PA2 report

R08943064 王菖浚

Implementation:

APIs: satSolver, cnf, aig, abc

The total concept is that I want to transform the input aig network into cnf form, so that I can take advantage of SAT solver to solve the unateness question. And for each prime input I will generate its positive cofactor and negative cofactor. By testing the implication relationship between these two cofactor, I can know the unateness for this prime input.

First, we will read on a aig network(Abc_Ntk_t), then transforms it to a cnf. I duplicate this cnf in order to generate two different cofactor function and add them into the SAT solver. Moreover, I add $(\neg\alpha_i \vee (x_i = y_i))$ into SAT solver for each prime input (x_i means the positive cofactor's ith prime input, y_i means the negative cofactor's ith prime input), which α_i can be viewed as an enable variable because I need to make all input the same expect the input under testing. In the end, I also add four assumption that $x_i = 1/0$, $y_i = 0/1$, $c_j = 1$, $c'_j = 0$ for testing positive and negative unateness respectively.

Can your implementation solve the remaining 7 test cases (possibly with a longer time limit)?

Because I basically only finish the baseline and don't have any improvement for optimization, some cases cannot be finished in time constraint. I think it's possible to solve the remaining tests for looser test limit.