1.8 The Compactness Theorem

We now return to propositional logic. In this section we will prove a basic result known as the Compactness Theorem and some applications of it.

1.8.A The Compactness Theorem

The compactness theorem has two equivalent versions.

Theorem 1.8.1 (Compactness Theorem I) *Let S be any set of formulas in propositional logic. If every finite subset $S_0 \subseteq S$ is satisfiable, then S is satisfiable.*

Theorem 1.8.2 (Compactness Theorem II) *Let S be any set of formulas in propositional logic, and A any formula. Then if $S \models A$, there is a* finite *subset $S_0 \subseteq S$ such that $S_0 \models A$.*

First we show that these two forms are equivalent.

I implies II. Assume I holds *for any S*. Fix then S and A with $S \models A$. Then $S' = S \cup \{\neg A\}$ is not satisfiable, so, by applying I to S', we have a finite $S'_0 \subseteq S' = S \cup \{A\}$, so that S'_0 is not satisfiable. Say $S'_0 \subseteq S_0 \cup \{\neg A\}$, with $S_0 \subseteq S$ finite. Then $S_0 \cup \{\neg A\}$ is not satisfiable, so $S_0 \models A$.

II implies I. Say S is not satisfiable. Then $S \models \perp$, where $\perp = p \wedge \neg p$. So by II $S_0 \models \perp$ for some finite $S_0 \subseteq S$. Then S_0 is unsatisfiable.

We will now prove form I of the Compactness Theorem.

Proof of 1.8.1. We assume that every finite subset $S_0 \subseteq S$ is satisfiable. We will then build a finite splitting tree which is infinite and thus, by König's Lemma, has an infinite branch. This infinite branch will give a truth assignment satisfying S.

First let us notice that we can enumerate in a sequence $A_1, A_2, A_3, \dots, A_n, \dots$ all wff. So we can enumerate in a sequence

$$S = \{B_1, B_2, B_3, \dots, B_n, \dots\}$$

all wff in S. Next $k_1 < k_2 < k_3 < \cdots < k_n < \dots$ are chosen so that

$$B_n = B_n(p_1, \dots, p_{k_n})$$

i.e, all the propositional variables of B_n are among $p_1, \dots, p_{k_n}$.

We will now build a tree as follows: Let v_0 be the root. The children of v_0 are all valuations $v_1 = (\epsilon_1, \ldots, \epsilon_{k_1}) \in \{0,1\}^{k_1}$ which satisfy B_1. Fix any such v_1, say $v_1 = (\epsilon_1, \ldots, \epsilon_{k_1})$. Its children are all valuations $v_2 = (\epsilon_1, \ldots, \epsilon_{k_1}, \epsilon_{k_1+1}, \ldots, \epsilon_{k_2}) \in \{0,1\}^{k_2}$, which agree with v_1 in their first k_1 values and also satisfy both B_1 and B_2, etc.

First we argue that this tree is infinite: Fix any $n \geq 1$. By assumption there is a valuation $v_n = \{\epsilon_1, \ldots, \epsilon_{k_n}\} \in \{0,1\}^{k_n}$ which satisfies $\{B_1, \ldots, B_n\}$. Let for $1 \leq m \leq n$, $v_m = \{\epsilon_1, \ldots, \epsilon_{k_m}\}$, i.e., the restriction of v_n to the first k_m variables. Then clearly v_m satisfies $\{B_1, \ldots, B_m\}$, so it is a vertex of our tree, and v_{m+1} is a child of v_m. So T has at least n vertices for each n, i.e., it is infinite. Clearly T is finite splitting. So, by König's Lemma, it has an infinite branch $v_0, v_1, v_2, \ldots, v_n, \ldots$ Then

$$\begin{aligned}
v_1 &= \{\epsilon_1, \ldots, \epsilon_{k_1}\} \\
v_2 &= \{\epsilon_1, \ldots, \epsilon_{k_1}, \epsilon_{k_1+1}, \ldots, \epsilon_{k_2}\} \\
&\vdots \\
v_n &= \{\epsilon_1, \ldots, \epsilon_{k_1}, \epsilon_{k_1+1}, \ldots, \epsilon_{k_2}, \epsilon_{k_2+1}, \ldots, \epsilon_{k_n}\} \\
&\vdots
\end{aligned}$$

So if $\nu = \{\epsilon_1, \epsilon_2, \ldots\}$, ν is a valuation which satisfies all the B_n, i.e., it satisfies S, and so S is satisfiable. $\dashv$

1.8.B A Proof of König's Lemma

We have proved the Compactness Theorem by using König's Lemma. On the other hand, we can also prove König's Lemma by using the Compactness Theorem as follows:

Consider a tree T with root v_0 which is infinite, but has finite splitting. Notice then that the set V of the vertices of T can be enumerated in a sequence. Introduce now a propositional variable p_v for each vertex v of T. (Our preceding remark implies that we can enumerate these variables in a sequence $p_1, p_2, \ldots$, but there is no point in doing that explicitly.) Consider now the following set S of wff, which we can view as "axioms", where the intuitive meaning of the variable p_v is that "p_v is true" iff "v is in the infinite branch we try to find".

(i) p_{v_0}

(ii) $p_v \Rightarrow \neg p_u$, if $v \neq u$ are at the same level or for some n, v is at level n, u is at level $n+1$ and u is not a child of v.

(iii) If for each n, $u_1, \dots, u_{k_n}$ are all the vertices at level n, then we introduce the wff $p_{u_1} \vee p_{u_2} \vee \cdots \vee p_{u_{k_n}} (n = 1, 2, \dots)$. (Here u is *at level* n if the path from v_0 to v has length n.)

Assume that this set S of wff is satisfiable, say by the valuation ν. Then consider the following set of vertices

$$v \in P \text{ iff } \nu(p_v) = 1.$$

We claim that P is an infinite branch, i.e., P contains v_0 and exactly one vertex at level $n = 1, 2, \dots$, say v_n, so that v_{n+1} is a child of v_n. First by (i), $v_0 \in P$. By (iii), for each $n \geq 1$, there is at least one vertex v at level n with $v \in P$ and by (ii) there is exactly one, say v_n. It only remains to prove that v_{n+1} is a child of v_n. Otherwise, by (ii), $\nu(p_{v_n} \Rightarrow \neg p_{v_{n+1}}) = 1$, but also $\nu(p_{v_n}) = \nu(p_{v_n+1}) = 1$, a contradiction.

By the Compactness Theorem, it only remains to show that every finite subset S_0 of S is satisfiable. Fix such an S_0. Then for some large enough N_0, all the "axioms" (i), (ii), (iii) occurring in S_0 contain variables p_v with v at a level $n \leq N_0$. Since T is infinite, every level is nonempty, so fix a node v_{N_0} at level N_0 and look at the path $v_0, v_1, \dots, v_{N_0-1}, v_{N_0}$ from v_0 to v_{N_0}. Consider then the valuation to all the variables p_v with v at a level $n \leq N_0$ defined as follows:

$$\nu(P_v) = 1 \text{ iff } v \text{ is one of } v_0, v_1, \dots, v_{N_0}.$$

Then all the "axioms" (i)-(iii) belonging to S_0 are satisfied, i.e., S_0 is satisfiable, and the proof is complete. $\dashv$

1.8.C Partial Orders

We will give another application of the Compactness Theorem involving partial orders.

Definition 1.8.3 A *partial order* on a set X is a relation $x < y$ between members of X such that it satisfies the following properties:

(i) If $x < y$, then $y \not< x$ (in particular $x \not< x$),

(ii) $x < y$ and $y < z$ imply $x < z$.

Definition 1.8.4 A partial order is called a *linear order* (or *total order*) if it also satisfies:

(iii) $x < y$ or $y < x$ or $x = y$.

Examples 1.8.5

(i) Let $X = P(A) =$ set of all subsets of A, and

$$x < y \text{ iff } x \subseteq y \text{ and } x \neq y.$$

This is a partial order but is not linear, if A has more than one element.

(ii) Let $X = \mathbb{N}$ and $<$ be the usual order of the integers. Then $<$ is a linear order.

Definition 1.8.6 We say that a linear order $<$ on a set X *extends* a partial order $<'$ on X if $x <' y$ implies $x < y$.

Example 1.8.7 On $\mathbb{N}^+ = \{1, 2, 3, \dots\}$ consider the following two partial orders:

(i) $<$ is the usual order (so it is linear).

(ii) $x <' y$ iff $x \neq y$ and x divides y (this is not linear).

Then $<$ extends $<'$. If on the other hand we consider

(iii) $x <' y$ iff (x is even and y is odd) or (x, y are even and $x < y$),

then $<$ does not extend $<'$.

Theorem 1.8.8 *Let $X = \{x_1, x_2, \dots\}$ be a countable set and $<'$ a partial order on X. Then $<'$ can be extended to a linear order $<$.*

Proof. First we note the following finite version of this theorem.

Lemma 1.8.9 *Let $X = \{x_1, \dots, x_n\}$ be a finite set. Then every partial order on X can be extended to a linear order.*

This can be proved by induction, and we will give the proof at the end.

Consider now the infinite case, where $X = \{x_1, x_2, \dots\}$, $x_i \neq x_j$, if $i \neq j$. Introduce for each pair (i, j), $i = 1, 2, \dots$, $j = 1, 2, \dots$, a propositional variable $p_{i,j}$. The intuitive meaning is that "$p_{i,j}$ is true" if "$x_i < x_j$ in the linear order $<$ we try to find". Let S consist of the following "axioms":

(i) $p_{i,j} \Rightarrow \neg p_{j,i}$ for all i, j;

(ii) $p_{i,j} \wedge p_{j,k} \Rightarrow p_{i,k}$, for all i, j, k;

(iii) $p_{i,j} \vee p_{j,i}$, for all $i \neq j$;

(iv) $p_{i,j}$, whenever $x_i <' x_j$.

Suppose S is satisfiable by some valuation ν. Then define the following relation on X:

$$x_i < x_j \text{ iff } \nu(p_{i,j}) = 1.$$

By (i)-(iii), $<$ is a linear order and by (iv) it extends $<'$. To show that S is satisfiable, it is enough, by the Compactness Theorem, to show that every finite subset $S_0 \subseteq S$ is satisfiable. Fix such an S_0. Then for some large enough N_0, S_0 contains only variables $p_{i,j}$ with $i, j \leq N_0$. Let $X_0 = \{x_1, \dots, x_{N_0}\}$ and restrict the partial order $<'$ to X_0. Call it $<'_{X_0}$. By the lemma, there is a linear order $<_{X_0}$ on X_0 extending $<'_{X_0}$. Use this to define the following valuation to the variables $p_{i,j}$ for $i, j \leq N_0$:

$$\nu(p_{i,j}) = 1 \text{ iff } x_i <_{X_0} x_j.$$

Then all the axioms (i)-(iv) in S_0 are satisfied, i.e., S_0 is satisfiable.

It remains to give the proof of the lemma: Let n be the cardinality of X. We prove this by induction on n. If $n = 1$ it is obvious, as the only partial order on a set of cardinality 1 is linear. So assume X has cardinality $n + 1$. Consider a partial order $<'$ on X. Since X is finite, it has a minimal element, say x_0 (x is *minimal* if there is no $y <' x$). Let $x_1, \dots, x_n$ be the rest of the elements of X. Restrict $<'$ to $\{x_1, \dots, x_n\}$ and by induction hypothesis extend this to a linear order $\prec$ on $\{x_1, \dots, x_n\}$. Then define a linear order $<$ on $\{x_0, x_1, \dots, x_n\}$ by letting

$$x_i < x_j \text{ iff } x_i \prec x_j, \text{ for } 1 \leq i, j \leq n$$

and

$$x_0 < x_i, \text{ for } 1 \leq i \leq n$$

(i.e., the order $<$ agrees with $\prec$ on $\{x_1, \ldots, x_n\}$ and x_0 is $<$ than any element of $\{x_1, \ldots x_n\}$.) $\dashv$

1.9 Ramsey Theory

Let $\mathbb{N}$ denote the set of natural numbers, $\mathbb{N} = \{0, 1, 2, \ldots\}$. For $0 < m, l, k, n \in \mathbb{N}$, set $\mathbf{m} = \{1, 2, \ldots, m\}, \mathbf{l} = \{1, 2, \ldots, l\}$ and let $m \to (n)^k_l$ be the following assertion.

Whenever $f : [\mathbf{m}]^k \to \mathbf{l}$, there is $H \in [\mathbf{m}]^n$ homogeneous for f.

Similarly, let $\mathbb{N} \to (\mathbb{N})^k_l$ mean that for every $f : [\mathbb{N}]^k \to \mathbf{l}$ there is $H \subseteq \mathbb{N}$ infinite and homogeneous for f. Here,

- $[X]^k$ is the collection of k-sized subsets of X.
- Given $f : [X]^k \to \mathbf{l}$, $H \subseteq X$ is *homogeneous* for f iff whenever $s, t \in [H]^k$, then $f(s) = f(t)$.

The classic Ramsey Theorem was proved in 1928. Frank Plumpton Ramsey was born in 1903 in Cambridge, and died in 1930 in London as a result of an attack of jaundice. An enthusiastic logician, he considered mathematics to be part of logic. His second paper, *On a problem of formal logic* was read to the London Mathematical Society on 13 December 1928 and published posthumously in the Proceedings of the London Mathematical Society in 1930. There he proves Ramsey Theorem and uses it to deduce a result on propositional logic.

1.9.A Ramsey Theorem, infinite version

Theorem 1.9.1 *For all $0 < k, l \in \mathbb{N}$, $\mathbb{N} \to (\mathbb{N})^k_l$. In English: Given any $f : [\mathbb{N}]^k \to \mathbf{l}$ there is an infinite subset of $\mathbb{N}$ homogeneous for f.*

Proof. The proof is by induction on k. For $k = 1$ the result is obvious; it simply says that if an infinite set is partitioned into finitely many pieces, one of the pieces is itself infinite, here we are identifying a number n with the singleton $\{n\}$.

Assume we know the result for k and we are given a function $f : [\mathbb{N}]^{k+1} \to \mathbf{l}$. Clearly, if X is infinite and $h : [X]^k \to \mathbf{l}$, then there is an infinite $Y \subseteq X$ homogeneous for h. This simple observation (that we can replace $\mathbb{N}$ with any infinite set) is key to the argument.

We start by defining a decreasing sequence of infinite subsets of $\mathbb{N}$, $A_1 \supset A_2 \supset A_3 \supset \cdots$ with the property that if $a_n = \min A_n$ then $a_1 < a_2 < \cdots$.

Let $A_1 = \mathbb{N}$ (so $a_1 = 1$). In general, given A_n, define $f_n : [A_n \setminus \{a_n\}]^k \to \mathbf{l}$ by setting $f_n(s) = f(\{a_n\} \cup s)$ and use the inductive hypothesis to find $A_{n+1} \subseteq A_n \setminus \{a_n\}$ infinite and homogeneous for f_n.

Now consider the set $A = \{a_1, a_2, \dots\}$. Notice that, by construction, if $s, t \in [A]^{k+1}$ and $a_n = \min(s) = \min(t)$ then $s \setminus \{a_n\}, t \setminus \{a_n\} \in [A_{n+1}]^k$, so $f_n(s \setminus \{a_n\}) = f_n(t \setminus \{a_n\})$ (since A_{n+1} is homogeneous for f_n), i.e., $f(s) = f(t)$. This means that $f(s)$ only depends on $\min(s)$ for any $s \in [A]^{k+1}$ (usually one says that A is *min-homogeneous* for f).

Consider now the function $g : A \to \mathbf{l}$ given by $g(a) = f(s)$ where s is *any* element of $[A]^{k+1}$ with $\min(s) = a$. By the case $k = 1$, there is an infinite $B \subset A$ homogeneous for g. But then it is easy to see that B is also homogeneous for f. $\dashv$

Remark. Another proof of Ramsey theorem can be obtained from König's lemma. Instead, we use König's lemma to deduce a finite version of Ramsey's result.

1.9.B Ramsey Theorem, finite version

Theorem 1.9.2 *For all $n, k, l \in \mathbb{N}$ there is m such that $m \to (n)^k_l$. In English: Given any n, k, l there is m sufficiently large that whenever $f : [\mathbf{m}]^k \to \mathbf{l}$, there is $H \subset \mathbf{m}$, $|H| = n$ that is homogeneous for f.*

Proof. We provide a proof using König's lemma and the infinite version of Ramsey theorem. Suppose towards a contradiction that for some fixed values of n, k, l there is no m as required. This means that for every m there is a function $f : [\mathbf{m}]^k \to \mathbf{l}$ without homogeneous sets of size n. Clearly, if $m_1 < m_2$ and $f : [\mathbf{m}_2]^k \to \mathbf{l}$ has no such homogeneous sets, then f extends a function g with domain $[\mathbf{m}_1]^k$ with the same property, and any such function f is an extension of some such function g. So we can define a rooted tree $\mathcal{T}$ as follows: Any node of the tree at level m is a function $f : [\mathbf{m}]^k \to \mathbf{l}$ without homogeneous sets of size n. The children of f are all the functions

$h : [\{1, \dots, m+1\}]^k \to \mathbf{l}$ extending f and with the same property. Clearly, $\mathcal{T}$ is finite splitting and our assumption implies that it is infinite. By König's lemma it has an infinite branch, which consists of functions with longer and longer domains that cohere. We can thus take their union and obtain a function $F : [\mathbb{N}]^k \to \mathbf{l}$. It is easy to see that F admits no homogeneous sets of size n. But this is impossible, by the infinite version of Ramsey Theorem. Contradiction. $\dashv$

Remark. This kind of argument is very powerful, but it has an important drawback. Namely, it is nonconstructive. There is no known method that allows us to extract from this proof for given n, k, l a bound on how large m must be. Different, more finitistic, proofs can be given that provide explicit bounds. The computation of actual values of m, the so-called *Ramsey numbers* (as opposed to mere upper or lower bounds) is very much an open problem.

For the application to logic that the theorem was originally conceived for, and many other applications of this method to combinatorics, we suggest to look at R. Graham, B. Rothschild, J. Spencer, **Ramsey Theory**, 2nd ed., Wiley, New-York, 1990.

In general one refers to this use of König's lemma that serves as a bridge between infinite and finite statements as a *compactness* argument.

1.9.C A few other applications of compactness

Say that a function $f : [\mathbf{m}]^k \to \{1, \dots, m-k\}$ is *regressive* iff $f(s) < \min(s)$ for all $s \in [\mathbf{m}]^k$. Similarly we can talk of a function $f : [\mathbb{N}]^k \to \mathbb{N}$ being regressive. Say that H is *min-homogeneous* for f iff $f(s) = f(t)$ whenever $s, t \in [H]^k$ and $\min(s) = \min(t)$.

If $f : [\mathbb{N}]^k \to \mathbb{N}$ is regressive, it does not need to admit infinite homogeneous sets. For example, let $f(s) = \min(s)$. However, it does admit min-homogeneous sets.

Theorem 1.9.3 *If $X \subseteq \mathbb{N}$ is infinite and $f : [X]^k \to \mathbb{N}$ is regressive, there is an infinite subset of X min-homogeneous for f.*

The proof is an easy modification of the proof of the infinite version of Ramsey Theorem given above, and it is a useful exercise to work out its details.

Using König's lemma very much as in the proof of the finite version of Ramsey Theorem, one obtains:

Corollary 1.9.4 *For all k and n there is m such that if*

$$f : [\mathbf{m}]^k \to \{1, \dots, m-k\}$$

is regressive, then it admits a min-homogeneous set of size n.

This was proven by Kanamori and McAloon in the late 80s. However, unlike the finite version of Ramsey Theorem, *there is no* purely finitistic proof of this result. This was also proven by Kanamori and McAloon, using methods of mathematical logic. See *On Gödel incompleteness and finite combinatorics*, Annals of Pure and Applied Logic **33 (1)** (1987), 23–41.

Suppose now that $X \subset \mathbb{N}$ and that $f : [X]^k \to \mathbf{l}$. Let H be a finite subset of X homogeneous for f. Say that H is *large-homogeneous* iff $\min(H) \le |H|$.

Theorem 1.9.5 *For all k, l, n there is m sufficiently large that any $f : [\mathbf{m}]^k \to \mathbf{l}$ admits a large-homogeneous set of size n.*

The proof of this result can be obtained by a König's lemma argument from an appropriate infinite version, whose proof is again an easy modification of the argument we gave for the infinite version of Ramsey Theorem. Again, it is not possible to prove this result by purely finitistic methods. This was shown by Harrington and Paris in *A mathematical incompleteness in Peano Arithmetic*, in **Handbook of Mathematical Logic**, Jon Barwise ed., North-Holland, 1977.

1.10 The Resolution Method

We would like to have a mechanical procedure (algorithm) for checking whether a given set of formulas logically implies another, that is, given $A_1, \dots, A_n, A$, whether

$$A_1, \dots, A_n \models A.$$

We know that this happens if and only if

$$\models (A_1 \wedge \cdots \wedge A_n) \Rightarrow A$$

which happens iff

$$A_1 \wedge \cdots \wedge A_n \wedge \neg A \text{ is unsatisfiable.}$$

So it suffices to have an algorithm to check the (un)satisfiability of a single wff. The method of truth tables gives one such algorithm. We will now develop another method which is often (with various improvements) more efficient in practice.

It will be also an example of a *formal calculus*. By that we mean a set of rules for generating a sequence of strings in a language. Formal calculi usually start with a certain string or strings as given, and then allow the application of one or more "rules of production" to generate other strings.

We have already encountered some formal calculi in the form of recursive definitions. Definition 1.2.6 (Well-Formed Formulas) gives a formal calculus consisting of certain given wffs (propositional variables) and certain rules for producing new wffs (negation and adding binary connectives). We will see yet another example of a formal calculus in section 1.11.A.

Suppose A is a wff which we want to test for satisfiability. First we note that although there is no known efficient algorithm for finding a wff A' in cnf (conjunctive normal form) equivalent to A, it is not hard to show that there is an efficient algorithm for finding a wff A^* in cnf such that:

$$A \text{ is satisfiable iff } A^* \text{ is satisfiable.}$$

This will be discussed in Assignment #4, Problem 1.

So from now on we will only consider wff in cnf, and the Resolution Method applies to such formulas only. Say

$$A = (\ell_{1,1} \vee \cdots \vee \ell_{1,n_1}) \wedge \cdots \wedge (\ell_{k,1} \vee \cdots \vee \ell_{k,n_k})$$

with $\ell_{i,j}$ literals. Since order and repetition in each conjunct

$$\ell_{i,1} \vee \cdots \vee \ell_{i,n_i} \tag{$*$}$$

are irrelevant (for semantic purposes), we can replace $(*)$ by the set of literals

$$c_i = \{\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,n_i}\}.$$

Such a set of literals is called a *clause*. It corresponds to the formula $(*)$. So the wff A above can be simply written as a set of clauses (again since the order of the conjunctions is irrelevant)

$$\begin{aligned} C &= \{c_1, \ldots, c_k\} \\ &= \{\{\ell_{i,1}, \ldots \ell_{i,n_1}\}, \ldots, \{\ell_{k,1}, \ldots, \ell_{k,n_k}\}\} \end{aligned}$$

Satisfiability of A means then simultaneous satisfiability of all of its clauses $c_1, \ldots, c_k$, i.e., finding a valuation ν which makes c_i true for each i, i.e., which for each i makes some $\ell_{i,j}$ true.

Example 1.10.1

$$\begin{aligned} A &= (p_1 \vee \neg p_2) \wedge (p_3 \vee p_3) \\ c_1 &= \{p_1, \neg p_2\} \\ c_2 &= \{p_3\} \\ C &= \{\{p_1, \neg p_2\}, \{p_3\}\}. \end{aligned}$$

From now on we will deal only with a set of clauses $C = \{c_1, c_2, \ldots\}$, which we would even allow to be infinite. Satisfying C means (again) that there is a valuation which satisfies all $c_1, c_2, \ldots$, i.e. if $c_i = \ell_{i,1} \vee \cdots \vee \ell_{i,n_i}$, then for all i there is j so that it makes $\ell_{i,j}$ true. (Of course in the case C comes from some wff A, C is a finite set of clauses.)

Notice that if the set of clauses C_A is associated as above to A (in cnf) and C_B to B, then

$$A \wedge B \text{ is satisfiable iff } C_A \cup C_B \text{ is satisfiable.}$$

By convention we also have the *empty clause* $\square$, which contains no literals. The empty clause is (by definition) unsatisfiable, since for a clause to be satisfied by a valuation, there has to be some literal in the clause which it makes true, but this is impossible for the empty clause, which has no literals.

For a literal u, let $\bar{u}$ denote its "conjugate", i.e.

$$\begin{aligned} \bar{u} &= \neg p, \text{ if } u = p, \\ \bar{u} &= p \text{ if } u = \neg p. \end{aligned}$$

Definition 1.10.2 Suppose now c_1, c_2, c are three clauses. We say that c is a *resolvent* of c_1, c_2 if there is a u such that $u \in c_1$, $\bar{u} \in c_2$ and

$$c = (c_1 \setminus \{u\}) \cup (c_2 \setminus \{\bar{u}\}).$$

We denote this by the diagram

$$\begin{tikzcd}[column sep=small] & c \arrow[dl, dash] \arrow[dr, dash] & \\ c_1 & & c_2 \end{tikzcd}$$

We allow here the case $c = \square$, i.e. $c_1 = \{u\},\ c_2 = \{\bar{u}\}$.

Examples 1.10.3

(i)

$$\begin{tikzcd}[column sep=small] & \{p, r\} \arrow[dl, dash] \arrow[dr, dash] & \\ \{p, \neg q, r\} & & \{q, r\} \end{tikzcd}$$

(ii)

$$\begin{tikzcd}[column sep=small] & \{q, \neg q\} \arrow[dl, dash] \arrow[dr, dash] & & & \{p, \neg p\} \arrow[dl, dash] \arrow[dr, dash] & \\ \{p, \neg q\} & & \{\neg p, q\} & \{p, \neg q\} & & \{\neg p, q\} \end{tikzcd}$$

(iii)

$$\begin{tikzcd}[column sep=small] & \square \arrow[dl, dash] \arrow[dr, dash] & \\ \{p\} & & \{\neg p\} \end{tikzcd}$$

Proposition 1.10.4 *If c is a resolvent of c_1, c_2, then $\{c_1, c_2\} \models c$. (We view here c_1, c_2, c as formulas.)*

Proof. Suppose a valuation ν satisfies both c_1, c_2 and let u be the literal used in the resolution. If $\nu(u) = 1$, then since $\nu(c_2) = 1$ we clearly have $\nu(c_2 \setminus \{\bar{u}\}) = 1$ and so $\nu(c) = 1$. If $\nu(u) = 0$, then $\nu(c_1 \setminus \{u\}) = 1$, so $\nu(c) = 1$. $\dashv$

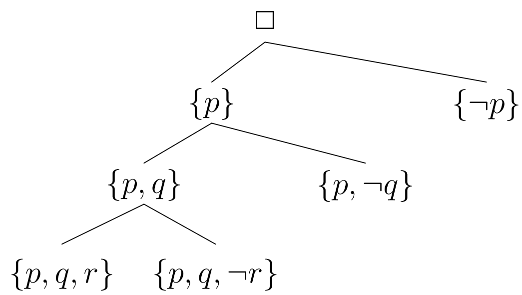
Definition 1.10.5 Let now C be a set of clauses (finite or infinite). A *proof by resolution* from C is a sequence $c_1, c_2, \ldots, c_n$ of clauses such that each c_i is either in C or else it is a resolvent of some c_j, c_k with $j, k < i$. We call c_n the *goal* or *conclusion* of the proof. If $c_n = \square$, we call this a *proof by resolution of a contradiction* from C or simply a *refutation* of C.

Example 1.10.6 Let $C = \{\{p, q, \neg r\}, \{\neg p\}, \{p, q, r\}, \{p, \neg q\}\}$. Then the following is a refutation of C:

$$\begin{aligned} c_1 &= \{p, q, \neg r\} && \text{(in } C\text{)} \\ c_2 &= \{p, q, r\} && \text{(in } C\text{)} \end{aligned}$$

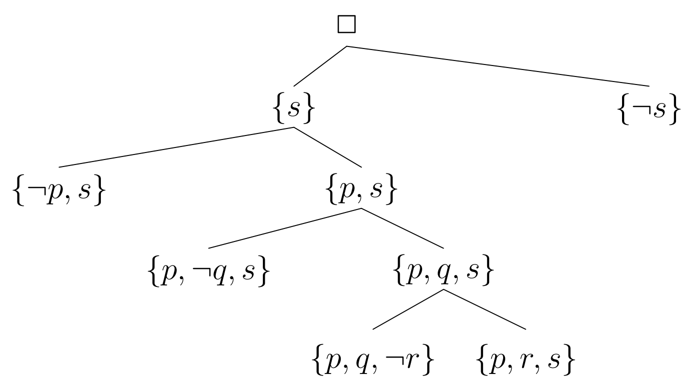
$$
\begin{array}{ll}
c_3 = \{p, q\} & \text{(resolvent of } c_1, c_2 \text{ (by } r\text{))} \\
c_4 = \{p, \neg q\} & \text{(in } C\text{)} \\
c_5 = \{p\} & \text{(resolvent of } c_3, c_4 \text{ (by } q\text{))} \\
c_6 = \{\neg p\} & \text{(in } C\text{)} \\
c_7 = \square & \text{(resolvent of } c_5, c_6 \text{ (by } p\text{)).}
\end{array}
$$

We can also represent this by a tree:



Terminal nodes correspond to clauses in C and each branching $\wedge$ corresponds to creating a resolvent. We call such a tree a *resolution tree*.

Example 1.10.7 Let $C = \{\{\neg p, s\}, \{p, \neg q, s\}, \{p, q, \neg r\}, \{p, r, s\}, \{\neg s\}\}$.



This can be also written as a proof as follows:

$$
\begin{array}{lcl}
c_1 & = & \{p, q, \neg r\} \\
c_2 & = & \{p, r, s\}
\end{array}
$$

$$
\begin{aligned}
c_3 &= \{p, q, s\} \\
c_4 &= \{p, \neg q, s\} \\
c_5 &= \{p, s\} \\
c_6 &= \{\neg p, s\} \\
c_7 &= \{s\} \\
c_8 &= \{\neg s\} \\
c_9 &= \Box
\end{aligned}
$$

(This proof is not unique. For example, we could move c_8 before c_3 and get another proof corresponding to the same resolution tree. The relationship between proofs by resolution and their corresponding trees is similar to that between parsing sequences and parse trees.)

The goal of proofs by resolution is to prove unsatisfiability of a set of clauses. The following theorem tells us that they achieve their goal.

Theorem 1.10.8 *Let $C = \{c_1, c_2, \dots\}$ be a set of clauses. Then C is unsatisfiable iff there is a refutation of C.*

Proof.

$\Leftarrow$: *Soundness of the proof system.*

Let $d_1, \dots, d_n$ be a proof of resolution from C. Then by Proposition 1.8.1, we can easily prove, by induction on $1 \le i \le n$, that

$$C \models d_i.$$

So if $d_n = \Box$, then $C \models \Box$, i.e., C is unsatisfiable.

$\Rightarrow$: *Completeness of the proof system.*

First we can assume that C has no clause c_i which contains, for some literal u, both u and $\bar{u}$ (since such a clause can be dropped from C without affecting its satisfiability).

Notation. If u is a literal, let $C(u)$ be the set of clauses resulting from C by canceling every occurrence of u within a clause of C and eliminating all clauses of C containing $\bar{u}$ (this effectively amounts to setting $u = 0$).

Example. Let $C = \{\{p, q, \neg r\}, \{p, \neg q\}, \{p, q, r\}, \{q, r\}\}$. Then

$$
\begin{aligned}
C(r) &= \{\{p, \neg q\}, \{p, q\}, \{q\}\} \\
C(\bar{r}) &= \{\{p, q\}, \{p, \neg q\}\}
\end{aligned}
$$

Note that $u, \bar{u}$ do not occur in $C(u),\ C(\bar{u})$. Note also that if C is unsatisfiable, so are $C(u),\ C(\bar{u})$. Because if ν is a valuation satisfying $C(u)$, then, since $C(u)$ does not contain $u, \bar{u}$, we can assume that ν does not assign a value to u. Then the valuation ν' which agrees on all other variables with ν and gives $\nu(u) = 0$ satisfies C. Similarly for $C(\bar{u})$.

So assume C is unsatisfiable, in order to construct a refutation of C. By the Compactness Theorem there is a finite subset $C_0 \subseteq C$ which is unsatisfiable, so we may as well assume from the beginning that C is finite. Say that all the propositional variables occurring in clauses in C are among $p_1, \ldots, p_n$. We prove then the result by induction on n. In other words, we show that for each n, if C is a finite set of clauses containing variables among $p_1, \ldots, p_n$ and C is unsatisfiable, there is a refutation of C.

$n = 1$. In this case, we must have $C = \{\{p_1\}, \{\neg p_1\}\}$, and hence we have the refutation $\{p_1\},\ \{\neg p_1\},\ \Box$.

$n \to n+1$. Assume this has been proved for sets of clauses with variables among $\{p_1, \ldots, p_n\}$ and consider a set of clauses C with variables among $\{p_1, \ldots, p_n, p_{n+1}\}$. Let $u = p_{n+1}$.

Then $C(u), C(\bar{u})$ are also unsatisfiable and do not contain p_{n+1}, so by induction hypothesis there is a refutation $d_1, \ldots d_m, d_{m+1} = \Box$ for $C(u)$ and a refutation $e_1, \ldots, e_k, e_{k+1} = \Box$ for $C(\bar{u})$.

Consider first $d_1, \ldots, d_{m+1}$. Each clause d_i is in $C(u)$ or comes as a resolvent of two previous clauses. Define then recursively $d'_1, \ldots, d'_m, d'_{m+1}$, so that either $d'_i = d_i$ or $d'_i = d_i \cup \{u\}$.

If $d_i \in C(u)$, then it is either in C and then we put $d'_i = d_i$ or else is obtained from some $d^*_i \in C$ by dropping u, i.e., $d_i = d^*_i \setminus \{u\}$. Then put $d'_i = d^*_i$.

The other case is where for some $j, k < i$, we have that d_i is a resolvent of d_j, d_k, and thus by induction d'_j, d'_k are already defined. The variable used in this resolution is in $\{p_1, \ldots, p_n\}$, so we can use this variable to resolve from d'_j, d'_k to get d'_i.

Thus $d'_{m+1} = \Box$ or $d'_{m+1} = \{p_{n+1}\}$, and $d'_1, \ldots d'_m, d'_{m+1}$ is a proof by resolution from C. If $d'_{m+1} = \Box$ we are done, so we can assume that $d'_{m+1} = \{p_{n+1}\}$, i.e., $d'_1, \ldots, d'_m, \{p_{n+1}\}$ is a proof by resolution from C. Similarly, working with $\bar{u}$, we can define $e'_1, \ldots e'_k, e'_{k+1}$, a proof by resolution from C with $e'_{k+1} = \Box$ or $e'_{k+1} = \{\neg p_{n+1}\}$. If $e'_{k+1} = \Box$ we are done, otherwise $e'_1, \ldots, e'_k, \{\neg p_{n+1}\}$ is a proof by resolution from C. Then

$$d'_1, \ldots, d'_m, \{p_{n+1}\}, e'_1, \ldots, e'_k, \{\neg p_{n+1}\}, \Box$$

is a refutation from C. $\dashv$

Example 1.10.9

$$\begin{aligned} C &= \{\{p,q,\neg r\},\{\neg p\},\{p,q,r\},\{p,\neg q\}\} \qquad (u=r) \\ C(r) &= \{\{\neg p\},\{p,q\},\{p,\neg q\}\} \\ C(\neg r) &= \{\{p,q\},\{\neg p\},\{p,\neg q\}\} \end{aligned}$$

Refutation from $C(r)$	Proof by resolution from C	Refutation from $C(\neg r)$	Proof by resolution from C
$\{p,q\} \rightarrow$	$\{p,q,r\}$	$\{p,q\} \rightarrow$	$\{p,q,\neg r\}$
$\{p,\neg q\} \rightarrow$	$\{p,\neg q\}$	$\{p,\neg q\} \rightarrow$	$\{p,\neg q\}$
$\{p\} \rightarrow$	$\{p,r\}$	$\{p\} \rightarrow$	$\{p,\neg r\}$
$\{\neg p\} \rightarrow$	$\{\neg p\}$	$\{\neg p\} \rightarrow$	$\{\neg p\}$
$\square \rightarrow$	$\{r\}$	$\square \rightarrow$	$\{\neg r\}$

$\square$

Remark. Notice that going from n to $n+1$ variables "doubles" the length of the proof, so this gives an exponential bound for the refutation.

Remark. The method of refutation by resolution is non-deterministic– there is no unique way to arrive at it. Various strategies have been devised for implementing it.

One is by following the recursive procedure used in the proof of theorem 1.10.8. Another is by brute force. Start with a finite set of clauses C. Let $C_0 = C$. Let $C_1 = C$ together with all clauses obtained by resolving all possible pairs in $C_0, C_2 = C_1$ together with all clauses obtained by resolving all possible pairs from C_1, etc. Since any set of clauses whose variables are among $p_1,\ldots,p_n$ cannot have more than 2^{2n} elements, this will stop in at most 2^{2n} many steps. Put $C_{2^{2n}} = C^*$. If $\square \in C^*$ then we can produce a refutation proof of about that size (i.e., 2^{2n}). Otherwise, $\square \notin C^*$ and C is satisfiable.

Other strategies are more efficient in special cases, e.g., for Horn formulas (see Assignment #4, for the definition of a Horn formula.)

1.11 A Hilbert-Type Proof System

We will now describe a different formal calculus of proofs, which corresponds more directly to our intuitive concept of a "proof." That is, it will start with axioms and use rules of inference to deduce consequences, rather than dealing with abstract clause sets.

This style of arguing in formal logic, using axioms and rules, was introduced by David Hilbert and his school. Hilbert was born in Königsberg, Prussia, in 1862, and died in Göttingen, Germany, in 1943. He is one of the most important mathematicians of the late 19-th and early 20-th century. A strong advocate of *nonconstructive* methods, this probably started with his work on "invariant theory," where he proved a Basis Theorem without exhibiting an explicit basis. The paper appeared in Mathematische Annalen in 1888. The result states that in any number of variables there is a finite set of generators for the invariants of *quantics*. A quantic is a symmetric tensor of a given degree constructed from tensor powers of a (finite dimensional) vector space. Invariance is measured under invertible operators of the space in itself.

Gordon, the world expert in invariant theory, strongly opposed in vain to the publication of Hilbert's result, due to its nonconstructive nature. On the other hand, Klein said of this result that "I do not doubt this is the most important work on general algebra that the Annalen has ever published."

Hilbert's work in number theory led to his book on the theory of algebraic number fields, recently translated into English. He started his work on logic with the publication in 1899 of the *Grundlagen der Geometrie*, where he advocates the axiomatic approach to mathematics.

In 1900 he delivered the invited address "Mathematical problems" to the Second International Congress of Mathematicians in Paris, where he listed 23 problems that in his opinion were essential for the progress of mathematics. From here comes his dictum "Here is the problem, seek the solution". The inscription on his tombstone, "We must know, we will know" comes from a speech in Königsberg in 1938. His address can be found in the Bulletin of the AMS, vol 37, number 4 (2000).

His work on integral equations and calculus of variations led to functional analysis and the concept of Hilbert space. He also solved Waring's problem, a well-known problem in number theory. His work in logic resulted in the development of *proof systems* and lead to the incompleteness theorems of K. Gödel.

1.11.A Formal Proofs

It will be convenient here to consider as the basic symbols in the language of propositional logic the following:

$$\neg, \Rightarrow,), (, p_1, p_2, p_3, \ldots$$

and view $(A \wedge B), (A \vee B), (A \Leftrightarrow B)$ as abbreviations:

$$\begin{aligned}(A \wedge B) &: \neg(A \Rightarrow \neg B)\\ (A \vee B) &: (\neg A \Rightarrow B)\\ (A \Leftrightarrow B) &: \neg((A \Rightarrow B) \Rightarrow \neg(B \Rightarrow A)).\end{aligned}$$

A *(logical) axiom* is any formula of one of the following forms:

(i) $A \Rightarrow (B \Rightarrow A)$

(ii) $(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$

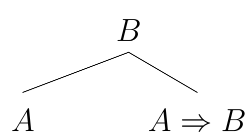
(iii) $((\neg B \Rightarrow \neg A) \Rightarrow ((\neg B \Rightarrow A) \Rightarrow B))$

(for arbitrary A, B, C). Notice that each one of these is a tautology.

We also have a *rule of inference* called *modus ponens* or MP: "From A and $A \Rightarrow B$, derive B." In formal calculi, a rule such as this for deriving one formula from others given, is sometimes written as follows:

$$\frac{A,\ A \Rightarrow B}{B}.$$

Modus ponens is short for latin "modus ponendo ponens" which means "proposing method". We can also draw this rule as a proof tree:



Notice that $A, A \Rightarrow B$ logically imply B.

Definition 1.11.1 Let S be any set (finite or infinite) of formulas. A *formal proof* from S is a finite sequence $A_1, A_2, \ldots, A_n$ of formulas such that each A_i is either a logical axiom, belongs to S or comes by applying modus ponens to some A_j, A_k with $j, k < i$ (i.e., A_k is the formula $A_j \Rightarrow A_n$). We call this sequence a *formal proof of* A_n *from* S.

Definition 1.11.2 If there is a formal proof of a formula A from S we say that A is a *formal theorem* of S and write

$$S \vdash A.$$

If $S = \emptyset$, we just write

$$\vdash A$$

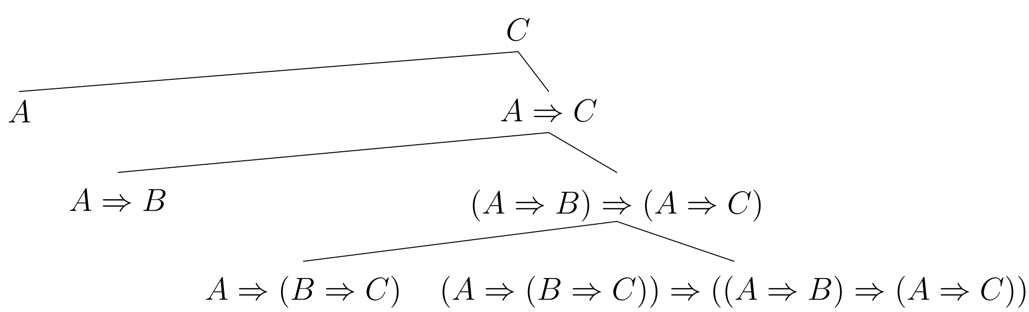
and call A a *formal theorem*.

Notice that $S \vdash A$ and $S \subseteq S'$ imply that $S' \vdash A$. Also notice that if $S' \vdash A$ and $S \vdash B$ for all $B \in S'$, then $S \vdash A$, since the formal proofs can simply be concatenated. Finally, $S \vdash A$ implies that there is *finite* $S_0 \subseteq S$ with $S_0 \vdash A$, since a formal proof can only be finitely long and hence can only use finitely many formulas of S.

Example 1.11.3 $S = \{(A \Rightarrow B),\ A \Rightarrow (B \Rightarrow C), A\}$. Here is a formal proof of C from S:

1.	$A \Rightarrow (B \Rightarrow C)$	(in S)
2.	$((A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C)))$	(axiom (ii))
3.	$(A \Rightarrow B) \Rightarrow (A \Rightarrow C)$	(MP from 1, 2)
4.	$A \Rightarrow B$	(in S)
5.	$A \Rightarrow C$	(MP from 3, 4)
6.	A	(in S)
7.	C	(MP from 5, 6)

We can also write this as a *proof tree*:



Example 1.11.4 The following formal proof shows that $\vdash A \Rightarrow A$:

1. $(A \Rightarrow (\underbrace{(A \Rightarrow A)}_{B} \Rightarrow \underbrace{A}_{C})) \Rightarrow ((A \Rightarrow \underbrace{(A \Rightarrow A)}_{B}) \Rightarrow (A \Rightarrow \underbrace{A}_{C}))$ (Ax. (ii))
2. $A \Rightarrow (\underbrace{(A \Rightarrow A)}_{B} \Rightarrow A)$ (Ax. (i))
3. $((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A))$ (MP 1, 2)
4. $A \Rightarrow (A \Rightarrow A)$ (Ax. (i))
5. $A \Rightarrow A$ (MP 3, 4).

The main result about this formal proof system is the following:

Theorem 1.11.5 *For any set of formulas S, and any formula A:*

$$S \models A \text{ iff } S \vdash A.$$

One direction of this theorem, i.e. *the soundedness of the proof system* ($S \vdash A$ implies that $S \models A$) is easy, and we prove it first.

1.11.B Soundness

Given a property $P(A)$ of formulas, suppose we want to prove that $P(A)$ holds for all formal theorems of S, i.e., we want to show that if $S \vdash A$, then $P(A)$ holds. We can do this by the following form of induction:

Basis.

(i) Show that $P(A)$ holds, when A is a logical axiom;

(ii) Show that $P(A)$ holds when A is in S.

Induction Step. Assuming that $P(A)$, $P(A \Rightarrow B)$ hold, and show that $P(B)$ holds.

We call this *induction on proofs of A from S*.

In our case $P(A)$ is the property

$$\text{“}S \models A\text{”}.$$

The basis of the induction is clear: $S \models A$ is certainly correct if A is in S or else A is a logical axiom (since it is then a tautology). The induction step is equally simple: If $S \models A$ and $S \models A \Rightarrow B$, clearly $S \models B$.

1.11.C Completeness

To prove the hard direction of the theorem, i.e. the *completeness of the proof system*, it will be convenient to develop first some basic properties of this system. These are useful independently of this and can help, for example, in constructing formal proofs. They are indeed formal analogs of some commonly used proof techniques. They are theorems about formal proofs and theorems, and so often called *metatheorems*.

Proposition 1.11.6 (The Deduction Theorem) *If $S \cup \{A\} \vdash B$, then $S \vdash A \Rightarrow B$.*

Remark. It is easy to see that if $S \vdash A \Rightarrow B$, then $S \cup \{A\} \vdash B$.

Proof. We will show this by induction on proofs of B from $S \cup \{A\}$.

Basis.

(i) B is a logical axiom. Then clearly $S \vdash B$. But $B \Rightarrow (A \Rightarrow B)$ is a logical axiom, thus $S \vdash B \Rightarrow (A \Rightarrow B)$, so, by modus ponens, $S \vdash A \Rightarrow B$.

(ii) B is in $S \cup \{A\}$. There are two subcases:

- (a) B is in S. Then again $S \vdash B$ and the proof is completed as in (i).
- (b) $B = A$. Then $\vdash A \Rightarrow A$ by example 1.11.4, so $S \vdash A \Rightarrow A$, which is the same as $S \vdash A \Rightarrow B$

Induction step. We assume that we have shown that $S \vdash A \Rightarrow (B \Rightarrow C)$ and $S \vdash A \Rightarrow B$, and want to show that $S \vdash A \Rightarrow C$. Now $((A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$ is a logical axiom, so

$$S \vdash ((A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C)),$$

and then, by MP applied twice, we have

$$S \vdash A \Rightarrow C.$$

$\dashv$

Definition 1.11.7 We call a set of formulas S *formally inconsistent* if for some formula A,

$$S \vdash A \text{ and } S \vdash \neg A.$$

Otherwise, S is *formally consistent*.

Proposition 1.11.8 (Proof by Contradiction) *If $S \cup \{A\}$ is formally inconsistent, then $S \vdash \neg A$.*

Proof. Left for Assignment #5. ⊣

Proposition 1.11.9 (Proof by Contrapositive) *If $S \cup \{A\} \vdash \neg B$, then $S \cup \{B\} \vdash \neg A$.*

Proof. Left for Assignment #5. ⊣

We embark now on the proof of the Completeness of our proof system. We claim that it is enough to show the following:

If a set of formulas S is formally consistent, then it is satisfiable. $(*)$

To see this, assume $(*)$ has been proved. If $S \models A$, then $S \cup \{\neg A\}$ is unsatisfiable, so by $(*)$, applied to $S \cup \{\neg A\}$, $S \cup \{\neg A\}$ is formally inconsistent. Thus, by proof by contradiction, $S \vdash \neg\neg A$, so by Assignment #5, Problem 1, $S \vdash A$, which is what we wanted. So it remains to prove $(*)$.

The idea of the proof is as follows: Let us call a set of formulas $\bar{S}$ *complete* if for any formula A,

$$A \in \bar{S} \text{ or } \neg A \in \bar{S}.$$

We will first prove that if S is a given formally consistent set of formulas, then we can find a formally consistent set of formulas $\bar{S} \supseteq S$ which is also complete. Then for any propositional variable p_i we must have that *exactly* one of the following holds

$$p_i \in \bar{S} \text{ or } \neg p_i \in \bar{S}.$$

Thus we can define a valuation ν by letting

$$\nu(p_i) = \begin{cases} 1 & \text{if } p_i \in \bar{S} \\ 0 & \text{if } \neg p_i \in \bar{S} \end{cases}.$$

It will then turn out that ν is a valuation that satisfies $\bar{S}$ and so S, i.e., S is satisfiable. Let's implement this plan.

Lemma 1.11.10 *Let S be a formally consistent set of formulas. Then we can find a formally consistent and complete set of formulas $\bar{S}$ such that $S \subseteq \bar{S}$.*

Proof. First we can enumerate in a sequence $A_1, A_2, \ldots$ all formulas. Then we can define recursively a sequence

$$S_0 = S,\ S_1, S_2, \ldots$$

of formulas such that

$$S_0 \subseteq S_1 \subseteq S_2 \subseteq \ldots$$

as follows:

$$S_0 = S,$$
$$S_{n+1} = \begin{cases} S_n \cup \{A_{n+1}\} & \text{if } S \cup \{A_{n+1}\} \text{ is formally consistent} \\ S_n \cup \{\neg A_{n+1}\} & \text{otherwise.} \end{cases}$$

We claim then that each S_n is formally consistent. We do this easily by induction using the following sublemma.

Sublemma 1.11.11 *If T is a formally consistent set of formulas and A is any formula, then at least one of $T \cup \{A\},\ T \cup \{\neg A\}$ is still formally consistent.*

Proof. Otherwise both $T \cup \{A\},\ T \cup \{\neg A\}$ are formally inconsistent. So $T \vdash A$ by proof of contradiction and Assignment #5, Problem 1, and since clearly $T \vdash B$ for any $B \in T$, it follows that $T \vdash B$ for any $B \in T \cup \{A\}$. Since $T \cup \{A\}$ is formally inconsistent, this shows that T is formally inconsistent, a contradiction. $\dashv$

Now let $\bar{S} = \bigcup_{n=1}^{\infty} S_n$. We claim this works. First $\bar{S}$ is complete, since if A is any given formula, then $A = A_n$ for some n and so by construction either $A \in S_n$ or $\neg A \in S_n$, i.e., $A \in \bar{S}$ or $\neg A \in \bar{S}$. Finally $\bar{S}$ is still formally consistent, since otherwise $\bar{S} \vdash A,\ \bar{S} \vdash \neg A$ for some formula A. Then by definition of a formal proof, there is a finite subset $S^* \subseteq \bar{S}$ such that $S^* \vdash A,\ S^* \vdash \neg A$. But then for some large enough n we have $S^* \subseteq S_n$, so $S_n \vdash A,\ S_n \vdash \neg A$, and so S_n is formally inconsistent, a contradiction. $\dashv$

Lemma 1.11.12 *Let $\bar{S}$ be formally consistent and complete. Define the valuation ν by $\nu(p_i) = 1$ if $p_i \in \bar{S}$, $\nu(p_i) = 0$ if $\neg p_i \in \bar{S}$. Then for any formula A,*

$$\nu(A) = 1 \textit{ iff } A \in \bar{S}.$$

Proof. Left for Assignment #5. $\dashv$

The combination of the two preceding lemmas finishes the proof of the completeness of the proof system. $\dashv$

We have already noted that if $S \vdash A$ then there is finite $S_0 \subseteq A$ with $S_0 \vdash A$. Therefore, as a corollary of the completeness of the proof system, we obtain a new proof of the Compactness Theorem for propositional logic.

1.11.D General Completeness

The Completeness Theorem for propositional logic is proved above only for formulas built using $\{\Rightarrow, \neg\}$. However, the result is valid for general formulas, provided a few logical axioms are added to the list. The proof given above can be easily adapted, and it is a useful exercise to see what additional steps are necessary. The axioms are designed to show that the formulas using the additional connectives are equivalent to formulas using only $\{\Rightarrow, \neg\}$. Here is a possible list.

(i) $(A \Rightarrow (B \Rightarrow A))$

(ii) $((A \Rightarrow (B \Rightarrow C)) \Longrightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C)))$

(iii) $((\neg A \Rightarrow \neg B) \Rightarrow (B \Rightarrow A))$

(iv) $((A \wedge B) \Rightarrow A)$

(v) $((A \wedge B) \Rightarrow B)$

(vi) $((A \Rightarrow B) \Longrightarrow ((A \Rightarrow C) \Rightarrow (A \Rightarrow (B \wedge C))))$

(vii) $(A \Rightarrow (A \vee B))$

(viii) $(B \Rightarrow (A \vee B))$

(ix) $((A \Rightarrow C) \Longrightarrow ((B \Rightarrow C) \Rightarrow ((A \vee B) \Rightarrow C)))$

(x) $((A \Leftrightarrow B) \Rightarrow (A \Rightarrow B))$

(xi) $((A \Leftrightarrow B) \Rightarrow (B \Rightarrow A))$

(xii) $((A \Rightarrow (B \Rightarrow C)) \Longrightarrow ((A \Rightarrow (C \Rightarrow B)) \Rightarrow (A \Rightarrow (B \Leftrightarrow C))))$