Pseudo-Random Test Vector Propagation for Fault Coverage Simulation Speed-up with Counter & LFSR Variations

Richard Remigoso, Tarek Alzeiback, Kevin Calva

Department of ECE, University of Illinois at Chicago, Chicago, IL
ECE464 - Testing and Reliability of Digital Systems, Fall 2019

Students: rremig2@uic.edu, malzei3@uic.edu, calva2@uic.edu
Professor: wenjing@uic.edu

ABSTRACT

With circuits being increasingly designed for complex applications and solutions, the amount of potential faults in them has increased. Large circuits are accompanied by more logic gates and other electronic components, which in turn also contribute to the overall incurred test time for the digital system's reliability. Existing test methods include feeding a circuit bench file with the full list of the possible test vectors. While this may guarantee the coverage of all possible faults, the fault detection circuit simulator may result with a very slow run time. This is especially true with large n-bit circuits. A possibly more efficient way of conducting fault detection is through the use of pseudo-randomly propagated test vectors. By generating this *'non-systematic'* test vector, it may be able to detect a higher amount of fault coverage with less test vectors used. Experiments are performed on different sized circuit bench files to test the extent of each set of generated test vectors. This paper answers the following question: "Does utilizing random TV propagation produce a more efficient and effective way for fault coverage than a full list of possible 2^n TVs, where n is the number of inputs in the circuit?"

1 INTRODUCTION

As more and more complex circuits are developed, finding a more efficient and effective way to test these circuits will be necessary. One method could be through the use of a *linear-feedback shift register (LFSR)* or a *counter* to produce a pseudo-randomly generated set of test vectors. The goal of this experiment is to indicate the fault coverage efficiency between each randomly generated sets of test vectors. Five different variations of LFSR and counters will be used. These are: ***(A)*** *single n-bit counter,* ***(B)*** *multiple 8-bit counters with similar seeds,* ***(C)*** *multiple 8-bit counters with different seeds,* ***(D)*** *multiple 8-bit LFSR with similar seeds, and* ***(E)*** *multiple 8-bit LFSR with different seeds.* It is hypothesized that using multiple LFSR with *different* seeds for test vector propagation will produce a more efficient fault coverage than any of the other methods. This will be tested by using 3 different circuits of increasing sizes.

2 METHODS AND PROCEDURES

A fault coverage circuit simulator from Project 2 will be extensively used for this experiment. Benchmark files of the 3 different circuits will be the input of the simulator to produce a list of faults. The simulator is responsible for producing both random and full test vector list, using **LFSR** and a **counter**, respectively. At most 256 test vectors will be created per set. Once, the test vectors are produced, both sets will be simulated and results will be collected to create a *Fault Coverage Graph.* Three graphs will be produced for each circuit bench file with different lines indicating each test vector sets. Using graphical data, the set with the most fault coverage by percentage will be determined as the most efficient and effective way of testing a circuit.

2.1 *Circuits and benchmarks:*

Using multiple benchmark sizes will give a better understanding of how circuit simulator would behave with different test vectors. Using only small scale circuits may not be enough, therefore, using different circuit sizes would be optimal for this experiment.

3 **RESULTS**

3.1 *Test Vector Overview:*

Depending on the input size n of the simulated circuit, the test vector size will be $\lceil \frac{n}{8} \rceil$. For the sake of consistency, the test vector size will always be displayed in groupings of 8. Therefore, there will be overhead bits if the input size is smaller than the test vector size. Despite this, the overall simulation result will be correct due to the nature of the simulator that only takes n amount of bits starting from the least significant bit.

3.2 *Circuit Data Tables by Batch of 10:*

The following are data tables of fault coverage for each circuit. Each table will have 5 lines that indicates which method is used to generate random test vector.

Legends table	
Label	**Method**
Set A	Single n-bit Counter
Set B	Multiple 8-bit Counters with similar seeds
Set C	Multiple 8-bit counters with different seeds
Set D	Multiple 8-bit LFSR with similar seeds
Set E	Multiple 8-bit LFSR with different seeds

5-Input Circuit						
Test Vector #	**Set A (%)**	**Set B (%)**	**Set C (%)**	**Set D (%)**	**Set E (%)**	**Batch Size**
1	77	77	77	94	94	10
2	98	98	98	98	98	20
3	100	100	100	100	100	30
4	100	100	100	100	100	40
5	100	100	100	100	100	50
...						
24	100	100	100	100	100	240
25	100	100	100	100	100	250

Table 1. Circuit with 5 inputs and 2 outputs. Overall, it seems that all sets of test vectors were able to cover all the faults available in the circuit starting at test vector group 3. However, it seems that Set D and E may have been more effective as there first two test vector groups have already covered more than 90% of the faults.

36-Input Circuit						
Test Vector #	**Set A (%)**	**Set B (%)**	**Set C (%)**	**Set D (%)**	**Set E (%)**	**Batch Size**
1	30	31	31	35	55	10
2	30	38	39	47	72	20
3	30	39	44	52	80	30
4	39	46	55	53	81	40
5	40	48	60	56	83	50
6	40	49	63	56	86	60
7	40	49	63	57	86	70
8	40	49	63	57	88	80
9	40	49	63	59	88	90
10	42	50	66	61	88	100
11	42	50	67	61	89	110
12	42	51	68	61	92	120
13	42	56	76	61	92	130
14	42	58	78	61	92	140
15	42	58	79	61	92	150
16	42	58	84	61	92	160
17	42	60	84	61	92	170
18	42	61	84	61	92	180
19	42	61	84	61	92	190
20	42	61	84	61	92	200
21	42	61	84	61	92	210
22	42	61	84	63	92	220
23	42	61	86	63	92	230
24	42	61	86	63	92	240
25	42	63	86	63	92	250

Table 2. Circuit with 36 inputs and 7 outputs. Based on the table provided, none of the test vector sets were able to cover the whole full fault list with the maximum coverage at 92% with Set E. Out of all the sets, Set E is the most efficient and effective with already 80% coverage starting at test vector group 3.

60-Input Circuit						
Test Vector #	**Set A** (%)	**Set B** (%)	**Set C** (%)	**Set D** (%)	**Set E** (%)	**Batch Size**
1	18	53	55	59	61	10
2	19	63	61	73	72	20
3	19	65	67	77	77	30
4	19	70	69	79	82	40
5	19	72	70	83	85	50
6	19	73	71	83	85	60
7	19	77	74	86	86	70
8	19	80	75	86	87	80
9	19	80	76	87	89	90
10	19	81	77	87	90	100
11	19	81	77	87	90	110
12	19	81	77	87	90	120
13	19	82	79	87	90	130
14	19	83	80	87	90	140
15	19	85	80	87	90	150
16	20	86	80	88	90	160
17	20	86	80	88	90	170
18	20	87	80	89	91	180
19	20	87	81	90	91	190
20	20	89	82	90	91	200
21	20	89	82	90	91	210
22	20	89	82	90	92	220
23	20	90	82	90	92	230
24	20	90	82	90	92	240
25	20	90	82	90	92	250

Table 3. Circuit with 60 inputs and 26 outputs. Similar to **Table 2** above, none of the test vector sets had a full coverage of the full fault list, and Set E has the most effective fault coverage of 92%. Set D and E produced a nearly identical result with a few deviations that would determine Set E to be the most efficient with 82% coverage starting at test vector group 4.

3.3 *Graphs:*

Refer to **Table 1-3** above to see the data points of each circuit.

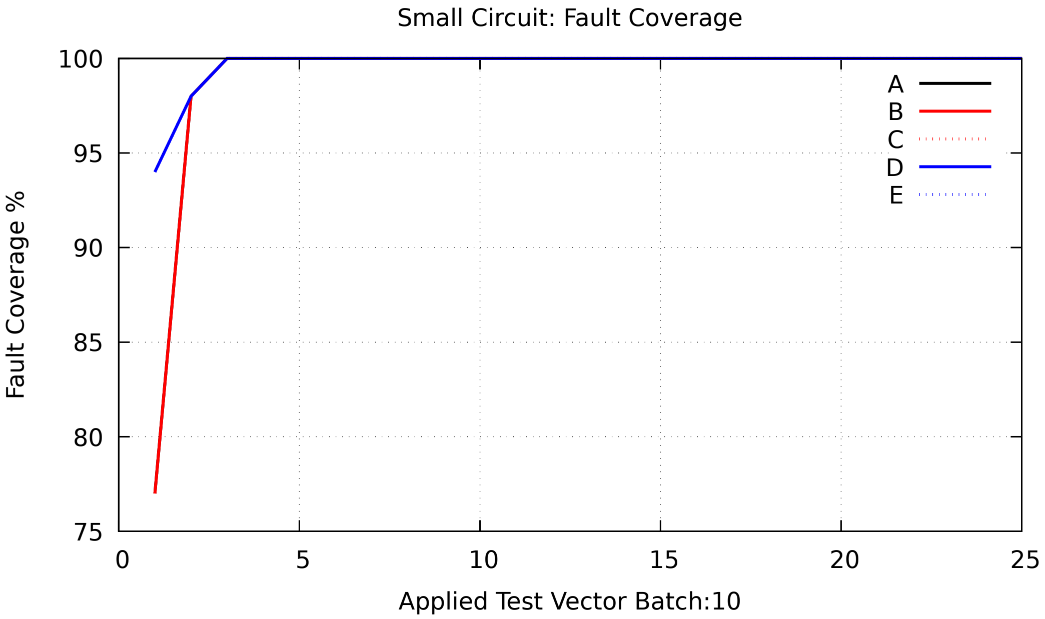

Figure 1. All test vector sets are able to achieve 100% fault coverage.

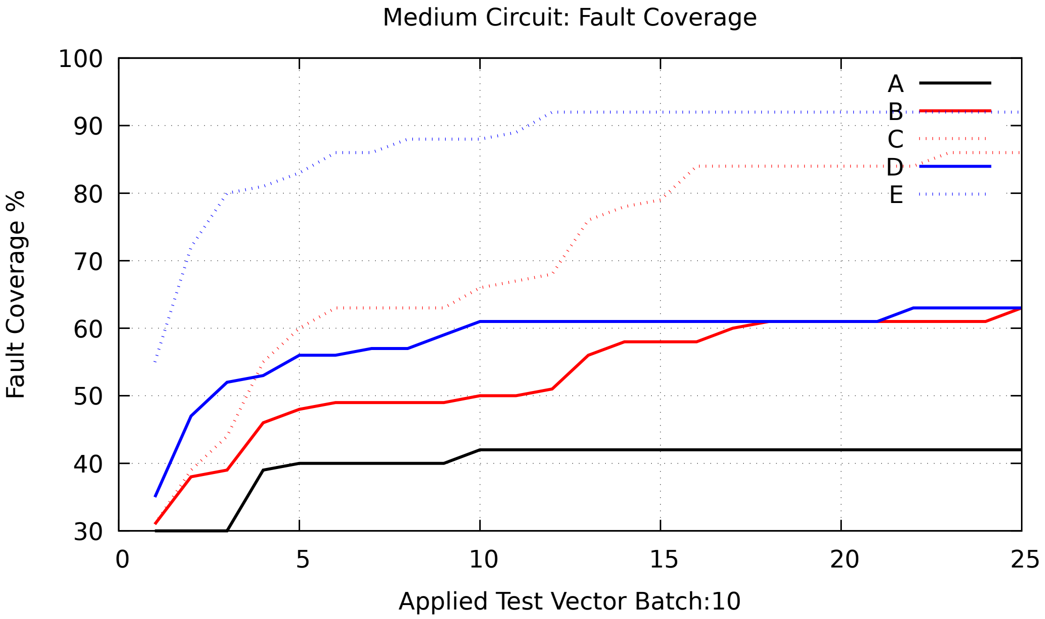

Figure 2. The multiple 8-bit LFSR with different seeds have the biggest coverage at 92% and is the most efficient out of all the other methods.

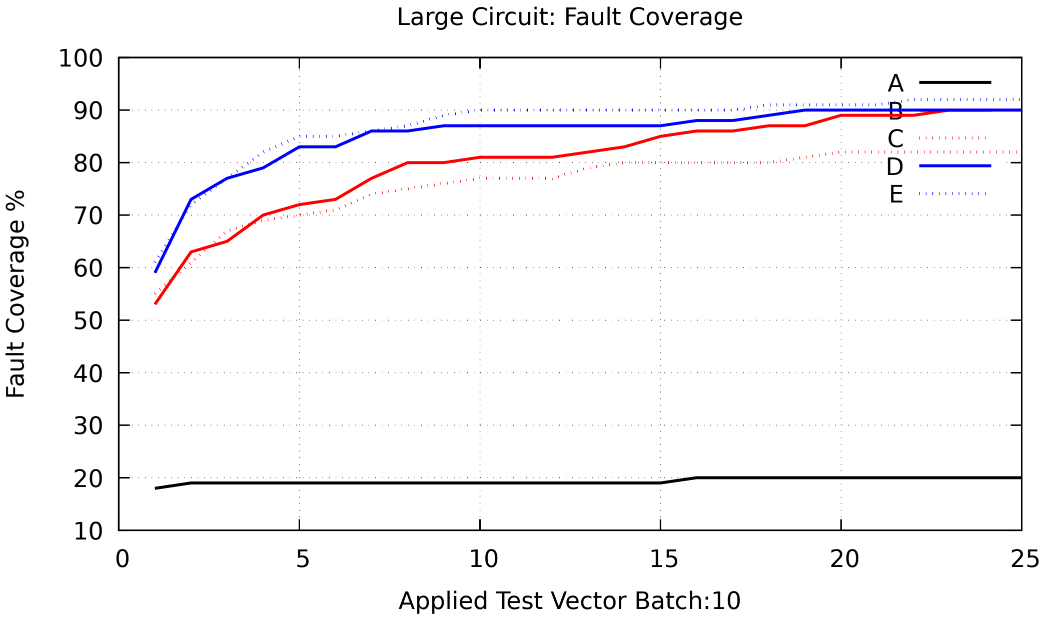

Figure 3. Set D and Set E are roughly identical, but based on the data, Set E seems to be the most effective and efficient at a slight amount.

4 DISCUSSION

To analyze which method works with the most efficiency and effectiveness, each method were tested using similar parameters. The initial seed for each circuit input is '1' and the batch size is '10'. Each method produces its own pseudo-randomly generated set of test vectors. Depending on the circuit input size, some test vector sets might be identical. This is true when the small circuit of 5-input was simulated. The resulting test vectors were similar for the all counter variations. This is also true for the LFSR variations with their own resulting set. Hence, the data table and the graph for this simulation are identical for Set A, B, and C, as well as for Set D and E. Regardless, the other circuits produced more useful data.

Efficiency is determined by how quickly a method can detect most or all of the full fault list. *Effectiveness* is determined by how well a method can detect the full fault list over time, given the constraints of only running 250 of the 256 generated test vectors. In the 5-Input Circuit, it seems that all were effective and efficient. Although Set D and Set E covered more fault coverage at the first test vector group, all methods had similar results starting at the second test vector group. In the 36-Input Circuit, Set E's greater efficiency and effectiveness can be seen. It had the highest effective coverage of 92% and efficiency with more than half already being covered in the first group whereas the others are still in the 30s. In the 60-Input Circuit, Set E was also determined as the most effective and efficient. Set B and Set D were also quite effective with their highest coverage of 90%. Set B's higher performance than Set C was surprising although welcomed. Based on the data, Set E is determined to be the best method to test large circuits with Set A as the most under performing method to use. These results provide a strong validation to our hypothesis. Not only did the data determined that the multiple 8-bit LFSR with different seeds was the most efficient, it also determined that it was the most effective with average coverage of 90% and above.

For an even more accurate data, more data collection through the use of even larger circuits can assist with this experiment. In addition, this experiment limited the counters and LFSRs to only using 8-bits. Simulating using more or less bits could provide useful analysis data.

CONCLUSIONS

Propagating pseudo-random test vectors through the use of *Multiple 8-Bit LFSR with Different Seeds* proved to be the most effective and efficient method to test the fault coverage of a circuit. This provides a potential solution that using this method may be the best way to detect faults. Even though this method only used three circuits for its simulation, the test results were consistent. The implementation focuses on coverage efficiency and effectiveness, making this experiment suitable for the analysis of larger systems and different variations of LFSRs.

ACKNOWLEDGEMENTS

The authors would like to thank Professor Wenjing Rao for her helpful lectures.

REFERENCES

1. Echo360 Platform | Academic Computing and Communications Center, University of Illinois at Chicago. *https://accc.uic.edu/services/teaching-learning/lecture-capture/echo360-platform/* (accessed November 17)