can introduce a new inference rule in our sequent calculus, the *cut*, which helps repairing the situation.

**5.3** The following inference rule is called *cut*:

$$\bowtie \frac{\Gamma \vdash A \quad A, \Gamma \vdash C}{\Gamma \vdash C}$$

**Axiom**

$$A, \Gamma \vdash A$$

**Cut**

$$\bowtie \frac{\Gamma \vdash A \quad A, \Gamma \vdash C}{\Gamma \vdash C}$$

**Structural Rule**

$$>_{\mathsf{L}} \frac{A, A, \Gamma \vdash C}{A, \Gamma \vdash C}$$

**Left Rules**

$$\supset_{\mathsf{L}} \frac{\Gamma \vdash A \quad B, \Gamma \vdash C}{A \supset B, \Gamma \vdash C}$$

$$\wedge_{\mathsf{L}} \frac{A, B, \Gamma \vdash C}{A \wedge B, \Gamma \vdash C}$$

$$\vee_{\mathsf{L}} \frac{A, \Gamma \vdash C \quad B, \Gamma \vdash C}{A \vee B, \Gamma \vdash C}$$

**Right Rules**

$$\supset_{\mathsf{R}} \frac{A, \Gamma \vdash B}{\Gamma \vdash A \supset B}$$

$$\wedge_{\mathsf{R}} \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$$

$$\vee_{\mathsf{RL}} \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \qquad \vee_{\mathsf{RR}} \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}$$

$$\bot_{\mathsf{R}} \frac{\Gamma \vdash \bot}{\Gamma \vdash A} \text{ where } A \neq \bot$$

**Fig. 7** *Intuitionistic Propositional Sequent Calculus with Cut* $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot,\bowtie}$

**5.4** The system $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot,\bowtie}$ of *intuitionistic propositional sequent calculus with cut* is defined in fig. 7.

The procedure $\mathsf{N}$ can be extended to $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot,\bowtie}$ in order to make it surjective over $\mathsf{N}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$. The converse procedure $\mathsf{G}$ may be defined: for every deduction $\Delta$ in $\mathsf{N}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$ an equivalent derivation $\mathsf{G}(\Delta)$ in $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot,\bowtie}$ can be obtained. Please look for the details of $\mathsf{G}$ and $\mathsf{N}$ in [1].

At this point the following natural question is mandatory: is $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot}$ equivalent to $\mathsf{G}_{\mathsf{i}}^{\supset,\wedge,\vee,\bot,\bowtie}$?

**Axiom**

$$A, \Gamma \vdash \Delta, A$$

**Cut**

$$\bowtie \frac{\Gamma \vdash \Delta, A \quad A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta}$$

**Structural Rules**

$$>_{\mathsf{L}} \frac{A, A, \Gamma \vdash \Delta}{A, \Gamma \vdash \Delta} \qquad >_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A}$$

**Left Rules**

$$\supset_{\mathsf{L}} \frac{\Gamma \vdash \Delta, A \quad B, \Gamma \vdash \Delta}{A \supset B, \Gamma \vdash \Delta}$$

$$\wedge_{\mathsf{L}} \frac{A, B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}$$

$$\vee_{\mathsf{L}} \frac{A, \Gamma \vdash \Delta \quad B, \Gamma \vdash \Delta}{A \vee B, \Gamma \vdash \Delta}$$

$$\neg_{\mathsf{L}} \frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}$$

$$\forall_{\mathsf{L}} \frac{A[t/x], \Gamma \vdash \Delta}{\forall x.A, \Gamma \vdash \Delta}$$

$$\exists_{\mathsf{L}} \frac{A[y/x], \Gamma \vdash \Delta}{\exists x.A, \Gamma \vdash \Delta}$$

**Right Rules**

$$\supset_{\mathsf{R}} \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \supset B}$$

$$\wedge_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B}$$

$$\vee_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B}$$

$$\neg_{\mathsf{R}} \frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A}$$

$$\forall_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A[y/x]}{\Gamma \vdash \Delta, \forall x.A}$$

$$\exists_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A[t/x]}{\Gamma \vdash \Delta, \exists x.A}$$

where in $\forall_{\mathsf{R}}$ and $\exists_{\mathsf{L}}$ the variable $y$ is not free in the conclusion

**Fig. 8** *First Order Classical Sequent Calculus with Cut* $\mathsf{G}_{\mathsf{c}}^{\supset,\wedge,\vee,\neg,\forall,\exists,\bowtie}$

## 6 Sequent Calculi for First-Order Classical Logic

**6.1** A *sequent* is a formal expression of the form

$$\Gamma \vdash \Delta,$$

where $\Gamma$ and $\Delta$ stand for finite multisets of formulae.

For classical logic, it is better introducing a unary connective for negation.

**6.2** From now on, $\neg A$ is no more an abbreviation for $A \supset \bot$.

**6.3** The system $\mathsf{G}_{\mathsf{c}}^{\supset,\wedge,\vee,\neg,\forall,\exists,\bowtie}$ of *first order classical sequent calculus with cut* is defined in fig. 8.

An equivalent system for first order classical logic, very close to the one adopted in 1935 by Gentzen to prove cut-elimination, is $\mathsf{LK}$.

**Axiom**

$$A \vdash A$$

**Cut**

$$\bowtie \frac{\Gamma \vdash \Delta, A \quad A, \Lambda \vdash \Theta}{\Gamma, \Lambda \vdash \Delta, \Theta}$$

**Structural Rules**

$$>_{\mathsf{L}} \frac{A, A, \Gamma \vdash \Delta}{A, \Gamma \vdash \Delta} \qquad >_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A} \qquad <_{\mathsf{L}} \frac{\Gamma \vdash \Delta}{A, \Gamma \vdash \Delta} \qquad <_{\mathsf{R}} \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}$$

**Left Rules**

$$\supset_{\mathsf{L}} \frac{\Gamma \vdash \Delta, A \quad B, \Gamma \vdash \Delta}{A \supset B, \Gamma \vdash \Delta}$$

$$\wedge_{\mathsf{LL}} \frac{A, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta} \qquad \wedge_{\mathsf{LR}} \frac{B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}$$

$$\vee_{\mathsf{L}} \frac{A, \Gamma \vdash \Delta \quad B, \Gamma \vdash \Delta}{A \vee B, \Gamma \vdash \Delta}$$

$$\neg_{\mathsf{L}} \frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}$$

$$\forall_{\mathsf{L}} \frac{A[t/x], \Gamma \vdash \Delta}{\forall x.A, \Gamma \vdash \Delta}$$

$$\exists_{\mathsf{L}} \frac{A[y/x], \Gamma \vdash \Delta}{\exists x.A, \Gamma \vdash \Delta}$$

**Right Rules**

$$\supset_{\mathsf{R}} \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \supset B}$$

$$\wedge_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B}$$

$$\vee_{\mathsf{RL}} \frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B} \qquad \vee_{\mathsf{RR}} \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \vee B}$$

$$\neg_{\mathsf{R}} \frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A}$$

$$\forall_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A[y/x]}{\Gamma \vdash \Delta, \forall x.A}$$

$$\exists_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A[t/x]}{\Gamma \vdash \Delta, \exists x.A}$$

where in $\forall_{\mathsf{R}}$ and $\exists_{\mathsf{L}}$ the variable $y$ is not free in the conclusion

**Fig. 9** *First Order Classical Sequent Calculus with Multiplicative Cut* $\mathsf{LK}$

**6.4** The system $\mathsf{LK}$ of *first order classical sequent calculus with multiplicative cut* is defined in fig. 9.

It is obtained from $\mathsf{G}_{\mathsf{c}}^{\supset,\wedge,\vee,\neg,\forall,\exists,\bowtie}$ by modifying the axiom and cut rules and the rules for $\wedge_{\mathsf{L}}$ and $\vee_{\mathsf{R}}$. The cut rule now takes a form called *multiplicative*, as opposed to the *additive* one in $\mathsf{G}_{\mathsf{c}}^{\supset,\wedge,\vee,\neg,\forall,\exists,\bowtie}$. A crucial role is played by the *weakening* rules $<_{\mathsf{L}}$ and $<_{\mathsf{R}}$. They, together with the *contraction* rules $>_{\mathsf{L}}$ and $>_{\mathsf{R}}$, essentially model idempotency in sequents, *i.e.* multisets actually behave as sets: $A \wedge A \equiv A$ and $A \vee A \equiv A$.

**6.5 Exercise** Prove that $\mathsf{G}_{\mathsf{c}}^{\supset,\wedge,\vee,\neg,\forall,\exists,\bowtie}$ and $\mathsf{LK}$ are equivalent by giving two effective procedures which transform derivations in one system to equivalent derivations in the other, and vice-versa.

We can answer positively the question we posed at the end of last section. In all sequent calculi we have seen so far, the cut rule is redundant, or *admissible*. We just state the theorem for $\mathsf{LK}$, and you can find the proof in [1].

**6.6 Theorem** *For every proof in* $\mathsf{LK}$ *there exists a proof with the same conclusion in which there is no use of the* $\bowtie$ *rule.*

The preceding *cut-elimination* theorem is the deepest result of proof theory. It is of special importance for automated deduction because it eliminates the need in proofs of a rule which introduces arbitrary formulae in the search (when reading inference rules in a bottom-up way).

**6.7 Exercise** Prove the following formulae in $\mathsf{LK}$:

1) $A \supset (B \supset A)$;
2) $(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))$;
3) $(\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)$;
4) $\forall x.A \supset A[t/x]$;
5) $\forall x.(A \supset B) \supset (A \supset \forall x.B)$, where $x$ is not free in $A$;
6) $(A \supset \exists x.B) \equiv \exists x.(A \supset B)$, where $x$ is not free in $A$;
7) $(\forall x.A \supset B) \equiv \exists x.(A \supset B)$, where $x$ is not free in $B$;
8) $\exists x.\forall y.(p(x) \supset p(y))$.

**6.8 Exercise** Prove that, in $\mathsf{LK}$, the axiom rule can be replaced by the rule

$$a \vdash a,$$

where $a$ is an atom, without affecting provability.

# 7 Informal Introduction to Abstract Logic Programming

In this section we shall introduce the main ideas of abstract logic programming by means of an example case.

**7.1** From now on we will denote atoms with letters $a$, $b$, $c$, ....

Consider *clauses* of the following form:

$$\forall \vec{x}.((b_1 \wedge \cdots \wedge b_h) \supset a),$$

where $h$ can be 0, and in this case a clause is a *fact* $\forall \vec{x}.a$. A *logic programming problem* may be defined as the problem of finding a substitution $\sigma$ such that

$$\mathsf{P} \vdash (a_1 \wedge \cdots \wedge a_k)\sigma$$

is provable in some formal system, where $k > 0$ and $\mathsf{P}$ is a collection of clauses. $\mathsf{P}$ is called the *program*, $(a_1 \wedge \cdots \wedge a_k)$ is the *goal* and $\sigma$ is the *answer substitution*.

Let us restrict ourselves to provability in $\mathsf{LK}$. Suppose that $\mathsf{P}$ is a multiset of clauses which contains the clause

$$\forall \vec{x}.((b_1 \wedge \cdots \wedge b_h) \supset a);$$

consider the goal $(a_1 \wedge \cdots \wedge a_k)$ and suppose that $\sigma$ is a unifier of $a$ and $a_l$, for some $l$ such that $1 \leqslant l \leqslant k$. Consider the following derivation $\Delta$, where we suppose that $h > 0$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{a\sigma \vdash a_l\sigma}}{\vdots}}{\mathsf{P}, a\sigma \vdash a_l\sigma}\,{<_{\mathsf{L}}} \quad \dfrac{\mathsf{P} \vdash (b_1 \wedge \cdots \wedge b_h)\sigma}{\mathsf{P} \vdash (b_1 \wedge \cdots \wedge b_h)\sigma, a_l\sigma}\,{<_{\mathsf{R}}}}{\mathsf{P}, ((b_1 \wedge \cdots \wedge b_h) \supset a)\sigma \vdash a_l\sigma}\,{\supset_{\mathsf{L}}}}{\vdots}\,{\forall_{\mathsf{L}}}}{\mathsf{P}, \forall \vec{x}.((b_1 \wedge \cdots \wedge b_h) \supset a) \vdash a_l\sigma}\,{\forall_{\mathsf{L}}}}{\mathsf{P} \vdash a_l\sigma}\,{>_{\mathsf{L}}}}{\vdots}\,{\wedge_{\mathsf{R}}} \quad \cdots \quad \dfrac{\mathsf{P} \vdash a_1\sigma}{\vdots}\,{\wedge_{\mathsf{R}}} \quad \cdots \quad \dfrac{\mathsf{P} \vdash a_k\sigma}{\vdots}\,{\wedge_{\mathsf{R}}}}{\mathsf{P} \vdash (a_1 \wedge \cdots \wedge a_k)\sigma}.$$

$\Delta$ has one leaf which is an axiom $(a\sigma \vdash a_l\sigma)$; one leaf which stands for the logic programming problem $\mathsf{P} \vdash (b_1 \wedge \cdots \wedge b_h)\sigma$ and zero or more leaves $\mathsf{P} \vdash a_i\sigma$, for $1 \leqslant i \leqslant k$ and $i \neq l$. In the special case where $h = 0$ we

consider the derivation $\Delta'$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{a\sigma \vdash a_l\sigma}\,{<_{\mathsf{L}}}}{\vdots}}{\mathsf{P}, a\sigma \vdash a_l\sigma}\,{<_{\mathsf{L}}}}{\vdots}}{\mathsf{P}, \forall\vec{x}.a \vdash a_l\sigma}\,{\forall_{\mathsf{L}}}}{\mathsf{P} \vdash a_l\sigma}\,{>_{\mathsf{L}}}}{\dfrac{\mathsf{P} \vdash a_1\sigma}{\vdots}\,{\wedge_{\mathsf{R}}} \quad \cdots \quad \dfrac{\mathsf{P} \vdash a_l\sigma}{\vdots} \quad \cdots \quad \dfrac{\mathsf{P} \vdash a_k\sigma}{\vdots}\,{\wedge_{\mathsf{R}}}}{\mathsf{P} \vdash (a_1 \wedge \cdots \wedge a_k)\sigma}\,{\wedge_{\mathsf{R}}}\,.$$

For both cases we can consider a special inference rule $\mathsf{b}$, called *backchain*, which stands for $\Delta$ or $\Delta'$:

$$\mathsf{b}\,\dfrac{\mathsf{P} \vdash a_1\sigma \;\cdots\; \mathsf{P} \vdash a_{l-1}\sigma \quad \mathsf{P} \vdash (b_1 \wedge \cdots \wedge b_h)\sigma \quad \mathsf{P} \vdash a_{l+1}\sigma \;\cdots\; \mathsf{P} \vdash a_k\sigma}{\mathsf{P} \vdash (a_1 \wedge \cdots \wedge a_k)\sigma}\,,$$

where $\mathsf{P} \vdash (b_1 \wedge \cdots \wedge b_h)\sigma$ may be absent if $h = 0$. The rule degenerates to an axiom when $h = 0$ and $k = 1$.

A backchain rule, when read from bottom to top, reduces one logic programming problem to zero or more logic programming problems. Please notice that every application of a backchain rule is associated to a substitution (in the case above it is $\sigma$) which allows us to use an axiom leaf. We can consider each such substitution as the answer substitution produced by the particular instance of the $\mathsf{b}$ rule. It is not necessarily an mgu! Of course, the "best" choice may be an mgu, which in the first-order case is unique up to variable renaming. The composition of all answer substitutions gives us the final answer substitution to a particular logic programming problem.

We can look for proofs solving a logic programming problem which are entirely built over instances of $\mathsf{b}$ rules. Here is an example.

**7.2 Example** Consider the following logic programming problem:

$$\mathsf{P} \vdash p(x),$$

where

$$\begin{aligned}\mathsf{P} = \{&\forall y.(q(y) \wedge r(y) \supset p(y)),\\ &q(1),\\ &r(1),\\ &r(2)\}.\end{aligned}$$

The search space of proofs for this problem is shown in fig. 10. Every maximal path in the transition system corresponds to a computation, where

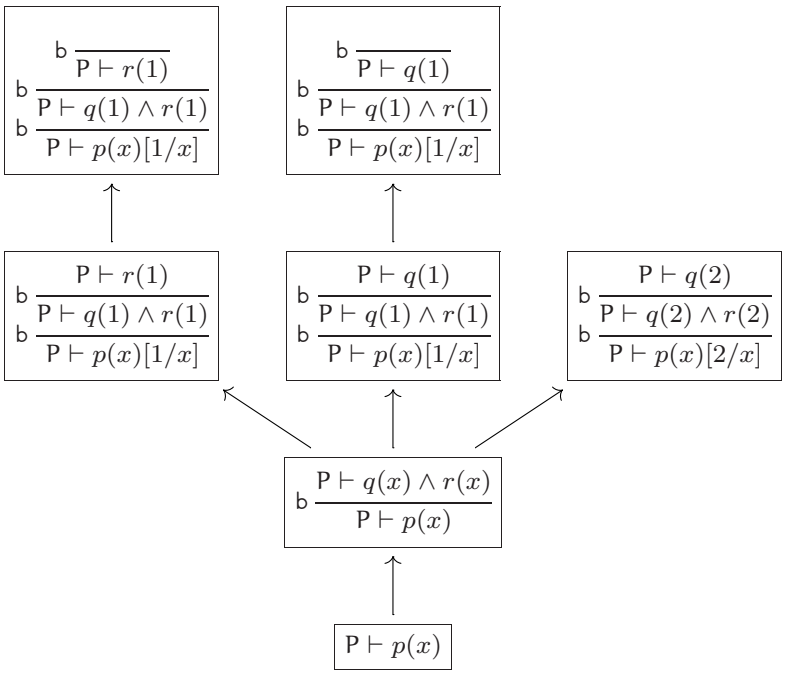



**Fig. 10** *Proof search space for example 7.2*

the substitution applied to the conclusion is the answer substitution. The left and center computations are *successful*, because a proof is found at the end; the right computation instead *fails*: there is no proof for the logic programming problem $\mathsf{P} \vdash q(2)$.

Infinite computations may be specified as well, as the following example shows.

**7.3 Example** Consider the following logic programming problem:

$$\mathsf{P} \vdash a,$$

where

$$\mathsf{P} = \{a \wedge b \supset a\}.$$

The search space of proofs for this problem is shown in fig. 11.

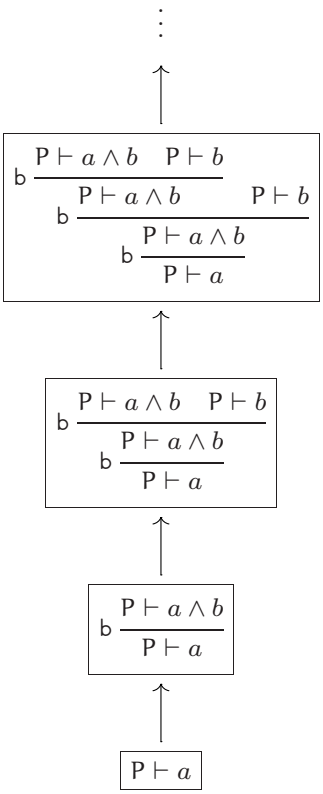



**Fig. 11** *Proof search space for example 7.3*

**Axioms**

$$a, \Gamma \vdash \Delta, a \qquad \bot, \Gamma \vdash \Delta, \bot \qquad \Gamma \vdash \Delta, \top$$

**Left Rules**

$$\supset_{\mathsf{L}} \frac{\Gamma \vdash \Delta, A \quad B, \Gamma \vdash \Theta}{A \supset B, \Gamma \vdash \Delta, \Theta}$$

$$\wedge_{\mathsf{L}} \frac{A, B, \Gamma \vdash \Delta}{A \wedge B, \Gamma \vdash \Delta}$$

$$\vee_{\mathsf{L}} \frac{A, \Gamma \vdash \Delta \quad B, \Gamma \vdash \Delta}{A \vee B, \Gamma \vdash \Delta}$$

$$\forall_{\mathsf{L}} \frac{A[t/x], \Gamma \vdash \Delta}{\forall x.A, \Gamma \vdash \Delta}$$

$$\exists_{\mathsf{L}} \frac{A[y/x], \Gamma \vdash \Delta}{\exists x.A, \Gamma \vdash \Delta}$$

**Right Rules**

$$\supset_{\mathsf{R}} \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \supset B}$$

$$\wedge_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B}$$

$$\vee_{\mathsf{RL}} \frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B} \qquad \vee_{\mathsf{RR}} \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \vee B}$$

$$\forall_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A[y/x]}{\Gamma \vdash \Delta, \forall x.A}$$

$$\exists_{\mathsf{R}} \frac{\Gamma \vdash \Delta, A[t/x]}{\Gamma \vdash \Delta, \exists x.A}$$

$$\bot_{\mathsf{R}} \frac{\Gamma \vdash \Delta, \bot}{\Gamma \vdash \Delta, A} \text{ where } A \neq \bot$$

where in $\forall_{\mathsf{R}}$ and $\exists_{\mathsf{L}}$ the variable $y$ is not free in the conclusion

**Fig. 12** *Miller's sequent system*

## 8 Abstract Logic Programming

Following [3], we introduce a new sequent system, where structural rules have been eliminated, and their "functionality" is included in the structure of sequents.

**8.1** Sequent system MS is built on sequents of the kind

$$\Gamma \vdash \Delta,$$

where $\Gamma$ and $\Delta$ are sets of formulae. $\top$ and $\bot$ are formulae different from atoms. Axioms and inference rules of MS are shown in fig. 12.

In MS negation is obtained through $\neg A \equiv A \supset \bot$.

**8.2 Exercise** Prove that MS and LK are equivalent (consider $\top \equiv A \vee \neg A$ and $\bot \equiv A \wedge \neg A$, for example).

**8.3** A proof in MS is called a *C-proof*. If in a proof each sequent has a singleton set as its right-hand side, the proof is called an *I-proof*.

**8.4** A *uniform proof* is an I-proof in which each occurrence of a sequent whose right-hand side contains a non-atomic formula is the lower sequent of the inference rule that introduces its top-level connective.

It is important to note that the backchain rule we introduced in the preceding section produces only uniform proofs.

**8.5 Exercise** Prove that there are no uniform proofs for $A \vee B \vdash A \vee B$ and $\vdash \exists x.\forall y.(p(x) \supset p(y))$.

Several languages of goals and clauses may be defined (not necessarily with formulae in $\mathsf{MS}$). Also, several intuitionistic sequent systems may be considered, different from $\mathsf{MS}$, for which uniform provability can be contemplated. Our goal now is to define a *completeness* notion which allows us to say, for a particular language, that looking for uniform proofs is equivalent to looking for proofs. Since the search for uniform proofs is computationally efficient (by definition!), we have a powerful guide to assist us in the syntax-driven definition of logic programming languages.

**8.6** Consider a language $\mathsf{D}$ of *clauses* and a language $\mathsf{G}$ of *goals*, both of them subsets of a language for which an entailment relation $\vdash$ is defined; let $\mathsf{P}$ be a finite set of clauses, called *program*; we say that $\langle \mathsf{D}, \mathsf{G}, \vdash \rangle$ is an *abstract logic programming language* if, whenever $\mathsf{P} \vdash G$ is provable, then there is a uniform proof for it, for every program $\mathsf{P}$ and goal $G$.

The property of being an abstract logic programming language is usually proved by examining the *permutability relations* between pairs of inference rules in the sequent system which defines $\vdash$. For example, consider the following fragment of derivation:

$$\supset_{\mathsf{L}} \frac{\Gamma \vdash A \qquad \vee_{\mathsf{RL}} \dfrac{B, \Gamma \vdash C}{B, \Gamma \vdash C \vee D}}{A \supset B, \Gamma \vdash C \vee D};$$

it clearly cannot be part of a uniform proof. The situation may be repaired if we *permute* the $\supset_{\mathsf{L}}$ and $\vee_{\mathsf{RL}}$ rules, like this:

$$\begin{array}{c} \supset_{\mathsf{L}} \dfrac{\Gamma \vdash A \quad B, \Gamma \vdash C}{A \supset B, \Gamma \vdash C} \\ \vee_{\mathsf{RL}} \dfrac{}{A \supset B, \Gamma \vdash C \vee D} \end{array}.$$

**8.7 Exercise** Consider the following languages of goals and clauses:

$$\mathsf{G} := \mathsf{A} \mid \mathsf{D} \supset \mathsf{A} \mid \mathsf{G} \vee \mathsf{G},$$
$$\mathsf{D} := \mathsf{A} \mid \mathsf{G} \supset \mathsf{A} \mid \forall x.\mathsf{D},$$

where $\mathsf{A}$ are atoms. Prove that $\langle \mathsf{D}, \mathsf{G}, \vdash \rangle$ is an abstract logic programming language, where $\vdash$ is entailment as it is defined by (the intuitionistic fragment of) $\mathsf{MS}$.

# 9 Solutions to Some Exercises

In these section one can find solutions to the exercises proposed in classes during the course.

**Exercise 4.8**

1a)

$$\supset_{\mathsf{E}} \frac{x\colon \neg\neg A, y\colon \neg A \vdash \neg\neg A \quad x\colon \neg\neg A, y\colon \neg A \vdash \neg A}{\mathsf{bc}\,\dfrac{x\colon \neg\neg A, y\colon \neg A \vdash \bot}{\supset_{\mathsf{I}}\,\dfrac{x\colon \neg\neg A \vdash A}{\vdash \neg\neg A \supset A}}}$$

1b)

$$\supset_{\mathsf{E}} \frac{x\colon A, y\colon \neg A \vdash \neg A \quad x\colon A, y\colon \neg A \vdash A}{\supset_{\mathsf{I}}\,\dfrac{x\colon A, y\colon \neg A \vdash \bot}{\supset_{\mathsf{I}}\,\dfrac{x\colon A \vdash \neg\neg A}{\vdash A \supset \neg\neg A}}}$$

2)

$$\supset_{\mathsf{E}} \frac{\supset_{\mathsf{E}} \dfrac{\supset_{\mathsf{E}} \dfrac{\Gamma \vdash \neg B \quad \supset_{\mathsf{E}} \dfrac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B}}{x\colon A \supset B, y\colon \neg B, z\colon A \vdash \bot}}{\supset_{\mathsf{I}}\,\dfrac{x\colon A \supset B, y\colon \neg B \vdash \neg A}{x\colon A \supset B \vdash \neg B \supset \neg A}}}{\supset_{\mathsf{I}}\,\dfrac{}{\vdash (A \supset B) \supset (\neg B \supset \neg A)}}$$

where $\Gamma = \{x\colon A \supset B, y\colon \neg B, z\colon A\}$.

3a)

$$\vee_{\mathsf{E}} \frac{\Gamma \vdash A \vee \neg B \quad \supset_{\mathsf{E}} \dfrac{\wedge_{\mathsf{EL}} \dfrac{\Gamma_1 \vdash \neg A \wedge B}{\Gamma_1 \vdash \neg A} \quad \Gamma_1 \vdash A}{\Gamma_1 \vdash \bot} \quad \supset_{\mathsf{E}} \dfrac{\Gamma_2 \vdash \neg B \quad \wedge_{\mathsf{ER}} \dfrac{\Gamma_2 \vdash \neg A \wedge B}{\Gamma_2 \vdash B}}{\Gamma_2 \vdash \bot}}{\supset_{\mathsf{I}}\,\dfrac{x\colon A \vee \neg B, y\colon \neg A \wedge B \vdash \bot}{\supset_{\mathsf{I}}\,\dfrac{x\colon A \vee \neg B \vdash \neg(\neg A \wedge B)}{\vdash (A \vee \neg B) \supset \neg(\neg A \wedge B)}}}$$

where $\Gamma = \{x\colon A \vee \neg B, y\colon \neg A \wedge B\}$, $\Gamma_1 = \Gamma \cup \{z\colon A\}$ and $\Gamma_2 = \Gamma \cup \{t\colon \neg B\}$.

3b)

$$\supset_{\mathsf{E}} \frac{\Gamma \vdash \neg(\neg A \wedge B) \quad \wedge_{\mathsf{I}} \dfrac{\supset_{\mathsf{E}} \dfrac{\Gamma_1 \vdash \neg(A \vee \neg B) \quad \vee_{\mathsf{IL}} \dfrac{\Gamma_1 \vdash A}{\Gamma_1 \vdash A \vee \neg B}}{\supset_{\mathsf{I}}\,\dfrac{\Gamma, z\colon A \vdash \bot}{\Gamma \vdash \neg A}} \quad \supset_{\mathsf{E}} \dfrac{\Gamma_2 \vdash \neg(A \vee \neg B) \quad \vee_{\mathsf{IR}} \dfrac{\Gamma_2 \vdash \neg B}{\Gamma_2 \vdash A \vee \neg B}}{\mathsf{bc}\,\dfrac{\Gamma, t\colon \neg B \vdash \bot}{\Gamma \vdash B}}}{\Gamma \vdash \neg A \wedge B}}{\mathsf{bc}\,\dfrac{x\colon \neg(\neg A \wedge B), y\colon \neg(A \vee \neg B) \vdash \bot}{\supset_{\mathsf{I}}\,\dfrac{x\colon \neg(\neg A \wedge B) \vdash A \vee \neg B}{\vdash \neg(\neg A \wedge B) \supset (A \vee \neg B)}}}$$

where $\Gamma = \{x\colon \neg(\neg A \wedge B), y\colon \neg(A \vee \neg B)\}$, $\Gamma_1 = \Gamma \cup \{z\colon A\}$ and $\Gamma_2 = \Gamma \cup \{t\colon \neg B\}$.

**Exercise 6.7**

1)

$$\dfrac{\dfrac{\dfrac{A \vdash A}{A, B \vdash A}\,{<_{\mathsf{L}}}}{A \vdash B \supset A}\,{\supset_{\mathsf{R}}}}{\vdash A \supset (B \supset A)}\,{\supset_{\mathsf{R}}}$$

2)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{A \supset (B \supset C), A \vdash A}\,{<_{\mathsf{L}}}}{A \supset (B \supset C), A \vdash A, C}\,{<_{\mathsf{R}}} \quad \dfrac{\dfrac{A \vdash A}{B, A \vdash A}\,{<_{\mathsf{L}}}}{B, A \vdash A, C}\,{<_{\mathsf{R}}} \quad \dfrac{\dfrac{B \vdash B}{B \vdash B, C}\,{<_{\mathsf{R}}} \quad \dfrac{C \vdash C}{C, B \vdash C}\,{<_{\mathsf{L}}}}{B \supset C, B \vdash C}\,{\supset_{\mathsf{L}}}}{B \supset C, B, A \vdash C}\,{<_{\mathsf{L}}}}{A \supset (B \supset C), B, A \vdash C}\,{\supset_{\mathsf{L}}}}{A \supset (B \supset C), A \supset B, A \vdash C}\,{\supset_{\mathsf{R}}}}{A \supset (B \supset C), A \supset B \vdash A \supset C}\,{\supset_{\mathsf{R}}}}{A \supset (B \supset C) \vdash (A \supset B) \supset (A \supset C)}\,{\supset_{\mathsf{R}}}}{\vdash (A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))}\,{\supset_{\mathsf{R}}}$$

3)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \vdash B}{\vdash \neg B, B}\,{\neg_{\mathsf{R}}}}{\neg B \supset A \vdash \neg B, B}\,{<_{\mathsf{L}}} \quad \dfrac{\dfrac{\dfrac{B \vdash B}{\vdash \neg B, B}\,{\neg_{\mathsf{R}}}}{\neg A \vdash \neg B, B}\,{<_{\mathsf{L}}} \quad \dfrac{\dfrac{A \vdash A}{\neg A, A \vdash}\,{\neg_{\mathsf{L}}}}{\neg A, A \vdash B}\,{<_{\mathsf{R}}}}{\neg A, \neg B \supset A \vdash B}\,{\supset_{\mathsf{L}}}}{\neg B \supset \neg A, \neg B \supset A \vdash B}\,{\supset_{\mathsf{L}}}}{\neg B \supset \neg A \vdash (\neg B \supset A) \supset B}\,{\supset_{\mathsf{R}}}}{\vdash (\neg B \supset \neg A) \supset ((\neg B \supset A) \supset B)}\,{\supset_{\mathsf{R}}}$$

4)

$$\dfrac{\dfrac{\dfrac{A[t/x] \vdash A[t/x]}{\forall x.A \vdash A[t/x]}\,{\forall_{\mathsf{L}}}}{\vdash \forall x.A \supset A[t/x]}\,{\supset_{\mathsf{R}}}$$

5)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{A \vdash A, B}\,{<_{\mathsf{R}}} \quad \dfrac{B \vdash B}{B, A \vdash B}\,{<_{\mathsf{L}}}}{A \supset B, A \vdash B}\,{\supset_{\mathsf{L}}}}{\forall x.(A \supset B), A \vdash B}\,{\forall_{\mathsf{L}}}}{\forall x.(A \supset B), A \vdash \forall x.B}\,{\forall_{\mathsf{R}}}}{\forall x.(A \supset B) \vdash A \supset \forall x.B}\,{\supset_{\mathsf{R}}}}{\vdash \forall x.(A \supset B) \supset (A \supset \forall x.B)}\,{\supset_{\mathsf{R}}}$$

where $x$ is not free in $A$.

6a)

$$\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{A \vdash A, B}\,{<_{\mathsf{R}}}}{\vdash A, A \supset B}\,{\supset_{\mathsf{R}}}}{\vdash A, \exists x.(A \supset B)}\,{\exists_{\mathsf{R}}} \quad \dfrac{\dfrac{\dfrac{B \vdash B}{A, B \vdash B}\,{<_{\mathsf{L}}}}{B \vdash A \supset B}\,{\supset_{\mathsf{R}}}}{B \vdash \exists x.(A \supset B)}\,{\exists_{\mathsf{R}}}}{\exists x.B \vdash \exists x.(A \supset B)}\,{\exists_{\mathsf{L}}}}{A \supset \exists x.B \vdash \exists x.(A \supset B)}\,{\supset_{\mathsf{L}}}}{\vdash (A \supset \exists x.B) \supset \exists x.(A \supset B)}\,{\supset_{\mathsf{R}}}$$

6b)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{A \vdash A, B}\,{<_R} \quad \dfrac{B \vdash B}{B, A \vdash B}\,{<_L}}{A \supset B, A \vdash B}\,{\supset_L}}{A \supset B, A \vdash \exists x.B}\,{\exists_R}}{\exists x.(A \supset B), A \vdash \exists x.B}\,{\exists_L}}{\exists x.(A \supset B) \vdash A \supset \exists x.B}\,{\supset_R}}{\vdash \exists x.(A \supset B) \supset (A \supset \exists x.B)}\,{\supset_R}$$

where $x$ is not free in $A$.

7a)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{A \vdash A, B}\,{<_R}}{\vdash A, A \supset B}\,{\supset_R}}{\vdash A, \exists x.(A \supset B)}\,{\exists_R}}{\vdash \forall x.A, \exists x.(A \supset B)}\,{\forall_R} \quad \dfrac{\dfrac{\dfrac{B \vdash B}{A, B \vdash B}\,{<_L}}{B \vdash A \supset B}\,{\supset_R}}{B \vdash \exists x.(A \supset B)}\,{\exists_R}}{\forall x.A \supset B \vdash \exists x.(A \supset B)}\,{\supset_L}}{\vdash (\forall x.A \supset B) \supset \exists x.(A \supset B)}\,{\supset_R}$$

7b)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{A \vdash A}{A \vdash A, B}\,{<_R} \quad \dfrac{B \vdash B}{B, A \vdash B}\,{<_L}}{A \supset B, A \vdash B}\,{\supset_L}}{A \supset B, \forall x.A \vdash B}\,{\forall_L}}{A \supset B \vdash \forall x.A \supset B}\,{\supset_R}}{\exists x.(A \supset B) \vdash \forall x.A \supset B}\,{\exists_L}}{\vdash \exists x.(A \supset B) \supset (\forall x.A \supset B)}\,{\supset_R}$$

where $x$ is not free in $B$.

8)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{p(z) \vdash p(z)}{p(z) \vdash p(z), p(y)}\,{<_R}}{\vdash p(z), p(z) \supset p(y)}\,{\supset_R}}{\vdash p(z), \forall y.(p(z) \supset p(y))}\,{\forall_R}}{\vdash p(z), \exists x.\forall y.(p(x) \supset p(y))}\,{\exists_R}}{\vdash \forall z.p(z), \exists x.\forall y.(p(x) \supset p(y))}\,{\forall_R} \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{p(y) \vdash p(y)}{p(x), p(y) \vdash p(y)}\,{<_L}}{p(y) \vdash p(x) \supset p(y)}\,{\supset_R}}{\forall z.p(z) \vdash p(x) \supset p(y)}\,{\forall_L}}{\forall z.p(z) \vdash \forall y.(p(x) \supset p(y))}\,{\forall_R}}{\forall z.p(z) \vdash \exists x.\forall y.(p(x) \supset p(y))}\,{\exists_R}}{\vdash \forall z.p(z), \exists x.\forall y.(p(x) \supset p(y))}\,{\bowtie} \quad \dfrac{\dfrac{\dfrac{\forall z.p(z) \vdash \forall z.p(z)}{\vdash \forall z.p(z), \neg\forall z.p(z)}\,{\neg_R} \quad \dfrac{\dfrac{\neg\forall z.p(z) \vdash \exists x.\forall y.(p(x) \supset p(y))}{\vdash \exists x.\forall y.(p(x) \supset p(y)), \neg\forall z.p(z)}\,{\neg_L}}{\vdash \forall z.p(z), \exists x.\forall y.(p(x) \supset p(y))}\,{\bowtie}}{\vdash \exists x.\forall y.(p(x) \supset p(y)), \exists x.\forall y.(p(x) \supset p(y))}\,{>_R}}{\vdash \exists x.\forall y.(p(x) \supset p(y))}$$

where cuts are, of course, redundant (they have been introduced just for pedagogical reasons). Try eliminating them according to the cut-elimination algorithm you find in [1]: you should obtain a much simpler proof, probably less intuitive from the semantical viewpoint.

## References

[1] J. Gallier. Constructive logics. Part I: A tutorial on proof systems and typed $\lambda$-calculi. *Theoretical Computer Science*, 110:249–339, 1993.

[2] J.-Y. Girard, Y. Lafont, and P. Taylor. *Proofs and Types*, volume 7 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1990.

[3] D. Miller, G. Nadathur, F. Pfenning, and A. Scedrov. Uniform proofs as a foundation for logic programming. *Annals of Pure and Applied Logic*, 51:125–157, 1991.

Typeset with Xy-pic

# Index of Symbols