Homework Assignment 1

Bailey Wickham & Alex MacLean
CSC530

May 20, 2021

1 Propositional Logic and Normal Forms

2. (5 points) Convert the following formula to an equisatisfiable one in CNF using Tseitin's encoding:

$$\neg(\neg r \to \neg(p \land q))$$

Write the final CNF as the answer. Use a_ϕ to denote the auxiliary variable for the formula ϕ; for example, $a_{p\land q}$ should be used to denote the auxiliary variable for $p \land q$. Your conversion should not introduce auxiliary variables for negations.

Substitutions:

$$a_\phi \leftrightarrow \neg(\neg r \to \neg a_{p\land q})$$
$$a_{p\land q} \leftrightarrow (p \land q)$$

Conjunction:

$$a_\phi \land (a_\phi \leftrightarrow \neg(\neg r \to \neg a_{p\land q})) \land (a_{p\land q} \leftrightarrow (p \land q))$$
$$\textit{where}$$
$$(a_\phi \leftrightarrow \neg(\neg r \to \neg a_{p\land q})) \equiv (\neg a_\phi \lor \neg r) \land (\neg a_\phi \lor a_{p\land q}) \land (r \lor \neg a_{p\land q} \lor a_\phi)$$
$$\textit{and}$$
$$(a_{p\land q} \leftrightarrow (p \land q)) \equiv (\neg a_{p\land q} \lor p) \land (\neg a_{p\land q} \lor q) \land (\neg p \lor \neg q \lor a_{p\land q})$$
$$\textit{so}$$
$$\phi \equiv a_\phi \land (\neg a_\phi \lor \neg r) \land (\neg a_\phi \lor a_{p\land q}) \land (r \lor \neg a_{p\land q} \lor a_\phi) \land$$
$$(\neg a_{p\land q} \lor p) \land (\neg a_{p\land q} \lor q) \land (\neg p \lor \neg q \lor a_{p\land q})$$

3. (10 points) Let ϕ be a propositional formula in NNF, and let I be an interpretation of ϕ. Let the *positive set* of I with respect to ϕ, denoted $pos(I, \phi)$, be the literals of ϕ that are satisfied by I. As an example, for the NNF formula $\phi = (\neg r \land p) \lor q$ and the interpretation $I = [r \mapsto \bot, p \mapsto \top, q \mapsto \bot]$, we have $pos(I, \phi) = \{\neg r, p\}$. Prove the following theorem about the monotonicity of NNF:

Monotonicity of NNF: For every interpretation I and I' such that $pos(I, \phi) \subseteq pos(I', \phi)$, if $I \models \phi$, then $I' \models \phi$.

(**Hint:** Use structural induction.)

Proof. Proceed by structural induction.

Base case: Suppose that ϕ is of the form: p or $\neg p$ and $I \models \phi$. Then $pos(I, \phi)$ must contain p or $\neg p$ respectively. Since $pos(I, \phi) \subseteq pos(I', \phi)$ then that element must also be in $pos(I', \phi)$. Therefore $I' \models \phi$.

Inductive hypothesis: Suppose there exists a ϕ_1 and ϕ_2 such that for every interpretation I and I' such that $pos(I, \phi_i) \subseteq pos(I', \phi_i)$, if $I \models \phi_i$, then $I' \models \phi_i$.

Inductive step: Let $\phi = \phi_1 \wedge \phi_2$ and I satisfy ϕ. Let $pos(I, \phi) \subseteq pos(I', \phi)$ for some interpretation I'. Then $I \models \phi_1$ and $pos(I, \phi_1) \subseteq pos(I, \phi) \subseteq pos(I', \phi)$ so by the inductive hypothesis $I' \models \phi_1$. A similar argument can be applied to ϕ_2. Since $I' \models \phi_1$ and $I' \models \phi_2$, $I' \models \phi$. The case for $\phi = \phi_1 \vee \phi_2$ follows similarly.

Conclusion: By induction, every interpretation I and I' such that $pos(I, \phi) \subseteq pos(I', \phi)$, if $I \models \phi$, then $I' \models \phi$.

$\square$

4. (10 points) Let ϕ be an NNF formula. Let $\hat{\phi}$ be a formula derived from ϕ using a modified version of Tseitin's encoding in which the CNF constraints are derived from implications rather than bi-implications. For example, given the formula

$$a_1 \wedge (a_2 \vee \neg a_3),$$

the new encoding is the CNF equivalent of the following, where x_0, x_1, x_2 are fresh auxiliary variables:

$$\begin{array}{ll} x_0 & \wedge \\ (x_0 \rightarrow a_1 \wedge x_1) & \wedge \\ (x_1 \rightarrow a_2 \vee x_2) & \wedge \\ (x_2 \rightarrow \neg a_3) & \end{array}$$

Note that Tseitin's encoding to CNF starts with the same formula, except that $\rightarrow$ is replaced with $\leftrightarrow$. As a result, the new encoding has roughly half as many clauses as the Tseitin's encoding.

Prove that $\hat{\phi}$ is satisfiable if and only if ϕ is satisfiable.

(**Hint**: Use the theorem from Problem 3.)