Partial Scan Selection Based on Dynamic Reachability and Observability Information

Michael S. Hsiao† Gurjeet S. Saund†† Elizabeth M. Rudnick††† Janak H. Patel†††
†Department of Electrical & Computer Engineering, Rutgers University, Piscataway, NJ
††Chromatic Research Inc., Sunnyvale, CA
†††Center for Reliable & High-Performance Computing, University of Illinois, Urbana, IL

Abstract

A partial scan selection strategy is proposed in which flip-flops are selected via newly proposed dynamic reachability and observability measures such that the remaining hard-to-detect faults are easily detected. This is done by taking advantage of the information available when a target fault is aborted by the test generator. A partial scan selection tool, IDROPS, has been developed which selects the best and smallest set of flip-flops to scan that will result in a high fault coverage. Results indicate that high fault coverages in hard-to-test circuits can be achieved using fewer scan flip-flops than in previous methods.

I Introduction

Partial scan design improves the effectiveness and speed of sequential circuit automatic test generation (ATG) greatly, especially for large circuits; however, this improvement comes with overhead in circuit area and speed from the insertion of scan flip-flops. How one selects scan flip-flops will determine both the overall fault coverage of the circuit-under-test and the amount of additional chip area. Thus, the problem of selecting which flip-flops to be placed in the scan chain becomes critical when one tries to reduce the area penalty while improving the fault coverage.

Two issues need to be addressed in regard to the effectiveness of partial scan selection tools. First, the selected scan flip-flops should be able to guide the specific ATG tool effectively. In other words, the scan flip-flops should assist the given ATG program when hard-to-excite and hard-to-propagate faults in the circuit are targeted. Second, since a less effective test generator may require extra scan flip-flops to achieve a comparable fault coverage, the selected ATG should be the best available.

Partial scan design directly enhances the testability of the circuit by improving the controllability and observability of the flip-flops in the scan chain, while indirectly improving testability of the flip-flops outside the scan chain. An alternative metric for flip-flop controllability at a higher level is state reachability and unreachability. A state that is easily reachable by a deterministic ATG may not be easily reached by a simulation-based ATG, and vice versa. Consequently, the circuits for which simulation-based ATG's produce high fault coverages may be troublesome for deterministic ATG's, and vice versa. To illustrate this phenomenon, we show test generation results in Table 1 for three different test generators for four of the ISCAS89 sequential benchmark circuits [1]: s526, s820, s1423, and s5378. HITEC [2][3] uses a deterministic, fault-oriented algorithm, while DIGATE [4] and STRATEGATE [5] are simulation-based, fault-oriented ATG's. Wide differences in fault coverages were observed for the three test generators. Differences in flip-flop switching activity were also observed. State reachability can be indirectly implied by the flip-flop switching activities (transition counts) of a circuit. Figure 1 illustrates the flip-flop switching patterns for s820 and s526. The switching patterns of flip-flops exhibited in the figure (displayed on a logarithmic scale) suggest that the set of states reached and frequency with which they are reached by the three ATG's differ. In circuit s820, for example, DIGATE test vectors generate more switching activity for the first flip-flop than both HITEC and

Table 1: Differences Among Various ATG's

Circuit	HITEC [2, 3]		DIGATE [4]		STRATEGATE [5]	
	Det	Vec	Det	Vec	Det	Vec
s526	365	2232	446	2864	454	2642
s820	813	984	621	465	814	590
s1423	776	177	1393	4044	1414	3943
s5378	3238	941	3447	10,500	3639	11,571

*This research was conducted at the University of Illinois and was supported in part by the Semiconductor Research Corporation under contract SRC 96-DP-109, in part by DARPA under contract DABT63-95-C-0069, and by Hewlett-Packard under an equipment grant.

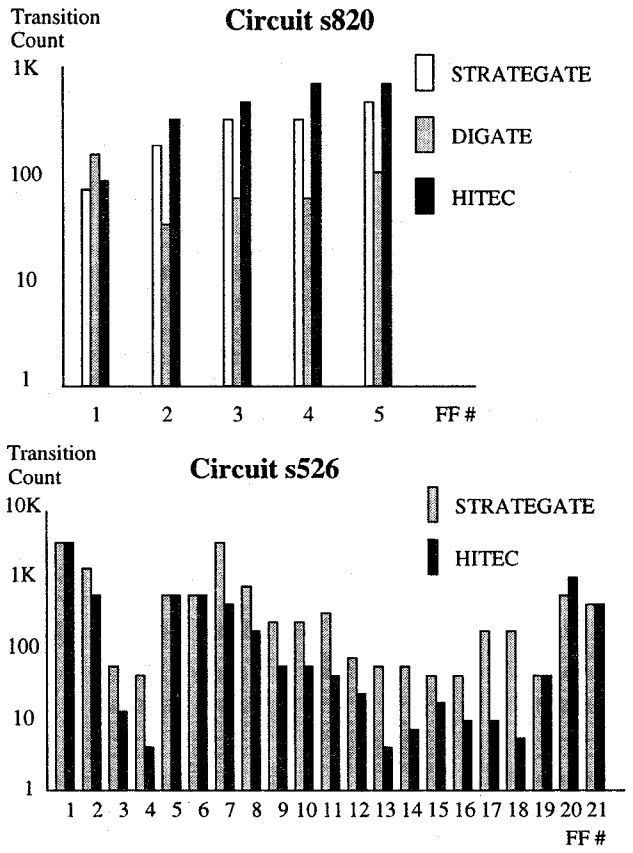

Figure 1: Flip-Flop Switching Patterns.

STRATEGATE test vectors, while significantly lower switching activities are observed for the remaining four flip-flops, and a much lower fault coverage results. In s526, STRATEGATE and HITEC test vectors produce similar switching activities for six of the twenty-one flip-flops: 1, 5, 6, 19, 20, and 21; however, large differences in activities exist for many of the remaining flip-flops. The test set lengths are similar for HITEC and STRATEGATE for s526; however, because significantly fewer states are visited by the HITEC test sequence, as evidenced by the lower switching activities on most flip-flops, a substantially lower fault coverage is obtained. Even when the difficulty of state justification is resolved in the simulation-based framework as in [5], the notion of state reachability is still quite different from that in the deterministic ATG. Recent work on partial scan selection for simulation-based ATG's has also suggested this phenomenon [6]. The flip-flop switching activity alone, however, does not provide sufficient guidelines for selecting the right flip-flops to scan, since many other factors contribute to the test generation difficulty.

Previous partial scan selection techniques can be divided into three categories: testability based, structure based, and ATG based. Low computational effort is required for testability-based approaches [7-10]; the drawback to testability-based approaches is that fault coverages may not be satisfactory for circuits with complex structures. Structure-based techniques [7, 11-17] aim to identify and remove feedback loops in the circuit, since feedback loops frequently are the cause of trouble for automatic test generators [11]. However, it has been shown in [7] that the ATG may not detect all faults even with all cycles (except self-loops) removed. Algorithms have been proposed to remove only a selected set of feedback loops necessary to improve ATG effectiveness by incorporating state information [17]. Finally, ATG-based techniques [6, 12, 17-19] utilize information provided by the ATG to target the undetected faults. OPUS-2 [12], BELLONA [19], and E-STG [20] target faults aborted by the ATG. OPUS-2 combines the test generation information with structural analysis to reduce computational cost. Justification and propagation requirements based on combinational circuit testing are used in BELLONA for the aborted faults to select the scan flip-flops. E-STG, on the other hand, attempts to reach a maximal set of aborted states given by the sequential ATG, in hopes of maximizing detection of aborted faults.

The motivation behind this work is to propose **better** guidelines for partial scan selection for simulation-based ATG's. We use a combination of testability and ATG-based techniques based on the following observation. Figure 2(a) shows the normalized SCOAP [21] controllability measures for each flip-flop in circuit s526; a higher controllability value for a flip-flop indicates more difficulty in controlling the corresponding flip-flop to a given value. Figure 2(b) illustrates the flip-flop histogram for all 200 aborted states provided by STRATEGATE [5]. An aborted state is defined as an unjustified state necessary for activation of a target fault. The measures of 0-count and 1-count are not equal to the total number of aborted states, since many values of flip-flops are *don't cares* (X) in many of the aborted states. A higher frequency of a flip-flop value in the graph indicates that the given flip-flop value is needed more frequently among all the aborted states. Notice that the patterns of the two graphs differ greatly. For example, flip-flop #14 is extremely hard to set to logic 1 according to the SCOAP measure in Figure 2(a); however, less than 10 out of the 200 aborted states require a logic 1 on this flip-flop (Figure 2(b)). Flip-flop #13 is quite easy to set to 0 based on SCOAP measures, but it is required in a large number of unjustified states. The weakness of the SCOAP measures is that they do not indicate if a particular flip-flop value is needed relatively often. Also, they do not take into

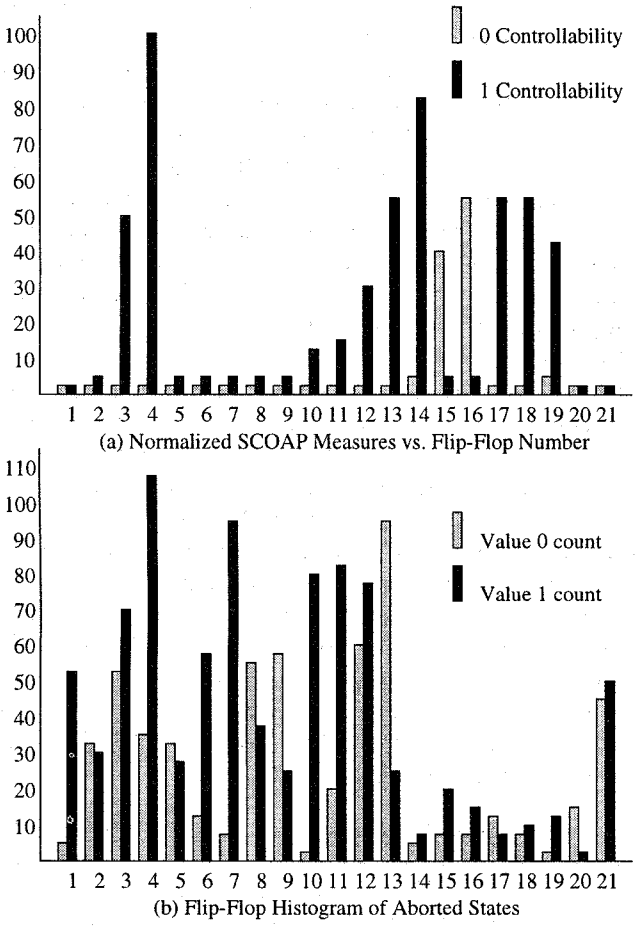

Figure 2: SCOAP vs. Aborted States.

account the controllability of a flip-flop in the context of other required flip-flop values when justifying a state. For this reason, we will use state related measures for controllability, namely, flip-flop transition counts and 0 and 1-value counts in the aborted states that could not be justified. For observability, we use the distinguishing sequence based dynamic measure introduced in [4]. These three measures form the basis of our partial scan selection criteria.

Our approach uses a genetic algorithm (GA) based test generator, STRATEGATE [5], since very high fault coverages have been reported for this test generator, better than any previously reported. Our algorithm finds the best flip-flops to scan using the hard-to-justify states, flip-flop switching patterns on the reached valid states, and flip-flop observability information. All the information is dynamically obtained; no state transition diagrams are needed. Flip-flops are selected such that the undetected faults which require the aborted (hard-to-justify) states for excitation or propagation can be detected. Consequently, each flip-flop added to the scan chain is used to target only the remaining undetected faults. In contrast, the approach proposed for simulation-based test generators in S-GATTO [6] uses only the flip-flop change counts and fault-effect propagation counts to guide the selection of scan flip-flops. No aborted state information is included; thus, it does not specifically target the remaining undetected faults. Many more scan flip-flops are required by S-GATTO than for other techniques for the benchmark circuits studied, indicating that using only flip-flop switching activity and propagation information is inadequate. Our approach also differs from the one used in [20], where state transition modeling is used to select the smallest number of scan flip-flops such that the number of hard-to-justify (aborted) states becoming justifiable is maximized. For large circuits with large numbers of flip-flops, there exist many equal-sized sets of flip-flops that reach the same number of aborted states. The choice of which set of flip-flops to scan in [20] is based on a profit function, and the choice becomes much more difficult when large state spaces are involved, especially when equal-sized sets of flip-flops are present in the circuit. Moreover, the switching patterns of the flip-flops are not utilized.

A partial scan selection tool, IDROPS (Iterative and Dynamic Reachability/Observability Partial-scan Selection), has been developed which selects the best and smallest set of flip-flops to scan that will result in a high fault coverage. The issues regarding our tool, IDROPS, are discussed in the remainder of this paper. Section II provides some definitions and guidelines used for dynamic testability. The selection algorithm is explained in Section III. Experimental results are discussed in Section IV, followed by conclusions in Section V.

II Dynamic Testability Metrics

Multiple metrics provided by the automatic test generator are used as guidelines for selecting scan flip-flops. These metrics provide criteria for selecting flip-flops from different perspectives; an intelligent combination of the metrics is crucial in producing an effective partial scan flip-flop selection algorithm.

Our goal is to improve the detectability of the faults aborted during test generation; these aborted faults are those that the ATG gave up on. The information available at the time the faults were aborted is of great value in making partial scan flip-flop selection decisions. Specifically, an unjustifiable state that could lead to activation of a particular fault f_i provides important information on how fault f_i can be activated. To illustrate this idea, let us consider a circuit with four flip-flops, where the aborted states for three undetected faults are **X1X0**, **01X1**, and **0X01**. The patterns of the aborted states may be a useful metric, and in this case, we may conclude that it would be useful to scan

the fourth flip-flop, since its value is specified in every aborted state.

Definition 1: An *abort index* A_i^v for flip-flop i of the circuit is the normalized frequency of value v for the corresponding flip-flop among all the aborted states. The normalization is with respect to the maximum absolute frequency.

For example, in the three aborted states listed above, A_2^1 is 0.67 (2/3) because the second flip-flop has a value of 1 occurring two times among the aborted states.

Aborted states are identified by the chosen ATG, STRATEGATE [5]. Briefly, STRATEGATE selects a target fault from the fault list, and an attempt is made to derive a sequence that excites the fault and propagates the fault effects to a primary output (PO) or to the flip-flops. If fault activation is difficult, single-time-frame fault activation is performed using an activation vector composed of primary input and flip-flop values for the single time frame. If the single-time-frame fault activation is successful, a state relaxation step is performed to relax any flip-flop values that the fault activation does not depend on; state relaxation attempts to reduce the complexity of state justification which immediately follows. Once the state is justified and the fault is activated, the fault effects are propagated from the flip-flops to the PO's in the second phase with the assistance of distinguishing sequences. A state is declared aborted when the state necessary to excite the target fault or propagate its effects to at least one flip-flop fails to be justified. Because of the state relaxation step, the aborted states are not completely specified (i.e., some flip-flops have don't-care values(X)). A weight is given to each aborted state to account for repeated aborted states.

Abort indices for the flip-flops are insufficient for an effective partial scan selection because much correlation exists among flip-flops. In other words, the frequent occurrence of flip-flop i with value v in the aborted states may be caused by the uncontrollability of another flip-flop j to a value u. Therefore, in addition to abort indices, other dynamic testability metrics are used.

Traditional measures of flip-flop controllability and observability have been used previously as guides for selecting scan flip-flops. These ideas are incorporated into our scheme as well, except that instead of static controllability and observability measures, dynamically-generated metrics are used. As explained in the introduction, an easily controllable flip-flop in a deterministic ATG framework may not be easily controllable in a simulation-based ATG environment. For this reason, a dynamic technique which *profiles* the switching behavior of the flip-flops is used to provide controllability measures of individual flip-flops. By a similar token, dynamic observability information for the flip-flops is collected during test generation [4].

Definition 2: *Switching index* S_i for flip-flop i indicates the frequency of switching for the corresponding flip-flop, based on the test vectors generated so far.

Definition 3: *Propagation index* P_i for flip-flop i is the normalized count of fault-effects failing to propagate from flip-flop i to a PO, based on the test vectors generated so far. The normalization is with respect to the maximum propagation failure count.

Both switching and propagation indices S_i and P_i are dynamically computed and can change during test generation. A high switching index for a flip-flop suggests that the given flip-flop toggles frequently and is more likely to be controllable to a specific logic value. Flip-flops with low switching indices, on the other hand, are less likely to switch to the opposite of their current logic values. Similarly, propagation indices provide guidelines about which flip-flops are less *observable* at the PO's; a higher P_i indicates more difficulty in propagating a fault-effect from the i^{th} flip-flop to a PO.

III Partial Scan Selection Algorithm

We now derive an algorithm that uses the dynamic testability metrics and cleverly selects the best flip-flops to scan. Scan flip-flops are chosen such that the detectability of the aborted faults increases most significantly. All testability measures are provided by the ATG; therefore, a quick run of the test generator is needed.

Because the goal is to detect the aborted faults, a new metric, aborted-fault detectability potential (ADP), is introduced that combines the three dynamic testability measures. The ADP measure for a given flip-flop indicates the potential increase in detectability of the aborted faults if the corresponding flip-flop is scanned. In our work, we define ADP for flip-flop i as

$$ADP(i) = \frac{0.7\ max(A_i^0, A_i^1) + 0.3\ (P_i)}{log_2(S_i)}\ .$$

The abort indices A_i and propagation indices P_i play important roles in determining the ADP since the patterns of aborted states and fault-effect propagation failures determine the best flip-flops for scan. More weight is given to the abort indices because state justification for the aborted states contributes to most of the aborted faults. This observation was also made in [6][19][20]. The weights are purely empirical. The switching index S_i is used to normalize the contributions from abort and propagation indices with respect to the flip-flop's dynamic controllability. However, normalization needs to be done carefully for the following reason. The switching indices may differ greatly

among flip-flops in the circuit. Some flip-flops may have switching indices in the thousands while others are only in the single digits. A low S_i value suggests that the corresponding flip-flop seldom switches and is thus difficult to control to the desired value; however, if few aborted states contain a specified value in the i^{th} flip-flop position with low S_i, we may not want to place flip-flop i in the scan chain. Moreover, a flip-flop with a switching index of 1000 may not necessarily be ten times more controllable than flip-flop **b** with an index of 100. The difference in the switching indices may simply be because most faults do not rely upon a specific value for flip-flop **b**. For this reason, a logarithmic operation is performed on the switching index.

The IDROPS framework is shown in Figure 3. Initially, the STRATEGATE test generator is invoked

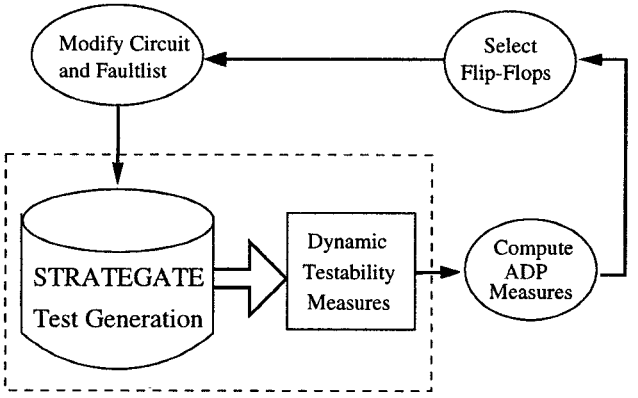

Figure 3: IDROPS Partial Scan Selection Framework.

on the original circuit without any scan flip-flops. The goal is to obtain the dynamic testability measures quickly, since ATG on a circuit without any scan is expensive. Switching, propagation, and abort indices for each flip-flop are dynamically computed as the ATG attempts to generate vector sequences for the targeted faults. The test generation process is halted when all faults have been targeted or a fixed number of aborted states have been collected. Next, ADP measures are computed for all flip-flops in the circuit from the dynamic testability measures, and a set of flip-flops is selected for scan based on the ADP measures. The circuit is modified and another iteration of test generation is performed. This time, STRATEGATE is applied to the aborted faults only (mapped to the modified circuit). The iterative process is repeated until a user-specified percentage of flip-flops have been selected for scan or until a satisfactory fault coverage is reached.

An example of the IDROPS flip-flop selection process is given for ISCAS89 sequential circuit s526, which has 21 flip-flops. The fault coverage obtained without any scan flip-flops is 79.5%. After selecting the best scan flip-flop (#4 with highest ADP), the ADP measures change drastically, from a range of 10–50 to a range of 1–8. These changes in ADP not only reflect the structural changes made in the circuit, but also the changes in detectability potential of the flip-flops for the remaining aborted faults after scanning one flip-flop. Consequently, the fault coverage increases to 86.5%. The relative magnitudes of the flip-flop ADP's do not change significantly after the second flip-flop is selected (#13), indicating that the addition of this flip-flop to the scan chain does not change the detectability of the aborted faults much, and more scan flip-flops are needed to detect them. Indeed, the fault coverage improves only 0.2% to 86.7%. When the next flip-flop is selected (#7), the ADP's for flip-flops #3 and #12 increase significantly, while the ADP for flip-flop #6 drops to half of its previous value, suggesting that the circuit structure has been changed significantly. Flip-flops #3 and #12 now play more important roles in detecting the remaining faults. Although the fault coverage now increases only slightly to 87.2%, the increase in the ADP's for flip-flops #3 and #12 provides a potential for detecting the remaining faults with an additional scan flip-flop. When the fourth and final flip-flop is scanned (#10), the ADP's for most flip-flops do in fact decrease significantly, and the fault coverage with four scan flip-flops reaches 94.2%.

For larger circuits, the ADP measures for the flip-flops change drastically after scanning a few flip-flops, resulting in detection of a large portion of the aborted faults, as will be shown in the next section. For large circuits, selecting one scan flip-flop at a time is very expensive, and the cost can be reduced by adding several flip-flops during each iteration. For this reason, five iterations are used in IDROPS, where the number of scan flip-flops selected is equally divided among iterations. In the case of a 20% user-specified percentage of flip-flops scanned, each iteration will select the 4% of the flip-flops that have the highest ADP's. It should be noted that selecting a smaller number of flip-flops during each iteration may result in a smaller set of final scan flip-flops at a higher computation cost.

IV Experimental Results

IDROPS was developed using the C++ language. Several ISCAS89 sequential benchmark circuits [1] were used for evaluating the effectiveness of IDROPS in selecting scan flip-flops. Circuits for which STRATEGATE achieves high fault coverages (i.e., few faults are aborted) are not considered here, since the improvements from partial scan would be insignificant, and the comparisons would be difficult to make.

Table 2 shows the results of IDROPS compared with other partial scan algorithms. For each partial scan

Table 2: IDROPS Results

Circuit (# FF)	S-GATTO [6]*			E-STG [20]			Opscan [17]			IDROPS		
	Scan	FC	Time	Scan	FC	Time	Scan	FC	Time	Scan	FC	Time
s420	0	47.4	55 s	0	6.2	1.7 h				0	5.3	551 s
(16)				2	9.7	1.7 h				3	20.9	1587 s
	3	59.3	48 s	3	21.1	1.3 h	-	-	-	4	22.9	1454 s
s444	0	92.4	219 s				0	87.3	2.0 h	0	86.9	84.1 s
(21)	3	93.7	152 s							3	93.2	67.9 s
	9	95.6	135 s	-	-	-	5	94.9	20.0 s	4	94.7	54.5 s
s526	0	83.2	214 s	0	9.2	3.0 h	0	9.2	3.0 h	0	79.5	311 s
(21)	3	91.7	581 s	2	58.7	1.3 h				3	87.2	225 s
				4	93.2	996 s	7	98.7	34.1 s	4	94.2	242 s
s1423	0	96.7	362 s	0	36.6	5.7 h				0	90.8	1542 s
(74)	11	98.0	348 s	7	62.2	3.9 h				9	95.3	685 s
	21	98.4	450 s	15	74.4	2.4 h	-	-	-	15	95.8	646 s
s5378	0	77.3	420 s	0	68.5	7.5 h	0	68.5	7.5 h	0	76.4	6.9 h
(179)	27	80.4	456 s	18	94.6	2858 s				21	94.7	2.4 h
	90	93.3	516 s	36	96.2	1476 s	80	97.5	134 s	34	98.4	1.4 h
s9234	0	5.9	19 s							0	12.0	2.2 h
(211)	36	37.3	1.1 h							27	79.9	4.2 h
	57	46.0	1.1 h	-	-	-	-	-	-	43	93.6	5.3 h
s13207	0	20.5	280 s							0	9.6	6.9 h
(638)	100	37.3	2217 s							78	76.5	5.7 h
				-	-	-	-	-	-	128	91.2	6.1 h
s15850	0	5.6	109 s							0	39.8	9.7 h
(534)	89	29.6	1176 s							66	65.2	5.6 h
	132	38.1	1321 s	-	-	-	-	-	-	108	89.6	12.2 h
s38417										0	3.6	2.8 h
(1636)	406	57.3	1.2 h	-	-	-	-	-	-	328	88.4	14.2 h †
s38584										0	72.2	22.1 h
(1426)										58	76.8	17.1 h
	-	-	-	-	-	-	-	-	-	290	91.6	12.4 h †

*: S-GATTO assumes a reset state for all circuits

†: flip-flop selection in only one iteration of 20% of total flip-flops

method, the number of scan flip-flops (**Scan**), fault coverage (**FC**), and execution time required for flip-flop selection are given for each circuit. The execution time for IDROPS includes the time for test generation used in evaluating the dynamic testability measures. Up to 20% of the flip-flops are scanned for all circuits in IDROPS. The algorithm proposed in S-GATTO [6] is an ATG-based partial scan selection method based on the simulation-based test generator GATTO [22], and it assumes that a reset state exists for all circuits; the other three methods do not make such assumptions. E-STG [20] is also an ATG-based partial selection algorithm using aborted state information from HITEC [2][3]. Opscan [17] combines cycle-cutting and testability for selecting scan flip-flops. Results for IDROPS are reported for the original circuit without any scan and also after inserting 12% and 20% scan.

Note that in the smaller circuits, IDROPS achieves higher fault coverages with fewer scan flip-flops when compared with E-STG and Opscan. Since we do not assume a reset state, the fault coverages obtained for some small circuits are not as high as those for S-GATTO [6]. This is especially true for circuits s420 and s1423, where many flip-flops are uninitializable when a fault is present. In s526, IDROPS obtains a higher fault coverage with the same number of flip-flops when compared with E-STG. The fault coverage is slightly lower than that achieved by S-GATTO or Opscan; however, Opscan used more scan flip-flops. The execution times for the smaller circuits are small as well; they are comparable to the execution times of S-GATTO and Opscan, and smaller when compared with E-STG.

Differences between IDROPS and the previous approaches are much more significant for the larger circuits. In s5378, IDROPS achieves the highest fault coverage with the smallest number of scan flip-flops; Opscan and S-GATTO need more than twice as many scan flip-flops, and the resulting fault coverages are still lower. E-STG obtains 2.2% lower fault coverage (more than 100 fewer detected faults) when compared to IDROPS. E-STG and Opscan do not report results for the remaining five large circuits because of the huge

numbers of flip-flops and possible states in the circuits. Results for four of these circuits were reported for S-GATTO. The hard-to-test circuit s9234, for which only 12% fault coverage is obtained when no scan flip-flops are present, becomes much easier to test with 27 scan flip-flops; the fault coverage with 27 flip-flops already surpasses the fault coverage obtained by S-GATTO with 57 scan flip-flops. The final fault coverage for this circuit is 93.6% with 43 scan flip-flops, over twice the coverage of S-GATTO. In circuit s13207, S-GATTO achieves higher fault coverage initially without any scan due to the assumption of a reset state; however, with 78 scan flip-flops, IDROPS obtains a much higher fault coverage than S-GATTO achieves with 100 scan flip-flops. The final fault coverage with only 20% (128) scan flip-flops is very high. Similar trends are seen in other large hard-to-test circuits.

Long execution times are needed for the large circuits initially because many of the faults are either undetectable or hard to test, and the corresponding fault coverages are very low. After adding scan flip-flops, the execution times are reduced; however, due to the cost of simulation for large circuits, the execution times are still not very short. All the times reported for IDROPS include the test generation time. Despite the extra execution times needed, IDROPS is able to obtain high fault coverages for the hard-to-test *large* circuits, which could not be efficiently handled by previous techniques.

V Conclusions

A new method of partial scan flip-flop selection has been proposed. Dynamically generated state reachability and flip-flop observability information are used for selection. The dynamic information includes three testability metrics that are provided by a simulation-based automatic test generator. The partial scan selection tool, IDROPS, proposed and developed in this work, targets simulation-based ATG's specifically. Large circuits can be handled with IDROPS as well, where the large state spaces do not hinder the quality of partial scan selection. IDROPS is effective in selecting a small set of flip-flops necessary to produce high fault coverages.

References

[1] F. Brglez, D. Bryan, and K. Kozminski, "Combinational profiles of sequential benchmark circuits," *Int. Symp. Circuits and Systems*, pp. 1929–1934, 1989.

[2] T. M. Niermann and J. H. Patel, "HITEC: A test generation package for sequential circuits," *Proc. European Conf. Design Automation*, pp. 214–218, 1991.

[3] T. M. Niermann and J. H. Patel, "Method for automatically generating test vectors for digital integrated circuits," *U.S. Patent No.* 5,377,197, December, 1994.

[4] M. S. Hsiao, E. M. Rudnick, and J. H. Patel, "Automatic test generation using genetically-engineered distinguishing sequences," *Proc. VLSI Test Symp.*, pp. 216–223, 1996.

[5] M. S. Hsiao, E. M. Rudnick, and J. H. Patel, "Sequential circuit test generation using dynamic state traversal," *Proc. European Conf. Design Automation*, pp. 22–28, 1997.

[6] F. Corno, P. Prinetto, M. Rebaudengo, and M. Sonza Reorda, "Partial scan flip flop selection for simulation-based sequential ATPG's," *Proc. Int. Test Conf.*, pp. 558–564, 1996.

[7] V. Chickermane and J. H. Patel, "An optimization based approach to the partial scan design problem," *Proc. Int. Test Conf.*, pp. 377–386, 1990.

[8] K. S. Kim and C. R. Kime, "Partial scan flip-flop selection by use of empirical testability," *Journal of Electronic Testing*, vol. 7, pp. 47–60, Aug./Oct. 1995.

[9] P. S. Parikh and M. Abramovici, "A cost based approach to partial scan design," *Proc. Design Automation Conf.*, pp. 255–259, 1993.

[10] P. S. Parikh and M. Abramovici, "Testability-based partial scan analysis," *Journal of Electronic Testing*, vol. 7, pp. 61–70, Aug./Oct. 1995.

[11] K.-T. Cheng and V. D. Agrawal, "A partial scan method for sequential circuits with feedback," *IEEE Trans. Computers*, vol. 39, pp. 544–548, 1990.

[12] V. Chickermane and J. H. Patel, "A fault oriented partial scan design approach," *Proc. Int. Conf. Computer-Aided Design*, pp. 400–403, 1991.

[13] S. E. Tai and D. Bhattacharya, "A three-stage partial scan design method to ease ATPG," *Journal of Electronic Testing*, vol. 7, pp. 95–104, Aug./Oct. 1995.

[14] S. T. Chakradhar, A. Balakrishman, and V. D. Agrawal, "An exact algorithm for selecting partial scan flip-flops," *Proc. Design Automation Conf.*, pp. 81–86, 1994.

[15] P. Ashar and S. Malik, "Implicit computation of minimum-cost feedback-vertex sets for partial scan and other applications," *Proc. Design Automation Conf.*, 77–80, 1994.

[16] T. Orensten, Z. Kohavi, and I. Pomeranz, "An optimal algorithm for cycle breaking in directed graphs," *Journal of Electronic Testing*, vol. 7, pp. 71–82, Aug./Oct. 1995.

[17] D. Xiang and J. H. Patel, "A global algorithm for the partial scan design problem using circuit state information," *Proc. Int. Test Conf.*, pp. 548–557, 1996.

[18] H. K. T. Ma, S. Devadas, A. R. Newton, and A. Sangiovanni-Vincentelli, "An incomplete scan design approach to test generation for sequential machines," *Proc. Int. Test Conf.*, pp. 730–734, 1988.

[19] I. Park, D. S. Ha, and G. Sim, "A new method for partial scan design based on propagation and justification requirements of faults," *Proc. Int. Test Conf.*, pp. 413–422, 1995.

[20] V. Boppana and W. K. Fuchs, "Partial scan design based on state transition modeling," *Proc. Int. Test Conf.*, pp. 538–547, 1996.

[21] L. M. Goldstein and E. L. Thegen, "SCOAP: Sandia controllability/observability analysis program," *Proc. Design Automation Conf.*, pp. 190–196, 1980.

[22] P. Prinetto, M. Rebaudengo, and M. Sonza Reorda, "An automatic test pattern generator for large sequential circuits based on genetic algorithms," *Proc. Int. Test Conf.*, pp. 240–249, 1994.