Robust Reconfigurable Scan Networks

Lylina, Natalia; Wang, Chih-Hao; Wunderlich, Hans-Joachim

Proceedings of the Conference on Design, Automation and Test in Europe (DATE'22);
Antwerp, Belgium; March 2022

doi: https://doi.org/10.23919/DATE54114.2022.9774770

Abstract: Reconfigurable Scan Networks (RSNs) access the evaluation results from embedded instruments and control their operation throughout the device lifetime. At the same time, a single fault in an RSN may dramatically reduce the accessibility of the instruments. During post-silicon validation, it may prevent extracting the complete data from a device. During online operation, the inaccessibility of runtime-critical instruments via a defect RSN may eventually result in a system failure. This paper addresses both scenarios above by presenting robust RSNs. We show that by making a small number of carefully selected spots in RSNs more robust, the entire access mechanism becomes significantly more reliable. A flexible cost function assesses the importance of specific control primitives for the overall accessibility of the instruments. Following the cost function, a minimized number of spots is hardened against permanent faults. All the critical instruments as well as most of the remaining instruments remain accessible through the resulting RSNs even in the presence of defects. In contrast to state-of-the-art fault-tolerant RSNs, the presented approach does not change the RSN topology and needs less hardware overhead. Selective hardening is formulated as a multi-objective optimization problem and solved by using an evolutionary algorithm. The experimental results validate the efficiency and the scalability of the approach.

Robust Reconfigurable Scan Networks

Natalia Lylina, Chih-Hao Wang, Hans-Joachim Wunderlich

ITI, University of Stuttgart, Pfaffenwaldring 47, D-70569 Stuttgart, Germany
{lylina, wangco, wu}@informatik.uni-stuttgart.de

***Abstract*—Reconfigurable Scan Networks (RSNs) access the evaluation results from embedded instruments and control their operation throughout the device lifetime. At the same time, a single fault in an RSN may dramatically reduce the accessibility of the instruments. During post-silicon validation, it may prevent extracting the complete data from a device. During online operation, the inaccessibility of runtime-critical instruments via a defect RSN may eventually result in a system failure.**

This paper addresses both scenarios above by presenting robust RSNs. We show that by making a small number of carefully selected spots in RSNs more robust, the entire access mechanism becomes significantly more reliable. A flexible cost function assesses the importance of specific control primitives for the overall accessibility of the instruments. Following the cost function, a minimized number of spots is hardened against permanent faults. All the critical instruments as well as most of the remaining instruments are accessible through the resulting RSNs even in the presence of defects. In contrast to state-of-the-art fault-tolerant RSNs, the presented scheme does not change the RSN topology and needs less hardware overhead. Selective hardening is formulated as a multi-objective optimization problem and solved by using an evolutionary algorithm. The experimental results validate the efficiency and the scalability of the approach.

***Keywords*-Reconfigurable Scan Networks, selective hardening, multi-objective optimization, synthesis**

I. Introduction

Reconfigurable Scan Networks (RSNs), as standardized by IEEE Std. 1687 [1] and IEEE Std. 1149.1 [2] and shown in Fig. 1, efficiently access instruments, which are used to support dependable operation and diagnosis of devices, through a number of scan segments. Control primitives configure a scan path through the segments, and thereby determine the currently accessed instruments. The *scan multiplexers* select certain branches depending on the control signal value.

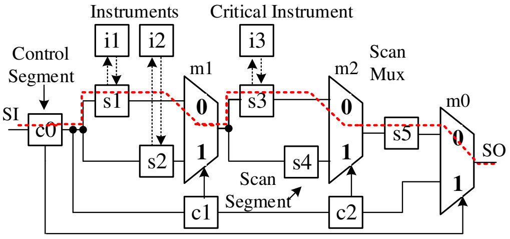

Fig. 1. Considered system

Hardening the most critical instruments and the corresponding scan segments against permanent faults is not enough to ensure robust access to the instruments since a single fault in RSN the control logic may corrupt scan paths and make certain instruments inaccessible. This affects two major tasks:

- *Post-silicon validation:* A fault in an RSN may prevent accessing a major part of instruments, such that only incomplete data can be extracted.
- *Runtime operation:* The device operation may be guided by runtime-adaptive instruments, e.g., Adaptive Voltage and Frequency Scaling (AVFS) or error rate adaption. Inaccessibility of such critical instruments due to a single fault in the RSN may cause a system failure.

The impairment of the system operation due to inaccessibility of an instrument via an RSN, can be used as a weight of the instrument. A cost function defining the importance of an RSN primitive can be computed by summing up the weights of those instruments, which are inaccessible if the primitive is defect. Minimizing the cost function value for all the RSN primitives keeps the most critical instruments and a major part of the remaining instruments accessible, thereby the scenarios above are addressed. To minimize the cost function, faults in RSNs can be tolerated, or avoided as proposed in this paper.

Using conventional approaches, such as Triple Modular Redundancy (TMR) [3], for the entire RSN requires high hardware costs. In [4], single faults are tolerated by augmenting the initial RSN with minimized additional connectivities. This approach requires diagnostic support [5], complicates routing for test [6–9], and also access in the presence of a fault, and does not consider the criticality of the components.

The probability of defects can be reduced by hardening some components and cells locally. The existing approaches in the field of design-for-manufacturability [10] can reach as far as applying TMR locally to single cells [11]. They decrease the probability of defects and do not affect test and diagnostic procedures but increase power consumption and area overhead. Hardening a few carefully selected components, while leaving the remaining device unprotected, still reduces the probability of a system failure and has acceptable costs [12]. Hardening is usually applied for soft error mitigation [13], but these are efficiently tolerated in RSNs by repeating a failing pattern, and are therefore, out of the scope of this paper.

This paper presents the first solution to synthesize cost-efficient robust RSNs. Even in the presence of defects, a robust RSN not only enables an efficient *post-silicon validation* with reliable access to the major part of the instruments but also provides *runtime* access to the most critical instruments. The presented selective hardening scheme is supported by an exact analysis, which assesses the criticality of a fault in any RSN primitive, as shown in Section IV. The criticality is calculated as a weighted sum of the instruments, which become inaccessible due to a fault in a given primitive. For each instrument, the accessibility for observation and control are considered separately to reflect the different requirements of a specific instrument. Based on the criticality analysis, a minimized number of RSN primitives is selected and hardened to reduce the damage caused by defects, as described in Section V. A trade-off between reducing the hardware costs of hardening and minimizing the remaining damage of defects is investigated by using an evolutionary algorithm [14, 15] for generating close-to pareto-optimal solutions.

II. Method Overview

The presented scheme has the following major goals:

- *Precise Criticality Analysis*: The criticality of scan primitives should be carefully assessed, and the most critical primitives in the RSN for the correct system operation should be identified, as shown in Section IV.
- *Cost-effective Selective Hardening*: The scheme should dramatically decrease the damage of defects in RSNs for the system operation, while minimizing the hardware costs, as shown in Section V. The trade-off between the criteria above is investigated. The scheme is independent of the actual hardening technique to be used, e.g. in [10].
- *Access Patterns Compatibility*: The resulting RSNs must follow the initial RSN topology. They should not only be compatible with the existing access, test and diagnosis procedures [6–8, 16, 17], but also be able to use the same access patterns as the initial unhardened RSN.

Example: *In Fig.1, $m1$ is a regular scan multiplexer, and $m2$ is hardened. The multiplexer $m1$ might propagate the data from wrong scan-input, while $m2$ remains functionally correct, and faults due to a defect in $m2$ are avoided.*

III. Modeling

An RSN is modeled as a directed graph $G := (V, E)$ with vertices V and edges E, as shown in Fig. 2 for the RSN from Fig. 1. Each vertex models a scan primitive (segment or multiplexer), a fan-out, or represents a primary scan-input or -output. Each edge represents a direct connectivity between the vertices. Segment Insertion Bits (SIB) insert a sub-RSN into the path or bypass in based on the control signal value. It is modeled as a combination of a scan segment and a multiplexer.

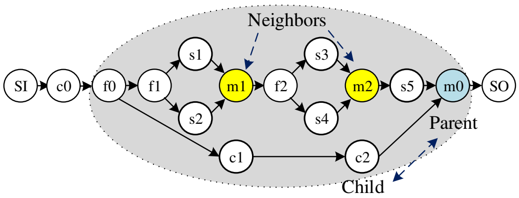

Fig. 2. RSN graph: the stem region of $f0$ is shown with grey color

If at least two disjoint paths exist between the source s and the destination d, then s is a *reconvergent fan-out stem*, and the d is its *reconvergence gate* [18]. In RSNs, only multiplexers are reconvergence gates. A reconvergence gate, which does not reach any other reconvergence gate of a fan-out stem, is called a *closing reconvergence*. A *stem region* includes all such primitives reachable from a given fan-out stem, such that its closing reconvergence is reachable from these primitives.

So-called hierarchical series-parallel RSNs have first been introduced in [19] to allow scalable processing of RSNs.

Definition 1: A *Series-Parallel (SP)* RSN graph is an RSN graph, which consists of two vertices connected via a single edge, or a composition of two SP-RSNs G_1 and G_2:

- *Parallel*: The source of G_1 is identified with the source of G_2; the sink of G_1 is identified with the sink of G_2.
- *Series* : The sink of G_1 is identified with the G_2 source.

Although most RSNs can be directly represented as hierarchical SP graphs, additional steps might be required to obtain a hierarchical representation. Then an SP-RSN model is obtained by adding a minimized number of virtual vertices into the initial graph. This representation is only used for the analysis, and does not require any physical changes, since in the resulting hardened RSN the applied changes are reverted [19]. A binary decomposition tree is built for an SP-RSN model, shown in Fig. 3. The blue ”S” vertices stay for serial connections, the green ”P” vertices – for parallel ones.

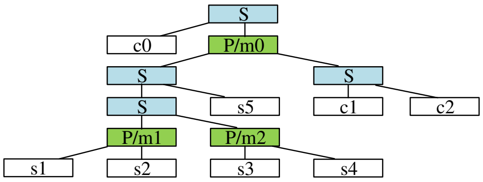

Fig. 3. Annotated binary decomposition tree for Fig. 1. Since all the paths through the segment $c2$ traverse the multiplexer $m0$, then $m0$ *dominates* $c2$. The segment $c2$ belongs to such stem region, where the $m0$ is a closing reconvergence. Then, $m0$ is referred as a *parent* of $c2$, and $c2$ is referred as a *1-child* of $m0$. The multiplexer $m2$ dominates $m1$, but is not its parent, so the multiplexers are *neighbors*.

IV. Criticality Analysis

A. Explicit Criticality Specification

An explicit criticality specification reflects the damage caused by the inaccessibility of specific instruments. It is implemented as a list of instruments, where each instrument i is associated with a pair of non-negative *damage weights*. The first one do_i defines the *damage* of losing the *observability*, while the second one ds_i – the *damage* of losing the *settability* of the instrument. The exact values of the *damage weights* are specified by a system designer, e.g.:

- *Sensors*: A relatively low value of do_i can be assigned to the damage of unobservability of one of many interchangeably used sensors. If multiple sensors are inaccessible, the damage is more severe and is calculated as a sum of the sensors' weights. If the settability of sensors is not required then ds_i is set to zero or close-to-zero.
- *Runtime-adaptive instruments*: The settability of a runtime-critical instrument is important for a correct system operation, and the corresponding damage weight ds_i is set to a relatively high value. At the same time, the damage due to its unobservability do_i is relatively low.

The damage weights of the instruments are annotated at the corresponding segments in the binary decomposition tree. To ensure the accessibility of the most critical instruments, the damage weight of any of those instruments, whose unobservability may lead to a system failure, should be at least as high as the sum of the damage weights of all other ”uncritical” instruments. The same applies for the settability weights.

B. Fault Effects

A single fault in an RSN might affect the intended connectivity properties of the instruments and might make the RSN disconnected. Next, the influence of specific faults in terms of graph connectivity is discussed.

1) Scan Segments: A fault f in a scan segment may break the integrity of all the scan paths, which traverse this segment. The existence of a faulty segment is modeled by removing the corresponding vertex from the RSN graph. The fault effect is isolated inside the branch controlled by the closest parental scan multiplexer of the given segment. The parental multiplexer is identified by traversing the binary decomposition tree in a reversed polish order starting from the affected segment. In the isolated branch, the segments located closer to the scan-output than the affected segment, are inaccessible for setting the value from the scan-in. In a decomposition tree, it is equivalent to removing the connectivity to the affected vertex and the segments on its right-hand side. This modified tree is further referred as a *settability tree* under a *fault f*. The same idea is applied to build a *observability tree* under a *fault f*. The segments located closer to the scan-in become unobservable and are disconnected in the observability tree.

2) Scan Multiplexer and SIB: In the event of a "stuck-at-id" fault, a multiplexer will permanently select only one input, independent of the value driving its address control port. So, its opposite branch becomes inaccessible through this multiplexer. For all the primitives in this branch, a path cannot be sensitized from a scan-in port and to a scan-out port. To model this fault, the connectivity from a vertex, which corresponds to a faulty multiplexer, to the inaccessible branch is removed from the binary decomposition tree as shown in Fig. 4.

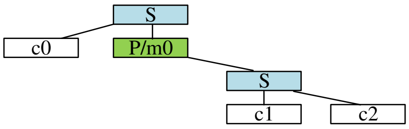

Fig. 4. Due to a "stuck-at-1" fault of the multiplexer $m0$ from Fig. 1 the instruments $i1$, $i2$ and $i3$ become inaccessible.

A SIB affected by a "stuck-at-asserted" fault always provides access to the sub-RSN. If the SIB is "stuck-at-deasserted", access to the sub-RSN is never granted. Fault effects in SIBs are considered as a combination of those for a scan segment and a multiplexer.

C. Combining Topology Analysis with the Specification

The relative criticality of each scan primitive compared to other scan primitives is determined by the possible damage to the system, if the primitive is faulty. The damage d_j of a primitive j is calculated as a weighted sum of the instruments, which become inaccessible due the fault in this primitive:

$$d_j = \sum_{i=1}^{N} do_i * y_{i,j} + \sum_{i=1}^{N} ds_i * z_{i,j} \tag{1}$$

where $y_{i,j} := 1$ if the instrument i becomes unobservable, when the primitive j is defect; $z_{i,j} := 1$ if the instrument i is not settable due to a defect in the primitive j.

The computation is done hierarchically, starting from the lowest left node of a binary decomposition tree and follows the order of a reverse polish notation. Thereby, the relative criticality of the primitives located in the lower levels of the tree is computed before the computation of their parents starts. For a series-parallel RSN, the values of $y_{i,j}$ and $z_{i,j}$ are efficiently assessed with the help of a decomposition tree:

- For a defect in primitive j, the value of $y_{i,j}$ is set to one, if the instrument i is disconnected from the primitive j in the observability tree. Otherwise, it is set to zero. Similarly, a settability tree is used to assess $z_{i,j}$.
- If a primitive j is hardened, a fault f is avoided and the initial decomposition tree is used to assess $y_{i,j}$ and $z_{i,j}$.

V. Selective Hardening

The desired solution to the hardening problem should satisfy the following optimization criteria:

- A *maximized number of instruments* remains observable and settable through the RSN even in the presence of defects. The *damage* to the system operation is *minimized*:

$$\sum_{j=1}^{N} d_j \rightarrow min \tag{2}$$

where N is the total number of scan primitives;
- The *total cost of hardening* is *minimized*:

$$\sum_{i=1}^{N} c_i * x_i \rightarrow min \tag{3}$$

where the weight c_i is the hardening cost for a primitive i; the variable $x_i := 1$, if the primitive is hardened.

Although, we can directly control only the values of x_i, the interdependence between the values of variables x_i and $y_{i,j}$ allows to implicitly control the values of $y_{i,j}$. If a certain primitive is hardened, it implies that a defect in this primitive cannot occur and the observability of its children is not affected. The same applies to the settability of the instruments.

Minimizing the costs of hardening increases in general the damage due to defects in the RSN, and vice versa. Therefore, a trade-off between these criteria is investigated by computing close to pareto-optimal solutions. The parameter space is explored by applying the evolutionary algorithm SPEA-2 [14] of the Opt4J framework from [20].

This subsection explains how the algorithm is applied to the selective hardening problem. Each problem instance is modeled as a gene, which is represented as a list of binary values, which show whether a given primitive is hardened. The optimization algorithm includes the following steps:

1) *Read the initial problem:* The information about an RSN is provided, and any of the r primitives may be hardened.
2) *Generate the initial population:* A diversified set of genes is generated, where random primitives are hardened.
3) *Calculate the fitness function:* The candidates are assessed with the criteria above. The dominant candidates are kept for mating, the dominated ones are dropped.
4) *Check the termination criteria:* If the allowed number of generations is exceeded, the computation terminates with a set of close-to pareto-dominant solutions.
5) *Generate the next population:* A limited number of individuals is selected from current population for mating.
6) *Perform mating:* Crossover and mutation are applied to the selected individuals with a determined probability:
   - *Individual bit mutation:* A random bit is flipped.
   - *One-point crossover:* First offspring: n bits are taken from the first gene; another $r - n$ are taken from the second gene. Second offspring: vice-versa.

   The computation continues from the step 3.

The RSN topology is not affected by the presented method, and the resulting RSNs are not only compatible with all the existing access, test and diagnosis procedures [6–8, 16, 17], but can also use the same access patterns, as the initial RSNs. If the initial RSN is security compliant [21] and/or testable [19], the presented scheme does not destroy these properties.

VI. Experimental Results

Experiments are conducted on an Intel® Core™ i7-8565U CPU @ 1.80GHz × 8 with 16 GB of main memory. The RSN benchmarks are taken from the ITC'16 [22] and the DATE'19 [23] sets. Due to spacing constraints, only medium- and large-sized RSNs are considered. The number of scan segments (Column 1) and multiplexers (Column 2) are given in Table I.

The analysis has been conducted considering an explicit specification, where 70% of all the instruments have randomly assigned non-zero damage weights of losing their observability, and another 70% - of losing the settability. Also, 10% random instruments are set as important for observation, another 10% - for control. The binary decomposition trees have been generated as in [19]. The primitives to harden are selected by using the evolutionary algorithm called SPEA-2 [14] implemented in the Opt4J framework from [20]. The parameters below have been used for the optimization:

TABLE I. Robust RSN Synthesis, SPEA-II, Varying Optimization Criteria

	Benchmark characteristics		Initial assessment		SPEA-II	Minimize cost, Damage $\leq 10\%$		Minimize damage, Cost $\leq 10\%$		Execution time
Design(1)	# Segments	# Multiplexers	Max. Cost	Max. Damage	Generations	Cost	Damage	Cost	Damage	[m:s]
TreeFlat	24	24	350	502	300	7	42	8	26	00:07
TreeUnbalanced	63	28	142	1,656	300	10	155	14	31	00:02
TreeBalanced	90	46	211	4,206	1,000	18	362	21	216	00:03
TreeFlat_Ex	123	60	289	597	2,000	29	57	28	60	00:04
q12710	47	25	127	576	300	8	27	12	19	00:03
a586710	79	47	155	1,010	2,000	5	90	15	24	00:15
p34392	245	142	482	7,932	700	8	683	48	68	00:34
t512505	288	160	713	7,146	1,000	21	699	71	121	00:16
p22810	537	283	1,298	22,911	1,000	33	2,215	28	3,712	01:01
p93791	1,241	653	2,946	293,771	3,500	38	28,681	286	561	06:10
MBIST_1_5_5	113	15	137	74,004	300	32	7,176	13	20,799	00:26
MBIST_1_5_20	1,523	15	362	632,421	400	35	62,264	36	60,344	02:21
MBIST_1_20_20	6,068	45	1,412	8,252,305	500	129	801,889	137	752,261	10:01
MBIST_2_5_5	1,091	28	137	83,509	500	19	8,141	13	12,081	03:45
MBIST_2_5_20	3,041	28	362	560,484	700	34	54,314	36	50,060	04:17
MBIST_2_20_20	12,131	88	1,412	8,174,778	700	129	788,085	138	722,191	08:18
MBIST_5_5_5	2,720	67	411	148,811	500	8	14,213	41	163	01:10
MBIST_5_20_20	30,320	217	385	6,175,005	900	127	614,605	36	1,343,502	15:02
MBIST_5_100_20	151,520	1,017	7,012	203,302,366	200	1,983	20,555,328	701	48,147,171	35:17
MBIST_5_100_100	671,520	1,017	93,447	2,138,755,955	1,500	17,066	213,650,290	8,625	405,742,391	92:01
MBIST_20_20_20	121,265	862	1,412	6,175,005	900	131	605,065	141	537,474	23:40
MBIST_55_20_5	216,305	8,102	512	814,369	500	112	78,595	51	208,782	05:43
MBIST_100_20_5	118,970	2,367	512	639,278	1,800	87	63,268	51	144,057	07:15
MBIST_100_100_5	1,080,305	20,102	2,512	20,977,832	1,200	273	2,096,139	248	2,396,324	59:32

- Size of the population: 300 for the benchmarks with more than 100 muxes, 100 for other benchmarks;
- Independent bit mutation probability: 0.01;
- Standard one-point crossover probability: 0.95.

In the defect-free case, all the instruments are accessible. The values of a cost function with respect to hardware cost and the resulting system damage is presented. First, the initial assessment of costs is provided, if all the primitives are hardened (Column 4). In Column 5 the damage in presence of single defects is provided, when none of the primitives is hardened. The number of generations of an evolutionary algorithm is provided in Column 6. Next, the damage and the costs are provided for two cases:

- The best damage-reducing solution, which requires at most 10% hardened primitives, in Columns 7 and 8.
- The most cost-efficient solution for reducing the damage down to 10% of the initially assessed value (Column 5) in Columns 9 and 10.

All the important instruments remain accessible via the resulting RSNs. The runtime is provided in Column 11 and is acceptable for all the benchmarks.

VII. Conclusion

A method to generate robust Reconfigurable Scan Networks is presented, which ensures reliable access to the most relevant instruments throughout the device lifetime. A minimized number of primitives uses hardened cells of high yield based on the precise criticality analysis, such that all the critical instruments and most of the remaining instruments are accessible through the RSNs even in the presence of defects. A trade-off between the hardening cost and the remaining damage of defects for the observability and the settability of the instruments is investigated by using an evolutionary algorithm and close to pareto-optimal solutions is computed. The experimental results show that efficient hierarchical processing enables scalability with the increasing RSN size and complexity. The resulting RSNs are compatible with the patterns, generated by the existing methods for the initial RSNs.

Acknowledgments

This work was supported by the German Research Foundation (DFG) under grant WU 245/17-2 (ACCESS) and partially supported by Advantest as part of the Graduate School "Intelligent Methods for Test and Reliability" (GS-IMTR) at the University of Stuttgart. We thank the group of Prof. Teich for providing access to the Opt4J framework.

Bibliography

[1] "IEEE Standard for Access and Control of Instrumentation Embedded within a Semiconductor Device," *IEEE Std 1687-2014*, pp. 1–283, 2014.

[2] "IEEE Standard for Test Access Port and Boundary-Scan Architecture," *IEEE Std 1149.1-2013 (Rev. of IEEE Std 1149.1-2001)*, pp. 1–444, 2013.

[3] R. E. Lyons and W. Vanderkulk, "The Use of Triple-Modular Redundancy to Improve Computer Reliability," *IBM Journal of Research and Development*, vol. 6, no. 2, pp. 200–209, 1962.

[4] S. Brandhofer, M. A. Kochte, and H. Wunderlich, "Synthesis of Fault-Tolerant Reconfigurable Scan Networks," in *Proc. Design, Automation Test in Europe Conf. Exhibition (DATE)*, Mar. 2020, pp. 798–803.

[5] E. Larsson, Z. Xiang, and P. Murali, "Graceful Degradation of Reconfigurable Scan Networks," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems (TVLSI)*, vol. 29, no. 7, pp. 1475–1479, 2021.

[6] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Reconfigurable Scan Networks: Modeling, Verification, and Optimal Pattern Generation," *ACM Trans. Design Automation of Electronic Systems (TODAES)*, vol. 20, no. 2, pp. 30:1–30:27, 2015.

[7] R. Cantoro, L. San Paolo *et al.*, "An Evolutionary Technique for Reducing the Duration of Reconfigurable Scan Network Test," in *Proc. IEEE Int.-l Symp. on Design and Diagnostics of Electronic Circuits Systems (DDECS)*, Apr. 2018, pp. 129–134.

[8] A. Damljanovic, A. Jutman *et al.*, "Post-Silicon Validation of IEEE 1687 Reconfigurable Scan Networks," in *Proc. IEEE European Test Symp. (ETS)*, May 2019, pp. 1–6.

[9] C.-H. Wang, N. Lylina *et al.*, "Concurrent Test of Reconfigurable Scan Networks for Self-Aware Systems," in *Proc. IEEE Int.-l Symp. on On-Line Testing And Robust System Design (IOLTS)*, Jun. 2021, pp. 1–7.

[10] D. M. Doman, *Engineering the CMOS Library: Enhancing Digital Design Kits for Competitive Silicon*. Wiley, 2012.

[11] J. Vial, A. Bosio *et al.*, "Using TMR Architectures for Yield Improvement," in *IEEE Int.-l Symp. on Defect and Fault Tolerance of VLSI Systems (DFT)*, Oct. 2008, pp. 7–15.

[12] C. G. Zoellin, H. Wunderlich *et al.*, "Selective Hardening in Early Design Steps," in *Proc. IEEE European Test Symp. (ETS)*, May 2008, pp. 185–190.

[13] S. Mitra, M. Zhang *et al.*, "Built-In Soft Error Resilience for Robust System Design," in *Proc. IEEE Int.-l Conf. on Int.-d Circuit Design and Technology (ICICDT)*, May 2007, pp. 1–6.

[14] M. L. E. Zitzler and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm For Multiobjective Optimization," *Technical Report 103*, May 2001.

[15] K. Deb, A. Pratap *et al.*, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[16] D. Ull, M. Kochte, and H.-J. Wunderlich, "Structure-Oriented Test of Reconfigurable Scan Networks," in *Proc. IEEE Asian Test Symp. (ATS)*, Nov. 2017, pp. 127–132.

[17] R. Cantoro, A. Damljanovic *et al.*, "A Novel Sequence Generation Approach to Diagnose Faults in Reconfigurable Scan Networks," *IEEE Trans. on Comp.*, vol. 69, no. 1, pp. 87–98, 2020.

[18] F. Maamari and J. Rajski, "A method of fault simulation based on stem regions," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 9, no. 2, pp. 212–220, 1990.

[19] N. Lylina, C.-H. Wang, and H.-J. Wunderlich, "Testability-Enhancing Resynthesis of Reconfigurable Scan Networks," in *To appear in Proc. of the IEEE Int.-l Test Conf.(ITC)*, Virtual, Oct. 2021, pp. 1–10.

[20] M. Lukasiewycz, M. Glaß *et al.*, "Opt4J - A Modular Framework for Meta-heuristic Optimization," in *Proc. Genetic and Evolutionary Computing Conf. (GECCO)*, Jul. 2011, pp. 1723–1730.

[21] N. Lylina, A. Atteya *et al.*, "Security Preserving Integration and Resynthesis of Reconfigurable Scan Networks," in *Proc. IEEE Int'l Test Conf. (ITC)*, Nov. 2020, pp. 1–10.

[22] A. Tsertov, A. Jutman *et al.*, "A suite of IEEE 1687 benchmark networks," in *Proc. IEEE Int'l Test Conf. (ITC)*, Nov. 2016, pp. 1–10.

[23] P. Raiola, B. Thiemann *et al.*, "On Secure Data Flow in Reconfigurable Scan Networks," in *Proc. Conf. on Design, Automation Test in Europe (DATE)*, Mar. 2019, pp. 1–6.