Weekly Report

March 25, 2019

1 Targets

1.1 Urgent

- Randomly selecting a node, compute its care set, change its care set, synthesize the local circuit, evaluate its area and error rate, accept it with a certain probability.

1.2 Important

- Use model counting to compute error rate, i.e., how many assignments satisfying a SAT problem.
- Use approximate confidence interval / hypothesis testing of Bernoulli experiments to evaluate the accuracy of error rate.
- Trade off the accuracy of batch error estimation for speed (even directly use Su's equation to update Boolean difference), perhaps use hypothesis testing to evaluate the accuracy.
- Combine the simulation of circuits with the simulation of Monte Carlo Tree Search. In other words, in one loop of Monte Carlo Tree Search, merge logic simulation and playout (only simulate circuit once and playout once).
- Represent circuit with AIG because of more potential LAC candidates. For each round, select one or more input wires and replace them with constant 0 or 1. Consider how to combine Wu's method (choose a subset of input wires and substitute).
- Accelerate Approximate Logic Synthesis Ordered by Monte Carlo Tree Search: reuse the result of batch error estimation in playout.
- How DC affects BDD
- Induce DC with PLA files

1.3 Worth Trying

- Enhance default policy with greedy approach or field domain knowledge.
- In expansion process of MCTS, expand more than one layers.
- Tune parameters in MCTS.
- Perform greedy flow on leaves of the final Monte Carlo Search Tree.
- Combine beam search and MCTS.

1.4 Potential Topics

- Relationship between power simulation and logic simulation.
- Combine Binarized Neural Network with approximate computing.
- Relationship between Boolean network and Bayesian Network.
- Approximate TMR.

2 Progress

To implement

$$\hat{\mathbf{y}} = (\hat{f}_1(\mathbf{x}), \ldots, \hat{f}_O(\mathbf{x})),$$

where

$$\hat{f}_i(\mathbf{x}) = f_i(\mathbf{x})|_{\mathbf{x}=\mathbf{a}_1} \vee \ldots \vee f_i(\mathbf{x})|_{\mathbf{x}=\mathbf{a}_R}$$

I did 2 experiments.

2.1 Experiment 1

Directly build the level circuit according to its SOP expression, and synthesize it with the script "resyn2". For c880, original area is 607, when $R = 100$, approximate circuit has an area of 3324. It is impractical, since it seems hard for the multi-level synthesis tool to simplify a 2-level SOP expression circuit.

2.2 Experiment 2

Let $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_R]$ be the care set. Simplify the windows $\mathbb{W}$ with the care set.
Also for c880,

R	area	ER
100	526	0.11104
200	539	0.024631
206	539	0.024631
212	542	0.004503
225	542	0.004503
250	542	0.004503
300	543	0.004503
500	554	0.003345
1000	553	0.000701

For c1908,

R	area	ER
100	374	0.082238
1000	567	0.028422
2000	625	0.008439
2500	657	0.000369

When ER is small, similar result compared with Su-TCAD is generated. But ER changes abruptly near $R = 210$.
Reason of great change might be:

- I use interpolation method to simplify the circuit. Sometimes when care set changes, interpolation does not change.
- I directly change the care set of primary input. Slightly change on PIs may cause large disturbance. Maybe partial change will work more elaborately.