1 Introduction

1. ‘Some’ is being used both as an existential quantifier and as an instantiation. Symbolically, the first statement is \( \exists x (\text{car}(x) \land \text{rattles}(x)) \), and the second is \( \text{car}(\text{mycar}) \). Clearly, \( \text{rattles}(\text{mycar}) \) does not follow.

2 Propositional Calculus: Formulas, Models, Tableaux

1. The truth tables can be obtained from the Prolog program. Here are the formation trees.
2. The proof is by an inductive construction that creates a formation tree from a derivation tree. Let \( \text{fml} \) be a nonterminal node with no occurrences of \( \text{fml} \) below it. If the node is

\[
\text{fml} \quad p
\]

for some atom \( p \), then the only formation tree is \( p \) itself. Otherwise, suppose that the non-terminal is

\[
\text{fml} \quad T_1 \quad op \quad T_2
\]

where \( T_1 \) and \( T_2 \) are formation trees and \( op \) is a binary operator. The only formation tree is

\[
op \quad T_1 \quad T_2
\]

3. The proof is by induction on the structure of an arbitrary formula \( A \). If \( A \) is an atom, there is no difference between an assignment and an interpretation. If \( A = A_1 \quad op \quad A_2 \) is a formula with a binary operator, then by the inductive hypothesis \( v(A_1) \) and \( v(A_2) \) are uniquely defined, so there is a single value that can be assigned to \( v(A) \) according to the table. The case for negation is similar.

Induction is also used to prove that assignments that agree on the atoms of a formula \( A \) agree on the formula. For an atom of \( A \), the claim is trivial, and the inductive step is straightforward.

4. Construct the truth tables for the formulas and compare that they are the same. For example, the table for the formulas in the fourth equivalence is:
where we have added an extra column for the subformula $A \land \lnot B$.

5. By associativity and idempotence, 

\[
((p \oplus q) \oplus q) \equiv (p \oplus (q \oplus q)) \equiv p \oplus \text{false}.
\]

Using the definition of $\oplus$, we find that $p \oplus \text{false} \equiv p$. Similarly, 

\[
((p \leftrightarrow q) \leftrightarrow q) \equiv (p \iff (q \iff q)) \equiv p \iff \text{true} \equiv p.
\]

6. We prove 

\[
A_1 \oplus A_2 \equiv B_1 \circ \cdots \circ B_n \equiv \lnot \cdots \lnot B_i
\]

by induction on $n$. If $n = 1$, clearly $B_i$ is either $A_1$ or $A_2$. If $n = 2$ and the definition of $\circ$ is

<table>
<thead>
<tr>
<th>$A_1$</th>
<th>$A_2$</th>
<th>$A_1 \circ A_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>$T$</td>
<td>$F$</td>
</tr>
<tr>
<td>$T$</td>
<td>$F$</td>
<td>$T$</td>
</tr>
<tr>
<td>$F$</td>
<td>$T$</td>
<td>$F$</td>
</tr>
<tr>
<td>$F$</td>
<td>$F$</td>
<td>$T$</td>
</tr>
</tbody>
</table>

then $A_1 \circ A_2 \equiv \lnot A_2$, $A_2 \circ A_1 \equiv \lnot A_1$, $A_1 \circ A_1 \equiv \lnot A_1$, $A_2 \circ A_2 \equiv \lnot A_2$, and symmetrically for

<table>
<thead>
<tr>
<th>$A_1$</th>
<th>$A_2$</th>
<th>$A_1 \circ A_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>$T$</td>
<td>$F$</td>
</tr>
<tr>
<td>$T$</td>
<td>$F$</td>
<td>$F$</td>
</tr>
<tr>
<td>$F$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
<tr>
<td>$F$</td>
<td>$F$</td>
<td>$T$</td>
</tr>
</tbody>
</table>

Suppose now that

\[
A_1 \oplus A_2 \equiv (B_1 \circ \cdots \circ B_k) \circ (B_{k+1} \circ \cdots \circ B_n).
\]

By the inductive hypothesis, $B_1 \circ \cdots \circ B_k \equiv \lnot \cdots \lnot B_i$, and $B_{k+1} \circ \cdots \circ B_n \equiv \lnot \cdots \lnot B_i$, where $\lnot \cdots \lnot B_i$ and $\lnot \cdots \lnot B_i$ are each logically equivalent to $A_1$, $\lnot A_1$, $A_2$, or $\lnot A_2$. By an argument similar to that used for $n = 2$, the claim follows.

7. Let $A$ be a formula constructed only from only $p$ and $\land$ or $\lor$. We prove by induction that $A \equiv p$. Clearly, if $A$ is an atom, $A$ is $p$. Suppose that $A$ is $A_1 \land A_2$ and that $A_1 \equiv A_2 \equiv p$. Then $A \equiv p \land p \equiv p$. Similarly, for $\lor$.

8. If $U = \{p\}$ and $B$ is $\lnot p$, then $U$ is satisfiable, but $U \cup \{B\} = \{p, \lnot p\}$ is not.

3
9. We prove Theorem 2.35; the others are similar. Suppose that $U - \{A_i\}$ is satisfiable and let $I$ be a model. But a valid formula is true in all interpretations, so $A$ is true in $I$. Thus, $I$ is a model for $U$, contradicting the assumption.

10. Theorem 2.38: Any interpretation which falsifies $U = \{A_1, \ldots, A_n\}$ assigns true to $A_1 \land \cdots \land A_n \rightarrow A$ by the definition of $\land$ and $\rightarrow$. Any model for $U = \{A_1, \ldots, A_n\}$, assigns true to $A$ by assumption.

Theorem 2.39: Adding an additional assumption can only reduce the number of interpretations that have to satisfy $A$.

Theorem 2.40: Since any interpretation satisfies a valid formulas, the set of models for $U$ is exactly the set of models for $U - \{B\}$.

11. For $\mathcal{T}(U)$ to be closed under logical consequence means that if $\{A_1, \ldots, A_n\} \models A$ where $A_i \in \mathcal{T}(U)$ then $A \in \mathcal{T}(U)$. Let $I$ be an arbitrary model for $U$. If $A_i \in \mathcal{T}(U)$, then $U \models A_i$, so $I$ is a model for $A_i$ for all $i$ and $I$ is a model for $A$. Thus $U \models A$ so $A \in \mathcal{T}(U)$.

12. A logical equivalence can be proven by replacing $\equiv$ by $\leftrightarrow$ and constructing a truth table or semantic tableau for the negation. Alternatively, truth tables can be constructed for both sides and checked for equality. The use of Venn diagrams is interesting in that it shows an equivalence between propositional logical and set theory or Boolean algebra. The Venn diagram for a proposition represents the set of interpretations for which it is true as demonstrated in the following diagram:

\[
\begin{array}{c}
A \lor B \\
\hline
A \\
A \land B \\
\hline
B
\end{array}
\]

How is $A \rightarrow B$ represented in a Venn diagram? $A \rightarrow B$ is logically equivalent to $\neg A \lor B$, so the diagram for $A \rightarrow B$ consists of the area outside $A$ together with the area for $B$. $A \leftrightarrow B$ is represented by the intersection of the areas for $A$ and $B$ (where both are true), together with the area outside both (where both are false).

Here are the proofs of the equivalences in terms of Venn diagrams:

$A \rightarrow B \equiv A \leftrightarrow (A \land B)$: If a point is in the area for $A \rightarrow B$, it is either in the area for $A \land B$ or outside both. So if it is in $A$, it must be within $A \land B$. 


A → B ≡ B ↔ (A ∨ B): Similarly, if it is in the union of the areas, it must be within B.

A ∧ B ≡ (A ↔ B) ↔ (A ∨ B): Points are in both of the areas for A ↔ B and A ∨ B if they are within the area for A ∧ B.

A ↔ B ≡ (A ∨ B) → (A ∧ B): If a point is in the union of the areas for A and B, it must be within the area for A ∧ B if it is to be within the area for A ↔ B.

A ↔ B ≡ (A ∨ B) → (A ∧ B): If a point is in the union of the areas for A and B, it must be within the area for A ∧ B if it is to be within the area for A ↔ B.

13. Let W(l) = 4e(l) + 1 · (3b(l) + n(l) + 3), where e(l) is the number of equivalence and non-equivalence operators. If e(l) is decreased by one, the new b(l) will be 2b(l) + 2 and the new n(l) will be at most 2n(l) + 2. A computation will show that:

\[ 4^{e(l)+1} \cdot (3b(l) + n(l) + 3) > 4^{e(l)} \cdot (6b(l) + 2n(l) + 8). \]

14. We have to show that if the label of a node contains a complementary pair of formulas, then any tableau starting from that node will close (atomically). The proof is by induction. The base case is trivial. Suppose that \{α, ¬α\} ⊆ U(n), and that we use the α-rule on α, resulting in \{α1, α2, ¬α\} ⊆ U(n'), and then the β-rule on ¬α, resulting in \{α1, α2, ¬α1\} ⊆ U(n'1) and \{α1, α2, ¬α2\} ⊆ U(n'2). The result follows by the inductive hypothesis. The case for \{β, ¬β\} ⊆ U(n) is similar.

15. Add facts to the alpha and beta databases for the decompositions on page 32.

16. A node can become closed only if the addition of a new subformula to the label contradicts an existing one. Rather than check all elements of the label against all others, include the check for contradiction in the predicates alpha_rule and beta_rule.

### 3 Propositional Calculus: Deductive Systems

1. A, B, ¬A
2. ¬B, B, ¬A
3. ¬(A → B), B, ¬A
4. ¬(A → B), ¬¬B, ¬A
5. ¬(A → B), (¬B → ¬A)
6. (A → B) → (¬B → ¬A)
1. \( A, \neg A, B \) \hspace{1cm} \text{Axiom}
2. \( A, \neg B, B \) \hspace{1cm} \text{Axiom}
3. \( A, \neg (\neg A \rightarrow B), B \) \hspace{1cm} \beta \rightarrow 1, 2
4. \( \neg B, \neg A, B \) \hspace{1cm} \text{Axiom}
5. \( \neg B, \neg B, B \) \hspace{1cm} \text{Axiom}
6. \( \neg B, \neg (\neg A \rightarrow B), B \) \hspace{1cm} \beta \rightarrow 4, 5
7. \( \neg (A \rightarrow B), \neg (\neg A \rightarrow B), B \) \hspace{1cm} \beta \rightarrow 3, 6
8. \( \neg (A \rightarrow B), (\neg A \rightarrow B) \rightarrow B \) \hspace{1cm} \alpha \rightarrow 7
9. \( (A \rightarrow B) \rightarrow ((\neg A \rightarrow B) \rightarrow B) \) \hspace{1cm} \alpha \rightarrow 8

1. \( \neg A, B, A \) \hspace{1cm} \text{Axiom}
2. \( A \rightarrow B, A \) \hspace{1cm} \alpha \rightarrow 1
3. \( \neg A, A \) \hspace{1cm} \text{Axiom}
4. \( \neg ((A \rightarrow B) \rightarrow A), A \) \hspace{1cm} \beta \rightarrow 2, 3
5. \( ((A \rightarrow B) \rightarrow A) \rightarrow A \) \hspace{1cm} \alpha \rightarrow 4

2. The proof is by induction on the structure of the proof. If \( \vdash U \) where \( U \) is an axiom, then \( U \) is a set of literals containing a complementary pair \( \{p, \neg p\} \), that is, \( U = U_0 \cup \{p, \neg p\} \). Obviously, there is a closed tableau for \( \bar{U} = U_0 \cup \{\neg p, p\} \).

Let the last step of the proof of \( U \) be an application an \( \alpha \)- or \( \beta \)-rule to obtain a formula \( A \in U \); we can write \( U = U_0 \cup \{A\} \). In the following, we use \( \lor \) and \( \land \) as examples for \( \alpha \)- and \( \beta \)-formulas.

Case 1: An \( \alpha \)-rule was used on \( U' = U_0 \cup \{A_1, A_2\} \) to prove \( U = U_0 \cup \{A_1 \lor A_2\} \). Using the tableau \( \alpha \)-rule, there is a closed tableau for \( \bar{U} = \bar{U}_0 \cup \{\neg A_1, \neg A_2\} \).

Case 2: An \( \beta \)-rule was used on \( U' = U_0 \cup \{A_1\} \) and \( U'' = U_0 \cup \{A_2\} \) to prove \( U = U_0 \cup \{A_1 \land A_2\} \). By the inductive hypothesis, there are closed tableaux for \( \bar{U}' = \bar{U}_0 \cup \{\neg A_1\} \) and \( \bar{U}'' = \bar{U}_0 \cup \{\neg A_2\} \). Using the tableau \( \beta \)-rule, there is a closed tableau for \( \bar{U} = \bar{U}_0 \cup \{\neg (A_1 \land A_2)\} \).

3.
1. \( \vdash (A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A) \) \hspace{1cm} \text{Theorem 3.24}
2. \( \vdash (A \rightarrow B) \) \hspace{1cm} \text{Assumption}
3. \( \vdash \neg B \rightarrow \neg A \) \hspace{1cm} \text{MP 1, 2}
4. \( \vdash \neg B \) \hspace{1cm} \text{Assumption}
5. \( \vdash A \) \hspace{1cm} \text{MP 3, 4}

4.
1. \( \neg A, \neg B, A \) \hspace{1cm} \text{Axiom}
2. \( \neg A, B \rightarrow A \) \hspace{1cm} \alpha \rightarrow 1
3. \( A \rightarrow (B \rightarrow A) \) \hspace{1cm} \alpha \rightarrow 2

For Axiom 2 we will use a shortcut by taking as an axiom any set of formulas containing a complementary pair of literals.
5. Of course, \( \leftrightarrow \) should be \( \rightarrow \).

1. \{\( A \lor B \lor C \)\} \( \vdash (A \lor B) \lor C \) \\
2. \{\( A \lor B \lor C \)\} \( \vdash \neg(A \rightarrow B) \rightarrow C \) \\
3. \{\( A \lor B \lor C \)\} \( \vdash C \rightarrow (\neg A \rightarrow B) \) \\
4. \{\( A \lor B \lor C \)\} \( \vdash (\neg C \rightarrow (A \rightarrow B)) \rightarrow (A \rightarrow B) \lor (B \rightarrow C) \) \\
5. \{\( A \lor B \lor C \)\} \( \vdash (A \lor B) \rightarrow (\neg C \rightarrow (A \rightarrow B)) \lor (A \rightarrow B) \lor (B \rightarrow C) \)

8. The proofs are trivial.
9. The second node below is obtained by applying the $\alpha$-rule for $\rightarrow$ three times.

\[
\neg((A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C)))
\]
\[
\downarrow
\]
\[
A \rightarrow (B \rightarrow C), A \rightarrow B, A, \neg C
\]
\[
A \rightarrow (B \rightarrow C), A \rightarrow B, A, \neg C
\]
\[
\times
\]
\[
\neg A, B, A, \neg C
\]
\[
B \rightarrow C, B, A, \neg C
\]
\[
\neg A, B, A, \neg C
\]
\[
\times
\]
\[
\neg B, B, A, \neg C
\]
\[
C, B, A, \neg C
\]
\[
\times
\]

10. Let $(A_1, \ldots, A_n)$ be the elements of $U - U'$ in some order.

1. $\vdash \bigvee U'$ Assumption
2. $\vdash \bigvee U' \lor A_1$ Theorem 3.31
   
\[
\vdots
\]
\[
n+1. \vdash \bigvee U' \lor A_1 \lor \cdots \lor A_n \quad \text{Theorem 3.31}
\]

So we have to prove that if $U'$ is a permutation of $U$ and $\vdash \bigvee U$ then $\vdash \bigvee U'$.

The proof is by induction on $n$ the number of elements in $U$. If $n = 1$, there is nothing to prove, and if $n = 2$, the result follows immediately from Theorem 3.32. Let $\bigvee U = \bigvee U_1 \lor \bigvee U_2$ and $\bigvee U' = \bigvee U_1' \lor \bigvee U_2'$ have $n$ elements. If $U_1'$ and $U_2'$ are permutations of $U_1$ and $U_2$, respectively, then the result follows by the inductive hypothesis and Theorem 3.31. Otherwise, without loss of generality, suppose that there is an element $A$ of $U_2'$ which is in $U_1$. Suppose that $\bigvee U_2' = A \lor \bigvee U_2''$, so that $\bigvee U' = \bigvee U_1' \lor (A \lor \bigvee U_2'')$. Then by Theorem 3.33, $\bigvee U' = (\bigvee U_1' \lor A) \lor \bigvee U_2''$. Thus all we have to prove is that $A_1 \lor \cdots \lor A_i \lor \cdots \lor A_n$ can be written $A_i \lor A_1 \lor \cdots \lor A_{i-1} \lor A_{i+1} \lor \cdots \lor A_k$ for arbitrary $i$. This is proved by a simply induction using Theorem 3.33.

11. The first formula was proved in Theorem 3.24.

1. $\{A \rightarrow B, \neg A \rightarrow B\} \vdash \neg A \rightarrow B$ Assumption
2. $\{A \rightarrow B, \neg A \rightarrow B\} \vdash \neg B \rightarrow A$ Contrapositive
3. $\{A \rightarrow B, \neg A \rightarrow B\} \vdash A \rightarrow B$ Assumption
4. $\{A \rightarrow B, \neg A \rightarrow B\} \vdash \neg B \rightarrow B$ Transitivity
5. $\{A \rightarrow B, \neg A \rightarrow B\} \vdash (\neg B \rightarrow B) \rightarrow B$ Theorem 3.29
6. $\{A \rightarrow B, \neg A \rightarrow B\} \vdash B$ MP 4, 5
7. $\{A \rightarrow B\} \vdash (\neg A \rightarrow B) \rightarrow B$ Deduction
8. $\vdash (A \rightarrow B) \rightarrow ((\neg A \rightarrow B) \rightarrow B)$ Deduction

8
14. If $\vdash \neg A$, then clearly $U \vdash \neg A$ and $U \vdash \neg A$, since we do not need to use the assumptions. But that is the same as $U \vdash A_1 \rightarrow (A_2 \rightarrow \cdots \rightarrow \neg A_n)$. Now $U \vdash A_1$ is trivial, so by $n - 1$ applications of modus ponens, $U \vdash \neg A_n$, which together with $U \vdash A_n$, prove that $U$ is inconsistent. Conversely, if $U$ is inconsistent, then $U \vdash A$ and $U \vdash \neg A$ for some $A$. But only a finite number of formulas $\{A_1, \ldots, A_n\} \subseteq U$ are used in either one of the proofs, so $\{A_1, \ldots, A_n\} \vdash A$ and $\{A_1, \ldots, A_n\} \vdash \neg A$. By $n$ applications of the
15. This theorem can be proved both syntactically and semantically. The syntactic
proof uses the same construction as the one for CNF except that the distributive
laws used are: \( A \land (B \lor C) \equiv (A \land B) \lor (A \land C) \) and \( (A \lor B) \land C \equiv (A \land C) \lor (B \land C) \).
For the semantic proof, start by constructing a truth table for the formula \( A \). For
each line in the truth table that evaluates to \( T \), construct a conjunction with the
literal \( p \) if \( p \) is assigned \( T \) in that row and \( \bar{p} \) if \( p \) is assigned \( F \). Let \( A' \) be the
disjunction of all these conjunctions. Then \( A \equiv A' \). Let \( v \) be an arbitrary model
for \( A \), that is, \( v(A) = T \). Then the assignments in \( v \) are such that the row of the
truth table contains \( T \), and by construction \( v(C) = T \) for the conjunction \( C \) built
from that row. Since \( A' \) is a disjunction of such conjunctions, it is sufficient for
one of them to be true for \( A' \) to be true. Hence \( v(A') = T \). For the converse,
if \( v \) is an arbitrary interpretation so that \( v(A') = T \), then by the structure of \( A' \),
v(\( C \)) = \( T \) for at least one conjunction \( C \) in \( A' \) (in fact, for exactly one such
conjunction). But for this assignment, \( v(A) = T \) by construction.

16. I would be pleased if someone would contribute a program!

4 Propositional Calculus: Resolution and BDDs

1. This theorem can be proved both syntactically and semantically. The syntactic
proof uses the same construction as the one for CNF except that the distributive
laws used are: \( A \land (B \lor C) \equiv (A \land B) \lor (A \land C) \) and \( (A \lor B) \land C \equiv (A \land C) \lor (B \land C) \).
For the semantic proof, start by constructing a truth table for the formula \( A \). For
each line in the truth table that evaluates to \( T \), construct a conjunction with the
literal \( p \) if \( p \) is assigned \( T \) in that row and \( \bar{p} \) if \( p \) is assigned \( F \). Let \( A' \) be the
disjunction of all these conjunctions. Then \( A \equiv A' \). Let \( v \) be an arbitrary model
for \( A \), that is, \( v(A) = T \). Then the assignments in \( v \) are such that the row of the
truth table contains \( T \), and by construction \( v(C) = T \) for the conjunction \( C \) built
from that row. Since \( A' \) is a disjunction of such conjunctions, it is sufficient for
one of them to be true for \( A' \) to be true. Hence \( v(A') = T \). For the converse,
if \( v \) is an arbitrary interpretation so that \( v(A') = T \), then by the structure of \( A' \),
v(\( C \)) = \( T \) for at least one conjunction \( C \) in \( A' \) (in fact, for exactly one such
conjunction). But for this assignment, \( v(A) = T \) by construction.

2. The formula constructed in the previous exercise is in complete DNF.
3. I would be pleased if someone would contribute a program!

4. This exercise is rather trivial because the sets of clauses are satisfiable and for $S$ a set of satisfiable clauses, $S \approx \{\}$ the valid empty set of clauses. For each of the sets, we give a sequence of sets obtained by using the various lemmas.

\[
\{p\bar{q}, q\bar{r}, rs, p\bar{s}\} \approx \{q\bar{r}, rs\} \approx \{q\bar{r}\} \approx \{\}
\]
\[
\{pqr, q, p\bar{r}s, qs, p\bar{s}\} \approx \{pr, p\bar{r}s, s, p\bar{s}\} \approx \{pr, p\} \approx \{\} \approx \{\}
\]
\[
\{p\bar{q}, q\bar{r}s, \bar{p}rs, qs, \bar{p}s\} \approx \{\bar{q}rs, \bar{p}rs, qs, \bar{p}s\} \approx \{\}
\]
\[
\{p\bar{q}, q\bar{r}s, \bar{p}rs, \bar{q}, \bar{r}\} \approx \{\bar{p}rs, \bar{r}\} \approx \{\} \approx \{\}
\]

5.

Refutation 1:
5. $\bar{q}r$
6. $r$
7. $\square$

Refutation 2:
5. $\bar{p}q$
6. $p$
7. $\bar{q}$
8. $q$
9. $\square$

6. The clausal form of the set is:

\[
(1) p, (2) \bar{p}qr, (3) \bar{p}\bar{q}r, (4) \bar{p}st, (5) \bar{p}\bar{s}t, (6) \bar{s}q, (7) rt, (8) t\bar{s}.
\]

A refutation is:
9. $\bar{p}\bar{sr}$
10. $\bar{p}s$
11. $\bar{p}\bar{q}s$
12. $\bar{p}s$
13. $\bar{p}$
14. $\square$

7. The clausal form of the formulas is:

\[
\{(1) \bar{s}b\bar{1}b\bar{2}, (2) \bar{s}b1b2, (3) sb\bar{1}b2, (4) sb1\bar{b}2, (5) \bar{c}b1, (6) \bar{c}b2, (7) c\bar{b}1b\bar{2}\}.
\]

The addition of the set of clauses: $\{(8) b1, (9) b2, (10) \bar{s}, (11) \bar{c}\}$, enables a refutation to be done by resolving clauses 11, 7, 8, 9. The addition of the clauses $\{(8) b1, (9) b2, (10) \bar{s}, (11) \bar{c}\}$ gives a satisfiable set by assigning $F$ to $s$ and $T$ to all other atoms (check!). The meaning of the satisfiable set is that $1 \oplus 1 = 0$ carry 1, by identifying 1 with $T$ and 0 with $F$. The unsatisfiable set shows that it is not true that $1 \oplus 1 = 0$ carry 0.
8. The statement of the claim should say: Prove adding a unit clause to a set of clauses such that the atom of the unit clause does not already appear in the set and . . . .

Let $S$ be the original set of clauses and $S'$ the new set of clauses obtained by adding $\{l\}$ to $S$ and $\lnot l$ to every other clause in $S$, and let $v$ be a model for $S$. Extend $v$ to $v'$ by defining $v'(p) = T$ if $l = p$, $v'(p) = F$ if $\lnot l = p$, and $v'$ is the same as $v$ on all other atoms. (Here is where we need the proviso on the new clause.) By construction, $v(l) = T$ so the additional clause in $S'$ is satisfied. For every other clause $C$, $v(C) = v'(C) = T$ since the addition of a literal to a clause (which is a disjunction) cannot falsify it.

9. By induction on the depth of the resolution tree. If the depth of the tree is 1, the result is immediate from Theorem 4.24. If the depth of the tree is $n$, then the children of the root are satisfiable by the inductive hypothesis, so the root is satisfiable by Theorem 4.24.

10. First prove a lemma: for any $v$, $v(A_1|_{p=T} \text{ op } A_2|_{p=T}) = v(A_1 \text{ op } A_2)$ if $v(p) = T$ and $v(A_1|_{p=F} \text{ op } A_2|_{p=F}) = v(A_1 \text{ op } A_2)$ if $v(p) = F$. The proof is by structural induction. Clearly, $v(p|_{p=T}) = T = v(p)$ and $q|_{p=T}$ is the same formula as $q$ for $q \neq p$. Suppose now that $v(p) = T$. By the inductive hypothesis, $v(A_1|_{p=T}) = v(A_1)$ and $v(A_2|_{p=T}) = v(A_2)$ so by the semantic definitions of the operators, $v(A_1|_{p=T} \text{ op } A_2|_{p=T}) = v(A_1 \text{ op } A_2)$. A similar argument holds for $F$.

We can now prove the Shannon expansion. Let $v$ be an arbitrary interpretation. If $v(p) = T$,

$$v((p \land (A_1|_{p=T} \text{ op } A_2|_{p=T})) \lor (\lnot p \land (A_1|_{p=F} \text{ op } A_2|_{p=F}))) = v(A_1|_{p=T} \text{ op } A_2|_{p=T})$$

which equals $v(A_1 \text{ op } A_2)$ by the lemma, and similarly if $v(p) = F$. Since $v$ was arbitrary, the formulas are logically equivalent.

11. From Example 4.62, the BDDs for $p \lor (q \land r)$ and for $A_1|_{r=T}$ and $A_1|_{r=F}$ are:

```

Using the algorithm apply with $v$ gives the middle BDD above for $p \lor q$: recursing on the left subBDD gives $q \lor F$ which is $q$ and recursing on the right

12```
subBDD is clearly $T$. Using the algorithm \texttt{apply} with $\land$ gives the right BDD above for $p$: recursing on the left subBDD gives the controlling operand $F$ for $\land$ and recursing on the right subBDD is clearly $T$.

12. The programs in the software archive implement the optimizations.

13. Let us number the clauses as follows:

$$
(1) \overline{p}q, (2) pq, (3) prs, (4) \overline{p}\overline{r}s, (5) \overline{p}r\overline{s}, (6) p\overline{r}\overline{s},
(7) \overline{q}r, (8) \overline{q}r, (9) \overline{q}r, (10) \overline{q}r, (11) \overline{q}r, (12) \overline{q}r.
$$

The refutation is not for the faint-hearted....

13. $qrs$ 1, 3
14. $qst$ 13, 10
15. $\overline{q}rs$ 2, 4
16. $\overline{q}st$ 15, 9
17. $st$ 14, 16
18. $s$ 17, 8
19. $\overline{q}r\overline{s}$ 1, 6
20. $q\overline{s}t$ 19, 11
21. $\overline{q}r\overline{s}$ 2, 5
22. $\overline{q}s\overline{t}$ 21, 12
23. $s\overline{t}$ 20, 22
24. $\overline{s}$ 23, 7
25. $\Box$ 18, 24

14. This is trivial as all the leaves are labeled false.

15. Here is the complete graph with the edges and vertices labeled.

![Graph Diagram]

The clauses are the eight even-parity clauses on the atoms $abhi$ and the eight odd-parity clauses on each of $aeji$, $degh$, $cij$ and $bcfg$, forty clauses in all. We leave the construction of a resolution refutation to the reader.
16. Suppose $\Pi(v_n) = b_n$ and let $C = l_1 \cdots l_k$ be an arbitrary clause associated with $n$. Then $C$ can be only falsified by the assignment

$$v(p_i) = F \text{ if } l_i = p_i \text{ and } v(p_i) = T \text{ if } l_i = \bar{p}_i.$$ 

Then

$$\Pi(C) = b_n \quad \text{(by definition)}$$
$$\text{parity of negated atoms of } C = b_n \quad \text{(by construction)}$$
$$\text{parity of literals assigned } T = b_n \quad \text{(by definition)}$$
$$\Pi(v_n) = b_n \quad \text{(by assumption)}$$

which contradicts the assumption that $C \in C(n)$. Thus if $\Pi(v_n) = b_n$, $v$ must satisfy all clauses in $C(n)$.

17. Decision procedure for satisfiability of sets of formulas whose only operators are $\neg$, $\leftrightarrow$ and $\oplus$:

- Use $p \oplus q \equiv \neg(p \leftrightarrow q) \equiv \neg p \leftrightarrow q$ and $\neg \neg p \equiv p$ to reduce the formulas to the form $q_1 \leftrightarrow \cdots \leftrightarrow q_n$.
- Use commutativity, associativity and:
  $$p \leftrightarrow p \equiv true,$$  $$p \leftrightarrow \neg p \equiv false,$$  $$p \leftrightarrow true \equiv p,$$  $$p \leftrightarrow false \equiv \neg p,$$
  to reduce the formulas to equivalences on distinct atoms.
- If all formulas reduce to $true$, the set is valid. If some formula reduces to $false$, the set is unsatisfiable.
- Otherwise, delete all formulas which reduce to $true$. Transform the formulas as in the lemma. Let $\{p_1, \ldots, p_m\}$ be the new atoms. Assign $true$ to $p_1$ and each $q^j_i$ to which it is equivalent. By induction, assign $true$ to $p_i$, unless some $q^j_i$ has already been assigned to; if so, assign its value to $p_i$. If $p_i$ has already been assigned a clashing value, the set of formulas is unsatisfiable.

Each of the three steps increases the size of the formula by a small polynomial, so the procedure is efficient.

5 Predicate Calculus: Formulas, Models, Tableaux

1. A falsifying interpretation is $({1, 2}, \{2\}, \{1\})$. Then $v(p(2)) = T$ so $v(\exists x p(x)) = T$, but $v(p(a)) = v(p(1)) = F$.

2. We will prove the validity by constructing a closed semantic tableau for the negation of each formula. To simply formatting, a linear representation will be