STUDIES IN LOGIC

AND

THE FOUNDATIONS OF MATHEMATICS

VOLUME 137

S. ABRAMSKY / S. ARTEMOV / R.A. SHORE / A.S. TROELSTRA

EDITORS

HANDBOOK OF PROOF THEORY

SAMUEL R. BUSS (Editor)

ELSEVIER

AMSTERDAM • LAUSANNE • NEW YORK • OXFORD • SHANNON • SINGAPORE • TOKYO

HANDBOOK OF PROOF THEORY

Edited by

SAMUEL R. BUSS
University of California, San Diego

1998

ELSEVIER

AMSTERDAM • LAUSANNE • NEW YORK • OXFORD • SHANNON • SINGAPORE • TOKYO

ELSEVIER SCIENCE B.V.
Sara Burgerhartstraat 25
P.O. Box 211, 1000 AE Amsterdam
The Netherlands

Library of Congress Cataloging-in-Publication Data

Handbook of proof theory / edited by Samuel R. Buss.
p. cm. -- (Studies in logic and the foundations of mathematics ; v. 137)
Includes bibliographic references and indexes.
ISBN 0-444-89840-9 (alk. paper)
1. Proof theory. I. Buss, Samuel R. II. Series.
QA9.54.H35 1998
511.3--dc21 98-18922
CIP

ISBN: 0-444-89840-9

♾ The paper used in this publication meets the requirements of ANSI/NISO Z39.48-1992 (Permanence of Paper).

Printed in The Netherlands

HANDBOOK
OF
PROOF THEORY

Preface

Proof theory is the study of proofs as formal objects and is concerned with a broad range of related topics. It is one of the central topics of mathematical logic and has applications in many areas of mathematics, philosophy, and computer science. Historically, proof theory was developed by mathematicians and philosophers as a formalization for mathematical reasoning; however, proof theory has gradually become increasingly important for computer science, and nowadays proof theory and theoretical computer science are recognized as being very closely connected.

This volume contains articles covering a broad spectrum of proof theory, with an emphasis on its mathematical aspects. The articles should not only be interesting to specialists in proof theory, but should also be accessible to a diverse audience, including logicians, mathematicians, computer scientists and philosophers. We have attempted to include many of the central topics in proof theory; but have opted to have self-contained expository articles, rather than to have encyclopedic coverage. Thus, a number of important topics have been largely omitted, but with the advantage that the included material is covered in more detail and at greater depth.

The chapters are arranged so that the two introductory articles come first; these are then followed by articles from the core classical areas of proof theory; finally the handbook ends with articles that deal with topics closely related to computer science.

This handbook was initiated at the suggestion of the publisher, as a partial successor to the very successful *Handbook of Mathematical Logic*, edited by J. Barwise. Only one quarter of the 1977 *Handbook of Mathematical Logic* was devoted to proof theory, and since then there has been considerable progress in this area; as a result, there is remarkably little overlap between the contents of the *Handbook of Mathematical Logic* and the present volume.

Sam Buss
La Jolla, California
November 1997

List of Contributors

J. Avigad, *Carnegie Mellon University* (Ch. V)

S.R. Buss, *University of California, San Diego* (Ch. I and II)

R.L. Constable, *Cornell University* (Ch. X)

M. Fairtlough, *University of Sheffield* (Ch. III)

S. Feferman, *Stanford University* (Ch. V)

G. Jäger, *Universität Bern* (Ch. IX)

G. Japaridze, *University of Pennsylvania* (Ch. VII)

D. de Jongh, *University of Amsterdam* (Ch. VII)

P. Pudlák, *Academy of Sciences of the Czech Republic* (Ch. VIII)

W. Pohlers, *Westfälische Wilhelms-Universität* (Ch. IV)

R.F. Stärk, *Universität Freiburg* (Ch. IX)

A.S. Troelstra, *University of Amsterdam* (Ch. VI)

S.S. Wainer, *University of Leeds* (Ch. III)

Table of Contents

CHAPTER I

An Introduction to Proof Theory

Samuel R. Buss
Departments of Mathematics and Computer Science, University of California, San Diego
La Jolla, California 92093-0112, USA

Contents

HANDBOOK OF PROOF THEORY
Edited by S. R. Buss

Proof Theory is the area of mathematics which studies the concepts of mathematical proof and mathematical provability. Since the notion of "proof" plays a central role in mathematics as the means by which the truth or falsity of mathematical propositions is established; Proof Theory is, in principle at least, the study of the foundations of all of mathematics. Of course, the use of Proof Theory as a foundation for mathematics is of necessity somewhat circular, since Proof Theory is itself a subfield of mathematics.

There are two distinct viewpoints of what a mathematical proof is. The first view is that proofs are social conventions by which mathematicians convince one another of the truth of theorems. That is to say, a proof is expressed in natural language plus possibly symbols and figures, and is sufficient to convince an expert of the correctness of a theorem. Examples of social proofs include the kinds of proofs that are presented in conversations or published in articles. Of course, it is impossible to precisely define what constitutes a valid proof in this social sense; and, the standards for valid proofs may vary with the audience and over time. The second view of proofs is more narrow in scope: in this view, a proof consists of a string of symbols which satisfy some precisely stated set of rules and which prove a theorem, which itself must also be expressed as a string of symbols. According to this view, mathematics can be regarded as a 'game' played with strings of symbols according to some precisely defined rules. Proofs of the latter kind are called "formal" proofs to distinguish them from "social" proofs.

In practice, social proofs and formal proofs are very closely related. Firstly, a formal proof can serve as a social proof (although it may be very tedious and unintuitive) provided it is formalized in a proof system whose validity is trusted. Secondly, the standards for social proofs are sufficiently high that, in order for a proof to be socially accepted, it should be possible (in principle!) to generate a formal proof corresponding to the social proof. Indeed, this offers an explanation for the fact that there are generally accepted standards for social proofs; namely, the implicit requirement that proofs can be expressed, in principle, in a formal proof system enforces and determines the generally accepted standards for social proofs.

Proof Theory is concerned almost exclusively with the study of formal proofs: this is justified, in part, by the close connection between social and formal proofs, and it is necessitated by the fact that only formal proofs are subject to mathematical analysis. The principal tasks of Proof Theory can be summarized as follows. First, to formulate systems of logic and sets of axioms which are appropriate for formalizing mathematical proofs and to characterize what results of mathematics follow from certain axioms; or, in other words, to investigate the proof-theoretic strength of particular formal systems. Second, to study the structure of formal proofs; for instance, to find normal forms for proofs and to establish syntactic facts about proofs. This is the study of proofs as objects of independent interest. Third, to study what kind of additional information can be extracted from proofs beyond the truth of the theorem being proved. In certain cases, proofs may contain computational or constructive information. Fourth, to study how best to construct formal proofs; e.g., what kinds of proofs can be efficiently generated by computers?

The study of Proof Theory is traditionally motivated by the problem of formalizing mathematical proofs; the original formulation of first-order logic by Frege [1879] was the first successful step in this direction. Increasingly, there have been attempts to extend Mathematical Logic to be applicable to other domains; for example, intuitionistic logic deals with the formalization of constructive proofs, and logic programming is a widely used tool for artificial intelligence. In these and other domains, Proof Theory is of central importance because of the possibility of computer generation and manipulation of formal proofs.

This handbook covers the central areas of Proof Theory, especially the mathematical aspects of Proof Theory, but largely omits the philosophical aspects of proof theory. This first chapter is intended to be an overview and introduction to mathematical proof theory. It concentrates on the proof theory of *classical* logic, especially propositional logic and first-order logic. This is for two reasons: firstly, classical first-order logic is by far the most widely used framework for mathematical reasoning, and secondly, many results and techniques of classical first-order logic frequently carryover with relatively minor modifications to other logics.

This introductory chapter will deal primarily with the sequent calculus, and resolution, and to lesser extent, the Hilbert-style proof systems and the natural deduction proof system. We first examine proof systems for propositional logic, then proof systems for first-order logic. Next we consider some applications of cut elimination, which is arguably the central theorem of proof theory. Finally, we review the proof theory of some non-classical logics, including intuitionistic logic and linear logic.

1. Proof theory of propositional logic

Classical propositional logic, also called sentential logic, deals with sentences and propositions as abstract units which take on distinct True/False values. The basic syntactic units of propositional logic are *variables* which represent atomic propositions which may have value either *True* or *False*. Propositional variables are combined with Boolean functions (also called connectives): a *k-ary Boolean function* is a mapping from $\{T,F\}^k$ to $\{T,F\}$ where we use T and F to represent *True* and *False*. The most frequently used examples of Boolean functions are the connectives $\top$ and $\bot$ which are the 0-ary functions with values T and F, respectively; the binary connectives $\wedge$, $\vee$, $\supset$, $\leftrightarrow$ and $\oplus$ for "and", "or", "if-then", "if-and-only-if" and "parity"; and the unary connective $\neg$ for negation. Note that $\vee$ is the inclusive-or and $\oplus$ is the exclusive-or.

We shall henceforth let the set of propositional variables be $V = \{p_1, p_2, p_3, \ldots\}$; however, our theorems below hold also for uncountable sets of propositional variables. The set of formulas is inductively defined by stating that every propositional variable is a formula, and that if A and B are formulas, then $(\neg A)$, $(A \wedge B)$, $(A \vee B)$, $(A \supset B)$, etc., are formulas. A *truth assignment* consists of an assignment of True/False values to the propositional variables, i.e., a truth assignment is a mapping $\tau : V \to \{T,F\}$.

A truth assignment can be extended to have domain the set of all formulas in the obvious way, according to Table 1; we write $\overline{\tau}(A)$ for the truth value of the formula A induced by the truth assignment τ.

Table 1
Values of a truth assignment $\overline{\tau}$

A	B	$(\neg A)$	$(A \wedge B)$	$(A \vee B)$	$(A \supset B)$	$(A \leftrightarrow B)$	$(A \oplus B)$
T	T	F	T	T	T	T	F
T	F	F	F	T	F	F	T
F	T	T	F	T	T	F	T
F	F	T	F	F	T	T	F

A formula A involving only variables among $p_1, \ldots, p_k$ defines a k-ary Boolean function f_A, by letting $f_A(x_1, ..., x_k)$ equal the truth value $\overline{\tau}(A)$ where $\tau(p_i) = x_i$ for all i. A *language* is a set of connectives which may be used in the formation of *L-formulas.* A language L is *complete* if and only if every Boolean function can be defined by an L-formula. Propositional logic can be formulated with any complete (usually finite) language L — for the time being, we shall use the language $\neg$, $\wedge$, $\vee$ and $\supset$.

A propositional formula A is said to be a *tautology* or to be *(classically) valid* if A is assigned the value T by every truth assignment. We write $\models A$ to denote that A is a tautology. The formula A is *satisfiable* if there is some truth assignment that gives it value T. If Γ is a set of propositional formulas, then Γ is *satisfiable* if there is some truth assignment that simultaneously satisfies all members of Γ. We say Γ *tautologically implies* A, or $\Gamma \models A$, if every truth assignment which satisfies Γ also satisfies A.

One of the central problems of propositional logic is to find useful methods for recognizing tautologies; since A is a tautology if and only if $\neg A$ is not satisfiable, this is essentially the same as the problem of finding methods for recognizing satisfiable formulas. Of course, the set of tautologies is decidable, since to verify that a formula A with n distinct propositional variables is a tautology, one need merely check that the 2^n distinct truth assignments to these variables all give A the value T. This brute-force 'method of truth-tables' is not entirely satisfactory; firstly, because it can involve an exorbitant amount of computation, and secondly, because it provides no intuition as to *why* the formula is, or is not, a tautology.

For these reasons, it is often advantageous to *prove* that A is a tautology instead of using the method of truth-tables. The next three sections discuss three commonly used propositional proof systems. The so-called Frege proof systems are perhaps the most widely used and are based on modus ponens. The sequent calculus systems provide an elegant proof system which combines both the possibility of elegant proofs and the advantage of an extremely useful normal form for proofs. The resolution refutation proof systems are designed to allow for efficient computerized search for proofs. Later, we will extend these three systems to first-order logic.

1.1. Frege proof systems

The mostly commonly used propositional proof systems are based on the use of *modus ponens* as the sole rule of inference. Modus ponens is the inference rule, which allows, for arbitrary A and B, the formula B to be inferred from the two hypotheses $A \supset B$ and A; this is pictorially represented as

$$\frac{A \qquad A \supset B}{B}$$

In addition to this rule of inference, we need *logical axioms* that allow the inference of 'self-evident' tautologies from no hypotheses. There are many possible choices for sets of axioms: obviously, we wish to have a sufficiently strong set of axioms so that every tautology can be derived from the axioms by use of modus ponens. In addition, we wish to specify the axioms by a finite set of schemes.

1.1.1. Definition. A *substitution* σ is a mapping from the set of propositional variables to the set of propositional formulas. If A is a propositional formula, then the result of applying σ to A is denoted $A\sigma$ and is equal to the formula obtained by simultaneously replacing each variable appearing in A by its image under σ.

An example of a set of axiom schemes over the language $\neg$, $\wedge$, $\vee$ and $\supset$ is given in the next definition. We adopt conventions for omitting parentheses from descriptions of formulas by specifying that unary operators have the highest precedence, the connectives $\wedge$ and $\vee$ have second highest precedence, and that $\supset$ and $\leftrightarrow$ have lowest precedence. All connectives of the same precedence are to be associated from right to left; for example, $A \supset \neg B \supset C$ is a shorthand representation for the formula $(A \supset ((\neg B) \supset C))$.

1.1.2. Definition. Consider the following set of axiom schemes:

$p_1 \supset (p_2 \supset p_1)$	$(p_1 \supset p_2) \supset (p_1 \supset \neg p_2) \supset \neg p_1$
$(p_1 \supset p_2) \supset (p_1 \supset (p_2 \supset p_3)) \supset (p_1 \supset p_3)$	$(\neg\neg p_1) \supset p_1$
$p_1 \supset p_1 \vee p_2$	$p_1 \wedge p_2 \supset p_1$
$p_2 \supset p_1 \vee p_2$	$p_1 \wedge p_2 \supset p_2$
$(p_1 \supset p_3) \supset (p_2 \supset p_3) \supset (p_1 \vee p_2 \supset p_3)$	$p_1 \supset p_2 \supset p_1 \wedge p_2$

The propositional proof system $\mathcal{F}$ is defined to have as its axioms every substitution instance of the above formulas and to have modus ponens as its only rule. An $\mathcal{F}$-proof of a formula A is a sequence of formulas, each of which is either an $\mathcal{F}$-axiom or is inferred by modus ponens from two earlier formulas in the proof, such that the final formula in the proof is A.

We write $\vdash_{\mathcal{F}} A$, or just $\vdash A$, to say that A has an $\mathcal{F}$-proof. We write $\Gamma \vdash_{\mathcal{F}} A$, or just $\Gamma \vdash A$, to say that A has a proof in which each formula either is deduced according the axioms or inference rule of $\mathcal{F}$ or is in Γ. In this case, we say that A is proved from the extra-logical hypotheses Γ; note that Γ may contain formulas which are not tautologies.

1.1.3. Soundness and completeness of $\mathcal{F}$. It is easy to prove that every $\mathcal{F}$-provable formula is a tautology, by noting that all axioms of $\mathcal{F}$ are valid and that modus ponens preserves the property of being valid. Similarly, whenever $\Gamma \vdash_{\mathcal{F}} A$, then Γ tautologically implies A. In other words, $\mathcal{F}$ is *(implicationally) sound*; which means that all provable formulas are valid (or, are consequences of the extra-logical hypotheses Γ).

Of course, any useful proof system ought to be sound, since the purpose of creating proofs is to establish the validity of a sentence. Remarkably, the system $\mathcal{F}$ is also *complete* in that it can prove any valid formula. Thus the semantic notion of validity and the syntactic notion of provability coincide, and a formula is valid if and only if it is provable in $\mathcal{F}$.

Theorem. *The propositional proof system $\mathcal{F}$ is complete and is implicationally complete; namely,*
(1) *If A is a tautology, then $\vdash_{\mathcal{F}} A$.*
(2) *If $\Gamma \vDash A$, then $\Gamma \vdash_{\mathcal{F}} A$.*

The philosophical significance of the completeness theorem is that a finite set of (schematic) axioms and rules of inference are sufficient to establish the validity of any tautology. In hindsight, it is not surprising that this holds, since the method of truth-tables already provides an algorithmic way of recognizing tautologies. Indeed, the proof of the completeness theorem given below, can be viewed as showing that the method of truth tables can be formalized within the system $\mathcal{F}$.

1.1.4. Proof. We first observe that part (2) of the completeness theorem can be reduced to part (1) by a two step process. Firstly, note that the compactness theorem for propositional logic states that if $\Gamma \vDash A$ then there is a finite subset Γ_0 of Γ which also tautologically implies A. A topological proof of the compactness theorem for propositional logic is sketched in 1.1.5 below. Thus Γ may, without loss of generality, be assumed to be a finite set of formulas, say $\Gamma = \{B_1, \ldots, B_k\}$. Secondly, note that $\Gamma \vDash A$ implies that $B_1 \supset B_2 \supset \cdots \supset A$ is a tautology. So, by part (1), the latter formula has an $\mathcal{F}$-proof, and by k additional modus ponens inferences, $\Gamma \vdash A$. (To simplify notation, we write $\vdash$ instead of $\vdash_{\mathcal{F}}$.)

It remains to prove part (1). We begin by establishing a series of special cases, (a)–(k), of the completeness theorem, in order to "bootstrap" the propositional system $\mathcal{F}$. We use symbols ϕ, ψ, χ for arbitrary formulas and Π to represent any set of formulas.

(a) $\vdash \phi \supset \phi$.
Proof: Combine the three axioms $(\phi \supset \phi \supset \phi) \supset (\phi \supset (\phi \supset \phi) \supset \phi) \supset (\phi \supset \phi)$, $\phi \supset (\phi \supset \phi)$ and $\phi \supset (\phi \supset \phi) \supset \phi$ with two uses of modus ponens.

(b) Deduction Theorem: $\Gamma, \phi \vdash \psi$ if and only if $\Gamma \vdash \phi \supset \psi$.
Proof: The reverse implication is trivial. To prove the forward implication, suppose $C_1, C_2, \ldots, C_k$ is an $\mathcal{F}$-proof of ψ from Γ, ϕ. This means that C_k is ψ and that each

C_i is ϕ, is in Γ, is an axiom, or is inferred by modus ponens. It is straightforward to prove, by induction on i, that $\Gamma \vdash \phi \supset C_i$ for each C_i.

(c) $\phi \supset \psi \vdash \neg\psi \supset \neg\phi$.
Proof: By the deduction theorem, it suffices to prove that $\phi \supset \psi, \neg\psi \vdash \neg\phi$. To prove this, use the two axioms $\neg\psi \supset (\phi \supset \neg\psi)$ and $(\phi \supset \psi) \supset (\phi \supset \neg\psi) \supset \neg\phi$ and three uses of modus ponens.

(d) $\phi, \neg\phi \vdash \psi$.
Proof: From the axiom $\phi \supset (\neg\psi \supset \phi)$, we have $\phi \vdash \neg\psi \supset \phi$. Thus, by (c) we get $\phi \vdash \neg\phi \supset \neg\neg\psi$, and by (b), $\phi, \neg\phi \vdash \neg\neg\psi$. Finally modus ponens with the axiom $\neg\neg\psi \supset \psi$ gives the desired result.

(e) $\neg\phi \vdash \phi \supset \psi$ and $\psi \vdash \phi \supset \psi$.
Proof: This former follows from (d) and the deduction theorem, and the latter follows from the axiom $\psi \supset (\phi \supset \psi)$.

(f) $\phi, \neg\psi \vdash \neg(\phi \supset \psi)$.
Proof: It suffices to prove $\phi \vdash \neg\psi \supset \neg(\phi \supset \psi)$. Thus, by (c) and the deduction theorem, it suffices to prove $\phi, \phi \supset \psi \vdash \psi$. The latter assertion is immediate from modus ponens.

(g) $\phi, \psi \vdash \phi \wedge \psi$.
Proof: Two uses of modus ponens with the axiom $\phi \supset \psi \supset (\phi \wedge \psi)$.

(h) $\neg\phi \vdash \neg(\phi \wedge \psi)$ and $\neg\psi \vdash \neg(\phi \wedge \psi)$.
Proof: For the first part, it suffices to show $\vdash \neg\phi \supset \neg(\phi \wedge \psi)$, and thus, by (c), it suffices to show $\vdash (\phi \wedge \psi) \supset \phi$, which is an axiom. The proof that $\neg\psi \vdash \neg(\phi \wedge \psi)$ is similar.

(i) $\phi \vdash \phi \vee \psi$ and $\psi \vdash \phi \vee \psi$.
Proof: $\phi \supset (\phi \vee \psi)$ and $\psi \supset (\phi \vee \psi)$ are axioms.

(j) $\neg\phi, \neg\psi \vdash \neg(\phi \vee \psi)$.
Proof: It suffices to prove $\neg\phi \vdash \neg\psi \supset \neg(\phi \vee \psi)$, and this, by (c), follows from $\neg\phi \vdash (\phi \vee \psi) \supset \psi$. For this, we combine

(*i*) $\neg\phi \vdash \phi \supset \psi$, by (e),
(*ii*) $\vdash \psi \supset \psi$, by (a),
(*iii*) $\vdash (\phi \supset \psi) \supset (\psi \supset \psi) \supset ((\phi \vee \psi) \supset \psi)$, an axiom,

with two uses of modus ponens.

(k) $\phi \vdash \neg\neg\phi$.
Proof: By (d), $\phi, \neg\phi \vdash \neg\neg\phi$, and obviously, $\phi, \neg\neg\phi \vdash \neg\neg\phi$. So $\phi \vdash \neg\neg\phi$ follows from the next lemma.

1.1.4.1. Lemma. *If* $\Gamma, \phi \vdash \psi$ *and* $\Gamma, \neg\phi \vdash \psi$, *then* $\Gamma \vdash \psi$.

Proof. By (b) and (c), the two hypotheses imply that $\Gamma \vdash \neg\psi \supset \neg\phi$ and $\Gamma \vdash \neg\psi \supset \neg\neg\phi$. These plus the two axioms $(\neg\psi \supset \neg\phi) \supset (\neg\psi \supset \neg\neg\phi) \supset \neg\neg\psi$ and $\neg\neg\psi \supset \psi$ give $\Gamma \vdash \psi$. □

1.1.4.2. Lemma. *Let the formula A involve only the propositional variables among $p_1, \ldots, p_n$. For $1 \leq i \leq n$, suppose that B_i is either p_i or $\neg p_i$. Then, either*

$$B_1, \ldots, B_n \vdash A \qquad \text{or} \qquad B_1, \ldots, B_n \vdash \neg A.$$

Proof. Define τ to be a truth assignment that makes each B_i true. By the soundness theorem, A (respectively, $\neg A$), can be proved from the hypotheses $B_1, \ldots, B_n$ only if $\overline{\tau}(A) = T$ (respectively $\overline{\tau}(A) = F$). Lemma 1.1.4.2 asserts that the converse holds too.

The lemma is proved by induction on the complexity of A. In the base case, A is just p_i: this case is trivial to prove since B_i is either p_i or $\neg p_i$. Now suppose A is a formula $A_1 \vee A_2$. If $\sigma(A) = T$, then we must have $\tau(A_i) = T$ for some $i \in \{1, 2\}$; the induction hypothesis implies that $B_1, \ldots, B_n \vdash A_i$ and thus, by (i) above, $B_1, \ldots, B_n \vdash A$. On the other hand, if $\tau(A) = F$, then $\tau(A_1) = \tau(A_2) = F$, so the induction hypothesis implies that $B_1, \ldots, B_n \vdash \neg A_i$ for both $i = 1$ and $i = 2$. From this, (j) implies that $B_1, \ldots, B_n \vdash \neg A$. The cases where A has outermost connective $\wedge$, $\supset$ or $\neg$ are proved similarly. □.

We are now ready to complete the proof of the Completeness Theorem 1.1.3. Suppose A is a tautology. We claim that Lemma 1.1.4.2 can be strengthened to have

$$B_1, \ldots, B_k \vdash A$$

where, as before each B_i is either p_i or $\neg p_i$, but now $0 \leq k \leq n$ is permitted. We prove this by induction on $k = n, n-1, \ldots, 1, 0$. For $k = n$, this is just Lemma 1.1.4.2. For the induction step, note that $B_1, \ldots, B_k \vdash A$ follows from $B_1, \ldots, B_k, p_{k+1} \vdash A$ and $B_1, \ldots, B_k, \neg p_{k+1} \vdash A$ by Lemma 1.1.4.1. When $k = 0$, we have that $\vdash A$, which proves the Completeness Theorem.
Q.E.D. Theorem 1.1.3

1.1.5. It still remains to prove the compactness theorem for propositional logic. This theorem states:

Compactness Theorem. *Let Γ be a set of propositional formulas.*
(1) *Γ is satisfiable if and only if every finite subset of Γ is satisfiable.*
(2) *$\Gamma \vDash A$ if and only if there is a finite subset Γ_0 of Γ such that $\Gamma_0 \vDash A$.*

Since $\Gamma \vDash A$ is equivalent to $\Gamma \cup \{\neg A\}$ being unsatisfiable, (2) is implied by (1). It is fairly easy to prove the compactness theorem directly, and most introductory books in mathematical logic present such a proof. Here, we shall instead, give a proof based on the Tychonoff theorem; obviously this connection to topology is the reason for the name 'compactness theorem.'

Proof. Let V be the set of propositional variables used in Γ; the sets Γ and V need not necessarily be countable. Let 2^V denote the set of truth assignments on V and endow 2^V with the product topology by viewing it as the product of $|V|$ copies of the two element space with the discrete topology. That is to say, the subbasis elements of 2^V are the sets $B_{p,i} = \{\tau : \tau(p) = i\}$ for $p \in V$ and $i \in \{T, F\}$. Note that these subbasis elements are both open and closed. Recall that the Tychonoff theorem states that an arbitrary product of compact spaces is compact; in particular, 2^V is compact. (See Munkres [1975] for background material on topology.)

For $\phi \in \Gamma$, define $D_\phi = \{\tau \in 2^V : \tau \vDash \phi\}$. Since ϕ only involves finitely many variables, each D_ϕ is both open and closed. Now Γ is satisfiable if and only if $\cap_{\phi\in\Gamma} D_\phi$ is non-empty. By the compactness of 2^V, the latter condition is equivalent to the sets $\cap_{\phi\in\Gamma_0} D_\phi$ being non-empty for all finite $\Gamma_0 \subset \Gamma$. This, in turn is equivalent to each finite subset Γ_0 of Γ being satisfiable. □

The compactness theorem for first-order logic is more difficult; a purely model-theoretic proof can be given with ultrafilters (see, e.g., Eklof [1977]). We include a proof-theoretic proof of the compactness theorem for first-order logic for countable languages in section 2.3.7 below.

1.1.6. Remarks. There are of course a large number of possible ways to give sound and complete proof systems for propositional logic. The particular proof system $\mathcal{F}$ used above is adapted from Kleene [1952]. A more detailed proof of the completeness theorem for $\mathcal{F}$ and for related systems can be found in the textbook of Mendelson [1987]. The system $\mathcal{F}$ is an example of a class of proof systems called *Frege proof systems*: a Frege proof system is any proof system in which all axioms and rules are schematic and which is implicationally sound and implicationally complete. Most of the commonly used proof systems similar to $\mathcal{F}$ are based on modus ponens as the only rule of inference; however, some (non-Frege) systems also incorporate a version of the deduction theorem as a rule of inference. In these systems, if B has been inferred from A, then the formula $A \supset B$ may also be inferred. An example of such a system is the propositional fragment of the natural deduction proof system described in section 2.4.8 below.

Other rules of inference that are commonly allowed in propositional proof systems include the *substitution rule* which allows any instance of ϕ to be inferred from ϕ, and the *extension rule* which permits the introduction of abbreviations for long formulas. These two systems *appear* to be more powerful than Frege systems in that they seem to allow substantially shorter proofs of certain tautologies. However, whether they actually are significantly more powerful than Frege systems is an open problem. This issues are discussed more fully by Pudlák in Chapter VIII.

There are several currently active areas of research in the proof theory of propositional logic. Of course, the central open problem is the P versus NP question of whether there exists a polynomial time method of recognizing tautologies. Research on the proof theory of propositional logic can be, roughly speaking, separated into three problem areas. Firstly, the problem of "proof-search" is the question of

what are the best algorithmic methods for searching for propositional proofs. The proof-search problem is important for artificial intelligence, for automated theorem proving and for logic programming. The most common propositional proof systems used for proof-search algorithms are variations of the resolution system discussed in 1.3 below. A second, related research area is the question of proof lengths. In this area, the central questions concern the minimum lengths of proofs needed for tautologies in particular proof systems. This topic is treated in more depth in Chapter VIII in this volume.

A third research area concerns the investigation of fragments of the propositional proof system $\mathcal{F}$. For example, propositional intuitionist logic is the logic which is axiomatized by the system $\mathcal{F}$ without the axiom scheme $\neg\neg A \supset A$. Another important example is linear logic. Brief discussions of these two logics can be found in section 3.

1.2. The propositional sequent calculus

The sequent calculus, first introduced by Gentzen [1935] as an extension of his earlier natural deduction proof systems, is arguably the most elegant and flexible system for writing proofs. In this section, the propositional sequent calculus for classical logic is developed; the extension to first-order logic is treated in 2.3 below.

1.2.1. Sequents and Cedents. In the Hilbert-style systems, each line in a proof is a formula; however, in sequent calculus proofs, each line in a proof is a *sequent*: a sequent is written in the form

$$A_1, \ldots, A_k \longrightarrow B_1, \ldots, B_\ell$$

where the symbol $\longrightarrow$ is a new symbol called the sequent arrow (not to be confused with the implication symbol $\supset$) and where each A_i and B_j is a formula. The intuitive meaning of the sequent is that the conjunction of the A_i's implies the disjunction of the B_j's. Thus, a sequent is equivalent in meaning to the formula

$$\bigwedge_{i=1}^{k} A_i \supset \bigvee_{j=1}^{\ell} B_j.$$

The symbols $\bigwedge$ and $\bigvee$ represent conjunctions and disjunctions, respectively, of multiple formulas. We adopt the convention that an empty conjunction (say, when $k = 0$ above) has value "True", and that an empty disjunction (say, when $\ell = 0$ above) has value "False". Thus the sequent $\longrightarrow A$ has the same meaning as the formula A, and the *empty sequent* $\longrightarrow$ is false. A sequent is defined to be valid or a tautology if and only if its corresponding formula is.

The sequence of formulas $A_1, \ldots, A_k$ is called the *antecedent* of the sequent displayed above; $B_1, \ldots, B_\ell$ is called its *succedent* . They are both referred to as *cedents*.