Logic in CS Autumn 2025

Problem Sheet 3

S. Krishna

1. Using resolution, show that $P_1 \wedge P_2 \wedge P_3$ is a consequence of

$$F := (\neg P_1 \vee P_2) \wedge (\neg P_2 \vee P_3) \wedge (P_1 \vee \neg P_3) \wedge (P_1 \vee P_2 \vee P_3)$$

2. We have discussed in class that the validity of a CNF can be determined in polynomial time (*How?*). Suppose you are provided with an algorithm, CNF-VAL, which takes a CNF formula as input and returns whether the formula is valid or not, in polynomial time. We claim to use CNF-VAL to check the satisfiability of any CNF formula as follows: First, we run CNF-VAL on the given CNF formula. If the output is `VALID`, we conclude that the formula is `SAT`. If not, we negate the formula, convert it into CNF, and run CNF-VAL again. If the output is `VALID` on the negated formula, we conclude the original formula is `UNSAT`; otherwise, it is `SAT`. Given that CNF-VAL operates in polynomial time, we claim that the satisfiability of any CNF formula can also be determined in polynomial time using the above procedure.

   Is this reasoning correct? (Note: For this question, we do not consider the empty CNF.)

3. From any CNF formula φ, is it possible to compute in polynomial time an equisatisfiable formula $\psi_1 \wedge \psi_2$ where ψ_1 is a Horn formula and ψ_2 is a 2-CNF formula?

4. Let $\mathcal{F}$ and $\mathcal{G}$ be two sets of formulae. We say $\mathcal{F} \equiv \mathcal{G}$ iff for any assignment α, $\alpha \models \mathcal{F}$ iff $\alpha \models \mathcal{G}$ ($\alpha \models \mathcal{F}$ iff $\alpha \models F_i$ for every $F_i \in \mathcal{F}$). Prove or disprove: For any $\mathcal{F}$ and $\mathcal{G}$, $\mathcal{F} \equiv \mathcal{G}$ iff

   (a) For each $G \in \mathcal{G}$, there exists $F \in \mathcal{F}$ such that $G \models F$, and

   (b) For each $F \in \mathcal{F}$, there exists $G \in \mathcal{G}$ such that $F \models G$,

5. A set of sentences $\mathcal{F}$ is said to be closed under conjunction if for any F and G in $\mathcal{F}$, $F \wedge G$ is also in $\mathcal{F}$. Suppose $\mathcal{F}$ is closed under conjunction and is inconsistent ($\mathcal{F} \vdash \bot$). Prove that for any $G \in \mathcal{F}$, there exists $F \in \mathcal{F}$ such that $\{F\} \vdash \neg G$.

6. Define *Positive resolution* as a restriction of ordinary resolution as follows: derive a resolvent from clauses C_1 and C_2 only if C_1 is a positive clause, i.e., it consists only of positive literals. Prove or disprove : If F is an unsatisfiable CNF formula then one can derive the empty clause from F using only positive resolution.

7. Show that there is a polynomial-time algorithm to decide satisfiability of those CNF formulas F in which each propositional variable occurs at most twice. Justify your answer. (Note that this question is not the same as 2-SAT which you did in last tutorial.)

8. Say that a set Σ_1 of wffs is equivalent to a set Σ_2 of wffs iff for any wff α, we have $\Sigma_1 \models \alpha$ iff $\Sigma_2 \models \alpha$. A set Σ is independent iff no member of Σ is tautologically implied by the remaining members in Σ. Show that a finite set of wffs has an independent equivalent subset by describing an algorithm to compute this independent equivalent subset. Prove that your algorithm returns a subset that is independent and equivalent.

9. Consider the parity function, $\mathsf{PARITY} : \{0,1\}^n \to \{0,1\}$, where PARITY evaluates to 1 iff an odd number of inputs is 1. In all of the CNFs below, we assume that each clause contains any variable at most once, i.e. no clause contains expressions of the form $p \wedge \neg p$ or $p \vee \neg p$. Furthermore, all clauses are assumed to be distinct.

   (a) Prove that any CNF representation of PARITY must have n literals (from distinct variables) in every clause.

   (b) Prove that any CNF representation of PARITY must have at least 2^{n-1} clauses.