CS226 Practice Problem Set 5 (Spring 2016)

Date posted: Mar 14, 2016 **Expected Solving Time: 1 hour**

- *Be brief, complete and stick to what has been asked.*
- *Unless asked for explicitly, you may cite results/proofs covered in class without reproducing them.*
- *If you need to make any assumptions, state them clearly.*
- ***The problems marked $\star$ are "must-solve" problems, i.e., you are required to submit solutions of these problems through moodle. The remainder of the problems are for practice.***
- ***You are strongly encouraged to solve these on your own to ensure you understand the material being taught in class.***
- ***Mutual discussion on the non-$\star$ problems is allowed. Please try to solve the $\star$ problems on your own. Please read the guidelines on the course webpage if you don't understand the distinction between the two.***

1$\star$. Consider the ROBDD with complement edges, shown in Fig. 1. We wish to construct an AIG for the Boolean function represented by the uncomplemented root edge (or root node) of this ROBDD. An algorithm for doing this is given below.

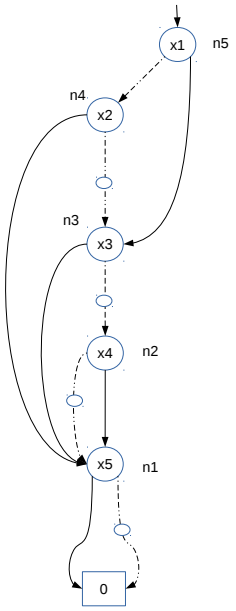

Figure 1: ROBDD with complement edges

(a) Process each ROBDD node n in bottom-to-top order (from the leaves to the root) as follows.

i. Construct an AIG for the function represented by the ROBDD node n. In order to do this, use AIGs for already processed ROBDD nodes as (possibly shared) sub-AIGs. You must use the AIG for the ite operator, given in the lecture slides, for this purpose. You must also use structural hashing and the re-write rules mentioned in the lecture slides **(and no other re-write rules)**.

Present your answer in the form of a table as shown below.

ROBDD node	AIG before applying re-writes	AIG after applying re-writes	Re-write rules applied
n_1			
n_2			
n_3			
n_4			
n_5			

Note that giving an answer without specifying the rewrite rules will not fetch any credit.

2. Consider a Boolean function f on the variables p, q, r, s and t defined as follows:

   - If q is 1, then $f(p, q, r, s, t)$ is 1 if and only if at least one of the variables p and t is 0.
   - If q is 0, then $f(p, q, r, s, t)$ is 1 if and only if r and s have different values.

   (a) Construct an ROBDD for the function f using the variable order $p < q < r < s < t$ (i.e. p above q above r ...)

   (b) Construct a FRAIG for f using the ROBDD obtained in part (a). You must clearly indicate which node in your FRAIG corresponds (represents the same function as) which node in the ROBDD.

   (c) The function f can also be expressed as $\mathsf{ite}(q, g(p, t), h(r, s))$, where $g(p, t) = (p.t)'$ and $h(r, s) = r \oplus s$. Starting from this description of the function, construct an AIG for $f(p, q, r, s, t)$, sharing nodes from the AIG constructed in part (b), whenever possible (use structural hashing).

   (d) Suppose we now fraig the combined AIG (from parts (b) and (c)) obtained above, merging functionally equivalent nodes and removing nodes that do not feed into other nodes. Show the resulting FRAIG that will result. Does your result depend on which of two functionally equivalent nodes you choose to retain when you do fraig-ing?

3⋆. Consider the circuit marked "ORIGINAL CIRCUIT" shown in Fig. 2. Here solid black circles represent connections between wires, and empty circles represent the presence of an inverter.

   (a) Give the satisfiability don't care (SDC) set at n as a sum-of-products expression on the variables n, m, b, c and d.

   (b) Give the satisfiability don't care (SDC) set at p as a sum-of-products expression on the variables p, b, c and d.

   (c) Give the observability don't care (ODC) sets of node n at output r and at output q, as two sum-of-products expressions on the variables b, c and d. Note you are required to compute two ODC SOP expressions.

   (d) A designer has found out that $\mathsf{ite}(d, b', a)$ is an external don't care (EXDC) set for this circuit. In other words, the values of a, b and d fed to this circuit can be assumed to be such that $\mathsf{ite}(d, b', a)$ won't evaluate to 1. Based on this, the designer proposes to eliminate the OR-gate with output n completely, and use the output l of the exor gate in its place. The resulting simplified circuit is shown as "SIMPLIFIED CIRCUIT" in Fig. 2.

   Of course, such a simplification cannot be done in general, but the designer claims that the various EXDC, SDC and ODC sets in this circuit are such that this can be done. You are required to either show that the designer's claim is correct, or it is incorrect.

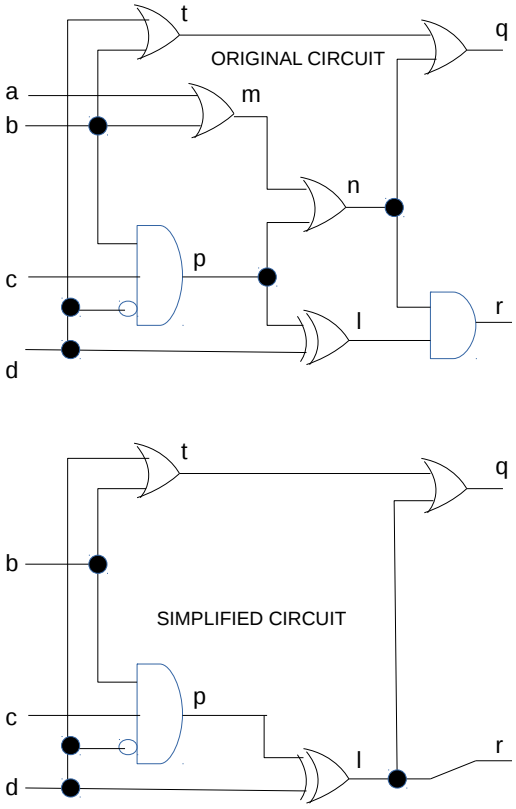

Figure 2: ROBDD with complement edges (and root edge)

If you think the designer's claim is correct, show how all minterms in $n \oplus l$ are covered by the given EXDC set, and from the SDC and ODC sets from the previous sub-questions.

If you think the designer's claim is incorrect, provide a set of input values (of a, b, c, d) such that this valuation lies outside the EXDC set (i.e. doesn't satisfy $\mathsf{ite}(d, b', a)$) and produces values of q and/or r that differ in the two circuits shown in Fig. 2.