CHAPTER 1

Logic

The main subject of Mathematical Logic is mathematical proof. In this introductory chapter we deal with the basics of formalizing such proofs. The system we pick for the representation of proofs is Gentzen's natural deduction, from [**5**]. Our reasons for this choice are twofold. First, as the name says this is a *natural* notion of formal proof, which means that the way proofs are represented corresponds very much to the way a careful mathematician writing out all details of an argument would go anyway. Second, formal proofs in natural deduction are closely related (via the so-called Curry-Howard correspondence) to terms in typed lambda calculus. This provides us not only with a compact notation for logical derivations (which otherwise tend to become somewhat unmanagable tree-like structures), but also opens up a route to applying the computational techniques which underpin lambda calculus.

Apart from classical logic we will also deal with more constructive logics: minimal and intuitionistic logic. This will reveal some interesting aspects of proofs, e.g. that it is possible und useful to distinguish beween existential proofs that actually construct witnessing objects, and others that don't. As an example, consider the following proposition.

There are irrational numbers a, b such that a^b is rational.

This can be proved as follows, by cases.

Case $\sqrt{2}^{\sqrt{2}}$ is rational. Choose $a = \sqrt{2}$ and $b = \sqrt{2}$. Then a, b are irrational and by assumption a^b is rational.

Case $\sqrt{2}^{\sqrt{2}}$ is irrational. Choose $a = \sqrt{2}^{\sqrt{2}}$ and $b = \sqrt{2}$. Then by assumption a, b are irrational and

$$a^b = \left(\sqrt{2}^{\sqrt{2}}\right)^{\sqrt{2}} = \left(\sqrt{2}\right)^2 = 2$$

is rational. □

As long as we have not decided whether $\sqrt{2}^{\sqrt{2}}$ is rational, we do not know which numbers a, b we must take. Hence we have an example of an existence proof which does not provide an instance.

An essential point for Mathematical Logic is to fix a formal language to be used. We take implication $\to$ and the universal quantifier $\forall$ as basic. Then the logic rules correspond to lambda calculus. The additional connectives $\perp$, $\exists$, $\vee$ and $\wedge$ are defined via axiom schemes. These axiom schemes will later be seen as special cases of introduction and elimination rules for inductive definitions.

1. Formal Languages

1.1. Terms and Formulas. Let a countable infinite set $\{ v_i \mid i \in \mathbb{N} \}$ of *variables* be given; they will be denoted by x, y, z. A first order language $\mathcal{L}$ then is determined by its *signature*, which is to mean the following.

- For every natural number $n \geq 0$ a (possible empty) set of n-ary *relation symbols* (also called *predicate symbols*). 0-ary relation symbols are called *propositional symbols*. $\perp$ (read "falsum") is required as a fixed propositional symbol. The language will *not*, unless stated otherwise, contain $=$ as a primitive.
- For every natural number $n \geq 0$ a (possible empty) set of n-ary *function symbols*. 0-ary function symbols are called *constants*.

We assume that all these sets of variables, relation and function symbols are disjoint.

For instance the language $\mathcal{L}_G$ of group theory is determined by the signature consisting of the following relation and function symbols: the group operation $\circ$ (a binary function symbol), the unit e (a constant), the inverse operation ${}^{-1}$ (a unary function symbol) and finally equality $=$ (a binary relation symbol).

$\mathcal{L}$-*terms* are inductively defined as follows.

- Every variable is an $\mathcal{L}$-term.
- Every constant of $\mathcal{L}$ is an $\mathcal{L}$-term.
- If $t_1, \dots, t_n$ are $\mathcal{L}$-terms and f is an n-ary function symbol of $\mathcal{L}$ with $n \geq 1$, then $f(t_1, \dots, t_n)$ is an $\mathcal{L}$-term.

From $\mathcal{L}$-terms one constructs $\mathcal{L}$-*prime formulas*, also called *atomic formulas* of $\mathcal{L}$: If $t_1, \dots, t_n$ are terms and R is an n-ary relation symbol of $\mathcal{L}$, then $R(t_1, \dots, t_n)$ is an $\mathcal{L}$-prime formula.

$\mathcal{L}$-*formulas* are inductively defined from $\mathcal{L}$-prime formulas by

- Every $\mathcal{L}$-prime formula is an $\mathcal{L}$-formula.
- If A and B are $\mathcal{L}$-formulas, then so are $(A \to B)$ ("if A, then B"), $(A \wedge B)$ ("A and B") and $(A \vee B)$ ("A or B").
- If A is an $\mathcal{L}$-formula and x is a variable, then $\forall x A$ ("for all x, A holds") and $\exists x A$ ("there is an x such that A") are $\mathcal{L}$-formulas.

Negation, classical disjunction, and the classical existential quantifier are defined by

$$\begin{aligned} \neg A &:= A \to \perp, \\ A \vee^{\mathsf{cl}} B &:= \neg A \to \neg B \to \perp, \\ \exists^{\mathsf{cl}} x A &:= \neg \forall x \neg A. \end{aligned}$$

Usually we fix a language $\mathcal{L}$, and speak of terms and formulas instead of $\mathcal{L}$-terms and $\mathcal{L}$-formulas. We use

r, s, t	for terms,
x, y, z	for variables,
c	for constants,
P, Q, R	for relation symbols,
f, g, h	for function symbols,
A, B, C, D	for formulas.

DEFINITION. The *depth* $\mathsf{dp}(A)$ of a formula A is the maximum length of a branch in its construction tree. In other words, we define recursively $\mathsf{dp}(P) = 0$ for atomic P, $\mathsf{dp}(A \circ B) = \max(\mathsf{dp}(A), \mathsf{dp}(B)) + 1$ for binary operators $\circ$, $\mathsf{dp}(\circ A) = \mathsf{dp}(A) + 1$ for unary operators $\circ$.

The *size* or *length* $|A|$ of a formula A is the number of occurrences of logical symbols and atomic formulas (parentheses not counted) in A: $|P| = 1$ for P atomic, $|A \circ B| = |A| + |B| + 1$ for binary operators $\circ$, $|\circ A| = |A| + 1$ for unary operators $\circ$.

One can show easily that $|A| + 1 \leq 2^{\mathsf{dp}(A)+1}$.

NOTATION (Saving on parentheses). In writing formulas we save on parentheses by assuming that $\forall, \exists, \neg$ bind more strongly than $\wedge, \vee$, and that in turn $\wedge, \vee$ bind more strongly than $\to, \leftrightarrow$ (where $A \leftrightarrow B$ abbreviates $(A \to B) \wedge (B \to A)$). Outermost parentheses are also usually dropped. Thus $A \wedge \neg B \to C$ is read as $((A \wedge (\neg B)) \to C)$. In the case of iterated implications we sometimes use the short notation

$$A_1 \to A_2 \to \dots A_{n-1} \to A_n \quad \text{for} \quad A_1 \to (A_2 \to \dots (A_{n-1} \to A_n) \dots).$$

To save parentheses in quantified formulas, we use a mild form of the *dot notation*: a dot immediately after $\forall x$ or $\exists x$ makes the scope of that quantifier as large as possible, given the parentheses around. So $\forall x.A \to B$ means $\forall x(A \to B)$, not $(\forall x A) \to B$.

We also save on parentheses by writing e.g. $Rxyz$, $Rt_0t_1t_2$ instead of $R(x, y, z)$, $R(t_0, t_1, t_2)$, where R is some predicate symbol. Similarly for a unary function symbol with a (typographically) simple argument, so fx for $f(x)$, etc. In this case no confusion will arise. But readability requires that we write in full $R(fx, gy, hz)$, instead of $Rfxgyhz$.

Binary function and relation symbols are usually written in *infix notation*, e.g. $x + y$ instead of $+(x, y)$, and $x < y$ instead of $<(x, y)$. We write $t \neq s$ for $\neg(t = s)$ and $t \not< s$ for $\neg(t < s)$.

1.2. Substitution, Free and Bound Variables. Expressions $\mathcal{E}, \mathcal{E}'$ which differ only in the names of bound variables will be regarded as identical. This is sometimes expressed by saying that $\mathcal{E}$ and $\mathcal{E}'$ are α-equivalent. In other words, we are only interested in expressions "modulo renaming of bound variables". There are methods of finding unique representatives for such expressions, for example the namefree terms of de Bruijn [**4**]. For the human reader such representations are less convenient, so we shall stick to the use of bound variables.

In the definition of "substitution of expression $\mathcal{E}'$ for variable x in expression $\mathcal{E}$", either one requires that *no* variable free in $\mathcal{E}'$ becomes bound by a variable-binding operator in $\mathcal{E}$, when the free occurrences of x are replaced by $\mathcal{E}'$ (also expressed by saying that there must be no "clashes of variables"), "*$\mathcal{E}'$ is free for x in $\mathcal{E}$*", or the substitution operation is taken to involve a systematic renaming operation for the bound variables, avoiding clashes. Having stated that we are only interested in expressions modulo renaming bound variables, we can without loss of generality assume that substitution is always possible.

Also, it is never a real restriction to assume that distinct quantifier occurrences are followed by distinct variables, and that the sets of bound and free variables of a formula are disjoint.

NOTATION. "FV" is used for the (set of) free variables of an expression; so $\mathsf{FV}(t)$ is the set of variables free in the term t, $\mathsf{FV}(A)$ the set of variables free in formula A etc.

$\mathcal{E}[x := t]$ denotes the result of substituting the term t for the variable x in the expression $\mathcal{E}$. Similarly, $\mathcal{E}[\vec{x} := \vec{t}]$ is the result of *simultaneously* substituting the terms $\vec{t} = t_1, \dots, t_n$ for the variables $\vec{x} = x_1, \dots, x_n$, respectively.

Locally we shall adopt the following convention. In an argument, once a formula has been introduced as $A(x)$, i.e., A with a designated variable x, we write $A(t)$ for $A[x := t]$, and similarly with more variables. $\square$

1.3. Subformulas. Unless stated otherwise, the notion of *subformula* we use will be that of a subformula in the sense of Gentzen.

DEFINITION. (Gentzen) subformulas of A are defined by

(a) A is a subformula of A;
(b) if $B \circ C$ is a subformula of A then so are B, C, for $\circ = \to, \wedge, \vee$;
(c) if $\forall x B$ or $\exists x B$ is a subformula of A, then so is $B[x := t]$, for all t free for x in B.

If we replace the third clause by:

(c)′ if $\forall x B$ or $\exists x B$ is a subformula of A then so is B,

we obtain the notion of *literal* subformula.

DEFINITION. The notions of *positive, negative, strictly positive* subformula are defined in a similar style:

(a) A is a positive and a stricly positive subformula of itself;
(b) if $B \wedge C$ or $B \vee C$ is a positive [negative, strictly positive] subformula of A, then so are B, C;
(c) if $\forall x B$ or $\exists x B$ is a positive [negative, strictly positive] subformula of A, then so is $B[x := t]$;
(d) if $B \to C$ is a positive [negative] subformula of A, then B is a negative [positive] subformula of A, and C is a positive [negative] subformula of A;
(e) if $B \to C$ is a strictly positive subformula of A, then so is C.

A strictly positive subformula of A is also called a *strictly positive part (s.p.p.)* of A. Note that the set of subformulas of A is the union of the positive and negative subformulas of A. *Literal* positive, negative, strictly positive subformulas may be defined in the obvious way by restricting the clause for quantifiers.

EXAMPLE. $(P \to Q) \to R \wedge \forall x R'(x)$ has as s.p.p.'s the whole formula, $R \wedge \forall x R'(x)$, R, $\forall x R'(x)$, $R'(t)$. The positive subformulas are the s.p.p.'s and in addition P; the negative subformulas are $P \to Q$, Q.

2. Natural Deduction

We introduce Gentzen's system of natural deduction. To allow a direct correspondence with the lambda calculus, we restrict the rules used to those for the logical connective $\to$ and the universal quantifier $\forall$. The rules come in pairs: we have an introduction and an elimination rule for each of these. The other logical connectives are introduced by means of axiom schemes: this is done for conjunction $\wedge$, disjunction $\vee$ and the existential quantifier $\exists$. The resulting system is called *minimal logic*; it has been introduced by Johannson in 1937 [**9**]. Notice that no negation is present.

If we then go on and require the *ex-falso-quodlibet* scheme for the nullary propositional symbol $\perp$ ("falsum"), we can embed *intuitionistic logic*. To obtain classical logic, we add as an axiom scheme the principle of *indirect proof*, also called *stability*. However, to obtain classical logic it suffices to restrict to the language based on $\to$, $\forall$, $\perp$ and $\wedge$; we can introduce classical disjunction $\vee^{\mathsf{cl}}$ and the classical existential quantifier $\exists^{\mathsf{cl}}$ via their (classical) definitions above. For these the usual introduction and elimination properties can then be derived.

2.1. Examples of Derivations. Let us start with some examples for natural proofs. Assume that a first order language $\mathcal{L}$ is given. For simplicity we only consider here proofs in pure logic, i.e. without assumptions (axioms) on the functions and relations used.

$$(A \to B \to C) \to (A \to B) \to A \to C \tag{1}$$

Assume $A \to B \to C$. To show: $(A \to B) \to A \to C$. So assume $A \to B$. To show: $A \to C$. So finally assume A. To show: C. We have A, by the last assumption. Hence also $B \to C$, by the first assumption, and B, using the next to last assumption. From $B \to C$ and B we obtain C, as required. □

$$(\forall x.A \to B) \to A \to \forall x B, \quad \text{if } x \notin \mathsf{FV}(A). \tag{2}$$

Assume $\forall x.A \to B$. To show: $A \to \forall x B$. So assume A. To show: $\forall x B$. Let x be arbitrary; note that we have not made any assumptions on x. To show: B. We have $A \to B$, by the first assumption. Hence also B, by the second assumption. □

$$(A \to \forall x B) \to \forall x.A \to B, \quad \text{if } x \notin \mathsf{FV}(A). \tag{3}$$

Assume $A \to \forall x B$. To show: $\forall x.A \to B$. Let x be arbitrary; note that we have not made any assumptions on x. To show: $A \to B$. So assume A. To show: B. We have $\forall x B$, by the first and second assumption. Hence also B. □

A characteristic feature of these proofs is that assumptions are introduced and eliminated again. At any point in time during the proof the free or "open" assumptions are known, but as the proof progresses, free assumptions may become cancelled or "closed" because of the implies-introduction rule.

We now reserve the word *proof* for the informal level; a formal representation of a proof will be called a *derivation*.

An intuitive way to communicate derivations is to view them as labelled trees. The labels of the inner nodes are the formulas derived at those points, and the labels of the leaves are formulas or terms. The labels of the nodes immediately above a node ν are the *premises* of the rule application, the formula at node ν is its *conclusion*. At the root of the tree we have the conclusion of the whole derivation. In natural deduction systems one works with *assumptions* affixed to some leaves of the tree; they can be *open* or else *closed*.

Any of these assumptions carries a *marker*. As markers we use *assumption variables* $\square_0, \square_1, \ldots$, denoted by $u, v, w, u_0, u_1, \ldots$. The (previous) variables will now often be called *object variables*, to distinguish them from assumption variables. If at a later stage (i.e. at a node below an assumption) the dependency on this assumption is removed, we record this by writing down the assumption variable. Since the same assumption can be used many times (this was the case in example (1)), the assumption marked with u (and communicated by $u\colon A$) may appear many times. However, we insist that distinct assumption formulas must have distinct markers.

An inner node of the tree is understood as the result of passing form premises to a *conclusion*, as described by a given *rule*. The label of the node then contains in addition to the conclusion also the name of the rule. In some cases the rule binds or closes an assumption variable u (and hence removes the dependency of all assumptions $u\colon A$ marked with that u). An application of the $\forall$-introduction rule similarly binds an object variable x (and hence removes the dependency on x). In both cases the bound assumption or object variable is added to the label of the node.

2.2. Introduction and Elimination Rules for $\to$ and $\forall$. We now formulate the rules of natural deduction. First we have an assumption rule, that allows an arbitrary formula A to be put down, together with a marker u:

$$u\colon A \quad \text{Assumption}$$

The other rules of natural deduction split into introduction rules (I-rules for short) and elimination rules (E-rules) for the logical connectives $\to$ and $\forall$. For implication $\to$ there is an introduction rule $\to^+ u$ and an elimination rule $\to^-$, also called *modus ponens*. The left premise $A \to B$ in $\to^-$ is called *major premise* (or *main* premise), and the right premise A *minor premise* (or *side* premise). Note that with an application of the $\to^+ u$-rule *all* assumptions above it marked with $u\colon A$ are cancelled.

$$\dfrac{\begin{array}{c}[u\colon A]\\ \mid M\\ B\end{array}}{A \to B}\,\to^+ u \qquad \dfrac{\begin{array}{c}\mid M\\ A \to B\end{array} \qquad \begin{array}{c}\mid N\\ A\end{array}}{B}\,\to^-$$

For the universal quantifier $\forall$ there is an introduction rule $\forall^+ x$ and an elimination rule $\forall^-$, whose right premise is the term r to be substituted. The rule $\forall^+ x$ is subject to the following *(Eigen-) variable condition*: The derivation M of the premise A should not contain any open assumption with x as a free variable.

$$\dfrac{\begin{array}{c}\mid M\\ A\end{array}}{\forall x A}\,\forall^+ x \qquad \dfrac{\begin{array}{c}\mid M\\ \forall x A\end{array} \qquad r}{A[x := r]}\,\forall^-$$

We now give derivations for the example formulas (1) – (3). Since in many cases the rule used is determined by the formula on the node, we suppress in such cases the name of the rule,

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{u\colon A \to B \to C \qquad w\colon A}{B \to C} \qquad \dfrac{v\colon A \to B \qquad w\colon A}{B}}{C}}{A \to C}\,\to^+ w}{(A \to B) \to A \to C}\,\to^+ v}{(A \to B \to C) \to (A \to B) \to A \to C}\,\to^+ u \tag{1}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{u\colon \forall x.A \to B \qquad x}{A \to B} \qquad v\colon A}{B}}{\forall x B}\,\forall^+ x}{A \to \forall x B}\,\to^+ v}{(\forall x.A \to B) \to A \to \forall x B}\,\to^+ u \tag{2}$$

Note here that the variable condition is satisfied: x is not free in A (and also not free in $\forall x.A \to B$).

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{u\colon A \to \forall x B \qquad v\colon A}{\forall x B} \qquad x}{B}}{A \to B}\,\to^+ v}{\forall x.A \to B}\,\forall^+ x}{(A \to \forall x B) \to \forall x.A \to B}\,\to^+ u \tag{3}$$

Here too the variable condition is satisfied: x is not free in A.

2.3. Axiom Schemes for Disjunction, Conjunction, Existence and Falsity. We follow the usual practice of considering all free variables in an axiom as universally quantified outside.

Disjunction. The introduction axioms are

$$\vee_0^+\colon A \to A \vee B$$
$$\vee_1^+\colon B \to A \vee B$$

and the elimination axiom is

$$\vee^-\colon (A \to C) \to (B \to C) \to A \vee B \to C.$$

Conjunction. The introduction axiom is

$$\wedge^+\colon A \to B \to A \wedge B$$

and the elimination axiom is

$$\wedge^-\colon (A \to B \to C) \to A \wedge B \to C.$$

Existential Quantifier. The introduction axiom is

$$\exists^+\colon A \to \exists x A$$

and the elimination axiom is

$$\exists^-\colon (\forall x.A \to B) \to \exists x A \to B \quad (x \text{ not free in } B).$$

Falsity. This example is somewhat extreme, since there is no introduction axiom; the elimination axiom is

$$\bot^- : \bot \to A.$$

In the literature this axiom is frequently called "ex-falso-quodlibet", written Efq. It clearly is derivable from its instances $\bot \to R\vec{x}$, for every relation symbol R.

Equality. The introduction axiom is

$$\mathsf{Eq}^+ : \mathsf{Eq}(x, x)$$

and the elimination axiom is

$$\mathsf{Eq}^- : \forall x R(x, x) \to \mathsf{Eq}(x, y) \to R(x, y).$$

It is an easy exercise to show that the usual equality axioms can be derived.

All these axioms can be seen as special cases of a general scheme, that of an *inductively defined predicate*, which is defined by some introduction rules and one elimination rule. We will study this kind of definition in full generality in Chapter 6. $\mathsf{Eq}(x, y)$ is a binary such predicate, $\bot$ is a nullary one, and $A \vee B$ another nullary one which however depends on the two parameter predicates A and B.

The desire to follow this general pattern is also the reason that we have chosen our rather strange $\wedge^-$-axiom, instead of the more obvious $A \wedge B \to A$ and $A \wedge B \to B$ (which clearly are equivalent).

2.4. Minimal, Intuitionistic and Classical Logic. We write $\vdash A$ and call A *derivable* (in *minimal logic*), if there is a derivation of A without free assumptions, from the axioms of 2.3 using the rules from 2.2, but *without using the ex-falso-quodlibet axiom, i.e., the elimination axiom $\bot^-$ for $\bot$.* A formula B is called derivable from assumptions $A_1, \dots, A_n$, if there is a derivation (without $\bot^-$) of B with free assumptions among $A_1, \dots, A_n$. Let Γ be a (finite or infinite) set of formulas. We write $\Gamma \vdash B$ if the formula B is derivable from finitely many assumptions $A_1, \dots, A_n \in \Gamma$.

Similarly we write $\vdash_i A$ and $\Gamma \vdash_i B$ if use of the ex-falso-quodlibet axiom is allowed; we then speak of derivability in *intuitionistic logic.*

For classical logic there is no need to use the full set of logical connectives: classical disjunction as well as the classical existential quantifier are defined, by $A \vee^{\mathsf{cl}} B := \neg A \to \neg B \to \bot$ and $\exists^{\mathsf{cl}} x A := \neg \forall x \neg A$. Moreover, when dealing with derivability we can even get rid of conjunction; this can be seen from the following lemma:

LEMMA (Elimination of $\wedge$). *For each formula A built with the connectives $\to, \wedge, \forall$ there are formulae $A_1, \dots, A_n$ without $\wedge$ such that $\vdash A \leftrightarrow \bigwedge_{i=1}^n A_i$.*

PROOF. Induction on A. **Case** $R\vec{t}$. Take $n = 1$ and $A_1 := R\vec{t}$. **Case** $A \wedge B$. By induction hypothesis, we have $A_1, \dots, A_n$ and $B_1, \dots, B_m$. Take $A_1, \dots, A_n, B_1, \dots, B_m$. **Case** $A \to B$. By induction hypothesis, we have $A_1, \dots, A_n$ and $B_1, \dots, B_m$. For the sake of notational simplicity assume $n = 2$ and $m = 3$. Then

$\vdash (A_1 \wedge A_2 \to B_1 \wedge B_2 \wedge B_3)$

$$\leftrightarrow (A_1 \to A_2 \to B_1) \wedge (A_1 \to A_2 \to B_2) \wedge (A_1 \to A_2 \to B_3).$$

Case $\forall x A$. By induction hypothesis for A, we have $A_1, \dots, A_n$. Take $\forall x A_1, \dots, \forall x A_n$, for

$$\vdash \forall x \bigwedge_{i=1}^{n} A_i \leftrightarrow \bigwedge_{i=1}^{n} \forall x A_i.$$

This concludes the proof. □

For the rest of this section, let us restrict to the language based on $\to$, $\forall$, $\bot$ and $\wedge$. We obtain *classical logic* by adding, for every relation symbol R distinct from $\bot$, the *principle of indirect proof* expressed as the so-called "stability axiom" (Stab_R):

$$\neg\neg R\vec{x} \to R\vec{x}.$$

Let

$$\mathsf{Stab} := \{\, \forall \vec{x}.\neg\neg R\vec{x} \to R\vec{x} \mid R \text{ relation symbol distinct from } \bot \,\}.$$

We call the formula A *classically derivable* and write $\vdash_c A$ if there is a derivation of A from stability assumptions Stab_R. Similarly we define classical derivability from Γ and write $\Gamma \vdash_c A$, i.e.

$$\Gamma \vdash_c A \quad :\Longleftrightarrow \quad \Gamma \cup \mathsf{Stab} \vdash A.$$

THEOREM (Stability, or Principle of Indirect Proof). *For every formula A (of our language based on $\to$, $\forall$, $\bot$ and $\wedge$),*

$$\vdash_c \neg\neg A \to A.$$

PROOF. Induction on A. For simplicity, in the derivation to be constructed we leave out applications of $\to^+$ at the end. **Case** $R\vec{t}$ with R distinct from $\bot$. Use Stab_R. **Case** $\bot$. Observe that $\neg\neg\bot \to \bot = ((\bot \to \bot) \to \bot) \to \bot$. The desired derivation is

$$\dfrac{v\colon (\bot \to \bot) \to \bot \qquad \dfrac{u\colon \bot}{\bot \to \bot}\,{\to^+ u}}{\bot}$$

Case $A \to B$. Use $\vdash (\neg\neg B \to B) \to \neg\neg(A \to B) \to A \to B$; a derivation is

$$\dfrac{u\colon \neg\neg B \to B \qquad \dfrac{\dfrac{v\colon \neg\neg(A \to B) \qquad \dfrac{\dfrac{u_1\colon \neg B \qquad \dfrac{u_2\colon A \to B \qquad w\colon A}{B}}{\bot}}{\neg(A \to B)}\,{\to^+ u_2}}{\bot}}{\neg\neg B}\,{\to^+ u_1}}{B}$$

Case $\forall x A$. Clearly it suffices to show $\vdash (\neg\neg A \to A) \to \neg\neg\forall x A \to A$; a derivation is

$$\dfrac{u\colon \neg\neg A \to A \qquad \dfrac{\dfrac{v\colon \neg\neg\forall x A \qquad \dfrac{\dfrac{u_1\colon \neg A \qquad \dfrac{u_2\colon \forall x A \qquad x}{A}}{\bot}}{\neg\forall x A}\,{\to^+ u_2}}{\bot}}{\neg\neg A}\,{\to^+ u_1}}{A}$$

The case $A \wedge B$ is left to the reader. $\square$

Notice that clearly $\vdash_c \bot \to A$, for stability is stronger:

$$\dfrac{\dfrac{\begin{array}{c} \mid M_{\mathsf{Stab}} \\ \neg\neg A \to A \end{array} \qquad \dfrac{u\colon \bot}{\neg\neg A}\to^+ v^{\neg A}}{A}}{\bot \to A}\to^+ u$$

where M_{Stab} is the (classical) derivation of stability.

Notice also that even for the $\to$, $\bot$-fragment the inclusion of minimal logic in intuitionistic logic, and of the latter in classical logic are proper. Examples are

$$\nvdash \bot \to P, \quad \text{but} \quad \vdash_i \bot \to P,$$
$$\nvdash_i ((P \to Q) \to P) \to P, \quad \text{but} \quad \vdash_c ((P \to Q) \to P) \to P.$$

Non-derivability can be proved by means of countermodels, using a semantic characterization of derivability; this will be done in Chapter 2. $\vdash_i \bot \to P$ is obvious, and the Peirce formula $((P \to Q) \to P) \to P$ can be derived in minimal logic from $\bot \to Q$ and $\neg\neg P \to P$, hence is derivable in classical logic.

2.5. Negative Translation. We embedd classical logic into minimal logic, via the so-called negative (or Gödel-Gentzen) translation.

A formula A is called *negative*, if every atomic formula of A distinct from $\bot$ occurs negated, and A does not contain $\vee$, $\exists$.

LEMMA. *For negative A, $\vdash \neg\neg A \to A$.*

PROOF. This follows from the proof of the stability theorem, using $\vdash \neg\neg\neg R\vec{t} \to \neg R\vec{t}$. $\square$

Since $\vee$, $\exists$ do not occur in formulas of classical logic, in the rest of this section we consider the language based on $\to$, $\forall$, $\bot$ and $\wedge$ only.

DEFINITION (Negative translation ^{g} of Gödel-Gentzen).

$$\begin{array}{ll} (R\vec{t})^g & := \neg\neg R\vec{t} \qquad (R \text{ distinct from } \bot) \\ \bot^g & := \bot, \\ (A \wedge B)^g & := A^g \wedge B^g, \\ (A \to B)^g & := A^g \to B^g, \\ (\forall x A)^g & := \forall x A^g. \end{array}$$

THEOREM. *For all formulas A,*

(a) $\vdash_c A \leftrightarrow A^g$,
(b) $\Gamma \vdash_c A$ *iff* $\Gamma^g \vdash A^g$, *where* $\Gamma^g := \{\, B^g \mid B \in \Gamma \,\}$.

PROOF. (a). The claim follows from the fact that $\vdash_c$ is compatible with equivalence. 2. $\Leftarrow$. Obvious $\Rightarrow$. By induction on the classical derivation. For a stability assumption $\neg\neg R\vec{t} \to R\vec{t}$ we have $(\neg\neg R\vec{t} \to R\vec{t})^g =$

$\neg\neg\neg\neg R\vec{t} \to \neg\neg R\vec{t}$, and this is easily derivable. *Case* $\to^+$. Assume

$$\dfrac{\begin{matrix}[u\colon A]\\ \mathcal{D}\\ B\end{matrix}}{A \to B}\to^+ u$$

Then we have by induction hypothesis

$$\begin{matrix}u\colon A^g\\ \mathcal{D}^g\\ B^g\end{matrix} \quad \text{hence} \quad \dfrac{\begin{matrix}[u\colon A^g]\\ \mathcal{D}^g\\ B^g\end{matrix}}{A^g \to B^g}\to^+ u$$

Case $\to^-$. Assume

$$\dfrac{\begin{matrix}\mathcal{D}_0\\ A \to B\end{matrix} \qquad \begin{matrix}\mathcal{D}_1\\ A\end{matrix}}{B}$$

Then we have by induction hypothesis

$$\begin{matrix}\mathcal{D}_0^g\\ A^g \to B^g\end{matrix} \qquad \begin{matrix}\mathcal{D}_1^g\\ A^g\end{matrix} \quad \text{hence} \quad \dfrac{\begin{matrix}\mathcal{D}_0^g\\ A^g \to B^g\end{matrix} \qquad \begin{matrix}\mathcal{D}_1^g\\ A^g\end{matrix}}{B^g}$$

The other cases are treated similarly. □

COROLLARY (Embedding of classical logic). *For negative A,*

$$\vdash_c A \iff \vdash A.$$

PROOF. By the theorem we have $\vdash_c A$ iff $\vdash A^g$. Since A is negative, every atom distinct from $\bot$ in A must occur negated, and hence in A^g it must appear in threefold negated form (as $\neg\neg\neg R\vec{t}$). The claim follows from $\vdash \neg\neg\neg R\vec{t} \leftrightarrow \neg R\vec{t}$. □

Since every formula is classically equivalent to a negative formula, we have achieved an embedding of classical logic into minimal logic.

Note that $\nvdash \neg\neg P \to P$ (as we shall show in Chapter 2). The corollary therefore does not hold for all formulas A.

3. Normalization

We show in this section that every derivation can be transformed by appropriate conversion steps into a normal form. A derivation in normal form does not make "detours", or more precisely, it cannot occur that an elimination rule immediately follows an introduction rule. Derivations in normal form have many pleasant properties.

Uniqueness of normal form will be shown by means of an application of Newman's lemma; we will also introduce and discuss the related notions of confluence, weak confluence and the Church-Rosser property.

We finally show that the requirement to give a normal derivation of a derivable formula can sometimes be unrealistic. Following Statman [**16**] and Orevkov [**13**] we give examples of formulas C_k which are easily derivable with non-normal derivations (of size linear in k), but which require a non-elementary (in k) size in any normal derivation.

This can be seen as a theoretical explanation of the essential role played by lemmas in mathematical arguments.

3.1. Conversion. A conversion eliminates a detour in a derivation, i.e., an elimination immediately following an introduction. We consider the following conversions:

$\to$*-conversion.*

$$\dfrac{\dfrac{\begin{matrix}[u\colon A]\\ \mid M\\ B\end{matrix}}{A\to B}\,\to^+u \qquad \begin{matrix}\mid N\\ A\end{matrix}}{B}\,\to^- \qquad\mapsto\qquad \begin{matrix}\mid N\\ A\\ \mid M\\ B\end{matrix}$$

$\forall$*-conversion.*

$$\dfrac{\dfrac{\begin{matrix}\mid M\\ A\end{matrix}}{\forall x A}\,\forall^+x \qquad r}{A[x:=r]}\,\forall^- \qquad\mapsto\qquad \begin{matrix}\mid M'\\ A[x:=r]\end{matrix}$$

3.2. Derivations as Terms. It will be convenient to represent derivations as terms, where the derived formula is viewed as the type of the term. This representation is known under the name *Curry-Howard correspondence*.

We give an inductive definition of derivation terms in the table below, where for clarity we have written the corresponding derivations to the left. For the universal quantifier $\forall$ there is an introduction rule $\forall^+x$ and an elimination rule $\forall^-$, whose right premise is the term r to be substituted. The rule $\forall^+x$ is subject to the following *(Eigen-) variable condition*: The derivation term M of the premise A should not contain any open assumption with x as a free variable.

3.3. Reduction, Normal Form. Although every derivation term carries a formula as its type, we shall usually leave these formulas implicit and write derivation terms without them.

Notice that every derivation term can be written uniquely in one of the forms

$$u\vec{M} \mid \lambda v M \mid (\lambda v M)N\vec{L},$$

where u is an assumption variable or assumption constant, v is an assumption variable or object variable, and M, N, L are derivation terms or object terms.

Here the final form is not normal: $(\lambda v M)N\vec{L}$ is called *β-redex* (for "reducible expression"). The *conversion rule* is

$$(\lambda v M)N \mapsto_\beta M[v:=N].$$

Notice that in a substitution $M[v:=N]$ with M a derivation term and v an object variable, one also needs to substitute in the formulas of M.

The *closure* of the conversion relation $\mapsto_\beta$ is defined by

- If $M \mapsto_\beta M'$, then $M \to M'$.
- If $M \to M'$, then also $MN \to M'N$, $NM \to NM'$, $\lambda v M \to \lambda v M'$ (*inner reductions*).

So $M \to N$ means that M *reduces in one step to* N, i.e., N is obtained from M by replacement of (an occurrence of) a redex M' of M by a conversum M'' of M', i.e. by a single conversion. The relation $\to^+$ ("*properly*

derivation	term
$u\colon A$	u^A
$\dfrac{\begin{matrix}[u\colon A]\\ \mid M\\ B\end{matrix}}{A\to B}\to^+u$	$(\lambda u^A M^B)^{A\to B}$
$\dfrac{\begin{matrix}\mid M\\ A\to B\end{matrix}\qquad\begin{matrix}\mid N\\ A\end{matrix}}{B}\to^-$	$(M^{A\to B}N^A)^B$
$\dfrac{\begin{matrix}\mid M\\ A\end{matrix}}{\forall xA}\forall^+x$ (with var.cond.)	$(\lambda x M^A)^{\forall xA}$ (with var.cond.)
$\dfrac{\begin{matrix}\mid M\\ \forall xA\end{matrix}\qquad r}{A[x:=r]}\forall^-$	$(M^{\forall xA}r)^{A[x:=r]}$

TABLE 1. Derivation terms for $\to$ and $\forall$

reduces to”) is the transitive closure of $\to$ and $\to^*$ (“*reduces to*”) is the reflexive and transitive closure of $\to$. The relation $\to^*$ is said to be the notion of reduction *generated* by $\mapsto$. $\leftarrow$, $\leftarrow^+$, $\leftarrow^*$ are the relations converse to $\to, \to^+, \to^*$, respectively.

A term M is *in normal form*, or M is *normal*, if M does not contain a redex. M *has a normal form* if there is a normal N such that $M \to^* N$.

A *reduction sequence* is a (finite or infinite) sequence $M_0 \to M_1 \to M_2 \dots$ such that $M_i \to M_{i+1}$, for all i.

Finite reduction sequences are partially ordered under the initial part relation; the collection of finite reduction sequences starting from a term M forms a tree, the *reduction tree* of M. The branches of this tree may be identified with the collection of all infinite and all terminating finite reduction sequences.

A term is *strongly normalizing* if its reduction tree is finite.

EXAMPLE.

$$
\begin{aligned}
&(\lambda x\lambda y\lambda z.xz(yz))(\lambda u\lambda v\, u)(\lambda u'\lambda v'\, u') \to\\
&(\lambda y\lambda z.(\lambda u\lambda v\, u)z(yz))(\lambda u'\lambda v'\, u') \qquad \to
\end{aligned}
$$

$$
\begin{aligned}
&(\lambda y \lambda z.(\lambda v\, z)(yz))(\lambda u' \lambda v'\, u') && \to \\
&(\lambda y \lambda z\, z)(\lambda u' \lambda v'\, u') && \to \lambda z\, z.
\end{aligned}
$$

LEMMA (Substitutivity of $\to$). (a) *If $M \to M'$, then $MN \to M'N$.*
(b) *If $N \to N'$, then $MN \to MN'$.*
(c) *If $M \to M'$, then $M[v := N] \to M'[v := N]$.*
(d) *If $N \to N'$, then $M[v := N] \to^* M[v := N']$.*

PROOF. (a) and (c) are proved by induction on $M \to M'$; (b) and (d) by induction on M. Notice that the reason for $\to^*$ in (d) is the fact that v may have many occurrences in M. □

3.4. Strong Normalization. We show that every term is strongly normalizing.

To this end, define by recursion on k a relation $\mathsf{sn}(M, k)$ between terms M and natural numbers k with the intention that k is an upper bound on the number of reduction steps up to normal form.

$$
\begin{aligned}
&\mathsf{sn}(M, 0) && :\Longleftrightarrow \quad M \text{ is in normal form,} \\
&\mathsf{sn}(M, k+1) && :\Longleftrightarrow \quad \mathsf{sn}(M', k) \text{ for all } M' \text{ such that } M \to M'.
\end{aligned}
$$

Clearly a term is strongly normalizable if there is a k such that $\mathsf{sn}(M, k)$. We first prove some closure properties of the relation sn.

LEMMA (Properties of sn). (a) *If $\mathsf{sn}(M, k)$, then $\mathsf{sn}(M, k+1)$.*
(b) *If $\mathsf{sn}(MN, k)$, then $\mathsf{sn}(M, k)$.*
(c) *If $\mathsf{sn}(M_i, k_i)$ for $i = 1 \dots n$, then $\mathsf{sn}(uM_1 \dots M_n, k_1 + \dots + k_n)$.*
(d) *If $\mathsf{sn}(M, k)$, then $\mathsf{sn}(\lambda v M, k)$.*
(e) *If $\mathsf{sn}(M[v := N]\vec{L}, k)$ and $\mathsf{sn}(N, l)$, then $\mathsf{sn}((\lambda v M)N\vec{L}, k + l + 1)$.*

PROOF. (a). Induction on k. Assume $\mathsf{sn}(M, k)$. We show $\mathsf{sn}(M, k+1)$. So let M' with $M \to M'$ be given; because of $\mathsf{sn}(M, k)$ we must have $k > 0$. We have to show $\mathsf{sn}(M', k)$. Because of $\mathsf{sn}(M, k)$ we have $\mathsf{sn}(M', k-1)$, hence by induction hypothesis $\mathsf{sn}(M', k)$.

(b). Induction on k. Assume $\mathsf{sn}(MN, k)$. We show $\mathsf{sn}(M, k)$. In case $k = 0$ the term MN is normal, hence also M is normal and therefore $\mathsf{sn}(M, 0)$. So let $k > 0$ and $M \to M'$; we have to show $\mathsf{sn}(M', k-1)$. From $M \to M'$ we have $MN \to M'N$. Because of $\mathsf{sn}(MN, k)$ we have by definition $\mathsf{sn}(M'N, k-1)$, hence $\mathsf{sn}(M', k-1)$ by induction hypothesis.

(c). Assume $\mathsf{sn}(M_i, k_i)$ for $i = 1 \dots n$. We show $\mathsf{sn}(uM_1 \dots M_n, k)$ with $k := k_1 + \dots + k_n$. Again we employ induction on k. In case $k = 0$ all M_i are normal, hence also $uM_1 \dots M_n$. So let $k > 0$ and $uM_1 \dots M_n \to M'$. Then $M' = uM_1 \dots M_i' \dots M_n$ with $M_i \to M_i'$; We have to show $\mathsf{sn}(uM_1 \dots M_i' \dots M_n, k-1)$. Because of $M_i \to M_i'$ and $\mathsf{sn}(M_i, k_i)$ we have $k_i > 0$ and $\mathsf{sn}(M_i', k_i - 1)$, hence $\mathsf{sn}(uM_1 \dots M_i' \dots M_n, k-1)$ by induction hypothesis.

(d). Assume $\mathsf{sn}(M, k)$. We have to show $\mathsf{sn}(\lambda v M, k)$. Use induction on k. In case $k = 0$ M is normal, hence $\lambda v M$ is normal, hence $\mathsf{sn}(\lambda v M, 0)$. So let $k > 0$ and $\lambda v M \to L$. Then L has the form $\lambda v M'$ with $M \to M'$. So $\mathsf{sn}(M', k-1)$ by definition, hence $\mathsf{sn}(\lambda v M', k)$ by induction hypothesis.

(e). Assume $\mathsf{sn}(M[v := N]\vec{L}, k)$ and $\mathsf{sn}(N, l)$. We need to show that $\mathsf{sn}((\lambda v M)N\vec{L}, k + l + 1)$. We use induction on $k + l$. In case $k + l = 0$ the term N and $M[v := N]\vec{L}$ are normal, hence also M and all L_i. Hence there is exactly one term K such that $(\lambda v M)N\vec{L} \to K$, namely $M[v := N]\vec{L}$, and this K is normal. So let $k + l > 0$ and $(\lambda v M)N\vec{L} \to K$. We have to show $\mathsf{sn}(K, k + l)$.

Case $K = M[v := N]\vec{L}$, i.e. we have a head conversion. From $\mathsf{sn}(M[v := N]\vec{L}, k)$ we obtain $\mathsf{sn}(M[v := N]\vec{L}, k + l)$ by (a).

Case $K = (\lambda v M')N\vec{L}$ with $M \to M'$. Then we have $M[v := N]\vec{L} \to M'[v := N]\vec{L}$. Now $\mathsf{sn}(M[v := N]\vec{L}, k)$ implies $k > 0$ and $\mathsf{sn}(M'[v := N]\vec{L}, k - 1)$. The induction hypothesis yields $\mathsf{sn}((\lambda v M')N\vec{L}, k - 1 + l + 1)$.

Case $K = (\lambda v M)N'\vec{L}$ with $N \to N'$. Now $\mathsf{sn}(N, l)$ implies $l > 0$ and $\mathsf{sn}(N', l - 1)$. The induction hypothesis yields $\mathsf{sn}((\lambda v M)N'\vec{L}, k + l - 1 + 1)$, since $\mathsf{sn}(M[v := N']\vec{L}, k)$ by (a),

Case $K = (\lambda v M)N\vec{L}'$ with $L_i \to L'_i$ for some i and $L_j = L'_j$ for $j \neq i$. Then we have $M[v := N]\vec{L} \to M[v := N]\vec{L}'$. Now $\mathsf{sn}(M[v := N]\vec{L}, k)$ implies $k > 0$ and $\mathsf{sn}(M[v := N]\vec{L}', k - 1)$. The induction hypothesis yields $\mathsf{sn}((\lambda v M)N\vec{L}', k - 1 + l + 1)$. □

The essential idea of the strong normalization proof is to view the last three closure properties of sn from the preceding lemma without the information on the bounds as an inductive definition of a new set SN:

$$\frac{\vec{M} \in \mathsf{SN}}{u\vec{M} \in \mathsf{SN}}\ (\mathsf{Var}) \qquad \frac{M \in \mathsf{SN}}{\lambda v M \in \mathsf{SN}}\ (\lambda) \qquad \frac{M[v := N]\vec{L} \in \mathsf{SN} \qquad N \in \mathsf{SN}}{(\lambda v M)N\vec{L} \in \mathsf{SN}}\ (\beta)$$

COROLLARY. *For every term $M \in \mathsf{SN}$ there is a $k \in \mathbb{N}$ such that $\mathsf{sn}(M, k)$. Hence every term $M \in \mathsf{SN}$ is strongly normalizable*

PROOF. By induction on $M \in \mathsf{SN}$, using the previous lemma. □

In what follows we shall show that *every* term is in SN and hence is strongly normalizable. Given the definition of SN we only have to show that SN is closed under application. In order to prove this we must prove simultaneously the closure of SN under substitution.

THEOREM (Properties of SN). *For all formulas A, derivation terms $M \in \mathsf{SN}$ and $N^A \in \mathsf{SN}$ the following holds.*

(a) $M[v := N] \in \mathsf{SN}$.
(a') $M[x := r] \in \mathsf{SN}$.
(b) *Suppose M derives $A \to B$. Then $MN \in \mathsf{SN}$.*
(b') *Suppose M derives $\forall x A$. Then $Mr \in \mathsf{SN}$.*

PROOF. By course-of-values induction on $\mathsf{dp}(A)$, with a side induction on $M \in \mathsf{SN}$. Let $N^A \in \mathsf{SN}$. We distinguish cases on the form of M.

Case $u\vec{M}$ by (Var) from $\vec{M} \in \mathsf{SN}$. (a). The SIH(a) (SIH means side induction hypothesis) yields $M_i[v := N] \in \mathsf{SN}$ for all M_i from $\vec{M}$. In case $u \neq v$ we immediately have $(u\vec{M})[v := N] \in \mathsf{SN}$. Otherwise we need $N\vec{M}[v := N] \in \mathsf{SN}$. But this follows by multiple applications of IH(b), since every $M_i[v := N]$ derives a subformula of A with smaller depth. (a'). Similar, and simpler. (b), (b'). Use (Var) again.

Case $\lambda v M$ by (λ) from $M \in \mathsf{SN}$. (a), (a'). Use (λ) again. (b). Our goal is $(\lambda v M)N \in \mathsf{SN}$. By ($\beta$) it suffices to show $M[v := N] \in \mathsf{SN}$ and $N \in \mathsf{SN}$. The latter holds by assumption, and the former by SIH(a). (b'). Similar, and simpler.

Case $(\lambda w M)K\vec{L}$ by (β) from $M[w := K]\vec{L} \in \mathsf{SN}$ and $K \in \mathsf{SN}$. (a). The SIH(a) yields $M[v := N][w := K[v := N]]\vec{L}[v := N] \in \mathsf{SN}$ and $K[v := N] \in \mathsf{SN}$, hence $(\lambda w M[v := N])K[v := N]\vec{L}[v := N] \in \mathsf{SN}$ by (β). (a'). Similar, and simpler. (b), (b'). Use (β) again. $\square$

COROLLARY. *For every term we have $M \in \mathsf{SN}$; in particular every term M is strongly normalizable.*

PROOF. Induction on the (first) inductive definition of derivation terms M. In cases u and $\lambda v M$ the claim follows from the definition of SN, and in case MN it follows from the preceding theorem. $\square$

3.5. Confluence. A relation R is said to be *confluent*, or to have the *Church–Rosser property (CR)*, if, whenever $M_0 \, R \, M_1$ and $M_0 \, R \, M_2$, then there is an M_3 such that $M_1 \, R \, M_3$ and $M_2 \, R \, M_3$. A relation R is said to be *weakly confluent*, or to have the *weak Church–Rosser property (WCR)*, if, whenever $M_0 \, R \, M_1, M_0 \, R \, M_2$ then there is an M_3 such that $M_1 \, R^* \, M_3$ and $M_2 \, R^* \, M_3$, where R^* is the reflexive and transitive closure of R.

Clearly for a confluent reduction relation $\to^*$ the normal forms of terms are unique.

LEMMA (Newman 1942). *Let $\to^*$ be the transitive and reflexive closure of $\to$, and let $\to$ be weakly confluent. Then the normal form w.r.t. $\to$ of a strongly normalizing M is unique. Moreover, if all terms are strongly normalizing w.r.t. $\to$, then the relation $\to^*$ is confluent.*

PROOF. Call M *good* if it satisfies the confluence property w.r.t. $\to^*$, i.e. if whenever $K \leftarrow^* M \to^* L$, then $K \to^* N \leftarrow^* L$ for some N. We show that every strongly normalizing M is good, by transfinite induction on the well-founded partial order $\to^+$, restricted to all terms occurring in the reduction tree of M. So let M be given and assume

$$\forall M'. M \to^+ M' \Longrightarrow M' \text{ is good.}$$

We must show that M is good, so assume $K \leftarrow^* M \to^* L$. We may further assume that there are M', M'' such that $K \leftarrow^* M' \leftarrow M \to M'' \to^* L$, for otherwise the claim is trivial. But then the claim follows from the assumed weak confluence and the induction hypothesis for M' and M'', as shown in the picture below. $\square$

3.6. Uniqueness of Normal Forms. We first show that $\to$ is weakly confluent. From this and the fact that it is strongly normalizing we can easily infer (using Newman's Lemma) that the normal forms are unique.

PROPOSITION. *$\to$ is weakly confluent.*

PROOF. Assume $N_0 \leftarrow M \to N_1$. We show that $N_0 \to^* N \leftarrow^* N_1$ for some N, by induction on M. If there are two inner reductions both on the same subterm, then the claim follows from the induction hypothesis using substitutivity. If they are on distinct subterms, then the subterms do not

$$
\begin{tikzcd}
 & & M \arrow[dl] \arrow[dr] & & \\
 & M' \arrow[dl, "*"'] \arrow[dr, "*"] \arrow[rr, phantom, "\text{weak conf.}"] & & M'' \arrow[dl, "*"'] \arrow[dr, "*"] & \\
K \arrow[dr, "*"'] \arrow[rr, phantom, "\mathrm{IH}(M')"] & & \exists N' \arrow[dl, "*"] & & L \arrow[ddll, "*"] \\
 & \exists N'' \arrow[dr, "*"'] & & \mathrm{IH}(M'') & \\
 & & \exists N & &
\end{tikzcd}
$$

TABLE 2. Proof of Newman's lemma

overlap and the claim is obvious. It remains to deal with the case of a head reduction together with an inner conversion.

$$
\begin{tikzcd}
 & (\lambda v M)N\vec{L} \arrow[dl] \arrow[dr] & \\
M[v:=N]\vec{L} \arrow[dr] & & (\lambda v M')N\vec{L} \arrow[dl] \\
 & M'[v:=N]\vec{L} &
\end{tikzcd}
\qquad
\begin{tikzcd}
 & (\lambda v M)N\vec{L} \arrow[dl] \arrow[dr] & \\
M[v:=N]\vec{L} \arrow[dr, "*"] & & (\lambda v M)N'\vec{L} \arrow[dl] \\
 & M[v:=N']\vec{L} &
\end{tikzcd}
$$

$$
\begin{tikzcd}
 & (\lambda v M)N\vec{L} \arrow[dl] \arrow[dr] & \\
M[v:=N]\vec{L} \arrow[dr] & & (\lambda v M)N\vec{L}' \arrow[dl] \\
 & M[v:=N]\vec{L}' &
\end{tikzcd}
$$

where for the lower left arrows we have used substitutivity again. □

COROLLARY. *Every term is strongly normalizing, hence normal forms are unique.* □

3.7. The Structure of Normal Derivations. Let M be a normal derivation, viewed as a prooftree. A sequence of f.o.'s (formula occurrences) $A_0, \dots, A_n$ such that (1) A_0 is a top formula (leaf) of the prooftree, and (2) for $0 \le i < n$, A_{i+1} is immediately below A_i, and (3) A_i is not the minor premise of an $\to^-$-application, is called a *track* of the deduction tree M. A track *of order* 0 ends in the conclusion of M; a track of *order* $n+1$ ends in the minor premise of an $\to^-$-application with major premise belonging to a track of order n.

Since by normality an E-rule cannot have the conclusion of an I-rule as its major premise, the E-rules have to precede the I-rules in a track, so the following is obvious: a track may be divided into an E-part, say $A_0, \dots, A_{i-1}$, a minimal formula A_i, and an I-part $A_{i+1}, \dots, A_n$. In the E-part all rules are E-rules; in the I-part all rules are I-rules; A_i is the conclusion of an E-rule and, if $i < n$, a premise of an I-rule. It is also easy to see that each f.o. of M belongs to some track. Tracks are pieces of branches of the tree with successive f.o.'s in the subformula relationship: either A_{i+1} is a subformula of A_i or vice versa. As a result, all formulas in a track $A_0, \dots, A_n$ are subformulas of A_0 or of A_n; and from this, by induction on the order of tracks, we see that every formula in M is a subformula either of an open assumption or of the conclusion. To summarize, we have seen:

LEMMA. *In a normal derivation each formula occurrence belongs to some track.*

PROOF. By induction on the height of normal derivations. □

THEOREM. *In a normal derivation each formula is a subformula of either the end formula or else an assumption formula.*

PROOF. We prove this for tracks of order n, by induction on n. □

3.8. Normal Versus Non-Normal Derivations. We now show that the requirement to give a normal derivation of a derivable formula can sometimes be unrealistic. Following Statman [**16**] and Orevkov [**13**] we give examples of formulas C_k which are easily derivable with non-normal derivations (whose number of nodes is linear in k), but which require a non-elementary (in k) number of nodes in any normal derivation.

The example is related to Gentzen's proof in [**6**] of transfinite induction up to ω_k in arithmetic. There the function $y \oplus \omega^x$ plays a crucial role, and also the assignment of a "lifting"-formula $A^+(x)$ to any formula $A(x)$, by

$$A^+(x) := \forall y.(\forall z{\prec}y)A(z) \to (\forall z \prec y \oplus \omega^x)A(z).$$

Here we consider the numerical function $y + 2^x$ instead, and axiomatize its graph by means of Horn clauses. The formula C_k expresses that from these axioms the existence of 2_k follows. A short, non-normal proof of this fact can then be given by a modification of Gentzen's idea, and it is easily seen that any normal proof of C_k must contain at least 2_k nodes.

The derivations to be given make heavy use of the existential quantifier $\exists^{\mathsf{cl}}$ defined by $\neg\forall\neg$. In particular we need:

LEMMA (Existence Introduction). $\vdash A \to \exists^{\mathsf{cl}} xA$.

PROOF. $\lambda u^A \lambda v^{\forall x \neg A}.vxu$. □

LEMMA (Existence Elimination). $\vdash (\neg\neg B \to B) \to \exists^{\mathsf{cl}} xA \to (\forall x.A \to B) \to B$ *if* $x \notin FV(B)$.

PROOF. $\lambda u^{\neg\neg B \to B} \lambda v^{\neg\forall x \neg A} \lambda w^{\forall x.A \to B}.u\lambda u_2^{\neg B}.v\lambda x \lambda u_1^A.u_2(wxu_1)$. □

Note that the stability assumption $\neg\neg B \to B$ is not needed if B does not contain an atom $\neq \perp$ as a strictly positive subformula. This will be the case for the derivations below, where B will always be a classical existential formula.

Let us now fix our language. We use a ternary relation symbol R to represent the graph of the function $y + 2^x$; so $R(y, x, z)$ is intended to mean $y + 2^x = z$. We now axiomatize R by means of Horn clauses. For simplicity we use a unary function symbol s (to be viewed as the successor function) and a constant 0; one could use logic without function symbols instead – as Orevkov does –, but this makes the formulas somewhat less readable and the proofs less perspicious. Note that Orevkov's result is an adaption of a result of Statman [**16**] for languages containing function symbols.

$$\mathsf{Hyp}_1\colon \forall y R(y, 0, s(y))$$
$$\mathsf{Hyp}_2\colon \forall y, x, z, z_1 . R(y, x, z) \to R(z, x, z_1) \to R(y, s(x), z_1)$$

The goal formula then is

$$C_k := \exists^{\mathsf{cl}} z_k, \dots, z_0 . R(0, 0, z_k) \wedge R(0, z_k, z_{k-1}) \wedge \dots \wedge R(0, z_1, z_0).$$

To obtain the short proof of the goal formula C_k we use formulas $A_i(x)$ with a free parameter x.

$$A_0(x) := \forall y \exists^{\mathsf{cl}} z\, R(y, x, z),$$
$$A_{i+1}(x) := \forall y . A_i(y) \to \exists^{\mathsf{cl}} z . A_i(z) \wedge R(y, x, z).$$

For the two lemmata to follow we give an informal argument, which can easily be converted into a formal proof. Note that the existence elimination lemma is used only with existential formulas as conclusions. Hence it is not necessary to use stability axioms and we have a derivation in minimal logic.

LEMMA. $\vdash \mathsf{Hyp}_1 \to \mathsf{Hyp}_2 \to A_i(0)$.

PROOF. **Case** $i = 0$. Obvious by Hyp_1.

Case $i = 1$. Let x with $A_0(x)$ be given. It is sufficient to show $A_0(s(x))$, that is $\forall y \exists^{\mathsf{cl}} z_1 R(y, s(x), z_1)$. So let y be given. We know

$$A_0(x) = \forall y \exists^{\mathsf{cl}} z\, R(y, x, z). \tag{4}$$

Applying (4) to our y gives z such that $R(y, x, z)$. Applying (4) again to this z gives z_1 such that $R(z, x, z_1)$. By Hyp_2 we obtain $R(y, s(x), z_1)$.

Case $i + 2$. Let x with $A_{i+1}(x)$ be given. It suffices to show $A_{i+1}(s(x))$, that is $\forall y . A_i(y) \to \exists^{\mathsf{cl}} z . A_i(z) \wedge R(y, s(x), z)$. So let y with $A_i(y)$ be given. We know

$$A_{i+1}(x) = \forall y . A_i(y) \to \exists^{\mathsf{cl}} z_1 . A_i(z_1) \wedge R(y, x, z_1). \tag{5}$$

Applying (5) to our y gives z such that $A_i(z)$ and $R(y, x, z)$. Applying (5) again to this z gives z_1 such that $A_i(z_1)$ and $R(z, x, z_1)$. By Hyp_2 we obtain $R(y, s(x), z_1)$. □

Note that the derivations given have a fixed length, independent of i.

LEMMA. $\vdash \mathsf{Hyp}_1 \to \mathsf{Hyp}_2 \to C_k$.

PROOF. $A_k(0)$ applied to 0 and $A_{k-1}(0)$ yields z_k with $A_{k-1}(z_k)$ such that $R(0, 0, z_k)$.

$A_{k-1}(z_k)$ applied to 0 and $A_{k-2}(0)$ yields z_{k-1} with $A_{k-2}(z_{k-1})$ such that $R(0, z_k, z_{k-1})$.

$A_1(z_2)$ applied to 0 and $A_0(0)$ yields z_1 with $A_0(z_1)$ such that $R(0, z_2, z_1)$.

$A_0(z_1)$ applied to 0 yields z_0 with $R(0, z_1, z_0)$. □

Note that the derivations given have length linear in k.

We want to compare the length of this derivation of C_k with the length of an arbitrary normal derivation.

Proposition. *Any normal derivation of C_k from* Hyp_1 *and* Hyp_2 *has at least* 2_k *nodes.*

Proof. Let a normal derivation M of falsity $\perp$ from Hyp_1, Hyp_2 and the additional hypothesis

$$u\colon \forall z_k, \ldots, z_0. R(0,0,z_k) \to R(0,z_k,z_{k-1}) \to \cdots \to R(0,z_1,z_0) \to \perp$$

be given. We may assume that M does not contain free object variables (otherwise substitute them by 0). The main branch of M must begin with u, and its side premises are all of the form $R(0, s^n(0), s^k(0))$.

Observe that any normal derivation of $R(s^m(0), s^n(0), s^k(0))$ from Hyp_1, Hyp_2 and u has at least 2^n occurrences of Hyp_1 and is such that $k = m + 2^n$. This can be seen easily by induction on n. Note also that such a derivation cannot involve u.

If we apply this observation to the above derivations of the side premises we see that they derive

$$R(0,0,s^{2^0}(0)), \quad R(0,s^{2^0}(0),s^{2^{2^0}}(0)), \quad \ldots \quad R(0,s^{2_{k-1}}(0),s^{2_k}(0)).$$

The last of these derivations uses at least $2^{2_{k-1}} = 2_k$-times Hyp_1. $\square$

4. Normalization including Permutative Conversions

The elimination of "detours" done in Section 3 will now be extended to the full language. However, incorporation of $\vee$, $\wedge$ and $\exists$ leads to difficulties. If we do this by means of axioms (or constant derivation terms, as in 2.3), we cannot read off as much as we want from a normal derivation. If we do it in the form of rules, we must also allow *permutative conversion*. The reason for the difficulty is that in the elimination rules for $\vee, \wedge, \exists$ the minor premise reappears in the conclusion. This gives rise to a situation where we first introduce a logical connective, then do not touch it (by carrying it along in minor premises of $\vee^-, \wedge^-, \exists^-$), and finally eliminate the connective. This is not a detour as we have treated them in Section 3, and the conversion introduced there cannot deal with this situation. What has to be done is a permutative conversion: permute an elimination immediately following an $\vee^-, \wedge^-, \exists^-$-rule over this rule to the minor premise.

We will show that any sequence of such conversion steps terminates in a normal form, which in fact is uniquely determined (again by Newman's lemma).

Derivations in normal form have many pleasant properties, for instance:

- **Subformula property:** every formula occurring in a normal derivation is a subformula of either the conclusion or else an assumption;
- **Explicit definability:** a normal derivation of a formula $\exists x A$ from assumptions not involving disjunctive of existential strictly positive parts ends with an existence introduction, hence also provides a term r and a derivation of $A[x := r]$;
- **Disjunction property:** a normal derivation of a disjunction $A \vee B$ from assumptions not involving disjunctions as strictly positive parts ends with a disjunction introduction, hence also provides either a derivation of A or else one of B;

4.1. Rules for $\vee$, $\wedge$ and $\exists$. Notice that we have not given rules for the connectives $\vee$, $\wedge$ and $\exists$. There are two reasons for this omission:

- They can be covered by means of appropriate axioms as constant derivation terms, as given in 2.3;
- For simplicity we want our derivation terms to be pure lambda terms formed just by lambda abstraction and application. This would be violated by the rules for $\vee$, $\wedge$ and $\exists$, which require additional constructs.

However – as just noted – in order to have a normalization theorem with a useful subformula property as a consequence we do need to consider rules for these connectives. So here they are:

Disjunction. The introduction rules are

$$\dfrac{\begin{array}{c}\mid M\\ A\end{array}}{A \vee B}\,\vee_0^+ \qquad \dfrac{\begin{array}{c}\mid M\\ B\end{array}}{A \vee B}\,\vee_1^+$$

and the elimination rule is

$$\dfrac{\begin{array}{c}\\ \mid M\\ A \vee B\end{array} \qquad \begin{array}{c}[u\colon A]\\ \mid N\\ C\end{array} \qquad \begin{array}{c}[v\colon B]\\ \mid K\\ C\end{array}}{C}\,\vee^- u, v$$

Conjunction. The introduction rule is

$$\dfrac{\begin{array}{c}\mid M\\ A\end{array} \qquad \begin{array}{c}\mid N\\ B\end{array}}{A \wedge B}\,\wedge^+$$

and the elimination rule is

$$\dfrac{\begin{array}{c}\\ \mid M\\ A \wedge B\end{array} \qquad \begin{array}{c}[u\colon A] \quad [v\colon B]\\ \mid N\\ C\end{array}}{C}\,\wedge^- u, v$$

Existential Quantifier. The introduction rule is

$$\dfrac{r \qquad \begin{array}{c}\mid M\\ A[x := r]\end{array}}{\exists x A}\,\exists^+$$

and the elimination rule is

$$\dfrac{\begin{array}{c}\\ \mid M\\ \exists x A\end{array} \qquad \begin{array}{c}[u\colon A]\\ \mid N\\ B\end{array}}{B}\,\exists^- x, u \text{ (var.cond.)}$$

The rule $\exists^- x, u$ is subject to the following *(Eigen-) variable condition*: The derivation N should not contain any open assumptions apart from $u\colon A$ whose assumption formula contains x free, and moreover B should not contain the variable x free.

It is easy to see that for each of the connectives $\vee$, $\wedge$, $\exists$ the rules and the axioms are equivalent, in the sense that from the axioms and the premises of a rule we can derive its conclusion (of course without any $\vee, \wedge, \exists$-rules), and conversely that we can derive the axioms by means of the $\vee, \wedge, \exists$-rules. This is left as an exercise.

The left premise in each of the elimination rules $\vee^-$, $\wedge^-$ and $\exists^-$ is called *major premise* (or *main* premise), and each of the right premises *minor premise* (or *side* premise).

4.2. Conversion. In addition to the $\to, \forall$-conversions treated in 3.1, we consider the following conversions:

$\vee$-*conversion.*

$$\dfrac{\dfrac{\begin{matrix}\mid M\\ A\end{matrix}}{A\vee B}\,\vee_0^+ \qquad \begin{matrix}[u\colon A]\\ \mid N\\ C\end{matrix} \qquad \begin{matrix}[v\colon B]\\ \mid K\\ C\end{matrix}}{C}\,\vee^- u,v \qquad\mapsto\qquad \begin{matrix}\mid M\\ A\\ \mid N\\ C\end{matrix}$$

and

$$\dfrac{\dfrac{\begin{matrix}\mid M\\ B\end{matrix}}{A\vee B}\,\vee_1^+ \qquad \begin{matrix}[u\colon A]\\ \mid N\\ C\end{matrix} \qquad \begin{matrix}[v\colon B]\\ \mid K\\ C\end{matrix}}{C}\,\vee^- u,v \qquad\mapsto\qquad \begin{matrix}\mid M\\ B\\ \mid K\\ C\end{matrix}$$

$\wedge$-*conversion.*

$$\dfrac{\dfrac{\begin{matrix}\mid M\\ A\end{matrix}\qquad\begin{matrix}\mid N\\ B\end{matrix}}{A\wedge B}\,\wedge^+ \qquad \begin{matrix}[u\colon A]\quad[v\colon B]\\ \mid K\\ C\end{matrix}}{C}\,\wedge^- u,v \qquad\mapsto\qquad \begin{matrix}\begin{matrix}\mid M\\ A\end{matrix}\qquad\begin{matrix}\mid N\\ B\end{matrix}\\ \mid K\\ C\end{matrix}$$

$\exists$-*conversion.*

$$\dfrac{\dfrac{r\qquad\begin{matrix}\mid M\\ A[x:=r]\end{matrix}}{\exists x A}\,\exists^+ \qquad \begin{matrix}[u\colon A]\\ \mid N\\ B\end{matrix}}{B}\,\exists^- x,u \qquad\mapsto\qquad \begin{matrix}\mid M\\ A[x:=r]\\ \mid N'\\ B\end{matrix}$$

4.3. Permutative Conversion. In a permutative conversion we permute an E-rule upwards over the minor premises of $\vee^-$, $\wedge^-$ or $\exists^-$.

$\vee$-*perm conversion.*

$$\dfrac{\dfrac{\begin{matrix}\mid M\\ A\vee B\end{matrix}\qquad\begin{matrix}\mid N\\ C\end{matrix}\qquad\begin{matrix}\mid K\\ C\end{matrix}}{C}\qquad\begin{matrix}\mid L\\ C'\end{matrix}}{D}\,\text{E-rule}\qquad\mapsto$$

$$\dfrac{\begin{matrix}\mid M\\ A\vee B\end{matrix}\qquad \dfrac{\begin{matrix}\mid N\\ C\end{matrix}\qquad\begin{matrix}\mid L\\ C'\end{matrix}}{D}\,\text{E-rule}\qquad \dfrac{\begin{matrix}\mid K\\ C\end{matrix}\qquad\begin{matrix}\mid L\\ C'\end{matrix}}{D}\,\text{E-rule}}{D}$$

$\wedge$-perm conversion.

$$\dfrac{\dfrac{\begin{matrix}\mid M\\ A\wedge B\end{matrix} \qquad \begin{matrix}\mid N\\ C\end{matrix}}{C} \qquad \begin{matrix}\mid K\\ C'\end{matrix}}{D}\,\text{E-rule} \quad \mapsto$$

$$\dfrac{\begin{matrix}\mid M\\ A\wedge B\end{matrix} \qquad \dfrac{\begin{matrix}\mid N\\ C\end{matrix} \qquad \begin{matrix}\mid K\\ C'\end{matrix}}{D}\,\text{E-rule}}{D}$$

$\exists$-perm conversion.

$$\dfrac{\dfrac{\begin{matrix}\mid M\\ \exists xA\end{matrix} \qquad \begin{matrix}\mid N\\ B\end{matrix}}{B} \qquad \begin{matrix}\mid K\\ C\end{matrix}}{D}\,\text{E-rule} \quad \mapsto$$

$$\dfrac{\begin{matrix}\mid M\\ \exists xA\end{matrix} \qquad \dfrac{\begin{matrix}\mid N\\ B\end{matrix} \qquad \begin{matrix}\mid K\\ C\end{matrix}}{D}\,\text{E-rule}}{D}$$

4.4. Derivations as Terms. The term representation of derivations has to be extended. The rules for $\vee$, $\wedge$ and $\exists$ with the corresponding terms are given in the table below.

The introduction rule $\exists^+$ has as its left premise the witnessing term r to be substituted. The elimination rule $\exists^- u$ is subject to an *(Eigen-) variable condition*: The derivation term N should not contain any open assumptions apart from $u\colon A$ whose assumption formula contains x free, and moreover B should not contain the variable x free.

4.5. Permutative Conversions. In this section we shall write derivation terms without formula superscripts. We usually leave implicit the extra (formula) parts of derivation constants and for instance write $\exists^+$, $\exists^-$ instead of $\exists^+_{x,A}$, $\exists^-_{x,A,B}$. So we consider derivation terms M, N, K of the forms

$$\begin{gathered} u \mid \lambda vM \mid \lambda yM \mid \vee^+_0 M \mid \vee^+_1 M \mid \langle M, N\rangle \mid \exists^+ rM \mid \\ MN \mid Mr \mid M(v_0.N_0, v_1.N_1) \mid M(v, w.N) \mid M(v.N); \end{gathered}$$

in these expressions the variables y, v, v_0, v_1, w get bound.

To simplify the technicalities, we restrict our treatment to the rules for $\to$ and $\exists$. It can easily be extended to the full set of rules; some details for disjunction are given in 4.6. So we consider

$$u \mid \lambda vM \mid \exists^+ rM \mid MN \mid M(v.N);$$

in these expressions the variable v gets bound.

We reserve the letters E, F, G for *eliminations*, i.e. expressions of the form $(v.N)$, and R, S, T for both terms and eliminations. Using this notation we obtain a second (and clearly equivalent) inductive definition of terms:

$$\begin{gathered} u\vec{M} \mid u\vec{M}E \mid \lambda vM \mid \exists^+ rM \mid \\ (\lambda vM)N\vec{R} \mid \exists^+ rM(v.N)\vec{R} \mid u\vec{M}ER\vec{S}. \end{gathered}$$

derivation	term
$\dfrac{\begin{array}{c}\mid M\\ A\end{array}}{A\lor B}\lor^+_0 \qquad \dfrac{\begin{array}{c}\mid M\\ B\end{array}}{A\lor B}\lor^+_1$	$(\lor^+_{0,B}M^A)^{A\lor B} \quad (\lor^+_{1,A}M^B)^{A\lor B}$
$\dfrac{\begin{array}{c}\mid M\\ A\lor B\end{array}\quad \begin{array}{c}[u\colon A]\\ \mid N\\ C\end{array}\quad \begin{array}{c}[v\colon B]\\ \mid K\\ C\end{array}}{C}\lor^- u,v$	$\bigl(M^{A\lor B}(u^A.N^C, v^B.K^C)\bigr)^C$
$\dfrac{\begin{array}{c}\mid M\\ A\end{array}\quad \begin{array}{c}\mid N\\ B\end{array}}{A\land B}\land^+$	$\langle M^A, N^B\rangle^{A\land B}$
$\dfrac{\begin{array}{c}\mid M\\ A\land B\end{array}\quad \begin{array}{c}[u\colon A]\quad [v\colon B]\\ \mid N\\ C\end{array}}{C}\land^- u,v$	$\bigl(M^{A\land B}(u^A, v^B.N^C)\bigr)^C$
$\dfrac{r\quad \begin{array}{c}\mid M\\ A[x:=r]\end{array}}{\exists xA}\exists^+$	$\bigl(\exists^+_{x,A} r M^{A[x:=r]}\bigr)^{\exists xA}$
$\dfrac{\begin{array}{c}\mid M\\ \exists xA\end{array}\quad \begin{array}{c}[u\colon A]\\ \mid N\\ B\end{array}}{B}\exists^- x,u$ (var.cond.)	$\bigl(M^{\exists xA}(u^A.N^B)\bigr)^B$ (var.cond.)

TABLE 3. Derivation terms for $\lor$, $\land$ and $\exists$

Here the final three forms are not normal: $(\lambda v M)N\vec{R}$ and $\exists^+ rM(v.N)\vec{R}$ both are *β-redexes*, and $u\vec{M}ER\vec{S}$ is a *permutative redex*. The conversion rules are

$$\begin{array}{lll}
(\lambda v M)N & \mapsto_\beta M[v:=N] & \beta_\to\text{-conversion,}\\
\exists^+_{x,A} rM(v.N) & \mapsto_\beta N[x:=r][v:=M] & \beta_\exists\text{-conversion,}\\
M(v.N)R & \mapsto_\pi M(v.NR) & \text{permutative conversion.}
\end{array}$$

The *closure* of these conversions is defined by

- If $M \mapsto_\beta M'$ or $M \mapsto_\pi M'$, then $M \to M'$.
- If $M \to M'$, then also $MR \to M'R$, $NM \to NM'$, $N(v.M) \to N(v.M')$, $\lambda vM \to \lambda vM'$, $\exists^+ rM \to \exists^+ rM'$ (*inner reductions*).

We now give the rules to inductively generate a set SN:

$$\frac{\vec{M} \in \mathsf{SN}}{u\vec{M} \in \mathsf{SN}}\,(\mathsf{Var}_0) \qquad \frac{M \in \mathsf{SN}}{\lambda vM \in \mathsf{SN}}\,(\lambda) \qquad \frac{M \in \mathsf{SN}}{\exists^+ rM \in \mathsf{SN}}\,(\exists)$$

$$\frac{\vec{M}, N \in \mathsf{SN}}{u\vec{M}(v.N) \in \mathsf{SN}}\,(\mathsf{Var}) \qquad \frac{u\vec{M}(v.NR)\vec{S} \in \mathsf{SN}}{u\vec{M}(v.N)R\vec{S} \in \mathsf{SN}}\,(\mathsf{Var}_\pi)$$

$$\frac{M[v := N]\vec{R} \in \mathsf{SN} \qquad N \in \mathsf{SN}}{(\lambda vM)N\vec{R} \in \mathsf{SN}}\,(\beta_\to)$$

$$\frac{N[x := r][v := M]\vec{R} \in \mathsf{SN} \qquad M \in \mathsf{SN}}{\exists^+_{x,A} rM(v.N)\vec{R} \in \mathsf{SN}}\,(\beta_\exists)$$

where in (Var_π) we require that v is not free in R.

Write $M\downarrow$ to mean that M is strongly normalizable, i.e., that every reduction sequence starting from M terminates. By analyzing the possible reduction steps we now show that the set $\mathsf{Wf} := \{\, M \mid M\downarrow \,\}$ has the closure properties of the definition of SN above, and hence $\mathsf{SN} \subseteq \mathsf{Wf}$.

LEMMA. *Every term in* SN *is strongly normalizable.*

PROOF. We distinguish cases according to the generation rule of SN applied last. The following rules deserve special attention.

Case (Var_π). We prove, as an auxiliary lemma, that

$$u\vec{M}(v.NR)\vec{S}\downarrow \text{ implies } u\vec{M}(v.N)R\vec{S}\downarrow,$$

by induction on $u\vec{M}(v.NR)\vec{S}\downarrow$ (i.e., on the reduction tree of this term). We consider the possible reducts of $u\vec{M}(v.N)R\vec{S}$. The only interesting case is $R\vec{S} = (v'.N')T\vec{T}$ and we have a permutative conversion of $R = (v'.N')$ with T, leading to the term $M = u\vec{M}(v.N)(v'.N'T)\vec{T}$. Now $M\downarrow$ follows, since

$$u\vec{M}(v.NR)\vec{S} = u\vec{M}(v.N(v'.N'))T\vec{T}$$

leads in two permutative steps to $u\vec{M}(v.N(v'.N'T))\vec{T}$, hence for this term we have the induction hypothesis available.

Case $(\beta_\to)$. We show that $M[v := N]\vec{R}\downarrow$ and $N\downarrow$ imply $(\lambda vM)N\vec{R}\downarrow$. This is done by a induction on $N\downarrow$, with a side induction on $M[v := N]\vec{R}\downarrow$. We need to consider all possible reducts of $(\lambda vM)N\vec{R}$. In case of an outer β-reduction use the assumption. If N is reduced, use the induction hypothesis. Reductions in M and in $\vec{R}$ as well as permutative reductions within $\vec{R}$ are taken care of by the side induction hypothesis.

Case $(\beta_\exists)$. We show that $N[x := r][v := M]\vec{R}\downarrow$ and $M\downarrow$ together imply $\exists^+ rM(v.N)\vec{R}\downarrow$. This is done by a threefold induction: first on $M\downarrow$, second on $N[x := r][v := M]\vec{R}\downarrow$ and third on the length of $\vec{R}$. We need to consider all possible reducts of $\exists^+ rM(v.N)\vec{R}$. In case of an outer β-reduction use the assumption. If M is reduced, use the first induction hypothesis. Reductions in N and in $\vec{R}$ as well as permutative reductions within $\vec{R}$ are taken care of by the second induction hypothesis. The only remaining case is when $\vec{R} = S\vec{S}$ and $(v.N)$ is permuted with S, to yield $\exists^+ rM(v.NS)\vec{S}$. Apply the third induction hypothesis, since $(NS)[x := r][v := M]\vec{S} = N[x := r][v := M]S\vec{S}$. $\square$

For later use we prove a slightly generalized form of the rule (Var_π):

PROPOSITION. *If $M(v.NR)\vec{S} \in \mathsf{SN}$, then $M(v.N)R\vec{S} \in \mathsf{SN}$.*

PROOF. Induction on the generation of $M(v.NR)\vec{S} \in \mathsf{SN}$. We distinguish cases according to the form of M.

Case $u\vec{T}(v.NR)\vec{S} \in \mathsf{SN}$. If $\vec{T} = \vec{M}$, use (Var_π). Otherwise we have $u\vec{M}(v'.N')\vec{R}(v.NR)\vec{S} \in \mathsf{SN}$. This must be generated by repeated applications of (Var_π) from $u\vec{M}(v'.N'\vec{R}(v.NR)\vec{S}) \in \mathsf{SN}$, and finally by (Var) from $\vec{M} \in \mathsf{SN}$ and $N'\vec{R}(v.NR)\vec{S} \in \mathsf{SN}$. The induction hypothesis for the latter yields $N'\vec{R}(v.N)R\vec{S} \in \mathsf{SN}$, hence $u\vec{M}(v.N'\vec{R}(v.N)R\vec{S}) \in \mathsf{SN}$ by (Var) and finally $u\vec{M}(v.N')\vec{R}(v.N)R\vec{S} \in \mathsf{SN}$ by (Var_π).

Case $\exists^+ rM\vec{T}(v.NR)\vec{S} \in \mathsf{SN}$. Similarly, with $(\beta_\exists)$ instead of (Var_π). In detail: If $\vec{T}$ is empty, by $(\beta_\exists)$ this came from $(NR)[x := r][v := M]\vec{S} = N[x := r][v := M]R\vec{S} \in \mathsf{SN}$ and $M \in \mathsf{SN}$, hence $\exists^+ rM(v.N)R\vec{S} \in \mathsf{SN}$ again by $(\beta_\exists)$. Otherwise we have $\exists^+ rM(v'.N')\vec{T}(v.NR)\vec{S} \in \mathsf{SN}$. This must be generated by $(\beta_\exists)$ from $N'[x := r][v' := M]\vec{T}(v.NR)\vec{S} \in \mathsf{SN}$. The induction hypothesis yields $N'[x := r][v' := M]\vec{T}(v.N)R\vec{S} \in \mathsf{SN}$, hence $\exists^+ rM(v'.N')\vec{T}(v.N)R\vec{S} \in \mathsf{SN}$ by $(\beta_\exists)$.

Case $(\lambda vM)N'\vec{R}(w.NR)\vec{S} \in \mathsf{SN}$. By $(\beta_\to)$ this came from $N' \in \mathsf{SN}$ and $M[v := N']\vec{R}(w.NR)\vec{S} \in \mathsf{SN}$. The induction hypothesis yields $M[v := N']\vec{R}(w.N)R\vec{S} \in \mathsf{SN}$, hence $(\lambda vM)N'\vec{R}(w.N)R\vec{S} \in \mathsf{SN}$ by $(\beta_\to)$. $\square$

In what follows we shall show that *every* term is in SN and hence is strongly normalizable. Given the definition of SN we only have to show that SN is closed under $\to^-$ and $\exists^-$. In order to prove this we must prove simultaneously the closure of SN under substitution.

THEOREM (Properties of SN). *For all formulas A,*

(a) *for all $M \in \mathsf{SN}$, if M proves $A = A_0 \to A_1$ and $N \in \mathsf{SN}$, then $MN \in \mathsf{SN}$,*
(b) *for all $M \in \mathsf{SN}$, if M proves $A = \exists xB$ and $N \in \mathsf{SN}$, then $M(v.N) \in \mathsf{SN}$,*
(c) *for all $M \in \mathsf{SN}$, if $N^A \in \mathsf{SN}$, then $M[v := N] \in \mathsf{SN}$.*

PROOF. Induction on $\mathsf{dp}(A)$. We prove (a) and (b) before (c), and hence have (a) and (b) available for the proof of (c). More formally, by induction on A we simultaneously prove that (a) holds, that (b) holds and that (a), (b) together imply (c).

(a). By induction on $M \in \mathsf{SN}$. Let $M \in \mathsf{SN}$ and assume that M proves $A = A_0 \to A_1$ and $N \in \mathsf{SN}$. We distinguish cases according to how $M \in \mathsf{SN}$ was generated. For (Var_0), (Var_π), $(\beta_\to)$ and $(\beta_\exists)$ use the same rule again.

Case $u\vec{M}(v.N') \in \mathsf{SN}$ by (Var) from $\vec{M}, N' \in \mathsf{SN}$. Then $N'N \in \mathsf{SN}$ by side induction hypothesis for N', hence $u\vec{M}(v.N'N) \in \mathsf{SN}$ by (Var), hence $u\vec{M}(v.N')N \in \mathsf{SN}$ by (Var_π).

Case $(\lambda v M)^{A_0 \to A_1} \in \mathsf{SN}$ by (λ) from $M \in \mathsf{SN}$. Use $(\beta_\to)$; for this we need to know $M[v := N] \in \mathsf{SN}$. But this follows from IH(c) for M, since N derives A_0.

(b). By induction on $M \in \mathsf{SN}$. Let $M \in \mathsf{SN}$ and assume that M proves $A = \exists x B$ and $N \in \mathsf{SN}$. The goal is $M(v.N) \in \mathsf{SN}$. We distinguish cases according to how $M \in \mathsf{SN}$ was generated. For (Var_π), $(\beta_\to)$ and $(\beta_\exists)$ use the same rule again.

Case $u\vec{M} \in \mathsf{SN}$ by (Var_0) from $\vec{M} \in \mathsf{SN}$. Use (Var).

Case $(\exists^+ r M)^{\exists x A} \in \mathsf{SN}$ by $(\exists)$ from $M \in \mathsf{SN}$. Use $(\beta_\exists)$; for this we need to know $N[x := r][v := M] \in \mathsf{SN}$. But this follows from IH(c) for $N[x := r]$, since M derives $A[x := r]$.

Case $u\vec{M}(v'.N') \in \mathsf{SN}$ by (Var) from $\vec{M}, N' \in \mathsf{SN}$. Then $N'(v.N) \in \mathsf{SN}$ by side induction hypothesis for N', hence $u\vec{M}(v.N'(v.N)) \in \mathsf{SN}$ by (Var) and therefore $u\vec{M}(v.N')(v.N) \in \mathsf{SN}$ by (Var_π).

(c). By induction on $M \in \mathsf{SN}$. Let $N^A \in \mathsf{SN}$; the goal is $M[v := N] \in \mathsf{SN}$. We distinguish cases according to how $M \in \mathsf{SN}$ was generated. For (λ), $(\exists)$, $(\beta_\to)$ and $(\beta_\exists)$ use the same rule again.

Case $u\vec{M} \in \mathsf{SN}$ by (Var_0) from $\vec{M} \in \mathsf{SN}$. Then $\vec{M}[v := N] \in \mathsf{SN}$ by SIH(c). If $u \neq v$, use (Var_0) again. If $u = v$, we must show $N\vec{M}[v := N] \in \mathsf{SN}$. Note that N proves A; hence the claim follows from (a) and the induction hypothesis.

Case $u\vec{M}(v'.N') \in \mathsf{SN}$ by (Var) from $\vec{M}, N' \in \mathsf{SN}$. If $u \neq v$, use (Var) again. If $u = v$, we must show $N\vec{M}[v := N](v'.N'[v := N]) \in \mathsf{SN}$. Note that N proves A; hence in case $\vec{M}$ empty the claim follows from (b), and otherwise from (a) and the induction hypothesis.

Case $u\vec{M}(v'.N')R\vec{S} \in \mathsf{SN}$ by (Var_π) from $u\vec{M}(v'.N'R)\vec{S} \in \mathsf{SN}$. If $u \neq v$, use (Var_π) again. If $u = v$, from the induction hypothesis we obtain

$$N\vec{M}[v := N](v'.N'[v := N]R[v := N]).\vec{S}[v := N] \in \mathsf{SN}$$

Now use the proposition above. □

Corollary. *Every term is strongly normalizable.*

Proof. Induction on the (first) inductive definition of terms M. In cases u and $\lambda v M$ the claim follows from the definition of SN, and in cases MN and $M(v.N)$ it follows from parts (a), (b) of the previous theorem. □

4.6. Disjunction. We describe the changes necessary to extend the result above to the language with disjunction $\vee$.

We have additional β-conversions

$$\vee_i^+ M(v_0.N_0, v_1.N_1) \mapsto_\beta M[v_i := N_i] \quad \beta_{\vee_i}\text{-conversion.}$$

The definition of SN needs to be extended by

$$\frac{M \in \mathsf{SN}}{\vee_i^+ M \in \mathsf{SN}}\ (\vee_i)$$

$$\frac{\vec{M}, N_0, N_1 \in \mathsf{SN}}{u\vec{M}(v_0.N_0, v_1.N_1) \in \mathsf{SN}}\ (\mathsf{Var}_\vee) \qquad \frac{u\vec{M}(v_0.N_0R, v_1.N_1R)\vec{S} \in \mathsf{SN}}{u\vec{M}(v_0.N_0, v_1.N_1)R\vec{S} \in \mathsf{SN}}\ (\mathsf{Var}_{\vee,\pi})$$

$$\frac{N_i[v_i := M]\vec{R} \in \mathsf{SN} \qquad N_{1-i}\vec{R} \in \mathsf{SN} \qquad M \in \mathsf{SN}}{\vee_i^+ M(v_0.N_0, v_1.N_1)\vec{R} \in \mathsf{SN}}\ (\beta_{\vee_i})$$

The former rules (Var), (Var_π) should then be renamed into $(\mathsf{Var}_\exists)$, $(\mathsf{Var}_{\exists,\pi})$.

The lemma above stating that every term in SN is strongly normalizable needs to be extended by an additional clause:

Case $(\beta_{\vee_i})$. We show that $N_i[v_i := M]\vec{R}{\downarrow}$, $N_{1-i}\vec{R}{\downarrow}$ and $M{\downarrow}$ together imply $\vee_i^+ M(v_0.N_0, v_1.N_1)\vec{R}{\downarrow}$. This is done by a fourfold induction: first on $M{\downarrow}$, second on $N_i[v_i := M]\vec{R}{\downarrow}$, $N_{1-i}\vec{R}{\downarrow}$, third on $N_{1-i}\vec{R}{\downarrow}$ and fourth on the length of $\vec{R}$. We need to consider all possible reducts of $\vee_i^+ M(v_0.N_0, v_1.N_1)\vec{R}$. In case of an outer β-reduction use the assumption. If M is reduced, use the first induction hypothesis. Reductions in N_i and in $\vec{R}$ as well as permutative reductions within $\vec{R}$ are taken care of by the second induction hypothesis. Reductions in N_{1-i} are taken care of by the third induction hypothesis. The only remaining case is when $\vec{R} = S\vec{S}$ and $(v_0.N_0, v_1.N_1)$ is permuted with S, to yield $(v_0.N_0S, v_1.N_1S)$. Apply the fourth induction hypothesis, since $(N_iS)[v := M]\vec{S} = N_i[v := M]S\vec{S}$.

Finally the theorem above stating properties of SN needs an additional clause:

- for all $M \in \mathsf{SN}$, if M proves $A = A_0 \vee A_1$ and $N_0, N_1 \in \mathsf{SN}$, then $M(v_0.N_0, v_1.N_1) \in \mathsf{SN}$.

PROOF. The new clause is proved by induction on $M \in \mathsf{SN}$. Let $M \in \mathsf{SN}$ and assume that M proves $A = A_0 \vee A_1$ and $N_0, N_1 \in \mathsf{SN}$. The goal is $M(v_0.N_0, v_1.N_1) \in \mathsf{SN}$. We distinguish cases according to how $M \in \mathsf{SN}$ was generated. For $(\mathsf{Var}_{\exists,\pi})$, $(\mathsf{Var}_{\vee,\pi})$, $(\beta_\to)$, $(\beta_\exists)$ and $(\beta_{\vee_i})$ use the same rule again.

Case $u\vec{M} \in \mathsf{SN}$ by (Var_0) from $\vec{M} \in \mathsf{SN}$. Use $(\mathsf{Var}_\vee)$.

Case $(\vee_i^+ M)^{A_0 \vee A_1} \in \mathsf{SN}$ by $(\vee_i)$ from $M \in \mathsf{SN}$. Use $(\beta_{\vee_i})$; for this we need to know $N_i[v_i := M] \in \mathsf{SN}$ and $N_{1-i} \in \mathsf{SN}$. The latter is assumed, and the former follows from main induction hypothesis (with N_i) for the substitution clause of the theorem, since M derives A_i.

Case $u\vec{M}(v'.N') \in \mathsf{SN}$ by $(\mathsf{Var}_\exists)$ from $\vec{M}, N' \in \mathsf{SN}$. For brevity let $E := (v_0.N_0, v_1.N_1)$. Then $N'E \in \mathsf{SN}$ by side induction hypothesis for N', so $u\vec{M}(v'.N'E) \in \mathsf{SN}$ by $(\mathsf{Var}_\exists)$ and therefore $u\vec{M}(v'.N')E \in \mathsf{SN}$ by $(\mathsf{Var}_{\exists,\pi})$.

Case $u\vec{M}(v_0'.N_0', v_1'.N_1') \in \mathsf{SN}$ by $(\mathsf{Var}_\vee)$ from $\vec{M}, N_0', N_1' \in \mathsf{SN}$. Let $E := (v_0.N_0, v_1.N_1)$. Then $N_i'E \in \mathsf{SN}$ by side induction hypothesis for N_i', so $u\vec{M}(v_0'.N_0'E, v_1'.N_1'E) \in \mathsf{SN}$ by $(\mathsf{Var}_\vee)$ and therefore $u\vec{M}(v_0'.N_0', v_1'.N_1')E \in \mathsf{SN}$ by $(\mathsf{Var}_{\vee,\pi})$.

Clause (c) now needs additional cases, e.g.,

Case $u\vec{M}(v_0.N_0, v_1.N_1) \in \mathsf{SN}$ by $(\mathsf{Var}_\vee)$ from $\vec{M}, N_0, N_1 \in \mathsf{SN}$. If $u \neq v$, use $(\mathsf{Var}_\vee)$. If $u = v$, we show $N\vec{M}[v := N](v_0.N_0[v := N], v_1.N_1[v := N]) \in \mathsf{SN}$. Note that N proves A; hence in case $\vec{M}$ empty the claim follows from (b), and otherwise from (a) and the induction hypothesis. □

4.7. The Structure of Normal Derivations. As mentioned already, normalizations aim at removing local maxima of complexity, i.e. formula occurrences which are first introduced and immediately afterwards eliminated.

However, an introduced formula may be used as a minor premise of an application of $\vee^-$, $\wedge^-$ or $\exists^-$, then stay the same throughout a sequence of applications of these rules, being eliminated at the end. This also constitutes a local maximum, which we should like to eliminate; for that we need the so-called permutative conversions. First we give a precise definition.

DEFINITION. A *segment* of (*length* n) in a derivation M is a sequence $A_1, \dots, A_n$ of occurrences of a formula A such that

(a) for $1 < i < n$, A_i is a minor premise of an application of $\vee^-$, $\wedge^-$ or $\exists^-$, with conclusion A_{i+1};
(b) A_n is not a minor premise of $\vee^-$, $\wedge^-$ or $\exists^-$.
(c) A_1 is not the conclusion of $\vee^-$, $\wedge^-$ or $\exists^-$.

(Note: An f.o. which is neither a minor premise nor the conclusion of an application of $\vee^-$, $\wedge^-$ or $\exists^-$ always belongs to a segment of length 1.) A segment is *maximal* or a *cut (segment)* if A_n is the major premise of an E-rule, and either $n > 1$, or $n = 1$ and $A_1 = A_n$ is the conclusion of an I-rule.

We shall use σ, σ' for segments. We shall say that σ is a *subformula* of σ' if the formula A in σ is a subformula of B in σ'. Clearly a derivation is normal iff it does not contain a maximal segment.

The argument in 3.7 needs to be refined to also cover the rules for $\vee, \wedge, \exists$. The reason for the difficulty is that in the E-rules $\vee^-, \wedge^-, \exists^-$ the subformulas of a major premise $A \vee B$, $A \wedge B$ or $\exists x A$ of an E-rule application do not appear in the conclusion, but among the assumptions being discharged by the application. This suggests the definition of track below.

The general notion of a track is designed to retain the subformula property in case one passes through the major premise of an application of a $\vee^-, \wedge^-, \exists^-$-rule. In a track, when arriving at an A_i which is the major premise of an application of such a rule, we take for A_{i+1} a hypothesis discharged by this rule.

DEFINITION. A *track* of a derivation M is a sequence of f.o.'s $A_0, \dots, A_n$ such that

(a) A_0 is a top f.o. in M not discharged by an application of an $\vee^-, \wedge^-, \exists^-$-rule;
(b) A_i for $i < n$ is not the minor premise of an instance of $\to^-$, and *either*
  (i) A_i is not the major premise of an instance of a $\vee^-, \wedge^-, \exists^-$-rule and A_{i+1} is directly below A_i, *or*
  (ii) A_i is the major premise of an instance of a $\vee^-, \wedge^-, \exists^-$-rule and A_{i+1} is an assumption discharged by this instance;
(c) A_n is *either*
  (i) the minor premise of an instance of $\to^-$, *or*
  (ii) the conclusion of M, *or*
  (iii) the major premise of an instance of a $\vee^-, \wedge^-, \exists^-$-rule in case there are no assumptions discharged by this instance.

PROPOSITION. *Let M be a normal derivation, and let $\pi = \sigma_0, \dots, \sigma_n$ be a track in M. Then there is a segment σ_i in π, the* minimum *segment or* minimum *part of the track, which separates two (possibly empty) parts of π,* *called the E-part (*elimination part*) and the I-part (*introduction part*) of* π *such that*

(a) *for each* σ_j *in the E-part one has* $j < i$*,* σ_j *is a major premise of an E-rule, and* σ_{j+1} *is a strictly positive part of* σ_j*, and therefore each* σ_j *is a s.p.p. of* σ_0*;*

(b) *for each* σ_j *which is the minimum segment or is in the I-part one has* $i \leq j$*, and if* $j \neq n$*, then* σ_j *is a premise of an I-rule and a s.p.p. of* σ_{j+1}*, so each* σ_j *is a s.p.p. of* σ_n*.*

DEFINITION. A *track of order* 0, or *main track*, in a normal derivation is a track ending either in the conclusion of the whole derivation or in the major premise of an application of a $\vee^-$, $\wedge^-$ or $\exists^-$-rule, provided there are no assumption variables discharged by the application. A *track of order* $n+1$ is a track ending in the minor premise of an $\to^-$-application, with major premise belonging to a track of order n.

A *main branch* of a derivation is a branch π in the prooftree such that π passes only through premises of I-rules and *major premises* of E-rules, and π begins at a top node and ends in the conclusion.

REMARK. By an obvious *simplification conversion* we may remove every application of an $\vee^-$, $\wedge^-$ or $\exists^-$-rule that discharges no assumption variables. If such simplification conversion are performed, each track of order 0 in a normal derivation is a track ending in the conclusion of the whole derivation.

If we search for a main branch going upwards from the conclusion, the branch to be followed is unique as long as we do not encounter an $\wedge^+$-application.

LEMMA. *In a normal derivation each formula occurrence belongs to some track.*

PROOF. By induction on the height of normal derivations. For example, suppose a derivation K ends with an $\exists^-$-application:

$$\dfrac{\begin{array}{c} \\ \mid M \\ \exists x A \end{array} \qquad \begin{array}{c} [u\colon A] \\ \mid N \\ B \end{array}}{B}\ \exists^- x, u$$

B in N belongs to a track π (induction hypothesis); either this does not start in $u\colon A$, and then π, B is a track in K which ends in the conclusion; or π starts in $u\colon A$, and then there is a track π' in M (induction hypothesis) such that π', π, C is a track in K ending in the conclusion. The other cases are left to the reader. $\square$

THEOREM (Subformula property). *Let* M *be a normal derivation where every application of an* $\vee^-$*,* $\wedge^-$ *or* $\exists^-$*-rule discharges at least one assumption variable. Then each formula occurring in the derivation is a subformula of either the end formula or else an assumption formula.*

PROOF. As note above, each track of order 0 in M is a track ending in the conclusion of M. We can now prove the theorem for tracks of order n, by induction on n. $\square$

THEOREM (Disjunction property). *If Γ does not contain a disjunction as s.p.p. (= strictly positive part, defined in 1.3), then, if $\Gamma \vdash A \vee B$, it follows that $\Gamma \vdash A$ or $\Gamma \vdash B$.*

PROOF. Consider a normal derivation M of $A \vee B$ from assumptions Γ not containing a disjunction as s.p.p. The conclusion $A \vee B$ is the final formula of a (main) track, whose top formula A_0 in M must be an assumption in Γ. Since Γ does not contain a disjunction as s.p.p., the segment σ with the conclusion $A \vee B$ is in the I-part. Skip the final $\vee_i^+$-rule and replace the formulas in σ by A if $i = 0$, and by B if $i = 1$. $\square$

There is a similar theorem for the existential quantifier:

THEOREM (Explicit definability under hypotheses). *Let $\Gamma \vdash \exists x A$.*

(a) *If Γ does not contain an existential s.p.p., then there are terms r_1, r_2, $\dots$, r_n such that $\Gamma \vdash A[x := r_1] \vee \dots \vee A[x := r_n]$.*
(b) *If Γ neither contains a disjunctive s.p.p., nor an existential s.p.p., then there is a term r such that $\Gamma \vdash A[x := r]$.*

PROOF. Consider a normal derivation M of $\exists x A$ from assumptions Γ not containing an existential s.p.p. We use induction on the derivation, and distinguish cases on the last rule.

(a). By assumption the last rule cannot be $\exists^-$. We only consider the case $\vee^-$ and leave the others to the reader.

$$\dfrac{\begin{matrix} \\ \mid M \\ B \vee C \end{matrix} \qquad \begin{matrix} [u\colon B] \\ \mid N_0 \\ \exists x A \end{matrix} \qquad \begin{matrix} [v\colon C] \\ \mid N_1 \\ \exists x A \end{matrix}}{\exists x A}\ \vee^- u, v$$

By assumption again neither B nor C can have an existential s.p.p. Applying the induction hypothesis to N_0 and N_1 we obtain

$$\dfrac{\begin{matrix} \\ \mid M \\ B \vee C \end{matrix} \qquad \begin{matrix} [u\colon B] \\ \mid N_0 \\ \bigvee_{i=1}^{n} A[x := r_i] \end{matrix} \qquad \begin{matrix} [v\colon C] \\ \mid N_1 \\ \bigvee_{i=n+1}^{n+m} A[x := r_i] \end{matrix}}{\bigvee_{i=1}^{n+m} A[x := r_i]}\ \vee^- u, v$$

(b). Similarly; by assumption the last rule can be neither $\vee^-$ nor $\exists^-$. $\square$

REMARK. *Rasiowa-Harrop formulas* (in the literature also called *Harrop* formulas) are formulas for which no s.p.p. is a disjunction or an existential formula. For Γ consisting of Rasiowa-Harrop formulas both theorems above hold.

5. Notes

The proof of the existence of normal forms w.r.t permutative conversions is originally due to Prawitz [**14**]. We have adapted a method developed by Joachimski and Matthes [**8**], which in turn is based on van Raamsdonk's and Severi's [**18**].