Knowledge Representation and Reasoning

Exercise 1

Albert-Ludwigs-Universität Freiburg

UNI FREIBURG

Bernhard Nebel, Gregor Behnke, Thorsten Engesser, Rolf-David Bergdoll, Leonardo Mieschendahl, Johannes Herrmann

October 29th 2021

Exercise 1.1 (a)

Question: Is the infinite set of clauses

$$S = \{\neg A_1 \vee \neg A_2, A_2 \vee \neg A_3, A_3 \vee \neg A_4, A_4 \vee \neg A_5, \dots\}$$

satisfiable?

Exercise 1.1 (a)

Question: Is the infinite set of clauses

$$S = \{\neg A_1 \vee \neg A_2, A_2 \vee \neg A_3, A_3 \vee \neg A_4, A_4 \vee \neg A_5, \dots\}$$

satisfiable?

Consider the interpretation $\mathcal{I}$ which assigns false to every variable:

$$\mathcal{I}(A_i) = \mathbf{F} \text{ for all } i \in \mathbb{N}^+$$

Since each clause contains at least one negative literal, S is satisfied under $\mathcal{I}$. Thus, S is satisfiable.

Exercise 1.1 (b)

Question: Show that $(C \land (D \lor \neg C)) \lor (A \land \neg(B \lor A))$ is logically equivalent to $(C \land D)$ by applying the equivalences from the lecture.

Exercise 1.1 (b)

$$\overset{1}{\equiv} (C \land (D \lor \neg C)) \lor (A \land (\neg B \land \neg A)) \qquad \text{(De Morgan)}$$

$$\overset{2}{\equiv} (C \land (D \lor \neg C)) \lor (A \land (\neg A \land \neg B)) \qquad \text{(Commutativity)}$$

$$\overset{3}{\equiv} (C \land (D \lor \neg C)) \lor ((A \land \neg A) \land \neg B) \qquad \text{(Associativity)}$$

$$\overset{4}{\equiv} (C \land (D \lor \neg C)) \lor (\bot \land \neg B) \qquad \text{(Contradiction)}$$

$$\overset{5}{\equiv} (C \land (D \lor \neg C)) \lor \bot \qquad \text{(Falsity)}$$

$$\overset{6}{\equiv} (C \land (D \lor \neg C)) \qquad \text{(Falsity)}$$

$$\overset{7}{\equiv} (C \land D) \lor (C \land \neg C) \qquad \text{(Distributivity)}$$

$$\overset{8}{\equiv} (C \land D) \lor \bot \qquad \text{(Contradiction)}$$

$$\overset{9}{\equiv} (C \land D) \qquad \text{(Falsity)}$$

Exercise 1.1 (c)

Question: Prove that there is no polynomial algorithm that transforms an arbitrary propositional logic formula into a logically equivalent formula in CNF.

Hint: Find a family $\{\phi_n\}_{n \in \mathbb{N}}$ of formulas in DNF for which you can show the following property: While the size of the formulas ϕ_n grows linear in n, for any given formula ϕ_n **every** equivalent formula in CNF must consist of at least 2^n clauses (and thus cannot be computed in polynomial time).

Exercise 1.1 (c)

Consider the family of DNF-formulas $\phi_n = \bigvee_{i=1}^{n}(X_i \wedge Y_i)$ for $n \in \mathbb{N}^+$.

Example: $n = 2$, exemplary transformation to CNF:

$$
\begin{aligned}
&(X_1 \wedge Y_1) \vee (X_2 \wedge Y_2) \\
\equiv&(X_1 \vee (X_2 \wedge Y_2)) \wedge (Y_1 \vee (X_2 \wedge Y_2)) \\
\equiv&(X_1 \vee X_2) \wedge (X_1 \vee Y_2) \wedge (Y_1 \vee X_2) \wedge (Y_1 \vee Y_2) \\
\equiv&\{\{X_1, X_2\}, \{X_1, Y_2\}, \{Y_1, X_2\}, \{Y_1, Y_2\}\}
\end{aligned}
$$

This particular CNF has $2^2 = 4$ clauses. But how can we show that there is no smaller CNF and, for the general case, that the minimal number of clauses is indeed exponential in n?

Exercise 1.1 (c)

Reconsider the example:

$$\begin{aligned}\phi_2 =& (X_1 \wedge Y_1) \vee (X_2 \wedge Y_2)\\ \equiv& \{\{X_1, X_2\}, \{X_1, Y_2\}, \{Y_1, X_2\}, \{Y_1, Y_2\}\}\end{aligned}$$

Proof Idea: Let ψ be a formula in CNF that is equivalent to ϕ_n

1. As a lemma, we first show that every non-trivial clause $\chi \in \psi$ must contain the atom X_i or the atom Y_i for each $i \in \{1, \dots, n\}$.
2. Using this lemma, we show that for all $A \subseteq \{1, \dots n\}$, the clauses $\chi_A = \{X_i \mid i \in A\} \cup \{Y_i \mid i \notin A\}$ must indeed all be "contained" in ψ. E.g., these are all clauses in our example: $\{X_1, X_2\}$ for $A = \{1, 2\}$, $\{X_1, Y_2\}$ for $A = \{1\}$, $\{Y_1, X_2\}$ for $A = \{2\}$ and $\{Y_1, Y_2\}$ for $A = \emptyset$.
3. Since there are 2^n different subsets of $\{1, \dots, n\}$, we now know that ψ must contain at least 2^n clauses. Since computing any function includes writing the function value as output, this clearly cannot be done in polynomial time.

Exercise 1.1 (c)

The DNF: $\phi_n = \bigvee_{i=1}^{n} (X_i \wedge Y_i)$

Lemma 1: Let ψ be a CNF-formula that is equivalent to ϕ_n. Then every non-trivial (i.e. not equivalent to $\top$) clause in ψ must contain the atom X_i or the atom Y_i for each $i \in \{1, \ldots, n\}$.

Proof: Assume ψ is a formula in CNF (logically equivalent to ϕ_n) with a clause χ without this property. Then χ is falsifiable (otherwise χ would be trivial) and an interpretation $\mathcal{I}$ that makes χ false makes χ still false if it sets $\mathcal{I}(X_i) = \mathcal{I}(Y_i) = \mathbf{T}$, because these variables do not occur in χ. Under this interpretation ψ is false, but ϕ_n is true. Thus, ψ cannot be logically equivalent to ϕ_n.

Exercise 1.1 (c)

The DNF: $\phi_n = \bigvee_{i=1}^{n}(X_i \wedge Y_i)$

Lemma 2: Every CNF-formula ψ that is equivalent to ϕ_n must contain for all $A \subseteq \{1, \ldots, n\}$ the clause $\chi_A = \{X_i \mid i \in A\} \cup \{Y_i \mid i \notin A\}$ or a superset of χ_A which may only contain additional negative literals that don't already occur positively in χ_A.

Proof: For each $A \subseteq \{1, \ldots n\}$, we consider the interpretation $\mathcal{I}$ with $\mathcal{I}(X_i) = \mathbf{T}$ iff $i \notin A$ and $\mathcal{I}(Y_i) = \mathbf{T}$ iff $i \in A$. Since there is no $i \in \{1, \ldots, n\}$ with $\mathcal{I}(X_i) = T$ and $\mathcal{I}(Y_i) = T$, we have $\mathcal{I} \not\models \phi_n$. An equivalent CNF ψ must thus also contain a clause $\chi \in \psi$ such that $\mathcal{I} \not\models \chi$. The clause χ thus can't contain any X_i with $i \notin A$ or any Y_i with $i \in A$. We know from Lemma 1 that if X_i is not contained, Y_i must be contained and vice versa. I.e., χ must be a superset of $\{X_i \mid i \in A\} \cup \{Y_i \mid i \notin A\}$ and may only contain additional negative literals which don't already occur positively in χ_a (since this would make χ valid).

Exercise 1.1 (c)

The DNF: $\phi_n = \bigvee_{i=1}^{n} (X_i \wedge Y_i)$

Lemma 2: Every CNF-formula ψ that is equivalent to ϕ_n must contain for all $A \subseteq \{1, \ldots, n\}$ the clause $\chi_A = \{X_i \mid i \in A\} \cup \{Y_i \mid i \notin A\}$ or a superset of χ_A which may only contain additional negative literals that don't already occur positively in χ_A.

Conclusion: Since the positive atoms in each χ_A differ from the positive atoms in each other $\chi_{A'}$ (given $A \neq A'$), the allowed supersets of χ_A and $\chi_{A'}$ are different. Since there are 2^n distinct subsets $A \subseteq \{1, \ldots, n\}$, we now know that ψ must contain at least 2^n clauses. And since computing any function includes writing the function value as output, this cannot be done in polynomial time. $\square$

Exercise 1.2 (a)

Question: Use resolution to show that

$$F = (\neg A \land \neg B \land C) \lor (A \land \neg B) \lor (\neg A \land \neg C) \lor B$$

is a tautology (valid).

Exercise 1.2 (a)

Idea: To show that F is a tautology, we show that $\neg F$ is unsatisfiable. With De Morgan's law we get F' which is logically equivalent to $\neg F$:

$$F' = (A \vee B \vee \neg C) \wedge (\neg A \vee B) \wedge (A \vee C) \wedge \neg B$$

If we write F' as set of clauses Δ, we get

$$\Delta = \{\{A, B, \neg C\}, \{\neg A, B\}, \{A, C\}, \{\neg B\}\}.$$

Now we can apply the following resolutions:

- $\{A, B, \neg C\}$ and $\{A, C\}$ resolve to $\{A, B\}$.
- $\{A, B\}$ and $\{\neg A, B\}$ resolve to $\{B\}$.
- $\{\neg B\}$ and $\{B\}$ resolve to $\square$.

As $\Delta \vdash \square$, formula F' is unsatisfiable and, thus, F is a tautology.

Exercise 1.2 (b)

Question: Use resolution to show that

$$\{B \land \neg C, (A \land B) \rightarrow (C \lor \neg A)\} \models \neg A$$

Exercise 1.2 (b)

We show the following logical implication:

$$\{B \wedge \neg C, (A \wedge B) \rightarrow (C \vee \neg A)\} \models \neg A$$

This holds iff every interpretation that makes all formulas in the set true makes A false. Therefore, we show that it is impossible to satisfy the formula set in conjunction with the negation of $\neg A$, expressed as a new formula G:

$$G = (B \wedge \neg C) \wedge ((A \wedge B) \rightarrow (C \vee \neg A)) \wedge A$$

We can write G as a set of clauses Δ:

$$\Delta = \{\{B\}, \{\neg C\}, \{\neg A, \neg B, C\}, \{A\}\}$$

Exercise 1.2 (b)

$$\Delta = \{\{B\}, \{\neg C\}, \{\neg A, \neg B, C\}, \{A\}\}$$

Now we can apply the following resolutions:

- $\{B\}$ and $\{\neg A, \neg B, C\}$ resolve to $\{\neg A, C\}$.
- $\{\neg C\}$ and $\{\neg A, C\}$ resolve to $\{\neg A\}$.
- $\{A\}$ and $\{\neg A\}$ resolve to $\square$.

As $\Delta \vdash \square$, formula G is unsatisfiable, which proves $\{B \wedge \neg C, (A \wedge B) \rightarrow (C \vee \neg A)\} \models \neg A$.