Proof Theory of Finite-valued Logics

Diplomarbeit

ausgeführt am
Institut für Algebra und Diskrete Mathematik
der Technischen Universität Wien

unter Anleitung von
Univ.-Doz. Dr. phil. Matthias Baaz

durch

Richard Zach
Ada-Christen-Gasse 2/C/3, A-1100 Wien

Wien, am 21. September 1993

Preface

Many-valued logic is not much younger than the whole field of symbolic logic. It was introduced in the early twenties of this century by Łukasiewicz [1920] and Post [1921] and has since developed into a very large area of research. Most of the early work done has concentrated on problems of axiomatizability on the one hand, and algebraical/model theoretic investigations on the other. The proof theory of many-valued systems has not been investigated to any comparable extent. Proof theory requires appropriate formalisms, such as sequent calculus, natural deduction, and tableaux for classical (and intuitionistic) logic. Several people have, since the 1950's, proposed ways to generalize such formalisms from the classical to the many-valued case. One particular method for systematically obtaining calculi for all finite-valued logics was invented independently by several researchers, with slight variations in design and presentation. (Section 3.1 contains a short overview of work done in this area). The main aim of this report is to develop the proof theory of finite-valued first order logics in a general way, and to present some of the more important results in this area. This report is actually a template, from which all results can be specialized to particular logics. This idea has also found its way into the notation: Throughout this report, we use V as denoting a set of m truth values, and $\square$ and Q as dummies representing n-ary connectives and quantifiers, respectively. Replace V by the set of values *true* and *false*, and $\square$ and Q by the usual logical connectives and quantifiers, and you have a treatise on proof theoretic systems for classical logic.

Some of the material presented here has appeared in different forms elsewhere; The main innovations of this report are: the use of signed formula expressions and partial normal forms to provide a unifying framework in which clause translation calculi (Chapter 2), sequent calculi (Chapter 3), natural deduction (Chapter 4) and also tableaux can be represented; the recognition of a duality between two types—"negative" and "positive"—of resolution, sequents and tableaux (see also Baaz *et al.* [1993a]); bounds for partial normal forms for general (Section 1.6) and induced quantifiers (Section 1.7); and negative resolution (Section 2.6). The cut-elimination theorems extend previous results, the midsequent theorem is new, and Chapters 4 and 5 are entirely new. Material on refinements for resolution or on tableaux systems have purposely not been included. For that, see Baaz and Fermüller [1993] and Hähnle [1993a], respectively.

I will refrain from trying to give a thorough motivation for many-valued logic in general, or even an answer to the question of Scott [1976], "Does many-valued logic have any use?" The interested reader will find ample material in introductory texts to the topic, e.g., Rescher [1969], Urquhart [1986], Gottwald [1989], or Bolc and Borowik [1992]. I would, however, like to remark that in recent years many-valued logic has enjoyed a growth in interest from computer science, in such areas as Artificial Intelligence (see, e.g., Ginsberg [1988]) and circuit verification (see Hähnle and Kernig [1993]). The last chapter, on approximations, also tries to make a step in the direction of using many-valued logic—and the whole machinery of proof theoretic calculi and theorem proving systems—to approximate other logics, such as those originally devised for resoning applications.

Acknowledgements. I would like to extend my gratitude to the following people: my family, for their unconditional and continuing support; my teachers Matthias Baaz, Chris Fermüller, Georg Gottlob, and Alexander Leitsch; the *Dead Logicians Society*, in particular Angelika Gängl-Ehrenwert, Peter Koppensteiner, Arie Ovtrucki, and Helmut Veith; Reiner Hähnle and Michel Parigot for comments on parts of this thesis; Elizabeth Dehning for checking my English; and many others too innumerable to mention.

Contents

Chapter 1

Basic Concepts

1.1 Languages and Formulas

In mathematical logic—and especially in proof theory—the syntactic level is very important. The way in which something is formalized may have many consequences, particularly for the statement of theorems and the technicalities of proofs. Sometimes even the validity of a proof theoretic argument depends essentially on the underlying language. One important point in our notion of (first order) logical syntax is the *syntactical* distinction between free and bound variables. In this, we follow TAKEUTI [1987] and, ultimately, GENTZEN [1934].

1.1.1. DEFINITION A *first-order logical language* $\mathcal{L}$ consists of the following symbols:

(1) Variables:

(a) Free variables: a_0, a_1, a_2, ..., a_j, ... $(j \in \omega)$

(b) Bound variables: x_0, x_1, x_2, ..., a_j, ... $(j \in \omega)$

(2) Constants:

(a) Function symbols of arity i $(i \in \omega)$: f^i_0, f^i_1, f^i_2, ..., f^i_j, ... $(j \in \omega)$. Nullary function symbols are called *constants*.

(b) Predicate symbols of arity i $(i \in \omega)$: P^i_0, P^i_1, P^i_2, ..., P^i_j, ... $(j \in \omega)$

(3) Logical symbols:

(a) Propositional connectives of arity n_j: $\Box^{n_0}_0$, $\Box^{n_1}_1$, ..., $\Box^{n_r}_r$

(b) Quantifiers: Q_0, Q_1, ..., Q_q

(4) Auxiliary symbols: (,), and , (comma).

1.1.2. DEFINITION *Terms* and *semi-terms* are inductively defined as follows:

(1) Every individual constant is a term (semi-term).

(2) Every free variable (and every bound variable) is a term (semi-term).

(3) If f^n is a function symbol of arity n, and $t_1, \ldots, t_n$ are terms (semi-terms), then $f^n(t_1, \ldots, t_n)$ is a term (semi-term).

1.1.3. Definition *Formulas, semi-formulas,* and *outermost logical symbols* are inductively defined as follows:

(1) If P^n is a predicate symbol of arity n, and $t_1, \ldots, t_n$ are terms (semi-terms), then $P^n(t_1, \ldots, t_n)$ is a formula (semi-formula). It is called *atomic* or an *atom*. It has no outermost logical symbol.

(2) If $A_1, A_2, \ldots, A_n$ are formulas (semi-formulas) and $\Box^n$ is a propositional connective of arity n, then $\Box^n(A_1, A_2, \ldots, A_n)$ is a formula (semi-formula) with outermost logical symbol $\Box^n$.

(3) If A is a formula (semi-formula) not containing the bound variable x, a is a free variable and Q is a quantifier, then $(\mathsf{Q}x)A(x)$, where $A(x)$ is obtained from A by replacing a by x at every occurrence of a in A, is a formula (semi-formula). Its outermost logical symbol is Q.

A formula is called *open*, if it contains free variables, and *closed* otherwise. A formula without quantifiers is called *quantifier-free*. We denote the set of formulas of a language $\mathcal{L}$ by $\mathrm{Frm}(\mathcal{L})$.

Of course, in defining specific logics, we shall introduce specific symbols for the propositional connectives and for the quantifiers.

1.1.4. Example The language $\mathcal{L}$ normally used for classical 2-valued logic and also for other m-valued logics contains the following propositional connectives: $\neg$ (not), $\wedge$ (and), $\vee$ (or), $\supset$ (implies), $\equiv$ (equivalent) (of which $\neg$ is unary and all others binary), and the following two quantifiers: $\forall$ (for all) and $\exists$ (there exists).

1.1.5. Definition The complexity of a formula F, or its *degree* $\deg(F)$, is inductively defined as follows:

(1) If F is atomic, then $\deg(F) = 1$.

(2) If F is of the form $\Box^n(A_1, \ldots, A_n)$, then $\deg(F) = 1 + \sum_{i=1}^{n} \deg(A_i)$.

(3) If F is of the form $(\mathsf{Q}x)A(x)$, then $\deg(F) = 1 + \deg(A(x))$.

In the ensuing chapters, we will talk about the object-level constructs, e.g., formulas, and later also sequences and sets of those, on a meta-level. For instance, a calculus may be given by a set of axiom and rule *schemata*, and every instance of such a schema is an axiom and an inference, respectively. Hence, we shall need meta-variables for the symbols of a language $\mathcal{L}$: As a notational convention we use lowercase letters from the beginning of the alphabet (a, b, c, ...) to denote free variables, letters from the middle of the alphabet (f, g, h, ...) for function symbols and constants, letters from the end of the alphabet (x, y, z, ...) for bound variables, symbols like $\Box$ for connectives, and Q to denote quantifiers, all possibly indexed by subscripts. Furthermore, A, B, C, $\ldots$ will stand for formulas; Γ, Δ, Λ, $\ldots$ for sequences and sets of formulas, t and s for terms or semi-terms. We will write $A(x)$ for a semi-formula possibly containing the bound variable x, and $A(a)$ resp. $A(t)$ for the formula obtained from A by replacing every occurrence of the variable x by the free variable a resp. the term t. Sometimes it will be convenient to handle this meta-notation more explicitly, particularly in defining and working with inference rules. Then we use α as a variable for free variables (an *eigenvariable*), and τ as a variable for terms (a *term variable*). A formula consisting of some formula variables, eigenvariables and term variables is called a *schema*. By a *pre-instance* A' of a schema A we mean an actual formula from $\mathrm{Frm}(\mathcal{L})$ which contains occurrences of the eigenvariables and term variables of A. By an *instance* A'' of A we mean a pre-instance A' of A where the eigen- and term variables have been replaced by free variables not occurring in A', and terms, respectively. For instance, consider the schema $A(\alpha, \tau)$ where α is an eigenvariable, and τ is a term variable. $P_1^2(\alpha, a_1) \wedge P_2^1(\tau) \supset P_1^2(\tau, \alpha)$ is a pre-instance of $A(\alpha, \tau)$, and $P_1^2(a_0, a_1) \wedge P_2^1(f_0(a_1)) \supset P_1^2(f_0(a_1), a_0)$ is an instance.

1.2 Substitutions and Unification

Substitutions are of great importance in proof theory and automated theorem proving. The resolution calculus is based on substitutions and the so-called *unification* computation. Unification means finding a substitution, such that two expressions become equal after the substitution has been applied. In this section, and when we refer to these concepts later on, we will use *variable*, *term*, and *expression* more loosely than defined previously: a variable may be free or bound, a term can refer to a term or a semi-term, and an expression can refer to (semi-)terms, formulas, tuples, sequences and sets of formulas, and similar constructs.

1.2.1. DEFINITION A *substitution* is a mapping of variables to terms with finite domain. Substitutions are denoted by lowercase greek letters and are written as sets of expressions of the form t/x, where t is a (semi-)term and x is a variable.

The application of a substitution σ to an expression E is denoted by juxtaposition. $E\sigma$ denotes the expression obtained from E by *simultaneously* replacing every occurrence of a variable $x \in \mathrm{dom}(\sigma)$ in E by its image $\sigma(x)$.

The *composition* $\sigma\vartheta$ of two substitutions σ and ϑ is defined as follows: $\sigma\vartheta(x) = (\sigma(x))\vartheta$ for every variable x.

1.2.2. DEFINITION A substitution σ is called a *unifier* of a set A of expressions, if $A\sigma$ is a singleton set. The substitution σ is called a *most general unifier* of A, if every unifier ϑ of A is an extension of σ, i.e., there is a substitution λ such that $\vartheta = \sigma\lambda$.

For finite sets of expressions A, the problem of whether there exists a unifier for A is decidable. The unification algorithm (see CHANG and LEE [1973]) calculates the most general unifier if it exists.

1.3 Semantics of First Order Logics

The motivation to study many-valued logic does originally arise from semantical analysis of logical statements: One rejects the assumption that a statement must be either true or false: it may have some other "truth value". Different contexts have led to different interpretations for a third truth value, finitely or even infinitely many many truth values. This makes it necessary to introduce a basic set of semantic notations, although we shall almost exclusively deal with syntactical properties of finite-valued logics, and not with their model theories. We refer the interested reader to Rescher [1969] and Zinov'ev [1963] for more philosophical investigations into many-valued logic. A treatise on the model theory of many-valued logic can be found in Rasiowa [1974].

1.3.1. Definition A *matrix* $\mathbf{L}$ for a language $\mathcal{L}$ is given by:

(1) a nonempty set of *truth values* V of cardinality m,

(2) a subset $V^+ \subseteq V$ of *designated truth values*,

(3) an algebra $\mathcal{V}$ with domain V of appropriate type: For every n-place connective $\Box$ of $\mathcal{L}$ there is an associated *truth function* $\widetilde{\Box}\colon V^n \to V$, and

(4) for every quantifier Q, an associated truth function $\widetilde{\mathsf{Q}}\colon \wp(V) \setminus \{\emptyset\} \to V$

The set of truth values of the logics considered here will always be finite, i.e., m is a finite number. The constructions given for calculi for many-valued logics can be extended to logics with an infinite set of truth values, although the constructions will then, in general, not be effective. For special cases, a meaningful proof theory can still be developed.

Throughout this work we will give definitions and theorems relative to some given logic, and we will implicitly refer to this logic's language and matrix. In particular, V will always denote the set of truth values of the given logic, and m its cardinality. We shall also assume that V is equipped with a linear order, and write $V = \{v_1, v_2, \dots, v_m\}$ for the truth values in their given order. It is this order which we will refer to when we give the definition of a sequent.

The semantical structure of a logic is effectively determined by its matrix; we use boldface type to denote a logic (with its associated language) and its matrix. For instance, we use **CL** to refer to classical two-valued first order logic and $\mathbf{Ł}_m$ to refer to m-valued quantificational Łukasiewicz logic.

The intended meaning of a truth function for a propositional connective is analogous to the two-valued case: Given formulas $A_1, \dots, A_n$, which take the truth values $w_1, \dots, w_n$, respectively, the truth value of $\Box(A_1, \dots, A_n)$ is given by $\widetilde{\Box}(w_1, \dots, w_n)$.

A truth function for quantifiers is a mapping from nonempty sets of truth values to truth values: Given a quantified formula $(\mathsf{Q}x)F(x)$, such a set of truth values describes the situation where the ground instances of F take exactly the truth values in this set as values under a given interpretation. In other words, for a non-empty set $M \subseteq V$, $(\mathsf{Q}x)F(x)$ takes the truth value $\widetilde{\mathsf{Q}}(M)$ if, for every truth value $v \in V$, it holds that $v \in M$ iff there is a domain element d such that the truth value of $F(d)$ is v (all relative to some interpretation). The set M is called the *distribution* of F. This generalization of quantifiers dates back at least to MOSTOWSKI [1963]. Quantifiers of this type have been dubbed *distribution quantifiers* by CARNIELLI [1987b]. In CARNIELLI [1987a], he has also investigated the problem of *quantificational completeness*, i.e., the problem of which sets of quantifiers can, together with a set of propositional connectives, express *any* distribution quantifier.

1.3.2. EXAMPLE The matrix for classical two-valued logic **CL** is as follows:

(1) The set of truth values $V = \{f, t\}$ (false, true)

(2) The set of designated truth values $V^+ = \{t\}$

(3) The truth functions for classical two-valued connectives:

$\neg$	
f	t
t	f

$\wedge$	f	t
f	f	f
t	f	t

$\vee$	f	t
f	f	t
t	t	t

$\supset$	f	t
f	t	t
t	f	t

(4) The truth functions for the quantifiers $\forall$ and $\exists$:

$$\begin{array}{rclrcl} \widetilde{\forall}(\{t\}) & = & t & \widetilde{\forall}(\{t,f\}) & = & \widetilde{\forall}(\{f\}) = f \\ \widetilde{\exists}(\{f\}) & = & f & \widetilde{\exists}(\{t\}) & = & \widetilde{\exists}(\{t,f\}) = t \end{array}$$

1.3.3. EXAMPLE The matrix for the three-valued Łukasiweicz logic $\mathbf{Ł}_3$ consists of:

(1) The set of truth values $V = \{f, p, t\}$ (false, possible, true)

(2) The set of designated truth values $V^+ = \{t\}$

(3) The truth functions for the connectives:

$\neg$	
f	t
p	p
t	f

$\wedge$	f	p	t
f	f	f	f
p	f	p	p
t	f	p	t

$\vee$	f	p	t
f	f	p	t
p	p	p	t
t	t	t	t

$\supset$	f	p	t
f	t	t	t
p	p	t	t
t	f	p	t

(4) The truth functions for the quantifiers $\forall$ and $\exists$ (generalized $\wedge$ and $\vee$):

$$\begin{array}{rclrcl} \widetilde{\forall}(\{t\}) & = & t & \widetilde{\exists}(\{t\}) & = & t \\ \widetilde{\forall}(\{t,p\}) & = & p & \widetilde{\exists}(\{t,p\}) & = & t \\ \widetilde{\forall}(\{t,f\}) & = & f & \widetilde{\exists}(\{t,f\}) & = & t \\ \widetilde{\forall}(\{t,p,f\}) & = & f & \widetilde{\exists}(\{t,p,f\}) & = & t \\ \widetilde{\forall}(\{p\}) & = & p & \widetilde{\exists}(\{p\}) & = & p \\ \widetilde{\forall}(\{p,f\}) & = & f & \widetilde{\exists}(\{p,f\}) & = & p \\ \widetilde{\forall}(\{f\}) & = & f & \widetilde{\exists}(\{f\}) & = & f \end{array}$$

1.3.4. DEFINITION Let $\mathcal{L}$ be a first-order language, and let D be any set. The *extended language* $\mathcal{L}(D)$ is obtained by adjoining to the set of constant symbols of $\mathcal{L}$ the set D (D is assumed to be disjoint from $\mathcal{L}$).

1.3.5. DEFINITION A *structure* $\mathcal{M} = \langle D, \Phi \rangle$ for a language $\mathcal{L}$ (an $\mathcal{L}$-structure) consists of the following:

(1) A non-empty set D, called the *domain* (elements of D are called *individuals).*

(2) A mapping Φ that satisfies the following:

  (a) Each n-ary function symbol f of $\mathcal{L}(D)$ is mapped to a function $\widetilde{f}\colon D^n \to D$ if $n > 0$, or to an element of D if $n = 0$. If $d \in D$, then $\Phi(d) = d$.

  (b) Each n-ary predicate symbol P of $\mathcal{L}(D)$ is mapped to a function $\widetilde{P}\colon D^n \to V$ if $n > 0$, or to and element of V if $n = 0$.

1.3.6. DEFINITION Let $\mathcal{M}$ be an $\mathcal{L}$-structure. An *assignment* s is a mapping from the free variables of $\mathcal{L}$ to individuals.

1.3.7. DEFINITION An *interpretation* $\mathbf{I} = \langle \mathcal{M}, s \rangle$ is an $\mathcal{L}$-structure $\mathcal{M} = \langle D, \Phi \rangle$ together with an assignment s.

1.3.8. DEFINITION Let $\mathbf{I} = \langle \langle D, \Phi \rangle, s \rangle$ be an interpretation. The mapping Φ can be extended in the obvious way to a mapping $\Phi_{\mathbf{I}}$ from terms to individuals:

(1) If t is a free variable, then $\Phi_{\mathbf{I}}(t) = s(t)$.

(2) If t is of the form $f(t_1, \ldots, t_2)$, where f is a function symbol of arity n and $t_1, \ldots, t_n$ are terms, then $\Phi_{\mathbf{I}}(t) = \Phi(f)(\Phi_{\mathbf{I}}(t_1), \ldots, \Phi_{\mathbf{I}}(t_n))$.

1.3.9. DEFINITION Given an interpretation $\mathbf{I} = \langle \mathcal{M}, s \rangle$, we define the *valuation* $\mathrm{val}_{\mathbf{I}}$ to be a mapping from formulas F of $\mathcal{L}(D)$ to truth values as follows:

(1) If F is atomic, i.e., of the form $P(t_1, \ldots, t_n)$, where P is a predicate symbol of arity n and $t_1, \ldots, t_n$ are terms, then $\mathrm{val}_{\mathbf{I}}(F) = \Phi(P)(\Phi_{\mathbf{I}}(t_1), \ldots, \Phi_{\mathbf{I}}(t_n))$.

(2) If the outermost logical symbol of F is a propositional connective $\square$ of arity n, i.e., F is of the form $\square(F_1, \ldots, F_n)$, where $F_1, \ldots, F_n$ are formulas, then $\mathrm{val}_{\mathbf{I}}(F) = \widetilde{\square}(\mathrm{val}_{\mathbf{I}}(F_1), \ldots, \mathrm{val}_{\mathbf{I}}(F_n))$.

(3) If the outermost logical symbol of F is a quantifier Q, i.e., F is of the form $(\mathsf{Q}x)G(x)$, then

$$\mathrm{val}_{\mathbf{I}}(F) = \widetilde{\mathsf{Q}}\Big(\bigcup_{d \in D} \mathrm{val}_{\mathbf{I}} G(d)\Big)$$

Note that s maps variables to domain elements which are themselves constant symbols in the extended language. An assignment, if constrained to a finite number of variables, can then be viewed as a ground substitution. For a formula F it obviously does not make a difference if val is applied to F or to Fs.

1.3.10. DEFINITION Let $\mathbf{I} = \langle \mathcal{M}, s \rangle$ be an interpretation. An interpretation $\mathbf{I}' = \langle \mathcal{M}, s' \rangle$ is called a *variant of* $\mathbf{I}$ *modulo* $a_1, \ldots, a_n$, in symbols $\mathbf{I} \sim_{a_1,\ldots,a_n} \mathbf{I}'$, iff s equals s' except in the values of the free variables $a_1, \ldots, a_n$.

1.3.11. PROPOSITION *Let A be a formula, $\mathbf{I} = \langle \mathcal{M}, s \rangle$ be an interpretation, and $s(a) = d \in D$ for the free variable a. Then* $\mathrm{val}_{\mathbf{I}}(A) = \mathrm{val}_{\mathbf{I}'}(A[d/a])$, *for all* $\mathbf{I}' \sim_a \mathbf{I}$.

1.3.12. DEFINITION A formula F is called a *many-valued tautology* iff, for every interpretation $\mathbf{I}$, it holds that $\mathrm{val}_{\mathbf{I}}(F) \in V^+$. It is called *satisfiable* iff there is an interpretation $\mathbf{I}$ s.t. $\mathrm{val}_{\mathbf{I}}(F) \in V^+$, and *unsatisfiable* otherwise.

Many-valued tautologies play an important rôle in the area of Hilbert-style axiomatizations of many-valued logics. We will not deal with Hilbert-style axiomatizations nor with many-valued tautologies here, and hence the set V^+ will not be very important in our considerations. We will, however, point out how to incorporate tautologies into our frameworks, e.g., in terms of provability.

1.4 Signed Formula Expressions

It is a well-known and not too surprising fact that the truth of many-valued formulas can be reduced, in a sense, to the truth of formulas in two-valued classical logic. In the words of ROSSER and TURQUETTE [1952] (p. 1), two-valued logic is *sufficient* for the development of many-valued logic. The idea behind this "two-valisation" of many-valued logics is as follows: We introduce m two-valued predicates $A^{v_1}, \ldots, A^{v_m}$ for every many-valued atom A. These predicates are called *signed atomic formulas*: A^w expresses that A takes the truth value w. This notation, and its corresponding theory, will serve as a convenient formalism for specifying many-valued logics. In fact, the truth table specification can easily be translated into a specification by means of *signed formula expressions*, and it is this specification from which we will be able to obtain calculi for the logics under consideration in a completely systematic and mechanizable way.

We will use signed formula expressions (sfes) to give meaning to syntactical constructs such as clauses and sequents, which will all have corresponding sfes of a particular form. Satisfaction, validity, etc. of, e.g., a sequent or a clause will be defined via satisfaction etc. of the corresponding sfes. On the other hand, sfe schemata provide a convenient means for specifying calculi by recurring on the semantics of sfes. Sfes of a certain kind will, e.g., be used to specify sequent calculi *rules*.

1.4.1. Definition A *signed formula* is an expression of the form A^{v_i}, where v_i is a truth value, and A is a first order formula. A *signed formula expression* is a formula built up from signed formulas using $\curlywedge$, $\curlyvee$, $\neg$.

A signed formula expression of the form A^w or $\neg A^w$ is called a *signed literal*. It is called an *atomic literal* iff A is atomic. In that case, A^w is also called a *signed atom*.

Signed formula expressions are Boolean expressions in the signed formulas, and every interpretation defines a Boolean truth value assignment to the signed formulas, and to the expression as a whole. We call an sfe Δ *valid* iff Δ is true under every interpretation via the induced valuation, and *satisfiable* iff there is an interpretation s.t. Δ is true in the induced valuation. Furthermore, assume that $\neg$ occurs in Δ only immediately in front of signed formulas, if at all. Then Δ is called *positive* iff all its signed literals are of the form A^{v_i} (i.e., $\neg$ does not occur at all), and *negative* iff they are all of the form $\neg A^{v_i}$.

We shall use the following abbreviations: If $A_1, \ldots, A_n$ are sfes, then $\bigvee_{i=1}^n A_i$ stands for $A_1 \curlyvee (A_2 \curlyvee \ldots (A_{n-1} \curlyvee A_n) \cdots)$, and $\bigwedge_{i=1}^n$ for $A_1 \curlywedge (A_2 \curlywedge \ldots (A_{n-1} \curlywedge A_n) \cdots)$. If $n = 0$, we obtain the empty disjunction and the empty conjunction, respectively. By convention, these represent sfes which are always false or always true, respectively. If they occur in the context of other disjunctions or conjunctions, respectively, they may be deleted in the obvious way. Furthermore, if $W \subseteq V$, then A^W denotes $\bigvee_{w \in W} A^w$. The usual conventions for omitting parentheses apply: binary operators associate to the right, and precedence is given in the decreasing order $\neg$, $\curlywedge$, $\curlyvee$. Furthermore, it is easy to see that the well-known equivalences for classical logic also hold for sfes. So, e.g., associative, commutative, distributive and de Morgan's laws hold for sfes.

1.4.2. Definition Let $\mathbf{I}$ be an interpretation and F be a signed formula expression. We say $\mathbf{I}$ *satisfies* F, in symbols: $\mathbf{I} \models F$, iff

(1) F is of the form A^w, where A is an atomic formula, and $\mathrm{val}_{\mathbf{I}}(A) = w$;

(2) F is of the form $\neg A$ and $\mathbf{I} \not\models A$;

(3) F is of the form $A \curlywedge B$ and $\mathbf{I} \models A$ and $\mathbf{I} \models B$;

(4) F is of the form $A \curlyvee B$ and $\mathbf{I} \models A$ or $\mathbf{I} \models B$;

Furthermore, F is called *valid*, in symbols $\models F$, iff every interpretation satisfies F; it is called *satisfiable* iff there is an interpretation $\mathbf{I}$ s.t. $\mathbf{I} \models F$, and *unsatisfiable* otherwise.

1.4.3. Proposition *Let F be a formula. Then the following are equivalent:*

(1) F is a tautology.

(2) The sfe $\bigvee_{w \in V^+} F^w$ is valid.

(3) The sfe $\bigwedge_{w \in V \setminus V^+} \neg F^w$ is valid.

(4) The sfe $\bigvee_{w \in V \setminus V^+} F^w$ is unsatisfiable.

(5) *The sfe* $\bigwedge_{w \in V^+} \neg F^w$ *is unsatisfiable.*

1.4.4. PROPOSITION *Let* F *be a formula. Then the following are equivalent:*

(1) F *is unsatisfiable.*

(2) *The sfe* $\bigvee_{w \in V^+} F^w$ *is unsatisfiable.*

(3) *The sfe* $\bigwedge_{w \in V \setminus V^+} \neg F^w$ *is unsatisfiable.*

(4) *The sfe* $\bigvee_{w \in V \setminus V^+} F^w$ *is valid.*

(5) *The sfe* $\bigwedge_{w \in V^+} \neg F^w$ *is valid.*

1.4.5. PROPOSITION *Every sfe* Δ *can be transformed to an equivalent positive sfe* $\mathrm{p}(\Delta)$ *(negative sfe* $\mathrm{n}(\Delta)$*).*

Proof. Use de Morgan's laws to bring $\neg$ immediately before the signed formulas. Replace every signed literal $\neg A^{v_i}$ (A^{v_i}) in Δ by $\bigvee_{j \neq i} A^{v_j}$ ($\bigwedge_{j \neq i} \neg A^{v_j}$). ■

1.4.6. PROPOSITION *Every sfe* Δ *containing negation only immediately in front of signed formulas can be transformed to an equivalent positive (negative) conjunction of disjunctions of signed formulas (a conjunctive form of* Δ*).*

Proof. We prove the proposition by induction on the complexity of Δ: If Δ is a signed atom A^{v_i}, then A^{v_i} is a positive and $\bigwedge_{j \neq i} \neg A^{v_j}$ a negative conjunction of disjunctions equivalent to Δ. If Δ is of the form $\Psi \wedge \Phi$, then by induction hypothesis, Ψ' is a conjunctive form of Ψ and Φ' is a conjunctive form of Φ: $\Psi' \wedge \Phi'$ is a conjunctive form of Δ. If Δ is of the form $\Psi \vee \Phi$, then let $\Psi = \bigwedge_{i=1}^{r} \bigvee_{j=1}^{k_i} A_{i,j}$ and $\Phi = \bigwedge_{i'=1}^{s} \bigvee_{j=1}^{l_{i'}} B_{i',j}$. We obtain a conjunctive form of Δ by:

$$\bigwedge_{i=1}^{r} \bigwedge_{i'=1}^{s} \left(\bigvee_{j=1}^{k_i} A_{i,j} \vee \bigvee_{j=1}^{l_{i'}} B_{i',j}\right) \quad ■$$

1.4.7. PROPOSITION $\mathbf{I} \models \Delta$ *iff* $\mathbf{I} \not\models \neg\Delta$.

1.4.8. PROPOSITION $\models \Delta$ *iff* $\neg\Delta$ *is unsatisfiable.*

1.5 Partial Normal Forms

Partial normal forms were introduced by ROSSER and TURQUETTE [1952], p. 53ff. as a convenient means of specifying the truth-functional behaviour of many-valued connectives and quantifiers. For a given connective $\Box$, the i-th partial normal form is a signed formula expression schema F_i equivalent to $\Box(A_1, \ldots, A_n)$ which contains only the (signed) formulas $A_1, \ldots, A_n$, and is in conjunctive normal form[1]. Partial normal forms are *exhaustive* and *mutually*

[1] ROSSER and TURQUETTE [1952] used disjunctive normal forms.

exclusive, i.e., under any given interpretation exactly one of the m partial normal forms comes out true. They can easily be constructed from the truth tables of a given logic, also for the quantifiers of that logic, and in essence provide us directly with the rules to be used in our calculi.

1.5.1. DEFINITION Let $\Box$ be a propositional connective of arity n. An sfe schema F is called an *i-th partial form* of $\Box(A_1, \ldots, A_n)$ if the following hold:

(1) The signed formulas in F are of the form ${A_j}^w$ $(1 \leq j \leq n,\ w \in V)$.

(2) For every instance F' of F and every interpretation $\mathbf{I}$ and it holds that $\mathbf{I} \models F'$ iff $\mathrm{val}_{\mathbf{I}}(\Box(A_1, \ldots, A_n)) = v_i$.

It is called a *partial normal form*, if it is in conjunctive normal form.

An i-th partial normal form for $\neg\Box(A_1, \ldots, A_n)$ is defined as above, with $=$ replaced by $\neq$.

Thus, in other words, an i-th partial normal form for $\Box(A_1, \ldots, A_n)$ (or $\neg\Box(A_1, \ldots, A_n)$) is a conjunctive normal form in the ${A_j}^w$ which is equivalent to $\Box(A_1, \ldots, A_n)^{v_i}$ (or $\neg\Box(A_1, \ldots, A_n)^{v_i}$). Note that we are actually talking about schemata (our meta-notation of § 1.1) as *objects*, rather than using them to specify a class of, e.g., sfes. A partial form is a schema, not its collection of instances. This will be important in later chapters, as partial forms are used to define rules of inference in sequent calculus and natural deduction, which are rule *schemata* themselves. It should be noted again that from *one* partial normal form we will obtain a clause translation rule, a sequent calculus introduction rule, and a natural deduction rule (for a connective and a place). In this sense, the pnfs provide a relationship between all these calculi.

1.5.2. EXAMPLE Positive partial normal forms for the connectives of classical two-valued logic **CL**:

$$\begin{array}{rclrcl}
(\neg A)^f & = & A^t & (\neg A)^t & = & A^f \\
(A \vee B)^f & = & A^f \mathbin{\curlywedge\!\!\curlywedge} B^f & (A \vee B)^t & = & A^t \mathbin{\curlyvee\!\!\curlyvee} B^t \\
(A \wedge B)^f & = & A^f \mathbin{\curlyvee\!\!\curlyvee} B^f & (A \wedge B)^t & = & A^t \mathbin{\curlywedge\!\!\curlywedge} B^t \\
(A \supset B)^f & = & A^t \mathbin{\curlywedge\!\!\curlywedge} B^f & (A \supset B)^t & = & A^f \mathbin{\curlyvee\!\!\curlyvee} B^t
\end{array}$$

Negative pnfs for their negations are given by:

$$\begin{array}{rclrcl}
\neg(\neg A)^f & = & \neg A^t & \neg(\neg A)^t & = & \neg A^f \\
\neg(A \vee B)^f & = & \neg A^f \mathbin{\curlyvee\!\!\curlyvee} \neg B^f & \neg(A \vee B)^t & = & \neg A^t \mathbin{\curlywedge\!\!\curlywedge} \neg B^t \\
\neg(A \wedge B)^f & = & \neg A^f \mathbin{\curlywedge\!\!\curlywedge} \neg B^f & \neg(A \wedge B)^t & = & \neg A^t \mathbin{\curlyvee\!\!\curlyvee} \neg B^t \\
\neg(A \supset B)^f & = & \neg A^t \mathbin{\curlyvee\!\!\curlyvee} \neg B^f & \neg(A \supset B)^t & = & \neg A^f \mathbin{\curlywedge\!\!\curlywedge} \neg B^t
\end{array}$$

1.5.3. EXAMPLE Partial normal forms for the connectives of three-valued Łukasiewicz logic $\mathbf{Ł}_3$ are as follows:

$$\begin{array}{rcl}
(\neg A)^f &=& A^t \\
(\neg A)^p &=& A^p \\
(\neg A)^t &=& A^f \\
(A \vee B)^f &=& A^f \wedge\!\!\!\wedge B^f \\
(A \vee B)^p &=& (A^p \vee\!\!\!\vee B^p) \wedge\!\!\!\wedge (A^f \vee\!\!\!\vee A^p) \wedge\!\!\!\wedge (B^f \vee\!\!\!\vee B^p) \\
(A \vee B)^t &=& A^t \vee\!\!\!\vee B^t \\
(A \wedge B)^f &=& A^f \vee\!\!\!\vee B^f \\
(A \wedge B)^p &=& (A^p \vee\!\!\!\vee B^p) \wedge\!\!\!\wedge (A^p \vee\!\!\!\vee A^t) \wedge\!\!\!\wedge (B^p \vee\!\!\!\vee B^t) \\
(A \wedge B)^t &=& A^t \wedge\!\!\!\wedge B^t \\
(A \supset B)^f &=& A^t \wedge\!\!\!\wedge B^f \\
(A \supset B)^p &=& (A^p \vee\!\!\!\vee A^t) \wedge\!\!\!\wedge (A^p \vee\!\!\!\vee B^p) \wedge\!\!\!\wedge (A^t \vee\!\!\!\vee B^f) \\
\text{or, } (A \supset B)^p &=& (A^p \vee\!\!\!\vee B^p) \wedge\!\!\!\wedge (A^t \vee\!\!\!\vee B^f) \wedge\!\!\!\wedge (B^f \vee\!\!\!\vee B^p) \\
(A \supset B)^t &=& (A^f \vee\!\!\!\vee A^p \vee\!\!\!\vee B^t) \wedge\!\!\!\wedge (A^f \vee\!\!\!\vee B^p \vee\!\!\!\vee B^t)
\end{array}$$

Negative pnfs for the negations are:

$$\begin{array}{rcl}
\neg(\neg A)^f &=& \neg A^t \\
\neg(\neg A)^p &=& \neg A^p \\
\neg(\neg A)^t &=& \neg A^f \\
\neg(A \vee B)^f &=& \neg A^f \vee\!\!\!\vee \neg B^f \\
\neg(A \vee B)^p &=& (\neg A^p \vee\!\!\!\vee \neg B^p) \wedge\!\!\!\wedge (\neg A^p \vee\!\!\!\vee \neg B^f) \wedge\!\!\!\wedge (\neg A^f \vee\!\!\!\vee \neg B^p) \\
\neg(A \vee B)^t &=& \neg A^t \wedge\!\!\!\wedge \neg B^t \\
\neg(A \wedge B)^f &=& \neg A^f \wedge\!\!\!\wedge \neg B^f \\
\neg(A \wedge B)^p &=& (\neg A^p \vee\!\!\!\vee \neg B^p) \wedge\!\!\!\wedge (\neg A^p \vee\!\!\!\vee \neg B^t) \wedge\!\!\!\wedge (\neg A^t \vee\!\!\!\vee \neg B^p) \\
\neg(A \wedge B)^t &=& \neg A^t \vee\!\!\!\vee \neg B^t \\
\neg(A \supset B)^f &=& \neg A^t \vee\!\!\!\vee \neg B^f \\
\neg(A \supset B)^p &=& (\neg A^p \vee\!\!\!\vee \neg B^f) \wedge\!\!\!\wedge (\neg A^t \vee\!\!\!\vee \neg B^p) \\
\neg(A \supset B)^t &=& \neg A^f \wedge\!\!\!\wedge \neg B^t \wedge\!\!\!\wedge (\neg A^p \vee\!\!\!\vee \neg B^p)
\end{array}$$

We can immediately extract a partial normal form from the truth table in a way analogous to the method of obtaining complete conjunctive normal forms for two-valued functions: The idea is to look at the n-tuples of values for which $\Box$ *does not* take the required truth value v_i, to describe the negations of these situations and to combine these descriptions conjunctively. More precisely, let

$$I = \{(w_1, \ldots, w_n) \mid \widetilde{\Box}(w_1, \ldots, w_n) \neq v_i\}.$$

Then $\Box(A_1, \ldots, A_n)^{v_i} =$

$$\bigwedge\!\!\!\!\bigwedge_{(w_1,\ldots,w_n)\in I} (\neg {A_1}^{w_1} \vee\!\!\!\vee \neg {A_2}^{w_2} \vee\!\!\!\vee \ldots \vee\!\!\!\vee \neg {A_n}^{w_n})$$

is a negative i-th partial normal form for $\Box$. We can obtain a positive partial normal form by replacing each $\neg A^w$ by $\bigvee\!\!\!\!\bigvee_{u\in V\setminus\{w\}} A^u$.

1.5.4. Example Consider $\wedge$ in **CL** for the truth value t: Here,

$$I = \{(f, f), (f, t), (t, f)\}.$$

The corresponding negative partial normal form is:

$$(A \wedge B)^t = (\neg A^f \vee \neg B^f) \wedge (\neg A^f \vee \neg B^t) \wedge (\neg A^t \vee \neg B^f).$$

We obtain a positive normal form

$$(A \wedge B)^t = (A^t \vee B^t) \wedge (A^t \vee B^f) \wedge (A^f \vee B^t).$$

A partial normal form constructed this way can have up to m^n conjuncts (if $\Box$ never takes the value v_i), but standard methods for minimizing combinational function, such as the Quine-McCluskey procedure, can be used to find minimal positive or negative normal forms. Here, "minimal" is meant with respect to the number of conjuncts and the number of formulas per conjunct. For instance, the partial normal form for $A \wedge B^t$ given in Example 1.5.2 is minimal, compare this to the one obtained in Example 1.5.4. Nevertheless, there are connectives which admit no "simple" rule. This will be shown in the next section. An adaptation of the Quine-McCluskey procedure has been implemented and gives minimal positive pnfs in reasonable time for small m (see Baaz *et al.* [1993]).

If a connective always takes the truth value v_i, then the i-th partial normal form is the empty conjunction, which is always true.

1.5.5. Definition Let Q be a quantifier. An sfe schema F is called an *i-th partial form* of $(\mathsf{Q}x)A(x)$ if the following hold:

(1) The signed atoms in F are among $\{A(\tau_j)^{u_{ji}} \mid 1 \le j \le p, 1 \le i \le m\} \cup \{A(\alpha_j)^{w_{ji}} \mid 1 \le j \le q, 1 \le i \le m\}$ where the α_i are eigenvariables and the τ_i term variables.

(2) For every pre-instance F' of F and every interpretation $\mathbf{I}$ it holds that

(a) If for all $d_1, \ldots, d_q \in D$ there are $e_1, \ldots, e_p \in D$ s.t.

$$\mathbf{I} \models F\{e_1/\tau_1, \ldots, e_p/\tau_p, d_1/\alpha_1, \ldots, d_q/\alpha_q\}$$

then $\mathrm{val}_{\mathbf{I}}((\mathsf{Q}x)A'(x)) = v_i$.

(b) If for all $e_1, \ldots, e_q \in D$ there are $d_1, \ldots, d_p \in D$ s.t.

$$\mathbf{I} \not\models F\{e_1/\tau_1, \ldots, e_p/\tau_p, d_1/\alpha_1, \ldots, d_q/\alpha_q\}$$

then $\mathrm{val}_{\mathbf{I}}((\mathsf{Q}x)A'(x)) \neq v_i$.

where A' is the instance of A determined by F'.

It is called a *partial normal form* iff it is in conjunctive normal form.

An i-th partial normal form for $\neg(\mathsf{Q}x)A(x)$ is defined as above, with $=$ replaced by $\neq$.

Term variables and eigenvariables express universal and existential conditions on formulas, in much the same way as the universal and existential quantifiers do in classical logic. It is therefore convenient to introduce quantifiers in sfes as a notational convention. The expression $(\mathbb{\forall}\, x)A(x)$ is then meant to stand for the sfe resulting from $A(x)$ by replacing x with some eigenvariable α, and similarly, $(\mathbb{\exists}\, x)A(x)$ for $A(\tau)$. Different quantifiers correspond to different eigenvariables/term variables. It is easy to prove the familiar equivalences (distribution of $\mathbb{\exists}$ over $\mathbb{\vee}$, of $\mathbb{\forall}$ over $\mathbb{\wedge}$, de Morgan's laws) for these quantifiers, allowing us to work with them just as in classical logic. For instance, if $(\mathbb{\forall}\, x)(A(x) \mathbin{\mathbb{\wedge}} B(x))$ then $(\mathbb{\forall}\, x)A(x) \mathbin{\mathbb{\wedge}} (\mathbb{\forall}\, x)B(x)$. This is seen as follows: The first expression stands for $F = A(\alpha) \mathbin{\mathbb{\wedge}} B(\alpha)$, the second for $G = A(\alpha) \mathbin{\mathbb{\wedge}} B(\beta)$. Let $\mathbf{I}$ be an interpretation. Now if for each assignment of $d \in D$ for α, $\mathbf{I} \models A(d/\alpha) \mathbin{\mathbb{\wedge}} B(d/\alpha)$, then also $\mathbf{I} \models A(d/\alpha)$ and $\mathbf{I} \models A(d/\alpha)$ for all $d \in D$ separately. We may rename one α and obtain $\mathbf{I} \models A(d/\alpha)$ for all $d \in D$ and $\mathbf{I} \models A(d/\beta)$ for all $d \in D$. But this means that G is true in $\mathbf{I}$.

1.5.6. EXAMPLE Partial normal forms for the quantifiers of **CL** are as follows:

$$\begin{array}{rclrcl} ((\forall x)A(x))^f & = & A(\tau)^f & ((\forall x)A(x))^t & = & A(\alpha)^t \\ ((\exists x)A(x))^f & = & A(\alpha)^f & ((\exists x)A(x))^t & = & A(\tau)^t \end{array}$$

Negative pnfs for their negations are given by:

$$\begin{array}{rclrcl} \neg((\forall x)A(x))^f & = & \neg A(\alpha)^f & \neg((\forall x)A(x))^t & = & \neg A(\tau)^t \\ \neg((\exists x)A(x))^f & = & \neg A(\tau)^f & \neg((\exists x)A(x))^t & = & \neg A(\alpha)^t \end{array}$$

1.5.7. EXAMPLE Partial normal forms for the quantifiers of $\mathbf{Ł}_3$ are as follows:

$$\begin{array}{c} ((\forall x)A(x))^f = A(\tau)^f \qquad ((\forall x)A(x))^t = A(\alpha)^t \\ ((\forall x)A(x))^p = A(\tau)^p \mathbin{\mathbb{\wedge}} (A(\alpha)^p \mathbin{\mathbb{\vee}} A(\alpha)^t) \\ ((\exists x)A(x))^f = A(\alpha)^f \qquad ((\exists x)A(x))^t = A(\tau)^t \\ ((\exists x)A(x))^p = A(\tau)^p \mathbin{\mathbb{\wedge}} (A(\alpha)^f \mathbin{\mathbb{\vee}} A(\alpha)^p) \end{array}$$

Negative pnfs for their negations are given by:

$$\begin{array}{c} \neg((\forall x)A(x))^f = \neg A(\alpha)^f \qquad \neg((\forall x)A(x))^t = \neg A(\tau)^t \\ \neg((\forall x)A(x))^p = \neg A(\tau)^p \mathbin{\mathbb{\wedge}} (\neg A(\alpha)^p \mathbin{\mathbb{\vee}} A(\tau)^t) \\ \neg((\exists x)A(x))^f = \neg A(\tau)^f \qquad \neg((\exists x)A(x))^t = \neg A(\alpha)^t \\ \neg((\exists x)A(x))^p = (A(\alpha)^p \mathbin{\mathbb{\vee}} A(\tau)^f) \mathbin{\mathbb{\wedge}} \neg A(\tau)^p \end{array}$$

As in the case of the propositional connectives, truth tables for quantifiers yield partial normal forms. Let $\mathrm{Distr}_{\mathbf{I}}(A(x)) = \{\mathrm{val}_{\mathbf{I}}(A(d)) \mid d \in D\}$, the so called *distribution* of $A(x)$ relative to $\mathbf{I}$. The truth table for a quantifier Q maps distributions to truth values. To describe the conditions for $(\mathsf{Q}x)A(x)$ to take the truth value v_i we proceed analogously to the case of a propositional connective: There, we looked at those combinations of truth values $w_1, \ldots, w_n$ for which $\widetilde{\Box}(w_1, \ldots, w_n)$ is different from the value v_i and described these situations. The conjunctions over the negations of these descriptions yields the required conjunctive partial normal form. Accordingly, let $W \subset V$ be a distribution s.t. $\widetilde{\mathsf{Q}}(W) \neq v_i$. For any particular interpretation $\mathbf{I}$ we have $\mathrm{Distr}_{\mathbf{I}}(A(x)) \neq W$ iff either there is some $d \in D$ s.t. $\mathbf{I} \models A(d)^{V \setminus W}$ (i.e., $\mathrm{Distr}_{\mathbf{I}}(A(x))$ contains a truth value not in W), or there is a truth value $w \in W$ s.t. for all $e \in D$, $\mathbf{I} \models A(e)^{V \setminus \{w\}}$ (i.e., there is a truth value in W which is never taken by $A(x)$ under $\mathbf{I}$). The conjunction of these conditions over all W s.t. $\widetilde{\mathsf{Q}}(W) \neq v_i$ fully describes $\widetilde{\mathsf{Q}}(W) = v_i$. Let

$$I = \{W \subseteq V \setminus \emptyset \mid \widetilde{\mathsf{Q}}(W) \neq v_i\}$$

The sfe expressing $\widetilde{\mathsf{Q}}(W) = v_i$ is as follows

$$\bigwedge_{W \in I} ((\exists x \in D) A(x)^{V \setminus W} \vee \bigvee_{w \in W} (\forall y) A(y)^{V \setminus \{w\}})$$

Note that the quantifiers can be brought to the front in the order required by Definition 1.5.5. The corresponding sfe is as follows:

$$\bigwedge_{W \in I} \left(\bigvee_{u \in V \setminus W} A(\tau)^u \vee \bigvee_{w \in W} \bigvee_{u \in V \setminus \{w\}} A(\alpha_w)^u \right)$$

We have an upper bound on the number of conjuncts of $2^m - 1$: This is the case if $\widetilde{\mathsf{Q}}$ never takes the value v_i.

If a quantifier never takes a particular truth value v_i, then the i-th partial normal form is the empty conjunction, which is always true.

1.5.8. EXAMPLE Consider the universal quantifier in **CL** at place t. We have $I = \{\{f\}, \{f, t\}\}$. The corresponding pnf is

$$((\forall x) A(x))^t = (A(\tau)^t \vee A(\alpha)^t) \wedge (A(\alpha_t)^f \vee A(\alpha_f)^t)$$

The second conjunct says that $A(x)$ is either false for all x or true for all x. The first expresses that there is an element e, such that $A(e)$ is true (or for all elements, $A(e)$ is true), hence the second alternative applies: $A(x)$ is uniformly true.

1.5.9. *Remark* ROSSER and TURQUETTE [1952] use a very general quantifier definition where a quantifier may take several variables and formulas: A quantified formula then has the form

$$(\mathsf{Q} x_1, \ldots, x_k)(A_1(x_1, \ldots, x_k), \ldots, A_l(x_1, \ldots, x_k)).$$

The case where $k = l = 1$ is that of our distribution quantifiers.

1.6 Bounds for Partial Normal Forms

In the last section we gave examples of pnfs for n-ary connectives and quantifiers having at most m^n and $2^m - 1$ conjuncts, respectively. In this section we show how this can be improved to pnfs of at most m^{n-1} and 2^{m-1} conjuncts, respectively. It is also shown that these bounds are tight.

The following proposition occurs as Lemma 1 in ROUSSEAU [1967].

1.6.1. PROPOSITION *Let $\Box$ be an n-ary propositional connective. There is an i-th positive partial normal form for $\Box(A_1, \ldots, A_n)$ of at most m^{n-1} conjuncts.*

Proof. Consider the following disjunctive form:

$$\bigvee_{(w_1,\ldots,w_{n-1})} ({A_1}^{w_1} \wedge \ldots \wedge {A_{n-1}}^{w_{n-1}} \wedge \bigvee_{\substack{w_n \\ \widetilde{\Box}(w_1,\ldots,w_{n-1},w_n) \neq v_i}} {A_n}^{w})$$

This sfe is equivalent to $\neg\Box(A_1, \ldots, A_n)^{v_i}$: Suppose that ${A_i}^{w_i}$ holds for $1 \leq i \leq n$) and that $\widetilde{\Box}(w_1, \ldots, w_{n-1}, w_n) \neq v_i$. But then the first part of the disjunct corresponding to $(w_1, \ldots, w_{n-1})$ is true, and the second part is true since ${A_n}^{w_n}$ is true. Conversely, assume that the disjunct corresponding to $(w_1, \ldots, w_{n-1})$ is true. Then ${A_i}^{w_i}$ holds for $1 \leq i \leq n-1$. Furthermore, from the second part of the disjunct, there is a w_n s.t. ${A_n}^{w_n}$ holds and that $\widetilde{\Box}(w_1, \ldots, w_{n-1}, w_n) \neq v_i$. But this means that $\neg\Box(A_1, \ldots, A_n)^{v_i}$.

By negating the whole sfe we obtain the following i-th partial normal form for $\Box(A_1, \ldots, A_n)$:

$$\bigwedge_{(w_1,\ldots,w_{n-1})} (\bigvee_{u \neq w_1} {A_1}^{u} \vee \ldots \vee \bigvee_{u \neq w_{n-1}} {A_{n-1}}^{u} \vee \bigvee_{\substack{w_n \\ \widetilde{\Box}(w_1,\ldots,w_{n-1},w_n) = v_i}} {A_n}^{w_n})$$

This partial normal form has at most m^{n-1} conjuncts. ∎

ROUSSEAU [1967] also stated that the bound given above is tight, although without detailed proof:

1.6.2. PROPOSITION *For every n, there is a propositional connective $\odot$ of arity n s.t. any i-th partial normal form for $\odot$ contains m^{n-1} conjuncts.*

Proof. Without loss of generality assume that the set of truth values $V = \{0, 1, \ldots, m-1\}$, and let $u = i - 1$. Furthermore, let $u' \in V$ be a truth value s.t. $u' \neq u$. Define $\odot$ as follows:

$$\widetilde{\odot}(i_1, \ldots, i_n) = \begin{cases} u' & i_1 + \cdots + i_n \equiv 0(m) \\ u & \text{otherwise} \end{cases}$$

Obviously, $\neg\odot(A_1, \ldots, A_n)^{v_i}$ holds iff A_j takes the truth value i_j $(1 \leq j \leq n)$ and $i_1 + \cdots + i_n \equiv 0(m)$.

We show that any given minimal normal form for $\odot$ has m^{n-1} conjuncts: Assume that

$$F = \bigvee_{(W_1,\ldots,W_n)} (\bigvee_{w_1 \in W_1} {A_1}^{w_1} \wedge \ldots \wedge \bigvee_{w_n \in W_n} {A_n}^{w_n})$$

is equivalent to $\neg\odot(A_1, \ldots, A_n)^{u}$ and is minimal in the number of disjuncts. This is the case iff

$$\bigwedge_{(W_1,\ldots,W_n)} (\bigvee_{w_1 \in V \setminus W_1} {A_1}^{w_1} \vee \ldots \vee \bigvee_{w_n \in V \setminus W_n} {A_n}^{w_n})$$

is a minimal conjunctive normal form for $\odot(A_1, \ldots, A_n)^u$.

Consider an interpretation $\mathbf{I}$ making the form for $\neg\odot(\cdot)^u$ true, i.e. there is a tuple $(W_1, \ldots, W_n)$ such that the corresponding disjunct is true. We show that $|W_j| = 1$ for all W_j: Assume $W_1 \supseteq \{k, l\}$ and that $\mathrm{val}_{\mathbf{I}}(A_1) = k$. Then the interpretation $\mathbf{I}'$ which is equal to $\mathbf{I}$ except that $\mathrm{val}_{\mathbf{I}}(A_1) = l$ also satisfies F, and thus $\neg\odot(\cdot)^u$. If $\mathrm{val}_{\mathbf{I}}(A_j) = \mathrm{val}_{\mathbf{I}}(A_j) = i_j$ for $2 \leq j \leq n$, then we have that

$$\begin{aligned} k + i_2 + \cdots + i_n &\equiv 0(m) \\ l + i_2 + \cdots + i_n &\equiv 0(m) \end{aligned}$$

by the definition of $\odot$. But this means that $k - l \equiv 0(m)$ and hence, since $1 \leq k, l < m$ that $k = l$. Hence $|W_1| = 1$. Similarly, we show $|W_j| = 1$ for $j \geq 2$. Thus F is of the form

$$\bigvee_{(i_1,\ldots,i_n)} ({A_1}^{i_1} \wedge \ldots \wedge {A_n}^{i_n}).$$

Every disjunct corresponds to exactly one tuple $(i_1, \ldots, i_n)$ s.t. $i_1 + \cdots + i_n \equiv 0(m)$. There are m^{n-1} such n-tuples. (In general, there are m^{n-1} tuples s.t. $\sum_{j=1}^{n} i_j \equiv k(m)$ for any given k. Proof by induction: for $n = 1$, only (k) satisfies the condition. Assume the statement holds for n. For any choice of i_{n+1} there are m^{n-1} choices for $(i_1, \ldots, i_n)$ s.t. $\sum_{j=1}^{n} i_j \equiv k - i_{n+1}(m)$.) ∎

1.6.3. PROPOSITION *For every quantifier* Q *there is an* i*-th partial normal form containing at most* 2^{m-1} *conjuncts.*

Proof. Consider the disjunctive form

$$\bigvee_{X \subseteq V \setminus \{w\}} \Big(\bigwedge_{u \in X} (\exists x) A(x)^u \wedge [B] \wedge (\forall x) \bigvee_{u \in X} (A(x)^u \vee [C]) \Big)$$

where B is $(\exists x)A(x)^v$ iff $\widetilde{\mathsf{Q}}(X) = v_i$, and C is $A(x)^w$ iff $\widetilde{\mathsf{Q}}(X \cup \{w\}) \neq v_i$.

This form is equivalent to $\neg(\mathsf{Q}x)A(x)^{v_i}$: Let $\mathbf{I}$ be an interpretation satisfying $\neg(\mathsf{Q}x)A(x)^{v_i}$. In other words, $\mathrm{Distr}_{\mathbf{I}}(A(x)) = Y$ s.t. $\widetilde{\mathsf{Q}}(Y) \neq v_i$. But then the disjunct corresponding to $Y \setminus \{w\}$ is true under $\mathbf{I}$, since all the existential conjuncts are true (for every truth value $u \in Y$ there is a $d \in D$ s.t. $A(d)^u$), and the universal conjunct is true as well (there is no $d \in D$ s.t. $A(d)^{u'}$ for some $u' \notin Y$).

Conversely, let $\mathbf{I}$ be an interpretation satisfying the disjunct corresponding to $X \subseteq V \setminus \{w\}$. We distinguish cases according to which of B and C are present in this disjunct: If B is present (not present) then $X \cup \{w\} \subseteq \mathrm{Distr}_{\mathbf{I}}(A(x))$ ($X \subseteq \mathrm{Distr}_{\mathbf{I}}(A(x))$), since all existential conjuncts are satisfied. Since the universal conjunct is true as well, there can be no $u \in \mathrm{Distr}_{\mathbf{I}}(A(x))$ s.t. $u \notin X$ ($u \notin X \cup \{w\}$ if C is present). Hence, $\mathrm{Distr}_{\mathbf{I}}(A(x)) \subseteq X$ ($\mathrm{Distr}_{\mathbf{I}}(A(x)) \subseteq X \cup \{w\}$). Now, if neither C nor D are present, we have $\mathrm{Distr}_{\mathbf{I}}(A(x)) = X$, and if both C and D are present, we have $\mathrm{Distr}_{\mathbf{I}}(A(x)) = X \cup \{w\}$. In either case $\widetilde{\mathsf{Q}}(X \cup \{w\}) \neq v_i$ by the conditions on C and D. If only C is present, then

$X \subseteq \mathrm{Distr}_{\mathbf{I}}(A(x)) \subseteq X \cup \{w\}$. But C alone is present iff $\widetilde{\mathsf{Q}}(Y) \neq v_i$ for both $Y = X$ and $Y = X \cup \{w\}$. The case where B alone is present is impossible (the the universal conjunct cannot be true because of $(\exists x)A(x)^w$).

By negating the whole expression we obtain a conjunctive form

$$\bigwedge_{X \subseteq V \setminus \{w\}} \Big(\bigvee_{u \in X} (\forall x) \bigvee_{u' \in V \setminus \{u,w\}} A(x)^{u'} \vee [B'] \vee (\exists x) \bigvee_{u \in V \setminus (X \cup \{w\})} A(x)^u \vee [C'] \Big)$$

where B' is $(\forall x)\bigvee_{u' \in V \setminus \{w\}} A(x)^{u'}$ and C' is $A(x)^w$. This conjunctive normal form contains at most 2^{m-1} conjuncts, the number of subsets of $V \setminus \{w\}$. ■

1.6.4. PROPOSITION *For every n, there is a distribution quantifier R s.t. any i-th partial normal form contains 2^{m-1} conjuncts.*

Proof. Let $u \neq v_i$, and let R be defined by

$$\widetilde{\mathsf{R}}(W) = \begin{cases} v_i & |W| \equiv 0(2) \\ u & \text{otherwise} \end{cases}$$

Let

$$\bigwedge_{\langle \bar{V}_1, \ldots, \bar{V}_r, \bar{W} \rangle} (\forall x) \bigvee_{w \in \bar{V}_1} A(x)^w \vee \ldots \vee (\forall x) \bigvee_{w \in \bar{V}_r} A(x)^w \vee (\exists x) \bigvee_{w \in \bar{W}} A(x)^w)$$

be a minimal i-th partial normal form for $(\mathsf{R}x)A(x)$ (Note that $\exists$ distributes over $\vee$, hence any minimal pnf can be written in the above form). This gives a disjunctive form for $\neg(\mathsf{R}x)A(x)^{v_i}$ which is minimal iff the above pnf is minimal and is of the form

$$F = \bigvee_{\langle V_1, \ldots, V_r, W \rangle} ((\exists x) \bigvee_{w \in V_1} A(x)^w \wedge \ldots \wedge (\exists x) \bigvee_{w \in V_r} A(x)^w \wedge (\forall x) \bigvee_{w \in W} A(x)^w)$$

where $V_j = V \setminus \bar{V}_j$ and $W = V \setminus \bar{W}$. Observe that from minimality it follows that no disjunct is redundant (i.e., for every disjunct there is an interpretation satisfying this disjunct and no other) and that no existential conjunct is redundant (i.e., for no $k \neq l$ it holds that $V_k \subseteq V_l$). We show the following three properties for this form:

(1) $\bigcup V_j \subseteq W$,

(2) $W \subseteq \bigcup V_j$ (and hence, $W = \bigcup V_j$),

(3) $|V_j| = 1$.

(1) If there were a j and a $v \in V_j$ with $v \notin W$, then either v is redundant in the sense that the form with v deleted from V_j is also equivalent to $\neg(\mathsf{R}x)A(x)^{v_i}$, or there is some interpretation with domain D where $A(d)^v$ holds (d is the only witness for $(\exists x)\bigvee_{w \in V_j} A(x)^w$). But then the universally quantified conjunct cannot be true, since $v \notin W$.

(2) Assume that there were a $w \in W$ with $w \notin V_j$ for $j = 1, \ldots, r$. Let $\mathbf{I}$ be some interpretation satisfying the disjunct under consideration, and let

$U = \{w_1, \ldots, w_r\}$ be truth values ($w_j \in V_j$) s.t. there are $d_1, \ldots, d_r \in D$ with $A(d_j)^{w_j}$. Without loss of generality we can assume that $D = \{d_1, \ldots, d_r\}$. It follows that $\widetilde{\mathsf{R}}(U) \neq v_i$ and hence that $|U| \equiv 1(2)$. Consider the interpretation $\mathbf{I}'$ with $D' = D \cup \{d_w\}$ and $\mathrm{val}_{\mathbf{I}'}(A(d_w)) = w$. Obviously, the above form is satisfied under $\mathbf{I}'$, i.e., $\widetilde{\mathsf{R}}(U \cup \{w\}) \neq v_i$. But $|U \cup \{w\}| \equiv 0(2)$, and this contradicts the definition of R.

(3) Assume there were some V_j with $V_j = V_j' \cup \{u, v\}$ where $u \neq v$. By way of contradiction, we show that there are two interpretations $\mathbf{I}'$ and $\mathbf{I}''$ which both satisfy F, but $\mathbf{I}' \not\models (\mathsf{R}x)A(x)^{v_i}$ and $\mathbf{I}'' \models (\mathsf{R}x)A(x)^{v_i}$.

Let $\mathbf{I}$ be some interpretation satisfying the disjunct under consideration, and let $U = \{w_1, \ldots, w_r\}$ be truth values ($w_j \in V_j$) s.t. there are $d_1, \ldots, d_r \in D$ with $A(d_j)^{w_j}$. Without loss of generality we can assume that $D = \{d_1, \ldots, d_r\}$. Assume $w_j = v$. We have two cases according to whether $u \in U$: If $u \notin U$, then take $\mathbf{I}' = \mathbf{I}$ and $U' = U$. Otherwise, let $k \neq j$ be such that $A(d_k)^u$ and $u \in V_k$. Now there is some w_k' in V_k with $w_k' \notin V_j$ (otherwise we would have $V_k \subseteq V_j$). Let $\mathbf{I}'$ be as follows: D' is D less all such d_k for which $A(d_k)^u$ holds, plus new domain elements d_k' and let $\mathrm{val}_{\mathbf{I}'}(A(d_k')) = w_k'$. The distribution of $A(x)$ under $\mathbf{I}'$ is $U' = U \setminus \{u\}$. We have that $|U'| \equiv 1(2)$.

Now consider $\mathbf{I}''$ with $D'' = D' \cup \{d_u\}$ and $\mathrm{val}_{\mathbf{I}''}(A(d_u)) = u$. $\mathbf{I}''$ satisfies the same disjunct as $\mathbf{I}'$. The distribution of $A(x)$ under $\mathbf{I}''$ is $U'' = U' \cup \{u\}$. But $|U''| \equiv 0(2)$, a contradiction.

In summary, we have that $|V_j| = 1$, that $W = \bigcup_{j=1}^{r} V_j$, and that $V_k \neq V_l$ for $k \neq l$. We see that no conjunct can describe more than one situation. But there are $2^{|V|-1}$ subsets of V of odd cardinality (and thus non-empty), hence there are as many disjuncts. ∎

1.7 Induced Quantifiers

There is, however, a class of distribution quantifiers which have a lower complexity of their pnf's than the worst cases in the last section. These are the quantifiers *induced* by certain connectives, namely those which are idempotent, associative and commutative. For example, the usual existential and universal quantifiers are induced by $\vee$ and $\wedge$, respectively.

1.7.1. Definition Let $\square$ be an idempotent, associative and commutative connective. The quantifier $\mathsf{Q}^{\square}$ *induced by* $\square$ is defined as follows:

$$\widetilde{\mathsf{Q}^{\square}}(\{w\}) = w \qquad \widetilde{\mathsf{Q}^{\square}}(\{w_1, \ldots, w_r\}) = \widetilde{\square}(w_1, \widetilde{\square}(w_2, \ldots, \widetilde{\square}(w_{r-1}, w_r) \cdots)$$

It is easily seen that the conditions of idempotence, associativity and commutativity are sufficient to make the induced quantifier well-defined. In fact, the induced quantifiers can be characterized as follows:

1.7.2. Proposition *A quantifier Q is an induced quantifier iff the following hold:*

(1) $\widetilde{\mathsf{Q}}(\{w\}) = w$ and

(2) $\widetilde{\mathsf{Q}}(U \cup W) = \widetilde{\mathsf{Q}}(U \cup \{\widetilde{\mathsf{Q}}(W)\})$ *for all* $U, W \neq \emptyset$ *with* $U \cup W \subseteq V$.

The inducing connective is given by $\widetilde{\Box}(u, w) = \widetilde{\mathsf{Q}}(\{u, w\})$.

Proof. If: Define $\Box$ as above. From the definition of $\Box$ it is easy to see that it is idempotent and commutative. Furthermore, we have

$$\begin{aligned}\widetilde{\Box}(u, \widetilde{\Box}(v, w)) &= \widetilde{\mathsf{Q}}(\{u\} \cup \widetilde{\mathsf{Q}}\{v, w\}) = \widetilde{\mathsf{Q}}(\{u, v, w\}) = \\ &= \widetilde{\mathsf{Q}}(\widetilde{\mathsf{Q}}(\{u, v\} \cup \{w\}) = \widetilde{\Box}(\widetilde{\Box}(u, v), w).\end{aligned}$$

We prove $\widetilde{\mathsf{Q}}(W) = \widetilde{\mathsf{Q}^{\Box}}(W)$ by induction on $|W|$: $\widetilde{\mathsf{Q}}(\{w\}) = \{w\} = \widetilde{\mathsf{Q}^{\Box}}(\{w\})$. Now, for the induction step we have

$$\begin{aligned}\widetilde{\mathsf{Q}}(\{w_1, \ldots, w_{r+1}) &= \widetilde{\mathsf{Q}}(\{w_1\} \cup \widetilde{\mathsf{Q}}(\{w_2, \ldots, w_{r+1})) = \\ &= \widetilde{\mathsf{Q}}(\{w_1\} \cup \widetilde{\mathsf{Q}^{\Box}}(\{w_2, \ldots, w_{r+1})) = \\ &= \widetilde{\Box}(w_1, \widetilde{\Box}(w_2, \ldots, \widetilde{\Box}(w_r, w_{r+1}) \cdots) = \widetilde{\mathsf{Q}^{\Box}}(w_1, \ldots, w_{r+1}).\end{aligned}$$

Only if: (1) and (2) immediately follow from Definition 1.7.1. ∎

There is a close connection between idempotent, associative and commutative connectives and upper semi-lattices over the set of truth values V:

1.7.3. PROPOSITION *Let* $\Box$ *be an idempotent, associative and commutative connective. The relation* $\sqsubseteq$ *defined by*

$$u \sqsubseteq w \quad \text{iff} \quad \Box(u, w) = w$$

is a partial order on V *with the property that the least upper bound of* $\{w_1, \ldots, w_k\} \subseteq V$ *equals* $\Box(w_1, \Box(w_2, \ldots, \Box(w_{k-1}, w_k) \cdots)$.

Proof. $\sqsubseteq$ is reflexive: $\Box$ is idempotent, $\Box(u, u) = u$, hence $u \sqsubseteq u$ for any $u \in V$. $\sqsubseteq$ is antisymmetric: If $u \sqsubseteq w$ and $w \sqsubseteq u$, then $\Box(u, w) = w$ and $\Box(w, u) = u$. But $\Box$ is commutative, hence $u = w$. $\sqsubseteq$ is transitive: If $u \sqsubseteq v$ and $v \sqsubseteq w$ we have $\Box(u, v) = v$ and $\Box(v, w) = w$. Thus $\Box(u, w) = \Box(u, \Box(v, w)) = \Box(\Box(u, v), w) = \Box(v, w) = w$ whence $u \sqsubseteq w$.

We prove that $\mathrm{lub}\{w_1, \ldots, w_k\} = \Box(w_1, \ldots, \Box(w_{k-1}, w_k))$ by induction on k: For $k = 1$ the equality holds since $\Box$ is idempotent. Assume the statement holds for $k > 1$. Obviously, $\Box(w_1, \ldots, \Box(w_k, w_{k+1}) \cdots)$ is an upper bound for $\{w_1, \ldots, w_{k+1}\}$. Now let u be an upper bound of $\{w_1, \ldots, w_{k+1}\}$, in other words: $\Box(w_j, u) = u$. In particular, u is an upper bound for $\{w_2, \ldots, w_{k+1}\}$, and by induction hypothesis $\Box(w_2, \ldots, \Box(w_k, w_{k+1}) \cdots) \sqsubseteq u$. But then

$$\begin{aligned}&\Box(\Box(w_1, \ldots, \Box(w_k, w_{k+1}) \cdots), u) = \\ &\quad = \Box(w_1, \Box(\Box(w_2, \ldots, \Box(w_k, w_{k+1}) \cdots), u)) = \\ &\quad = \Box(w_1, u) = u. \quad \blacksquare\end{aligned}$$

Conversely, every such order defines an idempotent, associative and commutative connective by $\Box(u, w) = \mathrm{lub}\{u, w\}$. For finite V such upper semi-lattices can be pictured as (non-planar) labeled, rooted trees. Hence, there are m^{m-1} idempotent, associative and commutative connectives and as many induced quantifiers: Cayley's formula (see COHEN [1978], Theorem 98) gives the number of labeled trees as m^{m-2}, and each of the m vertices can serve as the root. The induced quantifiers are obviously the supremum operators on the respective partial orders.

1.7.4. PROPOSITION *Let $\square$ be an idempotent, associative and commutative connective. There is an i-th partial normal form for $\mathsf{Q}^{\square}$ with at most $m-1$ conjuncts.*

Proof. We give a disjunctive normal form for $\neg(\mathsf{Q}^{\square}x)A(x)^{v_i}$: $(\mathsf{Q}^{\square}x)A(x)$ does not take the value v_i iff either there is some d s.t. $A(d)^u$ with $u \not\sqsubseteq v_i$ holds, or for one of v_i's lower neighbours, say w, we have that for all d, $\mathrm{val}_{\mathbf{I}}(A(d)) \sqsubseteq w$. This is expressed as follows: Let $\sqsubseteq(w) = \{u \mid u \sqsubseteq w\}$ and let $w_1, \ldots, w_k$ be all of v_i's lower neighbours.

$$(\exists x) \bigvee_{u \in V \setminus \sqsubseteq(v_i)} A(x)^u \vee \bigvee_{j=1}^{k} (\forall x) \bigvee_{u \in \sqsubseteq(w_j)} A(x)^u$$

The negation of this yields an i-th pnf for $\mathsf{Q}^{\square}$:

$$(\forall x) \bigvee_{u \in \sqsubseteq(v_i)} A(x)^u \wedge \bigwedge_{j=1}^{k} (\exists x) \bigvee_{u \in V \setminus \sqsubseteq(w_j)} A(x)^u$$

This pnf has less than or equal to $m-1$ conjuncts, except when $k = m-1$. But in this situation, the first conjunct is redundant since $\sqsubseteq(v_i) = V$. ∎

Chapter 2

Resolution

2.1 Introduction

With the rise of the digital computer in the 1950's, several logicians became interested in the possibilities of mechanizing proof search or theorem proving by computer. A major breakthrough in this area was the introduction of the resolution calculus for classical logic in ROBINSON [1965]. The resolution calculus is an elegant formalism, in that there is a single rule of inference, but no logical axioms. It uses a special notation for formulas: *clause syntax*. This notation, or actually the way one obtains clause syntax from ordinary formulas, hides much of the characteristics of predicate logic. Resolution is arguably the most widely used calculus for automated theorem proving, and also one of the most thoroughly investigated. Departing from the original formulation of Robinson, several *refinements* have been proposed and implemented. For an overview of resolution-based theorem proving see the classic textbooks of CHANG and LEE [1973] and LOVELAND [1978]. A more recent account is LEITSCH [1993]. See also AVRON [1993] for a study of connections between resolution, sequent calculus and tableaux.

Given its importance and ubiquity in the theorem proving arena, it is not surprising that resolution systems have also been developed for non-classical logics, among them many-valued logics. Probably the first such system was presented by MORGAN [1976] for a class of many-valued logics introduced in ROSSER and TURQUETTE [1952]. Since then resolution formulations for several (classes of) finite and infinite-valued logics (propositional and first order) have been introduced. For a historical survey of theorem proving formalisms for many-valued logics see HÄHNLE [1993a]. The work of Stachniak *et. al.* should be mentioned here in particular, see, e.g., STACHNIAK and O'HEARN [1990]. A different approach, based on a truth-functional analysis of classical resolution, was proposed by BAAZ [1984, 1989, 1992]. The resolution rule in this context takes the form

$$\frac{C_1 \qquad C_2}{(C_1 \setminus D_1)\sigma \cup (C_2 \setminus D_2)\sigma}$$

where $D_1 \subseteq C_1$, $D_2 \subseteq C_2$, the literals in D_1 and D_2 are signed with truth values w_1 and w_2, respectively, $w_1 \neq w_2$, and the atoms in D_1 and D_2 are unifiable with mgu σ. This is a straightforward generalization of the classical case, where

$w_1 = t$ and $w_2 = f$ and the signs are expressed by absence or presence of $\neg$, respectively.

One advantage of this approach is that several methods of refining the deductive system developed for the classical case can also be generalized to this framework. We shall not give such refinements here, but see Baaz and Fermüller [1992, 1993]. This chapter will be devoted to justifying the above rule. Some familiarity with classical resolution terminology is assumed.

2.2 Clauses and Herbrand Semantics

2.2.1. Definition A *(many-valued) clause* $C = \{A_1{}^{w_1}, \ldots, A_n{}^{w_n}\}$ is a finite set of signed atoms. By $\Box$ we denote the empty clause.

The *atom set* $\mathrm{at}(C)$ of a clause C is the set of its atomic formulas: $\mathrm{at}(C) = \{A_1, \ldots, A_n\}$.

The clause syntax is a convenient notation for automated theorem proving. In the classical case, a clause denotes a disjunction of atoms or negations of atoms, and a set of clauses a conjunction of such disjunctions. Clause sets are implicitly considered as universally closed. By bringing a formula into prenex form and then introducing Skolem functions, we can always obtain a formula in this form which is satisfiable iff the original formula is. In this sense, clause syntax is sufficient to capture full classical first order logic.

Similarly, for every set of many-valued clauses there is a corresponding sfe in conjunctive normal form: A clause $C = \{A_1{}^{w_1}, \ldots, A_p{}^{w_p}\}$, where A_i is an atomic formula and $w_i \in V$, denotes $\bigvee\!\!\!\!\bigvee_{i=1}^{p} A_i{}^{w_i}$. A set of clauses $\mathcal{C}$ denotes the conjunction of the disjunctions corresponding to its members. The sfe in conjunctive normal form corresponding to $\mathcal{C}$ is uniquely determined up to associativity and commutativity of $\bigwedge\!\!\!\!\bigwedge$ and $\bigvee\!\!\!\!\bigvee$. When we say, e.g., that $\mathcal{C}$ is true in an interpretation, we mean that the corresponding sfe is true.

2.2.2. Definition Let $\mathcal{M} = \langle D, \Phi\rangle$ be a structure. $\mathcal{M}$ *universally satisfies* a set of clauses $\mathcal{C}$ iff $\mathcal{C}$ is true in $\langle \mathcal{M}, s\rangle$ for every assignment s,.

2.2.3. Definition The *Herbrand universe* $H(\mathcal{C})$ of a set of clauses $\mathcal{C}$ is the set of all ground terms built up from the function symbols and constants in $\mathcal{C}$, or from the function symbols in $\mathcal{C}$ and a new constant a, if $\mathcal{C}$ contains no constants.

2.2.4. Definition The *Herbrand base* $A(\mathcal{C})$ of a set of clauses $\mathcal{C}$ is the set of all atomic formulas $P(t_1, \ldots, t_n)$ where P is a predicate symbol in $\mathcal{C}$, and t_1, $\ldots$, $t_n \in H(\mathcal{C})$.

2.2.5. Definition An *Herbrand interpretation (H-interpretation)* $\mathbf{H}$ of a set of clauses $\mathcal{C}$ is a set of ground literals $\{L^{\phi(L)} \mid L \in A(\mathcal{C})\}$, where ϕ is a function from $A(\mathcal{C})$ to V.

$\mathbf{H}$ *H-satisfies* $\mathcal{C}$, iff for every ground instance C' of a clause $C \in \mathcal{C}$ it holds that $C' \cap \mathbf{H} \neq \emptyset$. If no H-interpretation satisfies $\mathcal{C}$, then $\mathcal{C}$ is said to be *H-unsatisfiable*. If $\mathbf{H} \models \mathcal{C}$, then $\mathbf{H}$ is called an *Herbrand model* for $\mathcal{C}$.

In general, if A is a set of atoms, then a set of the form $\{L^{\phi(L)} \mid L \in A\}$ is called an *assignment* to A. A subset of an assignment is a *partial assignment* to A. Hence, an H-interpretation is an assignment to $A(\mathcal{C})$.

Intuitively, it is clear that every H-interpretation also constitutes a structure in the sense of Definition 1.3.7, and that to every ordinary interpretation there is a corresponding H-interpretation. It is then not surprising that H-satisfiability and satisfiability proper coincide in the sense that a set of clauses is H-satisfiable iff it (considered as an sfe) is satisfiable. Since in the case of classical logic, the meta-notation of sfes can be interpreted on the level of the object language, the following results subsume the corresponding results for classical logic as subcases.

2.2.6. DEFINITION Let $\mathbf{H}$ be an H-interpretation of a set of clauses $\mathcal{C}$. The *structure* $\mathcal{M}_{\mathbf{H}}$ *corresponding to* $\mathbf{H}$ is defined as follows: $\mathcal{M}_{\mathbf{H}} = \langle H(\mathcal{C}), \Phi \rangle$, with Φ as follows:

(1) function symbols: $\Phi(f) = \tilde{f}$ where $\tilde{f}$ is the n-ary function mapping the arguments $t_1, \ldots, t_n \in H(\mathcal{C})$ to $f(t_1, \ldots, t_n) \in H(\mathcal{C})$, or $\Phi(f) = f$ if f is a constant symbol.

(2) predicate symbols: $\Phi(P) = \widetilde{P}$ where $\widetilde{P}$ is the function mapping $\langle d_1, \ldots, d_n \rangle$ to $\phi(P(d_1, \ldots, d_n))$.

2.2.7. PROPOSITION *If* $\mathbf{H}$ *H-satisfies a set of clauses* $\mathcal{C}$*, then* $\mathcal{M}_{\mathbf{H}}$ *universally satisfies* $\mathcal{C}$*.*

Proof. Let C be a clause in $\mathcal{C}$ and let s be any assignment for $\mathcal{M}_{\mathbf{H}}$. This maps free variables to ground terms. C has a ground instance Cs obtained by replacing every variable in C by its image under s. By hypothesis, there is a literal $L^w \in Cs \cap \mathbf{H}$. The way $\mathcal{M}_{\mathbf{H}}$ was constructed ensures that $\mathrm{val}_{\mathcal{M}_{\mathbf{H}}}(L) = w$. But this means that $\mathcal{C}$ is true in $\langle \mathcal{M}_{\mathbf{H}}, s \rangle$. ∎

2.2.8. DEFINITION Let $\mathcal{M} = \langle D, \Phi \rangle$ be a structure, and $\mathcal{C}$ a set of clauses. An *H-interpretation* $\mathbf{H}_{\mathcal{M}}$ *corresponding to* $\mathcal{M}$ is an H-interpretation satisfying the following condition:

(C) If $\mathrm{val}_{\mathcal{M}}(F') = v$, where F' is a ground instance of some formula F, then $F'^v \in \mathbf{H}_{\mathcal{M}}$.

2.2.9. PROPOSITION *If* $\mathcal{M} = \langle D, \Phi \rangle$ *universally satisfies* $\mathcal{C}$*, then any corresponding* $\mathbf{H}_{\mathcal{M}}$ *H-satisfies* $\mathcal{C}$*.*

Proof. Assume that there were a ground instance C' of a clause in $\mathcal{C}$ and an H-interpretation $\mathbf{H}_{\mathcal{M}}$, s.t. $C' \cap \mathbf{H}_{\mathcal{M}} = \emptyset$. Let $C' = C\sigma$, where $\sigma = \{t_1/a_1, \ldots, t_n/a_n\}$ and $a_1, \ldots, a_n$ are all the variables occurring in C. Let $\sigma' = \{\Phi'(t_1)/a_1, \ldots, \Phi'(t_n)/a_n\}$, where Φ' is Φ extended by a value for the new constant a in case $\mathcal{C}$ contained no constant symbols.

By (C) we have for all $A_i'^{w_i} \in C'$ that $\mathrm{val}_{\mathcal{M}}(A_i') \neq w_i$ (where A_i' is an instance of a literal $A_i \in C$), and hence that the assignment σ' of domain elements to the free variables of C is such that C is not satisfied in $\langle \mathcal{M}, \sigma' \rangle$. This contradicts the assumption that $\mathcal{M}$ universally satisfies $\mathcal{C}$. ∎

It is easy to see that $\mathbf{H}$ is an H-interpretation corresponding to $\mathcal{M}_{\mathbf{H}}$ for a given $\mathcal{M}$.

2.3 Clause Translation Calculi

In the two-valued case, there is a direct interpretation of a set of clauses as a first-order formula, namely as the universal closure of a conjunction of disjunctions, where the literals in the disjunction are exactly the members of the individual clauses. Since any classical first-order formula can be written in this form by familiar equivalences of connectives, distribution rules, and skolemization, resolution becomes applicable as a refutational proof system. To establish the unsatisfiability of a formula, we have to test if the corresponding clause form derives the empty clause. In the many-valued case we also need to find such a translation mechanism in order for resolution to be of any practical value. It is not surprising that the partial normal forms for connectives and quantifiers provide us with such a translation mechanism. The conditions such a translation has to satisfy are soundness and completeness: A signed formula is universally unsatisfiable iff its clause translation is. By the last two propositions, this coincides with H-unsatisfiability on the clause level.

The clause translation calculi presented here are *language preserving*. A different, and more effective way to translate formulas to clause forms is the *structure preserving* method (see Hähnle [1993b] and Baaz and Fermüller [1993]).

2.3.1. Definition An *extended clause* is a finite set of signed formulas.

Here again, an extended clause is read as the sfe consisting of the disjunction of its members. A set of extended clauses is the conjunction of its members. Every clause is also an extended clause, and every signed formula can be thought of as a unit extended clause. Universal satisfiability is defined just as in the case of ordinary clause sets.

2.3.2. Proposition *Let F be a closed formula.*

(1) The extended clause $\{F^w \mid w \in W \subseteq V\}$ is satisfiable iff it is universally satisfiable.

(2) F is a many-valued tautology iff $\{F^w \mid w \in V \setminus V^+\}$ is universally unsatisfiable.

Proof. Immediate from Proposition 1.4.3, since F contains no free variables. ∎

Proof. By Proposition 1.4.3 ∎

2.3.3. DEFINITION Let $\bigwedge_{j\in I} \Delta_{\Box:i}(j)$ be a positive i-th partial normal form of $\Box(A_1,\ldots,A_n)$, where $\Box$ is of arity n, $\Delta_{\Box:i}(j)$ is a disjunction of signed atoms of the form ${A_i}^w$ $(1 \le i \le n)$, and let $\Delta'_{\Box:i}(j)$ be the clause corresponding to $\Delta_{\Box:i}(j)$

A *translation rule for* $\Box$ *at place* i is a schema of the form:

$$\frac{\mathcal{C} \cup \{C \cup \{\Box(A_1,\ldots,A_n)^{v_i}\}\}}{\mathcal{C} \cup \{C \cup \Delta'_{\Box:i}(j) \mid j \in I\}}\ \Box{:}i$$

2.3.4. THEOREM *Let* $\mathcal{C}$, C, $\Box(A_1,\ldots,A_n)^{v_i}$ *and* $\Delta'_{\Box:i}(j)$ *be as in Definition 2.3.3. Then*

$$\mathcal{D} = \mathcal{C} \cup \{C \cup \Delta'_{\Box:i}(j) \mid j \in I\}$$

is satisfiable iff

$$\mathcal{E} = \mathcal{C} \cup \{C \cup \{\Box(A_1,\ldots,A_n)^{v_i}\}\}$$

is satisfiable.

Proof. Let $\mathbf{H}$ be an H-interpretation satisfying $\mathcal{D}$ $(\mathcal{E})$. $\mathcal{M}_\mathbf{H}$ universally satisfies $\mathcal{D}$ $(\mathcal{E})$ by Proposition 2.2.7. Let s be an assignment to the free variables of $\mathcal{D}$(resp., $\mathcal{E}$). $\langle \mathcal{M}_\mathbf{H}, s\rangle$ satisfies all clauses in $\mathcal{C}$. If it does not satisfy C, then it has to satisfy all the $\Delta'_{\Box:i}(j)$ (resp., $\Box(A_1,\ldots,A_n)^{v_i}$). By the definition of a partial normal form for $\Box$, $\langle \mathcal{M}_\mathbf{H}, s\rangle$ then satisfies also $\Box(A_1,\ldots,A_n)^{v_i}$ (resp., all of $\Delta'_{\Box:i}(j)$). Hence $\mathcal{M}_\mathbf{H}$ universally satisfies $\mathcal{E}$ (resp., $\mathcal{D}$). By Proposition 2.2.9, $\mathcal{E}$ (resp., $\mathcal{D}$) is H-satisfied by $\mathbf{H}$. ■

2.3.5. DEFINITION Let $\bigwedge_{j\in I} \Delta_{\mathsf{Q}:i}(j)$ be a positive i-th partial normal form of $(\mathsf{Q}x)A(x)$, where $\Delta_{\mathsf{Q}:i}(j)$ is a disjunction of signed atoms of the form $A(\tau_i)^w$ $(1 \le i \le p)$ respectively $A(\alpha_i)^w$ $(1 \le i \le q)$, and let $\Delta'_{\mathsf{Q}:i}(j)$ be the clause corresponding to $\Delta_{\mathsf{Q}:i}(j)$.

A *translation rule for* Q *at place* i is a schema of the form:

$$\frac{\mathcal{E} = \mathcal{C} \cup \{C \cup \{((\mathsf{Q}x)A(x))^{v_i}\}\}}{\mathcal{D} = \mathcal{C} \cup \{C \cup \Delta''_{\mathsf{Q}:i}(j) \mid j \in I\}}\ \Box{:}i$$

where $\Delta''_{\mathsf{Q}:i}(j)$ is obtained from $\Delta'_{\mathsf{Q}:i}(j)$ by

(1) replacing term variables τ_i by terms of the form $f_i(a_1,\ldots,a_k)$, where f_i are distinct new k-ary function symbols and a_1, $\ldots$, a_k are all free variables in $C \cup \{((\mathsf{Q}x)A(x))^{v_i}\}$, and by

(2) replacing eigenvariables α_i by distinct new free variables b_i (not occurring in $\mathcal{E}$).

2.3.6. THEOREM *Let* $\mathcal{C}$, C, $((\mathsf{Q}x)A(x))^{v_i}$ *and* $\Delta''_{\Box:i}(j)$ *be as in Definition 2.3.5. Then*

$$\mathcal{D} = \mathcal{C} \cup \{C \cup \Delta''_{\mathsf{Q}:i}(j) \mid j \in I\}$$

is satisfiable iff

$$\mathcal{E} = \mathcal{C} \cup \{C \cup \{(\mathsf{Q}x)A(x)^{v_i}\}\}$$

is satisfiable.

Proof. If: Let $\mathbf{H}$ be an H-interpretation satisfying $\mathcal{E}$. $\mathcal{M}_{\mathbf{H}}$ universally satisfies $\mathcal{E}$ by Proposition 2.2.7. Assume, by way of contradiction, that $\mathcal{D}$ were not universally satisfiable, i.e., for every structure $\mathcal{M}'$ for $\mathcal{D}$, there is an assignment s' s.t. $\langle \mathcal{M}', s' \rangle$ does not satisfy $\mathcal{D}$. In particular, any structure $\mathcal{M}'_{\mathbf{H}}$ extending $\mathcal{M}_{\mathbf{H}}$ by providing an interpretation for the new function symbols cannot universally satisfy $\mathcal{D}$. We show that this would imply that $\mathcal{M}_{\mathbf{H}}$ does not universally satisfy $\mathcal{E}$:

Let $\mathcal{M}'_{\mathbf{H}}$ be such a structure extending $\mathcal{M}_{\mathbf{H}}$. There is an assignment s' s.t. $\langle \mathcal{M}'_{\mathbf{H}}, s' \rangle$ does not satisfy $\mathcal{D}$. If $\langle \mathcal{M}'_{\mathbf{H}}, s' \rangle$ falsifies a clause in $\mathcal{C}$, then so does $\langle \mathcal{M}_{\mathbf{H}}, s \rangle$ since the new function symbols and variables do not occur in $\mathcal{C}$. So assume that $\langle \mathcal{M}'_{\mathbf{H}}, s' \rangle$ falsifies some $\Delta''_{\mathsf{Q}:i}(j)$. That is, for the assignment of $\widetilde{f}_i(s'(a_1), \ldots, s'(a_k))$ to τ_i and $s'(b_i)$ for α_i we have $\langle \mathcal{M}'_{\mathbf{H}}, s' \rangle \not\models \Delta_{\mathsf{Q}:i}(j)$. We know that this property holds for every structure extending $\mathcal{M}_{\mathbf{H}}$, i.e., for every possible interpretation of the function symbols and every possible assignment to the variables, there are assignments to the new (eigen-) variables s.t. some $\Delta_{\mathsf{Q}:i}(j)$ is falsified. By Definition 1.5.5 (2b) of partial form for Q, $\mathrm{val}_{\mathcal{M}_{\mathbf{H}}}((\mathsf{Q}x)A(x)) \neq v_i$.

Only if: If $\mathcal{D}$ is satisfiable, it is also universally satisfiable. Let $\mathcal{M}$ be a structure universally satisfying $\mathcal{D}$, and let s be an assignment for $\mathcal{M}$. In $\langle \mathcal{M}, s \rangle$ either C is true or all of $\Delta''_{\mathsf{Q}:i}(j)$ $(j \in I)$. In the first case, $\langle \mathcal{M}, s \rangle$ also satisfies $\mathcal{E}$. Otherwise, let s' be an assignment with $s' \sim_{b_1,\ldots,b_q} s$. $\langle \mathcal{M}, s' \rangle$ also satsifies all of $\Delta''_{\mathsf{Q}:i}(j)$ $(j \in I)$, since the b_i do not occur in C. But this means that for all assignments to the eigenvariables (namely, all s'), there are assignments to the term variables (namely, $\Phi_{\langle \mathcal{M}, s' \rangle}(f_i(a_1, \ldots, a_k))$) s.t. the conjunction of the $\Delta''_{\mathsf{Q}:i}(j)$ is true in $\langle \mathcal{M}, s \rangle$. By Definition 1.5.5 (2a) then $\langle \mathcal{M}, s \rangle \models (\mathsf{Q}x)A(x)^{v_i}$. The assignment s was arbitrary, so $\mathcal{E}$ is universally satisfiable, and, by Proposition 2.2.9, H-satisfiable. ∎

Using the translation rules defined above, any set of extended clauses $\mathcal{E}$ can be transformed to a set of clauses $\mathcal{C}$ which is H-unsatisfiable iff $\mathcal{E}$ is universally unsatisfiable. In case $\mathcal{E}$ contains no free variables, this is just the case iff $\mathcal{E}$ is unsatisfiable. If we want to prove that a given closed formula F is unsatisfiable, Proposition 2.3.2 gives us a set of extended formulas which is unsatisfiable iff F is. Using the translation, we can apply resolution to obtain a proof.

2.3.7. Example The following are the well known translation rules for classical logic:

$$\frac{\mathcal{C} \cup \{C \cup \{\neg A^f\}\}}{\mathcal{C} \cup \{C \cup \{A^t\}\}}\ \neg{:}f \qquad \frac{\mathcal{C} \cup \{C \cup \{\neg A^t\}\}}{\mathcal{C} \cup \{C \cup \{A^f\}\}}\ \neg{:}t$$

$$\frac{\mathcal{C} \cup \{C \cup \{A \wedge B^f\}\}}{\mathcal{C} \cup \{C \cup \{A^f, B^f\}\}}\ \wedge{:}f \qquad \frac{\mathcal{C} \cup \{C \cup \{A \wedge B^t\}\}}{\mathcal{C} \cup \{C \cup \{A^t\}, C \cup \{B^t\}\}}\ \wedge{:}t$$

$$\frac{\mathcal{C} \cup \{C \cup \{A \supset B^f\}\}}{\mathcal{C} \cup \{C \cup \{A^t\}, C \cup \{B^f\}\}}\ \supset{:}f \qquad \frac{\mathcal{C} \cup \{C \cup \{A \supset B^t\}\}}{\mathcal{C} \cup \{C \cup \{A^f, B^t\}\}}\ \supset{:}t$$

$$\frac{\mathcal{C} \cup \{C \cup \{(\forall x)A(x)^f\}\}}{\mathcal{C} \cup \{C \cup \{A(f(a_1, \ldots, a_k))^f\}\}}\ \forall{:}f \qquad \frac{\mathcal{C} \cup \{C \cup \{(\forall x)A(x)^t\}\}}{\mathcal{C} \cup \{C \cup \{A(b)^t\}\}}\ \forall{:}t$$

2.4 Semantic Trees and Herbrand's Theorem

2.4.1. DEFINITION Let $\mathcal{C}$ be a set of clauses, and $K \subseteq A(\mathcal{C})$. Let $B(K) = \{A^{v_i} \mid A \in K, v_i \in V\}$ be the union of all assignments to K.

A *semantic tree* S for K is a finitary, rooted, downward directed tree with labeled edges s.t. the following hold:

(1) Every edge e is labeled by a subset $l(e) \subseteq B(K)$.

(2) If $e_1, \ldots, e_k$ are all the edges leaving some node X, then $\{l(e_1), \ldots, l(e_k)\}$ is H-unsatisfiable.

(3) For every branch B in S the following hold: Let B consist of the edges e_1, $e_2, \ldots$ There is exactly one assignment A_K to K s.t. $A_K \cap \bigcup_{i\geq 1} l(e_i) = \emptyset$. We say B *omits* A_K.

(4) For every assignment A_K to K there is a branch B in S which omits A_K.

By $I(X)$ we denote the union of labels on the unique path from the root to X. $I(X)$ is called the *refutation set* of X. If B is a branch in S then $I(B) = \bigcup_{X\in B} I(X)$.

2.4.2. *Remark* Note that the concept of semantic tree as defined here is dual to the usual one (see KOWALSKI and HAYES [1969]).

We see from the above definition that, given a semantic tree S for $A(\mathcal{C})$, every H-interpretation is omitted by a branch in S and every branch in S omits an H-interpretation.

2.4.3. EXAMPLE We construct a special binary semantic tree for K were every edge is labeled by a singleton. For this, let A be an atom. By $T(A)$ denote the tree constructed as follows:

(1) Stage 0: $T(A)_0$ contains a single node X.

(2) Stage $k+1$: Let X_s be a leaf of $T(A)_k$, and let $V(X) = \{w \mid A^w \in I(X_s)\}$ be the set of all truth value indices labeling the edges on the path from X to X_s. Let $i < j$ be minimal indices s.t. $v_i \notin V(X)$, $v_j \notin V(X)$. Append to X_s two nodes $X_{s,1}$ and $X_{s,2}$, and label the edges leading to them $\{A^{v_i}\}$ and $\{A^{v_j}\}$, respectively.

(3) Stop at stage $m-1$.

The following holds for $T(A)$: First, all branches of $T(A)$ have length $m-1$ and the labels on one branch are distinct. Hence, every branch omits exactly one literal.

Furthermore, every literal is omitted by some branch in $T(A)$. Proof by induction on m: For $m = 2$ this is evident. Now consider the tree $T(A)_{m-2}$: The maximal index on a branch is $m-1$. In fact, $T(A)_{m-2}$ equals the tree $T'(A)$ for $V' = V \setminus \{v_m\}$. By induction hypothesis, some branch B in $T(A)_{m-2}$ omits a given literal A^u. Whatever the v_i and v_j are in the construction of the successors of the last node on B, one of them is certainly distinct from u. The respective branch in $T(A)$ omits A^u.

Now let K be enumerated (without repetitions) by A_1, A_2, A_3, ... Define T in stages as follows:

(1) Stage 1: Write down $T(A_1)$.

(2) Stage $k + 1$. Replace all leaves of the tree constructed in stage k by $T(A_{k+1})$.

It is easy to see that T is indeed a semantic tree.

2.4.4. DEFINITION Let C be a clause and S a semantic tree. C *fails* at a node X in S iff there is a ground instance C' of C s.t. $C' \subseteq I(X)$.

A node X in S *falsifies* a set of clauses $\mathcal{C}$ iff some $C \in \mathcal{C}$ fails at X. If there is no node above X with this property, then X is called a *failure node* w.r.t. $\mathcal{C}$. If all successors of a node Y are failure nodes for $\mathcal{C}$, then Y is called an *inference node*. If every branch in S contains a failure node, then S is said to be *closed* w.r.t. $\mathcal{C}$.

2.4.5. PROPOSITION *A set of clauses $\mathcal{C}$ is unsatisfiable iff every semantic tree for $A(\mathcal{C})$ is closed w.r.t. $\mathcal{C}$.*

Proof. If: Let S be a semantic tree for $A(\mathcal{C})$ and let $\mathbf{H}$ be an H-interpretation for $\mathcal{C}$. By definition of a semantic tree, $\mathbf{H}$ is omitted by some branch B in S. S is closed, so there is a failure node X on B. For a ground instance C' of a clause $C \in \mathcal{C}$ we have $C' \subseteq I(X)$. If $\mathbf{H}$ would H-satisfy $\mathcal{C}$, then $C' \cap \mathbf{H} \neq \emptyset$ and consequently $I(X) \cap \mathbf{H} \neq \emptyset$. But this contradicts the assumption that B omits $\mathbf{H}$.

Only if: Let S be a semantic tree for $A(\mathcal{C})$. Every branch B of S omits an H-interpretation $\mathbf{H}$. Since $\mathcal{C}$ is H-unsatisfiable, we have that for a ground instance C' of some clause $C \in \mathcal{C}$ we have $C' \cap \mathbf{H} = \emptyset$. From this it follows that $C' \subseteq I(B)$ (since B omits $\mathbf{H}$, and hence $B(A(\mathcal{C})) \setminus \mathbf{H} = I(B)$), and C fails at some node X on B. So S is closed w.r.t. $\mathcal{C}$. ∎

2.4.6. PROPOSITION *The tree $R(S)$ resulting from pruning a closed semantic tree S at all its failure nodes is finite.*

Proof. Assume $R(S)$ were infinite. By König's Lemma, it would then contain an infinite branch. This contradicts the assumption that S is closed. ∎

2.4.7. THEOREM (Herbrand's Theorem) *A set of clauses $\mathcal{C}$ is unsatisfiable iff there is a finite unsatisfiable set of ground instances of clauses in $\mathcal{C}$.*

Proof. If: This follows immediately from the definition of H-satisfaction.

Only if: By Propositions 2.4.5 and 2.4.6, there is a closed semantic tree R, all of whose (finitely many) failure nodes X_i falsify ground instances C'_i of clauses in $\mathcal{C}$. If X_1, ..., X_n are all failure nodes of R, then $\{C'_1, \ldots, C'_n\}$ is a finite unsatisfiable set of ground instances of clauses in $\mathcal{C}$. ∎

2.5 Soundness and Completeness

2.5.1. DEFINITION Let C_1 and C_2 be clauses s.t. $\text{var}(C_1) \cap \text{var}(C_2) = \emptyset$, and let $D_1 \subseteq C_1$, $D_2 \subseteq C_2$ be such that every literal in D_1 is signed with a truth value v, every literal in D_2 with a truth value w, and $v \neq w$. If $A = \text{at}(D_1 \cup D_2)$ is unifiable with mgu σ, then $\langle D_1, D_2, A \rangle$ is called a *key triple with mgu* σ for C_1, C_2.

Resolution of two clauses C_1' and C_2' requires that they be variable disjoint, i.e., that $\text{var}(C_1') \cap \text{var}(C_2') = \emptyset$. We call C_1 and C_2 *variable disjoint renamings* of C_1' and C_2', if $C_1 = C_1'\lambda_1$ and $C_2 = C_2'\lambda_2$, where λ_i leaves the variables in $\text{var}(C_i) \setminus \text{var}(C_j)$ intact, and maps $\text{var}(C_i) \cap \text{var}(C_2)$ to different new variables not in $\text{ran}(\lambda_j)$ $(i = 1, 2;\ j = 2, 1)$.

2.5.2. DEFINITION (Resolvent) Let C_1' and C_2' be clauses, let C_1 and C_2 be variable disjoint renamings, respectively, and let $\langle D_1, D_2, A \rangle$ be a key triple with mgu σ for C_1, C_2. Then the clause $(C_1 \setminus D_1)\sigma \cup (C_2 \setminus D_2)\sigma$ is a *resolvent* of C_1', C_2'.

2.5.3. DEFINITION A *resolution deduction* of a clause D from a set of clauses $\mathcal{C}$ is a finite sequence of clauses R_1, R_2, ..., $R_s = D$ s.t. either $R_i \in \mathcal{C}$ or R_i is a resolvent of clauses R_j, $R_{j'}$, $j, j' < i$. In that case we write $\mathcal{C} \vdash D$.

2.5.4. LEMMA *Let R be a resolvent of C_1 and C_2. If a model $\mathbf{H}$ satisfies C_1 and C_2, then it also satisfies R.*

Proof. Clearly, $\mathbf{H}$ satisfies C_1 and C_2 iff it satisfies their variable disjoint renamings. W.l.o.g. we can therefore assume that $\text{var}(C_1) \cap \text{var}(C_2) = \emptyset$ and hence $\text{var}(R) \subseteq \text{var}(C_1) \cup \text{var}(C_2)$. Assume, by way of contradiction, that $\mathbf{H}$ would not satisfy R, i.e., that $R\lambda \cap \mathbf{H} = \emptyset$ for some ground substitution λ. If a literal $L^v\sigma\lambda$ occurs in $R\lambda$, then L^v occurs in C_1 or C_2 (σ is the mgu the resolution). The substitution $\sigma\lambda$ can be extended to a ground substitution η with $\text{ran}(\eta) = \text{var}(C_1) \cup \text{var}(C_2)$ s.t. $\eta \mid \text{var}(R) = \sigma\lambda$.

Now $(C_1 \setminus D_1)\eta \cap \mathbf{H} = \emptyset$. $D_1\eta$ is a singleton set, say, $\{L^v\}$, and since $\mathbf{H}$ satisfies C_1 we have that $\{L^v\} \cap \mathbf{H} \neq \emptyset$. In other words, $L^v \in \mathbf{H}$. Similarly, we obtain $D_2\eta = \{L^w\} \subseteq \mathbf{H}$. But $v \neq w$ and this contradicts the definition of Herbrand model. ■

2.5.5. THEOREM (Soundness) *If $\mathcal{C} \vdash \Box$ then $\mathcal{C}$ is unsatisfiable.*

Proof. If $\mathcal{C}$ were satisfiable, then a model of $\mathcal{C}$ would also be a model of the resolvents R_1, ..., R_s in the corresponding resolution deduction by Lemma 2.5.4. But $R_s = \Box$ and $\Box$ is unsatisfiable. ■

2.5.6. THEOREM (Lifting Lemma) *Let C_1' and C_2' be ground instances of clauses C_1 and C_2, respectively, and let R' be a resolvent of C_1' and C_2'. Then there is a resolvent R of C_1 and C_2 s.t. R' is a ground instance of R.*

Proof. Let $C'_1 = C_1\lambda_1$ and $C'_2 = C_2\lambda_2$, and $\langle D'_1, D'_2, A\rangle$ be the key triple for R with $D'_1 \subseteq C'_1$ and $D'_2 \subseteq C'_2$. D'_1 and D'_2 are ground, so the corresponding mgu is the empty substitution, $D'_1 = \{L'^v\}$ and $D'_2 = \{L'^w\}$. Let D_1 and D_2 be maximal subsets of C_1 and C_2, respectively, s.t. $D_1\lambda_1 = D'_1$ and $D_2\lambda_2 = D'_2$. Since $\lambda_1\lambda_2$ is a unifier of $\text{at}(D_1 \cup D_2)$, we have a key triple $\langle D_1, D_2, \text{at}(D_1 \cup D_2)\rangle$ with mgu σ. Since σ is more general than $\lambda_1\lambda_2$ we have that R' is an instance of $R = (C_1 \setminus D_1)\sigma \cup (C_2 \setminus D_2)\sigma$. ∎

2.5.7. *Remark* Note that the proof of the lifting lemma given in Chang and Lee [1973] is flawed, as has been pointed out by Leitsch [1989]. This is also the reason why we have followed Robinson's [1965] original formulation of resolution with key triples and implicit factoring as the starting point for the generalization to the many-valued case.

2.5.8. Theorem (Completeness) *If* $\mathcal{C}$ *is unsatisfiable then* $\mathcal{C} \vdash \Box$.

Proof. By Proposition 2.4.5, the semantic tree T for $A(\mathcal{C})$ of Example 2.4.3 is closed w.r.t. $\mathcal{C}$, and hence (by Proposition 2.4.6), the corresponding reduced tree $R(T)$ is finite. We prove, by induction on the number of nodes in $R(T)$, that there is a resolution deduction of $\Box$ from $\mathcal{C}$.

If $R(T)$ has only one node, then $\Box$ must be in $\mathcal{C}$, since no other clause is falsified at the root of a semantic tree. Otherwise $R(T)$ contains a node X, s.t. its immediate successors X_1, X_2 are failure nodes. The edges leaving X are labeled by $\{A^{u_1}\}$ and $\{A^{u_2}\}$, $u_1 \neq u_2$. There are ground instances C'_1 and C'_2 of clauses C_1 and C_2 in $\mathcal{C}$ s.t. $I(X_1)$ falsifies C'_1 and $I(X_2)$ falsifies C'_2, but $I(X)$ does not falsify either . Therefore, C'_1 must contain a literal A^{u_1} and C'_2 must contain a literal A^{u_2}. Now $\langle\{A^{u_1}\}, \{A^{u_2}\}, \{A\}\rangle$ is a key triple for C'_1 and C'_2 with mgu id. We obtain the resolvent $R' = (C'_1 \setminus \{A^{u_1}\}) \cup (C'_2 \setminus \{A^{u_2}\})$. R' fails at X, since otherwise there would be some literal $L \in R'$ with $L \notin I(X)$. Assume $L \in C'_1 \setminus \{A^{u_1}\}$. Then $L \notin I(X_1)$, contradicting the assumption that C'_1 fails at X_1. Similarly for $L \in C'_2 \setminus \{A^{u_2}\}$.

By the lifting lemma, there is a resolvent R of C_i and C_j, s.t. R' is an instance of R. T is also closed w.r.t. $\mathcal{C} \cup \{R\}$, and the tree $R(T)'$ obtained from $R(T)$ by pruning at the first node that falsifies R' (at worst, this is X) is a proper subtree of $R(T)$ containing all failure nodes. By induction hypothesis, there is a resolution deduction of $\Box$ from $\mathcal{C} \cup \{R\}$. ∎

2.5.9. Proposition *Any resolution deduction R from $\mathcal{C}$ can be rewritten as a resolution deduction R' in tree form where no variable renaming is necessary in the resolution steps, possibly using renamed copies (variants) of clauses in $\mathcal{C}$.*

Proof. By induction on the length l of $R = C_1, \dots, C_l$:

$l = 1$: $C_1 \in \mathcal{C}$. This proof is in tree form, $R' = R$.

$l > 1$: C_l is the resolvent of clauses C_i, C_j where $i, j < l$. The deductions of C_i and C_j from $\mathcal{C}$ have length $< l$. By induction hypothesis, they have tree like deductions R_1, R_2. Then the deduction R'

$$\dfrac{\begin{array}{cc} \vdots\, R_1 & \vdots\, R_2 \\ C_1 & C_2 \end{array}}{C_l}$$

is in tree form.

If we take vaiable-disjoint renamings of *all* topmost clauses and carry the renamings through to the conclusion, then all resolved clauses are variable-disjoint. ■

Being in tree form means that every clause occurence in R is used only once in a resolution step. R can therefore be written as a tree, where the nodes are the clauses in R. This tree has a well-defined height. We call resolution proofs of the form given in the preceding proposition *regular*. If R contains only ground clauses, then the corresponding proof in tree form uses only literal copies of the clauses in R.

2.5.10. DEFINITION The *cumulative substitution* $\rho(R)$ of a regular resolution proof is defined by induction on the height h of R:

$h = 1$: $\rho(R) = id$.

$h > 1$: R ends in a resolution of the form

$$\dfrac{\begin{array}{cc} \vdots\, R_1 & \vdots\, R_2 \\ C_1 & C_2 \end{array}}{C}$$

with mgu σ. C_1 and C_2 are variable-disjoint. $\rho(R) = (\rho(R_1)\rho(R_2))\sigma = (\rho(R_1) \cup \rho(R_2))\sigma$.

Regular resolution proofs have the property that they can be grounded, as the following lemma shows. This property will be used in the next section.

2.5.11. LEMMA *Let R be a regular resolution deduction of C from $\mathcal{C}$. There is a substitution γ which grounds R. That is, γ is s.t. $R\gamma$ contains only ground clauses and $R\gamma$ is a resolution deduction of $C\gamma$ from $\mathcal{C}\gamma$.*

Proof. Let var(R) be the set of all variables occurring in R. We define a partition of var(R) $= V_1 \cup \ldots \cup V_p$ by the equivalence relation $x \sim y \iff x\rho(R) = y\rho(R)$. Let $c_1, \ldots, c_p$ be p new, distinct, constant symbols, and λ the substitution mapping members of V_i to c_i. Then $\gamma = \rho(R)\lambda$ grounds R, if $R\rho(R)$ is a deduction of $C\rho(R)$ from $\mathcal{C}\rho(R)$. But this follows from the definition of $\rho(R)$. ■

2.5.12. *Remark* The so-called *ground projection* is a standard notion in the resolution literature. Cumulative substitutions exist for all (not only regular) resolution proofs which have ground projections. See BAAZ and LEITSCH [1992] for results relating proof complexity and ground projections. Their Theorem 3.6 also shows that the assumption of regularity in Lemma 2.5.11 is necessary.

2.6 Negative Resolution

2.6.1. DEFINITION A *negative clause* $\mathcal{C}$ is a set of negations of signed atoms.

The *positive translation* $\mathcal{C}^+$ of a set of negative clauses $\mathcal{C}$ is the set of clauses $\mathcal{C}$ obtained by replacing every negative clause $\{\neg A_1{}^{w_1}, \ldots, \neg A_n{}^{w_n}\}$ by $\bigcup_{j=1}^{n}\{A_j{}^{v_i} \mid v_i \neq w_j\}$. If $\mathcal{D} = \mathcal{C}^+$, then we define $\mathcal{C} = \mathcal{D}^-$.

The sfe corresponding to a negative clause and universal satisfiability are defined analogous to the positive case. It is obvious that a set of negative clauses is satisfiable iff its positive translation is. For this notion of clause the appropriate notion of resolution is as follows:

2.6.2. DEFINITION Let C_1, C_2, ..., C_m be negative clauses s.t. $\bigcap_{i=1}^{m} \mathrm{var}(C_i) = \emptyset$, and let $\neg D_i \subseteq C_i$ be such that every literal in $\neg D_i$ is signed with the truth value v_i, If $A = \mathrm{at}(\bigcup_{i=1}^{m} \neg D_i)$ is unifiable with mgu σ, then $\langle \neg D_1, \ldots, \neg D_m, A\rangle$ is called an *negative key tuple (n-key tuple) with mgu* σ for C_1, C_2, ..., C_m.

2.6.3. DEFINITION Let C_1', C_2', ..., C_m' be clauses, let C_1, C_2, ..., C_m be variable disjoint renamings, respectively, and let $\langle D_1, \ldots, D_m, A\rangle$ be an n-key tuple with mgu σ for $C_1, \ldots, C_m$. Then the clause $(C_1 \setminus D_1)\sigma \cup \ldots \cup (C_m \setminus D_m)\sigma$ is a *resolvent* of C_1', ..., C_m'.

Negative resolution will play a part in the cut-elimination theorem for negative sequent calculus in the next chapter. We need the following theorem:

2.6.4. THEOREM (Soundness) *If $\mathcal{C} \vdash \Box$ then $\mathcal{C}$ is unsatisfiable.*

Proof. We prove that for every interpretation $\mathbf{H}$ satisfying $\mathcal{C}$, if $\mathcal{C} \vdash_n C$ then $\mathbf{H}$ satisfies $\mathcal{C}$. By induction on the height h of the derivation tree of C from $\mathcal{C}$: If $h = 1$, then $C \in \mathcal{C}$ and hence is satisfied by every interpretation which satisfies $\mathcal{C}$. If $h > 1$, then let

$$\frac{C_1 \quad C_2 \quad \ldots \quad C_m}{C}$$

be the last resolution in the derivation, and let $\langle \neg D_1, \ldots, \neg D_m, A\rangle$ be its n-key tuple with mgu σ. Since D_1, ..., D_m are unifiable, we can simulate the step by a sequence of positive resolutions on the positive translations: Let $\neg C_i = \neg E_i \cup \neg D_i$, and let $\bar{C}_i$, $\bar{E}_i$ and $\bar{D}_i$ be the positive translations of $\neg C_i$, $\neg E_i$ and $\neg D_i$, respectively. By $\bar{D}_i^v$ we denote those elements of $\bar{D}_i$ with sign v. We show that there is a positive resolution derivation of $\bar{C}$ from $\{\bar{C}_1, \ldots, \bar{C}_m\}$ or, equivalently, a deduction of $\Box$ from $\{\bar{D}_1, \ldots, \bar{D}_m\}$.

In general, call a set of clauses $\mathcal{F} = \{F_1, \ldots, F_{l+1}\}$ *contractible* iff $\mathcal{F}$ is unifiable, for every F_i there is a set $W_i = \{w_1, \ldots, w_{m-l}\}$ of truth values s.t. $w_j \in W_i$ iff $F_i^{w_j}$ is empty, and for every $1 \leq i \leq l+1$ there are truth values $\overline{w}_i$, $\underline{w}_i \in W_i$ (not necessarily distinct) s.t. for all $j > i$, $\overline{w}_i \notin W_j$ and for all $k < i$, $\underline{w}_i \notin W_k$.

It is easy to see that $\{\bar{D}_1, \ldots, \bar{D}_m\}$ is contractible with $l = m-1$, $W_i = \{v_i\}$, $\overline{w}_i = \underline{w}_i = v_i$. We show by induction on l that $\Box$ is derivable by positive resolution from a contractible set $\mathcal{F}$:

$l = 0$: This means $\mathcal{F} = \{\Box\}$. (Note that $\{\Box\}$ is contractible.)

$l > 0$: We have $\mathcal{F} = \{F_1, \ldots, F_{l+1}\}$. Let $1 \leq i \leq l$ and consider F_i and F_{i+1}. $F_i^{\overline{w}_{i+1}}$ and $F_{i+1}^{\underline{w}_i}$ are nonempty. Hence we have that F_i and F_{i+1} have a resolvent G_i with key triple $\langle F_i^{\overline{w}_{i+1}}, F_{i+1}^{\underline{w}_i}, at(F_i^{\overline{w}_{i+1}} \cup F_{i+1}^{\underline{u}_i})\rangle$. The set of clauses $\mathcal{G} = \{G_1, \ldots, G_l\}$ is contractible: For G_i, the corresponding set W_i' is $W_i \cup W_{i+1}$ having $m - l + 1$ elements, $\overline{u'}_i = \overline{u}_i$ and $\underline{w'}_i = \underline{w}_{i+1}$.

The following diagram illustrates the case of $m = 4$.

l	$m - l$	$\neg A^1$		$\neg A^2$		$\neg A^3$		$\neg A^4$
3	1	A^2, A^3, A^4 $\underline{\overline{1}}$		A^1, A^3, A^4 $\underline{\overline{2}}$		A^1, A^2, A^4 $\underline{\overline{3}}$		A^1, A^2, A^3 $\underline{\overline{4}}$
2	2		A^3, A^4 $\overline{1}, 2$		A^1, A^4 $\overline{2}, \underline{3}$		A^1, A^2 $3, \underline{4}$	
1	3			A^4 $\overline{1}, 2, 3$		A^1 $2, 3, \underline{4}$		
0	4				$\Box$ $1, 2, 3, 4$			

Note how the lists of signed A's form a tree-like resolution deduction of the empty clause. ∎

2.6.5. LEMMA (Lifting Lemma) *Let C_1', C_2', …, C_m' be ground instances of clauses C_1, C_2, …, C_m, respectively, and let C' be a negative resolvent of C_1', C_2', …, C_m'. Then there is a negative resolvent C of C_1, C_2, …, C_n s.t. C' is a ground instance of C.*

Proof. Analogous to the positive case. ∎

2.6.6. LEMMA *Let R' be a ground proof of C' from $\mathcal{C}'$, where $\mathcal{C}'$ is a ground instance of $\mathcal{C}$. Then there is a proof R of C from $\mathcal{C}$, s.t. C' is a ground instance of C.*

Proof. By induction on the height h of R':

$h = 1$: Then $C' \in \mathcal{C}'$, and hence R consist only of the clause C.

$h > 1$: R' ends in a resolution step from the clauses C_1', …, C_m'. If C_i' is a topmost clause, then it is a ground instance of some clause $\mathcal{C}_i \in \mathcal{C}$. Otherwise, C_i' is the last clause in a resolution deduction R_i'. By induction hypothesis, there is a resolution deduction R_i of C_i from $\mathcal{C}$ where C_i' is a ground instance of C_i. From the lifting lemma, we know that there is some C of which C' is a ground instance, s.t. C is a resolvent of C_1, …, C_m. ∎

2.6.7. THEOREM (Completeness) *If $\mathcal{C}$ is unsatisfiable then $\mathcal{C} \vdash \Box$.*

Proof. Since $\mathcal{C}^+$ is unsatisfiable, there is a regular resolution deduction R of $\Box$ from $\mathcal{C}^+$ of length l. By Lemma 2.5.11, R can be grounded, resulting in a ground proof R' of $\Box$ from a ground instance $\mathcal{C}'^+$. We translate this deduction into a negative deduction $\nu(R')$ of $\Box$ by induction on the height h or R':

$h = 1$: Then $\Box \in \mathcal{C}$.

$h > 1$: We proceed in three stages: (1) First, we *mark* those literals throughout the proof, which result in the last literals resolved upon (in a sense to be made more precise). This results in a proof where some groups of literals in the initial clauses are marked. Every such group is the positive translation of one negative literal in the original set of negative clauses. (2) The proof is *pruned* by omitting each of these groups from the initial clauses, one at a time. This results in several proofs of $\Box$ of height less than h from subclauses of the initial clauses (all of which are positive translations of subclauses of the original negative clauses). (3) The induction hypothesis is applied to these proofs, yielding negative proofs of $\Box$ from subclauses of the original negative clauses. (4) We add the negative translations of the groups that were left out in step (2), obtaining several negative proofs of clauses of the form $\{\neg P^v\}$. The set of all these clauses is inconsistent, since there is a subproof of R' of $\Box$ from their positive translations. Hence there is one such clause for each $v \in V$, and furthermore, P is of the same form in all of these clauses. (5) By resolving, we obtain a negative ground proof of $\Box$ from ground instances of $\mathcal{C}$. By Lemma 2.6.6, there is a negative resolution proof of $\Box$ from $\mathcal{C}$.

(1) Let R' be as follows:

$$\frac{\begin{array}{c}\vdots\, R'_1 \\ \{A^{v_i}\}\end{array} \qquad \begin{array}{c}\vdots\, R'_2 \\ \{A^{v_j}\}\end{array}}{\Box}$$

We introduce an indexing system of literals in R'. The initial clauses $\mathcal{C}'^+$ are the positive translations of negative clauses in $\mathcal{C}'$. We provide every literal in $\mathcal{C}'$ by an index s.t. no two occurrences of literals in $\mathcal{C}'$ have the same index. If $\neg B^w \in C' \in \mathcal{C}'$ and the index of $\neg B^w$ is x, then we write the indexed literal as $\neg_x B^w$. The indexes are extended to literals in the positive translation of $\mathcal{C}$: If $\bigvee_{v \in V \setminus w} B^v$ is the positive translation of $\neg_x B^w$, then all the B^v are indexed by x, too. So, if $\{\neg_{x_1} A_1^{w_1}, \ldots, \neg_{x_n} A_n^{w_n}\} \in \mathcal{C}'$, then

$$\{_{x_1}A_1^v \mid v \in V \setminus w_1\} \cup \ldots \cup \{_{x_n}A_n^v \mid v \in V \setminus w_n\} \in \mathcal{C}'^+$$

This way, all literals in R' are marked.

Define the operator $u(S)$ on indexed ground proofs S inductively as follows: If S is only an initial clause, then $u(S) = S$. Otherwise, S ends in a resolution step:

$$\frac{\begin{array}{c}\vdots\, S_1 \\ D_1 \cup \{_xA^{w_1}\}\end{array} \qquad \begin{array}{c}\vdots\, S_2 \\ D_2 \cup \{_yA^{w_2}\}\end{array}}{D_1 \cup D_2}$$

let $D'_i \cup \{_{x'}A^{v_i}\}$ be the last clause of $u(S_i)$ $(i = 1, 2)$. Then $u(S)$ is

$$\frac{\begin{array}{c}\vdots\, u(S_1) \\ D'_1 \cup \{_{x'}A^{w_1}\}\end{array} \qquad \begin{array}{c}\vdots\, u(S_2)[x/y] \\ D'_2[x/y] \cup \{_{x'}A^{w_2}\}\end{array}}{D'_1 \cup D'_2[x/y]}$$

where $[x/y]$ means that the index y is replaced everywhere by the index x.

Furthermore, we introduce an operator $d(S, \mathcal{C}, X)$ which is defined for an indexed ground proof $u(S)$, a set of initial clauses $\mathcal{C}$ and a set of indices X as follows:

$h = 1$: $u(S)$ is only the initial clause D: if $D \in C$, then $d(S, X) = D^*$, where D^* is the clause obtained from D by omitting every literal indexed by some index $x \in X$. Otherwise $d(S, D, X) = D$.

$h > 1$: $u(S)$ is of the form

$$\dfrac{\begin{array}{cc} \vdots\, S_1 & \vdots\, S_2 \\ D_1 \cup \{{}_yA^{w_1}\} & D_2 \cup \{{}_yA^{w_2}\} \end{array}}{D_1 \cup D_2}$$

where $C = D_1 \cup D_2$. Let D_i^* be the last clause in $d(S_i, C, X)$ $(i = 1, 2)$. If ${}_yA^{w_1} \notin D_1^*$, then $d(S, C, X) = d(S_1, C, X)$. If ${}_yA^{w_2} \notin D_2^*$, then $d(S, C, X) = d(S_2, C, X)$. Otherwise, let $d(S, X)$ be

$$\dfrac{\begin{array}{cc} \vdots\, d(S_1, C, X) & \vdots\, d(S_2, C, X) \\ D_1^* & D_2^* \end{array}}{D_1^* \setminus \{{}_yA^{w_1}\} \cup D_2^* \setminus \{{}_yA^{w_2}\}}$$

Intuitively, u specifies a relation among literal occurrences in the indexed proof S where all literals are either related by having the same negative literal as their common ancestor, or are related to literals which are resolved upon in the proof. The operator s then is the pruning operator, parametric on a set of inices X. Given an indexed proof S, $d(S, \mathcal{C}, X)$ is the proof resulting from S by deleting all literals indexed by members of X from the initial clauses $\mathcal{C}$ and all resolutions acting on such literals.

(2) We have indexed the proofs R_1' and R_2', resulting in indexed proofs of $\{{}_xA^{v_i}\}$ and $\{{}_xA^{v_j}\}$, respectively. Let $C_{1,1}, \ldots, C_{1,p}$ and $C_{2,1}, \ldots, C_{2_q}$ be the initial clauses of R_1 and R_2 respectively. By applying d, we obtain proofs $d(R_1', \{C_{1,i}\}, \{x\})$ $(1 \le i \le p)$ and $d(R_2', \{C_{2,j}\}, \{x\})$ $(1 \le j \le q)$ with $\Box$ as their last clause. (3) These proofs are of length $< h$, hence the induction hypothesis applies. This yields negative resolution proofs $N_{1,i}$ and $N_{2,j}$ of $\Box$ from initial clauses among $\{D_{1,i}^-\}_i$ and $\{D_{2,j}^-\}_j$, respectively. In $N_{r,s}$, the initial clause $D_{r,k}^-$ is the negative translation of $C_{r,k}$ if $k \ne s$ and the negative translation of $d(C_{r,k}, C_{r,k}, \{x\})$ otherwise.

(4) By adding the negative literals $\neg L_{r,s}^{v_{r,s}}$ omitted from clause $C_{r,s}$ in $N_{r,s}$ again we obtain proofs $N_{r,s}'$ of $\{\neg L_{r,s}^{v_{r,s}}\}$. Now consider the original proof R'. Its pruning $d(R', \mathcal{C}, I \setminus \{x\})$ is a proof of $\Box$ from the positive translations of the clauses in $\mathcal{N} = \{\{\neg L_{r,s}^{v_{r,s}}\} \mid r, s\}$. This ensures that all $L_{r,s}$ are actually equal (i.e., unifiably with unifier id). Furthermore, the set of all $v_{r,s}$ is the whole set of truth values V. For if there were some $w \in V$ which is not among the $v_{r,s}$, the structure which makes $L_{r,s}$ take the value w satisfies $\mathcal{N}^+$. But this is impossible, since $\mathcal{N}$ is unsatisfiable.

(5) Thus, we obtain a negative resolution proof R'^- of $\Box$ from $\mathcal{C}'$. By Lemma 2.6.6, there is a negative resolution proof of $\Box$ from $\mathcal{C}$. ∎

Chapter 3

Sequent Calculus

3.1 Introduction

Sequent calculus was introduced by GENTZEN [1934] for classical and intuitionistic logic and has since proved to be an important formalism for proof theoretic studies of formal systems. Gentzen's sequents are expressions of the form $\Gamma \to \Delta$, where Γ and Δ are finite sequences of formulas. The full calculus consists of axioms of the form $A \to A$, rules for introducing propositional connectives resp. quantifiers in the left and right sides of the sequent, as well as structural rules (among them the *cut rule)*.

The sequent notation can be interpreted in two different ways: firstly, as expressing *entailment*. A sequent $A \to B$ stands for: A entails B, or more generally: $\Gamma \to \Delta$ stands for: the conjunction of the formulas in Γ entails the disjunction of the formulas in Δ. Secondly, the sequent may be interpreted *truth-functionally:* either one of the formulas in Γ is false or one of the formulas in Δ is true. For classical logic, these two interpretations coincide, since the truth-functional characterization is equivalent to the derivability of $\bigwedge \Gamma \supset \bigvee \Delta$, and this, by the deduction theorem, to $\bigwedge \Gamma \vdash \bigvee \Delta$.

For many-valued logics, however, these notions are in general distinct, and we have two fundamental alternatives as to what we mean by "sequent calculus for a many-valued logic": Firstly, we can seek to formalize the entailment relation (or, if we have a complete axiomatization of a given many-valued logic, of derivability) as a sequent calculus. AVRON [1992] has developed a general theory of consequence relations for logics, and has also applied this to the study of 3-valued logics in his [1991], where he gives a calculus of *hyper*sequents for Łukasiewicz logic.

Here, we will take the second approach via truth-functionality, a property every many-valued logic has. For this, we need a sequent with one place for every truth value in the logic: $\Gamma_{\mathbf{1}} \mid \Gamma_{\mathbf{2}} \mid \ldots \mid \Gamma_{\mathbf{m}}$, where m is the number of truth values. We interpret this sequent as true under a given interpretation iff one of the $\Gamma_{\mathbf{i}}$ contains a formula that takes the value v_i. From this point of view, we then see axioms as expressing the fact that every formula must take at least one truth value, and introduction rules as characterizing the truth value of a composite formula in terms of its subformulas. Yet we may also take another, in a sense *dual* standpoint: A sequent is true under an interpretation, if one of the $\Gamma_{\mathbf{i}}$ contains a formula that *does not* take the value v_i. Then, an axiom comes to state the fact that a formula may take *at most* one truth value. An introduction rule can be viewed as a rule for how to falsify a composite formula by falsifying a subformula. This second interpretation corresponds, in the classical case, exactly to the semantic tableaux of BETH [1955]. We call the first interpretation *positive*, the second *negative*.

These two perspectives give rise to two different, but closely related, sets of calculi for a many-valued logic—not only two sequent calculi, but also two tableau systems, two natural deduction systems, and two clause translation calculi (for positive and negative resolution, respectively). This correspondence has been described in BAAZ *et al.* [1993b].

This approach has often been used in the literature, and people have reinvented the wheel over and over again. The works differ in notation, and in degrees of generality. Some have used the positive, some the negative approach.

The first to introduce a calculus of sequents for many-valued logics (for the propositional case) seems to have been SCHRÖTER [1955]. He used the negative approach. His method of axiomatization has also been given treatment in the textbook of GOTTWALD [1989].

Seemingly independent, ROUSSEAU [1967] treated sequent calculus for first-order finitely-valued logics in full generality and also gave soundness and completeness proofs. In ROUSSEAU [1970] he investigated generalizations of *intuitionistic* sequent calculus **LJ** to the many-valued case (with a concept of sequent distinct from the one in his [1967]). TAKAHASHI [1967a] considers a sequent calculus for first-order many-valued logics with universal and existential quantifiers. Interpolation and definability theorems are proven using this calculus by MIYAMA [1974], and HANAZAWA and TAKANO [1985]. In TAKAHASHI [1970] a very general Gentzen-style proof theory for *continuous logics*, i.e., many-valued logics over a topological space as the set of truth values, has been developed. The model theory of continuous logics has been worked out in CHANG and KEISLER [1966]. OHYA [1967] extended the formalism of TAKAHASHI [1967a] to a system of simple type theory, for which he also shows cut-elimination along the lines of TAKAHASHI [1967b].

After Schröter's work, the negative approach has been taken also by BOROWIK [1985] for the propositional case only. Earlier, BECCHIO and PABION [1977] have outlined a negative sequent calculus for three-valued Łukasiewicz logic. A tableau system for m-valued Łukasiewicz logic was given in SUCHOŃ [1974]. Similar work has been done on equivalent tableau systems starting with SURMA [1977] and CARNIELLI [1987b]. Recently, CARNIELLI [1991] has presented a notational variant of our negative sequent calculi, based on his work on tableau calculi for many-valued logics in [1987b]. HÄHNLE [1992, 1993a] presents an in-depth study of many-valued tableaux systems. He introduces several refinements such as sets-as-signs, and investigates classes of many-valued logics (e.g., so-called normal logics) where rules can be efficiently and uniformly represented.

Several other papers deal with sequent calculi for specific many-valued logics, in particular Post logic. KIRIN [1966] has given sequent calculi based on semantic considerations in Post algebras (see also his [1968]). Similar work was done by RASIOWA [1972], SALONI [1972], and PRZYMUSIŃSKA [1980a].

3.2 Semantics of Sequents

3.2.1. DEFINITION An *(m-valued) sequent* Γ is an m-tuple $\Gamma_{\mathbf{1}} \mid \ldots \mid \Gamma_{\mathbf{m}}$of finite sequences $\Gamma_{\mathbf{i}}$ of formulas. If Γ is a sequent, then $\Gamma_{\mathbf{i}}$ denotes the i-th component of Γ.

If Δ is a sequence of formulas and $I \subseteq M = \{1, \ldots, m\}$ (or $W \subseteq V$), we denote by $[I{:}\,\Delta]$ ($[W{:}\,\Delta]$) the sequent whose i-th component is Δ if $i \in I$ ($v_i \in W$), and is empty otherwise. For $[\{i_1, \ldots, i_k\}{:}\,\Delta]$ ($[\{w_1, \ldots, w_k\}{:}\,\Delta]$) we write $[i_1, \ldots, i_k{:}\,\Delta]$ ($[w_1, \ldots, w_k{:}\,\Delta]$). If Γ and Γ' are sequents, then we write Γ, Γ' for the sequent $\Gamma_{\mathbf{1}}, \Gamma'_{\mathbf{1}} \mid \ldots \mid \Gamma_{\mathbf{m}}, \Gamma'_{\mathbf{m}}$.

3.2.2. DEFINITION Let $\mathbf{I}$ be an interpretation. $\mathbf{I}$ *p-satisfies (n-satisfies)* a sequent Γ iff there is an i ($1 \le i \le m$) s.t., for some formula $F \in \Gamma_{\mathbf{i}}$, $\mathrm{val}_{\mathbf{I}}(F) = v_i$ ($\mathrm{val}_{\mathbf{I}}(F) \neq v_i$). $\mathbf{I}$ is called a *p-model (n-model)* of Γ and we write $\mathbf{I} \models^{\mathrm{p}} \Gamma$ ($\mathbf{I} \models^{\mathrm{n}} \Gamma$).

Γ is called *p-satisfiable (n-satisfiable)* iff there is an interpretation $\mathbf{I}$ s.t. $\mathbf{I} \models^{\mathrm{p}} \Gamma$ ($\mathbf{I} \models^{\mathrm{n}} \Gamma$), and *p-valid (n-valid)* iff for every interpretation $\mathbf{I}$, $\mathbf{I} \models^{\mathrm{p}} \Gamma$ ($\mathbf{I} \models^{\mathrm{n}} \Gamma$).

Evidently, these two notions of satisfiability resp. validity are distinct: an interpretation may be a p-model of a sequent, but not an n-model; a sequent may be p-satisfiable but not n-satisfiable; it may be p-valid but not n-valid (or vice versa).

3.2.3. EXAMPLE To see how the notion of many-valued sequent is a generalization of the classical case, consider **PL** with $V = \{f, t\}$: A sequent takes the form $\Gamma \mid \Delta$ (usually written $\Gamma \rightarrow \Delta$). It is p-satisfied if some formula in Γ is false or some formula in Δ is true. It is n-satisfied if some formula in Γ is true or some formula in Δ is false. The calculus resulting from p-satisfaction is the calculus given by GENTZEN [1934], the one from n-satisfaction corresponds to BETH's 1955 tableaux. If we exchange t and f, then we also obtain Gentzen's system from n-sequents (cf. CARNIELLI [1991], p. 66). However, in the general case the correspondence is not one of "trivial" duality as in the classical case: you cannot in general pass from the positive to the negative system just by switching the sides of the sequents.

We can employ the metaformalism of sfes introduced in Chapter 1 to study the connection between p- and n-sequent calculi. The notions of satisfaction and validity of p- and n-sequents can be straightforwardly translated by interpreting sequents as sfes of a certain form: A sequent Γ corresponds to the sfe

$$\mathrm{psfe}(\Gamma) = \bigvee\!\!\!\!\bigvee_{A \in \Gamma_{\mathbf{1}}} A^{v_1} \vee\!\!\!\vee \ldots \vee\!\!\!\vee \bigvee\!\!\!\!\bigvee_{A \in \Gamma_{\mathbf{m}}} A^{v_m}$$

modulo p-satisfiability and to

$$\mathrm{nsfe}(\Gamma) = \bigvee\!\!\!\!\bigvee_{A \in \Gamma_{\mathbf{1}}} \neg A^{v_1} \vee\!\!\!\vee \ldots \vee\!\!\!\vee \bigvee\!\!\!\!\bigvee_{A \in \Gamma_{\mathbf{m}}} \neg A^{v_m}$$

modulo n-satisfiability. Note how the first sfe-translation of a sequent corresponds to a positive clause as in Definition 2.2.1, and of the second to a negative clause as in Definition 2.6.1.

We can now apply the metatheory of sfes to sequents. P-satisfiability corresponds to reading a sequent as a positive disjunction, and n-satisfiability to reading it as a negative disjunction of signed formulas. Call the translation of a sequent to a disjunction of positive signed formulas a *p-sequent*, and the translation to a negative disjunction an *n-sequent*. In light of Proposition 1.4.6, any sfe can be expressed as a conjunction of p- or n-sequents, and a conjunction of p-sequents (n-sequents) can be transformed to an equivalent conjunction of n-sequents (p-sequents) by Proposition 1.4.5. The negation of a p-sequent (n-sequent) is equivalent to a conjunction of n-sequents (p-sequents). Proposition 1.4.8 provides us with a relation between validity of a sequent and the unsatisfiability of its negation. For the special case where the sequent contains only one formula $i\colon A$ we obtain the following:

3.2.4. PROPOSITION *A sequent $[i\colon A]$ is p-unsatisfiable (n-unsatisfiable) iff it is n-valid (p-valid).*

Proof. The negation of the p-sequent A^{v_i} is the n-sequent $\neg A^{v_i}$. The claim follows from Proposition 1.4.8. ∎

On the other hand, the n-sequent $\neg A^{v_i}$ can also be written as a p-sequent $\bigvee_{j\neq i} A^{v_j}$, and hence the p-unsatisfiability of $[i\colon A]$ can be established by proving $[V \setminus \{v_i\}\colon A]$ p-valid. Proposition 3.2.4 gives us as an alternative to proving the latter sequent in order to establish the unsatisfiability of $[i\colon A]$. Instead, we can use a complete (w.r.t. n-validity) calculus for proving $[i\colon A]$ n-valid and, hence, p-unsatisfiable. This is of value especially if there is only one designated truth value:

3.2.5. PROPOSITION *Let F be a formula. Then the following are equivalent:*

(1) F is a tautology

(2) The sequent $[V^+\colon F]$ is p-valid

(3) The sequents $[j\colon F]$, where $j \in V \setminus V^+$, are all n-valid.

3.2.6. PROPOSITION *Let F be a formula. Then the following are equivalent:*

(1) F is a unsatisfiable

(2) The sequent $[V \setminus V^+\colon F]$ is p-valid

(3) The sequents $[j\colon F]$, where $j \in V^+$, are all n-valid.

3.3 Construction of Sequent Calculi

3.3.1. DEFINITION Let $\bigwedge_{j\in I} \Delta_{\Box:i}(j)$ be a positive (negative) i-th partial normal form of $\Box(A_1,\ldots,A_n)$ ($\neg\Box(A_1,\ldots,A_n)$), where $\Box$ is of arity n, $\Delta_{\Box:i}(j)$ is a disjunction of signed atoms (literals) of the form ${A_i}^w$ ($\neg{A_i}^w$), where $1 \le i \le n$, and let $\Delta'_{\Box:i}(j)$ be the sequent corresponding to $\Delta_{\Box:i}(j)$.

A *p- (n-) introduction rule for* $\Box$ *at place* i is a schema of the form:

$$\frac{\langle \Gamma, \Delta'_{\Box:i}(j)\rangle_{j\in I}}{\Gamma, [i\colon \Box(A_1,\ldots,A_n)]}\ \Box{:}i$$

3.3.2. EXAMPLE Consider the implication in three-valued Łukasiewicz logic **Ł**$_3$ given in Example 1.3.3. The partial normal forms from Example 1.5.3 yield the following introduction rules for **PŁ**$_3$:

$$\frac{\Gamma \mid \Delta \mid \Pi, A \quad \Gamma, B \mid \Delta \mid \Pi}{\Gamma, A \supset B \mid \Delta \mid \Pi}\ \supset{:}f$$

$$\frac{\Gamma \mid \Delta, A \mid \Pi, A \quad \Gamma \mid \Delta, A, B \mid \Pi \quad \Gamma, B \mid \Delta \mid \Pi, A}{\Gamma \mid \Delta, A \supset B \mid \Pi}\ \supset{:}p$$

$$\frac{\Gamma, A \mid \Delta, A \mid \Pi, B \quad \Gamma, A \mid \Delta, B \mid \Pi, B}{\Gamma \mid \Delta \mid \Pi, A \supset B}\ \supset{:}t$$

and the following rules for **NŁ**$_3$:

$$\frac{\Gamma \mid \Delta \mid \Pi, A \quad \Gamma, B \mid \Delta \mid \Pi, A}{\Gamma, A \supset B \mid \Delta \mid \Pi}\ \supset{:}f$$

$$\frac{\Gamma, A \mid \Delta, A \mid \Pi \quad \Gamma \mid \Delta, B \mid \Pi, A}{\Gamma \mid \Delta, A \supset B \mid \Pi}\ \supset{:}p$$

$$\frac{\Gamma, A \mid \Delta \mid \Pi \quad \Gamma \mid \Delta \mid \Pi, B \quad \Gamma \mid \Delta, A, B \mid \Pi}{\Gamma \mid \Delta \mid \Pi, A \supset B}\ \supset{:}t$$

3.3.3. DEFINITION Let $\bigwedge_{j\in I} \Delta_{\mathsf{Q}:i}(j)$ be a positive (negative) i-th partial normal form of $(\mathsf{Q}x)A(x)$ ($\neg(\mathsf{Q}x)A(x)$), where $\Delta_{\mathsf{Q}:i}(j)$ is a disjunction of signed atoms (literals) of the form $A(\alpha)^w$ or $A(\tau)^w$ ($\neg A(\alpha)^w$ or $\neg A(\tau)^w$), and let $\Delta'_{\mathsf{Q}:i}(j)$ be the sequent corresponding to $\Delta_{\mathsf{Q}:i}(j)$

A *p- (n-) introduction rule for* Q *at place* i is a schema of the form:

$$\frac{\left\langle \Gamma, \Delta'_{\mathsf{Q}:i}(j)\right\rangle_{j\in I}}{\Gamma, [i\colon (\mathsf{Q}x)A(x)]}\ \mathsf{Q}{:}i$$

where the free variables α occurring in the $\Delta'_{\mathsf{Q}:i}(j)$ satisfy the eigenvariable condition: No α occurs in the lower sequent.

In an introduction rule as above, the formula—$\Box(A_1,\ldots,A_n)$ or $(\mathsf{Q}x)A(x)$—being introduced is called the *main formula*, the formulas in $\Delta_{f:i}(j)$ are called the *auxiliary formulas* and the formulas in Γ are called the *side formulas*. The sequents above the line are called the *upper sequents* or *premises*, the sequent below the line is called the *lower sequent* or the *conclusion* of this rule.

3.3.4. EXAMPLE Consider the universal quantifier $\forall$ in the three-valued Łukasiewicz logic $\mathbf{Ł}_3$. From the pnfs in Example 1.5.7 we obtain the following rules:

$$\frac{\Gamma, A(\tau) \mid \Delta \mid \Pi, A}{\Gamma, (\forall x)A(x) \mid \Delta \mid \Pi}\ \forall{:}f \qquad \frac{\Gamma \mid \Delta \mid \Pi, A(\alpha)}{\Gamma \mid \Delta \mid \Pi, (\forall x)A(x)}\ \forall{:}t$$

$$\frac{\Gamma \mid \Delta, A(\alpha) \mid \Pi, A(\alpha) \quad \Gamma \mid \Delta, A(\tau) \mid \Pi}{\Gamma \mid \Delta, (\forall x)A(x) \mid \Pi}\ \forall{:}p$$

3.3.5. DEFINITION A *p-sequent calculus* **PL** for a logic **L** is given by:

(1) axiom schemas of the form: $[V{:}\, A]$,

(2) for every connective $\square$ and every truth value v_i a p-introduction rule $\square{:}i$,

(3) for every quantifier Q and every truth value v_i a p-introduction rule $\mathsf{Q}{:}i$,

(4) weakening rules for every place i:

$$\frac{\Gamma}{\Gamma, [i{:}\, A]}\ \text{w:}i$$

(5) exchange rules for every place i:

$$\frac{\Gamma, [i{:}\, B, A], \Delta}{\Gamma, [i{:}\, A, B], \Delta}\ \text{x:}i$$

(6) contraction rules for every place i:

$$\frac{\Gamma, [i{:}\, A, A]}{\Gamma, [i{:}\, A]}\ \text{c:}i$$

(7) cut rules for every two $i \neq j$:

$$\frac{\Gamma, [i{:}\, A] \quad \Delta, [j{:}\, A]}{\Gamma, \Delta}\ \text{cut:}ij$$

(2) and (3) are called *logical rules*, (4)–(7) are called *structural rules*.

3.3.6. DEFINITION An *n-sequent calculus* **NL** for a logic **L** is given by:

(1) axiom schemas of the form: $[i, j{:}\, A]$, where $i \neq j$,

(2) for every connective $\square$ and every truth value v_i an n-introduction rule $\square{:}i$,

(3) for every quantifier Q and every truth value v_i an n-introduction rule $\mathsf{Q}{:}i$,

(4) weakening rules for every place i:

$$\frac{\Gamma}{\Gamma, [i{:}\, A]}\ \text{w:}i$$

(5) exchange rules for every place i:

$$\frac{\Gamma, [i\colon B, A], \Delta}{\Gamma, [i\colon A, B], \Delta}\ \text{x:}i$$

(6) contraction rules for every place i:

$$\frac{\Gamma, [i\colon A, A]}{\Gamma, [i\colon A]}\ \text{c:}i$$

(7) the cut rule:

$$\frac{\Gamma_1, [1\colon A] \quad \ldots \quad \Gamma_m, [m\colon A]}{\Gamma_1, \ldots, \Gamma_m}\ \text{cut}$$

(2) and (3) are called *logical rules*, (4)–(7) are called *structural rules*.

3.3.7. Definition An upward tree P of sequents is called a *proof* in a sequent calculus **SL** iff every leaf is an instance of an axiom in **SL**, and all other sequents in it are obtained from the ones standing immediately above it by an application of one of the rules of **SL**. The sequent at the root of P is called its *end-sequent*. A sequent Γ is called *provable* in **SL** (in symbols: $\vdash^{\mathbf{SL}} \Gamma$) iff it is the end-sequent of some proof (in **SL**). In the case of **PL** (**NL**), we say that Γ is *p-provable (n-provable)*.

3.3.8. Theorem (Soundness for **PL**) *If a sequent is p-provable, then it is p-valid.*

Proof. By induction on the length l of a proof of a sequent Π.

$l = 1$: Π is an axiom. Since every formula takes some truth value, the disjunction over all truth values is always true.

$l > 1$: Π is the conclusion of a rule of inference. Let $\mathbf{I}$ be an interpretation. The induction hypothesis is that all upper sequents are p-valid, and in particular that each is satisfied by $\mathbf{I}$. We distinguish cases according to the rule J applied last:

(1) J is a weakening: $\mathbf{I} \models^{\mathrm{p}} \Pi$ since A implies A or B.

(2) J is an exchange: $\mathbf{I} \models^{\mathrm{p}} \Pi$ since disjunction is commutative.

(3) J is a contraction: $\mathbf{I} \models^{\mathrm{p}} \Pi$ since A or A implies A.

(4) J is a cut: The upper sequents are $\Gamma, [i\colon A]$ and $\Gamma', [j\colon A]$ ($i \neq j$), and $\Pi = \Gamma, \Gamma'$. Now either $\mathbf{I} \models^{\mathrm{p}} \Gamma$, in which case also $\mathbf{I} \models^{\mathrm{p}} \Gamma, \Gamma'$, or $\mathbf{I} \models^{\mathrm{p}} [i\colon A]$. But then $\mathbf{I} \not\models^{\mathrm{p}} [j\colon A]$ and hence $\mathbf{I} \models^{\mathrm{p}} \Gamma'$.

(5) J is an introduction rule for a propositional connective $\Box$ at place i: The upper sequents are instances $\Gamma, \Delta''_{\Box:i}(j)$ and $\Pi = \Gamma, [i\colon \Box(A_1, \ldots, A_n)]$. If $\mathbf{I} \models^{\mathrm{p}} \Gamma$, then also $\mathbf{I} \models^{\mathrm{p}} \Pi$. Otherwise, $\mathbf{I} \models \bigwedge\!\!\!\bigwedge \mathrm{psfe}(\Delta''_{\Box:i}(j))$. Hence, by Definitions 3.3.1 and 1.5.1 we have that $\mathbf{I} \models^{\mathrm{p}} [i\colon \Box(A_1, \ldots, A_n)]$.

(6) J is an introduction rule for a quantifier Q at place i: The upper sequents are instances $\Gamma, \Delta''_{\mathsf{Q}:i}(j)$, and $\Pi = \Gamma, [i\colon (\mathsf{Q}x)A(x)]$. If $\mathbf{I} \models^{\mathrm{p}} \Gamma$, then also $\mathbf{I} \models^{\mathrm{p}} \Pi$. Otherwise, $\mathbf{I} \models \bigwedge\!\!\!\bigwedge \mathrm{psfe}(\Delta''_{\mathsf{Q}:i}(j))$. Note that in $\Delta''_{\mathsf{Q}:i}(j)$ (and $\mathrm{psfe}(\Delta''_{\mathsf{Q}:i}(j))$) terms $t_1, \ldots, t_p$ take the place of the term variables and free variables $a_1, \ldots, a_q$ take the place of the eigenvariables. Since $a_1, \ldots, a_q$ do not occur in Γ, we have that for all $d_1, \ldots, d_q \in D$, there are $e_1, \ldots, e_p \in D$—namely $\Phi_{\mathbf{I}}(t_1[d_1/a_1, \ldots, d_q/a_q]), \ldots, \Phi_{\mathbf{I}}(t_p[d_1/a_1, \ldots, d_q/a_q])$—such that $\mathbf{I} \models \bigwedge\!\!\!\bigwedge \mathrm{psfe}(\Delta''_{\mathsf{Q}:i}(j)[d_1/a_1, \ldots, d_q/a_q, e_1/t_1, \ldots, e_p/t_p])$ (Note that the e_i may actually depend on the d_j, since the terms may contain eigenvariables). Hence, by Definitions 3.3.3 and 1.5.5, we have that $\mathbf{I} \models^{\mathrm{p}} [i\colon (\mathsf{Q}x)A(x)]$. ∎

3.3.9. THEOREM (Soundness for **NL**) *If a sequent is n-provable, then it is n-valid.*

Proof. The proof is analogous to the case for p-sequent calculus, simply replace $\models^{\mathrm{p}}$ by $\models^{\mathrm{n}}$ and $\mathrm{psfe}(\cdot)$ by $\mathrm{nsfe}(\cdot)$ throughout the proof. The cases of axioms and cut rules are handled as follows:

Π is an axiom, say $[i, j\colon A]$. Let $\mathbf{I}$ be an interpretation. If $\mathbf{I} \models \neg A^{v_i}$ or $\mathbf{I} \models \neg A^{v_j}$ then $\mathbf{I} \models^{\mathrm{n}} [i, j\colon A]$. Since A can take only one truth value, these are the only possible cases.

(4′) J is a cut: The upper sequents are $\Gamma_i, [i\colon A]$ $1 \le i \le m$, and $\Pi = \Gamma_1, \ldots, \Gamma_m$. Now either $\mathbf{I} \models^{\mathrm{n}} \Gamma_1$, in which case also $\mathbf{I} \models^{\mathrm{n}} \Pi$. Otherwise $\mathbf{I} \models^{\mathrm{n}} [1\colon A]$. But then $\mathbf{I} \not\models^{\mathrm{n}} [i\colon A]$ for all $2 \le i \le m$, hence $\mathbf{I} \models^{\mathrm{n}} \Gamma_i$ and *a fortiori* $\mathbf{I} \models^{\mathrm{n}} \Pi$. ∎

3.3.10. THEOREM (Completeness for **PL**) *If a sequent is p-valid, then it is p-provable without cuts from atomic axioms.*

Proof. We use the method of *reduction trees*, due to SCHÜTTE [1956] (see also TAKEUTI [1987], Ch. 1, § 8). We show that every sequent Γ is either provable in the sequent calculus or has a counter-model.

Let E be an enumeration of all tuples of terms over $\mathcal{L}$. Call a free variable *available at stage* k iff it occurs in the tree constructed before stage k (if there is no such variable, pick any and call it available) and *new* otherwise. A p-tuple $\bar{t}$ of terms is *available for the reduction of a formula F at place i with eigenvariables $a_1, \ldots, a_q$ at stage k on a branch B* iff

(1) $\bar{t}$ contains only variables which are available at stage k or which are among $a_1, \ldots, a_q$, and either

(2a) $\bar{t}$ has not been used at all for a reduction of F at place i on B in a stage before k, or

(2b) the pre-instance of the premise lying on B of a reduction of F at place i in a stage before k where $\bar{t}$ has been used did not contain any term variables.

A *reduction tree* T_Γ is an upward, rooted tree of sequents constructed from Γ in stages as follows:

Stage 0: Write Γ at the root of the tree.

Stage k: If the topmost sequent Γ' of a branch contains an atomic formula A s.t. $A \in \bigcap_{j \in I} \Gamma'_{\mathbf{j}}$ then stop the reduction for this branch. Call a branch *open* if it does not have this property.

Repeat the following reduction steps for every formula F occurring at place i in the topmost sequent Γ' of an open branch, which has neither already been reduced at place i on this branch in this stage, nor is the result of a reduction at this stage:

(1) $F \equiv \Box(A'_1, \ldots, A'_n)$: Replace Γ' in the reduction tree by:

$$\frac{\langle \Gamma', \Delta''_{\Box:i}(j) \rangle_{j \in I}}{\Gamma'}$$

where $\Delta''_{\Box:i}(j)$ is an instance of $\Delta'_{\Box:i}(j)$ in the rule $\Box{:}i$ introducing F as in Definition 3.3.1, obtained by instantiating $A_1, \ldots, A_n$ with $A'_1, \ldots, A'_n$, respectively.

(2) $F \equiv (\mathsf{Q}x)A'(x)$: Let $a_1, \ldots, a_p$ be all eigenvariables and $t_1, \ldots, t_q$ be all term variables in the premises of the rule schema $\mathsf{Q}{:}i$. Replace Γ' in the reduction tree by:

$$\frac{\left\langle \Gamma', \Delta''_{\mathsf{Q}:i}(j) \right\rangle_{j \in I}}{\Gamma'}$$

where $\Delta''_{\mathsf{Q}:i}(j)$ is an instance of $\Delta'_{\mathsf{Q}:i}(j)$ in $\mathsf{Q}{:}i$ introducing F as in Definition 3.3.3, obtained by instantiating A with A', the eigenvariables $\alpha_1, \ldots, \alpha_q$ with the first q-tuple $a_1, \ldots, a_q$ of new free variables in the enumeration E, and the term variables $\tau_1, \ldots, \tau_q$ with the first (w.r.t. E) p-tuple $t_1, \ldots, t_p$ of terms which is available for the reduction of F at place i with eigenvariables $a_1, \ldots, a_q$ at stage k on B. Observe that $F \in \Gamma'_{\mathbf{i}}$ and thus occurs in all upper sequents.

If T_Γ is finite, then every topmost sequent contains an atomic formula that occurs at each place in that sequent. A cut-free proof of Γ from axioms containing these formulas is easily constructed by inserting weakenings and exchanges.

If T_Γ is infinite it has an infinite branch B by König's Lemma. For every atomic formula $P(t_1, \ldots, t_n)$ in B, there is an l, $1 \le l \le m$, s.t. $P(t_1, \ldots, t_n)$ never occurs at position l in any sequent on B. We construct an interpretation $\mathbf{I}$ as follows: the domain is the set of terms, $\Phi_{\mathbf{I}}(t) = t$ (t a term), and $\Phi_{\mathbf{I}}(P(t_1, \ldots, t_n)) = v_l$, where v_l is the truth value corresponding to the place l.

If F is a formula occurring in B, and F occurs at place i anywhere in B, then $\mathrm{val}_{\mathbf{I}}(F) \neq v_i$. This is seen by induction on the complexity of F:

(1) F is atomic: $\mathrm{val}_{\mathbf{I}}(F) \neq v_i$ by the construction of $\mathbf{I}$.

(2) $F \equiv \Box(A_1, \ldots, A_n)$: F is reduced somewhere in B. Let $\Delta''_{\Box:i}(j)$ be the main formulas in the corresponding premises. By induction hypothesis none of the A_l in the premise that belongs to B evaluates to the truth value corresponding to its place in the premise. In other words, if $\Gamma', \Delta''_{\Box:i}(k)$ is the premise on B, the induction hypothesis says that $\mathbf{I} \not\models^{\mathrm{p}} \Delta''_{\Box:i}(k)$, and hence $\mathbf{I} \not\models \bigwedge_j \mathrm{psfe}(\Delta''_{\Box:i}(j))$. By Definitions 3.3.1 and 1.5.1, $\mathbf{I}(F) \neq v_i$.

(3) $F \equiv (\mathsf{Q}x)A(x)$: F is reduced somewhere in B. Let $\Delta'''_{\mathsf{Q}:i}(j)$ be the pre-instances of the main formulas in the premises, corresponding to $\Delta''_{\mathsf{Q}:i}(j)$. In particular, let $\Delta'''_{\mathsf{Q}:i}(k)$ be the one on B, and let $\alpha_1, \ldots, \alpha_r$ be the eigenvariables and $\tau_1, \ldots, \tau_s$ be the term variables occurring in it.

Now consider the substitution instances of $A(x)$ occurring on B above the lowermost reduction of F: F is reduced in every stage above this one in B. We want to show that for all terms (i.e., domain elements) $t_1, \ldots, t_p$ there are terms $t'_1, \ldots, t'_q$ s.t. $\mathbf{I} \not\models^{\mathrm{p}} \Delta'''_{\mathsf{Q}:i}(l)[t_1/\tau_1, \ldots, t_p/\tau_p, t'_1/\alpha_1, \ldots, t'_q/\alpha_q]$. Obviously, we need only consider the term- and eigenvariables actually occurring in $\Delta'''_{\mathsf{Q}:i}(l)$. If $\Delta'''_{\mathsf{Q}:i}(k)$ contains an eigenvariable α, then in $\Delta''_{\mathsf{Q}:i}(k)$ this α is replaced by a free variable a, and by induction hypothesis, $A(a)$ does not take the truth value at which it stands (in $\Delta'''_{\mathsf{Q}:i}(l)$). If $\Delta'''_{\mathsf{Q}:i}(l)$ contains a term variable τ, we have two cases:

(a) There are infinitely many reductions of F on B s.t. the pre-instance of the corresponding premise on B contains a term variable. Hence, B contains infinitely many occurrences of instances of $A[t/\tau]$ where t is any term, and in fact *all* such instances (This follows from the way available tuples of terms are chosen). None of these instances take the truth values corresponding to the place at which they stand (in $\Delta'''_{\mathsf{Q}:i}(j)$). Together with what has been said above for the eigenvariables, it follows that $\mathbf{I} \not\models^{\mathrm{p}} \Delta'''_{\mathsf{Q}:i}(l)[t_1/\tau_1, \ldots, t_p/\tau_p, t'_1/\alpha_1, \ldots, t'_q/\alpha_q]$ and hence that $\mathrm{val}_{\mathbf{I}}(F) \neq v_i$ by Definition 1.5.5.

(b) The reductions of F on B yield only a finite number (including zero) of instances $A[t/\tau_j]$ on B because there are only finitely many reductions of F on B at place i whose pre-instances of premises on B contain term variables. Since F is reduced infinitely often on B, there is a first reduction where the pre-instance of the corresponding premise on B contains no term variables and such that all reductions above it share this property. Let $\tilde{\Delta}'''_{\mathsf{Q}:i}(l')$ be the corresponding pre-instance of the premise on B. $\tilde{\Delta}'''_{\mathsf{Q}:i}(l')$ contains only eigenvariables, and we have that $\mathbf{I} \not\models^{\mathrm{p}} \tilde{\Delta}''_{\mathsf{Q}:i}(l')$ for the corresponding instance (by induction hypothesis) and hence that $\mathrm{val}_{\mathbf{I}}(F) \neq v_i$ by Definition 1.5.5.

In particular, no formula in Γ evaluates to the truth value corresponding to the position at which it stands. Hence $\mathbf{I}$ does not p-satisfy Γ. ∎

3.3.11. Corollary *Let A be any formula. $[V{:}\,A]$ is provable in* **PL** *without cuts from atomic axioms.*

Proof. Since $[V\colon A]$ is valid, the construction of $T_{[V:A]}$ terminates and gives a cut-free proof with atomic axioms. ∎

3.3.12. THEOREM (Completeness for **NL**) *If a sequent is n-valid, then it is n-provable without cuts from atomic axioms.*

Proof. The construction of the reduction tree for a sequent Γ is as in the positive case, only that a branch with topmost sequent Γ' is closed, if there is a formula A s.t. $A \in \Gamma'_{\mathbf{i}}$ and $A \in \Gamma'_{\mathbf{j}}$ for $i \neq j$.

If T_Γ is infinite it has, again, an infinite branch B by König's Lemma. For every atomic formula $P(t_1, \ldots, t_n)$ there is at most one position i where it ever appears in the sequents of B. We construct an interpretation $\mathbf{I}$ as follows: the domain is the set of terms, $\Phi_{\mathbf{I}}(t) = t$ (t a term) and $\Phi_{\mathbf{I}}(P(t_1, \ldots, t_n)) = v_i$, where v_i is the truth value corresponding to the position i.

If F is a formula occurring in B, and F occurs at place i anywhere in B, then $\mathrm{val}_{\mathbf{I}}(F) = v_i$. This is proved as in the positive case, *mutatis mutandis*. In particular, every formula in Γ evaluates to the truth value corresponding to the position at which it stands. Hence $\mathbf{I}$ does not n-satisfy Γ. ∎

3.4 Equivalent Formulations of Sequent Calculi

The way we defined sequent calculus is the one closest to Gentzen's original definition. There are several minor modifications to the definitions that can be made, concerning the structural rules and the sequents as *sequences* of formulas, according to taste and convenience. First of all, it is possible to divide the side formulas of introductions similar to the way the cuts are defined, as follows:

3.4.1. DEFINITION Let f be $\square$ or Q, let F be $\square(A_1, \ldots, A_n)$ or $(\mathsf{Q}x)A(x)$, respectively, and let $\Delta'_{f:i}(j)$ be as in Definition 3.3.1 or 3.3.3, respectively. Then the following is called a *combinational rule:*

$$\frac{\left\langle \Gamma_j, \Delta'_{f:i}(j) \right\rangle_{j \in I}}{\Gamma, [i\colon F]}\ f{:}i$$

where $\Gamma = \langle \Gamma_j \mid j \in I \rangle$. Furthermore, we allow an inference to be made, even if only subsequents of $\Delta'_{f:i}(j)$ are present in the premises.

The resulting calculus is called the *combinational sequent calculus.*

3.4.2. PROPOSITION *If Γ is provable, then a subsequent of Γ is provable without weakenings.*

Proof. We inductively translate a proof P of Γ to a proof $\eta(P)$ of a subsequent of Γ':

If Γ is an axiom, we take $\eta(P) = P$. Otherwise, distinguish cases according to the last rule J in P:

(1) J is an exchange: Γ is of the form $\Lambda, [i\colon B, A], \Pi$. By induction hypothesis we have a proof $\eta(P')$ of Λ', Δ, Π', where Λ' and Π' are subsequents of Λ and Π, respectively, and Δ is either empty, $[i\colon A]$, $[i\colon B]$, or $[i\colon A, B]$. In the former cases, we are done ($\eta(P) = \eta(P')$). In the latter case we get $\eta(P)$ as

$$\dfrac{\begin{array}{c}\vdots\ \eta(P') \\ \Lambda', [i\colon A, B], \Pi'\end{array}}{\Lambda', [i\colon B, A], \Pi'}\ \text{x:}i$$

(2) J is a contraction: Γ is of the form $\Lambda, [i\colon A, A]$. By induction hypothesis we have a proof $\eta(P')$ of Λ', Δ, where Λ' is a subsequent of Λ and Δ is either empty, $[i\colon A]$ or $[i\colon A, A]$. In the former cases, we are done ($\eta(P) = \eta(P')$). In the latter case we get $\eta(P)$ as

$$\dfrac{\begin{array}{c}\vdots\ \eta(P') \\ \Lambda', [i\colon A, A]\end{array}}{\Lambda', [i\colon A]}\ \text{c:}i$$

(3) J is a weakening: Γ is of the form $\Lambda, [i\colon A]$. By induction hypothesis we have a proof $\eta(P')$ of Λ', where Λ' is a subsequent of Λ. We take $\eta(P) = \eta(P')$, which contains no weakening.

(4) J is a cut: Γ is of the form $\Lambda_1, \ldots, \Lambda_k$. By induction hypothesis we have proofs $\eta(P_j)$ of Λ'_j, Δ'_j, where Λ'_j is a subsequent of Λ_j and Δ_j is either empty or $[i_j\colon A]$ (In the positive case, $k = 2$, in the negative case $k = m$ and $i_j = j$). If one of the Δ_j is empty, we take $\eta(P) = \eta(P_j)$. Otherwise, we get $\eta(P)$ as

$$\dfrac{\begin{array}{c}\vdots\ \eta(P_1) \\ \Lambda'_1, [i_1\colon A]\end{array} \quad \ldots \quad \begin{array}{c}\vdots\ \eta(P_k) \\ \Lambda'_k, [i_k\colon A]\end{array}}{\Lambda'_1, \ldots, \Lambda'_j}\ \text{cut}$$

(5) J is an introduction of f at place i: Π is of the form $\Gamma, [i\colon F]$ (F is $\Box(A_1, \ldots, A_n)$ or $(\forall x)A(x)$, according to whether f is $\Box$ or Q). By induction hypothesis we have proofs $\eta(P_j)$ of $\Gamma'_j, \Delta''_{f:i}(j)$, where Γ'_j and $\Delta''_{f:i}(j)$ are subsequents of Γ_j and $\Delta'_{f:i}(j)$. If one of the $\Delta''_{f:i}(j)$ is empty, we take $\eta(P) = \eta(P_j)$. Otherwise, we get $\eta(P)$ as

$$\dfrac{\left\langle \begin{array}{c}\vdots\ \eta(P_j) \\ \Gamma'_j, \Delta''_{f:i}(j)\end{array} \right\rangle_{j \in I}}{\Gamma', F}\ f\text{:}i \qquad \blacksquare$$

Note that by allowing inferences on premises with subsets of the auxiliary formulas present, we can in effect simulate the weakening rule. In the case of Gentzen's **LK**, this allowance need not be made to obtain the previous result. This is because the rules of **LK** have the property that every premise contains only one occurrence of an auxiliary formula. By inspection of the proof of the preceding proposition, we see that this is also sufficient (see also KLEENE [1952], § 80). In the many-valued sequent calculus, it is in fact possible to give an equivalent formulation with this property, but only for the propositional case. Consider an introduction rule for $\square$ at place i, where $\Delta'_{\square:i}(j)$ contains more than one formula, i.e., $\Delta'_{\square:i}(k) = \Delta, [l\colon A_p]$. Then we can split the rule into two as follows:

$$\frac{\left\langle \Gamma_j, \Delta'_{f:i}(j) \right\rangle_{j\in I\setminus k} \quad \Gamma_k, \Delta}{\Gamma, [i\colon \square(A_1,\dots,A_n)]}\ \square{:}i' \qquad \frac{\left\langle \Gamma_j, \Delta'_{f:i}(j) \right\rangle_{j\in I\setminus k} \quad \Gamma_k, [l\colon A_p]}{\Gamma, [i\colon \square(A_1,\dots,A_n)]}\ \square{:}i''$$

It is easily seen by induction that we can obtain for any propositional rule a set of rules with only one auxiliary formula per premise. The resulting system is called the *split-rule calculus*. This system is equivalent to the combinational sequent calculus: Every instance of a split rule can be made into an instance of the original rule by means of a series of weakenings. On the other hand, assume we have a deduction in the combinational calculus,

$$\frac{\left\langle \begin{array}{c} \vdots\ P_j \\ \Gamma_j, \Delta'_{f:i}(j) \end{array} \right\rangle_{j\in I}}{\Gamma, [i\colon \square(A_1,\dots,A_n)]}\ \square{:}i$$

Then we can give a deduction in the split-rule calculus. It suffices to give the case of a rule split into two, the general case is by induction:

$$\dfrac{\dfrac{\left\langle \begin{array}{c} \vdots\ P_j \\ \Gamma_j, \Delta'_{f:i}(j) \end{array} \right\rangle_{j\in I\setminus k} \qquad \dfrac{\dfrac{\dfrac{\left\langle \begin{array}{c} \vdots\ P_j \\ \Gamma_j, \Delta'_{f:i}(j) \end{array} \right\rangle_{j\in I\setminus k} \quad \dfrac{\begin{array}{c} \vdots\ P_k \\ \Gamma_k, \Delta, [l\colon A_p] \end{array}}{\Gamma_k, [l\colon A_p], \Delta}\ \mathrm{x}}{\Gamma, [l\colon A_p], [i\colon \square(A_1,\dots,A_n)]}\ \square{:}i'}{\Gamma, [i\colon \square(A_1,\dots,A_n)], [l\colon A_p]}\ \mathrm{x}}{}}{\Gamma', [i\colon \square(A_1,\dots,A_n), \square(A_1,\dots,A_n)]}\ \square{:}i''}{\Gamma, [i\colon \square(A_1,\dots,A_n)]}\ \mathrm{xc}$$

Nota bene that the method of splitting rules does in general not work for quantifier rules. Consider for instance the sequent $[i, j\colon A(\alpha)]$, which might be a premise for a quantifier introduction. It expresses the condition that, for all α, $A(\alpha)$ either takes the value v_i or the value v_j. The corresponding split sequent, however, says that either for all α, $A(\alpha)$ takes the value v_i or, for all α, $A(\alpha)$ takes the value v_j. In a word, $\mathbb{V}$ does not distribute over $\mathbb{W}$.

We could also define a sequent to be a tuple of *sets* (instead of sequences) of formulas. We call such sequents *set-sequents*. If sequent calculus formulated with set-sequents, the contraction and exchange rules become superfluous.

It should be pointed out that the study of sequent systems where combinational and ordinary rules are combined while some or all structural rules are left out is a difficult and highly interesting topic. Considerations along these lines lead to the investigation of so-called *substructural logics*, for instance Girard's Linear Logic, or Relevance Logic. For a short survey see TROELSTRA [1992], § 2.8.

3.5 The Cut-elimination Theorem for PL

The cut-elimination theorem, or *Hauptsatz*, was first proved for classical logic in GENTZEN [1934]. It is one of the most important theorems in proof theory. Its underlying idea is to *eliminate* structure from a given proof to extract information from it. This is an idea often used, and cut-elimination corresponds closely to methods used in other calculi to obtain similar results: the 2nd ε-Theorem in HILBERT and BERNAYS [1939] in ε-calculus, the normalization theorem of PRAWITZ [1971] in natural deduction, and the normalization results for free deduction obtained by PARIGOT [1992]. Important consequences of the *Hauptsatz* for classical (and intuitionistic) logic include the Midsequent Theorem, Herbrand's Theorem, Craig's Interpolation Theorem, and Beth's Definability Lemma (see TAKEUTI [1987], ch. 1, § 6).

The proof of the cut-elimination theorem for the family **PL** of many-valued sequent calculi is analogous to the proof of the classical case given by Gentzen. It proceeds by reducing cuts to cuts on formulas of smaller complexity and moving the cuts in a given proof upwards until they only involve axioms; such cuts are easily removed. The most important prerequisite for the proof is the ability to transform a cut on a composite formula to a derivation using only cuts acting on subformulas of the original cut-formula. In the classical case, this is established by case distinction; we, however, have to establish this uniformly by the lemmas below. Note how the way TAKAHASHI [1967a] defined his sequent calculus makes similar lemmas unnecessary for his proof of cut-elimination.

In the proof of the cut-elimination theorem, we replace the cut rule, for technical reasons, by an equivalent rule called *mix:*

$$\frac{\Pi \quad \Lambda}{\Pi^{(i)}, \Lambda^{(j)}}\ (A, i, j)$$

where A occurs in $\Pi_{\mathbf{i}}$ and $\Lambda_{\mathbf{j}}$, and $\Pi^{(i)}$ ($\Lambda^{(j)}$) is obtained from Π (Λ) by deleting every occurrence of A in $\Pi_{\mathbf{i}}$ ($\Lambda_{\mathbf{j}}$). A cut can be replaced by a mix followed by a sequence of weakenings, and a mix can be emulated by a series of cuts. Call the calculus obtained from **PL** by replacing the cut rule with the mix rule **PL**$'$.

3.5.1. LEMMA *Consider the derivation schema*

$$\frac{\dfrac{\langle \Pi, \Delta'_{\Box:i}(r)\rangle_r}{\Pi, [i\colon \Box(A_1, \ldots, A_n)]} \quad \dfrac{\langle \Lambda, \Delta'_{\Box:j}(s)\rangle_s}{\Lambda, [j\colon \Box(A_1, \ldots, A_n)]}}{\Pi, \Lambda}\ (\Box(A_1, \ldots, A_n), i, j)$$

where neither $\Pi_{\mathbf{i}}$ nor $\Lambda_{\mathbf{j}}$ contains $\Box(A_1, \ldots, A_n)$. Then there is a proof of Π, Λ from the sequents $\Pi, \Delta'_{\Box:i}(r)$ and $\Lambda, \Delta'_{\Box:j}(s)$ using only mixes on $A_1, \ldots, A_n$.

Proof. It suffices to show the lemma for $\Pi = \Gamma = \emptyset$: If we have a proof of the empty sequent from the sequents $\Delta_{\Box:i}(r)$ and $\Delta_{\Box:j}(s)$, then we can immediately obtain (using exchanges and contractions) a proof of Π, Λ by writing Π or Λ left of every sequent in the deduction, as appropriate.

The conjunction $\bigwedge_r \text{psfe}(\Delta'_{\Box:i}(r)) \wedge \bigwedge_s \text{psfe}(\Delta'_{\Box:j}(s))$ of the sfes corresponding to the sequents $\Delta'_{\Box:i}(r)$ and $\Delta'_{\Box:j}(s)$ is clearly unsatisfiable, since $\bigwedge_r \text{psfe}(\Delta'_{\Box:i}(r))$ is equivalent to $\Box(\cdot)^{v_i}$ and $\bigwedge_s \text{psfe}(\Delta'_{\Box:j}(s))$ is equivalent to $\Box(\cdot)^{v_j}$, where $i \neq j$. We translate this conjunction to a set of clauses, which is also unsatisfiable: Let $\mathcal{C} = \bigcup_{j \in I} C_{\Box:i}(j)$ be the set of clauses where

$$C_{\Box:i}(j) = \{P_A{}^{v_k} \mid A \in \Delta'_{\Box:i}(j)_{\mathbf{k}}, 1 \leq k \leq m\}$$

$\mathcal{C}$ is unsatisfiable, since the conjunction of the Δ's is. By the completeness of many-valued resolution (Theorem 2.5.8), there is a resolution deduction R of $\Box$ from $\mathcal{C}$. Note that the literals in $\mathcal{C}$ are all ground, and hence, in every resolution step, there is only one literal that is resolved upon. Without loss of generality, assume that the resolution proof is in tree form (cf. Proposition 2.5.9). Such a proof can immediately be translated to a $\mathbf{PL}'$ deduction as follows: Let

$$\frac{C_1 \cup \{A\lambda^{v_k}\} \quad C_2 \cup \{A\lambda^{v_l}\}}{C_1 \cup C_2}$$

be the last resolution step in R, where A is a of one of $A_1, \ldots, A_n$. The corresponding mix is:

$$\frac{\Delta_1, [k{:}\, A, \ldots, A] \quad \Delta_2, [l{:}\, A, \ldots, A]}{\Delta_1, \Delta_2}\ (A, k, l) \qquad \blacksquare$$

where Δ_1 and Δ_2 are the sequents corresponding to C_1 and C_2, respectively.

3.5.2. LEMMA *Consider the derivation schema*

$$\frac{\dfrac{\langle \Pi, \Delta_{\Box:i}(r)\rangle_r}{\Pi, i{:}\,(\mathsf{Q}x)A(x)} \quad \dfrac{\langle \Lambda, \Delta_{\mathsf{Q}:j}(s)\rangle_s}{\Lambda, j{:}\,(\mathsf{Q}x)A(x)}}{\Pi, \Lambda}\ ((\mathsf{Q}x)A(x), i, j)$$

where neither $\Pi_{\mathbf{i}}$ nor $\Lambda_{\mathbf{j}}$ contains $(\mathsf{Q}x)A(x)$, and the appropriate eigenvariable conditions are satisfied. Then there is a proof of Π, Λ from the sequents $\Pi, \Delta_{\mathsf{Q}:i}(l)$ and $\Lambda, \Delta_{\mathsf{Q}:j}(k)$ using only mixes on instances $A\sigma$ of formulas A occurring in the Δ's, where $\text{dom}(\sigma)$ contains only eigenvariables.

Proof. Again, it suffices to show the lemma for $\Pi = \Gamma = \emptyset$. The conjunction $\bigwedge_r \text{psfe}(\Delta'_{\mathsf{Q}:i}(r)) \wedge \bigwedge_s \text{psfe}(\Delta'_{\mathsf{Q}:j}(s))$ of the sfes corresponding to the sequents $\Delta_{\mathsf{Q}:i}(r)$ and $\Delta_{\mathsf{Q}:j}(s)$ is universally unsatisfiable (where universal satisfiability for sfes is defined analogous to Definition 2.2.2. We translate this conjunction to a set of clauses, which is also universally unsatisfiable: Let $\mathcal{C} = \bigcup_{j \in I} \mathcal{C}_{\mathsf{Q}:i}(j)$ be the set of clauses where

$$\begin{aligned} \mathcal{C}_{\mathsf{Q}:i}(j) \quad = \quad & \{P_A(c_\tau)^{v_k} \mid A(\tau) \in \Delta'_{\mathsf{Q}:i}(j)_{\mathbf{k}}, 1 \leq k \leq m\} \cup \\ & \cup \{P_A(a_\tau)^{v_k} \mid A(\alpha) \in \Delta'_{\mathsf{Q}:i}(j)_{\mathbf{k}}, 1 \leq k \leq m\} \end{aligned}$$

where the a_α are distinct free variables and the c_τ are distinct constant symbols. By the completeness of many-valued resolution (Theorem 2.5.8), there is a resolution deduction R of $\Box$ from $\mathcal{C}$, which is (w.l.o.g.) in tree form. We recursively translate $R\rho(R)$ (ρ the cumulative substitution of Definition 2.5.10) into a $\mathbf{PL}'$-derivation as follows: Let

$$\dfrac{\begin{array}{cc} \vdots\, R_1 & \vdots\, R_2 \\ C_1 & C_2 \end{array}}{(C_1 \setminus \{A\})\} \cup (C_2 \setminus \{A\}\}}$$

be the last resolution step in $R\rho(R)$, where $\langle \{A\}, \{A\}, \mathrm{at}(\{A\})\rangle$ is a key triple (with mgu id), A contains an atom of the form $P_A(a)$ or $P_A(c)$ (a is a variable and c a constant symbol). Recall that the cumulative substitution has been applied to R, so resolution steps actually do take this special, ground-like, form.

Let P_1 (P_2) be the result of the recursive translation of R_1 (R_2). The corresponding deduction in $\mathbf{PL}'$ is:

$$\dfrac{\begin{array}{cc} \vdots\, P_1 & \vdots\, P_2 \\ \Delta_1^{(k)}, [k\colon A] & \Delta_2^{(l)}, [l\colon A] \end{array}}{\Delta_1, \Delta_2}\ (A, r, s)$$

where Δ_1 and Δ_2 are the sequents corresponding to C_1 and C_2, respectively (where free variables are replaced by eigenvariables and constants by term variables). The corresponding σ is easily obtained from $\rho(R)$.

Note that here, too, we can add Π and Λ to the sequents without changing the term structure of formulas other than the ones in the Δ's, since only eigenvariables (which must not occur in Π or Λ) are substituted into. ∎

It is clear that σ will in general substitute into eigenvariables. This corresponds to the case in the cut-elimination theorem for **LK** where a cut on $(\forall x)A(x)$ is reduced to a cut on $A(t)$. The eigenvariable above the strong premise of the cut is replaced throughout the whole proof with the term occurring in the premise of the weak $\forall$-introduction above the right premise of the cut.

We are now ready to state and prove the *Hauptsatz* for **PL**:

3.5.3. THEOREM *A* **PL**-*proof with end sequent Λ can be transformed into a cut-free* **PL**-*proof of Λ.*

Proof. It suffices to show that *mixes* can be eliminated from $\mathbf{PL}'$-proofs. If every $\mathbf{PL}'$-proof with only one mix as its last inference can be transformed to a $\mathbf{PL}'$-proof without a mix, then the mixes can be eliminated from *all* $\mathbf{PL}'$-proofs. This is seen by induction on the number c of mixes in the proof: For $c = 1$ this is the hypothesis. If $c > 1$, then consider a mix s.t. the subproof ending in the mix contains only this mix as its last inference. By hypothesis, this mix can be eliminated, and the subproof replaced by a mix-free proof of the same endsequent. The resulting proof has $c - 1$ mixes, and the induction hypothesis applies. Hence, it suffices to show the following:

3.5.4. LEMMA *Let P be a* **PL**$'$*-proof containing only one mix (A, i, j) as its last inference. Then P can be transformed to a mix-free* **PL**$'$*-proof P' of the same end-sequent.*

Let P be a **PL**$'$-proof containing only one mix (A, i, j) which occurs as the last inference in P. The *degree* of P, denoted $d(P)$, is the complexity $\deg(A)$ of the mix formula A.

We call a thread in P containing the left (right) upper sequent of the mix a *left (right) thread*. The *rank* of a left (right) thread is the number of consecutive sequents counting upwards from the left (right) upper sequent of the mix which contain the mix formula at place i (j). The *left (right) rank of* P, denoted $r_l(P)$ ($r_r(P)$) is the maximum of the ranks of its left (right) threads. The *rank* of P, denoted $r(P)$, is the sum of its left and right rank: $r(P) = r_l(P) + r_r(P)$.

The proof is by double induction on the rank and degree of P:

(1) $r = 2$, i.e., left and right rank of P equal 1. We distinguish cases according to the type of the inferences above the mix:

(a) Π is an axiom $[V\colon A]$: P is of the form

$$\frac{[V\colon A] \qquad \begin{matrix}\vdots\, P_1\\ \Lambda\end{matrix}}{[V\setminus v_i\colon A], \Lambda^{(j)}}\,(A, i, j)$$

We can derive $\Pi^{(i)}, \Lambda^{(j)}$ without a mix as follows:

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\vdots\, P_1\\ \Lambda\end{matrix}}{[j\colon A, \ldots, A], \Lambda^{(j)}}\,\text{x}}{[j\colon A], \Lambda(j)}\,\text{c}}{[V\setminus v_i\colon A], \Lambda(j)}\,\text{w}$$

(b) Λ is an axiom. Similarly.

(c) Π is the conclusion of a structural inference. Since the left rank is 1, this inference must be a weakening at place i:

$$\frac{\dfrac{\begin{matrix}\vdots\, P_1\\ \Pi^{(i)}\end{matrix}}{\Pi^{(i)}, [i\colon A]}\,\text{w: } i \qquad \begin{matrix}\vdots\, P_2\\ \Lambda\end{matrix}}{\Pi^{(i)}, \Lambda^{(j)}}\,(A, i, j)$$

where Π is $\Pi', [i\colon A]$. We obtain $\Pi^{(i)}, \Lambda^{(j)}$ without a mix as follows:

$$\dfrac{\begin{matrix}\vdots\, P_1\\ \Pi^{(i)}\end{matrix}}{\Pi^{(i)}, \Lambda^{(j)}}\,\text{w}$$

(d) Λ is the conclusion of a structural inference. Similarly.

(e) Both Π and Λ are conclusions of introduction rules $\Box{:}i$ and $\Box{:}j$ for the connective $\Box$:

$$\dfrac{\dfrac{\langle \Pi, \Delta_{\Box:i}(r)\rangle_r}{\Pi, i{:}\, \Box(\cdots)} \quad \dfrac{\langle \Lambda, \Delta_{\Box:j}(s)\rangle_s}{\Lambda, j{:}\, \Box(\cdots)}}{\Pi, \Lambda}\ (\Box(\cdots), i, j)$$

By Lemma 3.5.1, there is a derivation of Π, Λ from $\Pi, \Delta_{\Box:i}(r)$ and $\Lambda, \Delta_{\Box:j}(s)$, using only mixes, exchanges, and contractions. Since the formulas in the Δ's are subformulas of $\Box(\cdots)$, the degrees of all the resulting mixes are less than the degree of the original mix. We iteratively apply the induction hypothesis to the new mixes and obtain a derivation without mixes of Π, Λ.

(f) Both Π and Λ are conclusions of introduction rules $\mathsf{Q}{:}i$ and $\mathsf{Q}{:}j$ for the quantifier Q:

$$\dfrac{\dfrac{\langle \Pi, \Delta_{\mathsf{Q}:i}(r)\rangle_r}{\Pi, i{:}\, (\mathsf{Q}x)A(x)} \quad \dfrac{\langle \Lambda, \Delta_{\mathsf{Q}:j}(s)\rangle_s}{\Lambda, j{:}\, (\mathsf{Q}x)A(x)}}{\Pi, \Lambda}\ ((\mathsf{Q}x)A(x), i, j)$$

By Lemma 3.5.2, there is a derivation of Π, Λ from $\Pi, \Delta_{\mathsf{Q}:i}(r)\sigma$ and $\Lambda, \Delta_{\mathsf{Q}:j}(s)\sigma$, using only mixes, exchanges, and contractions, where σ only substitutes into eigenvariables of the two quantifier introductions. Since the formulas in the Δ's are subformulas of $(\mathsf{Q}x)A(x)$, the degrees of all the resulting mixes are less than the degree of the original mix. We iteratively apply the induction hypothesis to the new mixes and obtain a derivation without mixes of Π, Λ.

(2) $r_r(P) > 1$: Again, we distinguish cases:

(a) $\Lambda_{\mathbf{i}}$ contains A: We obtain the following mix-free proof:

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\ P_1\\ \Pi\end{array}}{\Pi^{(i)}, [i{:}\, A, \ldots, A]}\ \text{e}}{\Pi^{(i)}, [i{:}\, A]}\ \text{c}}{\Pi^{(i)}, \Lambda^{(j)}}\ \text{w}$$

(b) $\Pi_{\mathbf{j}}$ contains A: We obtain the following mix-free proof:

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\ P_2\\ \Lambda\end{array}}{\Lambda^{(j)}, [j{:}\, A, \ldots, A]}\ \text{e}}{\Lambda^{(j)}, [j{:}\, A]}\ \text{c}}{\Pi^{(i)}, \Lambda^{(j)}}\ \text{w}$$

(c) Λ is the consequence of an inference J_2, which is either structural (but not cut), or a logical inference not introducing A at place j. P is of the form

$$\dfrac{\begin{matrix}\vdots\, P_1 \\ \Pi\end{matrix} \qquad \dfrac{\begin{matrix}\vdots\, P(1) \\ \Psi(1)\end{matrix} \quad \ldots \quad \begin{matrix}\vdots\, P(p) \\ \Psi(p)\end{matrix}}{\Lambda}\,J_2}{\Pi^{(i)}, \Lambda^{(j)}}\,(A, i, j)$$

Let $j_1, \ldots, j_s$, $1 \le j_k \le p$, be all indices s.t. $\Psi(j_k)$ contains A (There is at least one such j_k, otherwise the right rank of P would equal 1). Consider the proofs $P(j_k)'$:

$$\dfrac{\begin{matrix}\vdots \\ \Pi\end{matrix} \quad \begin{matrix}\vdots \\ \Psi(j_k)\end{matrix}}{\Pi^{(i)}, \Psi(j_k)^{(j)}}\,(A, i, j)$$

In $P(j_k)'$, $r_l(P(j_k)') = r_l(P)$ and $r_r(P(j_k)') \le r_r(P) - 1$, and in sum $r(P(j_k)') \le r(P) - 1$. Hence the induction hypothesis applies and we have mix-free proofs $P(j_k)''$ of $\Pi^{(i)}, \Psi(j_k)^{(j)}$. For indices l not occurring in the above list, we have that $\Psi(l)$ equals $\Psi(l)^{(j)}$, and we define $P(l)''$ as

$$\dfrac{\begin{matrix}\vdots\, P(l) \\ \Psi(l)\end{matrix}}{\Pi^{(i)}, \Psi(l)}\,\mathrm{w}$$

If J_2 is a (w:j) (and consequently, $p = 1$ and $\Psi(1)^{(j)} = \Lambda^{(j)}$), then $P(1)''$ serves as our transformed proof. Otherwise, construct a proof as follows:

$$\dfrac{\dfrac{\begin{matrix}\vdots\, P(1)'' \\ \Pi^{(i)}, \Psi(1)^{(j)}\end{matrix} \quad \ldots \quad \begin{matrix}\vdots\, P(p)'' \\ \Pi^{(i)}, \Psi(p)^{(j)}\end{matrix}}{\Pi^{(i)}, \Lambda^{(j)}}\,J_2}{\Pi^{(i)}, \Lambda^{(j)}}\,\mathrm{x}$$

(d) Λ is the consequence of a logical inference J_2 introducing A at place j. P is of the form

$$\dfrac{\begin{matrix}\vdots\, P_1 \\ \Pi\end{matrix} \qquad \dfrac{\begin{matrix}\vdots\, P(1) \\ \Lambda, \Delta(1)\end{matrix} \quad \ldots \quad \begin{matrix}\vdots\, P(p) \\ \Lambda, \Delta(p)\end{matrix}}{\Lambda, [j\colon A]}\,J_2}{\Pi^{(i)}, \Lambda^{(j)}}\,(A, i, j)$$

Consider the proofs $P(k)'$ (Note that $\Delta(k)$ does not contain A—only proper subformulas of A—and hence $\Delta(k)^{(j)}$ equals $\Delta(k)$):

$$\dfrac{\begin{matrix}\vdots\, P_1 \\ \Pi\end{matrix} \quad \begin{matrix}\vdots\, P(k) \\ \Lambda, \Delta(k)\end{matrix}}{\Pi^{(i)}, \Lambda^{(j)}, \Delta(k)}\,(A, i, j)$$

In $P(k)$, $r_l(P(k)') = r_l(P)$, $r_r(P(k)') \leq r_r(P) - 1$ and in sum $r(P(k)') \leq r(P) - 1$. Hence, the induction hypothesis applies and we obtain mix-free proofs $P(k)''$ of $\Pi^{(i)}, \Lambda^{(j)}, \Delta(k)$. Construct a proof P' as follows:

$$\dfrac{\begin{array}{c}\vdots\, P_1\\ \Pi\end{array} \qquad \dfrac{\begin{array}{c}\vdots\, P(1)''\\ \Pi^{(i)}, \Lambda^{(j)}, \Delta(1)\end{array} \quad \ldots \quad \begin{array}{c}\vdots\, P(p)''\\ \Pi^{(i)}, \Lambda^{(j)}, \Delta(p)\end{array}}{\Pi^{(i)}, \Lambda^{(j)}, [j\colon A]}\, J_2}{\Pi^{(i)}, \Pi^{(i)}, \Lambda^{(j)}}\, (A, i, j)$$

Note that A does not occur at place j in $\Pi^{(i)}$, since otherwise case (2)(b) would have applied, hence $r_r(P') = 1$. With $r_l(P') = r_l(P)$ we have that $r(P') < r(P)$ and the induction hypothesis yields a mix-free proof P'' of $\Pi^{(i)}, \Pi^{(i)}, \Lambda^{(j)}$. We obtain a mix-free proof:

$$\dfrac{\begin{array}{c}\vdots\, P'\\ \Pi^{(i)}, \Pi^{(i)}, \Lambda^{(j)}\end{array}}{\Pi^{(i)}, \Lambda^{(j)}}\, \text{xc}$$

(3) $r_r(P) = 1$ and $r_l(P) > 1$: This case is dealt with in the same way as (2) above, *mutatis mutandis.*

This completes the proof of the cut-elimination theorem. ■

3.6 The Cut-elimination Theorem for NL

The cut elimination theorem also holds for the negative sequent calculus. Carnielli [1991] gives a proof of cut-elimination for his (n-)sequent calculus via tableaux.

The induction proof itself is again very similar to the proof for the positive case. As above, we first have to ensure that cuts on composite formulas are decomposable. The mix in the negative sequent calculus takes the form

$$\dfrac{\Pi_1 \quad \ldots \quad \Pi_m}{\Pi_1^{(1)}, \ldots, \Pi_m^{(m)}}\, (A)$$

where A occurs in $\Pi_{i\mathbf{i}}$ $(1 \leq i \leq m)$, and $\Pi_i^{(i)}$ is obtained from Π_i by deleting every occurrence of A in $\Pi_{i\mathbf{i}}$. Call the calculus obtained from **NL** by replacing the cut rule with the mix rule **NL**$'$.

3.6.1. Lemma *Consider the derivation schema*

$$\dfrac{\dfrac{\langle \Pi_1, \Delta_{\square:1}(r_1)\rangle_{r_1}}{\Pi_1, [1\colon \square(A_1, \ldots, A_n)]} \quad \ldots \quad \dfrac{\langle \Pi_m, \Delta_{\square:m}(r_m)\rangle_{r_m}}{\Pi_m, [m\colon \square(A_1, \ldots, A_n)]}}{\Pi_1, \ldots, \Pi_m}\, (\square(A_1, \ldots, A_n))$$

where $\Pi_{i\mathbf{i}}$ does not contain $\square(A_1, \ldots, A_n)$. Then there is a proof of $\Pi_1, \ldots, \Pi_m$ from the sequents $\Pi_i, \Delta_{\square:i}(r_i)$ using only mixes on $A_1, \ldots, A_n$.

Proof. Completely analogous to the positive case, using negative resolution deductions. ■

3.6.2. LEMMA *Consider the derivation schema*

$$\dfrac{\dfrac{\langle \Pi_1, \Delta_{\mathsf{Q}:1}(r_1)\rangle_{r_1}}{\Pi_1, [1\colon (\mathsf{Q}x)A(x)]} \quad \ldots \quad \dfrac{\langle \Pi_m, \Delta_{\mathsf{Q}:m}(r_m)\rangle_{r_m}}{\Pi_m, [m\colon (\mathsf{Q}x)A(x)]}}{\Pi_1, \ldots, \Pi_m}\ ((\mathsf{Q}x)A(x))$$

where $\Pi_{i\mathbf{i}}$ does not contain $(\mathsf{Q}x)A(x)$, and the appropriate eigenvariable conditions are satisfied. Then there is a proof of $\Pi_1, \ldots, \Pi_m$ from the sequents $\Pi_i, \Delta_{\mathsf{Q}:i}(r_i)$ using only mixes on instances $A\sigma$ of formulas A occurring in the Δ's, where $\mathrm{dom}(\sigma)$ contains only eigenvariables.

Proof. Completely analogous to the positive case, using negative resolution deductions. ■

3.6.3. THEOREM *A* **NL***-proof with end sequent Λ can be transformed into a cut-free* **NL***-proof of Λ.*

Proof. It suffices to show the following:

Let P be a $\mathbf{NL}'$-proof containing only one mix A as its last inference. Then P can be transformed to a mix-free $\mathbf{NL}'$-proof P' of the same end-sequent.

Let P be a $\mathbf{NL}'$-proof containing only one mix A which occurs as the last inference in P. The *degree* of P, denoted $d(P)$, is the complexity of the mix formula A.

We call a thread in P containing the i-th upper sequent of the mix an *i-th thread*. The *rank* of an i-th thread is the number of consecutive sequents counting upwards from the i-th upper sequent of the mix which contain the mix formula at place i. The *i-th rank of P*, denoted $r_i(P)$ is the maximum of the ranks of its i-th threads. The *rank* of P, denoted $r(P)$, is the sum of its i-th ranks rank: $r(P) = \sum_{i=1}^{m} r_i(P)$.

The proof is by double induction on the rank and degree of P:

(1) $r = m$, i.e., all i-th ranks of P equal 1. We distinguish cases according to the type of the inferences above the mix:

   (a) Π_i is an axiom $[i, j\colon A]$: P is of the form

$$\dfrac{\begin{matrix}\vdots\ P_1 \\ \Pi_1\end{matrix} \quad \ldots [i, j\colon A] \quad \begin{matrix}\vdots\ P_m \\ \Pi_m\end{matrix}}{\Pi_1^{(1)}, \ldots, [j\colon A], \ldots, \Pi_m^{(m)}}\ (A)$$

   We can derive the conclusion without a mix as follows:

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\vdots\ P_j \\ \Pi_j\end{matrix}}{[j\colon A, \ldots, A], \Pi_j^{(j)}}\ \mathrm{x}}{[j\colon A], \Pi_j^{(j)}}\ \mathrm{c}}{\Pi_1^{(1)}, \ldots, [j\colon A], \ldots, \Pi_m^{(m)}}\ \mathrm{wx}$$

(b) Π_i is the conclusion of a structural inference. Since the i-th rank is 1, this inference must be a weakening at place i:

$$\dfrac{\begin{array}{c}\vdots\, P_1\\ \Pi_1\end{array} \quad \dots \quad \dfrac{\begin{array}{c}\vdots\, P_i\\ \Pi_i^{(i)}\end{array}}{\Pi_i^{(i)}, [i\colon A]}\,\text{w}\colon i \quad \dots \quad \begin{array}{c}\vdots\, P_m\\ \Pi_m\end{array}}{\Pi_1^{(1)}, \dots, \Pi_i^{(i)}, \dots, \Pi_m^{(m)}}\,(A)$$

where Π_i is $\Pi_i^{(i)}, [i\colon A]$. We obtain the conclusion without a mix as follows:

$$\dfrac{\begin{array}{c}\vdots\, P_i\\ \Pi_i^{(i)}\end{array}}{\Pi_1^{(1)}, \dots, \Pi_i^{(i)}, \dots, \Pi_m^{(m)}}\,\text{wx}$$

(c) All the Π_i are conclusions of introduction rules $\square{:}i$ for the connective $\square$:

$$\dfrac{\dfrac{\langle \Pi_1, \Delta_{\square:1}(r_1)\rangle_{r_1}}{\Pi_1, [1\colon \square(\cdots)]} \quad \dots \quad \dfrac{\langle \Pi_m, \Delta_{\square:m}(r_m)\rangle_{r_m}}{\Pi_m, [m\colon \square(\cdots)]}}{\Pi_1, \dots, \Pi_m}\,(\square(\cdots))$$

By Lemma 3.6.1, there is a derivation of $\Pi_1, \dots, \Pi_m$ from $\Pi_i, \Delta_{\square:i}(r_i)$ $(1 \le i \le m)$, using only mixes, exchanges, and contractions. Since the formulas in the Δ's are subformulas of $\square(\cdots)$, the degrees of all the resulting mixes are less than the degree of the original mix. We iteratively apply the induction hypothesis to the new mixes and obtain a derivation without mixes of $\Pi_1, \dots, \Pi_m$

(d) All the Π_i are conclusions of introduction rules $\mathsf{Q}{:}i$ for the quantifier Q:

$$\dfrac{\dfrac{\langle \Pi_1, \Delta_{\mathsf{Q}:1}(r_1)\rangle_{r_1}}{\Pi_1, [1\colon (\mathsf{Q}x)A(x)]} \quad \dots \quad \dfrac{\langle \Pi_m, \Delta_{\mathsf{Q}:m}(r_m)\rangle_{r_m}}{\Pi_m, [m\colon (\mathsf{Q}x)A(x)]}}{\Pi_1, \dots, \Pi_m}\,((\mathsf{Q}x)A(x))$$

By Lemma 3.6.2, there is a derivation of $\Pi_1, \dots, \Pi_m$ from $\Pi_i, \Delta_{\mathsf{Q}:i}(r_i)\sigma$ $(1 \le i \le m)$, using only mixes, exchanges, and contractions, where σ only substitutes into eigenvariables of the two quantifier introductions. Since the formulas in the Δ's are subformulas of $(\mathsf{Q}x)A(x)$, the degrees of all the resulting mixes are less than the degree of the original mix. We iteratively apply the induction hypothesis to the new mixes and obtain a derivation without mixes of $\Pi_1, \dots, \Pi_m$.

(2) $r_i(P) > 1$ and $r_j(P) = 1$ for all $j < i$: Again, we distinguish cases:

(a) $\Pi_{k\mathbf{1}}$ $(k \neq l)$ contains A: We obtain the following mix-free proof:

$$\cfrac{\cfrac{\cfrac{\begin{array}{c}\vdots\; P_l\\ \Pi_l\end{array}}{\Pi_l^{(l)}, [l\colon A, \dots, A]}\;\mathrm{e}}{\Pi_l^{(l)}, [l\colon A]}\;\mathrm{c}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}}\;\mathrm{wx}$$

(b) Π_i is the consequence of an inference J_2, which is either structural, or a logical inference not introducing A at place i. P is of the form

$$\cfrac{\begin{array}{c}\vdots\; P_1\\ \Pi_1\end{array} \quad \dots \quad \cfrac{\begin{array}{c}\vdots\; P(1)\\ \Psi(1)\end{array} \quad \dots \quad \begin{array}{c}\vdots\; P(p)\\ \Psi(p)\end{array}}{\Pi_i}\;J_2 \quad \dots \quad \begin{array}{c}\vdots\; P_m\\ \Pi_m\end{array}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}}\;A$$

Let $j_1, \dots, j_s$, $1 \leq j_k \leq p$, be all indices s.t. $\Psi(j_k)$ contains A (There is at least one such j_k, otherwise the i-th rank of P would equal 1). Consider the proofs $P(j_k)'$:

$$\cfrac{\begin{array}{c}\vdots\\ \Pi_1\end{array} \quad \dots \quad \begin{array}{c}\vdots\\ \Psi(j_k)\end{array} \quad \dots \quad \begin{array}{c}\vdots\\ \Pi_m\end{array}}{\Pi_1^{(1)}, \dots, \Psi(j_k)^{(i)}, \dots, \Pi_m^{(m)}}\;A$$

In $P(j_k)'$, $r_j(P(j_k)') = r_j(P)$ $(j \neq i)$ and $r_i(P(j_k)') \leq r_i(P) - 1$, and in sum $r(P(j_k)') \leq r(P) - 1$. Hence the induction hypothesis applies and we have mix-free proofs $P(j_k)''$ of $\Pi_1^{(1)}, \dots, \Psi(j_k)^{(i)}, \dots, \Pi_m^{(m)}$. For indices l not occurring in the above list, we have that $\Psi(l)$ equals $\Psi(l)^{(i)}$, and we define $P(l)''$ as

$$\cfrac{\begin{array}{c}\vdots\; P(l)\\ \Psi(l)\end{array}}{\Pi_1^{(1)}, \dots, \Psi(l), \dots, \Pi_m^{(m)}}\;\mathrm{w}$$

If J_2 is a (w:i) (and consequently, $p = 1$ and $\Psi(1)^{(i)} = \Pi_i^{(i)}$), then $P(1)''$ serves as our transformed proof. Otherwise, construct a proof as follows:

$$\cfrac{\cfrac{\cfrac{\begin{array}{c}\vdots\; P(1)''\\ \Pi_1^{(1)}, \dots, \Psi(1)^{(i)}, \dots, \Pi_m^{(m)}\end{array}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}, \Psi(1)^{(i)}}\;\mathrm{x} \quad \dots \quad \cfrac{\begin{array}{c}\vdots\; P(p)''\\ \Pi_1^{(1)}, \dots, \Psi(p)^{(i)}, \dots, \Pi_m^{(m)}\end{array}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}, \Psi(p)^{(i)}}\;\mathrm{x}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}, \Pi_i^{(i)}}\;J_2}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}}\;\mathrm{x}$$

(c) Π_i is the consequence of a logical inference J_2 introducing A at place i. P is of the form

$$\dfrac{\begin{matrix}\vdots\, P_1\\ \Pi_1\end{matrix} \quad \dots \quad \dfrac{\begin{matrix}\vdots\, P(1)\\ \Pi_i, \Delta(1)\end{matrix} \quad \dots \quad \begin{matrix}\vdots\, P(p)\\ \Pi_i, \Delta(p)\end{matrix}}{\Pi_i, [i\colon A]}\, J_2 \quad \dots \quad \begin{matrix}\vdots\, P_m\\ \Pi_m\end{matrix}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}}\,(A)$$

Consider the proofs $P(k)'$ (Note that $\Delta(k)$ does not contain A—only proper subformulas of A—and hence $\Delta(k)^{(i)}$ equals $\Delta(k)$):

$$\dfrac{\begin{matrix}\vdots\, P_1\\ \Pi_1\end{matrix} \quad \dots \quad \begin{matrix}\vdots\, P(k)\\ \Pi_i, \Delta(k)\end{matrix} \quad \dots \quad \begin{matrix}\vdots\, P_m\\ \Pi_m\end{matrix}}{\Pi_1^{(1)}, \dots, \Pi_i^{(i)}, \Delta(k), \dots, \Pi_m^{(m)}}\,(A)$$

In $P(k)$, $r_j(P(k)') = r_j(P)$ $(j \neq i)$, $r_i(P(k)') \leq r_i(P) - 1$ and in sum $r(P(k)') \leq r(P) - 1$. Hence, the induction hypothesis applies and we obtain mix-free proofs $P(k)''$ of $\Pi_1^{(1)}, \dots, \Pi_i^{(i)}, \Delta(k), \dots, \Pi_m^{(m)}$. Construct a proof P' as follows:

$$\dfrac{\begin{matrix}\vdots\, P_1\\ \Pi_1\end{matrix} \quad \dots \quad \dfrac{\dfrac{\begin{matrix}\vdots\, P(1)''\\ \Pi_1^{(1)}, \dots, \Pi_i^{(i)}, \Delta(1), \dots, \Pi_m^{(m)}\end{matrix}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}, \Delta(1)} \quad \dots \quad \dfrac{\begin{matrix}\vdots\, P(p)''\\ \Pi_1^{(1)}, \dots, \Pi_i^{(i)}, \Delta(p), \dots, \Pi_m^{(m)}\end{matrix}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}, \Delta(p)}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}, [i\colon A]}\, J_2 \quad \dots \quad \begin{matrix}\vdots\, P_m\\ \Pi_m\end{matrix}}{\Pi_1^{(1)}, \dots, \Pi_{i-1}^{(i-1)}, \Pi_1^{(1)}, \dots, \Pi_m^{(m)}, \Pi_{i+1}^{(i+1)}, \dots \Pi_m^{(m)}}\,(A)$$

Note that A does not occur at place i in $\Pi_j^{(j)}$ $(j \neq i)$, since otherwise case (2) would have applied, hence $r_i(P') = 1$. With $r_j(P') = r_j(P)$ $(j \neq i)$ we have that $r(P') < r(P)$ and the induction hypothesis yields a mix-free proof P'', whence we have:

$$\dfrac{\begin{matrix}\vdots\, P''\\ \Pi_1^{(1)}, \dots, \Pi_{i-1}^{(i-1)}, \Pi_1^{(1)}, \dots, \Pi_m^{(m)}, \Pi_{i+1}^{(i+1)}, \dots \Pi_m^{(m)}\end{matrix}}{\Pi_1^{(1)}, \dots, \Pi_m^{(m)}}\,\mathrm{xc}$$

This completes the proof of the cut-elimination theorem. ∎

3.7 Analytical Properties of PL

The cut-elimination theorem is often invoked to show certain simple properties of sequent calculus, such as the subformula property, the decidability of propositional logic, or consistency. In fact, these properties do not depend on the cut-elimination theorem itself, i.e., on the availability of an algorithm for eliminating cuts from a given proof, but on the completeness of the cut-free fragment. However, we may want to extract information from cut-free proofs. The cut-elimination algorithm makes the constructive extraction of such information possible, given an arbitrary proof (containing cuts).

The following *subformula property* says that a proof of a statement (i.e., a sequent) need only consist of subformulas of the formulas in the given sequent. This makes the notion precise that a sequent can be proved without "detours," or having to use information unrelated to the end sequent.

3.7.1. PROPOSITION (Subformula Property) *A cut-free* **PL**-*proof contains only formulas which are subformulas of the end-sequent.*

Proof. By induction on the length of proofs: All rules except the cut rule have the property that every formula occurring in the premises also occurs in the conclusion. ■

The midsequent theorem is what Gentzen called his *Verschärfter Hauptsatz*, meaning a strengthening of the cut-elimination theorem. It shows the existence of proofs which are not only cut-free, but also have all quantifier inferences below all propositional inferences.

3.7.2. THEOREM (Midsequent Theorem) *Let Π be a sequent consisting only of prenex formulas which is provable in* **PL**. *Then there is a cut-free proof of Π containing sequents $\Sigma_1, \ldots, \Sigma_p$, s.t.*

(1) Σ_j is quantifier-free,

(2) every inference above Σ_j is either structural or propositional, and

(3) every inference below Σ_j is either structural or a quantifier inference.

Proof. By the cut-elimination theorem and Corollary 3.3.11, there is a cut-free proof P of Π from atomic axiom sequents. The *order* of a quantifier introduction J in P is defined as the number of propositional inferences below J. The *order* $o(P)$ of P is the sum of the orders of all quantifier inferences occurring in P.

We prove the theorem by induction on the order of P:

(1) $o(P) = 0$: There is no propositional inference occurring below any quantifier inferences. Let B be a branch in P. If there is a propositional inference on B, then let Σ_B be the conclusion of the lowermost such, otherwise let Σ_B be the (atomic) axiom sequent on B. If Σ_B contains a quantified formula F, then F has to be introduced by weakenings. To see this, recall that Π contains only prenex formulas: By the subformula property, Σ_B is a subformula of a formula in Π, hence no propositional inferences apply to it. Eliminate F and all inferences applying to it from the part of B above Σ_B, and add appropriate weakenings and exchanges directly below Σ_B. The (finite) set of all Σ_B serves as the set of Σ_i in the statement of the theorem.

(2) $o(P) > 0$: Then there is a quantifier inference I with the following property: The topmost logical inference below I is a propositional inference, say J. The part of P between I and J takes the following form:

$$\dfrac{\begin{array}{c}\vdots\\ \Gamma_1\end{array} \quad \dots \quad \begin{array}{c} \dfrac{\left\langle \begin{array}{c}\vdots\\ \Gamma', \Delta_{\mathsf{Q}:i}(k)\end{array}\right\rangle_k}{\Gamma', [i\colon (\mathsf{Q}x)A(x)]}\,I \\ \vdots\; * \\ \Gamma''\end{array} \quad \dots \quad \begin{array}{c}\vdots\\ \Gamma_n\end{array}}{\Gamma}\,J$$

where the part denoted by $*$ contains only structural inferences, and Γ and Γ'' contains $(\mathsf{Q}x)A(x)$ as a sequent-formula. We can now lower the order by exchanging the positions of I and J:

$$\dfrac{\left\langle \dfrac{\dfrac{\dfrac{\begin{array}{c}\vdots\\ \Gamma_1\end{array}}{\Delta_{\mathsf{Q}:i}(k), \Gamma_1} \quad \dots \quad \begin{array}{c}\dfrac{\dfrac{\begin{array}{c}\vdots\\ \Gamma', \Delta_{\mathsf{Q}:i}(k)\end{array}}{\Gamma', \Delta_{\mathsf{Q}:i}(k), [i\colon (\mathsf{Q}x)A(x)]}\,\mathrm{w}}{\Delta_{\mathsf{Q}:i}(k), \Gamma', [i\colon (\mathsf{Q}x)A(x)]}\,\mathrm{e} \\ \vdots\; * \\ \Delta_{\mathsf{Q}:i}(k), \Gamma''\end{array} \quad \dots \quad \dfrac{\begin{array}{c}\vdots\\ \Gamma_n\end{array}}{\Delta_{\mathsf{Q}:i}(k), \Gamma_n}}{\Delta, \Gamma}\,J}{\Gamma, \Delta_{\mathsf{Q}:i}(k)}\,\mathrm{e} \right\rangle_k}{\dfrac{\Gamma, [i\colon (\mathsf{Q}x)A(x)]}{\Gamma}\,\mathrm{ec}}\,I \qquad \blacksquare$$

In classical logic, the quantifier rules all have only one premise each. From this it follows that there always is *one* midsequent. This midsequent contains, in essence, an Herbrand disjunction of the prenex formula in the end-sequent. If the many-valued logic under consideration has a similar structure, i.e., the quantifiers considered have only one premise, then an analogue of Herbrand's Theorem holds for this logic.

3.8 Interpolation

Interpolation is an interesting problem for logics with something resembling an implication connective. The question asked is, for two given formulas A and B where $A \supset B$, is there a formula C, called an *interpolant* of A and B, s.t. (1) $A \supset C$ and $C \supset B$, (2) C contains only predicate symbols common to A and B and (3) C contains only free variables common to A and B. For classical logic, the interpolation theorem was first proved by Craig [1957]. Maehara [1960] showed the interpolation theorem in a proof theoretical fashion by making use of what has become known as Maehara's Lemma (see also Takeuti [1987], p. 31ff). It also enabled Schütte [1962] to extend the theorem to intuitionistic predicate logic. Maehara's Lemma states that, if a sequent $\Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2$ is provable, then there is an interpolant such that $\Gamma_1, \to \Delta_1, C$ and $C, \Gamma_2 \to \Delta_2$ are provable and the analogues of (2) and (3) hold. We prove a version of Maehara's Lemma in the framework of the p-sequent calculus for many-valued logic. This was first done by MIYAMA [1974] and more recently in a slightly more general form by HANAZAWA and TAKANO [1985] for the formalism of TAKAHASHI [1967a]. GILL [1970] was the first to consider interpolation in three-valued logic.

The interpolation theorem for classical logic follows immediately from Maehara's Lemma since the sequent arrow and the implication are equivalent in the sense that $A \supset B$ is valid iff $A \to B$ is. The theorem is obtained from the lemma simply by taking $\Gamma_1 = A$, $\Delta_2 = B$, and $\Gamma_2 = \Delta_1 = \emptyset$. In the many-valued case of course, no such correspondence holds in general. In fact, Maehara's Lemma is true in any many-valued logic, while the interpolation theorem need not hold. For instance, none of the Łukasiewicz logics (other than two-valued) interpolate, as has been shown by KRZYSTEK and ZACHOROWSKI [1977].

Assume the logic under consideration contains the following truth functions (either as primitive connectives or representable):

(1) For every truth value v the constant truth function $\widetilde{C_v}$

(2) Every unary truth function $\widetilde{G^U_{v,w}}$, where $\widetilde{G^U_{v,w}}(v) = v$ if $v \in V \setminus U$ and $= w$ otherwise. (Note that these are all unary truth function taking only two truth values.)

(3) Every n-ary truth function $\widetilde{H^n_{u,w}}$, where $\widetilde{H^n_{v,w}}(v_1, \ldots, v_n) = w$ if $v_1 = \ldots = v_n = w$ and $= v$ otherwise.

(4) Every quantifier $\widetilde{\mathsf{K}_{v,w}}$, where $\widetilde{\mathsf{K}_{v,w}}(V) = v$ if $V = \{v\}$ and $= w$ otherwise.

We denote by C_i, $G^U_{v,w}$, $H^n_{v,w}$ and $\mathsf{K}_{v,w}$ the corresponding formula schemata, respectively.

3.8.1. THEOREM (Maehara's Lemma) *Let Γ, Λ be a p-provable (and hence p-valid) sequent where $\Gamma \cap \Lambda = \emptyset$. Then there is, for every pair $v \neq w$ of truth values, a formula C, called an interpolant of Γ and Λ, with the following properties:*

(1) $\Gamma, [v{:}\, C]$ and $\Lambda, [w{:}\, C]$ are both provable.

(2) Every predicate symbol of C occurs in both Γ and Λ.

(3) Every free variable of C occurs in both Γ and Λ.

Proof. The proof is by induction on the height h of the proof of Γ, Λ:

$h = 0$: Γ, Λ is an axiom. We distinguish cases:

(1) $\Gamma = \emptyset$: Take for C the constant formula C_v. Since $[u{:}\, C_u]$ is p-valid, $\Gamma, [v{:}\, C_v]$ is provable. Λ is provable by hypothesis, hence $\Lambda, [w{:}\, C_v]$ is provable by weakening.

(2) $\Lambda = \emptyset$: Similarly, with $C = C_w$.

(3) $\Gamma = [U\colon A]$ and $\Lambda = [V \setminus U\colon A]$: Take for C the formula $G^U_{v,w}(A)$. Let $\mathbf{I}$ be some interpretation. By the definition of $\widetilde{G^U_{v,w}}$, C takes the value w if A takes a value from U, and takes v otherwise. So when $\mathbf{I}$ p-satisfies Γ, i.e., does not p-satisfy Λ, then C takes the value w. Conversely, if Γ is not p-satisfied by $\mathbf{I}$, then C takes the value v. From this it follows that $\Gamma, [v\colon C]$ and $\Lambda, [w\colon C]$ are both p-valid (and hence provable by completeness).

$h > 0$: Γ, Λ is the conclusion of a rule of inference J. We distinguish cases according to this last rule in the proof:

(1) J is a weakening at place i: Assume the weakening formula A belongs to Γ (the other case is symmetric). J is of the form:

$$\frac{\Gamma', \Lambda}{\Gamma', [i\colon A], \Lambda}\ \text{w}$$

By induction hypothesis, there is an interpolant C of Γ' and Λ. C is also an interpolant of Γ and Λ, since $\Gamma, [v\colon C]$ is provable from $\Gamma', [v\colon C]$ by weakening, and C contains no predicate symbols or free variables of A which do not occur in Λ.

(2) J is a introduction rule for $\Box$ at place i: Assume the introduced formula $\Box(A_1, \ldots, A_n)$ belongs to Γ (the other case is symmetric). The proof ends in

$$\frac{\Gamma', \Delta_{\Box:i}(1), \Lambda \quad \ldots \quad \Gamma', \Delta_{\Box:i}(p), \Lambda}{\Gamma', [i\colon \Box(A_1, \ldots, A_n)], \Lambda}\ \Box{:}i$$

The induction hypothesis yields interpolants C_j of $\Gamma', \Delta_{\Box:i}(j)$ and Λ ($j = 1, \ldots, p$). Let C equal $H^p_{v,w}(C_1, \ldots, C_p)$. C is an interpolant of Γ and Λ: First, it contains no predicate symbols or free variables not common to Γ and Λ. Furthermore, $\Gamma, [v\colon C]$ and $\Lambda, [w\colon C]$ are both p-valid. Let $\mathbf{I}$ be an interpretation. $\mathbf{I}$ p-satisfies all of $\Gamma', \Delta_{\Box:i}(j), [v\colon C_j]$ ($j = 1, \ldots, p$). This is the case if either (a) $\mathbf{I}$ p-satisfies Γ', (b) $\mathbf{I}$ p-satisfies all of $\Delta_{\Box:i}(j)$, or (c) $\mathbf{I}$ p-satisfies at least one $[v\colon C_j]$. In case (a), $\Gamma', [i\colon \Box(A_1, \ldots, A_n)], [u\colon C]$ is p-satisfied, since Γ' is p-satisfied. In case (b), we know that $\Box(A_1, \ldots, A_n)$ takes the value v_i by the definition of partial normal forms. In case (c), C takes the value v, since not all of the arguments to $H^p_{v,w}$ equal w. In any case, then, we have that $\mathbf{I}$ p-satisfies $\Gamma, [v\colon C]$. On the other hand, if $\mathbf{I}$ does not p-satisfy Λ, it p-satisfies all of $[w\colon C_j]$ ($j = 1, \ldots, p$), since $\Lambda, [w\colon C_j]$ are all p-valid. But then all the arguments to $H^p_{v,w}$ in C equal w, hence C takes the value w. Hence, $\Lambda, [w\colon C]$ is also p-valid.

(3) J is an introduction rule for Q at place i: We will again only treat the case where the introduced formula $[i\colon (\mathsf{Q}x)A(x)]$ belongs to Γ. J takes the form

$$\frac{\Gamma', \Delta_{\mathsf{Q}:i}(1), \Lambda \quad \ldots \quad \Gamma', \Delta_{\mathsf{Q}:i}(p), \Lambda}{\Gamma', [i\colon (\mathsf{Q}x)A(x)], \Lambda}\ \mathsf{Q}{:}i$$

There are interpolants C_j for the premises $\Gamma', \Delta_{\mathsf{Q}:i}(j)$ and Λ ($j = 1, \dots, p$), by induction hypothesis. As in the last case, $D = H^p_{v,w}(C_1, \dots, C_p)$ is such that $\Gamma, [v{:}\, D]$ and $\Lambda, [w{:}\, D]$ are p-valid. Note that the C_j and hence D do not contain eigenvariables of the inference J. To see this, recall that C_j contains only free variables common to $\Gamma', \Delta_{\mathsf{Q}:i}(J)$ and Λ. But eigenvariables cannot have this property, since they must not appear in Λ.

However, since the C_j are interpolants of the premises, they may contain free variables which occur only in terms being replaced by the bound variable x and do not occur in Γ'. These variables occur only in Λ and no longer in $\Gamma', [v_i{:}\, (\mathsf{Q}x)A(x)]$. The truth or falsity of $\Gamma, [v{:}\, D]$ can obviously not depend on the assignment of such a variable in a given interpretation. We can then bind all these variables by the quantifier K.

To be more precise, let $a_1, \dots, a_q$ be all free variables occurring in D which do not occur in Γ. Let $\mathbf{I}$ be an interpretation. If $\Gamma, [v{:}\, D]$ is p-satisfied by $\mathbf{I}'$, then every interpretation $\mathbf{I}' \sim_{a_1,\dots,a_q} \mathbf{I}$ p-satisfies $\Gamma, [v{:}\, D]$ as well. In particular, this holds for every $\mathbf{I}'' \sim_{a_1} \mathbf{I}$. But this means that $\mathrm{Distr}_{\mathbf{I}}(D(x, a_2, \dots, a_q)) = \{u\}$ where $u = \mathrm{val}_{\mathbf{I}}(D(a_1, \dots, a_q))$. Hence, if $\mathbf{I}$ p-satisfies $[v{:}\, D]$ (i.e., $u = v$), then also $\mathbf{I} \models^{\mathrm{p}} [v{:}\, (\mathsf{K}x)D(x, a_2, \dots, a_q)]$. By induction, this holds also for $C = (\mathsf{K}x_1) \dots (\mathsf{K}x_q)D(x_1, \dots, x_q)$. In other words, $\Gamma, [v{:}\, C]$ is p-valid.

On the other hand, the truth of $\Lambda, [w{:}\, D(a_1, \dots, a_q)]$ may depend on the assignment of $a_1, \dots, a_q$, since they occur in Λ. But if there is an interpretation $\mathbf{I}' \sim_{a_1} \mathbf{I}$ s.t. $\mathbf{I} \models^{\mathrm{p}} \Lambda$ and $\mathbf{I}' \not\models^{\mathrm{p}} \Lambda$ but $\mathbf{I}' \models^{\mathrm{p}} [w{:}\, D(a_1, \dots, a_q)]$ this means that $\mathrm{Distr}_{\mathbf{I}}(D(x, a_2, \dots, a_q)) \neq \{v\}$. By the definition of K it holds that $\mathrm{val}_{\mathbf{I}}((\mathsf{K}x)D(x, a_2, \dots, a_q)) = w$. By induction we have that any given interpretation $\mathbf{I}$ either p-satisfies Λ or $[w{:}\, C]$, i.e., $\Lambda, [w{:}\, C]$ is p-valid. ■

In contrast to the proofs of Maehara [1960] and Miyama [1974], the above proof is not entirely proof theoretical in that the p-validity of certain sequents was shown by semantical considerations. This makes it necessary to invoke the completeness theorem. However, this is only due to the fact that the formulas/functions C_v, $G^U_{v,w}$, $H^n_{v,w}$ and $\mathsf{K}_{v,w}$ are, to a certain extent, left unspecified. One could, for instance, require that they be given as primitive connectives and quantifiers in the logic, and fix certain introduction rules for them. If this is done, it seems probable that the demonstrations of p-validity of certain sequents in the proof can be replaced by giving derivation schemata of these sequents. For instance, it should be possible to give a uniform way of deriving $\Gamma', [v_i{:}\, \square(A_1, \dots, A_n)], [v{:}\, C]$ from $\Gamma', \Delta_{\square:i}(j), [v{:}\, C_j]$, like in the proofs of Maehara and Miyama. This has the advantage of not only giving a constructive method of obtaining the interpolant (this is also accomplished by our proof), but also of obtaining proofs of $\Gamma, [v{:}\, C]$ and $\Lambda, [w{:}\, C]$.

Deeper investigations are also needed for exploring the connections between Maehara's Lemma and interpolation as a property of implication. For instance, what does an implication function have to look like so that Maehara's Lemma implies the interpolation property? Can one find simpler, or more natural, families of functions so that Maehara's Lemma can be proved?

Chapter 4

Natural Deduction

4.1 Introduction

GENTZEN [1934] formulated natural deduction for intuitionistic logic as the system **NJ**. In correspondence with the intuitionistic sequent calculus **LJ**, where the right side of a sequent is restricted to at most one formula, **NJ** deals with inference patterns ("Schlußweisen") of *one* conclusion from a set of assumptions. At the application of rules, assumptions of a certain form can be *discharged* in parts of the proof. A proof of a formula is a deduction tree where all assumptions have been discharged.

In **NJ**, the symbol "$\perp$" is used to denote falsehood, or equivalently, an empty conclusion. Gentzen gives introduction and elimination rules for the connectives and quantifiers, as well as the weakening rule for **NJ**:

$$\frac{\perp}{A}$$

Natural deduction for classical logic **NK** is then obtained by adding *tertium non datur* to **NJ**. Alternatively, one can drop the restriction to one formula in the conclusion and allow sets of formulas. For a discussion of the resulting systems see for instance UNGAR [1992], Chapters 4 and 5, or GIRARD [1987], § 2. PARIGOT [1992] has given computational meaning to the multi-conclusion natural deduction calculus for classical logic by normalization and Curry-Howard isomorphism (see also his [1993]).

A general construction of sound and complete natural deduction calculi leads to an adequate syntactical (proof-theoretic) characterization of many-valued logics for which one wants to emphasize the rôle of a particular (set of) truth value(s). We generalize the classical multi-conclusion system of natural deduction to the m-valued case. This is done in essence by splitting the many-valued (positive) sequent into the part corresponding to the designated truth values and the part corresponding to the non-designated ones. The former then corresponds to facts derived, the latter to facts assumed. We give, in a systematic way, *introduction rules* for every connective or quantifier for every designated position, and *elimination rules* for every non-designated position. This system, with the set of designated truth values restricted to $\{v_m\}$, has been presented in BAAZ *et al.* [1993b].

In contrast to the many different (more or less equivalent) approaches in the literature to sequent calculi and tableaux calculi, natural deduction has not been treated in this fashion before (to the best of the author's knowledge). For the three-valued logic of Łukasiwewicz, however, natural deduction systems have been developed, e.g., the calculus of White [1980]. The approach of Becchio and Pabion [1977] is very similar to the one used here. In fact, in the same paper they first consider three-part sequents to formalize $\mathbf{Ł}_3$. This is their starting point for a natural deduction system, which uses the operator M to indicate that a formula takes the intermediate truth value. Still, their formulation is closer to the original **NJ**.

Natural deduction is concerned with validity, or with valid patterns of inference. Hence, we make use of the theory of positive sequent calculi only. This seems more natural, as the positive sequent calculus is more closely related to the notion of validity, in the same way as the negative system is more closely related to unsatisfiability. It is obvious how to adapt the presentation for a dual system of negative natural deduction. All the results of this chapter remain valid for the negative case.

4.2 Natural Deduction Systems

Let Γ be a (set) sequent, $V^+ \subseteq V$ a set of *designated truth values*. The set of non-designated truth values is then $V^- = V \setminus V^+$. We divide the sequent Γ into its designated part Γ^+ and its non-designated part Γ^- in the obvious way:

$$\begin{aligned} \Gamma^+ &:= \langle \Gamma_{\mathbf{i}} \mid i = 1, \dots, m; v_i \in V^+ \rangle \\ \Gamma^- &:= \langle \Gamma_{\mathbf{i}} \mid i = 1, \dots, m; v_i \in V^- \rangle \end{aligned}$$

In natural deduction, we deal with inferences from certain assumptions Γ^- to certain conclusions Γ^+. Such an inference is written thus:

$$\begin{array}{c} \Gamma^- \\ \Gamma^+ \end{array}$$

The semantics employed here are the positive semantics for sequents. More precisely, an interpretation $\mathbf{I}$ makes an inference pattern true, if $\mathbf{I}$ p-satisfies Γ. In other words, whenever all formulas in Γ^- take a truth value *not* corresponding to the place at which they stand, then some formula in Γ^+ *does* take a truth value corresponding to its place. In the two-valued case we have the following situation: $V = \{t, f\}$, $V^+ = \{t\}$. An inference schema is of the form

$$\begin{array}{c} \Gamma \\ \Delta \end{array}$$

The meaning of this is that if everything in Γ is not false (i.e., true), then something in Δ is true. Hence, to say that it means "if Γ is true, then Δ is true" is misleading, since we are dealing with a *con*junction over Γ but a *dis*junction over Δ. The disanalogy is even more apparent in the many-valued case. Of course, in intuitionistic natural deduction, Δ is a singleton, so the problem only arises in the classical multiple-deduction calculus.

The natural deduction calculus acts on such inferences by giving rules how some of these inferences can be combined or altered to give new inferences. This will be made precise later. First, we give the deduction rules. These are based on the introduction rules of the positive sequent calculus, modified as described in Section 3.4 to deal with set-sequents and combinational rules.

4.2.1. DEFINITION Let f be either $\Box$ or Q, and let F be $\Box(A_1, \dots, A_n)$ or $(\mathsf{Q}x)A(x)$, respectively. Let an introduction rule for f at place $i \in V^+$ be given as in Definition 3.4.1. The (natural deduction) *introduction rule* f:I$_i$ *for* f *at place* $i \in V^+$ is given by:

$$\frac{\left\langle \begin{array}{c} \Gamma_j^-, \lceil \Delta'_{f:i}(j)^- \rceil \\ \Gamma_j^+, \Delta'_{f:i}(j)^+ \end{array} \right\rangle_{j \in I}}{\Gamma^+, [i\colon F]}$$

The formulas in $\lceil \dots \rceil$ are those which can be *discharged* at this inference. If f is Q, then the rule has to satisfy the appropriate eigenvariable conditions: The eigenvariables in $\Delta'_{f:i}(j)$ $(j \in I)$ must not occur in Γ or in $F = (\mathsf{Q}x)A(x)$.

The partial sequents $\Gamma_j^+, \Delta'_{f:i}(j)$ are called the *premises* of the rule; $\Gamma^+, [i\colon F]$ is called the *conclusion*.

4.2.2. DEFINITION Let f be either $\Box$ or Q, and let F be $\Box(A_1, \dots, A_n)$ or $(\mathsf{Q}x)A(x)$, respectively. Let an introduction rule for f at place $i \in V^-$ be given as in Definition 3.4.1. The *elimination rule* f:E$_i$ *for* f *at place* $i \in V^-$ is given by:

$$\frac{\begin{array}{c} \Gamma_0^-, \lceil [V^- \setminus v_i\colon F] \rceil \\ \Gamma_0^+, [V^+\colon F] \end{array} \quad \left\langle \begin{array}{c} \Gamma_j^-, \lceil \Delta'_{f:i}(j)^- \rceil \\ \Gamma_j^+, \Delta'_{f:i}(j)^+ \end{array} \right\rangle_{j \in I}}{\Gamma_0^+, \Gamma^+}$$

The formulas in square brackets are those which can be *discharged* at this inference. If f is Q, then the rule has to satisfy the appropriate eigenvariable conditions: The eigenvariables in $\Delta'_{f:i}(j)$ $(j \in I)$ must not occur in Γ_0, Γ or in $F = (\mathsf{Q}x)A(x)$.

The partial sequent $\Gamma_0^+, [V^+\colon F]$ is called the *major premise* of the rule, the partial sequents $\Gamma_j^+, \Delta_{f:i}(j)$ are called the *minor premises* of the rule, and $\Gamma_0^+ \Gamma^+$ is called the *conclusion*.

4.2.3. EXAMPLE The introduction rule for $\supset$ in the logic $\mathbf{Ł}_3$ is:

$$\frac{\begin{array}{cc} \Gamma, \lceil A \rceil \mid \Delta, \lceil A \rceil & \Gamma', \lceil A \rceil \mid \Delta', \lceil B \rceil \\ \Pi, B & \Pi', B \end{array}}{\Pi, \Pi', A \supset B}$$

The elimination rule at place p is:

$$\frac{\begin{array}{cccc} \Gamma, \lceil A \supset B \rceil \mid \Delta & \Gamma'' \mid \Delta'', \lceil A \rceil & \Gamma'' \mid \Delta'', \lceil A, B \rceil & \Gamma''', \lceil A \rceil \mid \Delta''' \rceil \\ \Pi, A \supset B & \Pi', A & \Pi'' & \Pi''', A \end{array}}{\Pi, \Pi', \Pi'', \Pi'''}$$

The elimination rule at place f is:

$$\frac{\begin{array}{ccc} \Gamma \mid \Delta, \lceil A \supset B \rceil & \Gamma'' \mid \Delta'' & \Gamma'', \lceil B \rceil \mid \Delta'' \\ \Pi, A \supset B & \Pi', A & \Pi'' \end{array}}{\Pi, \Pi', \Pi''}$$

4.2.4. *Remark* Note that some of Gentzen's original rules for **LK** are different from the those as obtained by Definition 3.3.1, in that a rule for a given place is split into two, which *together* give a complete characterization of the connective. These rules can also be translated into natural deduction rules as above.

4.2.5. Definition A *natural deduction system* for a logic **L** is given by:

(1) Assumptions of the form $[V^-\colon A]$ where A is any formula,

(2) For every connective $\Box$ and every truth value v_i an introduction rule $\Box$:I$_i$ (if $v_i \in V^+$) or an elimination rule $\Box$:E$_i$ (if $v_i \in V^-$);

(3) For every quantifier Q and every truth value v_i an introduction rule Q:I$_i$ (if $v_i \in V^+$) or an elimination rule Q:E$_i$ (if $v_i \in V^-$);

(4) The weakening rule for all $v_i \in V^+$:

$$\frac{\begin{array}{c}\Gamma^- \\ \Gamma^+\end{array}}{\Gamma^+, [i\colon A]}\ \mathrm{w}\colon i$$

Weakenings are considered as introductions.

4.2.6. Definition A *natural deduction derivation* is defined inductively as follows:

(1) Let A be any formula. Then

$$\frac{[V^-\colon A]}{[V^+\colon A]}$$

is a derivation of A from the assumption $[V^-\colon A]$ (an *initial derivation*).

(2) If D_k are derivations of Γ_k^+, Δ_k^+ from the assumptions $\Gamma_k^-, \hat{\Delta}_k^-$, and

$$\frac{\left\langle \begin{array}{c} \Gamma_k^-, \lceil \Delta_k^- \rceil \\ \Gamma_k^+, \Delta_k^+ \end{array} \right\rangle_{k \in K}}{\Pi^+}$$

is an instance of a deduction rule with $\hat{\Delta}_k^-$ a subsequent of Δ_k^-, and all eigenvariable conditions are satisfied, then

$$\frac{\langle D_k \rangle_{k\in K}}{\Pi^+}$$

is a derivation of Π^+ from the assumptions $\bigcup_{k\in K} \Gamma_k^-$. The formulas in $\hat{\Delta}_k^-$ which do not occur in $\bigcup_{k\in K} \Gamma_k^-$ are said to be *discharged* at this inference.

4.2.7. DEFINITION We call a formula occurrence A

(1) the *conclusion formula* of an introduction, if it is the formula being introduced, i.e., it is F in the conclusion $[i\colon F]$;

(2) a *premise formula* of an introduction, if it is one of the formulas in $\Delta'_{f:i}(j)^+$ in that introduction;

(3) a *major premise formula* of an elimination, if it is among the formula being eliminated, i.e., in the major premise $[V^+\colon F]$;

(4) a *minor premise formula* of an elimination, if it is among the formulas in $\Delta'_{f:i}(j)^+$ in that elimination,

(5) a *discharged assumption formula* of an elimination, if it stands immediately below an assumption which contains the formulas in $\Delta'_{f:i}(j)^-$ being discharged at that elimination.

A formula occurrence A is said to *follow* A', if both are of the same form and A' stands immediately above A at the same position.

4.2.8. THEOREM (Soundness) *If a partial sequent Γ^+ can be derived from the assumptions Γ^-, then the following holds for every interpretation $\mathcal{M}$: If no formula in $\Gamma^-{}_{\mathbf{i}}$ ($v_i \in V^-$) evaluates to the truth value v_i, then there a $v_j \in V^+$ and a formula in $\Gamma^+{}_{\mathbf{j}}$ that evaluates to v_j.*

Proof. The statement of the theorem is obviously equivalent to: If Γ^+ can be derived from assumptions Γ^-, then Γ is p-valid. We prove this by inductively translating every derivation D of Γ^+ from Γ^- to a positive sequent calculus proof of Γ:

(1) D is an initial derivation:

$$\begin{array}{c}[V^-\colon A]\\ [V^+\colon A]\end{array}$$

The translated proof $\pi(D)$ is

$$[V\colon A]$$

which is an axiom.

(2) D ends in an introduction rule at place i:

$$\frac{\left\langle \begin{array}{c}\Gamma_j^-, \lceil \Delta'_{f:i}(j)^- \rceil\\ \vdots\, D_j\\ \Gamma_j, \Delta'_{f:i}(j)\end{array}\right\rangle_{j\in I}}{\Gamma^+, F}\ f{:}\mathrm{I}_i$$

The corresponding sequent calculus proof $\pi(D)$ is:

$$\frac{\left\langle \begin{array}{c}\vdots\, \pi(D_j)\\ \Gamma_j, \Delta'_{f:i}(j)\end{array}\right\rangle_{j\in I}}{\Gamma, [i\colon F]}\ f{:}i$$

(3) D ends in an elimination rule at place i:

$$\frac{\begin{array}{c} \Gamma_0^-, \lceil [V^- \setminus v_i : F] \rceil \\ \vdots\, D_0 \\ \Gamma_0^+, [V^+ : F] \end{array} \quad \left\langle \begin{array}{c} \Gamma_j^-, \lceil \Delta'_{f:i}(j)^- \rceil \\ \vdots\, D_j \\ \Gamma_j^+, \Delta'_{f:i}(j)^+ \end{array} \right\rangle_{j \in I}}{\Gamma_0^+, \Gamma^+} \; f{:}\mathrm{E}_i$$

Now let S be the following sequent deduction:

$$\frac{\left\langle \begin{array}{c} \vdots\, \pi(D_j) \\ \Gamma_j, \Delta'_{f:i}(j) \end{array} \right\rangle_{j \in I}}{\Gamma, [i : F]} \; f{:}i$$

The corresponding sequent calculus proof $\pi(D)$ is:

$$\cfrac{\cfrac{\cfrac{\cfrac{\begin{array}{c}\vdots\, \pi(D_0) \\ \Gamma_0, [V \setminus \{v_i\} : F]\end{array} \quad \begin{array}{c}\vdots\, S \\ \Gamma, [i : F]\end{array}}{\Gamma_0, \Gamma, [V \setminus \{v_1, v_i\} : F]}\,\text{cut:}1i \quad \begin{array}{c}\vdots\, S \\ \Gamma, [i : F]\end{array}}{\Gamma_0, \Gamma, [V \setminus \{v_1, v_2, v_i\} : F]}\,\text{cut:}2i}{\begin{array}{c}\vdots \\ \Gamma_0, \Gamma, [v_m : F]\end{array}} \quad \begin{array}{c}\vdots\, S \\ \Gamma, [i : F]\end{array}}{\Gamma_0, \Gamma}\,\text{cut:}mi$$

(4) D ends in a weakening at place i: Add a weakening at place i to the translated sequent calculus proof.

Note that eigenvariable conditions remain satisfied in $\pi(D)$. ■

4.2.9. *Remark* Translating sequent rules for two-valued logic yield natural deduction elimination rules which differ from those given by Gentzen. However, Gentzen's rules can be obtained in a systematic way by a simplification of the constructed rules. The resulting schema falls outside of our definition of natural deduction rules. We demonstrate this simplification *pars pro toto* for the $\forall$-elimination rule. The classical version as given by Parigot [1992] is:

$$\frac{\begin{array}{c}\Gamma \\ \Delta, (\forall x) A(x)\end{array}}{\Delta, A(t)}$$

The constructed rule is:

$$\frac{\begin{array}{c}\Gamma \\ \Delta, (\forall x) A(x)\end{array} \qquad \begin{array}{c}\Gamma', \lceil A(t) \rceil \\ \Delta'\end{array}}{\Delta, \Delta'}$$

Taking $\{A(t)\}$ for Δ' and $\emptyset$ for Γ', we obtain Parigot's rule by disregarding the redundant right premise.

4.3 Normal Derivations

A maximum segment in the intuitionistic natural deduction calculus **NJ** is a sequence of formulas in a derivation that starts with an introduction and end with an elimination. In the classical multi-conclusion system, it is a sequence starting with an introduction of a formula and ending in an elimination acting on the same formula. A maximum segment constitutes a redundancy in the proof. In **NJ**, and also in multi-valued natural deduction, there are always proofs without such redundancies (see PRAWITZ [1971]).

4.3.1. DEFINITION A sequence $A_1, \ldots, A_r$ of occurrences of *one and the same* formula is called a *maximum segment*, if A_1 is the conclusion formula of an introduction, A_{j+1} stands immediately below A_j, and A_r is the the major premise formula in an elimination.

4.3.2. DEFINITION A *normal derivation* is a natural deduction derivation where no major premise of an elimination stands below an introduction.

4.3.3. PROPOSITION *A normal derivation contains no maximum segments.*

4.3.4. THEOREM *Every cut-free sequent calculus proof of a sequent Λ can be translated into a normal natural deduction derivation of Λ^+ from the assumptions $\hat{\Lambda}^-$, where $\hat{\Lambda}^- \subseteq \Lambda^-$.*

Proof. By induction on the height h of the proof P of Λ.

$h = 1$: Then Λ is an axiom of the form $[V\colon A]$. The corresponding natural deduction proof $\delta(P)$ is the initial derivation

$$\frac{[V^-\colon A]}{[V^+\colon A]}$$

$h > 1$: We distinguish cases according to the last rule in P:

(1) P ends in an introduction rule for f at place $v_i \in V^+$. Λ is of the form $\Gamma, [i\colon F]$:

$$\frac{\left\langle \begin{array}{c} \vdots\, P_j \\ \Gamma_j, \Delta'_{f:i}(j) \end{array} \right\rangle_{j \in I}}{\Gamma, [i\colon F]}\ f{:}i$$

Construct a natural deduction proof $\delta(P)$ of $\Gamma^+, [i\colon F]$ from the assumptions Γ^- as follows:

$$\frac{\left\langle \begin{array}{c} \Gamma_j^-, \lceil \Delta'_{f:i}(j)^- \rceil \\ \vdots\, \delta(P_j) \\ \Gamma_j^+, \Delta'_{f:i}(j)^+ \end{array} \right\rangle_{j \in I}}{\Gamma^+, [i\colon F]}\ f{:}\mathrm{I}_i$$

This only adds an introduction at the end of the derivation, hence $\delta(P)$ is normal.

(2) P ends in an introduction rule at place $v_i \in V^-$: $\varLambda$ is of the form $\varGamma, [i\colon F]$:

$$\dfrac{\left\langle \begin{array}{c} \vdots\ P_j \\ \varGamma_j, \varDelta'_{f:i}(j) \end{array} \right\rangle_{j \in I}}{\varGamma, [i\colon F]}\ f{:}i$$

Construct a natural deduction proof $\delta(P)$ of $\varGamma^+$ from the assumptions $\varGamma^-, [i\colon F]$ as follows:

$$\dfrac{\begin{array}{c} \lceil [V \setminus v_i\colon F] \rceil, [v_i\colon F] \\ [V^+\colon F] \end{array} \quad \left\langle \begin{array}{c} \varGamma_j^-, \lceil \varDelta'_{f:i}(j)^- \rceil \\ \vdots\ \delta(P_j) \\ \varGamma_j^+, \varDelta'_{f:i}(j)^+ \end{array} \right\rangle_{j \in I}}{\varGamma_m}\ f{:}\mathrm{E}_i$$

This only adds an elimination at the beginning of a normal derivation, hence $\delta(P)$ is normal.

(3) P ends in a weakening at place $v_i \in V^+$: Append a weakening at v_i to the natural deduction proof.

(4) P ends in a weakening at place $v_i \in V^-$: Do nothing.

Note that eigenvariable conditions remain satisfied in the translated proof. ∎

4.3.5. Corollary (Completeness) *Natural deduction systems are complete.*

Proof. By Theorem 3.3.10, cut-free sequent calculus is complete, hence every valid sequent $\varLambda$ has a cut-free sequent calculus proof. The translation of this proof yields a natural deduction derivation of $\varLambda^+$ from the assumptions $\hat{\varLambda}^-$ ($\hat{\varLambda}^- \subseteq \varLambda^-$). Every valid inference schema is of this form. ∎

4.3.6. Corollary (Normal Form Property) *For every natural deduction derivation, there exists a normal natural deduction derivation of the same set of formulas from a subset of the assumptions.*

Proof. If there is a derivation of $\varLambda^+$ from $\varLambda^-$ then by Theorem 4.2.8 there is a cut-free sequent calculus proof of $\varLambda$, whose translation yields a normal derivation. ∎

4.3.7. Definition A *path* in a natural deduction derivation is a sequence of occurrences of formulas $A_1, \ldots, A_r$ s.t.

(1) A_1 is either

  (a) a formula standing immediately below an assumption or

  (b) is the conclusion formula of an introduction *without premise formulas* (e.g., weakenings);

(2) A_r is either

  (a) an end formula of the derivation or
  (b) a minor premise formula of an elimination or
  (c) a major premise formula of an elimination *without discharged assumption formulas*, and

(3) A_{j+1} $(1 \le j \le r-1)$ is either

  (a) a discharged assumption formula of an elimination rule, if A_j is the major premise formula of that elimination, or
  (b) the conclusion formula of an introduction if A_j is a premise formula of that rule, or
  (c) follows A_j.

4.3.8. PROPOSITION *A path in a normal derivation can be divided into three (possibly empty) parts:*

(1) The analytical part $A_1, \ldots, A_p$, where each formula is the major premise formula of an elimination and stands immediately below an assumption; A_j is a subformula of A_{j-1} $(2 \le j \le p)$.

(2) The minimum part $A_{p+1}, \ldots, A_q$; A_j is equal to A_{j+1} $(p \le j \le q)$.

(3) The synthetical part $A_{q+1}, \ldots, A_r$; A_{q+1} is the conclusion formula of an introduction with premise formula A_q; A_{j-1} is a subformula of A_j $(q+1 \le j \le r)$.

4.3.9. *Remark* If a cut-free sequent calculus proof *with atomic axiom sequents* is translated as in the proof of Theorem 4.3.4, the minimum segment in a path with non-empty analytical and synthetical part is atomic.

4.3.10. *Remark* It remains to be investigated for which collections of operators one can achieve strong normalization (i.e. normal form transformations with Church-Rosser property) according to some reasonable definition. (See PARIGOT [1992] for positive and ZUCKER [1974] for negative results in the two-valued case.)

Chapter 5

Approximating Propositional Logics

5.1 Introduction

It is clear by what has been said so far that many-valued logics are, from the perspective of proof theory, very close to classical logic. In particular, procedures for algorithmic proof search are very similar to those developed for classical two valued logic, both in the concepts used and in their complexity. Furthermore, the structural proximity to classical logic allows the translation or adaptation of many strategies and heuristics which have been developed for classical logic. This is especially true of propositional many-valued logic. The resolution method can be carried much further than in Chapter 2, see for instance BAAZ and FERMÜLLER [1992] or BAAZ and FERMÜLLER [1993]. The method of tableaux, a calculus very similar to the (negative) sequent calculus, can also be used for mechanical theorem proving in many-valued logics, see HÄHNLE [1993a]. This approach has in fact produced an efficient tautology checker for many-valued logics (see HÄHNLE *et al.* [1992]). Although satisfiability in many-valued propositional logics is (as in classical logic) NP-complete (see MUNDICI [1987]), this is still much better than several other propositional formalisms currently under investigation. Especially in the area of logics for Artificial Intelligence, the corresponding decision problems are hard or complete for classes higher up in the polynomial hierarchy (see, e.g., EITER and GOTTLOB [1992]). From a practical point of view it seems natural to ask for other logics, which in a sense approximate these more complex logics to a satisfactory degree, but are not as complex themselves. The remarks above suggest that many-valued logic might be an appropriate candidate for this task. Of course, the results in this chapter are far from being applicable for real applications. But they are, if not a first step in a promising direction, at least theoretically appealing.

5.2 Propositional Logics

5.2.1. DEFINITION A *propositional language* $\mathcal{L}$ consists of the following:

(1) propositional variables: $X_0, X_1, X_2, \ldots, X_j, \ldots$ $(j \in \omega)$

(2) propositional connectives of arity n_j: $\Box_0^{n_0}, \Box_1^{n_1}, \ldots, \Box_r^{n_r}$. If $n_j = 0$, then

$\Box_j$ is called a *propositional constant*.

(3) Auxiliary symbols: (,), and , (comma).

Formulas are defined as usual. We denote the set of formulas over a language $\mathcal{L}$ by $\mathrm{Frm}(\mathcal{L})$, or by simply by Frm if the language is understood. A propositional many-valued logic $\mathbf{M}$ is given by a set of truth values $V(\mathbf{M}) = \{1, 2, \ldots, m\}$, the set of *designated truth values* $V^+(\mathbf{M})$, and a set of truth functions $\widetilde{\Box}_i\colon V(\mathbf{M})^i \to V(\mathbf{M})$ for all connectives $\Box_i$.

5.2.2. Example Examples are provided by classical propositional logic and propositional three-valued Łukasiewicz logic. These are given by the truth functions for the connectives given in Examples 1.3.2 and 1.3.3. Furthermore, we have the sequence of m-valued Gödel logics $\mathbf{G}_m$ given by $V(\mathbf{G}_m) = \{0, 1, \ldots, m-1\}$, the designated values $V^+(\mathbf{G}_m) = \{m-1\}$, and the following truth functions:

$$\begin{aligned}
\widetilde{\neg}_{\mathbf{G}_m}(v) &= \begin{cases} m-1 & \text{for } v = 0 \\ 0 & \text{for } v \neq 0 \end{cases} \\
\widetilde{\vee}_{\mathbf{G}_m}(v, w) &= \max(a, b) \\
\widetilde{\wedge}_{\mathbf{G}_m}(v, w) &= \min(a, b) \\
\widetilde{\supset}_{\mathbf{G}_m}(v, w) &= \begin{cases} m-1 & \text{for } v \leq w \\ w & \text{for } v > w \end{cases}
\end{aligned}$$

This sequence of logics was used in Gödel [1932] to show that intuitionistic logic can not be characterized by a finite matrix.

5.2.3. Definition A *valuation* $\mathbf{I}$ is a mapping from the set of propositional variables into $V(\mathbf{M})$. A valuation $\mathbf{I}$ can be extended in the standard way to a function from formulas to truth values. $\mathbf{I}$ *satisfies* a formula F, in symbols: $\mathbf{I} \models_{\mathbf{M}} F$, if $\mathbf{I}(F) \in V^+(\mathbf{M})$. In that case, $\mathbf{I}$ is called a *model* of F, otherwise a *countermodel*. A formula F is a *tautology* of $\mathbf{M}$ iff it is satisfied by every valuation. Then we write $\mathbf{M} \models F$. We denote the set of tautologies of $\mathbf{M}$ by $\mathrm{Taut}(\mathbf{M})$.

5.2.4. Definition A propositional logic $\mathbf{L}$ in the language $\mathcal{L}$ is given by a finite set of propositional rule schemas $R(\mathbf{L})$ being a subset of Frm^*. Rules of length 1 are called *axioms*. A formula F is a *theorem* of $\mathbf{L}$, if there is a derivation of F from $R(\mathbf{L})$, i.e., a finite sequence

$$F_1, F_2, \ldots, F_n = F$$

of formulas, s.t. for each F_i there is a rule $\langle A_1, \ldots, A_n \rangle \in R(\mathbf{L})$ where F_i is a substitution instance of A_n, and there are $k_1, \ldots, k_{n-1} < i$ s.t. F_{k_j} is a substitution instance of A_j $(1 \leq j \leq n-1)$. In that case we write $\mathbf{L} \vdash F$. The set of theorems of $\mathbf{L}$ is denoted by $\mathrm{Thm}(\mathbf{L})$.

5.2.5. *Remark* Propositional logics as defined here are always *finitely axiomatized* and have the substitution property. Many well-known propositional logics are finitely axiomatizable, e.g., classical and intuitionistic propositional logic, as well as many modal logics. The assumption of finite axiomatizability is used essentially in some proofs.

5.2.6. EXAMPLE Intuitionistic propositional logic **H** is given by the following axioms (We give the axiomatization of HEYTING [1930]):

$$\begin{array}{ll}
a_1 & A \supset A \wedge A \\
a_2 & A \wedge B \supset B \wedge A \\
a_3 & A \supset B \supset (A \wedge B \supset B \wedge C) \\
a_4 & (A \supset B) \wedge (B \supset C) \supset (A \supset C) \\
a_5 & B \supset (A \supset B) \\
a_6 & A \wedge (A \supset B) \supset B \\
a_7 & A \supset A \vee B \\
a_8 & A \vee B \supset B \vee A \\
a_9 & (A \supset C) \wedge (B \supset C) \supset (A \vee B \supset C) \\
a_{10} & \neg A \supset A \supset B \\
a_{11} & (A \supset B) \wedge (A \supset \neg B) \supset \neg A
\end{array}$$

and the following rules (in usual notation):

$$\frac{A \quad B}{A \wedge B}\, r_1 \qquad \frac{A \quad A \supset B}{B}\, r_2$$

5.3 Singular Approximations

First of all, we are interested in the relationship of many-valued logics to arbitrary logics in respect to the formulas they prove. In other words, what we are interested in is the containment relation between Taut(**M**) and Thm(**L**):

5.3.1. DEFINITION A many-valued logic **M** *approximates* a logic **L** iff Thm(**L**) $\subseteq$ Taut(**M**). **M** is called a *(singular) approximation* of **L**.

5.3.2. EXAMPLE Every m-valued logic having the set of formulas Frm($\mathcal{L}$) as its set of tautologies is a singular approximation of any logic. Less trivially, the m-valued Gödel logic $\mathbf{G}_m$ is an m-valued singular approximation of **H**.

If we have an approximation of a given logic **L**, we have a simple test if a formula is not a theorem of **L**: If F is not valid in **M**, then F cannot be a theorem of **L**. Since we cannot hope for the converse to hold, unless **L** is a many-valued logic, we want **M** to have as few theorems as possible. In fact, we would like to be able to *find* the best m-valued approximation for a given logic **L**. This is often possible, since

(1) all m-valued propositional logics for a language $\mathcal{L}$ can be enumerated mechanically, and there is only a finite number of them,

(2) it is decidable of two m-valued logics given by their truth tables, which of them has fewer theorems.

The additional condition which needs to be satisfied is this: It is decidable whether an m-valued logic **M** approximates a given propositional logic **L**. The naive test for this would be to check that **M** satisfies all axioms and rules of **L**. However, this test might fail even though **M** actually *is* an approximation of **L**, e.g., if there are redundant rules. Therefore, we restrict attention to many-valued logics where this test is positive. Such many-valued logics are called *sound approximations* of **L**.

Set inclusion of the sets of theorems orders all m-valued logics for a language into a finite complete lattice. The minimal elements in the lattice restricted to (sound) approximations of a given logic **L** then are the optimal approximations of **L**. We make all this precise.

5.3.3. PROPOSITION *There are $\prod_{j=1}^{r} m^{m^{n_j}}$ many m-valued logics.*

Proof. The number of different truth functions $V^{n_j} \to V$ equals $|V^{V^{n_j}}| = m^{m^{n_j}}$. ∎

5.3.4. PROPOSITION *An m-valued logic $\mathbf{M}$ approximates the logic $\mathbf{L}$ if (not iff)*

(A) for every rule r in $R(L)$: if a valuation of the variables in r makes all the premises of r true, it also makes the conclusion true.

Proof. If: Let $\mathbf{L} \vdash F$. We show that $\mathbf{M} \models F$ by induction on the length l of the derivation in **L**:

$l = 1$: This means F is a substitution instance of an axiom A. By hypothesis, $\mathbf{M} \models A$. Let $\mathbf{I}$ be a valuation of the variables in F. Let $X_1, X_2, \ldots, X_n$ be all the variables in A, let $F = A[B_1/X_1, \ldots, B_n/X_n]$ and let $v_i = \mathbf{I}(B_i)$ $(1 \le i \le n)$. By hypothesis, $\mathbf{M} \models A$ and, *a fortiori*, $\{X_1 \mapsto v_1, \ldots, X_n \mapsto v_n\} \models A$. But this means that $\mathbf{I} \models F$. Hence, $\mathbf{M} \models F$.

$l > 1$. F is the conclusion of a rule $r \in R(L)$. If r is $\langle A_1, \ldots, A_k, A\rangle$, and $X_1, X_2, \ldots, X_n$ are all the variables in $A, A_1, \ldots, A_k$, then the inference has the form

$$\frac{A_1[B_1/X_1, \ldots, B_n/X_n] \quad \ldots \quad A_k[B_1/X_1, \ldots, B_n/X_n]}{F = A[B_1/X_1, \ldots, B_n/X_n]}\ r$$

Let $\mathbf{I}$ be a valuation of the variables in F, and let $v_i = \mathbf{I}(B_i)$ $(1 \le i \le n)$. By induction hypothesis, the premises of r are valid. This implies that, for $1 \le i \le k$, we haveu $\{X_1 \mapsto v_1, \ldots, X_n \mapsto v_n\} \models A_i$. By hypothesis then, $\{X_1 \mapsto v_1, \ldots, X_n \mapsto v_n\} \models A$. But this means that $\mathbf{I} \models F$. Hence, $\mathbf{M} \models F$. ∎

5.3.5. DEFINITION An m-valued logic $\mathbf{M}$ *soundly approximates* a given logic $\mathbf{L}$ iff condition (A) of Proposition 5.3.4 holds.

5.3.6. COROLLARY *It is decidable if a given m-valued logic $\mathbf{M}$ soundly approximates a given propositional logic $\mathbf{L}$.*

Proof. There are only finitely many rules to check, and condition (A) of Proposition 5.3.4 can be checked in finite time. ∎

5.3.7. *Remark* The adjective *sound* has been chosen because a many-valued logic **M** which is an approximation but not a sound approximation of a given logic **L** has the following paradoxical property:

For **M** we can construct the positive sequent calculus as in Chapter 3. Every rule of **L** can be written as a sequent calculus rule as follows: Let $\langle F_1, F_2, \dots, F_n\rangle$ be the rule in question. The corresponding sequent calculus rule is:

$$\frac{\Gamma, [V^+\colon F_1] \quad \dots \quad \Gamma, [V^+\colon F_{n-1}]}{\Gamma, [V^+\colon F_n]}$$

The system resulting from combining these rules with the sequent calculus for **M** becomes unsound when all truth value constants are added.

5.3.8. EXAMPLE The m-valued Gödel logics $\mathbf{G}_m$ are sound approximations of **H**. For instance, take axiom a_3: $B \supset A \supset B$. This is a tautology in $\mathbf{G}_m$, for assume we assign some truth values a and b to A and B, respectively. We have two cases: If $a \leq b$, then $(A \supset B)$ takes the value $m-1$. Whatever b is, it certainly is $\leq m-1$, hence $B \supset A \supset B$ takes the designated value $m-1$. Otherwise, $A \supset B$ takes the value b, and again (since $b \leq b$), $B \supset A \supset B$ takes the value $m-1$.

Modus ponens (r_2) also passes the test: Assume A and $A \supset B$ both take the value $m-1$. This means that $a \leq b$. But $a = m-1$, hence $b = m-1$.

5.3.9. *Remark* In connection with the relation of approximations and sound approximations the following questions are open, which have an influence on the applicability and range of the results in this chapter:

By Proposition 5.3.4 it is decidable whether a given m-valued logic soundly approximates a given logic. Is the corresponding problem decidable or undecidable for approximations in general? Are there reasonable classes of logics where it is decidable? Is there a way to determine, for a given (class of) logic(s), whether it has only sound approximations?

What are the reasons for the discrepancy between approximations and sound approximations? We have the following situation: An m-valued logic **M** and a logic **L** are given. **M** singularly approximates **L**, but the test fails. This may be due to (at least) two reasons: (1) The axiom system for **L** contains a rule for which the test fails, but which is never actually used in a deduction in **L**. (2) If the test fails, this means that there is a rule r in **L** and an assignment to the variables occurring in r, s.t. the premises of r are true but the conclusion is true. Now it is prima facie conceivable that those formulas which, if substituted for the variables in the premises, make those premises provable, actually never take the truth values of the counterexample. For instance, r might be $F(X) \vdash G(X)$, and for an interpretation with $\mathrm{val}(X) = v$ we have that $F(X)$ is true but $G(X)$ is false. But what we would in fact want to test if whether it is possible for a *theorem* of **L** of the form $F(H)$ to be true in **M** while $G(H)$ is false. But it is conceivable that no formula H for which $F(H)$ is a theorem of **L** ever takes the truth value v in **M**. The questions now are: Do such effects ever happen? Do they happen in "interesting" logics? Is it decidable, given a set of rules, whether such things happen?

5.3.10. Definition An m-valued logic $\mathbf{M}_1$ is *better* than $\mathbf{M}_2$, $\mathbf{M}_1 \lhd \mathbf{M}_2$, iff $\mathrm{Taut}(\mathbf{M}_1) \subset Taut(\mathbf{M}_2)$.

5.3.11. Theorem *Let two m-valued logics $\mathbf{M}_1$, $\mathbf{M}_2$ be given. It is decidable whether $\mathbf{M}_1 \lhd \mathbf{M}_2$.*

Proof. It suffices to show the decidability of the following property: There is a formula A, s.t. (*) $\mathbf{M}_2 \models A$ but $\mathbf{M}_1 \not\models A$.

We show this by giving an upper bound on the depth of a minimal formula A satisfying the above property. Since the set of formulas of $\mathcal{L}$ is enumerable, bounded search will produce such a formula iff it exists. Note that the property (*) is decidable by enumerating all assignments.

Let A be a formula that satisfies (*), i.e., there is a valuation $\mathbf{I}$ s.t. $\mathbf{I} \not\models_{\mathbf{M}_1} A$. W.l.o.g. we can assume that A contains at most m different variables: if it contained more, some of them must be evaluated to the same truth value in the counterexample $\mathbf{I}$ for $\mathbf{M}_1 \not\models A$. Unifying these variables leaves (*) intact.

Let $B = \{B_1, B_2, \ldots\}$ be the set of all subformulas of A. Every formula B_j defines an m-valued truth function $f(B_j)$ of m variables where the values of the variables which actually occur in B_j determine the value of $f(B_j)$ via the matrix of $\mathbf{M}_2$. On the other hand, every B_j evaluates to a single truth value $t(B_j)$ in the countermodel $\mathbf{I}$.

Consider the formula A' constructed from A as follows: Let B_i be a subformula of A and B_j be a proper subformula of B_i (and hence, a proper subformula of A). If $f(B_i) = f(B_j)$ and $t(B_i) = t(B_j)$, replace B_i in A with B_j. A' is shorter than A, and it still satisfies (*). By iterating this construction until no two subformulas have the desired property we obtain a formula A^*. This procedure terminates, since A' is shorter than A; it preserves (*), since A' remains a tautology under $\mathbf{M}_2$ (we replace subformulas behaving in exactly the same way under all valuations) and the countermodel $\mathbf{I}$ is also a countermodel for A'.

The depth of A^* is bounded above by $m^{m^m+1} - 1$. This is seen as follows: If the depth of A^* is d, then there is a sequence $A^* = B'_0, B'_1, \ldots, B'_d$ of subformulas of A^* where B'_k is an immediate subformula of B'_{k-1}. Every such B'_k defines a truth function $f(B'_k)$ of m variables in $\mathbf{M}_2$ and a truth valued $t(B'_k)$ in $\mathbf{M}_1$ via $\mathbf{I}$. There are m^{m^m} m-ary truth functions of m truth values. The number of distinct truth function-truth value pairs then is m^{m^m+1}. If $d \geq m^{m^m+1}$, then two of the B'_k, say B'_i and B'_j where B'_j is a subformula of B'_i define the same truth function and the same truth value. But then B'_i could be replaced by B'_j, contradicting the way A^* is defined. ∎

5.3.12. Corollary *It is decidable if two m-valued logics define the same set of tautologies. The non-strict relation $\unlhd$ between m-valued logics is decidable.*

Proof. $\mathrm{Taut}(\mathbf{M}_1) = \mathrm{Taut}(\mathbf{M}_2)$ iff neither $\mathbf{M}_1 \lhd \mathbf{M}_2$ nor $\mathbf{M}_2 \lhd \mathbf{M}_1$. $\mathbf{M}_1 \unlhd \mathbf{M}_2$ iff $\mathbf{M}_1 \lhd \mathbf{M}_2$ or $\mathrm{Taut}(\mathbf{M}_1) = \mathrm{Taut}(\mathbf{M}_2)$. ∎

Let $\simeq$ be the equivalence relation on m-valued logics defined by: $M_1 \simeq M_2$ iff $\mathrm{Taut}(M_1) = \mathrm{Taut}(M_2)$, and let MVL_m be the set of all m-valued logics over L. By $\mathbf{M}_m$ we denote the set of all sets Taut(M) of tautologies of m-valued logics M. The partial order $\langle \mathbf{M}_m, \subseteq \rangle$ is isomorphic to $\langle \mathrm{MVL}_m / \simeq, \trianglelefteq \rangle$.

5.3.13. PROPOSITION *$\langle \mathbf{M}_m, \trianglelefteq \rangle$ is a finite complete partial order.*

Proof. The set of m-valued logics MVL_m is obviously finite, since there are at most $m^{n_1} m^{n_2} \cdots m^{n_c}$ different m-valued matrices for C. $\triangleleft$ is a partial order on $\mathrm{MVL}_m / \simeq$ with the smallest element $\bot := \mathrm{Frm}(\mathcal{L})$ and the largest element $\top := \emptyset$. ∎

The "best" logic is the one without theorems, generated by a matrix where no connective takes a designated truth value *anywhere*. The "worst" logic is the one where every formula of L is a tautology, it is generated by a matrix where every connective takes a designated truth value *everywhere*.

In every complete partial order over a finite set, there exist lub and glb for every two elements of the set. Hence, $\langle \mathbf{M}, \triangle, \bigtriangledown, \bot, \top \rangle$ is a finite complete lattice, where $\triangle$ is the lub in $\trianglelefteq$, and $\bigtriangledown$ is the glb in $\trianglelefteq$. Since $\trianglelefteq$ is decidable and $\mathbf{M}$ can be automatically generated the functions $\triangle$ and $\bigtriangledown$ are computable.

5.3.14. PROPOSITION *The optimal (i.e., minimal under $\subseteq$) m-valued sound approximations of a logic $\mathbf{L}$ are computable.*

Proof. Consider the set $A(\mathbf{L})$ of m-valued sound approximations of $\mathbf{L}$. Since $A(\mathbf{L})$ is finite and partially ordered by $\trianglelefteq$, $A(\mathbf{L})$ contains minimal elements. The relation $\trianglelefteq$ is decidable, hence the minimal sound approximations can be computed. ∎

5.3.15. *Remark* Of course, the best approximation one could wish for is a many-valued logic $\mathbf{M}$ whose tautologies *coincide* with the theorems of $\mathbf{L}$. $\mathbf{L}$ then provides an axiomatization of $\mathbf{M}$. This of course is not always possible, at least for *finite*-valued logics. Lindenbaum has shown that any logic (in our sense, given by a set of rules and with substitution) can be characterized by an *infinite*-valued logic, see ŁUKASIEWICZ and TARSKI [1930]. For a discussion of related questions see also RESCHER [1969], § 24. Note that Rescher's notions of *captures* and *adequate* correspond to our *singular approximation* and *sound approximation*, respectively.

5.4 Sequential Approximations

In the previous section we have shown that it is always possible to obtain the best sound m-valued approximation of a given logic, but there is no way to tell *how good* these approximations are. For all we know, the best approximations for a given logic $\mathbf{L}$ might only be those $\mathbf{M}$ having $\mathrm{Frm}(\mathcal{L})$ as their sets of tautologies, even though $\mathrm{Thm}(\mathbf{L}) \subset \mathrm{Frm}(\mathcal{L})$. It is hardly appropriate, then to call $\mathbf{M}$ an *approximation* of $\mathbf{L}$, since it is far from being "close" to $\mathbf{L}$. But how can we measure this "closeness" of many-valued logics to arbitrary logics? Clearly, a sensible measure is not easily defined by considering *single* many-valued logics. It is certainly more promising to take a step back and look at collections, say infinite sequences of many-valued logics.

5.4.1. Definition
Let a logic $\mathbf{L}$ be given and let $\mathbf{A} = \langle \mathbf{M}_1, \mathbf{M}_2, \mathbf{M}_3, \ldots, \mathbf{M}_j, \ldots \rangle$ $(j \in \omega)$ be a sequence of many-valued logics s.t.

(1) $\mathbf{M}_i \trianglelefteq \mathbf{M}_j$ iff $i \geq j$, and

(2) $\mathbf{M}_i$ is a singular approximation of $\mathbf{L}$.

$\mathbf{A}$ is called a *sequential approximation* of $\mathbf{L}$ iff $\mathrm{Thm}(\mathbf{L}) = \bigcap_{j \in \omega} \mathrm{Taut}(\mathbf{M}_j)$. We say $\mathbf{L}$ is *approximable*, if there is such a sequential approximation for $\mathbf{L}$.

5.4.2. Example Consider the sequence $\mathbf{G} = \langle \mathbf{G}_2, \mathbf{G}_3, \mathbf{G}_4, \ldots \rangle$ of Gödel logics and intuitionistic propositional logic $\mathbf{H}$. $\mathrm{Taut}(\mathbf{G}_i) \supset \mathrm{Thm}(\mathbf{H})$, since $\mathbf{G}_i$ is a sound approximation of $\mathbf{H}$. Furthermore, $\mathbf{G}_{i+1} \triangleleft \mathbf{G}_i$. This has been pointed out by Gödel [1932], for a detailed proof see Gottwald [1989], Satz 3.4.1. However, it is not a sequential approximation of $\mathbf{H}$: The formula $(A \supset B) \wedge (B \supset A)$, while not a theorem of $\mathbf{H}$, is a tautology of all $\mathbf{G}_i$. In fact, $\bigcap_{j \geq 2} \mathrm{Taut}(\mathbf{G}_i)$ is the set of tautologies of the infinite-valued Gödel logic $\mathbf{G}_\aleph$, which is axiomatized by the rules of $\mathbf{H}$ plus the above formula. This has been shown in Dummett [1959] (see also Gottwald [1989], § 3.4). Hence, $\mathbf{G}$ is a sequential approximation of $\mathbf{G}_\aleph = \mathbf{H} + (A \supset B) \wedge (B \supset A)$.

Jaśkowski [1963] gave a sequential approximation of $\mathbf{H}$. For this see also Rose [1958], Surma [1973], as well as Surma *et al.* [1975]. That $\mathbf{H}$ is approximable is also a consequence of Theorem 5.4.14, with the proof adapted to Kripke semantics for intuitionistic propositional logic, since $\mathbf{H}$ has the fmp (see Gabbay [1981], Ch. 4, Theorem 4(a)).

The notion of approximabilty is of interest in itself, but moreover it tells us something about the quality of singular approximations: As has been pointed out above, there is no way to guarantee that we can come as close to a given logic with our singular approximation as we want to (if we only allow sufficiently many truth values). However, if $\mathbf{L}$ is approximable, then every formula false in $\mathbf{L}$ will also be falsified in a singular approximation of $\mathbf{L}$ with sufficiently many truth values.

The natural question to ask is: Which logics are approximable? Here we can give a negative answer for undecidable logics, and a positive answer for (a class of) logics with the finite model property.

5.4.3. Proposition *Let* $\mathbf{L}$ *be an undecidable propositional logic. Then* $\mathbf{L}$ *is not approximable.*

Proof. Since $\mathbf{L}$ is given by an axiom system, its set of theorems is r.e. Consequently, its set of non-theorems cannot be r.e. If $\mathbf{L}$ were approximable, there were a sequence $\mathbf{A} = \langle \mathbf{M}_1, \mathbf{M}_2, \mathbf{M}_3, \ldots \rangle$ s.t. $\bigcap_{j \geq 2} \mathrm{Taut}(\mathbf{M}_j) = \mathrm{Thm}(L)$. If N is a non-theorem of $\mathbf{L}$, then there would be an index i s.t. N is false in $\mathbf{M}_i$. But this would yield a semi-decision procedure for non-theorems of $\mathbf{L}$: Try for each j whether N is false in $\mathbf{M}_j$. If N is a non-theorem, this will be established at $j = i$, if not, we may go on forever. This contradicts the fact that the non-theorems are not r.e. ∎

5.4.4. EXAMPLE This shows that a result similar to that of JAŚKOWSKI [1963] cannot be obtained for full propositional linear logic. This logic is undecidable, as has been shown by LINCOLN *et al.* [1990] (see also TROELSTRA [1992], Ch. 20).

For the positive result, note that a basic fact about sequential approximations is that every non-theorem of **L** is falsified in some $\mathbf{M}_i$. Compare this to the *finite model property* of modal logics, which says that every non-theorem A of a modal logic **L** is falsified in some finite Kripke model. These countermodels can be coded into many-valued logics which are singular approximations of **L** but which also falsify A, too. By a product construction, this gives a singular approximation of **L**.

The following definitions are taken from CHELLAS [1980].

5.4.5. DEFINITION A *modal logic* **L** has as its language $\mathcal{L}$ the usual propositional connectives plus two unary *modal operators:* $\Box$ (necessary) and $\Diamond$ (possible). A *Kripke model* for $\mathcal{L}$ is a triple $\langle W, R, P\rangle$, where

(1) W is any set: the set of *worlds*,

(2) $R \subseteq W^2$ is a binary relation on W: the *accessibility relation*,

(3) P is a mapping from the propositional variables to subsets of W.

A modal logic **L** is characterized by a class of Kripke models for **L**.

This is called the *standard semantics* for modal logics (see CHELLAS [1980], Ch. 3). The semantics of formulas in standard models is defined as follows:

5.4.6. DEFINITION Let **L** be a modal logic, $\mathcal{K}_{\mathbf{L}}$ be its characterizing class of Kripke models. Let $K = \langle W, R, P\rangle \in \mathcal{K}_{\mathbf{L}}$ be a Kripke model and A be a modal formula.

If $\alpha \in W$ is a possible world, then we say A is *true in* α, $\alpha \models_{\mathbf{L}} A$, iff the following holds:

(1) A is a variable: $\alpha \in P(X)$

(2) $A \equiv \neg B$: not $\alpha \models_{\mathbf{L}} B$

(3) $A \equiv B \wedge C$: $\alpha \models_{\mathbf{L}} B$ and $\alpha \models_{\mathbf{L}} C$

(4) $A \equiv B \vee C$: $\alpha \models_{\mathbf{L}} B$ or $\alpha \models_{\mathbf{L}} C$

(5) $A \equiv \Box B$: for all $\beta \in W$ s.t. $\alpha\, R\, \beta$ it holds that $\beta \models_{\mathbf{L}} B$

(6) $A \equiv \Diamond B$: there is a $\beta \in W$ s.t. $\alpha\, R\, \beta$ and $\beta \models_{\mathbf{L}} B$

We say A is true in K, $K \models_{\mathbf{L}} A$, iff for all $\alpha \in W$ we have $\alpha \models_{\mathbf{L}} A$. A is *valid in* **L**, $\mathbf{L} \models A$, iff A is true in every Kripke model $K \in \mathcal{K}_{\mathbf{L}}$. By Taut(**L**) we denote the set of all formulas valid in **L**.

5.4.7. EXAMPLE The modal logic **S5** is characterized by the class of universal models, i.e., Kripke models where $R = W^2$. It can be shown that it is also characterized by the class of models where R is an equivalence relation (see CHELLAS [1980], Theorem 3.13).

Modal logics as considered in the literature are also propositional logics in the sense of Definition 5.2.4, i.e., they can be axiomatized by a finite set of rules. If this is the case, then the valid formulas of **L**a.k.a. theorems are r.e.

5.4.8. EXAMPLE The modal logic **S5** is axiomatized by the following axioms (see CHELLAS [1980], § 1.2):

T	$\Box A \supset A$
5	$\Diamond A \supset \Box\Diamond A$
K	$\Box(A \supset B) \supset (\Box A \supset \Box B)$
Df$\Diamond$	$\Diamond A \equiv \neg\Box\neg A$

in addition to the usual axioms for classical propositional logic, and the following rules (in usual notation):

$$\frac{A}{\Box A}\,\text{RN} \qquad\qquad \frac{A \quad A \supset B}{B}\,\text{MP}$$

A common method for proving that a modal logic is decidable is connected with the so-called *finite model property:*

5.4.9. DEFINITION Let **L** be a modal logic characterized by $\mathcal{K}$. **L** has the *finite model property (fmp)* iff for every A s.t. $\mathbf{L} \not\models A$, there is a finite Kripke model $K = \langle W, R, P\rangle \in \mathcal{K}$ (i.e., W is finite), s.t. $K \not\models_{\mathbf{L}} A$.

We would like to exploit the fmp of a modal logic L to construct a sequential approximation of **L**. First, we define a product operator on many-valued logics:

5.4.10. DEFINITION Let $\mathbf{M}$ and $\mathbf{M}'$ be m and m'-valued logics, respectively. Then $\mathbf{M} \times \mathbf{M}'$ is the mm'-valued logic where $V(\mathbf{M} \times \mathbf{M}') = V(\mathbf{M}) \times V(\mathbf{M}')$, $V^+(\mathbf{M} \times \mathbf{M}') = V^+(\mathbf{M}) \times V(\mathbf{M}')$, and truth functions are defined componentwise. I.e., if $\Box$ is an n-ary connective, then

$$\widetilde{\Box}_{\mathbf{M}\times\mathbf{M}'}(w_1, \ldots, w_n) = \langle \widetilde{\Box}_{\mathbf{M}}, \widetilde{\Box}_{\mathbf{M}'}\rangle.$$

For convenience, we define the following: Let $\mathbf{I}$ and $\mathbf{I}'$ be valuations of $\mathbf{M}$ and $\mathbf{M}'$, respectively. $\mathbf{I} \times \mathbf{I}'$ is the valuation of $\mathbf{M} \times \mathbf{M}'$ defined by: $(\mathbf{I} \times \mathbf{I}')(X) = \langle \mathbf{I}(X), \mathbf{I}'(X)\rangle$. If $\mathbf{I}^\times$ is a valuation of $\mathbf{M} \times \mathbf{M}'$, then the valuations $\pi_1\mathbf{I}^\times$ and $\pi_2\mathbf{I}^\times$ of $\mathbf{M}$ and $\mathbf{M}'$, respectively, are defined by $\pi_1\mathbf{I}^\times(X) = v$ and $\pi_2\mathbf{I}^\times(X) = v'$ iff $\mathbf{I}^\times(X) = \langle v, v'\rangle$.

5.4.11. LEMMA $\text{Taut}(\mathbf{M} \times \mathbf{M}') = \text{Taut}(\mathbf{M}) \cap \text{Taut}(\mathbf{M}')$

Proof. Let A be a tautology of $\mathbf{M} \times \mathbf{M}'$ and $\mathbf{I}$ and $\mathbf{I}'$ be valuations of $\mathbf{M}$ and $\mathbf{M}'$, respectively. Since $\mathbf{I} \times \mathbf{I}' \models_{\mathbf{M}\times\mathbf{M}'} A$, we have $\mathbf{I} \models_{\mathbf{M}} A$ and $\mathbf{I}' \models_{\mathbf{M}'} A$ by the definition of $\times$. Conversely, let A be a tautology of both $\mathbf{M}$ and $\mathbf{M}'$, and let $\mathbf{I}^{\times}$ be a valuation of $\mathbf{M} \times \mathbf{M}'$. Since $\pi_2\mathbf{I}^{\times} \models_{\mathbf{M}} A$ and $\pi_2\mathbf{I}^{\times} \models_{\mathbf{M}'} A$, it follows that $\mathbf{I}^{\times} \models_{\mathbf{M}\times\mathbf{M}'} A$. ■

The definition and lemma are easily generalized to the case of finite products $\prod_i \mathbf{M}_i$ by induction.

5.4.12. DEFINITION Let $K = \langle W, R, P\rangle$ be a finite Kripke model. We define the many-valued logic $\mathbf{M}_K$ as follows:

(1) $V(\mathbf{M}_K) = \{0,1\}^W$, the set of 0-1-sequences with indices from W.

(2) $V^+(\mathbf{M}_K) = \{1\}^W$, the singleton of the sequence constantly equal to 1.

(3) $\widetilde{\neg}_{\mathbf{M}_K}$, $\widetilde{\vee}_{\mathbf{M}_K}$, $\widetilde{\wedge}_{\mathbf{M}_K}$, $\widetilde{\supset}_{\mathbf{M}_K}$ are defined componentwise from the classical truth functions

(4) $\widetilde{\Box}_{\mathbf{M}_K}$ is defined as follows:

$$\widetilde{\Box}_{\mathbf{M}_K}(\langle w_\alpha\rangle_{\alpha\in W})_\beta = \begin{cases} 1 & \text{if for all } \gamma \text{ s.t. } \beta\, R\, \gamma,\ w_\gamma = 1 \\ 0 & \text{otherwise} \end{cases}$$

(5) $\widetilde{\Diamond}_{\mathbf{M}_K}$ is defined as follows:

$$\widetilde{\Diamond}_{\mathbf{M}_K}(\langle w_\alpha\rangle_{\alpha\in W})_\beta = \begin{cases} 1 & \text{if there is a } \gamma \text{ s.t. } \beta\, R\, \gamma \text{ and } w_\gamma = 1 \\ 0 & \text{otherwise} \end{cases}$$

Furthermore, $\mathbf{I}_K$ is the valuation defined by $\mathbf{I}_K(X)_\alpha = 1$ iff $\alpha \in P(X)$ and $= 0$ otherwise.

5.4.13. LEMMA *Let $\mathbf{L}$ and K be as in Definition 5.4.12. Then the following hold:*

(1) Every valid formula of $\mathbf{L}$ is a tautology of $\mathbf{M}_K$.

(2) If $K \not\models_{\mathbf{L}} A$ then $\mathbf{I}_K \not\models_{\mathbf{M}_K} A$.

Proof. Let B be a modal formula, and $K' = \langle W, R, P'\rangle$. We prove by induction that $\mathrm{val}_{\mathbf{I}_{K'}}(B)_\alpha = 1$ iff $\mathcal{K}' \models_{\mathbf{L}} B$:

B is a variable: $P'(B) = W$ iff $\mathbf{I}_K(B)_\alpha = 1$ for all $\alpha \in W$ by definition of $\mathbf{I}_K$.

$B \equiv \neg C$: By the definition of $\widetilde{\neg}_{\mathbf{M}_K}$, $\mathrm{val}_{\mathbf{I}_K}(B)_\alpha = 1$ iff $\mathrm{val}_{\mathbf{I}_K}(C)_\alpha = 0$. By induction hypothesis, this is the case iff $\alpha \not\models_{\mathbf{L}} C$. This in turn is equivalent to $\alpha \models_{\mathcal{K}} B$. Similarly if B is of the form $C \wedge D$, $C \vee D$, and $C \supset D$.

$B \equiv \Box C$: $\mathrm{val}_{\mathbf{I}_K}(B)_\alpha = 1$ iff for all β with $\alpha\, R\, \beta$ we have $\mathrm{val}_{\mathbf{I}_K}(C)_\beta = 1$. By induction hypothesis this is equivalent to $\beta \models_{\mathbf{L}} C$. But by the definition of $\Box$ this obtains iff $\alpha \models_{\mathbf{L}} B$. Similarly for $\Diamond$.

(1) Every valuation $\mathbf{I}$ of $\mathbf{M}_K$ defines a function $P_{\mathbf{I}}$ via $P_{\mathbf{I}}(X) = \{\alpha \mid \mathbf{I}(X)_\alpha = 1\}$. Obviously, $\mathbf{I} = \mathbf{I}_{P_{\mathbf{I}}}$. If $\mathbf{L} \models B$, then $\langle W, R, P_{\mathbf{I}}\rangle \models_{\mathbf{L}} B$. By the preceding

argument then $\mathrm{val}_{\mathbf{I}}(B)_\alpha = 1$ for all $\alpha \in W$. Hence, B takes the designated value under every valuation.

(2) A is not true in K. This is the case only if there is a world α at which it is not true. Consequently, $\mathrm{val}_{\mathbf{I}_K}(A)_\alpha = 0$ and A takes a non-designated truth value under $\mathbf{I}_K$. ■

5.4.14. THEOREM *Let* $\mathbf{L}$ *be a modal logic with the fmp, and* $\langle A_1, A_2, \ldots\rangle$ *an enumeration of its non-theorems. A sequential approximation of* $\mathbf{L}$ *is given by* $\langle \mathbf{M}_1, \mathbf{M}_2, \ldots\rangle$ *where* $\mathbf{M}_1 = \mathbf{M}_{K_1}$, *and* $\mathbf{M}_{i+1} = \mathbf{M}_i \times \mathbf{M}_{K_{i+1}}$ *where* K_i *is the smallest finite model s.t.* $K_i \not\models_{\mathbf{L}} A_i$

Proof. (1) $\mathrm{Taut}(\mathbf{M}_i) \supseteq \mathrm{Taut}(\mathbf{L})$: By inducton on i: For $i = 1$ this is Lemma 5.4.13 (1). For $i > 1$ the statement follows from Lemma 5.4.11, since $\mathrm{Taut}(\mathbf{M}_{i-1}) \supseteq \mathrm{Taut}(\mathbf{L})$ by induction hypothesis, and $\mathrm{Taut}(\mathbf{M}_{K_i}) \supseteq \mathrm{Taut}(\mathbf{L})$ again by Lemma 5.4.13 (1).

(2) $\mathbf{M}_i \trianglelefteq \mathbf{M}_{i+1}$ from $A \cap B \subseteq A$ and Lemma 5.4.11.

(3) $\mathrm{Taut}(\mathbf{L}) = \bigcap_{i \geq 1} \mathrm{Taut}(\mathbf{M}_i)$. The $\subseteq$-direction follows immediately from (1). Furthermore, by Lemma 5.4.13 (2), no non-tautology of $\mathbf{L}$ can be a member of all $\mathrm{Taut}(\mathbf{M}_i)$, whence $\supseteq$ holds. ■

5.4.15. *Remark* Note that Theorem 5.4.14 does not hold in general if $\mathbf{L}$ is not finitely axiomatizable. This follows from Proposition 5.4.3 and the existence of an undecidable recursively axiomatizable modal logic which has the fmp (see URQUHART [1981]). Note also the condition in Theorem 5.4.14 that there is an enumeration of the non-theorems of $\mathbf{L}$. Since finitely axiomatizable logics with the fmp are decidable (HARROP [1958]), there always is such an enumeration for the logics we consider.

Bibliography

AVRON, A.

[1991] Natural 3-valued logics—characterization and proof theory. *J. Symbolic Logic*, **56**(1), 276–294.

[1992] Axiomatic systems, deduction and implication. *J. Logic Comput.*, **2**(1), 51–98.

[1993] Gentzen-type systems, resolution and tableaux. *J. Automated Reasoning*, **10**, 265–281.

BAAZ, M., C. G. FERMÜLLER, A. OVRUTCKI, and R. ZACH.

[1993] MULTLOG: A system for axiomatizing many-valued logics. In *Logic Programming and Automated Reasoning. Proceedings LPAR'93*, A. Voronkov, editor, LNAI 698, 345–347, Springer, Berlin.

BAAZ, M., C. G. FERMÜLLER, and R. ZACH.

[1993a] Dual systems of sequents and tableaux for many-valued logics. *Bull. EATCS*, **51**. paper read at *2nd Workshop on Tableau-based Deduction*, Marseille, April 1993.

[1993b] Systematic construction of natural deduction systems for many-valued logics. In *Proc. 23rd International Symposium on Multiple-valued Logic*, 208–213, IEEE Press, Sacramento, CA.

BAAZ, M. and C. G. FERMÜLLER.

[1992] Resolution for many-valued logics. In *Logic Programming and Automated Reasoning. Proceedings LPAR'92*, A. Voronkov, editor, LNAI 624, 107–118, Springer, Berlin.

[1993] Resolution-based theorem proving for many-valued logics. submitted.

BAAZ, M. and A. LEITSCH.

[1992] Complexity of resolution proofs and function introduction. *Ann. Pure Appl. Logic*, **57**, 181–215.

BAAZ, M.

[1984] *Die Anwendung der Methode der Sequenzialkalküle auf nichtklassische Logiken.* Dissertation, Universität Wien.

[1989] Automatisches Beweisen für endlichwertige Logiken. *Jahrb. Kurt-Gödel-Gesellschaft*, 105–107.

[1992] Automatisches Beweisen für endlichwertige Logiken. *Mitt. Math. Ges. Hamburg*, **12**(4), 1141–1155.

Becchio, D. and J.-F. Pabion.

[1977] Gentzen's techniques in the three-valued logic of Łukasiewicz (Abstract). *J. Symbolic Logic*, **42**, 123–124.

Beth, E. W.

[1955] Semantic entailment and formal derivability. *Med. Konink. Nederl. Acad. Wetensch. Aft. Letterkunde*, **18**(13), 309–342.

Bolc, L. and P. Borowik.

[1992] *Many-Valued Logics 1. Theoretical Foundations.* Springer, Berlin.

Borowik, P.

[1985] Multi-valued n-sequential propositional logic (Abstract). *J. Symbolic Logic*, **52**, 309–310.

Carnielli, W. A.

[1987a] The problem of quantificational completeness and the characterization of all perfect quantifiers in 3-valued logics. *Z. Math. Logik Grundlag. Math.*, **33**, 19–29.

[1987b] Systematization of finite many-valued logics through the method of tableaux. *J. Symbolic Logic*, **52**(2), 473–493.

[1991] On sequents and tableaux for many-valued logics. *J. Non-Classical Logic*, **8**(1), 59–76.

Chang, C. C. and J. Keisler.

[1966] *Continous Model Theory.* Princeton University Press, Princeton.

Chang, C.-L. and R. C.-T. Lee.

[1973] *Symbolic Logic and Mechanical Theorem Proving.* Academic Press, London.

Chellas, B. F.

[1980] *Modal Logic: An Introduction.* Cambridge University Press, Cambridge.

Cohen, D. I. A.

[1978] *Basic Techniques of Combinatorial Theory.* Wiley, New York.

Craig, W.

[1957] Three uses of the Herbrand-Gentzen theorem in relating model theory and proof theory. *J. Symbolic Logic*, **22**, 269–285.

Dummett, M.

[1959] A propositional calculus with denumerable matrix. *J. Symbolic Logic*, **24**, 97–106.

Eiter, T. and G. Gottlob.

[1992] On the Complexity of Propositional Knowledge Base Revision, Updates, and Counterfactuals. *Artificial Intelligence*, **57**(2–3), 227–270.

Gabbay, D. M.

[1981] *Semantical Investigations in Heyting's Intuitionistic Logic.* Synthese Library 148. Reidel, Dordrecht.

GENTZEN, G.

[1934] Untersuchungen über das logische Schließen I–II. *Math. Z.*, **39**, 176–210, 405–431.

GILL, R. R. R.

[1970] The Craig-Lyndon interpolation theorem in 3-valued logic. *J. Symbolic Logic*, **35**, 230–238.

GINSBERG, M. L.

[1988] Multivalued logics: A uniform approach to reasoning in artificial intelligence. *Comput. Intell.*, **4**, 265–316.

GIRARD, J.-Y.

[1987] Linear logic. *Theoret. Comput. Sci.*, **50**, 1–102.

GÖDEL, K.

[1932] Zum intuitionistischen Aussagenkalkül. *Anz. Akad. Wiss. Wien*, **69**, 65–66.

GOTTWALD, S.

[1989] *Mehrwertige Logik*. Akademie-Verlag, Berlin.

HÄHNLE, R., B. BECKERT, S. GERBERDING, and W. KERNIG.

[1992] The many-valued tableau-based theorem prover $_3T^AP$. IWBS Report, Wissenschaftliches Zentrum Heidelberg, IWBS, IBM Deutschland.

HÄHNLE, R. and W. KERNIG.

[1993] Verification of switch level designs with many-valued logic. In *Logic Programming and Automated Reasoning. Proceedings LPAR'93*, A. Voronkov, editor, LNAI 698, 158–169, Springer, Berlin.

HÄHNLE, R.

[1992] *Tableaux-Based Theorem Proving in Multiple-Valued Logics*. PhD thesis, Universität Karlsruhe, Institut für Logik, Komplexität und Deduktionssysteme.

[1993a] *Automated Proof Search in Multiple-Valued Logics*. Oxford University Press, Oxford.

[1993b] Short conjunctive forms in finitely-valued logics. In *Methodologies for Intelligent Systems. Proceedings ISMIS'93*, J. Komorowski and Z. W. Raś, editors, LNCS 689, Springer, Berlin.

HANAZAWA, M. and M. TAKANO.

[1985] An interpolation theorem in many-valued logic. *J. Symbolic Logic*, **51**(2), 448–452.

HARROP, R.

[1958] On the existence of finite models and decision procedures for propositional calculi. *Proceedings of the Cambridge Philosophical Society*, **54**, 1–13.

HEYTING, A.

[1930] Die formalen Regeln der intuitionistischen Logik. *Sitzungsber. Preuß. Akad. Wiss. Phys.-Math. Kl. II*, 42–56.

HILBERT, D. and P. BERNAYS.

[1939] *Grundlagen der Mathematik II.* Springer.

JAŚKOWSKI, S.

[1963] Recherches sur la système de la logique intuitioniste. In *Actes du Congrès International de Philosophie Scientifique 1936,* **6**, 58–61, Paris.

KIRIN, V. G.

[1966] Gentzen's method for the many-valued propositional calculi. *Z. Math. Logik Grundlag. Math.*, **12**, 317–332.

[1968] Post algebras as semantic bases of some many-valued logics. *Fund. Math.*, **63**, 278–294.

KLEENE, S. C.

[1952] *Introduction to Metamathematics.* North-Holland, Amsterdam.

KOWALSKI, R. and P. J. HAYES.

[1969] Semantic trees in automated theorem proving. In *Machine Intelligence 4*, B. Meltzer and D. Michie, editors, 87–101. Edinburgh University Press, Edinburgh.

KRZYSTEK, P. S. and S. ZACHOROWSKI.

[1977] Łukasiewicz logics have not the interpolation property. *Rep. Math. Logic*, **9**, 39–40.

LEITSCH, A.

[1989] On different concepts of resolution. *Z. Math. Logik Grundlag. Math.*, **35**, 71–77.

[1993] *The Resolution Calculus.* AILA Preprint 15. Associazione Italiana di Logica e sue Applicazioni. Lecture Notes from the 4th European Summer School in Logic, Language, and Information, August 1992.

LINCOLN, P. D., J. MITCHELL, A. SCEDROV, and N. SHANKAR.

[1990] Decision proplems for propositional linear logic. In *Proc. 31st IEEE Symp. Foundations of Computer Science FOCS*, St. Louis, Missouri.

LOVELAND, D. W.

[1978] *Automated Theorem Proving: A Logical Basis.* Fundamental Studies in Computer Science 6. North-Holland, Amsterdam.

ŁUKASIEWICZ, J. and A. TARSKI.

[1930] Untersuchungen über den Aussagenkalkül. *Comptes rendus des séances de la Société des Sciences et des Lettres de Varsovie Cl. III*, **23**, 1–21.

ŁUKASIEWICZ, J.

[1920] O logice tròjwartościowej. *Ruch Filozoficzny*, **5**, 169–171. translated in MCCALL [1967].

MAEHARA, S.

[1960] On the interpolation theorem of Craig (Japanese). *Sugaku*, **12**, 235–237.

McCall, S., editor.
[1967] *Polish Logic 1920–1939.* Oxford University Press, London.

Miyama, T.
[1974] The interpolation theorem and Beth's theorem in many-valued logic. *Math. Japon.*, **19**, 341–355.

Morgan, C. G.
[1976] A resolution principle for a class of many-valued logics. *Logique et Analyse*, 311–339.

Mostowski, A.
[1963] The Hilbert epsilon function in many-valued logics. *Acta Philos. Fenn.*, **16**, 169–188.

Mundici, D.
[1987] Satisfiability in many-valued sentential logic is NP-complete. *Theoret. Comput. Sci.*, **52**, 145–153.

Ohya, T.
[1967] Many valued logics extended to simply type theory. *Sci. Rep. Tokyo Kyoiku Daigaku Sect A*, **9**, 84–94.

Parigot, M.
[1992] $\lambda\mu$-Calculus: an algorithmic interpretation of classical natural deduction. In *Logic Programming and Automated Reasoning. Proceedings LPAR'92*, A. Voronkov, editor, LNAI 624, 190–201, Springer, Berlin.
[1993] The computational meaning of classical proofs. In *Computational Logic and Proof Theory. Proc. 3rd Kurt Gödel Colloquium*, G. Gottlob, A. Leitsch, and D. Mundici, editors, LNCS 731, 263–276, Springer, Berlin.

Post, E. L.
[1921] Introduction to a general theory of propositions. *Amer. J. Math.*, **43**, 163–185. reprinted in van Heijenoort [1967].

Prawitz, D.
[1971] Ideas and results in proof theory. In *Proceedings of the Second Scandinavian Logic Symposium*, J. E. Fenstad, editor, 235–307, North-Holland, Amsterdam.

Przymusińska, H.
[1980a] Gentzen-type semantics for ν-valued infinitary predicate calculi. *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, **28**(5–6), 203–206.
[1980b] Craig interpolation theorem and Hanf number for ν-valued infinitary predicate calculi. *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, **28**(5–6), 207–211.

RASIOWA, H.

[1972] The Craig interpolation theorem for m-valued predicate calculi. *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, **20**(5), 341–346.

[1974] *An Algebraic Approach to Non-classical Logics.* Studies in Logic 78. North Holland, Amsterdam.

RESCHER, N.

[1969] *Many-valued Logic.* McGraw-Hill, New York.

ROBINSON, J. A.

[1965] A machine-oriented logic based on the resolution principle. *J. Assoc. Comput. Mach.*, **12**(1), 23–41.

ROSE, G. F.

[1958] Propositional calculus and realizability. *Trans. Amer. Math. Soc.*, **75**, 1–19.

ROSSER, J. B. and A. R. TURQUETTE.

[1952] *Many-Valued Logics.* Studies in Logic. North-Holland, Amsterdam.

ROUSSEAU, G.

[1967] Sequents in many valued logic I. *Fund. Math.*, **60**, 23–33.

[1970] Sequents in many valued logic II. *Fund. Math.*, **67**, 125–131.

SALONI, Z.

[1972] Gentzen rules for the m-valued logic. *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, **20**(10), 819–826.

SCHRÖTER, K.

[1955] Methoden zur Axiomatisierung beliebiger Aussagen- und Prädikatenkalküle. *Z. Math. Logik Grundlag. Math.*, **1**, 241–251.

SCHÜTTE, K.

[1956] Ein System des verknüpfenden Schließens. *Arch. Math. Logik Grundlag.*, **2**, 55–67.

[1962] Der Interpolationssatz der intuitionistischen Prädikatenlogik. *Math. Ann.*, **148**, 192–200.

SCOTT, D. S.

[1976] Does many-valued logic have any use? In *Philosophy of Logic*, S. Körner, editor, 64–74. Basil Blackwell, Oxford.

STACHNIAK, Z. and P. O'HEARN.

[1990] Resolution in the domain of strongly finite logics. *Fund. Inform.*, **8**, 333–351.

SUCHOŃ, W.

[1974] La méthode de Smullyan de construire le calcul n-valent de Łukasiewicz avec implication et négation. *Rep. Math. Logic*, **2**, 37–42.

SURMA, S. J., A. WROŃSKI, and S. ZACHOROWSKI.

[1975] On Jaśkowski-type semantics for the intuitionistic propositional logic. *Studia Logica*, **34**, 145–148.

SURMA, S. J.
[1973] Jaśkowski's matrix criterion for the intuitionistic propositional calculus. In *Studies in the History of Mathematical Logic*, S. J. Surma, editor, 87–121. Wroclaw.
[1977] An algorithm for axiomatizing every finite logic. In *Computer Science and Multiple-valued Logic: Theory and Applications*, D. C. Rine, editor, 137–143. North-Holland, Amsterdam.

TAKAHASHI, M.
[1967a] Many-valued logics of extended Gentzen style I. *Sci. Rep. Tokyo Kyoiku Daigaku Sect A*, **9**, 271.
[1967b] A proof of cut-elimination in simple type theory. *J. Math. Soc. Japan*, **19**, 399–410.
[1970] Many-valued logics of extended Gentzen style II. *J. Symbolic Logic*, **35**, 493–528.

TAKEUTI, G.
[1987] *Proof Theory*. Studies in Logic 81. North-Holland, Amsterdam, 2nd edition.

TROELSTRA, A. S.
[1992] *Lectures on Linear Logic*. CSLI Lecture Notes 29. CSLI, Standford, CA.

UNGAR, A. M.
[1992] *Normalization, Cut-Elimination and the Theory of Proofs*. CSLI Lecture Notes 28. CSLI, Stanford, CA.

URQUHART, A.
[1981] Decidability and the finite model property. *J. Philos. Logic*, **10**, 367–370.
[1986] Many-valued logic. In *Handbook of Philosophical Logic*, D. M. Gabbay and F. Guenther, editors, volume 3, 71–116. Reidel, Dordrecht.

VAN HEIJENOORT, J., editor.
[1967] *From Frege to Gödel. A Source Book in Mathematical Logic, 1879–1931*. Harvard University Press, Cambridge, MA.

WHITE, R. B.
[1980] Natural deduction in the Łukasiewicz logics. In *Proc. 10th International Symposium on Multiple-valued Logic*, 226–232.

ZINOV'EV, A. A.
[1963] *Philosophical Problems of Many-valued Logic*. Reidel, Dordrecht.

ZUCKER, J.
[1974] The correspondence between cut-elimination and normalization I. *Ann. Math. Logic*, **7**, 1–112.