Chapter I

INTRODUCTION

This book studies logical systems which use restricted reasoning based on concepts from computational complexity. The complexity classes of interest lie mainly between the basic class $\boldsymbol{AC}^0$ (whose members are computed by polynomial-size families of bounded-depth circuits), and the polynomial hierarchy $\boldsymbol{PH}$, and include the sequence

$$\boldsymbol{AC}^0 \subset \boldsymbol{AC}^0(m) \subseteq \boldsymbol{TC}^0 \subseteq \boldsymbol{NC}^1 \subseteq \boldsymbol{L} \subseteq \boldsymbol{NL} \subseteq \boldsymbol{P} \subseteq \boldsymbol{PH} \tag{1}$$

where $\boldsymbol{P}$ is polynomial time. (See the Appendix for definitions.)

We associate with each of these classes a logical theory and a proof system for the (quantified) propositional calculus. The proof system can be considered a nonuniform version of the universal (or sometimes the bounded) fragment of the theory. The functions definable in the logical theory are those associated with the complexity class, and (in some cases) the lines in a polynomial size proof in the propositional system express concepts in the complexity class. Universal (or bounded) theorems of the logical theory translate into families of valid formulas with polynomial size proofs in the corresponding proof system. The logical theory proves the soundness of the proof system.

Conceptually the theory $\boldsymbol{VC}$ associated with a complexity class $\mathbf{C}$ can prove a given mathematical theorem if the induction hypotheses needed in the proof can be formulated using concepts from $\mathbf{C}$. We are interested in trying to find the weakest class $\mathbf{C}$ needed to prove various theorems of interest in computer science.

Here are some examples of the three-way association among complexity classes, theories, and proof systems:

class	$\boldsymbol{AC}^0$	$\boldsymbol{TC}^0$	$\boldsymbol{NC}^1$	$\boldsymbol{P}$	$\boldsymbol{PH}$	
theory	$\boldsymbol{V}^0$	$\boldsymbol{VTC}^0$	$\boldsymbol{VNC}^1$	$\boldsymbol{VP}$	$\boldsymbol{V}^\infty$	(2)
system	$\boldsymbol{AC}^0$-*Frege*	$\boldsymbol{TC}^0$-*Frege*	*Frege*	*eFrege*	$\langle \boldsymbol{G}_i \rangle$.	

Consider for example the class $\boldsymbol{NC}^1$. The uniform version is $\boldsymbol{ALogTime}$, the class of problems solvable by an alternating Turing machine in time $\mathcal{O}(\log n)$. The definable functions in the associated theory $\boldsymbol{VNC}^1$ are the $\boldsymbol{NC}^1$ functions, i.e., those functions whose bit graphs are $\boldsymbol{NC}^1$ relations.

A problem in nonuniform $\mathbf{NC}^1$ is defined by a polynomial-size family of log-depth Boolean circuits, or equivalently a polynomial-size family of propositional formulas. The corresponding propositional proof systems are called ***Frege*** systems, and are described in standard logic textbooks: a ***Frege*** proof of a tautology A consists of a sequence of propositional formulas ending in A, where each formula is either an axiom or follows from earlier formulas by a rule of inference. Universal theorems of $\boldsymbol{VNC}^1$ translate into polynomial-size families of ***Frege*** proofs. Finally $\boldsymbol{VNC}^1$ proves the soundness of ***Frege*** systems, and any proof system whose soundness is provable in $\boldsymbol{VNC}^1$ can be p-simulated by a ***Frege*** system (Theorem X.3.11).

The famous open question in complexity theory is whether the conjecture that $\mathbf{P}$ is a proper subset of $\mathbf{NP}$ is in fact true (we know $\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PH}$). If $\mathbf{P} = \mathbf{NP}$ then the polynomial hierarchy $\mathbf{PH}$ collapses to $\mathbf{P}$, but it is possible that $\mathbf{PH}$ collapses only to $\mathbf{NP}$ and still $\mathbf{P} \neq \mathbf{NP}$. What may be less well known is that not only is it possible that $\mathbf{PH} = \mathbf{P}$, but it is consistent with our present knowledge that $\mathbf{PH} = \mathbf{AC}^0(6)$, so that all classes in (1) might be equal except for $\mathbf{AC}^0$ and $\mathbf{AC}^0(p)$ for p prime. This is one motivation for studying the theories associated with these complexity classes, since it ought to be easier to separate the theories corresponding to the complexity classes than to separate the classes themselves (but so far the theories in (2) have not been separated, except for $\boldsymbol{V}^0$).

A common example used to illustrate the complexity of the concepts needed to prove a theorem is the Pigeonhole Principle (PHP). Our version states that if $n+1$ pigeons are placed in n holes, then some hole has two or more pigeons. We can present an instance of the PHP using a Boolean array $\langle P(i,j)\rangle$ $(0 \leq i \leq n, 0 \leq j < n)$, where $P(i,j)$ asserts that pigeon i is placed in hole j. Then the PHP can be formulated in the theory $\boldsymbol{V}^0$ by the formula

$$\forall i \leq n\, \exists j < n\, P(i,j) \supset \exists i_1, i_2 \leq n\, \exists j < n\, (i_1 \neq i_2 \wedge P(i_1,j) \wedge P(i_2,j)). \tag{3}$$

Ajtai [5] proved (in effect) that this formula is not a theorem of $\boldsymbol{V}^0$, and also that the propositional version (which uses atoms p_{ij} to represent $P(i,j)$ and finite conjunctions and disjunctions to express the bounded universal and existential number quantifiers) does not have polynomial size $\mathbf{AC}^0$-***Frege*** proofs. The intuitive reason for this is that a counting argument seems to be required to prove the PHP, but the complexity class $\mathbf{AC}^0$ cannot count the number of ones in a string of bits. On the other hand, the class $\mathbf{NC}^1$ can count, and indeed Buss proved that the propositional PHP does have polynomial size ***Frege*** proofs, and his method shows that (3) is a theorem of the theory $\boldsymbol{VNC}^1$. (In fact it is a theorem of the apparently weaker theory $\boldsymbol{VTC}^0$.)

A second example comes from linear algebra. If A and B are $n \times n$ matrices over some field, then

$$AB = I \supset BA = I. \tag{4}$$

A standard proof of this uses Gaussian elimination, which is a polynomial-time process. Indeed Soltys showed that (4) is a theorem of the theory ***VP*** corresponding to polynomial-time reasoning, and it follows that its propositional translation (say over the field of two elements) has polynomial-size proofs in the corresponding proof system ***eFrege***. It is an open question whether (4) over ***GF***(***2***) (or any field) can be proved in $\boldsymbol{VNC}^1$, or whether the propositional version has polynomial-size ***Frege*** proofs.

The preceding example (4) is a universal theorem, in the sense that its statement has no existential quantifier. Another class of examples comes from existential theorems. From linear algebra, a natural example about $n \times n$ matrices is

$$\forall A \exists B \neq 0 (AB = I \vee AB = 0). \tag{5}$$

The complexity of finding B for a given A, even over ***GF***(***2***), is thought not to be in $\boldsymbol{NC}^1$ (it is hard for log space). Assuming that this is the case, it follows that (5) is not a theorem of $\boldsymbol{VNC}^1$, since only $\boldsymbol{NC}^1$ functions are definable in that theory. This conclusion is the result of a general witnessing theorem, which states that if the formula $\forall x \exists y \varphi(x, y)$ (for suitable formulas φ) is provable in the theory associated with complexity class ***C***, then there is a Skolem function $f(x)$ whose complexity is in ***C*** and which satisfies $\forall x \varphi(x, f(x))$.

The theory $\boldsymbol{VNC}^1$ proves that (4) follows from (5). Both (4) and (5) are theorems of the theory ***VP*** associated with polynomial time.

Another example of an existential theorem is “Fermat’s Little Theorem”, which states that if n is a prime number and $1 \leq a < n$, then $a^{n-1} \equiv 1 \pmod{n}$. Its existential content is captured by its contrapositive form

$$(1 \leq a < n) \wedge (a^{n-1} \not\equiv 1 \pmod{n}) \supset \exists d (1 < d < n \wedge d | n). \tag{6}$$

It is not hard to see that the function $a^{n-1} \bmod n$ can be computed in time polynomial in the lengths of a and n, using repeated squaring. If (6) is provable in ***VP***, then by the witnessing theorem mentioned above it would follow that there is a polynomial time function $f(a, n)$ whose value $d = f(a, n)$ provides a proper divisor of n whenever a, n satisfy the hypothesis in (6). With the exception of the so-called Carmichael numbers, which can be factored in polynomial time, every composite n satisfies the hypothesis of (6) for at least half of the values of a, $1 \leq a < n$. Hence $f(a, n)$ would provide a probabilistic polynomial time algorithm for integer factoring. Such an algorithm is not known to exist, and would provide a method for breaking the RSA public-key encryption scheme.

Thus Fermat's Little Theorem is not provable in $\boldsymbol{VP}$, assuming that there is no probabilistic polynomial time factoring algorithm.

Propositional tautologies can be used to express universal theorems such as (3) (in which the Predicate P is implicitly universally quantified and the bounded number quantifiers can be expanded in translation) and (4), but are not well suited to express existential theorems such as (5) and (6). However the latter can be expressed using formulas in the quantified propositional calculus (QPC), which extends the propositional calculus by allowing quantifiers $\forall p$ and $\exists p$ over propositional variables p. Each of the complexity classes in (2) has an associated QPC system, and in fact the systems $\langle \boldsymbol{G}_i \rangle$ mentioned for $\boldsymbol{PH}$ form a hierarchy of QPC systems.

Most of the theories presented in this book, including those in (2), have the same "second-order" underlying vocabulary $\mathcal{L}_A^2$, introduced by Zambella. The vocabulary $\mathcal{L}_A^2$ is actually a vocabulary for the two-sorted first-order predicate calculus, where one sort is for numbers in $\mathbb{N}$ and the second sort is for finite sets of numbers. Here we regard an object of the second sort as a finite string over the alphabet $\{0, 1\}$ (the i-th bit in the string is 1 iff i is in the set). The strings are the objects of interest for the complexity classes, and serve as the main inputs for the machines or circuits that determine the class. The numbers serve a useful purpose as indices for the strings when describing properties of the strings. When they are used as machine or circuit inputs, they are presented in unary notation.

In the more common single-sorted theories such as Buss's hierarchies $\boldsymbol{S}_2^i$ and $\boldsymbol{T}_2^i$ the underlying objects are numbers which are presented in binary notation as inputs to Turing machines. Our two-sorted treatment has the advantage that the underlying vocabulary has no primitive operations on strings except the length function $|X|$ and the bit predicate $X(i)$ (meaning $i \in X$). This is especially important for studying weak complexity classes such as $\boldsymbol{AC}^0$. The standard vocabulary for single-sorted theories includes number multiplication, which is not an $\boldsymbol{AC}^0$ function on binary strings.

Chapter II provides a sufficient background in first-order logic for the rest of the book, including Gentzen's proof system $\boldsymbol{LK}$. An unusual feature is our treatment of anchored (or "free-cut-free") $\boldsymbol{LK}$-proofs. The completeness of these restricted systems is proved directly by a simple term-model construction as opposed to the usual syntactic cut-elimination method. The second form of the Herbrand Theorem proved here has many applications in later chapters for witnessing theorems.

Chapter III presents the necessary background on Peano Arithmetic (the first-order theory of $\mathbb{N}$ under $+$ and $\times$) and its subsystems, including the bounded theory $\boldsymbol{I}\Delta_0$. The functions definable in $\boldsymbol{I}\Delta_0$ are precisely those in the complexity class known as $\boldsymbol{LTH}$ (the Linear Time Hierarchy). An important theorem needed for this result is that the predicate $y = 2^x$ is definable in the vocabulary of arithmetic using a bounded formula

(Section III.3.3). The universal theory $\overline{\boldsymbol{I}\Delta}_0$ has function symbols for each function in the Linear Time Hierarchy, and forms a conservative extension of $\boldsymbol{I}\Delta_0$. This theory serves as a prototype for universal theories defined in later chapters for other complexity classes.

Chapter IV introduces the syntax and intended semantics for the two-sorted theories, which will be used throughout the remaining chapters. Here Σ_0^B is defined to be the class of formulas with no string quantifiers, and with all number quantifiers bounded. The Σ_1^B-formulas begin with zero or more bounded existential string quantifiers followed by a Σ_0^B-formula, and more generally Σ_i^B-formulas begin with at most i alternating blocks of bounded string quantifiers $\exists\forall\exists\dots$. Representation theorems are proved which state that formulas in the syntactic class Σ_0^B represent precisely the (two-sorted) $\boldsymbol{AC}^0$ relations, and for $i \geq 1$, formulas in Σ_i^B represent the relations in the i-th level of the polynomial hierarchy.

Chapter V introduces the hierarchy of two-sorted theories $\boldsymbol{V}^0 \subset \boldsymbol{V}^1 \subseteq \boldsymbol{V}^2 \subseteq \cdots$. For $i \geq 1$, $\boldsymbol{V}^i$ is the two-sorted version of Buss's single-sorted theory $\boldsymbol{S}_2^i$, which is associated with the ith level of the polynomial hierarchy. In this chapter we concentrate on $\boldsymbol{V}^0$, which is associated with the complexity class $\boldsymbol{AC}^0$. All two-sorted theories considered in later chapters are extensions of $\boldsymbol{V}^0$. A Buss-style witnessing theorem is proved for $\boldsymbol{V}^0$, showing that the existential string quantifiers in a Σ_1^B-theorem of $\boldsymbol{V}^0$ can be witnessed by $\boldsymbol{AC}^0$-functions. Since Σ_1^B-formulas have all string quantifiers in front, both the statement and the proof of the theorem are simpler than for the usual Buss-style witnessing theorems. (The same applies to the witnessing theorems proved in later chapters.) The final section proves that $\boldsymbol{V}^0$ is finitely axiomatizable.

Chapter VI concentrates on the theory $\boldsymbol{V}^1$, which is associated with the complexity class $\boldsymbol{P}$. All (and only) polynomial time functions are Σ_1^B-definable in $\boldsymbol{V}^1$. The positive direction is shown in two ways: by analyzing Turing machine computations and by using Cobham's characterization of these functions. The witnessing theorem for $\boldsymbol{V}^1$ is shown using (two-sorted versions of) the anchored proofs described in Chapter II, and implies that only polynomial time functions are Σ_1^B-definable in $\boldsymbol{V}^1$.

Chapter VII gives a general definition of propositional proof system. The goal is to associate a proof system with each theory so that each Σ_0^B-theorem of the theory translates into a polynomial size family of proofs in the proof system. Further, the theory should prove the soundness of the proof system, but this is not shown until Chapter X. In Chapter VII, translations are defined from $\boldsymbol{V}^0$ to bounded-depth $\boldsymbol{PK}$-proofs (i.e. bounded-depth Frege proofs), and also from $\boldsymbol{V}^1$ to extended Frege proofs. Systems $\boldsymbol{G}_i$ and $\boldsymbol{G}_i^\star$ for the quantified propositional calculus are defined, and for $i \geq 1$ we show how to translate bounded theorems of $\boldsymbol{V}^i$

to polynomial size families of proofs in the system $\boldsymbol{G}_i^\star$. The two-sorted treatment makes these translations simple and natural.

Chapter VIII begins by introducing other two-sorted theories associated with polynomial time. The finitely axiomatized theory $\boldsymbol{VP}$ and its universal conservative extension $\boldsymbol{VPV}$ both appear to be weaker than $\boldsymbol{V}^1$, although they have the same Σ_1^B theorems as $\boldsymbol{V}^1$. $\boldsymbol{VP} = \boldsymbol{TV}^0$ is the base of the hierarchy of theories $\boldsymbol{TV}^0 \subseteq \boldsymbol{TV}^1 \subseteq \cdots$, where for $i \geq 1$, $\boldsymbol{TV}^i$ is isomorphic to Buss's single-sorted theory $\boldsymbol{T}_2^i$. The definable problems in $\boldsymbol{TV}^1$ have the complexity of Polynomial Local Search. A form of the Herbrand Theorem known as KPT Witnessing is proved and applied to show independence of the Replacement axiom scheme from some theories, and to relating the collapse of the $\boldsymbol{V}^\infty$ hierarchy with the provable collapse of the polynomial hierarchy. The Σ_j^B-definable search problems in $\boldsymbol{V}^i$ and $\boldsymbol{TV}^i$ are characterized for many i and j. The RSUV isomorphism theorem between $\boldsymbol{S}_2^i$ and $\boldsymbol{V}^i$ is proved.

See Table 3 on page 250 for a summary of which search problems are definable in $\boldsymbol{V}^i$ and $\boldsymbol{TV}^i$.

Chapter IX gives a uniform way of introducing minimal canonical theories for many complexity classes between $\boldsymbol{AC}^0$ and $\boldsymbol{P}$, including those mentioned earlier in (1). Each finitely axiomatized theory is defined as an extension of $\boldsymbol{V}^0$ obtained by adding a single axiom stating the existence of a computation solving a complete problem for the associated complexity class. Evidence for the "minimality" of each theory is presented by defining a universal theory whose axioms are simply a set of basic axioms for $\boldsymbol{V}^0$ together with the defining axioms for all the functions in the associated complexity class. These functions are defined as the function $\boldsymbol{AC}^0$-closure of the complexity class, or (as is the case for $\boldsymbol{P}$) using a recursion-theoretic characterization of the function class. The main theorem in each case is that the universal theory is a conservative extension of the finitely axiomatized theory.

Table 1 on page 7 gives a summary of the two-sorted theories presented in Chapter IX and elsewhere, and Table 2 on page 8 gives a list of some theorems provable (or possibly not provable) in the various theories.

Chapter X extends Chapter VII by presenting quantified propositional proof systems associated with various complexity classes, and defining translations from the bounded theorems of the theories introduced in Chapter IX to the appropriate proof system. Witnessing theorems for subsystems of $\boldsymbol{G}$ (quantified propositional calculus) are proved. The notion of *reflection principle* (soundness of a proof system) is defined, and many results showing which kinds of reflection principle for various systems can (or probably cannot) be proved in various theories. It is shown how reflection principles can be used to axiomatize some of the theories.

CLASS	THEORY	SEE
$\mathbf{AC}^0$	$\mathbf{V}^0$	Section V.1
	$\overline{\mathbf{V}}^0$	Section V.6
$\mathbf{AC}^0(2)$	$\mathbf{V}^0(2), \widehat{\mathbf{V}^0(2)}, \overline{\mathbf{V}^0(2)}$	Section IX.4.2
	$\mathbf{VAC}^0(2)\mathbf{V}$	Section IX.4.4
$\mathbf{AC}^0(m)$	$\mathbf{V}^0(m), \widehat{\mathbf{V}^0(m)}, \overline{\mathbf{V}^0(m)}$	Section IX.4.6
$\mathbf{AC}^0(6)$	$\mathbf{VAC}^0(6)\mathbf{V}$	Section IX.4.8
$\mathbf{ACC}$	$\mathbf{VACC}$	Section IX.4.6
$\mathbf{TC}^0$	$\mathbf{VTC}^0, \widehat{\mathbf{VTC}}^0, \overline{\mathbf{VTC}}^0$	Section IX.3.2
	$\mathbf{VTC}^0\mathbf{V}$	Section IX.3.4
$\mathbf{NC}^1$	$\mathbf{VNC}^1, \widehat{\mathbf{VNC}}^1, \overline{\mathbf{VNC}}^1$	Section IX.5.3
	$\mathbf{VNC}^1\mathbf{V}$	Section IX.5.5
$\mathbf{L}$	$\mathbf{VL}, \widehat{\mathbf{VL}}, \overline{\mathbf{VL}}$	Section IX.6.3
	$\mathbf{VLV}$	Section IX.6.4
$\mathbf{NL}$	$\mathbf{VNL}, \widehat{\mathbf{VNL}}, \overline{\mathbf{VNL}}$	Section IX.6.1
	$\mathbf{V}^1$-$\mathbf{KROM}$	Section IX.6.2
$\mathbf{AC}^k$ $(k \geq 1)$	$\mathbf{VAC}^k$	Section IX.5.6
$\mathbf{NC}^{k+1}$ $(k \geq 1)$	$\mathbf{VNC}^{k+1}$	Section IX.5.6
$\mathbf{NC}$	$\mathbf{VNC}$	Section IX.5.6
	$\mathbf{U}^1$	Section IX.5.6
$\mathbf{P}$	$\mathbf{VP}$	Section VIII.1
	$\mathbf{VPV}$	Section VIII.2
	$\mathbf{TV}^0$	Section VIII.3
	$\mathbf{V}^1$-$\mathbf{HORN}$	Section VIII.4
	$\mathbf{V}^1$	Chapter VI
$\mathbf{C}$ (for $\mathbf{C} \subseteq \mathbf{P}$)	$\mathbf{VC}, \widehat{\mathbf{VC}}, \overline{\mathbf{VC}}$	Section IX.2.1
$\mathbf{CC}(\mathbf{PLS})$	$\mathbf{TV}^1$	Section VIII.5
	$\mathbf{V}^2$	Section VIII.7.2

TABLE 1. Theories and their Σ_1^B-definable classes.

THEORY	(NON)THEOREM(?)	SEE
$\boldsymbol{V^0}$	(seq.) Jordan Curve Theorem	[84]
	$\nvdash$ ***PHP***	Corollary VII.2.4
	$\nvdash$ onto ***PHP***, $\nvdash$ ***Count***$_m$	Section IX.4.3
$\boldsymbol{V^0}(2)$	onto ***PHP***, ***Count***$_2$	Section IX.4.3
	(set) Jordan Curve Theorem	Section IX.4.5
	PHP?, ***Count***$_3$?	Section IX.7.4
$\boldsymbol{V^0}(m)$	***Count***$_{m'}$ (if $\gcd(m, m') > 1$)	Section IX.4.7
	Count$_{m'}$? (if $\gcd(m, m') = 1$)	Section IX.7.4
	PHP?	Section IX.7.4
$\boldsymbol{VTC^0}$	sorting	Exercise IX.3.9
	Reflection Principles for d-***PTK***	Section X.4.2
	PHP	Section IX.3.5
	Finite Szpilrajn's Theorem	Section IX.3.7
	Bondy's Theorem	Section IX.3.8
	define $\lfloor X/Y \rfloor$?	Section IX.7.3
$\boldsymbol{VNC^1}$	Reflection Principle for ***PK***	Theorem X.3.9
	Barrington's Theorem	Sec. IX.5.5 & [82]
	NUMONES	Section IX.5.4
VL	Lind's characterization of ***L***	Section IX.6.4
	Reingold's Theorem?	Section IX.7.2
VNL	Grädel's Theorem (for ***NL***)	Theorem IX.6.24
$\boldsymbol{VNC^2}$	Cayley–Hamilton Theorem?	Section IX.7.1
$\boldsymbol{VP} = \boldsymbol{TV^0}$	Reflection Principle for ***ePK***	Exercise X.2.22
	Grädel's Theorem (for ***P***)	Theorem VIII.4.8
	$\nvdash$ Fermat's Little Theorem (cond.)	page 3
$\boldsymbol{V^1}$	Prime Factorization Theorem	Exercise VI.4.4
$\boldsymbol{V^i}$ $(i \geq 1)$	Π^q_i-$RFN_{\boldsymbol{G}_{i-1}}$, Π^q_{i+2}-$RFN_{\boldsymbol{G}^\star_i}$	Theorem X.2.17
$\boldsymbol{TV^i}$ $(i \geq 0)$	Π^q_{i+2}-$RFN_{\boldsymbol{G}^\star_{i+1}}$, Π^q_{i+1}-$RFN_{\boldsymbol{G}_i}$	Theorem X.2.20

TABLE 2. Some theories and their (non)theorems/solvable problems (and open questions). ("cond." stands for conditional.) Many theorems of ***VP***, such as Kuratowski's Theorem, Hall's Theorem, Menger's Theorem are not discussed here.

Chapter II

THE PREDICATE CALCULUS AND THE SYSTEM *LK*

In this chapter we present the logical foundations for theories of bounded arithmetic. We introduce Gentzen's proof system ***LK*** for the predicate calculus, and prove that it is sound, and complete even when proofs have a restricted form called "anchored". We augment the system ***LK*** by adding equality axioms. We prove the Compactness Theorem for predicate calculus, and the Herbrand Theorem.

In general we distinguish between syntactic notions and semantic notions. Examples of syntactic notions are variables, connectives, formulas, and formal proofs. The semantic notions relate to meaning; for example truth assignments, structures, validity, and logical consequence.

The first section treats the simple case of propositional calculus.

II.1. Propositional Calculus

Propositional formulas (called simply *formulas* in this section) are built from the logical constants $\bot$, $\top$ (for False, True), propositional variables (or atoms) $P_1, P_2, \dots$, connectives $\neg, \vee, \wedge$, and parentheses (,). We use $P, Q, R, \dots$ to stand for propositional variables, $A, B, C, \dots$ to stand for formulas, and $\Phi, \Psi, \dots$ to stand for sets of formulas. When writing formulas such as $(P \vee (Q \wedge R))$, our convention is that $P, Q, R, \dots$ stand for distinct variables.

Formulas are built according to the following rules:

- $\bot$, $\top$, P, are formulas (also called *atomic formulas*) for any variable P.
- If A and B are formulas, then so are $(A \vee B)$, $(A \wedge B)$, and $\neg A$.

The implication connective $\supset$ is not allowed in our formulas, but we will take $(A \supset B)$ to stand for $(\neg A \vee B)$. Also $(A \leftrightarrow B)$ stands for $((A \supset B) \wedge (B \supset A))$.

We sometimes abbreviate formulas by omitting parentheses, but the intended formula has all parentheses present as defined above.

A *truth assignment* is an assignment of truth values F, T to atoms. Given a truth assignment τ, the truth value A^τ of a formula A is defined inductively as follows: $\bot^\tau = F$, $\top^\tau = T$, $P^\tau = \tau(P)$ for atom P, $(A \wedge B)^\tau = T$ iff both $A^\tau = T$ and $B^\tau = T$, $(A \vee B)^\tau = T$ iff either $A^\tau = T$ or $B^\tau = T$, $(\neg A)^\tau = T$ iff $A^\tau = F$.

DEFINITION II.1.1. A truth assignment τ *satisfies* A iff $A^\tau = T$; τ *satisfies* a set Φ of formulas iff τ satisfies A for all $A \in \Phi$. Φ is *satisfiable* iff some τ satisfies Φ; otherwise Φ is *unsatisfiable*. Similarly for A. $\Phi \models A$ (i.e., A is a *logical consequence* of Φ) iff τ satisfies A for every τ such that τ satisfies Φ. A formula A is *valid* iff $\models A$ (i.e., $A^\tau = T$ for all τ). A valid propositional formula is called a *tautology*. We say that A and B are *equivalent* (written $A \Longleftrightarrow B$) iff $A \models B$ and $B \models A$.

Note that $\Longleftrightarrow$ refers to semantic equivalence, as opposed to $=_{\text{syn}}$, which indicates syntactic equivalence. For example, $(P \vee Q) \Longleftrightarrow (Q \vee P)$, but $(P \vee Q) \neq_{syn} (Q \vee P)$.

II.1.1. Gentzen's Propositional Proof System *PK*. We present the propositional part ***PK*** of Gentzen's sequent-based proof system ***LK***. Each line in a proof in the system ***PK*** is a *sequent* of the form

$$A_1, \dots, A_k \longrightarrow B_1, \dots, B_\ell \tag{7}$$

where $\longrightarrow$ is a new symbol and $A_1, \dots, A_k$ and $B_1, \dots, B_\ell$ are sequences of formulas $(k, \ell \geq 0)$ called *cedents*. We call the cedent $A_1, \dots, A_k$ the *antecedent* and $B_1, \dots, B_\ell$ the *succedent* (or *consequent*).

The semantics of sequents is given as follows. We say that a truth assignment τ *satisfies* the sequent (7) iff either τ falsifies some A_i or τ satisfies some B_i. Thus the sequent is equivalent to the formula

$$\neg A_1 \vee \neg A_2 \vee \cdots \vee \neg A_k \vee B_1 \vee B_2 \vee \cdots \vee B_\ell. \tag{8}$$

(Here and elsewhere, a disjunction $C_1 \vee \cdots \vee C_n$ indicates parentheses have been inserted with association to the right. For example, $C_1 \vee C_2 \vee C_3 \vee C_4$ stands for $(C_1 \vee (C_2 \vee (C_3 \vee C_4)))$. Similarly for a disjunction $C_1 \wedge \cdots \wedge C_n$.) In other words, the conjunction of the A's implies the disjunction of the B's. In the cases in which the antecedent or succedent is empty, we see that the sequent $\longrightarrow A$ is equivalent to the formula A, and $A \longrightarrow$ is equivalent to $\neg A$, and just $\longrightarrow$ (with both antecedent and succedent empty) is false (unsatisfiable). We say that a sequent is *valid* if it is true under all truth assignments (which is the same as saying that its corresponding formula is a tautology).

DEFINITION II.1.2. A ***PK*** *proof* of a sequent S is a finite tree whose nodes are (labeled with) sequents, whose root (called the *endsequent*) is S and is written at the bottom, whose leaves (or *initial sequents*) are logical axioms (see below), such that each non-leaf sequent follows from the sequent(s) immediately above by one of the rules of inference given below.