2.2 Classical Logic

The inference rules so far only model *intuitionistic logic*, and some classically true propositions such as $A \vee \neg A$ (for an arbitrary A) are not derivable, as we will see in Section **??**. There are three commonly used ways one can construct a system of *classical natural deduction* by adding one additional rule of inference. $\bot_C$ is called *Proof by Contradiction* or *Rule of Indirect Proof*, $\neg\neg_C$ is the *Double Negation Rule*, and XM is referred to as *Excluded Middle*.

$$\dfrac{\begin{array}{c}\dfrac{}{\neg A}\,u\\ \vdots\\ \bot\end{array}}{A}\,\bot_C^u \qquad \dfrac{\neg\neg A}{A}\,\neg\neg_C \qquad \dfrac{}{A \vee \neg A}\,\mathrm{XM}$$

The rule for classical logic (whichever one chooses to adopt) breaks the pattern of introduction and elimination rules. One can still formulate some reductions for classical inferences, but natural deduction is at heart an intuitionistic calculus. The symmetries of classical logic are much better exhibited in sequent formulations of the logic. In Exercise 2.3 we explore the three ways of extending the intuitionistic proof system and show that they are equivalent.

Another way to obtain a natural deduction system for classical logic is to allow multiple conclusions (see, for example, Parigot [Par92]).

2.3 Localizing Hypotheses

In the formulation of natural deduction from Section 2.1 correct use of hypotheses and parameters is a global property of a derivation. We can localize it by annotating each judgment in a derivation by the available parameters and hypotheses. We give here a formulation of natural deduction for intuitionistic logic with localized hypotheses, but not parameters. For this we need a notation for hypotheses which we call a *context*.

$$\textit{Contexts} \quad \Gamma \quad ::= \quad \cdot \mid \Gamma, u{:}A$$

Here, "$\cdot$" represents the empty context, and $\Gamma, u{:}A$ adds hypothesis $\vdash A$ labelled u to Γ. We assume that each label u occurs at most once in a context in order to avoid ambiguities. The main judgment can then be written as $\Gamma \vdash A$, where

$$\cdot, u_1{:}A_1, \ldots, u_n{:}A_n \vdash A$$

stands for

$$\begin{array}{c}\dfrac{}{\vdash A_1}\,u_1 \quad \ldots \quad \dfrac{}{\vdash A_n}\,u_n\\ \vdots\\ \vdash A\end{array}$$

in the notation of Section 2.1.

We use a few important abbreviations in order to make this notation less cumbersome. First of all, we may omit the leading "$\cdot$" and write, for example, $u_1{:}A_1, u_2{:}A_2$ instead of $\cdot, u_1{:}A_1, u_2{:}A_2$. Secondly, we denote concatenation of contexts by overloading the comma operator as follows.

$$\begin{array}{rcl} \Gamma, \cdot & = & \Gamma \\ \Gamma, (\Gamma', u{:}A) & = & (\Gamma, \Gamma'), u{:}A \end{array}$$

With these additional definitions, the localized version of our rules are as follows.

Introduction Rules

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\wedge\mathrm{I}$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\vee\mathrm{I_L} \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\vee\mathrm{I_R}$$

$$\frac{\Gamma, u{:}A \vdash B}{\Gamma \vdash A \supset B}\supset\mathrm{I}^u$$

$$\frac{\Gamma, u{:}A \vdash p}{\Gamma \vdash \neg A}\neg\mathrm{I}^{p,u}$$

$$\frac{}{\Gamma \vdash \top}\top\mathrm{I}$$

no $\perp$ *introduction*

$$\frac{\Gamma \vdash [a/x]A}{\Gamma \vdash \forall x.\ A}\forall\mathrm{I}^a$$

$$\frac{\Gamma \vdash [t/x]A}{\Gamma \vdash \exists x.\ A}\exists\mathrm{I}$$

Elimination Rules

$$\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\wedge\mathrm{E_L} \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\wedge\mathrm{E_R}$$

$$\frac{\Gamma \vdash A \vee B \qquad \Gamma, u{:}A \vdash C \qquad \Gamma, w{:}B \vdash C}{\Gamma \vdash C}\vee\mathrm{E}^{u,w}$$

$$\frac{\Gamma \vdash A \supset B \qquad \Gamma \vdash A}{\Gamma \vdash B}\supset\mathrm{E}$$

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash \neg A}{\Gamma \vdash C}\neg\mathrm{E}$$

no $\top$ *elimination*

$$\frac{\Gamma \vdash \perp}{\Gamma \vdash C}\perp\mathrm{E}$$

$$\frac{\Gamma \vdash \forall x.\ A}{\Gamma \vdash [t/x]A}\forall\mathrm{E}$$

$$\frac{\Gamma \vdash \exists x.\ A \qquad \Gamma, u{:}[a/x]A \vdash C}{\Gamma \vdash C}\exists\mathrm{E}^{a,u}$$

We also have a new rule for hypotheses which was an implicit property of the hypothetical judgments before.

$$\frac{}{\Gamma_1, u{:}A, \Gamma_2 \vdash A}u$$

Other general assumptions about hypotheses, namely that they may be used arbitrarily often in a derivation and that their order does not matter, are indirectly reflected in these rules. Note that if we erase the context Γ from the judgments throughout a derivation, we obtain a derivation in the original notation.

When we discussed local reductions in order to establish local soundness, we used the notation

$$\begin{array}{c} \dfrac{\mathcal{D}}{\vdash A}\,u \\ \mathcal{E} \\ \vdash C \end{array}$$

for the result of substituting the derivation $\mathcal{D}$ of $\vdash A$ for all uses of the hypothesis $\vdash A$ labelled u in $\mathcal{E}$. We would now like to reformulate the property with localized hypotheses. In order to prove that the (now explicit) hypotheses behave as expected, we use the principle of *structural induction* over derivations. Simply put, we prove a property for all derivations by showing that, whenever it holds for the premisses of an inference, it holds for the conclusion. Note that we have to show the property outright when the rule under consideration has no premisses. Such rules are the base cases for the induction.

Theorem 2.1 (Structural Properties of Hypotheses) *The following properties hold for intuitionistic natural deduction.*

1. *(Exchange) If* $\Gamma_1, u_1{:}A, \Gamma_2, u_2{:}B, \Gamma_2 \vdash C$ *then* $\Gamma_1, u_2{:}B, \Gamma_2, u_1{:}A, \Gamma_2 \vdash C$.
2. *(Weakening) If* $\Gamma_1, \Gamma_2 \vdash C$ *then* $\Gamma_1, u{:}A, \Gamma_2 \vdash C$.
3. *(Contraction) If* $\Gamma_1, u_1{:}A, \Gamma_2, u_2{:}A, \Gamma_2 \vdash C$ *then* $\Gamma_1, u{:}A, \Gamma_2, \Gamma_3 \vdash C$.
4. *(Substitution) If* $\Gamma_1, u{:}A, \Gamma_2 \vdash C$ *and* $\Gamma_1 \vdash A$ *then* $\Gamma_1, \Gamma_2 \vdash C$.

Proof: The proof is in each case by straightforward induction over the structure of the first given derivation.

In the case of exchange, we appeal to the inductive assumption on the derivations of the premisses and construct a new derivation with the same inference rule. Algorithmically, this means that we exchange the hypotheses labelled u_1 and u_2 in every judgment in the derivation.

In the case of weakening and contraction, we proceed similarly, either adding the new hypothesis $u{:}A$ to every judgment in the derivation (for weakening), or replacing uses of u_1 and u_2 by u (for contraction).

For substitution, we apply the inductive assumption to the premisses of the given derivation $\mathcal{D}$ until we reach hypotheses. If the hypothesis is different from u we can simply erase $u{:}A$ (which is unused) to obtain the desired derivation. If the hypothesis is $u{:}A$ the derivation looks like

$$\mathcal{D} = \dfrac{}{\Gamma_1, u{:}A, \Gamma_2 \vdash A}\,u$$

so $C = A$ in this case. We are also given a derivation $\mathcal{E}$ of $\Gamma_1 \vdash A$ and have to construct a derivation $\mathcal{F}$ of $\Gamma_1, \Gamma_2 \vdash A$. But we can just repeatedly apply weakening to $\mathcal{E}$ to obtain $\mathcal{F}$. Algorithmically, this means that, as expected, we substitute the derivation $\mathcal{E}$ (possibly weakened) for uses of the hypotheses $u{:}A$ in $\mathcal{D}$. Note that in our original notation, this weakening has no impact, since unused hypotheses are not apparent in a derivation. □

It is also possible to localize the derivations themselves, using *proof terms*. As we will see in Section 2.4, these proof terms form a λ-calculus closely related to functional programming. When parameters, hypotheses, and proof terms are all localized our main judgment becomes decidable. In the terminology of Martin-Löf [ML94], the main judgment is then *analytic* rather than *synthetic*. We no longer need to go outside the judgment itself in order to collect evidence for it: An analytic judgment encapsulates its own evidence.

2.4 Proof Terms

The basic judgment of the system of natural deduction is the derivability of a formula A, written as $\vdash A$. It has been noted by Howard [How69] that there is a strong correspondence between (intuitionistic) derivations and λ-terms. The formulas A then act as types classifying λ-terms. In the propositional case, this correspondence is an isomorphism: formulas are isomorphic to types and derivations are isomorphic to simply-typed λ-terms. These isomorphisms are often called the *propositions-as-types* and *proofs-as-programs* paradigms.

If we stopped at this observation, we would have obtained only a fresh interpretation of familiar deductive systems, but we would not be any closer to the goal of providing a language for reasoning about properties of programs. However, the correspondences can be extended to first-order and higher-order logics. Interpreting first-order (or higher-order) formulas as types yields a significant increase in expressive power of the type system. However, maintaining an isomorphism during the generalization to first-order logic is somewhat unnatural and cumbersome. One might expect that a proof contains more information than the corresponding program. Thus the literature often talks about *extracting programs from proofs* or *contracting proofs to programs*. We do not discuss program extraction further in these notes.

We now introduce a notation for derivations to be carried along in deductions. For example, if M represents a proof of A and N represents a proof of B, then the pair $\langle M, N\rangle$ can be seen as a representation of the proof of $A \wedge B$ by $\wedge$-introduction. We write $\Gamma \vdash M : A$ to express the judgment *M is a proof term for A under hypotheses Γ*. We also repeat the local reductions and expansions from the previous section in the new notation. For local expansion we state the proposition whose truth must established by the proof term on the left-hand side. This expresses restrictions on the application of the expansion rules.

Conjunction. The proof term for a conjunction is simply the pair of proofs of the premisses.

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \wedge B}\wedge\mathrm{I}$$

$$\frac{\Gamma \vdash M : A \wedge B}{\Gamma \vdash \mathrm{fst}\, M : A}\wedge\mathrm{E_L} \qquad \frac{\Gamma \vdash M : A \wedge B}{\Gamma \vdash \mathrm{snd}\, M : B}\wedge\mathrm{E_R}$$

The local reductions now lead to two obvious local reductions of the proof terms. The local expansion is similiarly translated.

$$\begin{array}{rcl} \mathrm{fst}\,\langle M, N\rangle & \longrightarrow_R & M \\ \mathrm{snd}\,\langle M, N\rangle & \longrightarrow_R & N \\ M : A \wedge B & \longrightarrow_E & \langle \mathrm{fst}\, M, \mathrm{snd}\, M\rangle \end{array}$$

Implication. The proof of an implication $A \supset B$ will be represented by a function which maps proofs of A to proofs of B. The introduction rule explicitly forms such a function by λ-abstraction and the elimination rule applies the function to an argument.

$$\frac{\Gamma, u{:}A \vdash M : B}{\Gamma \vdash (\lambda u{:}A.\ M) : A \supset B}\supset\mathrm{I}^u \qquad \frac{\Gamma \vdash M : A \supset B \qquad \Gamma \vdash N : A}{\Gamma \vdash M\, N : B}\supset\mathrm{E}$$

The binding of the variable u in the conclusion of $\supset$I correctly models the intuition that the hypothesis is discharged and not available outside deduction of the premiss. The abstraction is labelled with the proposition A so that we can later show that the proof term uniquely determines a natural deduction. If A were not given then, for example, $\lambda u.\ u$ would be ambigous and serve as a proof term for $A \supset A$ for any formula A. The local reduction rule is β-reduction; the local expansion is η-expansion.

$$\begin{array}{rcl} (\lambda u{:}A.\ M)\, N & \longrightarrow_R & [N/u]M \\ M : A \supset B & \longrightarrow_E & \lambda u{:}A.\ M\, u \end{array}$$

In the reduction rule, bound variables in M that are free in N must be renamed in order to avoid variable capture. In the expansion rule u must be new—it may not already occur in M.

Disjunction. The proof term for disjunction introduction is the proof of the premiss together with an indication whether it was inferred by introduction on the left or on the right. We also annotate the proof term with the formula which did not occur in the premiss so that a proof terms always proves exactly one proposition.

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash \mathrm{inl}^B\, M : A \vee B}\vee\mathrm{I_L} \qquad \frac{\Gamma \vdash N : B}{\Gamma \vdash \mathrm{inr}^A\, N : A \vee B}\vee\mathrm{I_R}$$

The elimination rule corresponds to a case construction.

$$\frac{\Gamma \vdash M : A \vee B \qquad \Gamma, u{:}A \vdash N_1 : C \qquad \Gamma, w{:}B \vdash N_2 : C}{\Gamma \vdash (\,\mathbf{case}\ M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow N_1 \mid \mathrm{inr}\, w \Rightarrow N_2) : C}\vee\mathrm{E}^{u,w}$$

Since the variables u and w label assumptions, the corresponding proof term variables are *bound* in N_1 and N_2, respectively. The two reduction rules now also look like rules of computation in a λ-calculus.

$$\begin{array}{lcl}\mathbf{case}\ \mathrm{inl}^B\, M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow N_1 \mid \mathrm{inr}\, w \Rightarrow N_2 & \longrightarrow_R & [M/u]N_1 \\ \mathbf{case}\ \mathrm{inr}^A\, M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow N_1 \mid \mathrm{inr}\, w \Rightarrow N_2 & \longrightarrow_R & [M/w]N_2\end{array}$$

$$M : A \vee B \quad \longrightarrow_E \quad \mathbf{case}\ M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow \mathrm{inl}^B\, u \mid \mathrm{inr}\, w \Rightarrow \mathrm{inr}^A\, w$$

The substitution of a deduction for a hypothesis is represented by the substitution of a proof term for a variable.

Negation. This is similar to implication. Since the premise of the rule is parametric in p the corresponding proof constructor must bind a propositional variable p, indicated by μ^p. Similar, the elimination construct must record the formula to maintain the property that every valid term proves exactly one proposition. This is indicated as a subscript C to the infix operator "$\cdot$".

$$\frac{\Gamma, u{:}A \vdash M : p}{\Gamma \vdash \mu^p u{:}A.\ M : \neg A}\neg\mathrm{I}^{p,u} \qquad\qquad \frac{\Gamma \vdash M : \neg A \qquad \Gamma \vdash N : A}{\Gamma \vdash M \cdot_C N : C}\neg\mathrm{E}$$

The reduction performs formula and proof term substitutions.

$$\begin{array}{rcl}(\mu^p u{:}A.\ M) \cdot_C N & \longrightarrow_R & [N/u][C/p]M \\ M : \neg A & \longrightarrow_E & \mu^p u{:}A.\ M \cdot_p u\end{array}$$

Truth. The proof term for $\top$I is written $\langle\,\rangle$.

$$\frac{}{\Gamma \vdash \langle\,\rangle : \top}\top\mathrm{I}$$

Of course, there is no reduction rule. The expansion rule reads

$$M : \top \quad \longrightarrow_E \quad \langle\,\rangle$$

Falsehood. Here we need to annotate the proof term abort with the formula being proved to avoid ambiguity.

$$\frac{\Gamma \vdash M : \bot}{\Gamma \vdash \mathrm{abort}^C\, M : C}\bot\mathrm{E}$$

Again, there is no reduction rule, only an expansion rule.

$$M : \bot \quad \longrightarrow_E \quad \mathrm{abort}^\bot\, M$$

In summary, we have

$$
\begin{array}{llcll}
\text{Terms} & M & ::= & u & \textit{Hypotheses} \\
 & & \mid & \langle M_1, M_2\rangle \mid \mathrm{fst}\, M \mid \mathrm{snd}\, M & \textit{Conjunction} \\
 & & \mid & \lambda u{:}A.\ M \mid M_1\, M_2 & \textit{Implication} \\
 & & \mid & \mathrm{inl}^A\, M \mid \mathrm{inr}^A\, M & \textit{Disjunction} \\
 & & \mid & (\,\mathbf{case}\ M\ \mathbf{of}\ \mathrm{inl}\, u_1 \Rightarrow M_1 \mid \mathrm{inr}\, u_2 \Rightarrow M_2) & \\
 & & \mid & \mu^p u{:}A.\ M \mid M_1 \cdot_A M_2 & \textit{Negation} \\
 & & \mid & \langle\,\rangle & \textit{Truth} \\
 & & \mid & \mathrm{abort}^A\, M & \textit{Falsehood}
\end{array}
$$

and the reduction rules

$$
\begin{array}{rcc}
\mathrm{fst}\,\langle M, N\rangle & \longrightarrow_R & M \\
\mathrm{snd}\,\langle M, N\rangle & \longrightarrow_R & N \\
(\lambda u{:}A.\ M)\, N & \longrightarrow_R & [N/u]M \\
\mathbf{case}\ \mathrm{inl}^B\, M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow N_1 \mid \mathrm{inr}\, w \Rightarrow N_2 & \longrightarrow_R & [M/u]N_1 \\
\mathbf{case}\ \mathrm{inr}^A\, M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow N_1 \mid \mathrm{inr}\, w \Rightarrow N_2 & \longrightarrow_R & [M/w]N_2 \\
(\mu^p u{:}A.\ M) \cdot_C N & \longrightarrow_R & [N/u][C/p]M \\
\textit{no rule for truth} & & \\
\textit{no rule for falsehood} & &
\end{array}
$$

The expansion rules are given below.

$$
\begin{array}{rcl}
M : A \wedge B & \longrightarrow_E & \langle \mathrm{fst}\, M, \mathrm{snd}\, M\rangle \\
M : A \supset B & \longrightarrow_E & \lambda u{:}A.\ M\, u \\
M : A \vee B & \longrightarrow_E & \mathbf{case}\ M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow \mathrm{inl}^B\, u \mid \mathrm{inr}\, w \Rightarrow \mathrm{inr}^A\, w \\
M : \neg A & \longrightarrow_E & \mu^p u{:}A.\ M \cdot_p u \\
M : \top & \longrightarrow_E & \langle\,\rangle \\
M : \bot & \longrightarrow_E & \mathrm{abort}^\bot\, M
\end{array}
$$

We can now see that the formulas act as types for proof terms. Shifting to the usual presentation of the typed λ-calculus we use τ and σ as symbols for types, and $\tau \times \sigma$ for the product type, $\tau \to \sigma$ for the function type, $\tau + \sigma$ for the disjoint sum type, 1 for the unit type and 0 for the empty or void type. Base types b remain unspecified, just as the basic propositions of the propositional calculus remain unspecified. Types and propositions then correspond to each other as indicated below.

$$
\begin{array}{rcl}
\text{Types}\ \tau & ::= & b \mid \tau_1 \times \tau_2 \mid \tau_1 \to \tau_2 \mid \tau_1 + \tau_2 \mid 1 \mid 0 \\
\text{Propositions}\ A & ::= & p \mid A_1 \wedge A_2 \mid A_1 \supset A_2 \mid A_1 \vee A_2 \mid \top \mid \bot
\end{array}
$$

We omit here the negation type which is typically not used in functional programming and thus does not have a well-known counterpart. We can think of $\neg A$ as corresponding to $\tau \to 0$, where τ corresponds to A (see Exercise **??**). We now summarize and restate the rules above, using the notation of types instead of propositions (omitting only the case for negation). Note that contexts Γ now declare variables with their types, rather than hypothesis labels with their proposition.

$\Gamma \triangleright M : \tau$ *Term M has type τ in context Γ*

$$\frac{\Gamma \triangleright M : \tau \qquad \Gamma \triangleright N : \sigma}{\Gamma \triangleright \langle M, N \rangle : \tau \times \sigma}\,\mathsf{pair}$$

$$\frac{\Gamma \triangleright M : \tau \times \sigma}{\Gamma \triangleright \mathrm{fst}\, M : \tau}\,\mathsf{fst} \qquad \frac{\Gamma \triangleright M : \tau \times \sigma}{\Gamma \triangleright \mathrm{snd}\, M : \sigma}\,\mathsf{snd}$$

$$\frac{\Gamma, u{:}\tau \triangleright M : \sigma}{\Gamma \triangleright (\lambda u{:}\tau.\ M) : \tau \to \sigma}\,\mathsf{lam} \qquad \frac{u : \tau \text{ in } \Gamma}{\Gamma \triangleright u : \tau}\,\mathsf{var}$$

$$\frac{\Gamma \triangleright M : \tau \to \sigma \qquad \Gamma \triangleright N : \tau}{\Gamma \triangleright M\, N : \sigma}\,\mathsf{app}$$

$$\frac{\Gamma \triangleright M : \tau}{\Gamma \triangleright \mathrm{inl}^{\sigma}\, M : \tau + \sigma}\,\mathsf{inl} \qquad \frac{\Gamma \triangleright N : \sigma}{\Gamma \triangleright \mathrm{inr}^{\tau}\, N : \tau + \sigma}\,\mathsf{inr}$$

$$\frac{\Gamma \triangleright M : \tau + \sigma \qquad \Gamma, u{:}\tau \triangleright N_1 : \nu \qquad \Gamma, w{:}\sigma \triangleright N_2 : \nu}{\Gamma \triangleright (\,\mathbf{case}\ M\ \mathbf{of}\ \mathrm{inl}\, u \Rightarrow N_1 \mid \mathrm{inr}\, w \Rightarrow N_2) : \nu}\,\mathsf{case}$$

$$\frac{}{\Gamma \triangleright \langle\,\rangle : 1}\,\mathsf{unit} \qquad \frac{\Gamma \triangleright M : 0}{\Gamma \triangleright \mathrm{abort}^{\nu}\, M : \nu}\,\mathsf{abort}$$

2.5 Exercises

Exercise 2.1 Prove the following by natural deduction using only intuitionistic rules when possible. We use the convention that $\supset$, $\wedge$, and $\vee$ associate to the right, that is, $A \supset B \supset C$ stands for $A \supset (B \supset C)$. $A \equiv B$ is a syntactic abbreviation for $(A \supset B) \wedge (B \supset A)$. Also, we assume that $\wedge$ and $\vee$ bind more tightly than $\supset$, that is, $A \wedge B \supset C$ stands for $(A \wedge B) \supset C$. The scope of a quantifier extends as far to the right as consistent with the present parentheses. For example, $(\forall x.\ P(x) \supset C) \wedge \neg C$ would be disambiguated to $(\forall x.\ (P(x) \supset C)) \wedge (\neg C)$.

1. $\vdash A \supset B \supset A$.
2. $\vdash A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$.
3. (Peirce's Law). $\vdash ((A \supset B) \supset A) \supset A$.
4. $\vdash A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$.
5. $\vdash A \supset (A \wedge B) \vee (A \wedge \neg B)$.
6. $\vdash (A \supset \exists x.\ P(x)) \equiv \exists x.\ (A \supset P(x))$.
7. $\vdash ((\forall x.\ P(x)) \supset C) \equiv \exists x.\ (P(x) \supset C)$.

8. $\vdash \exists x.\, \forall y.\, (P(x) \supset P(y))$.

Exercise 2.2 We write $A \vdash B$ if B follows from hypothesis A and $A \dashv\vdash B$ for $A \vdash B$ and $B \vdash A$. Which of the following eight parametric judgments are derivable intuitionistically?

1. $(\exists x.\, A) \supset B \dashv\vdash \forall x.\, (A \supset B)$
2. $A \supset (\exists x.\, B) \dashv\vdash \exists x.\, (A \supset B)$
3. $(\forall x.\, A) \supset B \dashv\vdash \exists x.\, (A \supset B)$
4. $A \supset (\forall x.\, B) \dashv\vdash \forall x.\, (A \supset B)$

Provide natural deductions for the valid judgments. You may assume that the bound variable x does not occur in B (items 1 and 3) or A (items 2 and 4).

Exercise 2.3 Show that the three ways of extending the intuitionistic proof system are equivalent, that is, the same formulas are deducible in all three systems.

Exercise 2.4 Assume we had omitted disjunction and existential quantification and their introduction and elimination rules from the list of logical primitives. In the classical system, give a definition of disjunction and existential quantification (in terms of other logical constants) and show that the introduction and elimination rules now become *admissible rules of inference*. A rule of inference is *admissible* if any deduction using the rule can be transformed into one without using the rule.

Exercise 2.5 Assume we would like to design a system of natural deduction for a simple temporal logic. The main judgment is now "*A is true at time t*" written as

$$\vdash^t A.$$

1. Explain how to modify the given rules for natural deduction to this more general judgment and show the rules for implication and universal quantification.
2. Write out introduction and elimination rules for the temporal operator $\bigcirc A$ which should be true if A is true at the next point in time. Denote the "next time after t" by $t + 1$.
3. Show the local reductions and expansions which show the local soundness and completness of your rules.
4. Write out introduction and elimination rules for the temporal operator $\Box A$ which should be true if A is true at all times.
5. Show the local reductions and expansions.

Exercise 2.6 Design introduction and elimination rules for the connectives

1. $A \equiv B$, usually defined as $(A \supset B) \wedge (B \supset A)$,

2. $A \mid B$ (exclusive or), usually defined as $(A \wedge \neg B) \vee (\neg A \wedge B)$,

without recourse to other logical constants or operators. Also show the corresponding local reductions and expansions. For each of the following proposed connectives, write down appropriate introduction and eliminations rules and show the local reductions and expansion or indicate that no such rule may exist.

3. $A \overline{\wedge} B$ for $\neg(A \wedge B)$,

4. $A \overline{\vee} B$ for $\neg(A \vee B)$,

5. $A \overline{\supset} B$ for $\neg(A \supset B)$,

6. $+A$ for $\neg\neg A$,

7. $\exists^* x.\ A$ for $\neg \forall x.\ \neg A$,

8. $\forall^* x.\ A$ for $\neg \exists x.\ \neg A$,

9. $A \Rightarrow B \mid C$ for $(A \supset B) \wedge (\neg A \supset C)$.

Exercise 2.7 A given introduction rule does not necessarily uniquely determine matching elimination rules and vice versa. Explore if the following alternative rules are also sound and complete.

1. Replace the two elimination rules for conjunction by

$$\dfrac{\vdash A \wedge B \qquad \begin{array}{c} \overline{\vdash A}^{\,u} \quad \overline{\vdash B}^{\,w} \\ \vdots \\ \vdash C \end{array}}{\vdash C}\ \wedge\mathrm{E}^{u,w}$$

2. Add the following elimination rule for truth.

$$\dfrac{\vdash \top \qquad \vdash C}{\vdash C}\ \top\mathrm{E}$$

3. Add the following introduction rule for falsehood.

$$\dfrac{\vdash p}{\vdash \bot}\ \bot\mathrm{I}^p$$

Consider if any other of the standard connectives might permit alternative introduction or elimination rules which preserve derivability.

Exercise 2.8 For each of 14 following proposed entailments either write out a proof term for the corresponding implication or indicate that it is not derivable.

1. $A \supset (B \supset C) \dashv\vdash (A \wedge B) \supset C$
2. $A \supset (B \wedge C) \dashv\vdash (A \supset B) \wedge (A \supset C)$
3. $A \supset (B \vee C) \dashv\vdash (A \supset B) \vee (A \supset C)$
4. $(A \supset B) \supset C \dashv\vdash (A \vee C) \wedge (B \supset C)$
5. $(A \vee B) \supset C \dashv\vdash (A \supset C) \wedge (B \supset C)$
6. $A \wedge (B \vee C) \dashv\vdash (A \wedge B) \vee (A \wedge C)$
7. $A \vee (B \wedge C) \dashv\vdash (A \vee B) \wedge (A \vee C)$

Exercise 2.9 The de Morgan laws of classical logic allow negation to be distributed over other logical connectives. Investigate which directions of the de Morgan equivalences hold in intuitionistic logic and give proof terms for the valid entailments.

1. $\neg(A \wedge B) \dashv\vdash \neg A \vee \neg B$
2. $\neg(A \vee B) \dashv\vdash \neg A \wedge \neg B$
3. $\neg(A \supset B) \dashv\vdash A \wedge \neg B$
4. $\neg(\neg A) \dashv\vdash A$
5. $\neg\top \dashv\vdash \bot$
6. $\neg\bot \dashv\vdash \top$
7. $\neg\forall x.\, A \dashv\vdash \exists x.\, \neg A$
8. $\neg\exists x.\, A \dashv\vdash \forall x.\, \neg A$