From Compactness To Completeness

or how to make Propositional Logic, First-Order Logic, and other logics fun and easier to learn

Assaf Kfoury

30 June 2020

Contents

Preface

Though my emphases are a little different in these lecture notes, I follow a trend tried by a few others in recent years of introducing and proving the Compactness Theorem before the Completeness Theorem. Doing it this way, Completeness becomes a consequence of Compactness. The other way around, which is standard in many textbooks, invokes Completeness (as well as Soundness) to prove Compactness .[1]

There are good reasons for reversing the traditional approach. Perhaps the chief reason is to avoid getting immersed in the fiddly details of formal-proof systems and, thus, to also avoid the concern of dealing with as many proofs of Completeness as there are proof systems (a welcome avoidance when part of our study involves different logics and their proof systems).

From my own teaching experience, many students of computer science, and some of the very best, are not inspired by the relatively large amount of syntactic details they have to absorb in the traditional approach before reaching applications related to their own interests. Formal proofs are far less permissive of syntactic imprecision than semi-formal or informal proofs, and it takes more than one semester to learn and appreciate the benefits of the former over the latter, especially when the latter can be very partially spelled out with no loss of rigor (often the case). Put differently, the traditional approach would force students to learn a good deal of *proof theory* and its unsparing precision on syntax earlier than later, before reaching more interesting applications based on as much of *model theory* as time permits towards the end of the semester. Starting with Compactness, we reverse this order, emphasize more model theory and semantic notions early on, and can present interesting (though still small) applications much earlier in the semester, with a distinctly algebraic or semantic flavor and a smaller amount of definitions related to syntax and formal derivations.

But there are other advantages to starting with Compactness. It makes it easier to grasp topological aspects of the notion (and the origin of its name). We can go deep in a topological direction, by explaining Compactness purely in terms of notions such as the *finite-intersection property*, *ultrafilters*, *ultraproducts*, and others, but that would take us further afield from the focus on formal logic. I choose a watered-down approach. In the proof of Compactness in Section 1, we construct a maximal satisfiable set of propositional wff's without any explicit reference to topological notions, although these are lurking right under the surface. I delay making explicit connections to topology until Appendix F.

The simplest logic we consider in these notes is *propositional logic*, and the most expressive is *first-order logic*. Also unusual is the gradual transition from the former to the latter, as five intermediate logics are introduced: the *logic of quantified Boolean formulas* (in Section 2), *equality logic* (in Section 3), what I call *zeroth-order logic* (in Section 4), and *equational logic* and *quasi-equational logic* (in Section 5). This is not a linear progression in expressive power from *propositional logic* to *first-order logic*, as some of our intermediate logics are extended by unrelated features (*i.e.*, incomparable in their standard interpretation); it is more a progression in the difficulty of studying them.

We reduce Compactness for *first-order logic*, and for every intermediate logic, to Compactness for *propositional logic*. Thus, there is only one proof of Compactness in these notes, that of Compactness for *propositional logic*, from which Compactness for all the other logics follow – this is the common thread holding them all together.[2]

[1] A typical example is the proof of the Compactness Theorem in the textbook *A Mathematical Introduction to Logic* by H.B. Enderton [4]; the proof at the end of Section 2.5 invokes the Completeness Theorem, as well as the Soundness Theorem, to prove Compactness. Another textbook, common in computer science departments, is *Logic in Computer Science* by M. Huth and M. Ryan [5], which omits altogether proofs for Soundness and Completeness (on page 96), and then invokes Completeness to prove Compactness as a consequence (on page 137).

[2] I do not claim originality for this approach. You will find it in some books from the 1960's and perhaps earlier, though it did not

Which is again different from the traditional approach. Some textbooks include proofs of Completeness twice, and then Compactness as a corollary via Soundness twice too: first for *propositional logic*, and again for *first-order logic*, on the grounds that the proof of Completeness for the latter is markedly different from that for the former (and it is, at least in the details). Other textbooks do the proof of Completeness for only one of the two logics, referring to some other textbook for a proof for the other logic. But in either case, connections related to Compactness between the two logics are lost or glossed over.[3]

The preceding are all among the many benefits of starting with Compactness and proving it only once for all the logics. But there is no free lunch and there is work to do. Most of the hard work is about setting up the means for the transition from Compactness for *propositional logic* to Compactness for the other logics, *i.e.*, the means to reduce the latter to the former. The reductions for the *logic of quantified Boolean formulas* and for *equality logic* are relatively straightforward based on two simple ideas (quantifier-elimination and a method of substitution) that are greatly amplified in what is called *Herbrand theory* in later sections. *Herbrand theory* is what we need for the reductions of the more complex logics: *zeroth-order logic*, *equational logic*, *quasi-equational logic*, and *first-order logic*.

An additional benefit of an excursion through *Herbrand theory* is that it has other important uses outside these notes. It plays the role of a *transfer principle* by reducing many questions of first-order logic to questions of propositional logic, all separate from Compactness. It can thus provide a unifying background for the study of other topics beyond the scope of these notes, such as the *tableaux* and *resolution* methods and *unification theory* (all good material for a follow-up course stressing *algorithmic* and *proof-theoretic* methods).

Finally, the obvious question: Why do we stop our presentation of "*From Compactness To Completeness*" at first-order? Can't we extend it to *second-order logic*, if only to fragments of the latter? The short answer: It takes a tiny amount of second-order quantification to collapse the entire edifice built on Compactness.

How to read these lecture notes

The Preface (this section) and the last Section 7 are for context and can be read separately from everything else.

The material in Section 1 on *propositional logic* gradually builds up, through successive sections on more complex logics, and ends in Section 6 on *first-order logic*. These sections, one for every logic, are intended to be read sequentially. Throughout, I insert small exercises (those that are untitled) right after incomplete or outlined arguments, which typically ask for supplying missing details; I consider them an integral part of the material and doing them should give a better grip on separating what is essential from the chaff in logical arguments. In the last subsection of every section, I include a few small applications and related exercises (those are titled) of the kind students in computer science encounter elsewhere in their studies.

I rarely cover all these sections in their entirety in a 14-week semester, where approximately half of the course work is devoted to separate material on the theory and pragmatics of using *SAT/SMT solvers* and *automated*

seem to gain wide acceptance. For example, this approach is implicit in R.M. Smullyan's book *First-Order Logic* [8] (see the proofs of Theorem 6 at the end of Chapter VI and Theorem 2 in Chapter VII). And it is explicit in G. Kreisel's and J.L. Krivine's book *Elements of Mathematical Logic* [6] (see their *Finiteness Theorem*, Theorem 12, in Chapter 2). However, it takes some doing to decode the notation in these two books, somewhat different from that in more recent publications.

[3] Here are three examples. In the book *Models and Ultraproducts: An Introduction* by J.L. Bell and A.B. Slomson [1], Completeness is proved twice, once for *propositional logic* in Section 2.3 and once for *first-order logic* in Section 3.5, the latter not given as a consequence of the former, and Compactness in both cases given after Completeness in Sections 2.4 and 5.4, respectively. In the book *Mathematical Logic* by J.D. Monk [7], Completeness is proved for *propositional logic* in Theorems 8.28 and 8.29, and again separately not as a corollary for *first-order logic* in Theorems 11.19 and 11.20; Compactness is then given as a consequence of the latter only, in Theorem 11.22. In the book *Logic and Structure* by D. van Dalen [10], Completeness is proved twice differently, once for *propositional logic* in Section 1.5, once for *first-order logic* in Section 3.1, with Compactness for the latter given as a consequence in Section 3.2.

theorem provers.[4] Material in these notes is therefore the basis of about one-half of the lectures, typically interspersed with the other half. I try to choose a pace that suits most of the students, who are typically first-year graduate students with a background in computer science. At a slower pace, I may reach Section 4 or Section 5; at a faster pace with a smaller or more advanced group of students, I may reach the end of Section 6.

All the appendices should be read as needed, or as much as students desire to read on their own. Except possibly for Appendices D and E, their material is not part of what I present in lecture, though I refer to them in homework assignments and in case there are questions in lecture. Appendices A, B, and C, are mostly a review of terminology and notation used in earlier parts of the notes. My preference is to leave Appendices D and E to students to read and learn by themselves, giving more time for applications in lectures; those two appendices present concrete examples (the familiar *de Morgan's laws*, *prenex forms*, and *skolemization*) of how we can deal with the same formulas in two different ways, semantically and proof-theoretically. I include Appendix F for those interested in knowing the topological connections of Compactness.

[4]Typically, I have used the SAT/SMT solver **Z3** with a **Python** or **OCaml** interface for friendlier interaction. The automated theorem prover **Prover9**, and its companion counter-example searcher **Mace4**, have been particularly easy to set up and use.

1 Propositional Logic (PL)

Let $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ be the set of well-formed formulas of *propositional logic* over the set $\mathcal{P}$ of *propositional variables*. We say a set $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ is *finitely satisfiable* iff every finite subset of Γ is satisfiable. If Γ is a finite set, then "finitely satisfiable" coincides with "satisfiable".

We write $\mathsf{models}(\Gamma)$ to denote the set of models of Γ. In the propositional case, $\mathsf{models}(\Gamma)$ is the set of all truth assignments to the propositional variables that satisfy every $\varphi \in \Gamma$. The next lemma is a preliminary result for the Compactness Theorem.

Lemma 1. *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ and $\varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. If Γ is finitely satisfiable, then $\Gamma \cup \{\varphi\}$ or $\Gamma \cup \{\neg\varphi\}$ (or possibly both) is finitely satisfiable.*

Proof. Suppose the conclusion of the lemma does not hold: Both $\Gamma \cup \{\varphi\}$ and $\Gamma \cup \{\neg\varphi\}$ are not finitely satisfiable. Hence, there are finite subsets $\Gamma_1 \subseteq \Gamma$ and $\Gamma_2 \subseteq \Gamma$ such that both $\Gamma_1 \cup \{\varphi\}$ and $\Gamma_2 \cup \{\neg\varphi\}$ are not satisfiable. Hence, both:

$$\mathsf{models}(\Gamma_1) \cap \mathsf{models}(\varphi) \;=\; \varnothing \qquad \text{and} \qquad \mathsf{models}(\Gamma_2) \cap \mathsf{models}(\neg\varphi) \;=\; \varnothing.$$

Hence, both $\mathsf{models}(\Gamma_1) \subseteq \mathsf{models}(\neg\varphi)$ and $\mathsf{models}(\Gamma_2) \subseteq \mathsf{models}(\varphi)$. Hence,

$$\mathsf{models}(\Gamma_1 \cup \Gamma_2) \;=\; \mathsf{models}(\Gamma_1) \cap \mathsf{models}(\Gamma_2) \;\subseteq\; \mathsf{models}(\neg\varphi) \cap \mathsf{models}(\varphi) \;=\; \varnothing.$$

Hence, the finite subset $\Gamma_1 \cup \Gamma_2$ does not have models, *i.e.*, is not satisfiable. Hence, Γ is not finitely satisfiable, and the hypothesis of the lemma does not hold either. $\square$

1.1 Compactness in PL

Theorem 2 (Compactness for Propositional Logic, Version I)**.** *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. It then follows that: Γ is satisfiable $\Leftrightarrow$ Γ is finitely satisfiable.*

Proof. The implication "$\Rightarrow$" is immediate. The non-trivial implication is "$\Leftarrow$": If Γ is finitely satisfiable, then Γ is satisfiable.

The set of propositional variables is $\mathcal{P} = \{p_0, p_1, p_2, \ldots\}$. Let $\varphi_1, \varphi_2, \varphi_3, \ldots$ be a fixed, countably infinite, enumeration of all the formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. We define a nested sequence of supersets of Γ as follows:

$$\Delta_0 \quad = \Gamma,$$
$$\Delta_{i+1} = \begin{cases} \Delta_i \cup \{\varphi_i\} & \text{if } \Delta_i \cup \{\varphi_i\} \text{ is finitely satisfiable,} \\ \Delta_i \cup \{\neg\varphi_i\} & \text{otherwise.} \end{cases}$$

Clearly, $\Gamma = \Delta_0 \subseteq \Delta_1 \subseteq \Delta_2 \subseteq \Delta_3 \subseteq \cdots$. By induction on $i \geqslant 0$, using Lemma 1, every Δ_i is a finitely satisfiable set of propositional wff's. We now define:

$$\Delta \;=\; \bigcup_i \Delta_i \qquad \text{(the limit of the } \Delta_i\text{'s)}$$

Two facts about Δ follow from its definition:

1. For every propositional wff φ, either $\varphi \in \Delta$ or $\neg\varphi \in \Delta$, but not both.
   This is why Δ is said *maximal finitely satisfiable*, soon to be shown just *maximal satisfiable*.

2. Since every propositional variable p_i is a wff itself, either $p_i \in \Delta$ or $\neg p_i \in \Delta$, but not both.

We next define a truth assignment σ as follows:

$$\sigma(p_i) \;=\; \begin{cases} \mathsf{T} & \text{if } p_i \in \Delta, \\ \mathsf{F} & \text{if } \neg p_i \in \Delta. \end{cases}$$

Claim: σ satisfies a propositional wff φ iff $\varphi \in \Delta$. We leave the proof of this claim as an (easy) exercise.

Hence, σ is a valuation satisfying every wff in Δ, *i.e.*, $\sigma \in \mathsf{models}(\Delta)$. Hence, because $\Gamma \subseteq \Delta$, it is also the case that σ satisfies every wff in Γ. Hence, Γ is satisfiable. $\square$

Exercise 3. Provide the details in the preceding proof showing that there is "a fixed, countably infinite, enumeration of all the formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$". Although not needed in the proof, we can state a stronger assertion: The fixed enumeration of all the formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ is *computable*, *i.e.*, can be generated by an infinitely-running computer program. $\square$

Exercise 4. In the definition of the nested sequence of Δ_i's in the preceding proof, we did *not* write:

$$\Delta_{i+1} \;=\; \begin{cases} \Delta_i \cup \{\varphi_i\} & \text{if } \Delta_i \cup \{\varphi_i\} \text{ is finitely satisfiable,} \\ \Delta_i \cup \{\neg\varphi_i\} & \text{if } \Delta_i \cup \{\neg\varphi_i\} \text{ is finitely satisfiable.} \end{cases}$$

Explain why. *Hint*: Exhibit a set Γ of wff's and a single wff φ such that both $\Gamma \cup \{\varphi\}$ and $\Gamma \cup \{\neg\varphi\}$ are satisfiable. $\square$

Exercise 5. Prove the **claim** in the penultimate paragraph of the proof of Theorem 2. There is no harm in simplifying the syntax a little, by restricting the logical connectives to two, say, $\{\neg, \vee\}$ or $\{\neg, \wedge\}$. *Hint*: Use structural induction on propositional wff's. $\square$

Lemma 6. *Let* $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ *and* $\varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$*, both arbitrary. We then have:*
$\Gamma \models \varphi \Leftrightarrow$ "$\Gamma \cup \{\neg\varphi\}$ *is unsatisfiable" – or, equivalently,* $\Gamma \not\models \varphi \Leftrightarrow$ "$\Gamma \cup \{\neg\varphi\}$ *is satisfiable".*

Proof. We have the following sequence of equivalences:

$$\begin{aligned} \Gamma \models \varphi \quad &\Leftrightarrow \quad \mathsf{models}(\Gamma) \subseteq \mathsf{models}(\varphi) \\ &\Leftrightarrow \quad \mathsf{models}(\Gamma) \cap \mathsf{models}(\neg\varphi) = \varnothing \\ &\Leftrightarrow \quad \mathsf{models}(\Gamma \cup \{\neg\varphi\}) = \varnothing \\ &\Leftrightarrow \quad \Gamma \cup \{\neg\varphi\} \text{ is unsatisfiable,} \end{aligned}$$

which is the desired conclusion. $\square$

Corollary 7 (Compactness for Propositional Logic, Version II)**.** *Let* $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ *and* $\varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$*, both arbitrary. We then have:* $\Gamma \models \varphi \Leftrightarrow$ *there is a finite subset* $\Gamma_0 \subseteq \Gamma$ *such that* $\Gamma_0 \models \varphi$*.*

Proof. The implication "$\Leftarrow$" is immediate. For the implication "$\Rightarrow$", we prove the contrapositive. So, suppose $\Gamma_0 \not\models \varphi$ for every finite subset $\Gamma_0 \subseteq \Gamma$. We have the following equivalences:

$$\begin{aligned} \Gamma_0 \not\models \varphi \text{ for every finite } \Gamma_0 \subseteq \Gamma \quad &\Leftrightarrow \quad \Gamma_0 \cup \{\neg\varphi\} \text{ satisfiable for every finite } \Gamma_0 \subseteq \Gamma \text{ (by Lemma 6)} \\ &\Leftrightarrow \quad \Gamma \cup \{\neg\varphi\} \text{ finitely satisfiable (by definition)} \\ &\Leftrightarrow \quad \Gamma \cup \{\neg\varphi\} \text{ satisfiable (by Theorem 2)} \\ &\Leftrightarrow \quad \Gamma \not\models \varphi \text{ (by Lemma 6)}\ , \end{aligned}$$

which is the desired conclusion. $\square$

Exercise 8 shows one way in which Compactness breaks down. The exercise involves an extension of PL which is called the *infinitary propositional logic* (*Infinitary PL*).

Exercise 8. We can restrict the logical connectives to $\{\neg, \vee, \wedge\}$. The set of propositional variables is again $\mathcal{P} = \{p_0, p_1, p_2, \ldots\}$, which is countably infinite. Suppose we extend this syntax with two new connectives, denoted $\bigvee$ and $\bigwedge$, each taking as a single argument a countably infinite set of previously defined wff's. The resulting syntax is one version of *Infinitary PL*. If Γ is a countably infinite set of the form $\Gamma = \{\varphi_1, \varphi_2, \varphi_3, \ldots\}$, then:

$$\bigvee \Gamma \;=\; \varphi_1 \vee \varphi_2 \vee \varphi_3 \vee \cdots ,$$

and similarly for $\bigwedge \Gamma$. There are three parts in this exercise:

1. Define the syntax of *Infinitary PL*, preferably in an extended BNF (Backus-Naur Form). Try to be as precise as you can, paying special attention to the presence of ellipses "..." in the definition – or can you think of a mathematical formulation that avoids any mention of ellipses?

2. Define the semantics of *Infinitary PL*, by structural induction on the syntax in Part 1, starting from an assignment σ of truth values to every member of $\mathcal{P}$ (for the base case of the induction).

3. Show that Theorem 2 does not hold, and therefore nor does Corollary 7, for *Infinitary PL*.

   Hint: Define a countably infinite set Γ of wff's such that every finite $\Gamma_0 \subseteq \Gamma$ is satisfiable, but Γ is not. *Further Hint*: Include the wff $\varphi = \bigvee\{\neg p_0, \neg p_1, \neg p_2, \ldots\}$ in your proposed Γ. □

1.2 From Compactness in PL to Completeness in PL

We are now ready for the transition. The next lemma is a weaker form of the Completeness Theorem; it does not need Compactness for its proof. The Completeness Theorem in full generality is Theorem 10 whose proof uses Compactness in an essential way (which we choose to take in the form of Corollary 7).

Lemma 9. *Let $\varphi_1, \ldots, \varphi_n, \psi$ be propositional wff's. If $\varphi_1, \ldots, \varphi_n \models \psi$ then $\varphi_1, \ldots, \varphi_n \vdash \psi$.*

Proof. This lemma is the Completeness Theorem as stated in the book by M. Huth and M. Ryan [5], in Section 1.4.4; specifically, this is the left-to-right implication in Corollary 1.39. □

The details of the preceding proof very much depend on the kind of proof system it is based on. But Lemma 9, or lemmas essentially asserting the same thing, in fact hold again for all the finitary proof systems of propositional logic other than *natural deduction*. The phrase "finitary proof system" is a bit loose, but you can take it to qualify a formal system that generates new finite expressions (*e.g.*, the sequents of propositional logic in natural-deduction style or the wff's of propositional logic in Hilbert style) from previously generated ones by means of finitely many rules that require each finitely many antecedents – without using any notion of infinite sequence or any notion of infinite set.[5]

[5] The words "finitary" and "infinitary" are used in several areas of mathematics and theoretical computer science with different though related meanings, and sometimes a little too loosely. They are all intended to mean certain things are "finite" and "infinite", but more precisely in relation to particular aspects of mathematically defined notions in different contexts.

For example, a *finitary operation* (resp. *relation*) is one which has *finite arity*, otherwise it is said to be an *infinitary operation* (resp. *infinitary relation*). So, if we say an algebra or relational structure is *finite* (resp. *infinite*), we mean its domain or universe is finite (resp. infinite) – this convention is firmly established – but if some authors say an algebra or relational structure is *finitary* (resp. *infinitary*),

Theorem 10 (Completeness for Propositional Logic)**.** *Let Γ be a set of propositional wff's (possibly infinite), and ψ a propositional wff. If $\Gamma \models \psi$, then $\Gamma \vdash \psi$.*

Proof. If $\Gamma \models \psi$, then there is a finite subset $\Gamma_0 \subseteq \Gamma$ such that $\Gamma_0 \models \psi$, by Corollary 7. By Lemma 9, it follows that $\Gamma_0 \vdash \psi$. Padding Γ_0 with the redundant premises in $(\Gamma - \Gamma_0)$, we conclude $\Gamma \vdash \psi$. □

1.3 Applications and Exercises

All the examples and exercises in this subsection are applications of Compactness. The final result in each of these applications can certainly be obtained by other means, but this is not immediately obvious and may require some tricky combinatorics. Compactness provides an elegant alternative. In each of the applications, most of the hard work is to formulate necessary and sufficient conditions for a solution in the form of an infinite set of propositional wff's; after which, satisfaction of those conditions is obtained by a relatively easy invocation of Compactness.

Example 11 (*Topological Sorting*)**.** A standard exercise in an undergraduate course on discrete algorithms is to show that every finite *directed acyclic graph* (dag) G can be *topologically sorted*, which means that the vertices of G can be linearly ordered on a horizontal line such that all the edges of G are drawn in the same direction, from left to right. We extend this result to infinite graphs: *Every infinite dag can be topologically sorted.*

Let $G \stackrel{\text{def}}{=} (V, E)$ be an infinite directed graph, where V is the set of vertices which we choose to name with the positive integers $\{1, 2, \ldots\}$, and $E \subseteq V \times V$ is the set of edges. For convenience, we use two sets of doubly-indexed propositional variables, $\mathcal{Q}$ and $\mathcal{R}$, instead of $\mathcal{P}$:

$$\mathcal{Q} \stackrel{\text{def}}{=} \Big\{ q_{i,j} \;\Big|\; i, j \in \{1, 2, \ldots\} \Big\} \quad \text{and} \quad \mathcal{R} \stackrel{\text{def}}{=} \Big\{ r_{i,j} \;\Big|\; i, j \in \{1, 2, \ldots\} \Big\}.$$

The propositional wff's in this example are in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{R})$. To facilitate our modeling of G's properties below, we purposely use names of vertices, such as i and j, as indices to identify variables $q_{i,j}$ and $r_{i,j}$. We consider initial finite fragments of the graph G, based on increasingly larger subsets of vertices:

$$V_1 \stackrel{\text{def}}{=} \{1\},\; V_2 \stackrel{\text{def}}{=} \{1, 2\}, \ldots,\; V_n \stackrel{\text{def}}{=} \{1, 2, \ldots, n\},\; \ldots \quad \text{so that also } V_1 \subseteq V_2 \subseteq \;\cdots\; \subseteq V_n \subseteq \cdots .$$

We write G_n for the finite subgraph of G induced by the vertices in V_n, *i.e.*, $G_n \stackrel{\text{def}}{=} (V_n, E_n)$ where $E_n = E \cap (V_n \times V_n)$. By this definition, G_n is a finite subgraph of $G_{n'}$, which is in turn a finite subgraph of the full graph G, for all $1 \leqslant n < n'$. Satisfaction of the following wff:

$$\pi_n \stackrel{\text{def}}{=} \bigwedge \Big\{ q_{i,j} \;\Big|\; (i, j) \in E_n \Big\} \;\wedge\; \bigwedge \Big\{ \neg q_{i,j} \;\Big|\; (i, j) \notin E_n \Big\}$$

determines subgraph G_n up to isomorphism, where "$\bigwedge$" stands for multiple conjunction, *i.e.*, $\bigwedge\{\varphi_1, \varphi_2, \ldots, \varphi_k\}$ means $(\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_k)$. This is so because $q_{i,j}$ is satisfied (*i.e.*, assigned truth-value $\mathbf{T}$) iff there is an edge from i to j, and $\neg q_{i,j}$ is satisfied (*i.e.*, assigned truth-value $\mathbf{T}$) iff there is no edge from i to j.

---

they mean its operations and relations have each finite arity (resp. at least one has infinite arity), regardless of the size of its universe.

In a different context, a *finitary* logic means one whose formulas and formal derivations all have finite length, otherwise the logic is said to be *infinitary*. Elsewhere, some authors have used the phrase *finitary mathematics* to mean mathematics that can be expressed without invoking infinite sets in any way. If you want to go deeper into the usage of "finite" versus "finitary", and "infinite" versus "infinitary", search the Web.

Said differently, satisfaction of $q_{i,j}$ corresponds to the assertion "there is a path of length $= 1$ from vertex i to vertex j". We want to use satisfaction of variable $r_{i,j}$ to model the more general assertion "there is a finite path of length $\geqslant 1$ from vertex i to vertex j". We thus define another wff ρ_n as:

$$\rho_n \stackrel{\text{def}}{=} \bigwedge\Big\{\, q_{i,j} \to r_{i,j} \;\Big|\; (i,j) \in E_n \,\Big\} \;\wedge\; \bigwedge\Big\{\, q_{i,j} \wedge r_{j,k} \to r_{i,k} \;\Big|\; (i,j) \in E_n \text{ and } k \in \{1,\ldots,n\} \,\Big\}$$

Note carefully how the second part of ρ_n is defined. Informally in words, satisfaction of $r_{i,j}$ implies the existence of a path from i to j in G_n or, equivalently, the existence of an edge in the *transitive closure* of G_n.

We define one more wff θ_n, which models the assertion that "for one of the vertices i there is a path from i back to i in the transitive closure of G_n":

$$\theta_n \stackrel{\text{def}}{=} \bigvee\Big\{\, r_{i,i} \;\Big|\; i \in \{1,\ldots,n\} \,\Big\}.$$

The wff θ_n is satisfied by G_n iff G_n contains a cycle. Hence, $\neg\theta_n$ is satisfied by G_n iff G_n is acyclic. Finally, consider the infinite set Γ of propositional wff's defined by:

$$\Gamma \stackrel{\text{def}}{=} \Big\{\, \pi_n \;\Big|\; n \geqslant 1 \,\Big\} \cup \Big\{\, (\pi_n \wedge \rho_n \to \neg\theta_n) \;\Big|\; n \geqslant 1 \,\Big\}.$$

If the full graph G is acyclic, then each of the finite subgraphs G_n is acyclic, which in turn implies that every finite subset of Γ is satisfiable. By Compactness, the full set of wff's Γ is satisfiable, which implies the full set of vertices $\{1, 2, \ldots, n, \ldots\}$ can be linearly ordered such that all the edges are drawn from left to right. □

Exercise 12 (*Graph Coloring*)**.** An undirected graph G is said *k-colorable* if it is possible to assign only one of k colors to every vertex of G such that the two endpoints of every edge are assigned different colors. A famous (and very difficult to prove) result of graph theory is that *every finite planar graph is 4-colorable*. In this exercise you are asked to show that this result can be extended to *infinite* planar graphs using Compactness for PL.

We specify a graph $G \stackrel{\text{def}}{=} (V, E)$, finite or infinite, by its set of vertices V, which we take as an initial fragment of the positive integers $\{1, 2, \ldots\}$, and its set of edges $E \subseteq V \times V$. Graphs in this exercise are simple: undirected, with no self-loops and no multiple edges connecting the same two vertices.

Let $k \geqslant 1$ be fixed, the number of available colors. For convenience, use two separate sets of propositional variables, $\mathcal{Q}$ and $\mathcal{C}$, instead of $\mathcal{P}$:

$$\mathcal{Q} \stackrel{\text{def}}{=} \Big\{\, q_{i,j} \;\Big|\; i, j \in \{1, 2, \ldots\} \,\Big\} \quad \text{and} \quad \mathcal{C} \stackrel{\text{def}}{=} \Big\{\, c_i^j \;\Big|\; i \in \{1, 2, \ldots\} \text{ and } 1 \leqslant j \leqslant k \,\Big\}.$$

All wff's in this exercise should be in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q} \cup \mathcal{C})$. Use variables in $\mathcal{Q}$ to model a given graph G: there is an edge connecting two distinct vertices i and j iff $q_{i,j}$ is set to truth value $\mathbf{T}$. Use variables in $\mathcal{C}$ to model G's coloring: vertex i is assigned color $j \in \{1, \ldots, k\}$ iff c_i^j is set to truth value $\mathbf{T}$. There are two parts in this exercise:

1. Let $G \stackrel{\text{def}}{=} (V, E)$ be finite, with $V = \{1, 2, \ldots, n\}$ for some $n \geqslant 1$. Write a wff φ_n which is satisfied iff G is k-colorable.

   Hint: Define φ_n in two parts, one part specifies the structure of G (including conditions that there are no self-loops and that G is undirected), and one part specifies that G is k-colorable.

2. Let $G \stackrel{\text{def}}{=} (V, E)$ be an infinite planar graph, with $V = \{1, 2, \ldots\}$, and let $k = 4$. Use Compactness for PL to write a rigorous argument showing that G is 4-colorable.

   Hint: If an infinite G is k-colorable, then so is every finite subgraph of G. Compactness should give you the converse. □

Exercise 13 (*Queens Problem*)**.** The *n-Queens Problem* is the problem of placing n queens on an $n \times n$ chessboard so that no two queens can attack each other. A solution of the problem when $n = 6$ is shown on the left of Figure 1 and three solutions are shown in Figure 2. In this exercise we specify the requirements of a solution for the n-Queens Problem as a propositional wff ψ_n, with one such wff for every $n \geqslant 4$. (There are no solutions for $n = 2$ and $n = 3$.) For convenience, we use a set $\mathcal{Q}$ of doubly-indexed propositional variables, instead of $\mathcal{P}$, where the indices range over the positive integers:

$$\mathcal{Q} \stackrel{\text{def}}{=} \left\{ q_{i,j} \;\middle|\; i, j \in \{1, 2, \dots\} \right\}.$$

The desired wff ψ_n in this exercise is in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q})$. We set the variable $q_{i,j}$ to truth value **T** (resp. **F**) if there is (resp. there is not) a queen placed in position (i, j) of the board, where we take the first index i (resp. the second index j) to range over the vertical axis downward (resp. the horizontal axis rightward); that is, i is a row number and j is a column number.[6] There are four parts in this exercise:

1. Write the wff ψ_n and justify how it accomplishes its task.

   Hint: Write ψ_n as a conjunction $\psi_n^{\text{row}} \wedge \psi_n^{\text{col}} \wedge \psi_n^{\text{diag1}} \wedge \psi_n^{\text{diag2}}$, where:

   (a) ψ_n^{row} is satisfied iff there is exactly one queen in each row,

   (b) ψ_n^{col} is satisfied iff there is exactly one queen in each column,

   (c) ψ_n^{diag1} is satisfied iff there is at most one queen in each diagonal,

   (d) ψ_n^{diag2} is satisfied iff there is at most one queen in each antidiagonal.

   Further Hint: Given any two distinct positions (i_1, j_1) and (i_2, j_2) along a diagonal, it is always the case that $i_1 - j_1 = i_2 - j_2$. And if the two positions are along an antidiagonal, then it is always the case that $i_1 + j_1 = i_2 + j_2$.

2. Imagine now an infinite chessboard, which occupies the entire south-east quadrant of the Cartesian plane. The coordinates along the vertical and horizontal axes are, respectively, i (increasing downward) and j (increasing rightward), both ranging over the positive integers $\{1, 2, \ldots\}$. In an attempt to repeat the argument in Example 11 and Exercise 12, someone once defined the set of wff's $\Gamma \stackrel{\text{def}}{=} \{\, \psi_n \mid n \geqslant 4 \,\}$, and wrote the following (in outline here):

   > The set Γ is finitely satisfiable and, therefore, satisfiable by Compactness. Hence, a solution of the *Infinite Queens Problem* exists, which satisfies conditions $\{$(a), (b), (c), (d)$\}$ for all $n \geqslant 4$.

   What is wrong with the preceding argument? The answer is subtle and you need to be careful.

3. What we call a *good board* has size $(5 \cdot 3^k) \times (5 \cdot 3^k)$, which can be divided into 9 sub-boards each of size $(5 \cdot 3^{k-1}) \times (5 \cdot 3^{k-1})$ for every $k \geqslant 1$. Figure 3 depicts the shape of a *good board*. The *initial good board* of size $(5 \cdot 3^1) \times (5 \cdot 3^1) = 15 \times 15$ is shown in Figure 4; call it B_1. For every $k \geqslant 2$, the corresponding *good board*, call it B_k, is obtained by inserting a copy of B_{k-1} in each of the three sub-boards of size $(5 \cdot 3^{k-1}) \times (5 \cdot 3^{k-1})$, in positions: north-west, south-middle, and east-middle. We have thus defined an infinite sequence of boards: $B_1, B_2, \ldots, B_k, \ldots$ such that for every $k \geqslant 1$, the board B_k is embedded in the north-west corner of B_{k+1}. If we take $B \subseteq B'$ to mean board B is a sub-board of B', we may write:[7]

   $$B_1 \subseteq B_2 \subseteq \cdots \subseteq B_k \subseteq \cdots .$$

[6] This is the standard convention of identifying rows and columns in a two-dimensional matrix, which is not how we usually view the coordinates of the Cartesian plane, where the first coordinate is along the horizontal axis (going rightward) and the second coordinate is along the vertical axis (going upward). See Figure 1 for our convention. Also, following the conventions of two-dimensional matrices, a *diagonal* is a (-45^o)-*diagonal* directed downward starting from the west or north edge, and an *antidiagonal* is a $(+45^o)$-*diagonal* directed upward starting from the west or south edge.

[7] We choose here to take *good boards* to have size $(5 \cdot 3^k) \times (5 \cdot 3^k)$, starting from an *initial good board* of size 15×15. But there are many other alternatives. For example, we can take *good boards* to have size $(7 \cdot 3^k) \times (7 \cdot 3^k)$ – or resp., size $(9 \cdot 3^k) \times (9 \cdot 3^k)$ –

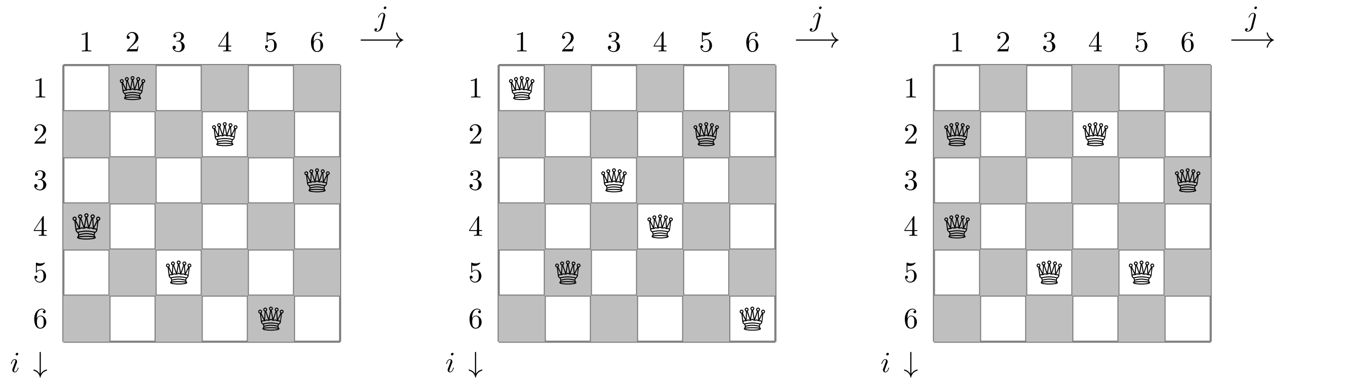

Figure 1: The 6-Queens Problem in Exercise 13: a solution *on the left*, a non-solution *in the middle* (satisfying conditions (a) and (b) only), a non-solution *on the right* (satisfying conditions (c) and (d) only).

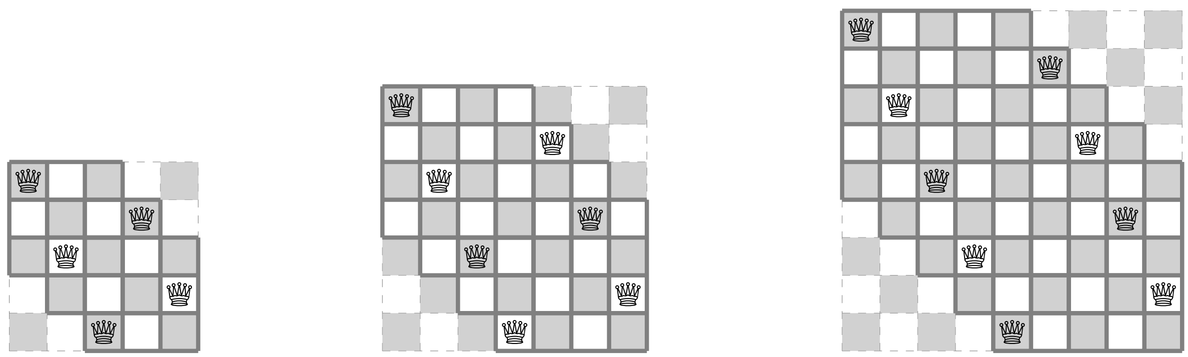

Figure 2: Solutions for the n-Queens Problem in Exercise 13 when $n \in \{5, 7, 9\}$. For every odd n, we use the same pattern to generate a solution, here called a *smooth board*, whereby the n queens are placed on n adjacent diagonals, and the $(n-1)$ empty diagonals are equally divided between the south-west and north-east corners.

For arbitrary $k \geqslant 1$, write a propositional wff $\theta_k \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q})$ whose satisfaction uniquely describes the *good board* B_k as a solution for the $(5 \cdot 3^k)$-queens problem.

Hint: Start by defining θ_1. Proceed by induction on $k \geqslant 2$ to define θ_k. Exploit the fact that a truth assignment satisfying θ_k also satisfies θ_{k-1}.

4. Define the set of wff's $\Theta \stackrel{\text{def}}{=} \{\, \theta_k \mid k \geqslant 1 \,\}$ and use Compactness for PL to give a rigorous argument that the *Infinite Queens Problem* has indeed a solution. (No credit if you try an argument that does not invoke Compactness precisely.) □

Exercise 14 (*Not-Three-In-Line Problem*)**.** This is an old problem of discrete geometry, not yet fully resolved as of this writing, which asks for the maximum number of pebbles that can be placed on an $n \times n$ chessboard so that no three pebbles are collinear, *i.e.*, not on the same row or column or diagonal. An upper bound on the number of pebbles is $2n$ because, by the Pigeonhole Principle, placing $2n + 1$ pebbles makes one row or one column necessarily contain three of them. But is $2n$ a reachable upper bound for all n? See Figure 5 for two solutions when $n = 10$, in which case the upper bound $2n = 20$ is reached.

We can take the *Not-Three-In-Line Problem* as a more complex variation on the n*-Queens Problem* in Exercise 13, with "pebbles" instead of "queens". Specifically, we use the same set $\mathcal{Q}$ of doubly-indexed proposi-

starting from an *initial good board* of size 21×21 – or resp., size 27×27 – in which are inserted three copies of the (7×7) *smooth board* – or resp., the (9×9) *smooth board*. See Figure 2 for the definition of *smooth boards*. If you are interested in other solutions for the *Infinite Queens Problem* besides those presented here, see the recent article by Dekking, Shallit, and Sloane [3].

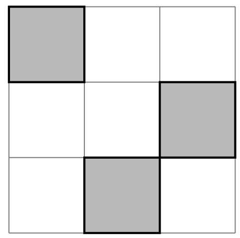

Figure 3: Shape of the *good board* B_k in part 3 of Exercise 13: Its size is $(5 \cdot 3^k) \times (5 \cdot 3^k)$, where $k \geqslant 1$. The non-empty sub-boards (shaded areas) are each a copy of B_{k-1} when $k \geqslant 2$; the *initial good board* B_1 is shown in Figure 4. For every $k \geqslant 1$, B_k is a solution of the $(5 \cdot 3^k)$-queens problem, with all queens placed on adjacent diagonals and with $\big((5 \cdot 3^k) - 1\big)$ empty diagonals equally divided between the south-west and north-east corners.

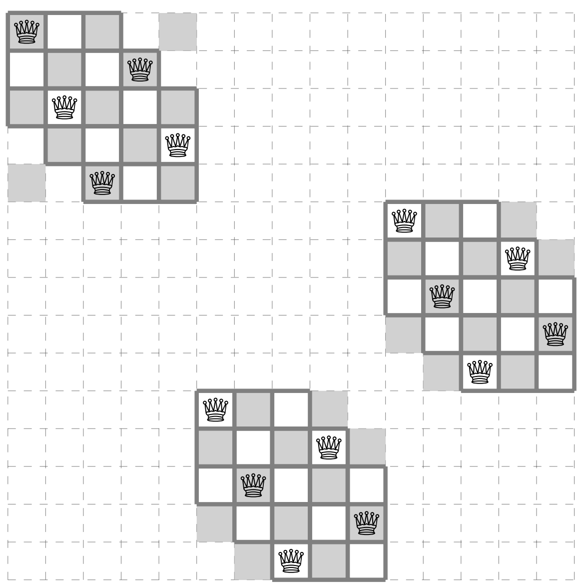

Figure 4: The *initial good board* B_1 of size 15×15 in Exercise 13, which solves the 15-queens problem, with all queens on adjacent diagonals and all empty diagonals equally divided between the south-west and north-east corners.

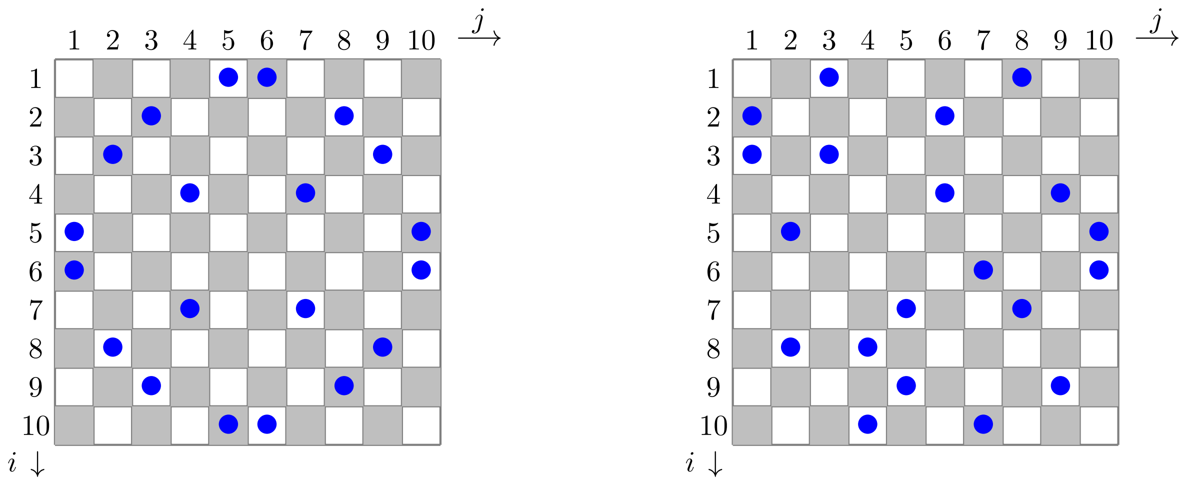

Figure 5: Two solutions of the Not-Three-In-Line Problem when $n = 10$ in Exercise 14, both satisfying conditions (a), (b), (c), and (d), in the exercise.

tional variables, with variable $q_{i,j}$ assigned truth-value **T** (resp. **F**) if there is (resp. there is not) a pebble placed in position (i, j) of the board, where the index i (resp. the index j) ranges over the vertical axis downward (resp. the horizontal axis rightward). There are two parts in this exercise:

1. Write a wff $\psi_n \stackrel{\text{def}}{=} \psi_n^{\text{row}} \wedge \psi_n^{\text{col}} \wedge \psi_n^{\text{diag1}} \wedge \psi_n^{\text{diag2}}$ in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{Q})$ according to the following specification:
   (a) ψ_n^{row} is satisfied iff there are exactly two pebbles in each row,
   (b) ψ_n^{col} is satisfied iff there are exactly two pebbles in each column,
   (c) ψ_n^{diag1} is satisfied iff there are at most two pebbles in each diagonal,
   (d) ψ_n^{diag2} is satisfied iff there are at most two pebbles in each antidiagonal.

   Hint: Given two distinct positions (i_1, j_1) and (i_2, j_2) along a diagonal, it is always that $i_1 - j_1 = i_2 - j_2$. And if the two positions are along an antidiagonal, then it is always that $i_1 + j_1 = i_2 + j_2$.

2. We extend the *Not-Three-In-Line Problem* to an infinite chessboard, which occupies the entire south-east quadrant of the Cartesian plane. Give a precise argument, based on Compactness, showing that if the problem can be solved for each $n \geqslant 4$ (there is no solution for $n < 4$), then a solution exists for the *Infinite Not-Three-In-Line Problem*.[8] □

[8] It is known that the *Not-Three-In-Line Problem* has a solution for every $n \leqslant 46$, but not for $n > 46$ as here formulated. It is conjectured that, for $n > 46$, no matter how you place $2n$ pebbles on the board, you are doomed to find three of them that are collinear. Put differently, for $n > 46$, it is conjectured that you are forced to place fewer than $2n$ pebbles to avoid three collinear pebbles. For more information on the *Not-Three-In-Line Problem*, search the Web.

2 The Logic of QBF's (QBF)

We can prove Compactness for the logic of *quantified Boolean formulas* (QBF's) by reducing it to Compactness for *propositional logic*. We have already done much of the preliminary work in Section 1. We write $\mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$ for the set of all QBF's over the set $\mathcal{P}$ of propositional variables.

Lemma 15. *Let Γ be a subset, finite or infinite, of $\mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$. We can construct a set Γ' of quantifier-free formulas in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ such that:*

1. *Γ is finitely satisfiable iff Γ' is finitely satisfiable.*
2. *Γ is satisfiable iff Γ' is satisfiable.*

The construction in the proof below establishes a stronger result: Γ and Γ' are more than *finitely equisatisfiable* and *equisatisfiable*; they are in fact *equivalent* (informally, "they say the same thing"). Specifically, for every wff $\varphi \in \Gamma$ there is a wff $\varphi' \in \Gamma'$ such that φ and φ' are equivalent; and, similarly, for every propositional wff $\varphi' \in \Gamma'$ there is a wff $\varphi \in \Gamma$ such that φ and φ' are equivalent.[9]

Proof. If φ is a propositional wff, we write "$\varphi[p := \bot]$" and "$\varphi[p := \top]$" to denote the substitution of the symbols $\bot$ and $\top$, respectively, for every occurrence of variable p in φ.

We define a translation from QBF to PL, named "$\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}$" by structural induction:[10]

1. $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(p) \stackrel{\text{def}}{=} p$ (for every variable p)
2. $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\neg\varphi) \stackrel{\text{def}}{=} \neg \boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)$
3. $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi \wedge \psi) \stackrel{\text{def}}{=} \boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi) \wedge \boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\psi)$
4. $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi \vee \psi) \stackrel{\text{def}}{=} \boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi) \vee \boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\psi)$
5. $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi \to \psi) \stackrel{\text{def}}{=} \boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi) \to \boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\psi)$
6. $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\forall p\, \varphi) \stackrel{\text{def}}{=} \Big(\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)\Big)[p := \bot] \;\wedge\; \Big(\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)\Big)[p := \top]$
7. $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\exists p\, \varphi) \stackrel{\text{def}}{=} \Big(\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)\Big)[p := \bot] \;\vee\; \Big(\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)\Big)[p := \top]$

Claim: For every QBF φ, the transformation $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)$ satisfies the following properties:

(a) $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)$ is a propositional wff,

(b) the set of free variables $\mathrm{FV}(\varphi)$ in φ are exactly all the variables occurring in $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)$, and

(c) if $X = \mathrm{FV}(\varphi)$, then for every truth assignment σ to the members of X, it holds that σ satisfies φ iff σ satisfies $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}(\varphi)$.

[9] If so, why use QBF instead of PL? Applications and exercises in Section 2.3 illustrate some of the advantages – just a few out of many – of using QBF rather than PL. In particular, *quantified Boolean formulas* are central in the study of what is called the *polynomial-time hierarchy* in computational complexity, something outside the scope of these notes.

[10] I quickly run out of notation. To simplify my task, I denote translations of syntax in a particular way: Each is denoted by a framed box and what is inside the box, here "QBF $\mapsto$ PL", suggests what the translation does. The box and its contents is a single name.

Part (c) in this claim shows that φ and $\boxed{\textsf{QBF} \mapsto \textsf{PL}}(\varphi)$ are not only equisatisfiable, but also equivalent. We leave the proof of this claim as an exercise. Given an arbitrary subset $\Gamma \subseteq \textsf{WFF}_{\textsf{QBF}}(\mathcal{P})$, we now define Γ' by:

$$\Gamma' \stackrel{\text{def}}{=} \left\{ \boxed{\textsf{QBF} \mapsto \textsf{PL}}(\varphi) \;\middle|\; \varphi \in \Gamma \right\}$$

By the preceding claim, we conclude that for every truth assignment σ to $\mathcal{P}$:

- for every finite subset $\Delta \subseteq \Gamma$ there is a finite subset $\Delta' \subseteq \Gamma'$ s.t. σ satisfies Δ iff σ satisfies Δ',
- for every finite subset $\Delta' \subseteq \Gamma'$ there is a finite subset $\Delta \subseteq \Gamma$ s.t. σ satisfies Δ iff σ satisfies Δ',
- σ satisfies Γ iff σ satisfies Γ'.

We leave the missing details in the proof of the preceding three bullet points as an exercise. $\square$

Exercise 16. Prove the **claim** in the proof of Lemma 15. *Hint*: Use structural induction on QBF's, following the seven steps in the definition of the transformation $\boxed{\textsf{QBF} \mapsto \textsf{PL}}$. $\square$

Exercise 17. In the statement of Lemma 15 and its proof, the set Γ of QBF's and the set Γ' of propositional wff's are equivalent. Specify:

1. Conditions under which $\big|\Gamma\big| = \big|\Gamma'\big|$, and
2. Conditions under which $\big|\Gamma\big| > \big|\Gamma'\big|$,

where $\big|\Gamma\big|$ is the cardinality of the set Γ. *Hint*: Consider, for example, the case when all the QBF's in Γ are *closed*; what is Γ' in this case? $\square$

Exercise 18. Supply the missing details in the proof of the three bullet points at the end of the proof of Lemma 15. *Hint*: This is subtler than at first blush; do Exercise 17 before you attempt this one. $\square$

2.1 From Compactness in PL to Compactness in QBF

Theorem 19 (Compactness for the Logic of QBF's, Version I)**.** *Let* $\Gamma \subseteq \textsf{WFF}_{\textsf{QBF}}(\mathcal{P})$*. It then holds that* Γ *is satisfiable iff* Γ *is finitely satisfiable.*

Proof. The left-to-right implication is immediate. The non-trivial is the right-to-left implication, *i.e.*, we have to prove that if Γ is finitely satisfiable, then Γ is satisfiable. Let Γ' be the set of propositional wff's defined from Γ according to Lemma 15.

By Lemma 15, Γ is finitely satisfiable iff Γ' is finitely satisfiable. By Theorem 2, Γ' is finitely satisfiable iff Γ' is satisfiable. By Lemma 15 once more, Γ' is satisfiable iff Γ is satisfiable. Hence, if Γ is finitely satisfiable, then Γ is satisfiable, as desired. $\square$

For the next lemma and its corollary, review the formal semantics of QBF's in Appendix B.

Lemma 20. *Let* $\Gamma \subseteq \textsf{WFF}_{\textsf{QBF}}(\mathcal{P})$ *and* $\varphi \in \textsf{WFF}_{\textsf{QBF}}(\mathcal{P})$*, both arbitrary. It then holds that* $\Gamma \models \varphi$ *iff* $\Gamma \cup \{\neg\varphi\}$ *is unsatisfiable – or, equivalently,* $\Gamma \not\models \varphi$ *iff* $\Gamma \cup \{\neg\varphi\}$ *is satisfiable.*

Proof. Identical to the proof of Lemma 6, except that here Γ is a set of QBF's and φ is a QBF. $\square$

Corollary 21 (Compactness for the Logic of QBF's, Version II)**.** *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$ and $\varphi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$. It then holds that $\Gamma \models \varphi$ iff there is a finite subset $\Gamma_0 \subseteq \Gamma$ such that $\Gamma_0 \models \varphi$.*

Proof. Identical to the proof of Corollary 7, except that here Γ is a set of QBF's and φ is a QBF. Moreover, here we invoke Lemma 20 instead of Lemma 6, and Theorem 19 instead of Theorem 2. $\square$

2.2 From Compactness in QBF to Completeness in QBF

Before turning to Completeness for the logic of QBF's, review the proof rules for QBF's in Appendix C. Although the proof rules in Appendix C are in *natural deduction* style, Completeness holds for any of the other available proof systems for the logic of QBF's.

The next lemma is a weaker form of the Completeness Theorem for QBF's, *i.e.*, it is restricted to a finite set of formulas $\{\varphi_1, \ldots, \varphi_n\}$. The Completeness Theorem for QBF's in full generality is Theorem 24.

Lemma 22. *Let $\varphi_1, \ldots, \varphi_n, \psi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$. If $\varphi_1, \ldots, \varphi_n \models \psi$ then $\varphi_1, \ldots, \varphi_n \vdash \psi$.*

Proof. This lemma is proved in the same way as Lemma 9, following the steps of the proof of the Completeness Theorem for propositional logic, as stated in the book [LCS], in Section 1.4.4; specifically, this is the left-to-right implication in Corollary 1.39.[11] $\square$

Exercise 23. Write the details of the proof for Lemma 22. $\square$

Theorem 24 (Completeness for the Logic of QBF's)**.** *Let $\Gamma \subseteq \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$ and $\psi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$. If $\Gamma \models \psi$, then $\Gamma \vdash \psi$.*

Proof. Identical to the proof of Theorem 10, except that all formulas are now QBF's, not just propositional wff's. Moreover, we need to invoke Corollary 21 instead of Corollary 7, and Lemma 22 instead of Lemma 9. $\square$

2.3 Applications and Exercises

Example 25 (*Transition Systems*)**.** A *transition system* (sometime called a *state-transition system*) is specified as a structure $\mathcal{M} \stackrel{\text{def}}{=} (\mathsf{States}, R, \mathsf{Init}, \mathsf{End})$ where States is a finite or infinite set, $R \subseteq \mathsf{States} \times \mathsf{States}$ is a binary relation (*the transition relation*), and $\mathsf{Init} \subseteq \mathsf{States}$ and $\mathsf{End} \subseteq \mathsf{States}$ are the subsets of *initial states* and *end states*, respectively.

When States is a finite set, $\mathcal{M}$ is conveniently represented by a finite directed graph; an example is shown in Figure 6. Each state of the system is a *node* in the graph and each possible transition from a state to another is a directed *edge*.

We can uniquely identify each state by a bit vector, with $B = \{\bot, \top\}$ as the set of bits. For the system in Figure 6 with 4 states, 2-bit vectors suffice for this encoding; in this cae, we can model the *transition relation*

[11] Michael Huth and Mark Ryan, *Logic in Computer Science*, Second Edition, Cambridge University Press, 2004.

by a propositional wff θ with propositional variables $\{p_1, p_2, p_3, p_4\}$ where we use the pairs (p_1, p_2) and (p_3, p_4) to encode the *from-state* and *to-state* of a transition, respectively. The setup in full generality is thus:

$$\begin{array}{rcll}
\mathsf{encode} & : & \mathsf{States} \to B^n & \text{(where } n = \lceil \log_2 \mathsf{size}(\mathsf{States}) \rceil\text{),} \\
\mathsf{init} & : & B^n \to \{\mathbf{F}, \mathbf{T}\} & \text{(the set of initial states),} \\
\mathsf{end} & : & B^n \to \{\mathbf{F}, \mathbf{T}\} & \text{(the set of end states),} \\
\theta & : & B^n \times B^n \to \{\mathbf{F}, \mathbf{T}\} & \text{(the transition relation).}
\end{array}$$

Whichever is more convenient, we write $\{\mathsf{init}, \mathsf{end}, \theta\}$ as functions (as above) or sometimes as unary and binary relations; either way, they are translated into propositional wff's whose interpretations are values in $\{\mathbf{F}, \mathbf{T}\}$. Note that the symbol "$R$", "$\mathsf{Init}$", and "$\mathsf{End}$", are not part of the vocabulary of PL and QBF, which is why we need to write three wff's $\{\,\theta,\ \mathsf{init},\ \mathsf{end}\,\}$ in the syntax of PL and QBF to formally model these relations.

For the particular transition system in Figure 6 where $n = \log_2 4 = 2$, the setup is thus:

$$\begin{array}{rcll}
\mathsf{encode}(\mathsf{States}) & \stackrel{\text{def}}{=} & \Big\{(\bot,\bot),(\bot,\top),(\top,\bot),(\top,\top)\Big\}, & \\
\mathsf{init}(p_1,p_2) & \stackrel{\text{def}}{=} & (p_1 \leftrightarrow \bot) \wedge (p_2 \leftrightarrow \bot) = (\neg p_1 \wedge \neg p_2) \ \text{ or also } \ \mathsf{init} \stackrel{\text{def}}{=} \{(\bot,\bot)\}, & \\
\mathsf{end}(p_1,p_2) & \stackrel{\text{def}}{=} & (p_1 \leftrightarrow \top) \wedge (p_2 \leftrightarrow \top) = (p_1 \wedge p_2) \ \text{ or also } \ \mathsf{end} \stackrel{\text{def}}{=} \{(\top,\top)\}, & \\
\theta(p_1,p_2,p_3,p_4) & \stackrel{\text{def}}{=} & \big((\neg p_1 \wedge \neg p_2) \ \to \ (\neg p_3 \wedge \neg p_4) \vee (\neg p_3 \wedge p_4)\big) & \text{(from } s_1\text{)} \\
& & \wedge \ \big((\neg p_1 \wedge p_2) \ \to \ (\neg p_3 \wedge p_4) \vee (p_3 \wedge p_4)\big) & \text{(from } s_2\text{)} \\
& & \wedge \ \big((p_1 \wedge \neg p_2) \ \to \ (\neg p_3 \wedge p_4) \vee (\neg p_3 \wedge \neg p_4) \vee (p_3 \wedge p_4)\big) & \text{(from } s_3\text{)} \\
& & \wedge \ \big((\neg p_3 \wedge \neg p_4) \ \to \ (\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2)\big) & \text{(to } s_1\text{)} \\
& & \wedge \ \big((\neg p_3 \wedge p_4) \ \to \ (\neg p_1 \wedge \neg p_2) \vee (\neg p_1 \wedge p_2) \vee (p_1 \wedge \neg p_2)\big) & \text{(to } s_2\text{)} \\
& & \wedge \ \big((p_3 \wedge \neg p_4) \ \to \ (\neg p_1 \wedge p_2)\big) & \text{(to } s_3\text{)} \\
& & \wedge \ \big((p_3 \wedge p_4) \ \to \ (p_1 \wedge \neg p_2)\big) & \text{(to } s_4\text{)}
\end{array}$$

Note how we write θ: it is a conjunction of seven implications, three implications for the *from-state* part of transitions (or the *tail* end of edges) and four implications for the *to-state* part of transitions (or the *head* end of edges). Convince yourself that θ faithfully models the behavior of the relation R (a little painstaking task!).

With the preceding we can now express problems of *reachability* in the transition system, namely, whether some states are reachable from other states. Consider the following wff's as an example, where we purposely use a different set of propositional variables $\{q_1, q_2, \ldots\}$:

$$\begin{array}{rcl}
\varphi_1(q_1,\ldots,q_4) & \stackrel{\text{def}}{=} & \mathsf{init}(q_1,q_2) \wedge \theta(q_1,q_2,q_3,q_4) \wedge \mathsf{end}(q_3,q_4), \\
\varphi_2(q_1,\ldots,q_6) & \stackrel{\text{def}}{=} & \mathsf{init}(q_1,q_2) \wedge \theta(q_1,q_2,q_3,q_4) \wedge \theta(q_3,q_4,q_5,q_6) \wedge \mathsf{end}(q_5,q_6), \\
\varphi_3(q_1,\ldots,q_8) & \stackrel{\text{def}}{=} & \mathsf{init}(q_1,q_2) \wedge \theta(q_1,q_2,q_3,q_4) \wedge \theta(q_3,q_4,q_5,q_6) \wedge \theta(q_5,q_6,q_7,q_8) \wedge \mathsf{end}(q_7,q_8).
\end{array}$$

The wff φ_1 (resp. φ_2, resp. φ_3) encodes the problem of whether it is possible to go from *initial state* s_1 to *end state* s_4 in one step (resp. two steps, resp. three steps). By inspection, the transition system $\mathcal{M}$ in Figure 6 does not satisfy φ_1 and φ_2, while $\mathcal{M}$ does satisfy φ_3. More succintly, using quantifiers allowed by the syntax of QBF, it holds that:[12]

$$\Gamma \not\models (\exists q_1 \cdots q_4.\, \varphi_1), \quad \Gamma \not\models (\exists q_1 \cdots q_6.\, \varphi_2), \quad \text{and } \Gamma \models (\exists q_1 \cdots q_8.\, \varphi_3),$$

$$\text{where} \quad \Gamma \stackrel{\text{def}}{=} \Big\{ \forall p_1\, p_2.\, \mathsf{init}(p_1, p_2)\, , \forall p_1\, p_2.\, \mathsf{end}(p_1, p_2)\, , \forall p_1 \cdots p_4.\, \theta(p_1, p_2, p_3, p_4) \Big\}.$$

Things become more complicated when the transition relation expressed by θ has to account for many more states than only four in this example, or when the path from an *initial state* to an *end state* must satisfy some restriction. Exercises 26 and 27 pursue the analysis started in this example further. □

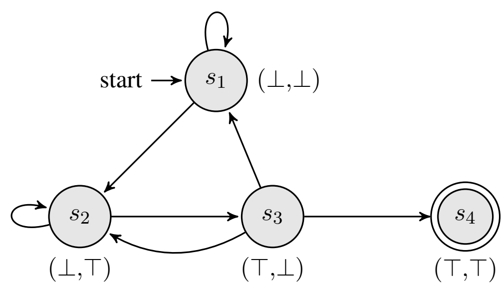

Figure 6: Graphical representation of a transition system with 4 states $\{s_1, s_2, s_3, s_4\}$ in Example 25, which can be encoded by 2-bit vectors $\{(\bot,\bot), (\bot,\top), (\top,\bot), (\top,\top)\}$, where s_1 is an *initial state* and s_4 is an *end state*.

Exercise 26 (*Reachability in Transition Systems*)**.** This is a continuation of the analysis in Example 25 in relation to the particular transition system in Figure 6. If we want to model reachability of s_4 from s_1 for some large number k of steps, the resulting wff will be unwieldy with k copies of θ as sub-wff's. A way out is to resort to a wff φ_k involving quantifiers and a single copy of θ, as follows:

$$\begin{aligned} \varphi_k \stackrel{\text{def}}{=} \exists\, & q_1\, q_2\, \cdots\, q_{2k+1}\, q_{2k+2}\, .\; \mathsf{init}(q_1, q_2) \;\wedge\; \mathsf{end}(q_{2k+1}, q_{2k+2}) \;\wedge \\ & \forall r_1\, r_2\, r_3\, r_4\, .\Big(\Big(\bigvee\nolimits_{0 \leqslant i \leqslant k-1} r_1 = q_{2i+1} \wedge r_2 = q_{2i+2} \wedge r_3 = q_{2i+3} \wedge r_4 = q_{2i+4}\Big) \;\to\; \theta(r_1, r_2, r_3, r_4)\Big) \end{aligned}$$

where "$p = q$" is an abbreviation for "$p \leftrightarrow q$", and "$p \leftrightarrow q$" is an abbreviation for "$(p \to q) \wedge (q \to p)$". Give a precise argument showing that φ_k correctly models reachability of s_4 from s_1 in k steps. □

Exercise 27 (*The Unwind Property in Transition Systems*)**.** A finite transition system $\mathcal{M}$ is said to have the *unwind property* if there is a natural number n such that every execution path from an *initial state* to an *end state* halts within at most n steps (or n single-edge transitions in the graph representation of $\mathcal{M}$).

As it stands, the system in Figure 6 does not have the unwind property. From the *initial state* s_1 to the *end state* s_4, there are arbitrarily long executions paths, thus preventing any "unwinding" or "unrolling" of the system into an equivalent and finite loop-free transition system.

We now consider operating the system under two separate restrictions (assumed to be enforced by mechanisms not mentioned in our definitions here):

(a) An execution path from s_1 to s_4 is *valid* provided each of the states in $\{s_1, s_2, s_3\}$ is visited an equal number $n \geqslant 1$ of times.

[12] It is tempting to replace "$\Gamma \not\models \ldots$" and "$\Gamma \models \ldots$" by "$\mathcal{M} \not\models \ldots$" and "$\mathcal{M} \models \ldots$", respectively. Although the intent is clear, the latter notation is not permitted by the definition of "$\models$" in the semantics of QBF. The set of wff's Γ completely captures the behavior of the transition system.

(b) An execution path from s_1 to s_4 is *valid* provided s_1 is visited at most $n \leqslant 2$ times, and each of s_2 and s_3 is visited $2n$ times.

There are four parts in this exercise. For the last two parts, you may find it helpful to do Exercise 26 first, taking advantage of the succintness that quantifiers allow in writing wff's:

1. Write the propositional wff θ_a which models the transition relation when the system operates under restriction (a).

   Hint: θ_a is a restriction of θ defined in Example 25. The requirement that all the states in $\{s_1, s_2, s_3\}$ are visited an equal number of times precludes the use of the self-loops around s_1 and s_2, as well as the loop "$s_2 \to s_3 \to s_2$", but not the loop "$s_1 \to s_2 \to s_3 \to s_1$".

2. Write the propositional wff θ_b which models the transition relation when the system operates under restriction (b).

   Hint: The requirement that s_2 and s_3 are visited an equal number times precludes the use of the self-loop around s_2, but not the use of the other loops.

3. Give a formal logic-based argument (no "hand waving") showing the transition system under restriction (a) does not have the *unwind property*. Thus, under restriction (a), the existence of arbitrarily long (finite) valid executions implies the system does not always halt, *i.e.*, there are non-terminating valid executions.

   Hint: Exhibit an infinite set Δ of QBF wff's expressing the existence of an infinite path starting at s_1 with infinitely many valid finite prefixes. Show that Δ is finitely satisfiable and invoke Compactness.

4. Give a precise argument showing that the transition system under restriction (b) does have the *unwind property*. Thus, under restriction (b), the system always halts and all valid execution paths are finite.

   Hint: This will be (mostly) a counting argument based on what restriction (b) entails. □

7 Concluding Remarks

What have we missed? Plenty.

The material we have covered in these lecture notes is but a tiny fraction of a much larger body of knowledge, which has developed into a sophisticated and very robust area of mathematics over more than a century. It can be presented from many equally valid angles. It is then unavoidable that many elegant arguments that you will find elsewhere are ignored from the perspective of these notes, which gives precedence to semantic notions over proof-theoretic notions.

Chief among these is perhaps Henkin's proof of Completeness, which bypasses Compactness to reach its end and, of course, makes the latter a corollary (provided Soundness is also available). Henkin's proof works like magic, from the syntactic raw material, it finds a way to build a model.[22]

(MORE TO COME)

[22]And it is a little surprising the first time you see it, as it gives you the feeling of a bootstrapping that may not work – or at least that was my reaction when I first encountered it as a student.

References

[1] J.L. Bell and A.B. Slomson. *Models and Ultraproducts: An Introduction*. North-Holland, 1974. 3

[2] CC Chang and H Jerome Keisler. *Model theory; 3rd ed.* Dover Books on Mathematics. Dover, New York, NY, 2012. 27

[3] F. Michel Dekking, Jeffrey Shallit, and N. J. A. Sloane. Queens in exile: non-attacking queens on infinite chess boards, 2019. preprint, `https://arxiv.org/abs/1907.09120`. 7

[4] Herbert B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, 2001. 1

[5] Michael Huth and Mark Ryan. *Logic in Computer Science: Modelling and Reasoning about Systems*. Cambridge University Press, 2 edition, 2004. 1, 1.2

[6] Georg Kreisel and Jean-Louis Krivine. *Elements of Mathematical Logic*. North-Holland, 1967. 2

[7] J. Donald Monk. *Mathematical Logic*. Springer-Verlag, 1976. 3

[8] Raymond M. Smullyan. *First-Order Logic*. Springer-Verlag, 1968. 2

[9] Terence Tao. The Completeness and Compactness Theorems of First-Order Logic, April 2009. Available here. 13

[10] Dirk van Dalen. *Logic and Structure, Third Edition*. Springer-Verlag, 1997. 3

A Syntax of Well-Formed Formulas

This appendix is a compendium of syntactic conventions we use in the main body of these lecture notes. It is intended as a handy reference, which can be quickly consulted whenever you need clarification on notations in the main body.

We first cover the syntax of *well-formed formulas* (wff's) of the following: *propositional logic*, the *logic of quantified boolean formulas*, and *first-order logic*. We thus define the sets $\mathsf{WFF}_{\mathsf{PL}}$, $\mathsf{WFF}_{\mathsf{QBF}}$ and $\mathsf{WFF}_{\mathsf{FOL}}$ first. The syntax of $\mathsf{WFF}_{\mathsf{FOL}}$ includes that of *zeroth-order logic*, *equality logic*, *equational logic*, and *quasi-equational logic*, as four special cases which are therefore left to the end of this appendix. The resulting sets of wff's are denoted $\mathsf{WFF}_{\mathsf{ZOL}}$, $\mathsf{WFF}_{\mathsf{eL}}$, $\mathsf{WFF}_{\mathsf{EL}}$, and $\mathsf{WFF}_{\mathsf{QEL}}$.

Throughout, we use lower-case Greek letters from the end of the alphabet (mostly φ and ψ) and occasionally from the beginning of the alphabet (α, β, and γ) as metavariables denoting well-formed formulas (wff's). We use upper-case Greek letters $\Gamma, \Delta, \ldots$ as metavariables denoting sets of wff's.

A.1 Well-Formed Formulas of PL

The syntax of *propositional logic* (PL) is built up from a set $\mathcal{P}$ of variables and a few logical connectives:

- $\mathcal{P} = \{p_0, p_1, \ldots\}$ is a countably infinite set of *propositional variables* (also called *propositional* or *Boolean atoms*). We use p and lower-case Roman letters nearby $\{q, r, s, \ldots\}$, possibly decorated, as metavariables ranging over $\mathcal{P}$.
- The set of *logical connectives* is $\{\neg, \wedge, \vee, \rightarrow\}$. We use the symbol "$\diamond$" as a metavariable ranging over the binary connectives $\wedge$, $\vee$, and $\rightarrow$.

The set $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ of well-formed propositional formulas over $\mathcal{P}$ is the least set such that:

$$\begin{aligned}\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P}) \;\supseteq\; &\mathcal{P} \,\cup\, \{\bot, \top\} \,\cup\, \left\{ (\neg\varphi) \;\middle|\; \varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P}) \right\} \\ &\cup\, \left\{ (\varphi \diamond \psi) \;\middle|\; \varphi, \psi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P}) \text{ and } \diamond \in \{\wedge, \vee, \rightarrow\} \right\}.\end{aligned}$$

It is customary to omit parentheses whenever possible, using the following precedences:

- $\{\neg\}$ binds more tightly than binary connectives $\{\wedge, \vee, \rightarrow\}$, *e.g.*, $\neg\varphi_1 \wedge \varphi_2$ means $((\neg\varphi_1) \wedge \varphi_2)$.
- binary connectives $\{\wedge, \vee\}$ associate to the left, *e.g.*, $\varphi_1 \wedge \varphi_2 \wedge \varphi_3$ means $((\varphi_1 \wedge \varphi_2) \wedge \varphi_3)$.
- the binary connective $\{\rightarrow\}$ associates to the right, *e.g.*, $\varphi_1 \rightarrow \varphi_2 \rightarrow \varphi_3$ means $(\varphi_1 \rightarrow (\varphi_2 \rightarrow \varphi_3))$.
- $\{\wedge, \vee\}$ have higher precedence than $\{\rightarrow\}$, *e.g.*, $\varphi_1 \rightarrow \varphi_2 \wedge \varphi_3$ means $(\varphi_1 \rightarrow (\varphi_2 \wedge \varphi_3))$.

Whenever in doubt about the conventions, insert matching parentheses to disambiguate wff's. Also, to break precedences of logical connectives, insert parentheses; for example, if the intended wff is $((\varphi_1 \rightarrow \varphi_2) \rightarrow \varphi_3)$, we can omit the outer matching parentheses as in $(\varphi_1 \rightarrow \varphi_2) \rightarrow \varphi_3$, but not the inner matching parentheses as in $\varphi_1 \rightarrow \varphi_2 \rightarrow \varphi_3$, otherwise the wff is understood to mean $(\varphi_1 \rightarrow (\varphi_2 \rightarrow \varphi_3))$.

Exercise 67. There is an implicit induction in our definitions of $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ above. Make this induction explicit in an alternative definition of $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$, using BNF or extended BNF notation. $\square$

A.2 Well-Formed Formulas of QBF

The *logic of quantified boolean formulas* (QBF) extends PL by introducing the quantifiers $\{\forall, \exists\}$. The set $\mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$ of well-formed formulas of QBF over $\mathcal{P}$ is the least set such that:

$$\begin{aligned}\mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P}) \supseteq\ & \mathcal{P} \cup \{\bot, \top\} \cup \left\{ (\neg\varphi) \,\middle|\, \varphi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P}) \right\} \\ & \cup \left\{ (\varphi \diamond \psi) \,\middle|\, \varphi, \psi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P}) \text{ and } \diamond \in \{\wedge, \vee, \to\} \right\} \\ & \cup \left\{ (\forall p\, \varphi) \,\middle|\, \varphi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P}) \text{ and } p \in \mathcal{P} \right\} \cup \left\{ (\exists p\, \varphi) \,\middle|\, \varphi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P}) \text{ and } p \in \mathcal{P} \right\}.\end{aligned}$$

To omit parentheses for better readability, we use the same precedences as in PL, in addition to the following conventions for quantifiers:

- $\forall p.\, \varphi$ means $(\forall p\, \varphi)$ and $\exists p.\, \varphi$ means $(\exists p\, \varphi)$.
- $\forall p\, q.\, \varphi$ means $(\forall p\, (\forall q\, \varphi))$ and $\exists p\, q.\, \varphi$ means $(\exists p\, (\exists q\, \varphi))$.

A.3 Well-Formed Formulas of FOL

Let $\Sigma = \mathcal{R} \cup \mathcal{F} \cup \mathcal{C}$ be a first-order signature, where:

- $\mathcal{R} = \{R_1, R_2, \ldots\}$ is a countable set, possibly infinite, of relation symbols, each with an arity $\geqslant 1$. We use R and upper-case Roman letters nearby $\{P, Q, S, \ldots\}$, possibly decorated, as metavariables ranging over $\mathcal{R}$.[23]
- $\mathcal{F} = \{f_1, f_2, \ldots\}$ is a countable set, possibly infinite, of function symbols, each with an arity $\geqslant 1$. We use f and lower-case Roman letters nearby $\{f, g, h, \ldots\}$, possibly decorated, as metavariables ranging over $\mathcal{F}$.
- $\mathcal{C} = \{c_1, c_2, \ldots\}$ is a countable set, possible infinite, of constant symbols. We use c and lower-case Roman letters nearby $\{d, e, \ldots\}$, possibly decorated, as metavariables ranging over $\mathcal{C}$.

Following most presentations of *first-order logic* (FOL), wff's may include a symbol for the equality relation, which is here denoted "$\approx$" and used in infix position, as in "$t_1 \approx t_2$". We consider the symbol "$\approx$" to be outside the signature Σ.

Besides symbols from the signature, wff's of FOL may contain variables:

- $X = \{x_0, x_1, x_2, \ldots\}$ is a countably infinite set of variables. We use letters from the end of the Roman alphabet $\{x, y, z, \ldots\}$, possibly decorated, as metavariables ranging over X.

We build up wff's gradually, starting with the set of terms $\mathsf{Terms}(\Sigma, X)$, followed by the set of atomic formulas $\mathsf{Atoms}(\Sigma, X)$, followed by the full set $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ of wff's. These are the three stages:

1. $\mathsf{Terms}(\Sigma, X)$ is the least set satisfying the condition:

$$\mathsf{Terms}(\Sigma, X) \supseteq \mathcal{C} \cup X \cup \left\{ f(t_1, \ldots, t_n) \,\middle|\, f \in \mathcal{F} \text{ has arity } n \geqslant 1,\ t_1, \ldots, t_n \in \mathsf{Terms}(\Sigma, X) \right\}.$$

   Since there are no relation symbols in terms, we may write $\mathsf{Terms}(\mathcal{F} \cup \mathcal{C}, X)$ instead of $\mathsf{Terms}(\Sigma, X)$.

[23] Some authors prefer the words "predicate" and "predicate symbol" to what we call "relation" and "relation symbol".

2. $\mathsf{Atoms}(\Sigma, X)$ is the set defined by:

$$\mathsf{Atoms}(\Sigma, X) \stackrel{\text{def}}{=} \{\,\bot, \top\,\} \cup \left\{\, R(t_1, \ldots, t_n) \;\middle|\; R \in \mathcal{R} \text{ has arity } n \geqslant 0,\ t_1, \ldots, t_n \in \mathsf{Terms}(\Sigma, X) \,\right\}.$$

3. $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ is the least set satisfying the condition:

$$\begin{aligned}\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X) \supseteq\ & \mathsf{Atoms}(\Sigma, X) \cup \left\{\, (\neg\varphi) \;\middle|\; \varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X) \,\right\} \\ & \cup \left\{\, (\varphi \diamond \psi) \;\middle|\; \varphi, \psi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X) \text{ and } \diamond \in \{\wedge, \vee, \rightarrow\} \,\right\} \\ & \cup \left\{\, (\forall x\, \varphi) \;\middle|\; \varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X) \text{ and } x \in X \,\right\} \\ & \cup \left\{\, (\exists x\, \varphi) \;\middle|\; \varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X) \text{ and } x \in X \,\right\}.\end{aligned}$$

We follow standard practice of omitting parentheses whenever possible, using the following conventions, which extend those already mentioned for PL and QBF:

- $\{\neg\}$ binds more tightly than binary connectives $\{\wedge, \vee, \rightarrow\}$, *e.g.*, $\neg\varphi_1 \wedge \varphi_2$ means $((\neg\varphi_1) \wedge \varphi_2)$.
- binary connectives $\{\wedge, \vee\}$ associate to the left, *e.g.*, $\varphi_1 \wedge \varphi_2 \wedge \varphi_3$ means $((\varphi_1 \wedge \varphi_2) \wedge \varphi_3)$.
- binary connective $\{\rightarrow\}$ associates to the right, *e.g.*, $\varphi_1 \rightarrow \varphi_2 \rightarrow \varphi_3$ means $(\varphi_1 \rightarrow (\varphi_2 \rightarrow \varphi_3))$.
- $\{\wedge, \vee\}$ have higher precedence than $\{\rightarrow\}$, *e.g.*, $\varphi_1 \rightarrow \varphi_2 \wedge \varphi_3$ means $(\varphi_1 \rightarrow (\varphi_2) \wedge \varphi_3))$.
- $\forall x.\, \varphi$ means $(\forall x\, \varphi)$ and $\exists x.\, \varphi$ means $(\exists x\, \varphi)$.
- $\forall x\, y.\, \varphi$ means $(\forall x\, (\forall y\, \varphi))$ and $\exists x\, y.\, \varphi$ means $(\exists x\, (\exists y\, \varphi))$.

Whenever in doubt about the conventions, insert matching parentheses to disambiguate the formula.

Exercise 68. There is an implicit induction in our definitions of $\mathsf{Terms}(\Sigma, X)$ and $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ above. Make this induction explicit in alternative definitions of $\mathsf{Terms}(\Sigma, X)$ and $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$, using BNF or extended BNF notation. $\square$

If we expand the signature Σ with fresh function symbols or relation symbols, then the sets $\mathsf{Terms}(\Sigma, X)$, $\mathsf{Atoms}(\Sigma, X)$, and $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ are extended in the obvious way. For example, If we introduce a new relation symbol $R \notin \mathcal{R}$ of some arity $n \geqslant 0$, the set $\mathsf{Atoms}(\Sigma, X)$ is extended as follows:

$$\mathsf{Atoms}\big(\Sigma \cup \{R\}, X\big) \stackrel{\text{def}}{=} \mathsf{Atoms}(\Sigma, X) \cup \{\, R(t_1, \ldots, t_n) \mid t_1, \ldots, t_n \in \mathsf{Terms}(\Sigma, X) \,\}.$$

and the set $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ is extended to $\mathsf{WFF}_{\mathsf{FOL}}\big(\Sigma \cup \{R\}, X\big)$, defined by substituting $\mathsf{Atoms}\big(\Sigma \cup \{R\}, X\big)$ for $\mathsf{Atoms}(\Sigma, X)$ in the definition of $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$.

An important instance of the preceding is when we allow the equality symbol "$\approx$" to occur in wff's. In this case the set of atomic wff's is now denoted $\mathsf{Atoms}\big(\Sigma \cup \{\approx\}, X\big)$ and the corresponding $\mathsf{WFF}_{\mathsf{FOL}}\big(\Sigma \cup \{\approx\}, X\big)$ is obtained by substituting $\mathsf{Atoms}\big(\Sigma \cup \{\approx\}, X\big)$ for $\mathsf{Atoms}(\Sigma, X)$ in the definition of $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$.

A.4 Well-Formed Formulas of ZOL

We write $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma, \varnothing)$ for the set of wff's of *zeroth-order logic* (ZOL), a proper subset of $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ that mention no variables in X and no quantifiers in $\{\forall, \exists\}$. The definition is in three stages:

1. $\mathsf{Terms}(\Sigma,\varnothing)$ is the same as the set of variable-free terms of FOL:

$$\mathsf{Terms}(\Sigma,\varnothing) \;\stackrel{\text{def}}{=}\; \Big\{\, t \;\Big|\; t\in\mathsf{Terms}(\Sigma,X) \text{ and } \mathrm{FV}(t)=\varnothing \,\Big\}.$$

   Since there are no relation symbols in terms, we may write $\mathsf{Terms}(\mathcal{F}\cup\mathcal{C},\varnothing)$ instead of $\mathsf{Terms}(\Sigma,\varnothing)$.

2. $\mathsf{Atoms}(\Sigma,\varnothing)$ is the same as the set of variable-free atomic formulas of FOL:

$$\mathsf{Atoms}(\Sigma,\varnothing) \;\stackrel{\text{def}}{=}\; \{\bot,\top\} \,\cup\, \Big\{\, \varphi \;\Big|\; \varphi\in\mathsf{Atoms}(\Sigma,X) \text{ and } \mathrm{FV}(\varphi)=\varnothing \,\Big\}.$$

3. $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,\varnothing)$ is the least set satisfying the condition:

$$\begin{aligned}\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,\varnothing) \;\supseteq\; &\mathsf{Atoms}(\Sigma,\varnothing) \,\cup\, \Big\{\, (\neg\varphi) \;\Big|\; \varphi\in\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,\varnothing) \,\Big\}\\ &\cup\, \Big\{\, (\varphi\diamond\psi) \;\Big|\; \varphi,\psi\in\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,\varnothing) \text{ and } \diamond\in\{\wedge,\vee,\to\} \,\Big\}.\end{aligned}$$

In words, $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,\varnothing)$ is the set of all variable-free and quantifier-free formulas of *first-order logic* over signature Σ.

It is possible to define a *zeroth-order logic* which allows variables but disallows quantifiers. The set of wff's of such a logic is $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,X)$, indicated by the second argument $X\neq\varnothing$. $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,X)$ is intermediate between $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,\varnothing)$ and $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma,X)$, more expressive than $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,\varnothing)$ but less expressive than $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma,X)$. For all of its interesting properties, we do not examine $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma,X)$ in these lecture notes.

Of particular interest for our presentation is the extension $\mathsf{WFF}_{\mathsf{ZOL}}\big(\Sigma\cup\{\approx\},\varnothing\big)$ which allows the equality symbol "$\approx$" to occur in wff's, but still precludes variables and quantifiers. See our examination in Section 4.

A.5 Well-Formed Formulas of eL, EL, and QEL

We write:

- $\mathsf{WFF}_{\mathsf{eL}}\big(\{\approx\},X\big)$ for the set of wff's of *equality logic* (eL),
- $\mathsf{WFF}_{\mathsf{EL}}\big(\mathcal{F}\cup\mathcal{C}\cup\{\approx\},X\big)$ for the set of wff's of *equational logic* (EL),
- $\mathsf{WFF}_{\mathsf{QEL}}\big(\mathcal{F}\cup\mathcal{C}\cup\{\approx\},X\big)$ for the set of wff's of *quasi-equational logic* (QEL),

which are all proper subsets of $\mathsf{WFF}_{\mathsf{FOL}}\big(\Sigma\cup\{\approx\},X\big)$, the set of first-order wff's where the equality symbol $\approx$ may occur. In these three logics, relation symbols are precluded from wff's.[24]

Starting with $\mathsf{WFF}_{\mathsf{eL}}\big(\{\approx\},X\big)$, the definition is once more in three stages:

1. $\mathsf{Terms}\big(\varnothing,X\big)$ is simply the set of all variables, as no symbols from the signature are allowed:

$$\mathsf{Terms}\big(\varnothing,X\big) \;\stackrel{\text{def}}{=}\; X.$$

2. $\mathsf{Atoms}\big(\{\approx\},X\big)$ is restricted to the equality symbol "$\approx$" and its members are called *equalities* (between first-order variables):

$$\mathsf{Atoms}\big(\{\approx\},X\big) \;\stackrel{\text{def}}{=}\; \{\bot,\top\} \,\cup\, \Big\{\, (x\approx y) \;\Big|\; x,y\in X \,\Big\}.$$

[24] The distinction between "*equality*" and "*equation*" is a little confusing and does not conform to how we use the same words in other contexts. We use two different words here so that we can name differently two distinct logics, eL and EL. To confuse the matter a little more, what we here call *equations* and *quasi-equations* are called elsewhere *identities* and *quasi-identities*.

3. $\mathsf{WFF}_{\mathsf{eL}}\big(\{\approx\}, X\big)$ is the least set satisfying the condition:

$$\mathsf{WFF}_{\mathsf{eL}}\big(\{\approx\}, X\big) \;\supseteq\; \mathsf{Atoms}(\{\approx\}, X) \;\cup\; \Big\{\, (\neg\varphi) \;\Big|\; \varphi \in \mathsf{WFF}_{\mathsf{eL}}(\{\approx\}, X) \,\Big\}$$
$$\cup\; \Big\{\, (\varphi \diamond \psi) \;\Big|\; \varphi, \psi \in \mathsf{WFF}_{\mathsf{eL}}(\{\approx\}, X) \text{ and } \diamond \in \{\wedge, \vee, \to\} \,\Big\}.$$

In words, $\mathsf{WFF}_{\mathsf{eL}}(\{\approx\}, X)$ is the set of all (quantifier-free) propositional combinations of equalities between variables.

We define $\mathsf{WFF}_{\mathsf{EL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$ and $\mathsf{WFF}_{\mathsf{QEL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$ simultaneously in three stages:

1. $\mathsf{Terms}\big(\mathcal{F} \cup \mathcal{C}, X\big)$ is the same as the set of terms of FOL, the least satisfying the condition:

$$\mathsf{Terms}(\mathcal{F} \cup \mathcal{C}, X) \;\supseteq\; \mathcal{C} \cup X \cup \Big\{\, f(t_1, \ldots, t_n) \;\Big|\; f \in \mathcal{F} \text{ has arity } n \geqslant 1,\; t_1, \ldots, t_n \in \mathsf{Terms}(\Sigma, X) \,\Big\}.$$

2. $\mathsf{Atoms}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$ is restricted to the equality symbol "$\approx$" and its members are called *equations* (between terms):

$$\mathsf{Atoms}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big) \;\stackrel{\text{def}}{=}\; \{\bot, \top\} \;\cup\; \Big\{\, (t_1 \approx t_2) \;\Big|\; t_1, t_2 \in \mathsf{Terms}\big(\mathcal{F} \cup \mathcal{C}, X\big) \,\Big\}.$$

3. $\mathsf{WFF}_{\mathsf{EL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$ and $\mathsf{WFF}_{\mathsf{QEL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$ are defined by:

$$\mathsf{WFF}_{\mathsf{EL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big) \;\stackrel{\text{def}}{=}\; \mathsf{Atoms}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$$

$$\mathsf{WFF}_{\mathsf{QEL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big) \;\stackrel{\text{def}}{=}\; \Big\{\, (\varphi_1 \wedge \cdots \wedge \varphi_k \to \psi) \;\Big|\; \varphi_1, \ldots, \varphi_k, \psi \in \mathsf{Atoms}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big) \text{ and } k \geqslant 0 \,\Big\}.$$

We call *equations* and *quasi-equations* the wff's in $\mathsf{WFF}_{\mathsf{EL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$ and $\mathsf{WFF}_{\mathsf{QEL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$, respectively. Every equation is a quasi-equation, but not conversely, so that $\mathsf{WFF}_{\mathsf{EL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$ is a proper subset of $\mathsf{WFF}_{\mathsf{QEL}}\big(\mathcal{F} \cup \mathcal{C} \cup \{\approx\}, X\big)$.

In these lecture notes, every *equality* is an *equation*, but not vice-versa. Moreover, although every equality is an equation, it is not the case that every wff of *equality logic* (eL) is a wff of *equational logic* (EL); wff's of eL can include all the logical connectives in $\{\neg, \wedge, \vee, \to\}$, wff's of EL do not mention any of these logical connectives.

B Semantics of Well-Formed Formulas

We try as much as possible to give a uniform presentation of the semantics of all the formal logics considered in these lecture notes. We start with the semantics of *propositional logic*, then follow with the semantics of *quantified Boolean formulas*, and then with the semantics of *first-order logic*. The semantics of the other logics are special cases of the semantics of *first-order logic* and do not need a separate treatment.

B.1 Semantics of $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$

We interpret the wff's of *propositional logic* in a 2-element Boolean algebra $\mathcal{B}$, which we can take in the form:

$$\mathcal{B} \stackrel{\text{def}}{=} (B, \mathsf{Not}, \mathsf{And}, \mathsf{Or}, \mathsf{Implies}, \mathbf{F}, \mathbf{T}) \qquad \text{with } B \stackrel{\text{def}}{=} \{\mathbf{F}, \mathbf{T}\},$$

where Not is a unary operation, and each of the operations in $\{\mathsf{And}, \mathsf{Or}, \mathsf{Implies}\}$ is binary. Since the domain B is finite, these operations can be conveniently defined in tabular forms as follows:[25]

	Not
F	**T**
T	**F**

And	**F**	**T**
F	**F**	**F**
T	**F**	**T**

Or	**F**	**T**
F	**F**	**T**
T	**T**	**T**

Implies	**F**	**T**
F	**T**	**T**
T	**F**	**T**

A *truth assignment* for the set $\mathcal{P}$ of propositional variables is any map $\sigma : \mathcal{P} \to \{\mathbf{F}, \mathbf{T}\}$. Having fixed the interpretations of the symbols $\{\neg, \wedge, \vee, \to\}$ as the operations $\{\mathsf{Not}, \mathsf{And}, \mathsf{Or}, \mathsf{Implies}\}$ of the Boolean algebra $\mathcal{B}$, the satisfaction of a propositional wff φ, *i.e.*, the truth value of φ, depends only on the assignment σ.

We next lift the truth assignment σ to all propositional wff's. We use a notation favored by computer scientists: the meaning of a syntactic object φ is denoted by inserting it between double brackets, as in "$\llbracket\varphi\rrbracket$" or, more precisely here, "$\llbracket\varphi\rrbracket_\sigma$" since it depends on σ. The definition of $\llbracket\varphi\rrbracket_\sigma$ is by *structural induction*, *i.e.*, on the "shape" of φ:[26]

$$\begin{aligned}
\llbracket p\rrbracket_\sigma &\stackrel{\text{def}}{=} \sigma(p)\\
\llbracket \bot\rrbracket_\sigma &\stackrel{\text{def}}{=} \mathbf{F}\\
\llbracket \top\rrbracket_\sigma &\stackrel{\text{def}}{=} \mathbf{T}\\
\llbracket \neg\varphi\rrbracket_\sigma &\stackrel{\text{def}}{=} \mathsf{Not}\big(\llbracket\varphi\rrbracket_\sigma\big)\\
\llbracket \varphi\wedge\psi\rrbracket_\sigma &\stackrel{\text{def}}{=} \mathsf{And}\big(\llbracket\varphi\rrbracket_\sigma, \llbracket\psi\rrbracket_\sigma\big)\\
\llbracket \varphi\vee\psi\rrbracket_\sigma &\stackrel{\text{def}}{=} \mathsf{Or}\big(\llbracket\varphi\rrbracket_\sigma, \llbracket\psi\rrbracket_\sigma\big)\\
\llbracket \varphi\to\psi\rrbracket_\sigma &\stackrel{\text{def}}{=} \mathsf{Implies}\big(\llbracket\varphi\rrbracket_\sigma, \llbracket\psi\rrbracket_\sigma\big)
\end{aligned}$$

[25] These are not what are usually called the *truth-tables* of the Boolean operations, which are typically written as:

p	$\neg p$
F	**T**
T	**F**

p	q	$p \wedge q$
F	**F**	**F**
F	**T**	**F**
T	**F**	**F**
T	**T**	**T**

p	q	$p \vee q$
F	**F**	**F**
F	**T**	**T**
T	**F**	**T**
T	**T**	**T**

p	q	$p \to q$
F	**F**	**T**
F	**T**	**T**
T	**F**	**F**
T	**T**	**T**

Our tabular forms for the Boolean operations here is the same tabular forms we use elsewhere in these notes whenever we deal with unary and binary operations over finite domains. In particular for binary operations, they are more compact than truth-tables, but their generalization to higher-arity functions lose their graphical appeal and are practically useless.

[26] Or, as computer scientists often like to say, the definition is *syntax-directed*.

Following convention:

- we write $\sigma \models \varphi$ and say σ *satisfies* φ iff $[\![\varphi]\!]_\sigma = \mathsf{T}$,
- we write $\sigma \not\models \varphi$ and say σ *does not satisfy* φ iff $[\![\varphi]\!]_\sigma = \mathsf{F}$,
- if for every assignment σ we have $\sigma \models \varphi$, we may write $\models \varphi$ and say φ is *valid* or is a *tautology*,
- if for some assignment σ we have $\sigma \not\models \varphi$, we may write $\not\models \varphi$ and say φ is *falsifiable*,
- if for every assignment σ we have $\sigma \not\models \varphi$, we may say φ is *unsatisfiable* or is a *contradiction*.

It is worth noting that the double-bracket notation is a convenient visual aid to separate syntax from semantics: everything inside the pair "$[\![$" and "$]\!]$" is a piece of syntax, and everything outside is about its semantics. Similarly, "$\models$" conveniently separates syntax from semantics: what is to the right of "$\models$" is a piece of syntax and what is to the left of "$\models$" is something that determines the semantics of the former. These are by now firmly established notational conventions.[27]

B.2 Semantics of $\mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$

Consider the definition of $[\![\varphi]\!]_\sigma$ by *structural induction* when $\varphi \in \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$. We want to extend it to wff's in $\mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$ which mention the quantifiers $\forall$ and $\exists$. There are two steps that are missing for this extension and, before supplying them, we agree on how to write a *(one-point) adjustment* of a truth assignment σ. If $p \in \mathcal{P}$, an adjustment of σ at p is denoted "$\sigma[p \mapsto \mathsf{F}]$" or "$\sigma[p \mapsto \mathsf{T}]$". The precise definition is, for all $q \in \mathcal{P}$:

$$\sigma[p \mapsto \mathsf{F}]\ (q) \ \stackrel{\text{def}}{=} \ \begin{cases} \mathsf{F} & \text{if } p = q, \\ \sigma(q) & \text{if } p \neq q, \end{cases}$$

$$\sigma[p \mapsto \mathsf{T}]\ (q) \ \stackrel{\text{def}}{=} \ \begin{cases} \mathsf{T} & \text{if } p = q, \\ \sigma(q) & \text{if } p \neq q. \end{cases}$$

Now for the two missing steps in the structural induction:

$$[\![\forall p\, \varphi]\!]_\sigma \ \stackrel{\text{def}}{=} \ \mathsf{And}\left([\![\varphi]\!]_{\sigma[p\mapsto\mathsf{F}]}, [\![\varphi]\!]_{\sigma[p\mapsto\mathsf{T}]}\right)$$

$$[\![\exists p\, \varphi]\!]_\sigma \ \stackrel{\text{def}}{=} \ \mathsf{Or}\left([\![\varphi]\!]_{\sigma[p\mapsto\mathsf{F}]}, [\![\varphi]\!]_{\sigma[p\mapsto\mathsf{T}]}\right)$$

Exercise 69. Based on the preceding definition, show that the following are equivalent assertions:

- $\sigma \ \models \ \forall p\, \varphi$,
- $\sigma \ \models \ \varphi[p := \bot] \ \wedge \ \varphi[p := \top]$.

And similarly, show that the following are equivalent assertions:

- $\sigma \ \models \ \exists p\, \varphi$,
- $\sigma \ \models \ \varphi[p := \bot] \ \vee \ \varphi[p := \top]$.

We write $\varphi[p := \bot]$ and $\varphi[p := \top]$ to denote the substitution of $\bot$ and $\top$ for every free occurrence of p in φ. $\square$

[27] The double-bracket notation "$[\![\ldots]\!]$" was probably first used by computer scientists working on the denotational semantics of programming languages in the early 1970's. The double-turnstile notation "$\models$" was first introduced by mathematical logicians at least a decade earlier. The symbol "$\models$" appears throughout the classic book *Model Theory* by C.C. Chang and H. Jerome Keisler [2]; the authors point out, in the Preface of the first edition, that their book grew out of lecture notes in circulation since the early 1960's.

Exercise 70. A wff $\varphi \in \mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$ is *closed* if $\mathrm{FV}(\varphi) = \varnothing$, *i.e.*, every occurrence of a variable p in φ falls in the scope of some "$\forall p$" or "$\exists p$". Use structural induction to show that, if φ is closed, then for every assignment σ it holds that either $\sigma \models \varphi$ or $\sigma \not\models \varphi$.

In words, if φ is closed, then φ is either a tautology or a contradiction.

Hint: This is subtle. In the structural induction, keep track of variables that occur free in a wff, there are finitely many of them in any wff. Formalize the idea that only a finite part of an assignment σ is relevant for the truth-value returned by $[\![\varphi]\!]_\sigma$, namely, the part that assigns a truth value to a variable occurring free in φ. $\square$

B.3 Semantics of the Other Logics

All the other logics considered in these lecture notes are: *equality logic*, *zeroth-ary logic*, *equational logic*, *quasi-equational logic*, and *first-order logic*. The first four in this list are fragments of the last one, *first-order logic*. So, we restrict attention to the semantics of $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ and $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma \cup \{\approx\}, X)$.

Given signature $\Sigma = \mathcal{R} \cup \mathcal{F} \cup \mathcal{C}$, a Σ-*structure* has the form:

$\mathcal{A} \stackrel{\text{def}}{=} (A, \mathcal{R}^{\mathcal{A}}, \mathcal{F}^{\mathcal{A}}, \mathcal{C}^{\mathcal{A}})$ where

A is a non-empty set, finite or infinite, called the *universe* or *domain* of $\mathcal{A}$,

$$\begin{aligned}
\mathcal{R}^{\mathcal{A}} &\stackrel{\text{def}}{=} \{\, R^{\mathcal{A}} \subseteq \underbrace{A \times \cdots \times A}_{n} \mid R \in \mathcal{R} \text{ has arity } n \geqslant 1 \,\},\\
\mathcal{F}^{\mathcal{A}} &\stackrel{\text{def}}{=} \{\, f^{\mathcal{A}} : \underbrace{A \times \cdots \times A}_{n} \to A \mid f \in \mathcal{F} \text{ has arity } n \geqslant 1 \,\},\\
\mathcal{C}^{\mathcal{A}} &\stackrel{\text{def}}{=} \{\, c^{\mathcal{A}} \in A \mid c \in \mathcal{C} \,\}.
\end{aligned}$$

In words, a Σ-*structure* $\mathcal{A}$ assigns an interpretation to every symbol in Σ over some set of elements A. If the equality symbol $\approx$ occurs in wff's, we need to expand $\mathcal{A}$ to include an interpretation for it and write:

$$\mathcal{A} \stackrel{\text{def}}{=} (A, \approx^{\mathcal{A}}, \mathcal{R}^{\mathcal{A}}, \mathcal{F}^{\mathcal{A}}, \mathcal{C}^{\mathcal{A}}) \text{ or also } (A, =, \mathcal{R}^{\mathcal{A}}, \mathcal{F}^{\mathcal{A}}, \mathcal{C}^{\mathcal{A}})$$

since $\approx^{\mathcal{A}}$ is always interpreted as the equality "$=$" on the universe A.

A *valuation* for the set X of variables in the Σ-*structure* $\mathcal{A}$ is a map $\sigma : X \to A$. Note that σ maps every member of X, which is an infinite set, to an element of A. In case A is finite, σ necessarily maps many members of X to the same element of A.[28] A *(one-point) adjustment* of σ at variable x is a new valuation denoted $\sigma[x \mapsto a]$ where $a \in A$:

$$\sigma[x \mapsto a]\ (y) \stackrel{\text{def}}{=} \begin{cases} a & \text{if } x = y,\\ \sigma(y) & \text{if } x \neq y. \end{cases}$$

We use a Σ-structure $\mathcal{A}$ together with a valuation σ to give a meaning to every $\varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma \cup \{\approx\}, X)$. Below is the *structural induction* we use to interpret every such wff, it includes some of the steps already used for wff's in $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$ and $\mathsf{WFF}_{\mathsf{QBF}}(\mathcal{P})$. Following the three stages in the definition of $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma \cup \{\approx\}, X)$, we start with the interpretation of *terms*, continue with the interpretation of *atomic wff's*, and conclude with the interpretation of first-order wff's in general:

[28] Some authors call a valuation an *assignment* or also an *environment*. We reserve the word "assignment" to appear in the expessions "truth assignment" and "assignment of truth values", a way of distinguishing it from what the word "valuation" is used for.

1. Interpretation of terms:

$$
\begin{aligned}
[\![x]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \sigma(x) \\
[\![c]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} c^{\mathcal{A}} && \text{for every } c \in \mathcal{C} \\
[\![f(t_1,\ldots,t_n)]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} f^{\mathcal{A}}\big([\![t_1]\!]_{\mathcal{A},\sigma},\ldots,[\![t_n]\!]_{\mathcal{A},\sigma}\big) && \text{for every } f \in \mathcal{F} \text{ of arity } n \geqslant 1
\end{aligned}
$$

2. Interpretation of atomic wff's:

$$
\begin{aligned}
[\![\bot]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \mathsf{F} \\
[\![\top]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \mathsf{T} \\
[\![R(t_1,\ldots,t_n)]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} R^{\mathcal{A}}\big([\![t_1]\!]_{\mathcal{A},\sigma},\ldots,[\![t_n]\!]_{\mathcal{A},\sigma}\big) && \text{for every } R \in \mathcal{R} \text{ of arity } n \geqslant 1 \\
[\![t_1 \approx t_n]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \begin{cases} \mathsf{F} & \text{if } [\![t_1]\!]_{\mathcal{A},\sigma} \neq [\![t_2]\!]_{\mathcal{A},\sigma} \\ \mathsf{T} & \text{if } [\![t_1]\!]_{\mathcal{A},\sigma} = [\![t_2]\!]_{\mathcal{A},\sigma} \end{cases}
\end{aligned}
$$

3. Interpretation of first-order wff's in general:

$$
\begin{aligned}
[\![\neg\varphi]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \mathsf{Not}\big([\![\varphi]\!]_{\mathcal{A},\sigma}\big) \\
[\![\varphi \wedge \psi]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \mathsf{And}\big([\![\varphi]\!]_{\mathcal{A},\sigma}, [\![\psi]\!]_{\mathcal{A},\sigma}\big) \\
[\![\varphi \vee \psi]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \mathsf{Or}\big([\![\varphi]\!]_{\mathcal{A},\sigma}, [\![\psi]\!]_{\mathcal{A},\sigma}\big) \\
[\![\varphi \to \psi]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \mathsf{Implies}\big([\![\varphi]\!]_{\mathcal{A},\sigma}, [\![\psi]\!]_{\mathcal{A},\sigma}\big) \\
[\![\forall x\,\varphi]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \begin{cases} \mathsf{F} & \text{if } [\![\varphi]\!]_{\mathcal{A},\sigma[x\mapsto a]} = \mathsf{F} \text{ for some } a \in A \\ \mathsf{T} & \text{if } [\![\varphi]\!]_{\mathcal{A},\sigma[x\mapsto a]} = \mathsf{T} \text{ for every } a \in A \end{cases} \\
[\![\exists x\,\varphi]\!]_{\mathcal{A},\sigma} &\stackrel{\text{def}}{=} \begin{cases} \mathsf{F} & \text{if } [\![\varphi]\!]_{\mathcal{A},\sigma[x\mapsto a]} = \mathsf{F} \text{ for every } a \in A \\ \mathsf{T} & \text{if } [\![\varphi]\!]_{\mathcal{A},\sigma[x\mapsto a]} = \mathsf{T} \text{ for some } a \in A \end{cases}
\end{aligned}
$$

We can view $[\![\ldots]\!]_{\mathcal{A},\sigma}$ as a two-sorted function, from a two-sorted domain $\{\text{terms}\} \cup \{\text{wff's}\}$ to a two-sorted co-domain $A \cup \{\mathsf{F},\mathsf{T}\}$: It maps every term t to an element in the universe A, and every wff φ to a truth value.

Following convention:

- we write $\mathcal{A},\sigma \models \varphi$ and say $(\mathcal{A},\sigma)$ *satisfies* φ iff $[\![\varphi]\!]_{\mathcal{A},\sigma} = \mathsf{T}$,
- we write $\mathcal{A},\sigma \not\models \varphi$ and say $(\mathcal{A},\sigma)$ *does not satisfy* φ iff $[\![\varphi]\!]_{\mathcal{A},\sigma} = \mathsf{F}$,
- if for every valuation σ we have $\mathcal{A},\sigma \models \varphi$, we write $\mathcal{A} \models \varphi$ and say $\mathcal{A}$ *satisfies* φ or φ is *true in* $\mathcal{A}$,
- if for every Σ-structure $\mathcal{A}$ and valuation σ we have $\mathcal{A},\sigma \models \varphi$, we write $\models \varphi$ and say φ is *valid*.

Exercise 71. This continues Exercise 70. Let $\mathcal{A}$ be a fixed Σ-structure. Use structural induction to show that, if $\varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma, X)$ is closed, then for every valuation σ it holds that either $\mathcal{A},\sigma \models \varphi$ or $\mathcal{A},\sigma \not\models \varphi$.

Hence, when φ is closed, we can ignore σ and write $\mathcal{A} \models \varphi$ ("φ is true in $\mathcal{A}$") or $\mathcal{A} \not\models \varphi$ ("φ is false in $\mathcal{A}$"). $\square$

Exercise 72. Let $\varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma \cup \{\approx\}, X)$ and $\mathrm{FV}(\varphi) = \{x_1, \ldots, x_n\}$. The *existential closure* of φ is the closed wff $\exists x_1 \cdots \exists x_n.\, \varphi$ and the *universal closure* of φ is the closed wff $\forall x_1 \cdots \forall x_n.\, \varphi$.

1. Show that φ is satisfiable iff the existential closure of φ is satisfiable.
2. Show that φ is valid iff the universal closure of φ is valid. $\square$

Let Γ and Δ be sets, possibly infinite, of wff's in $\mathsf{WFF}_{\mathsf{FOL}}(\Sigma \cup \{\approx\}, X)$.

- We say a Σ-structure $\mathcal{A}$ is a *model* of Γ iff for every $\varphi \in \Gamma$ it holds that $\mathcal{A} \models \varphi$, in which case we may also write $\mathcal{A} \models \Gamma$.
- We say Γ *semantically entails* or *implies* (others say *logically entails* or *implies*) Δ iff every model of Γ is a model of Δ (but not necessarily the converse), and we may write $\mathsf{models}(\Gamma) \subseteq \mathsf{models}(\Delta)$.

Sometimes we may want to make explicit the signature Σ of a model $\mathcal{A}$ of Γ, in which case we may say that $\mathcal{A}$ is a Σ-model. If the equality symbol $\approx$ occurs in Γ, we may say $\mathcal{A}$ is a $(\Sigma \cup \{\approx\})$-model.

Let $\varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma \cup \{\approx\}, X)$ and let $\mathrm{FV}(\varphi) = \{x_1, \ldots, x_n\}$. By a slight abuse of notation, we may write $\varphi(x_1, \ldots, x_n)$ to indicate which variables have free occurrences in φ. Let $(\mathcal{A}, \sigma)$ be an interpretation for φ, consisting of a Σ-structure $\mathcal{A}$ and a valuation $\sigma : X \to A$, and let:

$$\sigma(x_1) = a_1, \ldots, \sigma(x_n) = a_n.$$

Then, instead of writing $\mathcal{A}, \sigma \models \varphi$, we may write:

$$\mathcal{A}, a_1, \ldots, a_n \models \varphi \quad \text{or also} \quad \mathcal{A} \models \varphi[a_1, \ldots, a_n]$$

with the understanding that the elements $a_1, \ldots, a_n \in A$ are substituted for the variables $x_1, \ldots, x_n$ (in the same order) in φ. Note the additional abuse of notation when we write "$\varphi[a_1, \ldots, a_n]$" (what is it?).

C Systems of Formal Proofs

The syntax and semantics of all the formal logics in these lecture notes are basically the same that you will find elsewhere in the published literature. The differences are unessential, mostly in the notation, in the presentation, and sometimes in the terminology.

This is no longer the case when we consider their proof systems. For each of our formal logics, there are many proof systems, and each system has its own advantages and disadvantages – though at the end, they all fulfill the requirement of the same Completeness Theorem, which asserts, in a nutshell: "whatever is true according to the semantics is also formally provable." But that is not the only requirement by which we choose a proof system, and there are indeed requirements fulfilled by some but not all proof systems.

The profusion of proof systems for the same formal logic is always a little bewildering for newcomers to this material. It takes time and effort to understand and appreciate the reasons for their differences, all related to different aspects of formal proofs (*e.g.*, the efficient implementation of procedures for deciding validity, or an examination of what is called *cut-elimination* and its implications regarding consistency, or the existence of what are called *interpolation theorems*, and other proof-theoretic matters). These are all important topics beyond the scope of these lecture notes.

But we still have to select at least one proof system to round off our presentation. We choose here a particular way of setting it up, so-called *natural deduction*, and a particular way of defining its formal rules and organizing its formal derivations. There is no overarching reason to choose *natural deduction* over the many other alternatives, except that it is a little easier to present and seems to be favored by computer scientists, particularly by researchers in areas related to automated theorem provers and interactive proof assistants.

In each of our logics, if we can formally derive a wff ψ (the *conclusion*, also called *consequent*) from a finite set of wff's $\{\varphi_1, \ldots, \varphi_n\}$ (the *premises*, also called *antecedents* or *hypotheses*) according to the rules of natural deduction, we can assert this fact by writing:

$$\varphi_1, \ldots, \varphi_n \;\vdash\; \psi \quad \text{or, if we want to make explicit the logic we use,} \quad \varphi_1, \ldots, \varphi_n \vdash_{\mathcal{L}} \psi,$$

where $\mathcal{L} \in \{\mathsf{PL}, \mathsf{QBF}, \mathsf{eL}, \mathsf{ZOL}, \mathsf{EL}, \mathsf{QEL}, \mathsf{FOL}\}$. Such an expression is called a *sequent* in these lecture notes, even though the word is not used by all authors and with the same intention. In our setup, the symbol "$\vdash$" is outside the formalism of natural deduction. It is only after a natural-deduction proof is completed that we use "$\vdash$", to separate wff's that are assumed to hold with no justification necessary (these are the premises) from the wff appearing on the last line (the conclusion).[29] Several examples for how to use the proof rules are in Appendix D and Appendix E.

C.1 Rules for PL

Following tradition, rules are given suggestive names. For the logical connectives, here limited to $\{\wedge, \vee, \rightarrow, \neg\}$, rules come in pairs. Each pair has one *introduction* rule and one *elimination* rule, indicated by the letters "I" and "E", respectively. Sometimes the *introduction* rule has two parts, *e.g.*, rule $(\vee\mathrm{I})$ has two parts: $(\vee\mathrm{I}_1)$ and $(\vee\mathrm{I}_2)$; and sometimes the *elimination* rule has two parts, *e.g.*, rule $(\wedge\mathrm{E})$ has two parts: $(\wedge\mathrm{E}_1)$ and $(\wedge\mathrm{E}_2)$.

[29] The symbol "$\vdash$" is often called *turnstile* because of its resemblance to a typical turnstile if viewed from above. You may read it as "formally yields", or "formally proves", or "formally derives". According to Wikipedia, "$\vdash$" was first introduced towards the end of the 19th Century, by the mathemaical logician Gottlob Frege in 1879. It thus preceded its companion "$\models$" by many decades, which also reflects how concerns of mathematical logicians evolved over time, initially focusing on proof-theoretic issues and subsequently adding model-theoretic issues. See footnote 27 for comments on "$\models$".

C.1.1 *Introduction* and *elimination* rules for each of $\{\wedge, \vee, \to, \neg\}$:

$$\frac{\varphi \quad \psi}{\varphi \wedge \psi}\ (\wedge\mathrm{I}) \qquad\qquad \frac{\varphi \wedge \psi}{\varphi}\ (\wedge\mathrm{E}_1) \qquad\qquad \frac{\varphi \wedge \psi}{\psi}\ (\wedge\mathrm{E}_2)$$

$$\frac{\varphi}{\varphi \vee \psi}\ (\vee\mathrm{I}_1) \qquad\qquad \frac{\psi}{\varphi \vee \psi}\ (\vee\mathrm{I}_2) \qquad\qquad \frac{\varphi \vee \psi \quad \boxed{\begin{array}{c}\varphi\\ \vdots\\ \theta\end{array}} \quad \boxed{\begin{array}{c}\psi\\ \vdots\\ \theta\end{array}}}{\theta}\ (\vee\mathrm{E})$$

$$\frac{\boxed{\begin{array}{c}\varphi\\ \vdots\\ \psi\end{array}}}{\varphi \to \psi}\ (\to\mathrm{I}) \qquad\qquad \frac{\varphi \quad \varphi \to \psi}{\psi}\ (\to\mathrm{E})$$

$$\frac{\boxed{\begin{array}{c}\varphi\\ \vdots\\ \bot\end{array}}}{\neg\varphi}\ (\neg\mathrm{I}) \qquad\qquad \frac{\varphi \qquad \neg\varphi}{\bot}\ (\neg\mathrm{E})$$

Remark: In the rules with boxes $\{\,(\to\mathrm{I}),\ (\vee\mathrm{E}),\ (\neg\mathrm{I})\,\}$, the ellipsis points '. . .' may be empty. Thus, in $(\to\mathrm{I})$, it is possible that $\varphi = \psi$, and similarly in the rule $(\vee\mathrm{E})$, it is possible that $\varphi = \theta$ or $\psi = \theta$ (or both). In the rule $(\neg\mathrm{I})$, if the ellipsis points are empty, then $\varphi = \bot$, in which case the conclusion of the rule is $\neg\varphi = \neg\bot$.

C.1.2 An *elimination* rule for $\bot$, an *introduction* rule for $\top$:

$$\frac{\bot}{\varphi}\ (\bot\mathrm{E})$$ (also called *ex falso quodlibet* or just *ex falso*)

$$\frac{}{\top}\ (\top\mathrm{I})$$

There is no introduction rule for $\bot$ and no elimination rule fo $\top$. So far, there are 10 rules, not all of equal importance: You will be right in guessing that you get more traction from $(\to\mathrm{I})$ and $(\to\mathrm{E})$ in our proof system than from the other rules, while $(\top\mathrm{I})$ is useless (why?). But this question ("given a subset of the rules, what can be said about the wff's that are in its deductive closure?") is for another study outside these lecture notes.

C.1.3 Rules (LEM), (PBC), $(\neg\neg\mathrm{E})$, and (Peirce's) :

These four rules have a special status. Without any of these four, the preceding rules define a proof system for what is called *intuitionistic propositional logic*; such a proof system is complete for a semantics of *propositional logic* based on what are called *Heyting algebras*, which include as a special case the familiar *Boolean algebras*. This is another matter outside the scope of these notes.

Adding anyone of the four rules in $\{(\text{LEM}), (\text{PBC}), (\neg\neg\text{E}), (\text{Peirce's})\}$ augments the deductive power of the proof system and makes it complete for the semantics we use for *propositional logic* in these notes, one based on *Boolean algebras*. The system so augmented is sometimes called *classical propositional logic*, the qualifier "classical" being use to make explicit the distinction with "intuitionistic".

LEM is a shorthand for *Law of Excluded Middle*. **PBC** is a shorthand for *Proof by Contradiction*. As its name indicates, **¬¬E** means *elimination of double negation*. **Peirce's** stands for *Peirce's Law* and is named for the 19th Century mathematical logician Charles Sanders Peirce. Here are the precise formulations of the four rules:

$$\frac{}{\varphi \vee \neg\varphi}\ (\text{LEM}) \qquad\qquad \frac{\boxed{\begin{array}{c}\neg\varphi\\ \vdots\\ \bot\end{array}}}{\varphi}\ (\text{PBC})$$

$$\frac{\neg\neg\varphi}{\varphi}\ (\neg\neg\text{E}) \qquad\qquad \frac{}{((\varphi \to \psi) \to \varphi) \to \varphi}\ (\text{Peirce's})$$

Exercise 73. Show that any two of the four rules $\{(\text{LEM}), (\text{PBC}), (\neg\neg\text{E}), (\text{Peirce's})\}$ are inter-derivable. In fact, they are inter-derivable using only two rules, ($\to$I) and ($\to$E).

Hint: One way is to consider $\binom{4}{2} = 6$ cases, one for each pair from the set of four rules, with each pair involving two derivations, for a total of 12 derivations. A much simpler approach requires only 4 derivations:

(a) (Peirce's) is derivable from (PBC),

(b) (LEM) is derivable from (Peirce's),

(c) ($\neg\neg$E) is derivable from (LEM),

(d) (PBC) is derivable from ($\neg\neg$E).

Schematically, you have to show that (PBC) $\Rightarrow$ (Peirce's) $\Rightarrow$ (LEM) $\Rightarrow$ ($\neg\neg$E) $\Rightarrow$ (PBC). □

Exercise 74. In some accounts of natural deduction, the rule for *disjunction elimination* is given as ($\vee$E*):

$$\frac{\boxed{\begin{array}{c}\varphi\\ \vdots\\ \theta\end{array}} \quad \boxed{\begin{array}{c}\psi\\ \vdots\\ \theta\end{array}}}{(\varphi \vee \psi) \to \theta}\ (\vee\text{E}^*)$$

which is often more convenient to use than the standard ($\vee$E). You have to show that ($\vee$E) and ($\vee$E*) are inter-derivable. Specifically, there are two parts:

1. Show ($\vee$E) is derivable from ($\vee$E*) and ($\to$E).
2. Show ($\vee$E*) is derivable from ($\vee$E) and ($\to$I). □

C.2 Rules for QBF

All the rules in Section C.1 for *propositional logic* can be used, in addition to *introduction* and *elimination* rules for the quantifiers, namely, two rules $\{(\forall\mathrm{I}), (\forall\mathrm{E})\}$ for "$\forall$" and two rules $\{(\exists\mathrm{I}), (\exists\mathrm{E})\}$ for "$\exists$":

$$\dfrac{\boxed{\begin{array}{ll} q & \text{fresh var} \\ \vdots & \\ \varphi[p := q] & \end{array}}}{\forall p\, \varphi}\ (\forall\mathrm{I}) \qquad\qquad \dfrac{\forall p\, \varphi}{\varphi[p := t]}\ (\forall\mathrm{E})$$

$$\dfrac{\varphi[p := t]}{\exists p\, \varphi}\ (\exists\mathrm{I}) \qquad\qquad \dfrac{\exists p\, \varphi \qquad \boxed{\begin{array}{ll} q & \text{fresh var} \\ \varphi[p := q] & \text{assumption} \\ \vdots & \\ \psi & \end{array}}}{\psi}\ (\exists\mathrm{E})$$

The rules for quantifiers must be used with extra care, in a way to respect the following side conditions:

- "$\varphi[p := q]$" (resp. "$\varphi[p := t]$") means that q (resp. t) is substituted for every *free occurrence* of the propositional variable p in φ.
- In the rules $(\forall\mathrm{E})$ and $(\exists\mathrm{I})$, we use t as a metavariable ranging over $\{\bot, \top\} \cup \mathcal{P}$. Moreover, if t is the propositional variable $q \in \mathcal{P}$, then q must be *substitutable for* p in φ, *i.e.*, every occurrence of the substituted q must be outside the scope of a pre-existing binding quantifier, "$\forall q$" or "$\exists q$", in φ.

It takes some practice to use the quantifier rules without tripping on many pattern-matching complications. It is helpful to keep in mind informal justifications for the rules, at least the two rules with boxes:

- Informal justification for $(\forall\mathrm{I})$: *If we can derive $\varphi[p := q]$ with a fresh propositional variable q substituted for p in φ, then we can derive $(\forall p\, \varphi)$. The crucial qualification is that q is* fresh*, i.e., it does not occur anywhere outside the box. Thus, since we assume nothing about this q, the derivation works for any propositional variable substituted for q.*
- Informal justification for $(\exists\mathrm{E})$: *If we can derive $(\exists p\, \varphi)$, then φ must hold for at least one value. We then proceed by case analysis over possible values, writing q for a generic value representing them all. If we can derive ψ, which does not mention q, from the assumption $\varphi[p := q]$, then ψ must hold regardless of the value q stands for.*

C.3 Rules for FOL

All the rules in Section C.1 for *propositional logic* can be used, in addition to the four rules for quantifiers below. In fact, these four have the same exact form as the four quantifier rules in Section C.2, except that now the quantification is over first-order variables rather than propositional variables. The context making clear which are intended, I choose to identify them with the same four names $\{(\forall\mathrm{I}), (\forall\mathrm{E}), (\exists\mathrm{I}), (\exists\mathrm{E})\}$:

$$\dfrac{\boxed{\begin{array}{ll} y & \qquad\text{fresh var} \\ \vdots & \\ \varphi[x := y] & \end{array}}}{\forall x\ \varphi}\ (\forall \mathrm{I}) \qquad\qquad \dfrac{\forall x\ \varphi}{\varphi[x := t]}\ (\forall \mathrm{E})$$

$$\dfrac{\varphi[x := t]}{\exists x\ \varphi}\ (\exists \mathrm{I}) \qquad\qquad \dfrac{\exists x\ \varphi \qquad \boxed{\begin{array}{ll} y & \qquad\text{fresh var} \\ \varphi[x := y] & \qquad\text{assumption} \\ \vdots & \\ \psi & \end{array}}}{\psi}\ (\exists \mathrm{E})$$

The side conditions for the rules of first-order quantifiers here are nearly the same as those in Section C.2:

- "$\varphi[x := y]$" (resp. "$\varphi[x := t]$") means that first-order variable y (resp. term t) is substituted for every *free occurrence* of the first-order variable x in φ.
- In the rules ($\forall$E) and ($\exists$I), the term t must be *substitutable for* x in φ, *i.e.*, every variable $y \in \mathrm{FV}(t)$ must be outside the scope of a pre-existing binding quantifier, "$\forall y$" or "$\exists y$", in φ.

If we allow the equality symbol "$\approx$" in the syntax of first-order logic, then we need two additional rules: rule ($\approx$I) which introduces one occurrence of $\approx$, and rule ($\approx$E) which eliminates one occurrence of $\approx$. They read as follows:

$$\dfrac{}{t \approx t}\ (\approx \mathrm{I}) \qquad\qquad \dfrac{t_1 \approx t_2 \qquad \varphi[x := t_1]}{\varphi[x := t_2]}\ (\approx \mathrm{E})$$

subject to the following side conditions:

- t, t_1, t_2 range over the set of first-order terms.
- In the rule ($\approx$E), terms t_1 and t_2 must be *substitutable for* x, *i.e.*, every variable $y \in \mathrm{FV}(t_1) \cup \mathrm{FV}(t_2)$ must be outside the scope of a pre-existing binding quantifier, "$\forall y$" or "$\exists y$", in φ.

The rule ($\approx$I) guarantees that $\approx$ is reflexive. The other usual properties of equality, *symmetry* and *transitivity*, follow from ($\approx$I) and ($\approx$E), as shown in the next exercise.

Exercise 75. Alternative suggestive names for ($\approx$I) and ($\approx$E) are ($\approx$*reflexive*) and ($\approx$*congruent*), respectively. Show that both of the following rules:

$$\dfrac{t_1 \approx t_2}{t_2 \approx t_1}\ (\approx \textit{symmetric})$$

$$\dfrac{t_1 \approx t_2 \qquad t_2 \approx t_3}{t_1 \approx t_3}\ (\approx \textit{transitive})$$

are derivable from ($\approx$I) and ($\approx$E). □

Exercise 76. Show that the following rule is derivable from ($\approx$E):

$$\dfrac{t_1 \approx u_1 \ \cdots\ t_n \approx u_n \qquad \varphi[x_1 := t_1, \ldots, x_n := t_n]}{\varphi[x_1 := u_1, \ldots, x_n := u_n]}$$

Particular cases of ($\approx$E*) is when φ is $R(x_1, \ldots, x_n)$ where R is a n-ary relation symbol or when φ is $f(x_1, \ldots, x_n) \approx y$ where f is a n-ary function symbol. □

C.4 Rules for eL

The set of wff's of *equality logic* is $\mathsf{WFF}_{\mathsf{eL}}(\{\approx\}, X)$. These wff's do not include quantifiers, which implies that the rules in $\{(\forall\mathrm{I}), (\forall\mathrm{E}), (\exists\mathrm{I}), (\exists\mathrm{E})\}$ do not apply to them.

The rules of natural deduction for *equality logic* are therefore all the rules in Section C.1 for *propositional logic* in addition to the rules ($\approx$I) and ($\approx$E) in Section C.3 and the rules derived from the preceding, namely: ($\approx$ *symmetric*), ($\approx$ *transitive*), and ($\approx$E*). See Exercises 75 and 76. Keep in mind that t, t_i, and u_i, in these rules are limited to range over variables in X when used for *equality logic*.

C.5 Rules for ZOL

The set of wff's of *zeroth-order logic* is $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma, \varnothing)$ or, when $\approx$ is allowed, $\mathsf{WFF}_{\mathsf{ZOL}}(\Sigma \cup \{\approx\}, \varnothing)$. These wff's do not include quantifiers, which implies that the rules in $\{(\forall\mathrm{I}), (\forall\mathrm{E}), (\exists\mathrm{I}), (\exists\mathrm{E})\}$ do not apply to them.

The rules of natural deduction for *zeroth-order logic* are therefore all the rules in Section C.1 for *propositional logic* in addition to the rules ($\approx$I) and ($\approx$E) in Section C.3 and the rules derived from the preceding, namely: ($\approx$ *symmetric*), ($\approx$ *transitive*), and ($\approx$E*). Keep in mind that t, t_i, and u_i, in these rules are limited to range over variable-free first-order terms, *i.e.*, over the set $\mathsf{Atoms}(\Sigma, \varnothing)$, when used for *zeroth-order logic*.

C.6 Rules for EL and QEL

(MORE TO COME)

C.7 Soundness and Weak Completeness

Among the important relationships connecting the semantics and the proof theory of a logic $\mathcal{L}$ is *Soundness*. Another is *Weak Completeness*, a weaker version of *Completeness* in full generality which is examined in the main body of these lecture notes.

We first present *Weak Completeness* for PL only, which is also the only *Weak Completeness* we invoke elsewhere in these notes.

Theorem 77 (Weak Compleness for PL)**.** *For any* finite *set* $\Gamma \cup \{\varphi\}$ *of wff's in* $\mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$*, it holds that:*

$$\Gamma \models_{PL} \varphi \quad \textit{implies} \quad \Gamma \vdash_{PL} \varphi.$$

In words, if φ is semantically entailed/implied by Γ, then φ is formally deducible from Γ.

Note that the set Γ is restricted to be finite in Weak Completeness*, which can thus be proved independently of* Compactness*.* Completenes *in full generality lifts the restriction that Γ is finite and, as a result, depends on* Compactness *in an essential way.*

(MORE TO COME)

Soundness is a minimal requirement for any system of formal proofs: it means that formal *deducibility* (others say *derivability*) implies *semantic validity* (others say *truth*). We want a proof system to be as strong as possible, *i.e.*, to formally derive as many semantically valid wff's as possible, without deriving a contradiction.

Theorem 78 (Soundness)**.** *For any of the logics* $\mathcal{L} \in \{\mathsf{PL}, \mathsf{QBF}, \mathsf{eL}, \mathsf{ZOL}, \mathsf{EL}, \mathsf{QEL}, \mathsf{FOL}\}$ *defined in these lecture notes, and for any set* $\Gamma \cup \{\varphi\}$ *of wff's in* $\mathcal{L}$*, it holds that:*

$$\Gamma \vdash_{\mathcal{L}} \varphi \quad \textit{implies} \quad \Gamma \models_{\mathcal{L}} \varphi.$$

In words, if φ *is formally deducible from* Γ*, then* φ *is semantically entailed/implied by* Γ*. The proof system for* $\mathcal{L}$ *is thus not too strong.*

Proof. For every logic $\mathcal{L}$ in these notes, the proof of soundness follows the same pattern: a straightforward (though somewhat laborious) induction on the length $\geqslant 1$ of the natural deduction for the sequent $\Gamma \vdash_{\mathcal{L}} \varphi$. We take the length of a natural deduction to be the number of lines containing a wff, with one wff per line. We do *not* include in the count a line which introduces a fresh variable, as "fresh var" in the rules ($\forall$I) and ($\forall$E); so, whenever we say "line" in this proof, we mean "wff" and we view a natural deduction as written top-down as a sequence of wff's.

We restrict attention to the case when $\mathcal{L}$ is FOL, which is the most involved logic in these notes. We thus omit the subscript "$\mathcal{L}$" on "$\vdash$" and "$\models$" in the rest of the proof and omit many of the obvious details. Given a natural deduction $\mathcal{D}$ for the sequent $\Gamma \vdash \varphi$, we say that $\mathcal{D}$ *satisfies the conclusion of the theorem* iff $\Gamma \models \varphi$.

The basis of the induction is when the natural deduction $\mathcal{D}$ for $\Gamma \vdash \varphi$ consists of a single line, *i.e.*, the single wff φ. In this case, $\varphi \in \Gamma$, which also implies that $\Gamma \models \varphi$, *i.e.*, $\mathcal{D}$ satisfies the conclusion of the theorem.

The induction proceeds by considering a natural deduction $\mathcal{D}$ with $k + 1$ lines, with the induction hypothesis being that every natural deduction $\mathcal{D}'$ with at most k lines satisfies the conclusion of the theorem, where $k \geqslant 1$. If the natural deduction $\mathcal{D}$ has $k + 1$ lines, we consider the proof rule according to which the last line in $\mathcal{D}$ is obtained – for each of the possible proof rules. There are the proof rules of PL, which can be used again in FOL, and there are the proof rules specifically belonging to FOL, namely, $\{(\forall \mathrm{I}), (\forall \mathrm{E}), (\exists \mathrm{I}), (\exists \mathrm{E}), (\approx \mathrm{I}), (\approx \mathrm{E})\}$.

Consider the rules inherited from PL first. So, suppose the last line in the natural deduction $\mathcal{D}$ with $k+1$ lines is obtained by applying the rule ($\wedge$I). Thus, $\mathcal{D}$ is a natural deduction for a sequent of the form $\Gamma \vdash (\varphi_1 \wedge \varphi_2)$. This implies there are two natural deductions $\mathcal{D}_1$ and $\mathcal{D}_2$ whose respective last lines are φ_1 and φ_2. Let the respective sets of premises in $\mathcal{D}_1$ and $\mathcal{D}_2$ be Γ_1 and Γ_2, so that $\mathcal{D}_1$ and $\mathcal{D}_2$ are natural deductions for the sequents $\Gamma_1 \vdash \varphi_1$ and $\Gamma_2 \vdash \varphi_2$. It also follows that $\Gamma \supseteq \Gamma_1 \cup \Gamma_2$.

By the induction hypothesis, we have $\Gamma_1 \models \varphi_1$ and $\Gamma_2 \models \varphi_2$. Let $(\mathcal{A}, \sigma)$ be an interpretation such that $\mathcal{A}, \sigma \models \Gamma$, which implies that both $\mathcal{A}, \sigma \models \Gamma_1$ and $\mathcal{A}, \sigma \models \Gamma_2$, because Γ_1 and Γ_2 are subsets of Γ. Hence, both $\mathcal{A}, \sigma \models \varphi_1$ and $\mathcal{A}, \sigma \models \varphi_2$, because $\Gamma_1 \models \varphi_1$ and $\Gamma_2 \models \varphi_2$. Hence, $\mathcal{A}, \sigma \models (\varphi_1 \wedge \varphi_2)$. Hence, $\Gamma \models (\varphi_1 \wedge \varphi_2)$, as desired.

We omit the cases when the last line in the natural deduction $\mathcal{D}$ with $k + 1$ lines is a wff of the form $(\varphi_1 \vee \varphi_2)$ or $(\varphi_1 \to \varphi_2)$ or $(\neg \varphi)$, which are totally similar to the case when the last line is $(\varphi_1 \wedge \varphi_2)$.

For one more rule inherited from PL, consider the case when the last line in $\mathcal{D}$, which has $k+1$ lines, is obtained by applying the rule (PBC) and showing that a sequent $\Gamma \vdash \varphi$ holds. Thus, just before the last line in $\mathcal{D}$, there is a closed box, call it B, whose first line is $\neg\varphi$ (it is an "assumption" or "local hypothesis") and whose last line is $\bot$. The rule (PCB) is invoked to close B and to write the last line of $\mathcal{D}$ which is φ. If we add $\neg\varphi$ as a premise to the entire deduction, we can open the box B (*i.e.*, remove the frame of B but not its contents!) and obtain a natural deduction with k lines for the sequent $\Gamma \cup \{\neg\varphi\} \vdash \bot$. By the induction hypothesis, $\Gamma \cup \{\neg\varphi\} \models \bot$. Since for all interpretations $(\mathcal{A}, \sigma)$ we have that $\mathcal{A}, \sigma \not\models \bot$, it follows that for all interpretations $(\mathcal{A}, \sigma)$ we also have $\mathcal{A}, \sigma \not\models \Gamma \cup \{\neg\varphi\}$, *i.e.*, $\Gamma \cup \{\neg\varphi\}$ is unsatisfiable. Hence, $\Gamma \models \varphi$, as desired.[30]

[30]For additional details for this last step, see Lemma 6 and its proof.

We next consider proof rules that are specific to FOL, $\{(\forall\text{I}), (\forall\text{E}), (\exists\text{I}), (\exists\text{E}), (\approx\text{I}), (\approx\text{E})\}$, and examine a natural deduction $\mathcal{D}$ with $k+1$ lines whose last line is obtained by applying one of those rules. We limit our examination to the case of rules (∀I) and (∀E), the case of the other rules being totally similar.

So, suppose the last line in $\mathcal{D}$ is obtained by applying rule (∀E) to show $\Gamma \vdash \varphi[x := t]$. Thus, the last line is the wff $\varphi[x := t]$, and the last but one line is $(\forall x\ \varphi)$. Let $\mathcal{D}'$ be the natural deduction consisting of the first k lines in $\mathcal{D}$, which establishes the sequent $\Gamma \vdash (\forall x\ \varphi)$. By the induction hypothesis, it holds that $\Gamma \models (\forall x\ \varphi)$. This means that for all interpretations $(\mathcal{A}, \sigma)$, if $\mathcal{A}, \sigma \models \Gamma$ then $\mathcal{A}, \sigma \models (\forall x\ \varphi)$, which in turn implies that $\mathcal{A}, \big(\sigma[x \mapsto a]\big) \models \varphi$ for all $a \in A$ where A is the universe of $\mathcal{A}$. In the notation of Section B.3, this is the same as $[\![\varphi]\!]_{\mathcal{A},\sigma[x\mapsto a]} = \mathbf{T}$ for all $a \in A$. Now observe that:

$$\Big\{\, [\![t]\!]_{\mathcal{A},\sigma[x\mapsto a]} \;\Big|\; a \in A \Big\} \;\subseteq\; \Big\{\, [\![x]\!]_{\mathcal{A},\sigma[x\mapsto a]} \;\Big|\; a \in A \Big\} \;=\; A.$$

Hence, $[\![\varphi[x := t]]\!]_{\mathcal{A},\sigma[x\mapsto a]} = \mathbf{T}$ for all $a \in A$. Hence, $\mathcal{A}, \big(\sigma[x \mapsto a]\big) \models \varphi[x := t]$, for every interpretation $(\mathcal{A}, \sigma)$ and every $a \in A$. This in turn implies $\mathcal{A}, \sigma \models \varphi[x := t]$ for every interpretation $(\mathcal{A}, \sigma)$, as desired.

Finally, consider the case when the last line in $\mathcal{D}$ is obtained by applying rule (∀I) to show $\Gamma \vdash (\forall x\ \varphi)$. Thus, the last line is the wff $(\forall x\ \varphi)$. We invoke (∀I) to close a box, call it B. B starts with a fresh variable y (which does not count as a separate line in $\mathcal{D}$) and ends with the wff $\varphi[x := y]$ where y occurs free. We therefore have $\Gamma \vdash \varphi[x := y]$. By the induction hypothesis, $\Gamma \models \varphi[x := y]$, *i.e.*, for every interpretation $(\mathcal{A}, \sigma)$ it holds that if $\mathcal{A}, \sigma \models \Gamma$ then $\mathcal{A}, \sigma \models \varphi[x := y]$. Equivalently, for every $a \in A$, if $\mathcal{A}, \big(\sigma[x \mapsto a]\big) \models \Gamma$ then $\mathcal{A}, \big(\sigma[x \mapsto a]\big) \models \varphi[x := y]$. Since y does not occur in Γ, we also have for every $(\mathcal{A}, \sigma)$, if $\mathcal{A}, \sigma \models \Gamma$ then for every $a \in A$, it holds that $\mathcal{A}, \big(\sigma[x \mapsto a]\big) \models \varphi[x := t]$. Hence, $\Gamma \models (\forall x\ \varphi)$, as desired. □

D De Morgan's Laws: Semantically and Proof-Theoretically

De Morgan's Laws can be asserted as four semantically valid wff's:

1. $\models \neg(p \vee q) \quad \to \ (\neg p \wedge \neg q)$
2. $\models (\neg p \wedge \neg q) \ \to \ \neg(p \vee q)$
3. $\models (\neg p \vee \neg q) \ \to \ \neg(p \wedge q)$
4. $\models \neg(p \wedge q) \quad \to \ (\neg p \vee \neg q)$

Their semantic validity can be established using *truth tables*. For example, for the first and fourth laws we can write the following tables:

p	q	$p \vee q$	$\neg(p \vee q)$	$\neg p$	$\neg q$	$(\neg p \wedge \neg q)$	$\neg(p \vee q) \to (\neg p \wedge \neg q)$
F	**F**	**F**	**T**	**T**	**T**	**T**	**T**
F	**T**	**T**	**F**	**T**	**F**	**F**	**T**
T	**F**	**T**	**F**	**F**	**T**	**F**	**T**
T	**T**	**T**	**F**	**F**	**F**	**F**	**T**

p	q	$p \wedge q$	$\neg(p \wedge q)$	$\neg p$	$\neg q$	$(\neg p \vee \neg q)$	$\neg(p \wedge q) \to (\neg p \vee \neg q)$
F	**F**	**F**	**T**	**T**	**T**	**T**	**T**
F	**T**	**F**	**T**	**T**	**F**	**T**	**T**
T	**F**	**F**	**T**	**F**	**T**	**T**	**T**
T	**T**	**T**	**F**	**F**	**F**	**F**	**T**

The two leftmost columns in the two tables list all possible truth assignments to the pair (p, q). The rightmost column in the first table assigns a truth-value to the wff $\neg(p \vee q) \to (\neg p \wedge \neg q)$ for each of the assignments of (p, q), and since every entry in the rightmost column is **T**, the wff is semantically valid. And similarly for the rightmost column in the second table, which establishes the semantic validity of the wff $\neg(p \wedge q) \to (\neg p \vee \neg q)$.

Exercise 79. Write the truth tables for the remaining de Morgan's Laws: 2 and 3, to show that they are all semantically valid. □

De Morgan's Laws can also be asserted in the form of four formally deducible sequents according to the proof rules in Section C.1:

1. $\vdash \neg(p \vee q) \quad \to \ (\neg p \wedge \neg q)$
2. $\vdash (\neg p \wedge \neg q) \ \to \ \neg(p \vee q)$
3. $\vdash (\neg p \vee \neg q) \ \to \ \neg(p \wedge q)$
4. $\vdash \neg(p \wedge q) \quad \to \ (\neg p \vee \neg q)$

Below are natural-deduction formal proofs for the first and fourth de Morgan's Laws:

1	$\neg(p \vee q)$	assume
2	p	assume
3	$p \vee q$	$\vee$I 2
4	$\bot$	$\neg$E $1, 3$
5	$\neg p$	$\neg$I 2-4
6	q	assume
7	$p \vee q$	$\vee$I 6
8	$\bot$	$\neg$E $1, 7$
9	$\neg q$	$\neg$I 6-8
10	$\neg p \wedge \neg q$	$\wedge$I $5, 9$
11	$\neg(p \vee q) \rightarrow (\neg p \wedge \neg q)$	$\rightarrow$I 1-10

1	$\neg(p \wedge q)$	assume
2	$\neg(\neg p \vee \neg q)$	assume
3	$\neg p$	assume
4	$(\neg p \vee \neg q)$	$\vee$I 3
5	$\bot$	$\neg$E $2, 4$
6	$\neg\neg p$	$\neg$I 3-5
7	$\neg q$	assume
8	$\neg p \vee \neg q$	$\vee$I 7
9	$\bot$	$\neg$E $2, 8$
10	$\neg\neg q$	$\neg$I 7-9
11	p	$\neg\neg$E 6
12	q	$\neg\neg$E 10
13	$p \wedge q$	$\wedge$I $11, 12$
14	$\bot$	$\neg$E $1, 13$
15	$\neg\neg(\neg p \vee \neg q)$	$\neg$I 2-14
16	$(\neg p \vee \neg q)$	$\neg\neg$E 15
17	$\neg(p \wedge q) \rightarrow (\neg p \vee \neg q)$	$\rightarrow$I 1-16

Remark: Our formal proof of de Morgan's Law 1 does not use any rule in $\{$(LEM), (PBC), $(\neg\neg$E), (Peirce's)$\}$, in contrast to our formal proof of de Morgan's Law 4, which uses $(\neg\neg$E) twice. It turns out that any formal proof for de Morgan's Law 4 must use one of the rules in $\{$(LEM), (PBC), $(\neg\neg$E), (Peirce's)$\}$; this is not a trivial result and requires a deeper examination of formal-proof systems (not in these lecture notes).

See Section C.1.3 and Exercise 73 for an explanation of how the rules in $\{$(LEM), (PBC), $(\neg\neg$E), (Peirce's)$\}$ are related.

Exercise 80. Write natural-deduction proofs for the remaining de Morgan's Laws: 2 and 3, to show that they are all formally derivable. For full credit, avoid using any of the four rules in $\{$(LEM), (PBC), $(\neg\neg$E), (Peirce's)$\}$. *Hint*: For de Morgan's Laws 2 and 3 this is possible, though it may require a little more care. $\square$

From the preceding exercise and earlier remark, de Morgan's Law 4 has a special status: the first three de Morgan's Laws are valid intuitionistically, while de Morgan's Law 4 is not.

It should be clear by now that writing formal proofs is a tedious task, generally requiring many failed attempts to add a line in the deduction and causing as many backtrackings. The search for a legal deduction, *i.e.*, a deduction which is produced according to the proof rules, is therefore a process of repeated backtrackings in general. If this process does not terminate, it may be because our starting premises do not in fact imply our conjectured conclusion – not because we have not tried hard enough. More examples and exercises of natural-deduction proofs are in Appendix E.

E Prenex Form and Skolemization

The process of transforming a wff with quantifiers into its *prenex form*, and the additional process of *skolemizing* it, applies equally well to quantified Boolean wff's and first-order wff's. The transformation of the two kinds of wff's being entirely similar, we restrict our presentation to first-order wff's.

E.1 Prenex Form

A first-order wff $\varphi \in \mathsf{WFF}_{\mathsf{FOL}}(\Sigma \cup \{\approx\}, X)$ is in *prenex form* (or *prenex normal form*) iff φ consists of a (possibly empty) string of quantifiers followed by a quantifier-free wff. The string of quantifiers in φ is its *prefix* and the quantifier-free subformual of φ is its *matrix*.

We call our transformation "$\boxed{\mathsf{prenex}}$", and the result of applying it to an arbitrary wff φ is denoted "$\boxed{\mathsf{prenex}}(\varphi)$". In what follows, we use Q, possibly subscripted, to range over $\{\forall, \exists\}$. Moreover, if Q is $\forall$ (resp. $\exists$), then $\overline{Q}$ is $\exists$ (resp. $\forall$), *i.e.*, $\overline{\forall}$ denotes $\exists$ and $\overline{\exists}$ denotes $\forall$. For an arbitrary first-order wff φ, the wff $\boxed{\mathsf{prenex}}(\varphi)$ is therefore of the form $(Q_1 x_1 \cdots Q_n x_n.\, \psi)$ where $Q_1, \ldots, Q_n \in \{\forall, \exists\}$ and ψ is quantifier-free.

The definition of $\boxed{\mathsf{prenex}}(\varphi)$ is by structural induction on φ. This can be done in one of two ways: *top-bottom* (as in the definition of $\boxed{\mathsf{QBF} \mapsto \mathsf{PL}}$ in the proof of Lemma 15), or *bottom-up*. In *top-bottom*, we start with φ fully given and we think of $\boxed{\mathsf{prenex}}$ as being pushed down recursively through the sub-wff's of φ. In *bottom-up*, we define $\boxed{\mathsf{prenex}}(\varphi)$ simultaneously with φ, as the latter is being built up inductively. Our induction here is *bottom-up*, which is a bit simpler:

1. If φ is quantifier-free, then $\boxed{\mathsf{prenex}}(\varphi) \stackrel{\text{def}}{=} \varphi$.

2. If $\varphi \stackrel{\text{def}}{=} (\neg\psi)$ and $\boxed{\mathsf{prenex}}(\psi) = (Q_1 x_1 \cdots Q_n x_n.\, \theta)$ where θ is quantifier-free, then:

$$\boxed{\mathsf{prenex}}(\varphi) \stackrel{\text{def}}{=} \big(\overline{Q}_1 x_1 \cdots \overline{Q}_n x_n.\, \neg\theta\big).$$

   In the next two cases, let:

$$\boxed{\mathsf{prenex}}(\varphi_1) = (Q_1 y_1 \cdots Q_n y_n.\, \theta_1),$$

$$\boxed{\mathsf{prenex}}(\varphi_2) = (Q_1 z_1 \cdots Q_p z_p.\, \theta_2),$$

   where θ_1 and θ_2 are quantifier-free. By renaming bound variables in $\boxed{\mathsf{prenex}}(\varphi_1)$ and $\boxed{\mathsf{prenex}}(\varphi_2)$, we can assume that the variables in $\{y_1, \ldots, y_n, z_1, \ldots, z_p\}$ are all distinct and that:

$$\{y_1, \ldots, y_n\} \cap \mathrm{FV}\big(\boxed{\mathsf{prenex}}(\varphi_2)\big) = \varnothing \quad \text{and} \quad \{z_1, \ldots, z_p\} \cap \mathrm{FV}\big(\boxed{\mathsf{prenex}}(\varphi_1)\big) = \varnothing.$$

3. If $\varphi \stackrel{\text{def}}{=} (\varphi_1 \diamond \varphi_2)$ where $\diamond \in \{\wedge, \vee\}$, then:

$$\boxed{\mathsf{prenex}}(\varphi) \stackrel{\text{def}}{=} \big(Q_1 y_1 \cdots Q_n y_n\; Q_1 z_1 \cdots Q_p z_p.\, (\theta_1 \diamond \theta_2)\big).$$

4. If $\varphi \stackrel{\text{def}}{=} (\varphi_1 \to \varphi_2)$, then:

$$\boxed{\mathsf{prenex}}(\varphi) \stackrel{\text{def}}{=} \big(\, \overline{Q}_1 y_1 \cdots \overline{Q}_n y_n\; Q_1 z_1 \cdots Q_p z_p.\, (\theta_1 \to \theta_2)\, \big)$$

5. If $\varphi \stackrel{\text{def}}{=} (Qx.\,\psi)$ where $Q \in \{\forall, \exists\}$, then:

$$\boxed{\mathsf{prenex}}(\varphi) \stackrel{\text{def}}{=} \big(Qx.\,\boxed{\mathsf{prenex}}(\psi)\big).$$

We can show, for an arbitrary first-order wff φ, that φ and $\boxed{\mathsf{prenex}}(\varphi)$ are equivalent in one of two ways:

- semantically, *i.e.*, $\models\ \big(\varphi \to \boxed{\mathsf{prenex}}(\varphi)\big)\ \wedge\ \big(\boxed{\mathsf{prenex}}(\varphi) \to \varphi\big)$, or
- proof-theoretically, *i.e.*, $\vdash\ \big(\varphi \to \boxed{\mathsf{prenex}}(\varphi)\big)\ \wedge\ \big(\boxed{\mathsf{prenex}}(\varphi) \to \varphi\big)$.

Either way, we can follow the bottom-up induction which we used to define φ and $\boxed{\mathsf{prenex}}(\varphi)$ simultaneously. All we need for this are Lemma 81 and Lemma 88.

Lemma 81. *Let φ be an arbitrary first-order wff. Then:*

1. $\neg(\exists x.\,\varphi)$ *and* $\big(\forall x.\,\neg\varphi\big)$ *are equivalent wff's.*
2. $\neg(\forall x.\,\varphi)$ *and* $\big(\exists x.\,\neg\varphi\big)$ *are equivalent wff's.*

Proof. We give formal natural-deduction proofs, and we ask you to give (much easier) semantic proofs in Exercise 82. For part 1, it suffices to show (why?):

$\neg\exists x.\ \varphi(x)\ \vdash\ \forall x.\ \neg\varphi(x)$ instead of $\vdash\ \big(\neg\exists x.\ \varphi(x)\big) \to \big(\forall x.\ \neg\varphi(x)\big)$, and
$\forall x.\ \neg\varphi(x)\ \vdash\ \neg\exists x.\ \varphi(x)$ instead of $\vdash \forall x.\ \neg\varphi(x)\ \to\ \neg\exists x.\ \varphi(x)$.

1.	$\neg\exists x\ \varphi(x)$	premise
2.	y	fresh variable
3.	$\varphi(y)$	assumption
4.	$\exists x\ \varphi(x)$	∃I 3
5.	$\bot$	¬E 1, 4
6.	$\neg\varphi(y)$	¬I 3–5
7.	$\forall x\ \neg\varphi(x)$	∀I 2–6

1.	$\forall a\ \neg\varphi(a)$	premise
2.	$\exists a\ \varphi(a)$	assumption
3.	a_0	fresh variable
4.	$\varphi(a_0)$	assumption
5.	$\neg\varphi(a_0)$	∀E 1
6.	$\bot$	¬E 4,5
7.	$\bot$	∃E 2, 3–6
8.	$\neg\exists a\ \varphi(a)$	PBC 2–7

The natural deduction on the left says "$\neg\exists x.\ \varphi(x)\ \vdash\ \forall x.\ \neg\varphi(x)$", and the natural deduction on the right says "$\forall x.\ \neg\varphi(x)\ \vdash\ \neg\exists x.\ \varphi(x)$". For part 2 of the lemma, we can write the following:

1.	$\exists x\ \neg\varphi(x)$	premise
2.	y	fresh variable
3.	$\neg\varphi(y)$	assumption
4.	$\forall x\ \varphi(x)$	assumption
5.	$\varphi(y)$	∀E 4
6.	$\bot$	¬E 3, 5
7.	$\neg\forall x\ \varphi(x)$	¬I 4–6
8.	$\neg\forall x\ \varphi(x)$	∃E 1, 2–7

1.	$\neg\forall x\ \varphi(x)$	premise
2.	$\neg\exists x\ \neg\varphi(x)$	assumption
3.	y	fresh variable
4.	$\neg\varphi(y)$	assumption
5.	$\exists x\ \neg\varphi(x)$	∃I 4
6.	$\bot$	¬E 2, 5
7.	$\varphi(y)$	PBC 4–6
8.	$\forall x\ \varphi(x)$	∀I 3–7
9.	$\bot$	¬E 1, 8
10.	$\exists x\ \neg\varphi(x)$	PBC 2–9

From the left, we conclude $\big(\exists x.\, \neg\varphi(x)\big) \vdash \neg\big(\forall x.\, \varphi(x)\big)$, and from the right, $\neg\big(\forall x.\, \varphi(x)\big) \vdash \big(\exists x.\, \neg\varphi(x)\big)$. $\square$

Exercise 82. Write semantic proofs for the two equivalences in Lemma 81, noting that each equivalence consists of two implications. For example, for the first equivalence, you have to show both of the following:

- $\mathcal{A} \models \neg(\exists x.\, \varphi) \to \big(\forall x.\, \neg\varphi\big)$,
- $\mathcal{A} \models \big(\forall x.\, \neg\varphi\big) \to \neg(\exists x.\, \varphi)$,

where $\mathcal{A}$ is an arbitrary Σ-structure and, for simplicity, you can assume $\mathrm{FV}(\varphi) = \{x\}$. Do the same for the second equivalence in Lemma 81. $\square$

As a warm-up for the proof of Lemma 88 and the exercises following it, you may try the following examples. They all involve natural deductions showing that a sequent of the form $\varphi \vdash \psi$ holds; we omit the (typically easier) proof of the corresponding semantic validity $\varphi \models \psi$.

Example 83. We write two natural deductions showing that: $\neg\varphi \vee \psi \vdash \varphi \to \psi$ and $\varphi \to \psi \vdash \neg\varphi \vee \psi$.

1.	$\neg\varphi \vee \psi$	premise
2.	$\neg\varphi$	assumption
3.	φ	assumption
4.	$\bot$	$\neg$E 2, 3
5.	ψ	$\bot$E 4
6.	$\varphi \to \psi$	$\to$I 3–5
7.	ψ	assumption
8.	φ	assumption
9.	ψ	repeat 7
10.	$\varphi \to \psi$	$\to$I 8–9
11.	$\varphi \to \psi$	$\vee$E 1, 2–6, 7–10

1.	$\varphi \to \psi$	premise
2.	$\varphi \vee \neg\varphi$	LEM
3.	$\neg\varphi$	assumption
4.	$\neg\varphi \vee \psi$	$\vee\mathrm{I}_1$ 3
5.	φ	assumption
6.	ψ	$\to$E 1, 5
7.	$\neg\varphi \vee \psi$	$\vee\mathrm{I}_2$ 6
8.	$\neg\varphi \vee \psi$	$\vee$E 2, 3–4, 5–7

The two preceding natural deductions show that $(\varphi \to \psi)$ and $(\neg\varphi \vee \psi)$ are equivalent wff's. Note that the deduction on the left does not use any of the four rules in $\{$(LEM), (PBC), ($\neg\neg$E), (Peirce's)$\}$ which is therefore legal intuitionistically, whereas the deduction on the right uses (LEM). It can be shown (not in these notes) that it is not possible to write a deduction for $(\varphi \to \psi) \to (\neg\varphi \vee \psi)$ without invoking one of those four rules. $\square$

Example 84. Permuting two adjacent universal quantifiers does not change the meaning of a wff, *i.e.*, the following sequent holds: $\forall x\, \forall y\, \varphi(x, y) \;\vdash\; \forall y\, \forall x\, \varphi(x, y)$, as confirmed by the following natural deduction.

1.	$\forall x\, \forall y\, \varphi(x, y)$	premise
2.	y_0	fresh y_0
3.	x_0	fresh x_0
4.	$\forall y\, \varphi(x_0, y)$	$\forall$E 1
5.	$\varphi(x_0, y_0)$	$\forall$E 4
6.	$\forall x\, \varphi(x, y_0)$	$\forall$I 3–5
7.	$\forall y\, \forall x\, \varphi(x, y)$	$\forall$I 2–6

where only the rules ($\forall$E) and ($\forall$I) are used. $\square$

Example 85. An existential quantifier can be distributed over a logical or "$\vee$", *i.e.*, the following sequent holds: $\exists x\ (\varphi(x) \vee \psi(x)) \ \vdash\ \exists x\ \varphi(x)\ \vee\ \exists x\ \psi(x)$, as confirmed by the following natural deduction:

1.	$\exists x\ (\varphi(x) \vee \psi(x))$	premise
2.	x_0	fresh variable
3.	$\varphi(x_0) \vee \psi(x_0)$	assumption
4.	$\varphi(x_0)$	assumption
5.	$\exists x\ \varphi(x)$	$\exists$I 4
6.	$\exists x\ \varphi(x) \vee \exists x\ \psi(x)$	$\vee$I 5
7.	$\psi(x_0)$	assumption
8.	$\exists x\ \psi(x)$	$\exists$I 7
9.	$\exists x\ \varphi(x) \vee \exists x\ \psi(x)$	$\vee$I 8
10.	$\exists x\ \varphi(x) \vee \exists x\ \psi(x)$	$\vee$E 3, 4–6, 7–9
11.	$\exists x\ \varphi(x) \vee \exists x\ \psi(x)$	$\exists$E 1, 2–10

where only rules for '$\vee$' and '$\exists$' are used, both for introduction and elimination. $\square$

Exercise 86. Write a natural deduction to establish the converse of the sequent in Example 85, to formally prove that the following sequent holds: $\exists x\ \varphi(x)\ \vee\ \exists x\ \psi(x)\ \vdash\ \exists x\ (\varphi(x) \vee \psi(x))$. This shows that we can "push" to the "left" existential quantifiers out of the scope of a "$\vee$" immediately preceding them. $\square$

Exercise 87. Read Example 85 and do Exercise 86 before attempting this exercise. Write natural deductions to establish the two following sequents:

- $\forall x\ \varphi(x)\ \wedge\ \forall x\ \psi(x)\ \vdash\ \forall x\ (\varphi(x) \wedge \psi(x))$.
- $\forall x\ (\varphi(x) \wedge \psi(x))\ \vdash\ \forall x\ \varphi(x)\ \wedge\ \forall x\ \psi(x)$.

We can "push" to the "left" universal quantifiers out of the scope of a "$\wedge$" immediately preceding them. $\square$

Lemma 88. *Let φ and ψ be arbitrary first-order wff's, such that $x \notin \mathrm{FV}(\psi)$. Then:*

1. $\big((\forall x\, \varphi) \wedge \psi\big)$ *and* $\big(\forall x\, (\varphi \wedge \psi)\big)$ *are equivalent wff's.*
2. $\big((\exists x\, \varphi) \wedge \psi\big)$ *and* $\big(\exists x\, (\varphi \wedge \psi)\big)$ *are equivalent wff's.*
3. $\big((\forall x\, \varphi) \vee \psi\big)$ *and* $\big(\forall x\, (\varphi \vee \psi)\big)$ *are equivalent wff's.*
4. $\big((\exists x\, \varphi) \vee \psi\big)$ *and* $\big(\exists x\, (\varphi \vee \psi)\big)$ *are equivalent wff's.*
5. $\big((\forall x\, \varphi) \to \psi\big)$ *and* $\big(\exists x\, (\varphi \to \psi)\big)$ *are equivalent wff's.*
6. $\big((\exists x\, \varphi) \to \psi\big)$ *and* $\big(\forall x\, (\varphi \to \psi)\big)$ *are equivalent wff's.*
7. $\big(\psi \to (\forall x\, \varphi)\big)$ *and* $\big(\forall x\, (\psi \to \varphi)\big)$ *are equivalent wff's.*
8. $\big(\psi \to (\exists x\, \varphi)\big)$ *and* $\big(\exists x\, (\psi \to \varphi)\big)$ *are equivalent wff's.*

In parts $\{1, 2, 3, 4\}$, we omit the cases when the two components of the logical connectives are permuted, as in $\big(\psi \wedge (\exists x\, \varphi)\big)$ instead of $\big((\exists x\, \varphi) \wedge \psi\big)$, because "$\wedge$" and "$\vee$" are commutative binary connectives.

Proof. Left as an exercise, which should be straightforward after studying Examples 83, 84, and 85, and doing Exercises 86 and 87. $\square$

Exercise 89. Prove each of the four odd-numbered (or the four even-numbered) equivalences in Lemma 88 twice: once by writing natural deductions (more tedious), and once by providing rigorous semantic arguments (simpler and easier). □

Proposition 90. *Let φ be an arbitrary first-order wff and $\psi \stackrel{\text{def}}{=} \boxed{\textsf{prenex}}(\varphi)$. Then φ and ψ are equivalent wff's.*

Proof. We repeat the *bottom-up* induction that defines φ and $\boxed{\textsf{prenex}}(\varphi)$ simultaneously. But now, at every step of the induction, we also show that φ and $\boxed{\textsf{prenex}}(\varphi)$ are equivalent wff's. At step 2 of the induction, you need to us Lemma 81 repeatedly to "move" quantifiers to the left past the logical negation "$\neg$". At steps 3 and 4, you need to use Lemma 88 to "move" quantifiers outside the logical connectives "$\wedge$", "$\vee$", and "$\rightarrow$". All obvious details omitted. □

E.2 Skolem Form

Let ψ be a first-order wff in prenex form. Again here, as in Section E.1, ψ may be a quantified Boolean wff or a first-order wff. We limit our examination to the first-order case, the case of quantified Boolean wff's being totally similar.

The *Skolemization* of ψ produces another first-order wff, call it θ, in prenex form where the prefix of quantifiers mentions only the universal "$\forall$". Our name for the transformation from ψ to θ is "$\boxed{\textsf{skolem}}$". The wff θ is obtained by initially setting θ to ψ and then repeatedly applying the following three-step sequence to it:[31]

1. Find the leftmost $\exists$ in the quantifier prefix of ψ, which binds a variable x and appears as "$\exists\, x$",
2. Introduce a fresh function symbol f_x of arity equal to the number of $\forall$'s to the left of "$\exists\, x$",
3. If the $\forall$'s to the left of "$\exists\, x$" are "$\forall y_1 \cdots \forall y_n$", then cross out "$\exists\, x$" from the quantifier prefix and replace all occurrences of x in the matrix of ψ by the term $f_x(y_1, \ldots, y_n)$.

This process is bound to terminate because the initial prefix of quantifiers in ψ has finite length. We denote the resulting θ by writing "$\boxed{\textsf{skolem}}(\psi)$", and refer to it as the *Skolem form* of ψ.

Note that there are as many new fresh function symbols "f_x" in θ as there are existential quantifiers "$\exists\, x$" in the prefix of the initial wff ψ in prenex form. These fresh function symbols are called *Skolem functions*. Note also that if the leftmost "$\exists\, x$" in the initial ψ is not preceded by any $\forall$, the associated Skolem function f_x has arity $= 0$, *i.e.*, f_x is a constant symbol.

If φ is an arbitrary first-order wff, not necessarily in prenex form, then we write $\boxed{\textsf{sko,pre}}(\varphi)$ to denote the two-stage transformation of φ – first, into prenex form and, second, into Skolemized form – and we also call $\boxed{\textsf{sko,pre}}(\varphi)$ the Skolemization of φ.

While φ and $\boxed{\textsf{prenex}}(\varphi)$ are logically equivalent ("they say the same thing"), it does not make sense to talk about the equivalence (or non-equivalence) of φ and $\boxed{\textsf{sko,pre}}(\varphi)$ because the signature of the latter is different from the signature of φ. Nevertheless, we have the following result. Recall that a *sentence* φ is a closed formula, *i.e.*, $\text{FV}(\varphi) = \varnothing$.

[31] The words *Skolemize* and *Skolemization* are derived from the name of the mathematical logician Thoralf Skolem. If you want to find out more about the many uses of *Skolemization*, click here.

Proposition 91. *Let φ and Γ be an arbitrary first-order sentence and set of first-order sentences. We then have:*

1. *φ is satisfiable iff* $\boxed{\textsf{sko,pre}}(\varphi)$,
2. *Γ is satisfiable iff* $\boxed{\textsf{sko,pre}}(\Gamma)$.

In Part 2, we have to be careful that, when we Skolemize distinct wff's φ_1 and φ_2 of Γ, we introduce distinct Skolem functions for each wff, i.e., the Skolem functions of φ_1 do not interfere with the Skolem functions of φ_2.

Proof. We leave the proof of Part 2 as an easy exercise implied by Part 1. For Part 1, we can assume that φ is already in prenex form, by Proposition 90. It suffices to show how the elimination of the leftmost existential quantifier from the prefix of φ produces another prenex form, say θ, which is equisatisfiable with φ, and then the same process can be repeated for the elimination of all the other existential quantifers in the prefix of φ. Let then φ be of the form:

$$\varphi \stackrel{\text{def}}{=} \forall x_1 \cdots \forall x_n \exists y\ \varphi_0$$

where $n \geqslant 0$ and φ_0 is a a prenex form such that $\text{FV}(\varphi_0) \subseteq \text{FV}(\varphi) \cup \{x_1, \ldots, x_n, y\}$ and, because φ is closed, in fact $\text{FV}(\varphi_0) = \{x_1, \ldots, x_n, y\}$. According to the Skolemization process, θ is of the form:

$$\theta \stackrel{\text{def}}{=} \forall x_1 \cdots \forall x_n\ \big(\varphi_0[y := f_y(x_1, \ldots, x_n)]\big)$$

where f_y is a fresh n-ary function symbol. Σ and $\Sigma' \stackrel{\text{def}}{=} \Sigma \cup \{f_y\}$ are the signatures of φ and θ, respectively.

Let $\mathcal{A}$ be a Σ-structure. The expansion $\mathcal{A}' \stackrel{\text{def}}{=} (\mathcal{A}, f_y^{\mathcal{A}'})$ of $\mathcal{A}$ is a Σ'-structure. Let A be the universe of $\mathcal{A}$, which is also the universe of $\mathcal{A}'$. If $\mathcal{A}' \models \theta$, it is easy to check that $\mathcal{A} \models \varphi$. Hence, if θ is satisfiable, so is φ.

Conversely, let $\mathcal{A} \models \varphi$ and let $\sigma : X \to A$ be an arbitrary valuation where A is the universe of $\mathcal{A}$. We construct a Σ'-structure $\mathcal{A}'$ by expanding $\mathcal{A}$ so that for every $a_1, \ldots, a_n \in A$, the function $f_y^{\mathcal{A}'}$ maps $(a_1, \ldots, a_n)$ to b where:[32]

$$\mathcal{A},\ \big(\sigma[x_1 \mapsto a_1, \ldots, x_n \mapsto a_n, y \mapsto b]\big)\ \models\ \ \varphi_0.$$

We choose the interpretation $f_y^{\mathcal{A}'}$ of the Skolem function f_y precisely so that the preceding satisfaction holds. It is now easy to check that $\mathcal{A}' \models \theta$. Hence, if φ is satisfiable, then so is θ. □

Exercise 92. What goes wrong in the proof of Proposition 91 if φ is an open wff?

Hint: Try the open wff $\varphi(y) \stackrel{\text{def}}{=} \exists! v \forall w \big(R(a, w) \wedge R(v, w)\big) \to \exists x \big(R(a, y) \wedge R(x, y)\big)$, where "$\exists!$" means "there exists exactly one", R is a binary relation symbol and a is a constant symbol. Show that $\models \varphi(y)$, but the construction in the proof of Proposition 91 produces an open wff $\theta(y)$ not satisfied by any structure $\mathcal{A}$, unless we introduce additional constraints at the meta-level on $\mathcal{A}$. □

Exercise 93. Let R be a binary relation symbol and f a unary function symbol.

1. Show that the sentence $\varphi \stackrel{\text{def}}{=} \forall x\, R\big(x, f(x)\big) \to \forall x \exists y\, R(x, y)$ is valid. Do it in two different ways:
   (a) proof-theoretically, $\vdash \varphi$, using natural deduction, and
   (b) semantically, $\models \varphi$.

[32] Review the definition of $\sigma[x \mapsto a]$ in Section B.3.

2. Show that the sentence $\psi \stackrel{\text{def}}{=} \forall x \exists y\, R(x,y) \to \forall x\, R\big(x, f(x)\big)$ is not valid. Note that ψ is just the converse implication of φ.

   Hint: Try a semantic approach, *i.e.*, show $\not\models \psi$. You need to define a structure $\mathcal{A}$ so that the left-hand side of "$\to$" in ψ is true in $\mathcal{A}$ but the right-hand side of "$\to$" is false in $\mathcal{A}$.

3. Conclude that $\forall x \exists y\, R(x,y)$ and $\forall x\, R\big(x, f(x)\big)$ are not equivalent first-order wff's.

Remark: Despite the conclusion in part 3, Proposition 91 asserts that $\forall x \exists y\, R(x,y)$ and $\forall x\, R\big(x, f(x)\big)$ are equisatisfiable, *i.e.*, if there is a model for one, then there is a model for the other, and vice-versa. $\square$

F Alternative Proofs of Compactness

We present two alternative proofs of Compactness for *propositional logic*. At bottom, these are not "new" proofs, but different presentations of the same fundamental idea (or topological core, if you will) underlying the proof of Theorem 2 in Section 1. This fundamental idea is what *König's Lemma* asserts. The difference here is that they make the connection with topology a little more explicit by naming and presenting the same key concepts differently. One can read the first alternative proof as an elaboration of the proof in Section 1, and the second alternative proof as an elaboration of the first.[33]

Lemma 94 (König's Lemma)**.** *Every infinite, finitely branching, tree $\mathcal{T}$ has an infinite path.*

Proof. Using induction, we define an infinite sequence of nodes $\alpha_0, \alpha_1, \ldots$, forming an infinite path in $\mathcal{T}$. At stage 0 of the induction, let α_0 be the root node of $\mathcal{T}$, which has infinitely many successors by the hypothesis that $\mathcal{T}$ is infinite. At every stage $n \geqslant 1$, assume we have already selected nodes $\alpha_0, \alpha_1, \ldots, \alpha_{n-1}$ so far, forming a path of length $(n-1)$, such that α_{n-1} has infinitely many successors. By hypothesis, $\mathcal{T}$ is finitely branching, which implies α_{n-1} has only finitely many immediate successors. Hence, one of the immediate successors of α_{n-1}, say β, must have infinitely many successors. Define α_n to be β, which has infinitely many successors in $\mathcal{T}$, and proceed to stage $n+1$ of the induction. $\square$

The preceding proof is not constructive: We do not have an algorithm to select the next node β at stage n after having already selected nodes $\alpha_0, \alpha_1, \ldots, \alpha_{n-1}$. We only know that one of the immediate successors of α_{n-1} is the root node of an infinite subtree; we know that it exists, but we do not know which one it is. So, at every stage we invoke what is called the Axiom of Choice to select the next node β.

For the next alternative proof of Compactness for PL, we specialize *König's Lemma* (KL) to the case of binary trees, where every node has exactly two successors. The *full binary tree* is a tree without leaf nodes, which therefore has $2^{\aleph_0}$ distinct infinite paths. Every binary tree can be viewed as an initial fragment of the full binary tree, *i.e.*, by inserting a copy of the full binary tree at every leaf node of the former.

Another way of stating KL relative to binary trees is to say: *If a binary tree has arbitrarily long full finite paths, then it has an infinite path*, which is the form we use in the next proof. By a "full finite path" we mean a path that starts at the root node and ends at a leaf node. This form of KL specialized to binary trees is sometimes called *Weak König's Lemma* (WKL).

The next exercise is a little application of WKL, which has a distinctly topological flavor.

Exercise 95 (*Sequential Compactness*)**.** We write $A = [0, 1]$ for the closed interval of all real numbers between 0 and 1. Let $\boldsymbol{a} \stackrel{\text{def}}{=} (a_n \mid n \in \mathbb{N})$ be an infinite sequence of elements in A. Show there is an infinite subsequence $\boldsymbol{a}'$ of $\boldsymbol{a}$, say $\boldsymbol{a}' \stackrel{\text{def}}{=} (a_k \mid k \in K)$ where $K \subseteq \mathbb{N}$, such that $\boldsymbol{a}'$ converges to an element $b \in A$. (Take the elements of the sequence $\boldsymbol{a}$ and subsequence $\boldsymbol{a}'$ to be listed in order of increasing indices.)

Alternative Proof I of Theorem 2 (Compactness for Propositional Logic). As in the earlier proof of Theorem 2 in Section 1, we only need to consider the non-trivial implication "$\Leftarrow$": If Γ is finitely satisfiable, then Γ is satisfiable.

[33] And there are still other proofs with a decidedly algebraic or topological content. A particular construction nicely complementing the material in this appendix is Łoś's Theorem which proves Compactness using what are called *ultrafilters* and *ultraproducts*. Search the Web for "propositional compactness via ultraproducts" and "first-order compactness via ultraproducts".

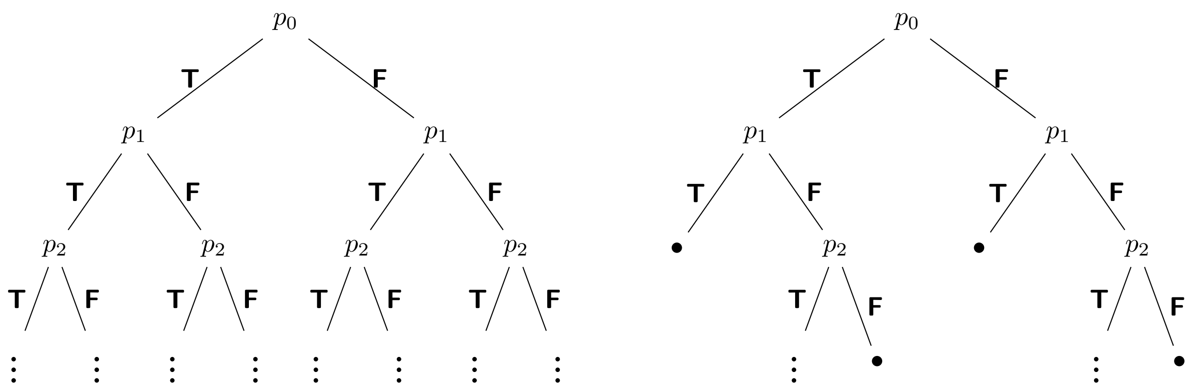

Figure 8: *On the left*: The top three levels of the full binary tree $\mathcal{T}_{\text{full}}$, with each left/right edge labelled T/F. *On the right*: An example of how $\mathcal{T}_{\text{full}}$ is pruned if p_0 occurs nowhere in Γ (and therefore has no effect on the satisfiability of Γ) and Γ contains the wff $(\neg p_1 \wedge p_2)$. Starting from the root, if a path reaches a leaf node "$\bullet$", the corresponding truth assignment is doomed to falsify Γ.

The set $\mathcal{P}$ of propositional variables is countably infinite: $\{p_0, p_1, p_2, \ldots\}$. Assume a fixed ordering of $\mathcal{P}$, in the order of their indices $0, 1, 2, \ldots$. We view a truth assignment $\sigma : \mathcal{P} \to \{\mathsf{T}, \mathsf{F}\}$ as defining an infinite path in the full binary tree, call it $\mathcal{T}_{\text{full}}$, by using T as the label of every left edge and F as the label of every right edge. See left of Figure 8 for a partial graphic representation of $\mathcal{T}_{\text{full}}$.

From $\mathcal{T}_{\text{full}}$ we define another binary tree $\mathcal{T}(\Gamma)$ by pruning some of the infinite paths as follows: Given infinite path $\sigma \stackrel{\text{def}}{=} t_0 t_1 t_2 \cdots t_n \cdots$ in $\mathcal{T}_{\text{full}}$ where $t_n \in \{\mathsf{T}, \mathsf{F}\}$ for every $n \geqslant 0$, let k be the smallest integer (if any) such the truth assignment corresponding to σ falsifies some wff in Γ; if such a k exists, delete from $\mathcal{T}_{\text{full}}$ all paths extending the finite path $t_0 t_1 \cdots t_k$. At the node where $\mathcal{T}_{\text{full}}$ is pruned, we replace p_{k+1} by a leaf node denoted "$\bullet$". See right of Figure 8 for an example of how $\mathcal{T}_{\text{full}}$ is pruned when p_0 occurs nowhere in Γ and Γ includes the wff $(\neg p_1 \wedge p_2)$. By this definition of $\mathcal{T}(\Gamma)$, note that $\mathcal{T}_{\text{full}}$ is none other than $\mathcal{T}(\varnothing)$, which is $\mathcal{T}_{\text{full}}$ without any pruning.

The resulting $\mathcal{T}(\Gamma)$ contains some full finite paths (possibly none) and some infinite paths (possibly none). Γ is satisfiable iff $\mathcal{T}(\Gamma)$ contains an infinite path, so that also Γ is not satisfiable iff $\mathcal{T}(\Gamma)$ does not contain an infinite path. By WKL, if $\mathcal{T}(\Gamma)$ does not contain an infinite path, then $\mathcal{T}(\Gamma)$ does not contain arbitrarily long full finite paths, *i.e.*, there is a finite bound $k \geqslant 1$ such that all full finite paths have length $\leqslant k$. But this implies there is a finite subset of Γ which is not satisfiable. $\square$

Exercise 96. This exercise is couched in a language a little more familiar to computer scientists. Given a set $\Gamma \subseteq \mathsf{WFF}_{\mathsf{PL}}(\mathcal{P})$, the binary tree $\mathcal{T}(\Gamma)$ induced by Γ is defined in the proof above. An arbitrary binary tree $\mathcal{U}$ can be represented by a subset of $\{\mathsf{T}, \mathsf{F}\}^*$, which denotes the set of all finite strings over the alphabet $\{\mathsf{T}, \mathsf{F}\}$, satisfying two conditions:

- $\mathcal{U}$ is prefix-closed, *i.e.*, for all $\pi_1, \pi_2 \in \{\mathsf{T}, \mathsf{F}\}^*$, if $\pi_1 \in \mathcal{U}$ and π_2 is a prefix of π_1, then $\pi_2 \in \mathcal{U}$.
- For all $\pi \in \{\mathsf{T}, \mathsf{F}\}^*$, it holds that $\pi\mathsf{T} \in \mathcal{U}$ iff $\pi\mathsf{F} \in \mathcal{U}$, *i.e.*, every non-leaf node has two successors.

Let Γ_0, Γ_1, and Γ_2, be defined as follows:

$$\Gamma_0 \stackrel{\text{def}}{=} \{\, \neg p_1 \wedge p_2 \},$$

$$\Gamma_1 \stackrel{\text{def}}{=} \Big\{\, p_1 \to \neg(p_2 \wedge \cdots \wedge \neg p_k) \;\Big|\; k \geqslant 2 \Big\},$$

$$\Gamma_2 \stackrel{\text{def}}{=} \Big\{\, \neg p_1 \to \neg(p_2 \wedge \cdots \wedge \neg p_k) \;\Big|\; k \geqslant 2 \Big\}.$$

There are two parts in this exercise:

1. Define the binary trees $\mathcal{T}(\Gamma_0)$, $\mathcal{T}(\Gamma_1 \cup \Gamma_2)$, and $\mathcal{T}(\Gamma_0 \cup \Gamma_1 \cup \Gamma_2)$, as subsets of $\{\mathsf{T}, \mathsf{F}\}^*$.
2. For each of the three binary trees defined in part 1, explain how the tree indicates whether the corresponding set of wff's is satisfiable or not.

You will find it useful to consult Figure 8. $\square$

The next proof makes explicit reference to notions in topology. As indicated in the preceding alternative proof, a truth assignment σ can be denoted by a path in the full binary tree $\mathcal{T}_{\text{full}}$, now viewed as an ω-sequence in the product space $\{\mathsf{T}, \mathsf{F}\}^\omega$. (We write ω for the first infinite ordinal, which is the set of natural numbers listed in their standard order.)

We view $\{\mathsf{T}, \mathsf{F}\}^\omega$ as the underlying space of a product topology $\big(\{\mathsf{T}, \mathsf{F}\}^\omega, \mathcal{O}\big)$, thus making every truth assignment a "point" in that topology. $\mathcal{O}$ is a family of *open sets* that define the topology, which are in this case subsets of points in $\{\mathsf{T}, \mathsf{F}\}^\omega$ and satisfy the usual requirements of a topology:

- The empty set $\varnothing$ and the full space $\{\mathsf{T}, \mathsf{F}\}^\omega$ are in $\mathcal{O}$,
- $\mathcal{O}$ is closed under *arbitrary* unions,
- $\mathcal{O}$ is closed under *finite* intersections.

We can define a subset $U \subseteq \{\mathsf{T}, \mathsf{F}\}^\omega$ to be *open* iff there is a finite set of indices $I \subseteq \omega$ such that:

$$U \;=\; \prod \left\{\, A_i \;\middle|\; i \in \omega \text{ and } A_i \subseteq \{\mathsf{T}, \mathsf{F}\} \,\right\} \quad \text{where } A_i = \{\mathsf{T}, \mathsf{F}\} \text{ for every } i \in \omega - I.$$

In words, in the infinite product $U = A_0 \times A_1 \times \cdots \times A_i \times \cdots$, for all but finitely many indices i it is the case that $A_i = \{\mathsf{T}, \mathsf{F}\}$. A set $U \subseteq \{\mathsf{T}, \mathsf{F}\}^\omega$ is *closed* iff it is the complement of an open set. In the case of the product topology, every open subset $U \subseteq \{\mathsf{T}, \mathsf{F}\}^\omega$ is also closed, and thus called *clopen*.

Let A be a subset of $\{\mathsf{T}, \mathsf{F}\}^\omega$. An *open covering of* A is a family of open sets $\{\, U_i \mid i \in I \,\} \subseteq \mathcal{O}$ such that $A \subseteq \bigcup\{\, U_i \mid i \in I \,\}$. And A is said *compact* if every open covering of A has a finite subcovering; *i.e.*, there exists a finite subfamily $U_{i_1}, U_{i_2}, \ldots, U_{i_n}$ of $\{\, U_i \mid i \in I\}$ such that $A \subseteq (U_{i_1} \cup U_{i_2} \cup \ldots \cup U_{i_n})$. By Tychonoff's Theorem in topology, the product topology $\big(\{\mathsf{T}, \mathsf{F}\}^\omega, \mathcal{O}\big)$ is *compact*, which means that every open covering of a subset of points $A \subseteq \{\mathsf{T}, \mathsf{F}\}^\omega$ has a finite subcovering.

Alternative Proof II of Theorem 2 (Compactness for Propositional Logic). Again here, we only need to consider the non-trivial implication "$\Leftarrow$": If Γ is finitely satisfiable, then Γ is satisfiable.

For every propositional wff φ, let $A_\varphi \subseteq \{\mathsf{T}, \mathsf{F}\}^\omega$ be the collection of all points/truth assignments that satisfy φ. The set A_φ is a closed (and open) subset of $\{\mathsf{T}, \mathsf{F}\}^\omega$ in the topology $\big(\{\mathsf{T}, \mathsf{F}\}^\omega, \mathcal{O}\big)$, which follows from the fact that φ only mentions finitely many propositional variables. It is easy to check that, for every finite subset Δ of Γ, if Δ is satisfiable, then $\bigcap\{\, A_\varphi \mid \varphi \in \Delta \,\}$ is not empty. Hence, the family of closed sets $\{\, A_\varphi \mid \varphi \in \Gamma \,\}$ satisfies the *finite intersection property*. Moreover, the product topology $\big(\{\mathsf{T}, \mathsf{F}\}^\omega, \mathcal{O}\big)$ is compact, as noted above. Hence, $\bigcap\{\, A_\varphi \mid \varphi \in \Gamma \,\}$ is not empty, which is the desired conclusion. $\square$