Applying Artificial Neural Networks to Test-point Insertion: Delay Fault Coverage and Training Circuit Generation

Spencer Millican, Yang Sun, Soham Roy, and Vishwani Agrawal
Auburn University, Auburn, AL, USA

Abstract

This presentation demonstrates the benefits of using artificial neural networks (ANNs) to select test-point (TP) locations in a logic built-in self-test (LBIST) environment. ANNs were trained to evaluate the impact a TP has on delay fault coverage, and when implemented and compared to other methods in literature, increased stuck-at and delay fault coverage is obtained with significantly less computation time. Future endeavors will be directed to removing reliance on intellectual property (IP) for ANN training and using features to further increase the ANN's utility.

1. Introduction

TP insertion (TPI) has been studied repeatedly in literature and is used throughout industry to increase the effectiveness of LBIST. LBIST is used throughout industry due to its significant fault coverage and ability to be applied "in-the-field" to guarantee continued device reliability: this latter feature is essential to modern circuits used in life-critical applications. To improve the quality of LBIST tests, TPs are used to modify circuits under test to detect random pattern-resistant (RPR) faults, to which TPs complement additional LBIST improvements, such as weighted random patterns and deterministic seeding.

Although TPI literature has an established history, TPI computation time and quality requires improvement as circuit complexity increases. TPI methods rely on heuristics to reduce computation time without sacrificing fault coverage, and the computational complexity of these heuristics is proportional to the size of the circuit being analyzed. Unfortunately, this rate of computational complexity can easily increase faster than available computational resources. This forces designers to reduce TPI effort and therefore decrease LBIST defect coverage.

By leveraging the latest computing paradigms, i.e. ANNs, existing computational resources can be leveraged with increased efficiency and higher fault coverages can be obtained using TPI. Recent literature has demonstrated ANNs applied to various EDA and test problems, including TPI [1]. This presentation shows extensions to [1] which show how to further apply ANNs to increase both stuck-at and delay fault coverage, which is needed since TPs have been shown to decrease delay fault coverage [2]. Additionally, a potential source of training data without accessing third-party IP circuits is explored.

2. Discussion

The first contribution of this presentation is a method to train an ANN to select TPs such that transition delay fault (TDF) coverage is maximized. This method is compared against an analogous method in literature [3] which uses the same TP quality-measuring information, and these methods are implemented in original software to implement a fair computational resource comparison. A corollary contribution finds delay fault coverage is significantly increased with identical TP overhead with significantly reduced CPU time. Additionally, stuck-at fault coverage is comparable to the previous study [1], which suggests targeting TDF faults is sufficient to achieve adequate stuck-at fault coverage, as was suggested in [2].

The second contribution of this presentation is an exploration of random circuit generation and its effect on ANN training quality. This and other EDA projects which use ANNs currently rely on industry circuits to train their ANNs, but the availability of such circuits is questionable given the secretive nature of IP circuits. This study generated random circuits and found ANNs trained using these circuits have utility, and this motivates future studies to better emulate industry-representative circuits.

The results in this presentation motivate many future directions in applying ANNs to established DFT techniques. TPI can further be improved by analyzing additional circuit features, ATPG can be improved using ANNs for backtracking, LBIST weights can be selected using ANNs, etc.

References

[1] Y. Sun and S. Millican, "Test Point Insertion Using Artificial Neural Networks," in *Proc. IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, Miami, FL, 2019.

[2] S. Roy, B. Stiene, S. Millican, and V. Agrawal, "Improved Random Pattern Delay Fault Coverage Using Inversion Test Points," in *Proc. IEEE 28th North Atlantic Test Workshop (NATW)*, Essex, VT, 2019.

[3] H.-C. Tsai, K.-T. Cheng, C.-J. Lin, and S. Bhawmik, "A Hybrid Algorithm for Test Point Selection for Scan-based BIST," in *Proc. 34th Design Automation Conference (DAC)*, 1997, pp. 478–483.