Introduction to EDA 2024 Bonus Assignment
IWLS 2024 Programming Contest

2024/04/18

1 Problem Description

Please check the IWLS 2024 programming contest `https://www.iwls.org/iwls2024/contest/IWLS%202024%20Programming%20Contest.pdf` for the detailed description. In this bonus assignment, we only consider circuits in AIG. The benchmarks are given as truth tables. The objective is to derive an equivalent AIG with minimum number of nodes. **Please submit your circuits in AIG format and make sure that the submitted circuit is functionally equivalent to the given truth table.**

2 Bonus Criteria

There are 100 cases in total. We will release the baseline node counts for all cases. After the deadline, for each case, we will keep the smallest equivalent AIG (smaller than the baseline) as the best result. **If multiple students submit AIGs with the same number of nodes, the first submitted AIG will be kept.** Finally, for each student, the bonus point (in semester score) is given by

$$\log_2((\text{number of best results}) + 1).$$

Each student can get up to 5 points. In addition, the student must specify exactly how to derive the submitted AIG from the given truth table to get this bonus point.

3 Submission

- AIG Sumbission Website: `http://alcom.ee.ntu.edu.tw/iwls`.
- AIG Submission Deadline: 6/14 (Fri) 23:59
- Report (Source Code) Submission Deadline: 6/16 (Sun) 23:59

Please submit your AIG along with your student ID and the case number (from 00 to 99) on the website. The current best and the current winners of all cases are listed on the website. The equivalence will be checked using ABC upon submission and the AIG will be accepted only if it is equivalent to the truth table provided by the IWLS Programming Contest.

In addition, please submit a report on how you derive the solution to NTU COOL. If you use a combination of ABC commands, please provide the used synthesis scripts. If you write your own procedure to derive the solution, please provide your source code and a README file on how we can compile/run your code. Please submit your source code along with your report to NTU COOL.

4 ABC Usage

Add option `-h` in any command to see the detailed description and usages. For more commands, please refer to `https://people.eecs.berkeley.edu/~alanmi/abc/`

4.1 Read/Write

- "`read_truth -f [.truth file]`": read the given truth table.
- "`strash`": convert current network into an AIG.
- "`write_aiger [.aig file]`": save the current AIG.

4.2 Verifying Results

- To check the equivalence, after reading the given truth table, run "`cec -n [.aig file]`" to verify the equivalence between the minimized AIG and the truth table (option "`-n`" is for disabling matching PO by names).

4.3 Optimization (ABC9)

These commands start with `&` and use a different data structure (GIA) to represent AIGs.

- `&r [File Name]`: read AIG.
- `&w [File Name]`: write current AIG.
- `&ps`: print the statistics of current AIG.
- `&put`: transfer the current network into the old ABC.
- `&get`: transfer the current network into ABC9.
- `&deepsyn`, `&dc2`, `&b`, `&resub`, `&fx`: some synthesis commands