In Proceedings of Cadence Technical Conference (CTC), May 1998.

A Model and Algorithm for Computing Minimum-Size Test Patterns

Paulo F. Flores, Horácio C. Neto, Krishnendu Chakrabarty and João P. Marques Silva
Cadence European Laboratories
IST/INESC
Lisbon, Portugal

Abstract —

This paper addresses the problem of test pattern generation for single stuck-at faults in combinational circuits, under the additional constraint that the number of specified primary input assignments is minimized. This problem has different applications in testing, including the identification of don't care conditions to be used in the synthesis of Built-In Self-Test (BIST) logic. The proposed solution is based on an integer linear programming (ILP) formulation which builds on an existing Propositional Satisfiability (SAT) model for test pattern generation. This ILP formulation is linear on the size of the original SAT model for test generation, which is linear on the size of the circuit. Nevertheless, the resulting ILP instances represent complex optimization problems, that require dedicated ILP algorithms. Preliminary results on benchmark circuits validate the practical applicability of the test pattern minimization model and associated ILP algorithm.

1. Introduction

Automatic test pattern generation (ATPG) for stuck-at faults in combinational circuits is now a mature field, with an impressive number of highly effective models and algorithms [3, 4, 7, 8-11][1]. Furthermore, besides being effective at detecting the target faults, recent ATPG tools have aimed the heuristic minimization (i.e. compaction) of the total number of test patterns required for detecting all faults in a circuit [1, 8, 9]. In general, the degree of test pattern compaction is expected to be related to the number of unspecified input assignments in each test pattern. Moreover, recent work on using deterministic test patterns for the synthesis of Built-In Self-Test (BIST) logic [1] also motivates the computation of test patterns for which the number of unspecified primary input assignments is minimized. Indeed, if the test set is used as input to a logic synthesis tool with the purpose of synthesizing BIST logic, then by maximizing the number of unspecified input assignments, i.e. by maximizing the don't care set of each test pattern, the logic synthesis tool is in general able to yield smaller synthesized logic. Thus the maximization of the don't care set of each test pattern, or conversely, the computation of test patterns of minimum-size, can have significant practical consequences.

1. A more comprehensive bibliographic review of recent ATPG algorithms can be found in [2].

Nevertheless, there exists no model or algorithm in the literature for computing test patterns for which the number of unspecified primary input assignments is maximized. Accordingly, the main objective of this paper is to propose a first attempt at solving this problem. We develop a new model for test pattern generation, based on propositional satisfiability (SAT), in the presence of unspecified input assignments. Afterwards, we derive an integer linear programming (ILP) model for maximizing the number of unspecified primary input assignments. Finally, we provide preliminary results that justify using the proposed model in medium-size combinational circuits and describe an ATPG methodology, which can incorporate the proposed model and supporting algorithm, and which can also be applied to large-size combinational circuits. Besides its practical applicability, to our best knowledge this is the first formal non-heuristic model towards computing minimum size test patterns.

2. Model and Algorithm

In this section we briefly outline the integer optimization model for computing minimum-size test patterns. The main steps for constructing the model are as follows:

1. The first step is to represent circuits and fault detection problems using Conjunctive Normal Form (CNF) formulas. In this paper, the model of [10] is assumed but the models of [3, 11] could also be used.
2. The next step is to develop a CNF model in which variables can have unspecified assignments. Notice that solving SAT requires that all variables must be specified. Consequently, a dedicated formal model needs to be developed. (This model is detailed in [2].)
3. Afterwards, we apply the resulting CNF model to the representation of circuits and fault detection problems.
4. We can then map the resulting CNF into an ILP model. This step is straightforward, since clauses can always be viewed as algebraic inequalities.
5. Finally, we specify the cost function of the resulting integer optimization model so that the total number of specified assignments is minimized.

It can be shown that the proposed ILP model is indeed correct [2]. Furthermore, this model has a search space of

Benchmark	#F	HITEC		ATALANTA		MTP	
		#T	#SI	#T	#SI	#T	#SI
9symml	388	144	1296	80	720	80	696
cht	400	30	1410	15	705	10	241
cm138a	63	18	108	11	66	12	70
cm150a	99	66	1386	35	735	34	236
cm163a	106	32	512	13	208	11	110
cmb	121	70	1120	30	480	26	327
majority	28	20	100	11	55	11	53
misex1	127	34	272	—	—	18	87
misex2	249	102	2550	—	—	39	459
mux	100	68	1428	34	714	34	238
sao2	286	104	1040	47	470	50	460

Table 1: Total number of tests and specified input assignments

$O\left(2^{2|PI|}\right)$, hence significantly larger than the search space for plain fault detection. As a result dedicated ILP algorithms, targeted at highly constrained ILPs, have been developed [6]. These algorithms are built on top of Propositional Satisfiability (SAT) algorithms, that are specifically targeted at solving highly constrained instances of SAT.

3. Experimental Results

Preliminary experimental results, comparing a prototype tool for computing minimum-size test patterns, MTP, and two existing test-pattern generators, HITEC [7] and ATALANTA [4], are shown in Table 1. For these results, we selected some of the MCNC [5] benchmarks. (In the final version of the paper a more complete set of experimental results will be provided.)

For each benchmark circuit and for each tool, we include the total number of test patterns (**#T**) as well as the total number of specified primary input assignments (**#SI**). As can be observed, the proposed algorithm, MTP, allows the identification of minimum-size test patterns and hence it yields a significantly smaller number of specified input assignments in all cases but one; the exception being observed when the final number of test patterns is larger in MTP.

4. Conclusions

In this paper we introduce a new integer optimization model for computing minimum-size test patterns. Preliminary experimental results validate the practical applicability of the model. Additional research work includes developing more effective ILP algorithms and applying the proposed model and algorithm in BIST logic synthesis.

References

[1] K. Chakrabarty, B. T. Murray, J. Liu and M. Zhu, “Test Width Reduction for Built-In Self Testing”, in *Proceedings of the International Test Conference*, October 1997.

[2] P. F. Flores, J. P. Marques Silva, H. C. Neto and K. Chakrabarty, “An Exact Solution to the Minimum-Size Test Pattern Problem,” Technical Report RT/12/97, INESC, Portugal, December 1997. (Available from http://algos.inesc.pt/pub/users/jpms/tech-reports/RT-12-97.ps.gz.) (Submitted for publication.)

[3] T. Larrabee, “Test Pattern Generation Using Boolean Satisfiability,” *IEEE Transactions on Computer-Aided Design*, vol. 11, no. 1, pp. 4-15, January 1992.

[4] H. K. Lee and D. S. Ha, “On the Generation of Test Patterns for Combinational Circuits,” Technical Report No. 12_93, Department t of Electrical Engineering, Virginia Polytechnic Institute and State University, 1993.

[5] IWLS’89 Benchmark Suite, available from http://www.cbl.ncsu.edu/pub/Benchmark_dirs/LGSynth89/.

[6] V. Manquinho, P. Flores, J. P. M. Silva and A. Oliveira, “Prime Implicant Computation Using Satisfiability Algorithms,” in *Proc. of the IEEE International Conference on Tools with Artificial Intelligence*, November 1997.

[7] T. M. Niermann and J. H. Patel, “HITEC: A test generation package for sequential circuits,” in *Proceedings of the European Conference on Design Automation (EDAC)*, February 1991.

[8] I. Pomeranz, L.N. Reddy, S.M. Reddy, “COMPACTEST: A Method to Generate Compact Test Sets for Combinational Circuits,” in *IEEE Transactions on Computer-Aided Design*, vol. 12, no. 7, pp. 1040-1049, July 1993.

[9] M. H. Schulz and E. Auth, “Improved Deterministic Test Pattern Generation with Applications to Redundancy Identification,” *IEEE Transactions on Computer-Aided Design*, vol. 8, no. 7, pp. 811-816, July 1989.

[10] J. P. M. Silva and K. A. Sakallah, “Robust Search Algorithms for Test Pattern Generation,” in *Proceedings of the Fault-Tolerant Computing Symposium*, June 1997.

[11] P. R. Stephan, R. K. Brayton and A. L. Sangiovanni-Vincentelli, “Combinational Test Generation Using Satisfiability,” *IEEE Transactions on Computer-Aided Design*, vol. 15, no. 9, pp. 1167-1176, September 1996.