Vietnam Journal of Computer Science (2018) 5:263–278
https://doi.org/10.1007/s40595-018-0121-0

REGULAR PAPER

Genetic algorithm as self-test path and circular self-test path design method

Miłosław Chodacki[1]

Received: 28 August 2017 / Accepted: 17 July 2018 / Published online: 27 July 2018

Abstract
The paper presents the use of Genetic Algorithm to search for non-linear Autonomous Test Structures (ATS) in Built-In Testing approach. Such structures can include essentially STP and CSTP and their modifications. Non-linear structures are more difficult to analyze than the widely used structures such as independent Test Pattern Generator and the Test Response Compactor realized by Linear Feedback Shift Registers. To reduce time-consuming test simulation of sequential circuit, it was used an approach based on the stochastic model of pseudo-random testing. The use of stochastic model significantly affects the time effectiveness of the search for evolutionary autonomous structures. In test simulation procedure, the block of sequential circuit memory is not disconnected. This approach does not require a special selection of memory registers such as BILBOs. A series of studies to test circuits set ISCAS'89 are made. The results of the study are very promising.

Keywords Built-in self-test · Self-test path · Digital testing · Genetic algorithm

1 Introduction

Digital systems should provide services according to the specifications reliably. Impairments of dependability are associated with a large class of faults, errors and failures. These impairments may be caused by design, produce or rarely operational imperfections and improper use. There are lots of possible circuit failures: single stuck at 0 or 1 faults, delay and synchronization faults, bridging and open faults, in Metal Oxide Semiconductor technique (MOS) these faults consist in transistor stuck on or stuck off in a logical gates [1]. Some faults cannot be logically represented. Other class of faults can be connected with operational timing frequency and they are related to change impedance parameters, but in that case the built-in testing is one of the most resistant technique because of common silicon space. Faults that are stimulated may manifest itself as an error. For do that the fault have to be stimulated and propagated to one of internal (to memory module of sequential circuit) or external (primary) circuit output. The error, that is accessible from circuit output, is an information on detected fault and indicates that functional specification of circuit is violated. There is therefore a need for hardware testing.

For Very Large Scale Integration circuits (VLSI), the Built-In Self-Testing (BIST) concept is well used. Embedding the whole or major part of the tester into the circuit is considered as BIST. Production standard involves the use of Linear Feedback Shift Registers (LSFR) that are used as a Test Pattern Generators (TPG) and Test Response Compactors (TRC) in a signature analysis [2]. These LFSR registers can generate pseudo-random test vectors that may cover many faults. For the evaluation of the effectiveness of coating defects by sequence of test vectors, the Fault Coverage (FC) is applied. Multi-Input Signature Registers (MISR) and Single-Input Signature Registers (SISR) are mainly used as TRC registers [3]. These Compactors perform data compression generally lossy, but are known lossless Zero-Aliasing ones without faults masking phenomena.

There is also a non-linear technique with Self-Test Path (STP) or Circular STP (CSTP). Some modifications of these self-testing techniques are also known, e.g., Circular CSTP (C2STP) [4]. Contrary to linear technique, the Circuit Under Test (CUT) in non-linear technique is a feedback of STP or CSTP, thus posing a problem with parameter selection for these structures. These structures can be implemented in Field Programmable Gate Arrays (FPGA) [5], Application Specific Integrated Circuit (ASIC), System-on-Chip (SOC),

✉ Miłosław Chodacki
miloslaw.chodacki@us.edu.pl

[1] Institute of Computer Science, Silesian University, Katowice, Poland

which consist of many virtual Intellectual Property modules (IP Core). For SOC, the STP structures can also link IP modules [6].

Nevertheless, simulations presented in this paper show that it is possible to design such BIST, modeled as NLFSR, that achieve higher effectiveness than those of solutions reported in the literature and often are minimized. The minimization is related to the concept of external self-testing, where internal Memory Module (MM) of the circuit is disconnected during test, thus no additional conditions are imposed on its operation. It should be noted that both in linear and non-linear testing techniques, the circuit MM is typically included into self-testing structure registers as results from ability to improve testability and application of Design for Testability (DFT). An important observations was made in [7].

The properties of evolutionary algorithms make possible to solve a non-linear structures designing problem. The modeling of NLFSRs is important for the sake of correct representation of it. In this paper, many evolutionary models of non-linear register for evolutionary computer simulations are shown. In addition, many different design methods, often with the use of Genetic Algorithms (GATTO, GATTO+, GATTO*, SELFISH GENE GA [8–10]) and deterministic systems based, among other things, on Automatic Test Pattern Generation (ATPG) [11], Cellular Automata (CA) [12], Finite State Machines (FSM), and Binary Decision Diagrams (BDD) are used to built-in testing design. There are some solutions that can create a sequence of test vectors. In this paper, we are searching for not only a selection of sequences, but for structures that can generate these sequences with good FC parameter.

The paper is organized as follows. In Sect. 2, basic information on NLFSR as ATS, essentially STP and evolutionary are presented. Section 3 includes some description of the Genetic Algorithm and its using to create ATS structures. Next, in Sect. 4, some results of evolutionary searching for STP/CSTP structures are presented, and finally Sect. 5 concludes the paper.

The paper is an extended version of a previously published article titled "Genetic Algorithm for Self-Test Path and Circular Self-Test Path Design" that was presented at ACIIDS 2017 conference [7].

2 Non-linear feedback shift register as STP and CSTP model

Self-Test Path, Circular Self-Test Path and Condensed Circular Self-Test Path (C2STP) and in general NLFSRs can be seen as realization of Autonomous Test Structure (ATS).

In Fig. 1, a schema of autonomous structure STP that works accordingly to (1) is presented.

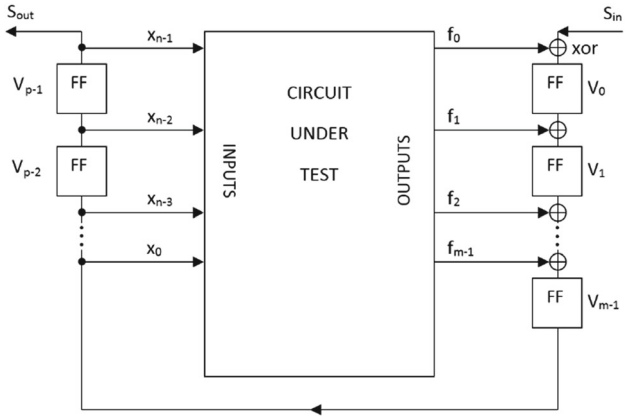

Fig. 1 Self-test path model

In Fig. 1, V_i is an element of STP and it is mainly D-type flip-flop, t is a discrete time (clock time) and $p = n + m - 1$ is a length of STP (number of flip-flops).

$$\begin{vmatrix} V_0(t+1) \\ V_1(t+1) \\ \vdots \\ V_{m-1}(t+1) \\ \vdots \\ V_{p-2}(t+1) \\ V_{p-1}(t+1) \end{vmatrix} = T * \begin{vmatrix} V_0(t) \\ V_1(t) \\ \vdots \\ V_{m-1}(t) \\ \vdots \\ V_{p-2}(t) \\ V_{p-1}(t) \end{vmatrix} \oplus \begin{vmatrix} f_0(V_{\text{input}}(t)) \oplus S_{\text{in}} \\ f_1(V_{\text{input}}(t)) \\ \vdots \\ f_{m-1}(V_{\text{input}}(t)) \\ \vdots \\ 0 \\ 0 \end{vmatrix}, \quad (1)$$

where $\oplus$ denotes an addition operator over GF(2) and

$$V_{\text{input}}(t) = V_{m-1}(t), \ldots, V_{p-2}(t), V_{p-1}(t) \quad (2)$$

and

$$T = \begin{vmatrix} 0 & 0 & \ldots & 0 & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \ldots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 0 & \ldots & 1 & 0 \end{vmatrix} \quad (3)$$

is the Connection Matrix for D-type flip-flops which are used to create STP register. If some ith flip-flop is a T-type one then $[T_{i,i}] = 1$. The last element in the first row of the matrix $[T_{0,p-1}] = 1$ for CSTP structure and in the rest of the paper ATS based on CSTP. The schema of CSTP is shown in Fig. 2.

2.1 Using additional linear feedback

By taking into account the type of additional STP linear feedback, it can be distinguished different connection matrix

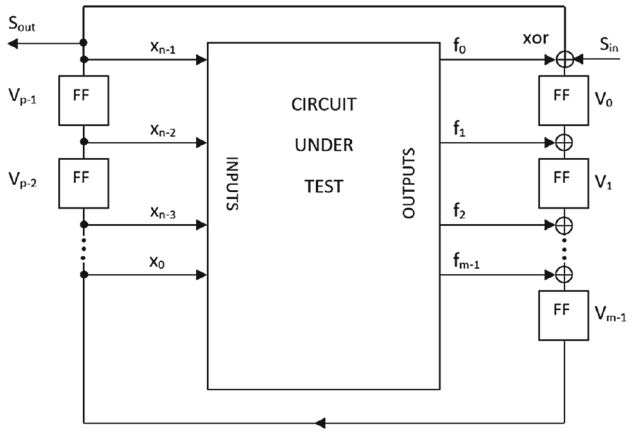

Fig. 2 Circular self-test path model

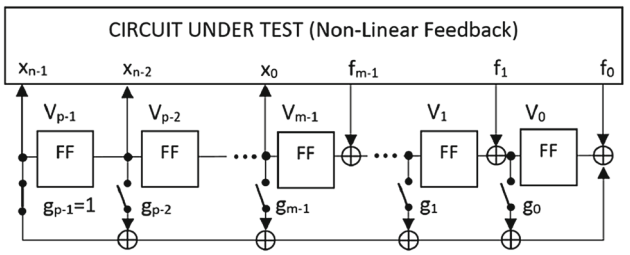

Fig. 3 External, top-feedback ExOR register

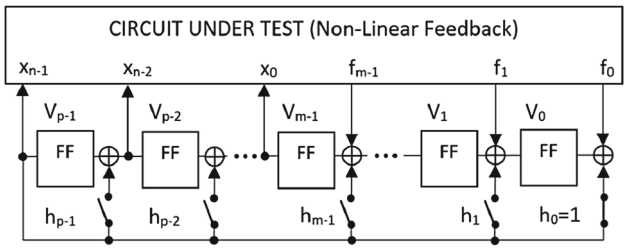

Fig. 4 Internal, bottom-feedback ExOR register

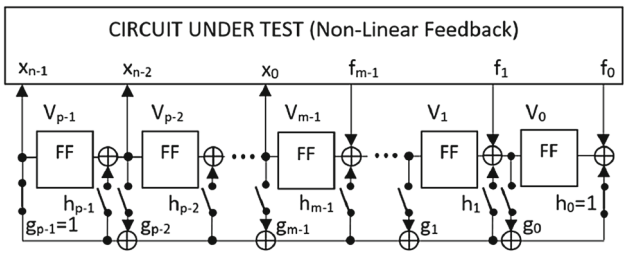

Fig. 5 External–internal, top–bottom ExOR register

forms: external, Top-Feedback ExOR (4) (Fig. 3), internal, Bottom-Feedback ExOR (5) (Fig. 4) and external–internal, Top–Bottom-Feedback ExOR (6) (Fig. 5).

$$T_g = \begin{vmatrix} g_0 & g_1 & \dots & g_{m-1} & \dots & g_{p-2} & g_{p-1}=1 \\ 1 & 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 1 & 0 \end{vmatrix}, \tag{4}$$

$$T_h = \begin{vmatrix} 0 & 0 & \dots & 0 & \dots & 0 & h_0=1 \\ 1 & 0 & \dots & 0 & \dots & 0 & h_1 \\ \vdots & \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \dots & 0 & h_{m-1} \\ \vdots & \vdots & \dots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 1 & h_{p-1} \end{vmatrix}, \tag{5}$$

$$T_{gh} = \begin{vmatrix} g_0 & g_1 & \dots & g_{m-1} & \dots & g_{p-2} & g_{p-1}=h_0 \\ 1 & a_{1,1} & \dots & a_{1,m-1} & \dots & a_{1,p-2} & h_1 \\ \vdots & \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \dots & a_{m-1,p-2} & h_{m-1} \\ \vdots & \vdots & \dots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 1 & h_{p-1} \end{vmatrix}, \tag{6}$$

where

$$a_{i,j} = \begin{cases} 1 & \text{if } h_i = g_j = 1, \\ 0 & \text{if } h_i \neq g_j \text{ or } h_i = g_j = 0, \end{cases} \tag{7}$$

and $g_{p-1} = h_0 = 1$, $g_i, h_j, a_{i,j} \in \mathrm{GF}(2)$.

2.2 Using connection matrices

NLFSR register can be connected to CUT in different ways. Equation (1) can be expressed simply by (8)

$$V(t+1) = T * V(t) \oplus F(V(t)). \tag{8}$$

It is possible to change connection schema from CUT to STP/CSTP register using output matrices (OM) in a few modes ((10), (11), (12)) according to (9)

$$V(t+1) = T * V(t) \oplus \mathrm{OM} * F(V(t)). \tag{9}$$

Output matrix can be realized by

$$\mathrm{OM_E} = \begin{vmatrix} 1 & 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 0 & 0 \end{vmatrix}, \tag{10}$$

$$\mathrm{OM}_1 = \begin{vmatrix} 0 & 1 & \dots & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 1 & 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 0 & 0 \end{vmatrix}, \tag{11}$$

$$\mathrm{OM}_{\mathrm{Free}} = \begin{vmatrix} 1 & 1 & \dots & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 1 & 1 & \dots & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & \dots & 0 & 0 \end{vmatrix}. \tag{12}$$

The $\mathrm{OM_E}$ matrix is identity matrix, but OM_1 matrix must meet the following condition (13):

$$\forall_i \exists_{0\le j\le m-1}![\mathrm{OM}_{i,j}]=1 \quad \text{and} \quad \forall_j \exists_{0\le i\le m-1}![\mathrm{OM}_{i,j}] = 1, \tag{13}$$

where i and j represent the rows and columns of OM_1 matrix, respectively, and m is a number of circuit outputs. Matrix $\mathrm{OM_{Free}}$ must meet the following condition (14):

$$\forall_{0\le i\le m-1} \exists_{0\le j\le m-1} [\mathrm{OM}_{i,j}] = 1. \tag{14}$$

Depending on the contents of the minor, the three types of output connection matrices can be distinguished as output matrix E, output matrix 1 and output matrix free (Fig. 6). The similar notations can be applied to input connection matrices (IM) [7].

Observe that $[X_{\mathrm{input}}]_n = [x_0, \dots, x_{n-1}]^{\mathrm{T}}$ symbolizes circuit inputs, then input matrix can be represented simple as (16) according to (15)

$$X_{\mathrm{input}}(t+1) = \mathrm{IM} * V(t+1), \tag{15}$$

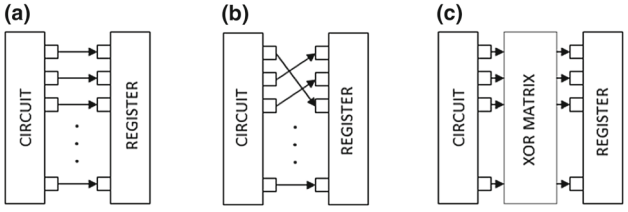

Fig. 6 Circuit outputs to register inputs connection type **a** output matrix E, **b** output matrix 1, **c** output matrix free

$$\mathrm{IM} = \begin{vmatrix} 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ i_{m-1,0} & \dots & i_{m-1,m-1} & \dots & i_{m-1,p-2} & i_{m-1,p-1} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ i_{p-2,0} & \dots & i_{p-2,m-1} & \dots & i_{p-2,p-2} & i_{p-2,p-1} \\ i_{p-1,0} & \dots & i_{p-1,m-1} & \dots & i_{p-1,p-2} & i_{p-1,p-1} \end{vmatrix}. \tag{16}$$

There are few forms of input matrix, analogously to OMs (17), (18), (19) and specific ones (20) and (21):

$$\mathrm{IM_E} = \begin{vmatrix} 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 0 & \dots & 1 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & \dots & 1 & 0 \\ 0 & \dots & 0 & \dots & 0 & 1 \end{vmatrix}, \tag{17}$$

$$\mathrm{IM}_1 = \begin{vmatrix} 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 0 & \dots & 0 & \dots & 1 & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & \dots & 0 & 1 \\ 0 & \dots & 1 & \dots & 0 & 0 \end{vmatrix}, \tag{18}$$

$$\mathrm{IM_{Free}} = \begin{vmatrix} 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 0 & \dots & 1 & \dots & 0 & 1 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & \dots & 1 & 0 \\ 0 & \dots & 1 & \dots & 1 & 1 \end{vmatrix}, \tag{19}$$

$$\mathrm{IM_{1Long}} = \begin{vmatrix} 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 1 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 1 & \dots & 0 & 0 \\ 0 & \dots & 0 & \dots & 1 & 0 \end{vmatrix}, \tag{20}$$

$$\mathrm{IM_{FreeLong}} = \begin{vmatrix} 0 & \dots & 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \dots & \vdots & \vdots \\ 1 & \dots & 1 & \dots & 1 & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 1 & \dots & 1 & 0 \\ 1 & \dots & 0 & \dots & 1 & 1 \end{vmatrix}. \tag{21}$$

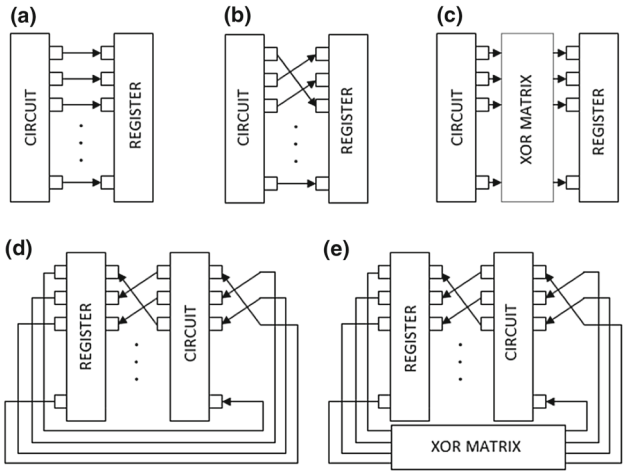

Fig. 7 Register outputs to circuit inputs connection type **a** input matrix E, **b** input matrix 1, **c** input matrix free, **d** input matrix 1 long, **e** input matrix free long

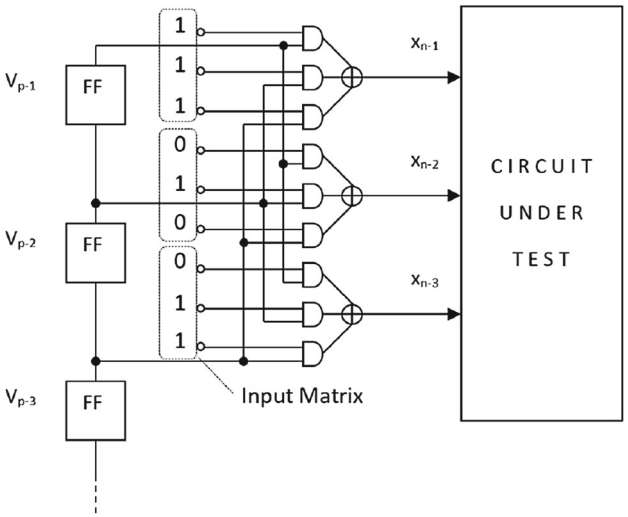

Fig. 8 Input matrix connection schema interpretation

All of the input matrices must satisfy the following formula (22):

$$\forall_{0\le i<m-1,0\le j\le p-1}[\mathrm{IM}_{i,j}]=0. \tag{22}$$

Additionally, matrices $\mathrm{IM_E}$, $\mathrm{IM_1}$ and $\mathrm{IM_{Free}}$ must satisfy the following formula (23):

$$\forall_{i\le p-1,0\le j<m-1}[\mathrm{IM}_{i,j}]=0. \tag{23}$$

$\mathrm{IM_E}$ matrix is identity matrix, but $\mathrm{IM_1}$ matrix must meet the following conditions (24 and 25):

$$\forall_{m-1\le i\le p-1}\exists_{m-1\le j\le p-1}![\mathrm{IM}_{i,j}]=1, \tag{24}$$

and

$$\forall_{m-1\le j\le p-1}\exists_{m-1\le i\le p-1}![\mathrm{IM}_{i,j}]=1, \tag{25}$$

where i and j represent the rows and columns of $\mathrm{IM_1}$ matrix, respectively.

Matrix $\mathrm{IM_{Free}}$ must meet the following condition (26):

$$\forall_{m-1\le i\le p-1}\exists_{m-1\le j\le p-1}[\mathrm{IM}_{i,j}]=1. \tag{26}$$

Matrices $\mathrm{IM_{1Long}}$ and $\mathrm{IM_{FreeLong}}$ satisfy similar conditions to $\mathrm{IM_1}$ and $\mathrm{IM_{FreeLong}}$, respectively, without fulfilling the (23) condition.

In Fig. 7, schema of all input matrix types are presented, and in Fig. 8, the example of connection schema (XOR matrix circuit) coded by some $\mathrm{IM_{Free}}$ is shown.

2.3 Configuration of ATS model

The following linear feedback types can be chosen when configuring the ATS model:

- AIJ Top–Bottom LFSR (1–1500), additional external and internal linear feedbacks are possible,
- Bottom LFSR (1500–3000), additional internal linear feedback is possible,
- Shift Register (3000–4500), no additional linear feedback,
- Top LFSR (4500–6000), additional external linear feedback is possible,
- Top–Bottom LFSR (6000–7500), additional external and internal linear feedbacks (other than AIJ Top–Bottom LFSR) are possible.

To configure the STP register connections with the tested circuit, the following connection diagram types were distinguished: for circuit inputs,

- Input Matrix 1 (1–300), complex connections available to the part of the STP register that controls inputs of the tested circuit,
- Input Matrix 1 Long (300–600), complex connection, while allowing connections with any component of the STP register,
- Input Matrix E (600–900), simple connections (as shown in Fig. 1),
- Input Matrix Free (900–1200), connections through XOR matrices, but only with those STP register components that control inputs of the tested circuit,
- Input Matrix Free Long (1200–1500), connection through XOR matrices with any STP register components.

for circuit outputs:

- Output Matrix 1 (1–100), complex connections, available for those components of STP register that are responsible circuit response.
- Output Matrix E (100–200), simple connections (as shown in Fig. 1),
- Output Matrix Free (200–300), connections through XOR matrices, but only with those STP components that are responsible for circuit response receiving.

In brackets, there are identifiers being useful in analysis of simulation graphs presented in Fig. 11 and so on. In Fig. 9, all STP configurations are presented (CSTP configurations have a similar notation that starts from 7500 to 15000).

3 Genetic algorithm as NLFSR design method

Genetic algorithm (GA) has some useful features, such as the ability to deliver multiple point solutions, and so the lack of concentration of solutions around a certain subclass of STP/CSTP structure and configuration. The algorithm mimics natural evolutionary processes, and therefore there exists the possibility of self-control calculations in such a way that a solution better adapted to a greater extent affects the entire population of solutions (selective pressure).

The GA directs the search in the space of feasible solutions by environmental evaluation of the fitness function of each solution (individual). The course of the algorithm is presented in Fig. 10.

The process of STP/CSTP design creation is a complicated one, especially due to the difficulty of non-linear circuit feedback and BIST simulation time. Every individual has to be simulated and this process is a great time-consuming task. In [7], the stochastic model of pseudo-random testing, which significantly reduces this problem, was described. Using the stochastic model, the simulation of each solution is well reduced due to the conversion of exploration FC in search of a suitable length of sequence. Fitness function can be described as some optimization problem in which one is looking for such $x^* \in V(p)$ that maximizes the following formula (27):

$$f(x^*) = \max_{x \in V(p)} f(x), \tag{27}$$

where $V(p)$ is a multidimensional vector of parameters. The fitness function is defined as follows (28):

$$\begin{aligned}\text{Fitness}(x \in V(p)) = {} & w_0 x_0 + w_1(x_{1\max} - x_1) \\ & + w_2(x_{2\max} - x_2),\end{aligned} \tag{28}$$

ATS ID (	ID >	REGISTER TYPE	INPUT MATRIX TYPE	OUTPUT MATRIX TYPE
1	100	AIJ TOP-BOTTOM LINEAR FEEDBACK SHIFT REGISTER	INPUT MATRIX 1	OUTPUT MATRIX 1
100	200			OUTPUT MATRIX E
200	300			OUTPUT MATRIX FREE
300	400		INPUT MATRIX 1 LONG	OUTPUT MATRIX 1
400	500			OUTPUT MATRIX E
500	600			OUTPUT MATRIX FREE
600	700		INPUT MATRIX E	OUTPUT MATRIX 1
700	800			OUTPUT MATRIX E
800	900			OUTPUT MATRIX FREE
900	1000		INPUT MATRIX FREE	OUTPUT MATRIX 1
1000	1100			OUTPUT MATRIX E
1100	1200			OUTPUT MATRIX FREE
1200	1300		INPUT MATRIX FREE LONG	OUTPUT MATRIX 1
1300	1400			OUTPUT MATRIX E
1400	1500			OUTPUT MATRIX FREE
1500	1600	BOTTOM LINEAR FEEDBACK SHIFT REGISTER	INPUT MATRIX 1	OUTPUT MATRIX 1
1600	1700			OUTPUT MATRIX E
1700	1800			OUTPUT MATRIX FREE
1800	1900		INPUT MATRIX 1 LONG	OUTPUT MATRIX 1
1900	2000			OUTPUT MATRIX E
2000	2100			OUTPUT MATRIX FREE
2100	2200		INPUT MATRIX E	OUTPUT MATRIX 1
2200	2300			OUTPUT MATRIX E
2300	2400			OUTPUT MATRIX FREE
2400	2500		INPUT MATRIX FREE	OUTPUT MATRIX 1
2500	2600			OUTPUT MATRIX E
2600	2700			OUTPUT MATRIX FREE
2700	2800		INPUT MATRIX FREE LONG	OUTPUT MATRIX 1
2800	2900			OUTPUT MATRIX E
2900	3000			OUTPUT MATRIX FREE
3000	3100	SHIFT REGISTER	INPUT MATRIX 1	OUTPUT MATRIX 1
3100	3200			OUTPUT MATRIX E
3200	3300			OUTPUT MATRIX FREE
3300	3400		INPUT MATRIX 1 LONG	OUTPUT MATRIX 1
3400	3500			OUTPUT MATRIX E
3500	3600			OUTPUT MATRIX FREE
3600	3700		INPUT MATRIX E	OUTPUT MATRIX 1
3700	3800			OUTPUT MATRIX E
3800	3900			OUTPUT MATRIX FREE
3900	4000		INPUT MATRIX FREE	OUTPUT MATRIX 1
4000	4100			OUTPUT MATRIX E
4100	4200			OUTPUT MATRIX FREE
4200	4300		INPUT MATRIX FREE LONG	OUTPUT MATRIX 1
4300	4400			OUTPUT MATRIX E
4400	4500			OUTPUT MATRIX FREE
4500	4600	TOP LINEAR FEEDBACK SHIFT REGISTER	INPUT MATRIX 1	OUTPUT MATRIX 1
4600	4700			OUTPUT MATRIX E
4700	4800			OUTPUT MATRIX FREE
4800	4900		INPUT MATRIX 1 LONG	OUTPUT MATRIX 1
4900	5000			OUTPUT MATRIX E
5000	5100			OUTPUT MATRIX FREE
5100	5200		INPUT MATRIX E	OUTPUT MATRIX 1
5200	5300			OUTPUT MATRIX E
5300	5400			OUTPUT MATRIX FREE
5400	5500		INPUT MATRIX FREE	OUTPUT MATRIX 1
5500	5600			OUTPUT MATRIX E
5600	5700			OUTPUT MATRIX FREE
5700	5800		INPUT MATRIX FREE LONG	OUTPUT MATRIX 1
5800	5900			OUTPUT MATRIX E
5900	6000			OUTPUT MATRIX FREE
6000	6100	TOP-BOTTOM LINEAR FEEDBACK SHIFT REGISTER	INPUT MATRIX 1	OUTPUT MATRIX 1
6100	6200			OUTPUT MATRIX E
6200	6300			OUTPUT MATRIX FREE
6300	6400		INPUT MATRIX 1 LONG	OUTPUT MATRIX 1
6400	6500			OUTPUT MATRIX E
6500	6600			OUTPUT MATRIX FREE
6600	6700		INPUT MATRIX E	OUTPUT MATRIX 1
6700	6800			OUTPUT MATRIX E
6800	6900			OUTPUT MATRIX FREE
6900	7000		INPUT MATRIX FREE	OUTPUT MATRIX 1
7000	7100			OUTPUT MATRIX E
7100	7200			OUTPUT MATRIX FREE
7200	7300		INPUT MATRIX FREE LONG	OUTPUT MATRIX 1
7300	7400			OUTPUT MATRIX E
7400	7500			OUTPUT MATRIX FREE

Fig. 9 ATS-STP configuration

where x_0 is a length of sequence, x_1 is a number of ExORs used to create additional linear feedback, x_2 is a number of T-type flip-flops used to design NLFSR and $\sum_{i=0}^{n=2} w_i = 1$.

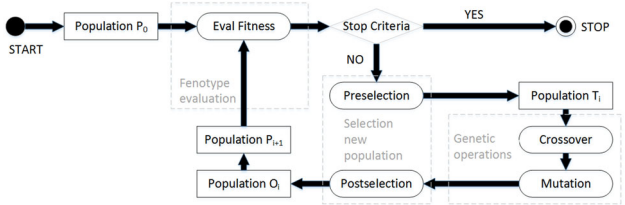

Fig. 10 Evolution in genetic algorithm

Table 1 ISCAS'89 subset

Circuit	Inputs	Outputs	Dffs	Gates	Inverts	Faults
s208.1	10	1	8	66	38	217
s298	3	6	14	75	44	308
s349	9	11	15	104	57	350
s382	3	6	21	99	59	399
s444	3	6	21	119	62	474
s820	18	19	5	256	33	850
s1196	14	14	18	388	141	1242
s1238	14	14	18	428	80	1355
s1423	17	5	74	490	167	1515
s1494	8	19	6	558	89	1506

The linear code of solution (chromosome) is a binary array of bit values that stores

- Initial state of register;
- Initial state of circuit memory module (MM);
- Type of register flip-flops, e.g., D, T;
- Schema of additional external/internal linear feedback;
- Type and values of Input Matrix IM;
- Type and values of Output Matrix OM;
- Number of included circuit memory elements MM as part of ATS.

In the initial stage of the genetic algorithm, essentially random P_0 population base is created. The population is further assessed by the environment (fitness function). Based on the adaptation of individuals, their reproduction to temporary populations T_i is made. Then from T_i using genetic operators crossover and mutation with some probabilities, the descendant population (Offspring) O_i is created. Next evaluation of the newly established offspring population P_{i+1} takes place iteratively until the stopping criteria are fulfilled. This evolutionary process is common to all Genetic Algorithms. The ability to generate invalid connection matrices, i.e., not meeting the above conditions, was excluded by a particular type of chromosome encoding. With this approach, it is not necessary to use repair algorithms or penalty function for a faulty solution.

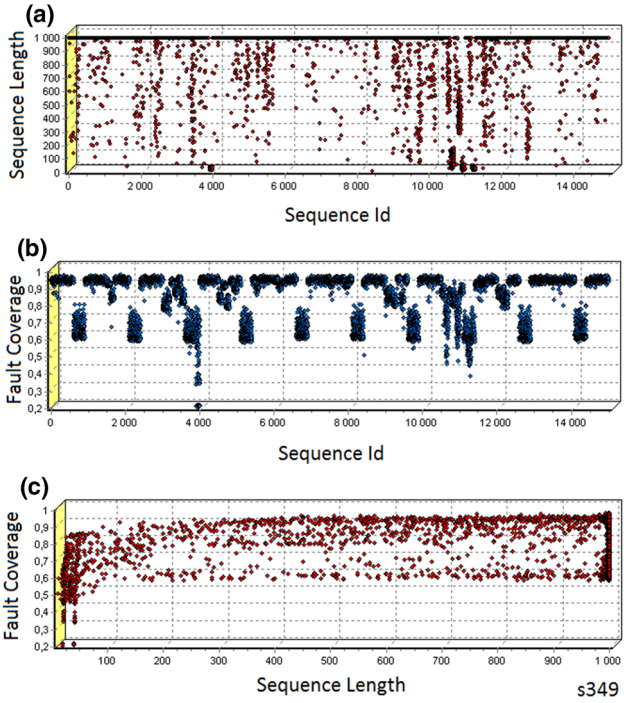

Fig. 11 Simulation graph for s349 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–15000)

The criterion for stopping the algorithm is to reach an acceptable FC value or exceed a predetermined number of generations.

4 Results

The experiments presented here were performed for a subset of ISCAS'89 sequential circuits presented in Table 1.

In Fig. 11b, one can notice a specific repeatability of FC 0.6–0.8 for STP/CSTP resulting from the presence of type 1 Long connection matrix (e.g., Seq. Id. 300–600, 1800–2100 and so on). This type of matrix can actually reduce the length of the ATS structure and thereby reduce the length of the test sequence (Fig. 11a). For other connection matrices, FC reaches the value of 1. However, the described phenomenon occurs for the s349, only.

Figure 12a shows that for the s382 within certain ATS structures, the focused values of FC of small discrepancies are obtained. Figure 12b for the same circuit can be noted that only a few configurations of ATS structure can be able to obtain the FC at around 0.9 (Seq. Id 2100–2300 for STP, 9700–9800 and 12800–12900 for CSTP). In these structures, matrices such as Input Matrix Free and Output Matrix Free are used. An interesting area is that identified by Id Seq. 10500 and 11700, there is ATS structure realized by a simple Shift Register (SR) generating short sequences, which, how-

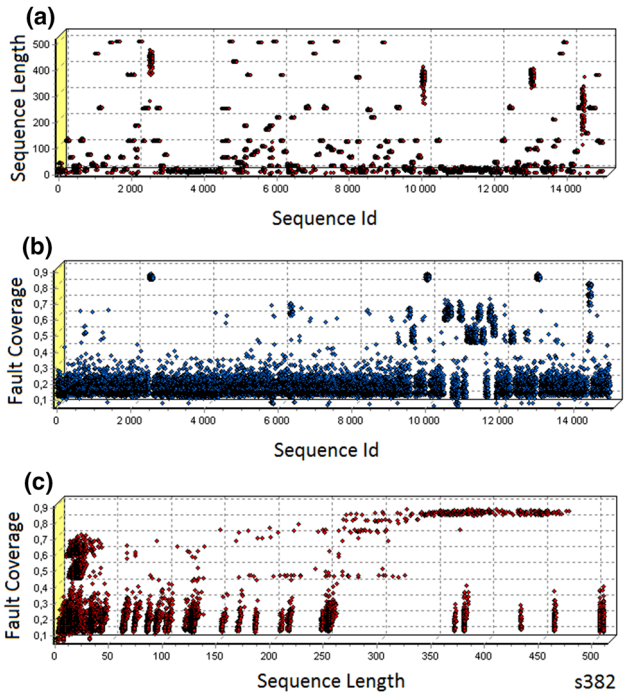

Fig. 12 Simulation graph for s382 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–1500)

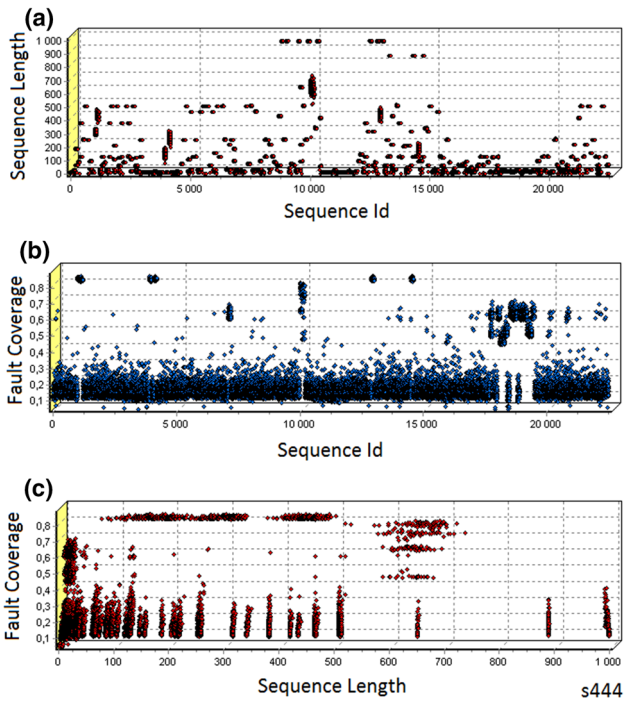

Fig. 13 Simulation graph for s444 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–15000), CSTP with additional DFF (Id 15001–22500)

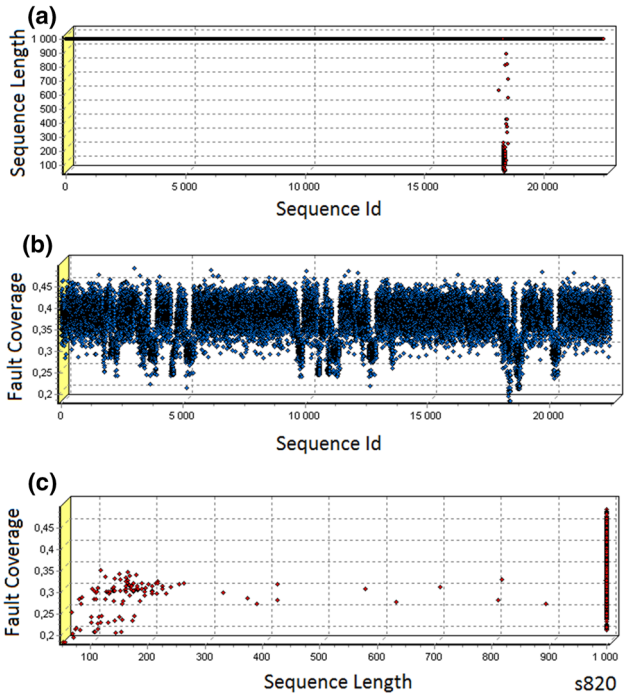

Fig. 14 Simulation graph for s820 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–15000), CSTP with additional DFF (Id 15001–22500)

ever, allow for the acquisition of a relatively high FC value. In this area, there is an additional CSTP feedback. The charts shown in Fig. 13 for the s444 are in some ways similar to the graph in Fig. 12 for the s382. Both test circuits are traffic light controllers and have the BCD counters (timers).

In Fig. 14 for the s820, it can be seen that almost independently of the ATS structure the FC is recovered in the range of 0.3–0.45. The best achieved result was FC = 0.598. The s820 circuit has only a few flip-flops but some portion of the state space which should be taken into account is fault dependent.

Statistical analysis of the results showed a correlation from low (below than 0.1) to strong (greater than 0.9), between the length of the sequences and FC for the specified ATS structures. For example, for s208.1, the correlation ranges between 0.19 and 0.914 (Table 2). Figure 15 shows the FC dependence on the length of test sequence for many ATS (more precisely STP) configurations. The correlation coefficient value is 0.7. Figure 16 can be seen that the vast majority of the structure ATS (sequence ID) generates test sequences of length 1000, except for a singularity identified by the 10500–11500th corresponds to the CSTP structure based on the Shift Register without the additional linear feedback. Interestingly, in Fig. 16, there is a high correlation between the length of the generated sequence and the value of FC. Most ATS structures for this circuit cover from 0.7 to nearly 0.9 faults beyond the aforementioned singularity. The results of the study for s1238

Table 2 ATS statistical results for s208.1 circuit

ATS ID	$Seq._{Len.}$	$\sigma_{Seq.Len}$	FC	σ_{FC}	Corr
001–100	354	75.04	0.991	0.168	0.708
100–200	574	116.26	0.986	0.203	0.846
200–300	362	65.22	0.963	0.172	0.675
300–400	121	22.37	0.958	0.204	0.805
400–500	238	51.87	0.986	0.221	0.757
500–600	210	42.55	0.940	0.211	0.676
600–700	611	136.75	0.991	0.134	0.838
700–800	518	109.14	**1.000**	0.187	0.842
800–900	699	164.93	0.995	0.122	0.700
900–1000	491	94.45	0.986	0.174	0.664
1000–1100	345	69.97	0.995	0.153	0.454
1100–1200	510	140.21	0.991	0.170	0.810
1200–1300	271	74.91	0.986	0.226	0.914
1300–1400	787	176.60	0.986	0.149	0.813
1400–1500	735	74.25	0.986	0.063	0.530
1500–1600	554	84.29	**1.000**	0.101	0.725
1600–1700	300	51.83	0.981	0.199	0.652
1700–1800	582	141.41	0.991	0.158	0.804
1800–1900	55	8.991	0.756	0.111	0.652
1900–2000	172	35.62	0.737	0.117	0.196
2000–2100	69	11.95	0.516	0.084	0.339
2100–2200	618	130.26	0.963	0.142	0.759
2200–2300	545	95.23	**1.000**	0.163	0.882
2300–2400	656	129.30	0.991	0.131	0.661
2400–2500	630	120.47	0.991	0.103	0.568
2500–2600	510	116.14	**1.000**	0.197	0.849
2600–2700	589	120.53	0.986	0.082	0.627
2700–2800	779	115.81	0.995	0.086	0.385
2800–2900	558	100.94	0.995	0.130	0.798
2900–3000	392	120.10	0.977	0.227	0.892

ATS-STP ID from 1 to 3000 (Fig. 9), the numbers located in second and fourth columns are maximum
Bold values indicate covering all faults

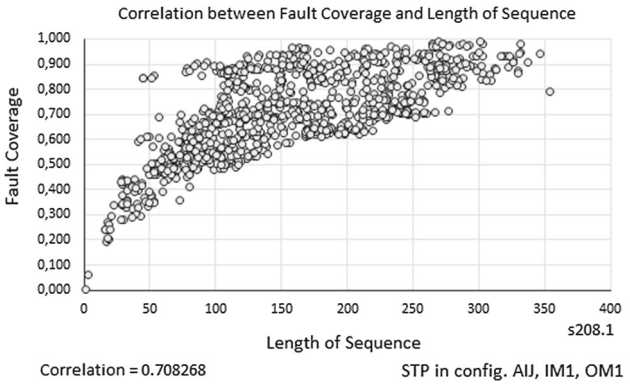

Fig. 15 Correlation between the length of sequence and FC for s208.1 circuit. ATS configuration ID from 1 to 100 (Fig. 9)

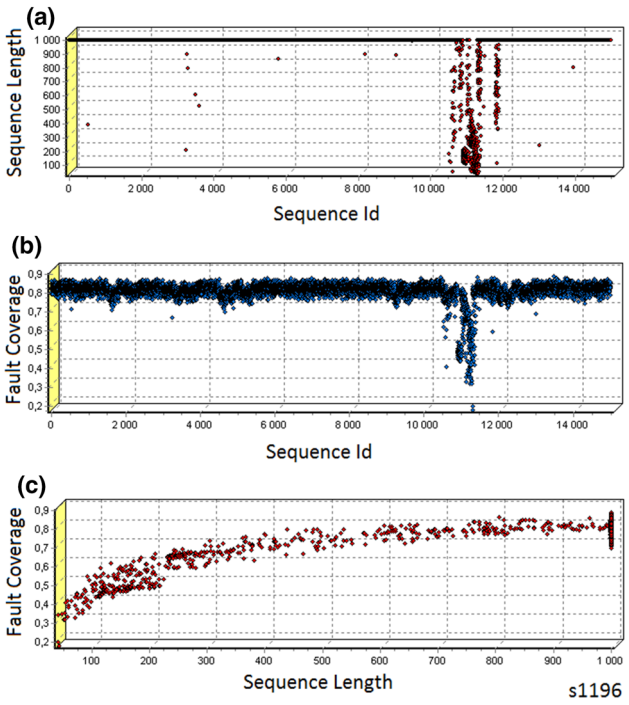

Fig. 16 Simulation graph for s1196 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–15000), CSTP with additional DFF (Id 15001–22500)

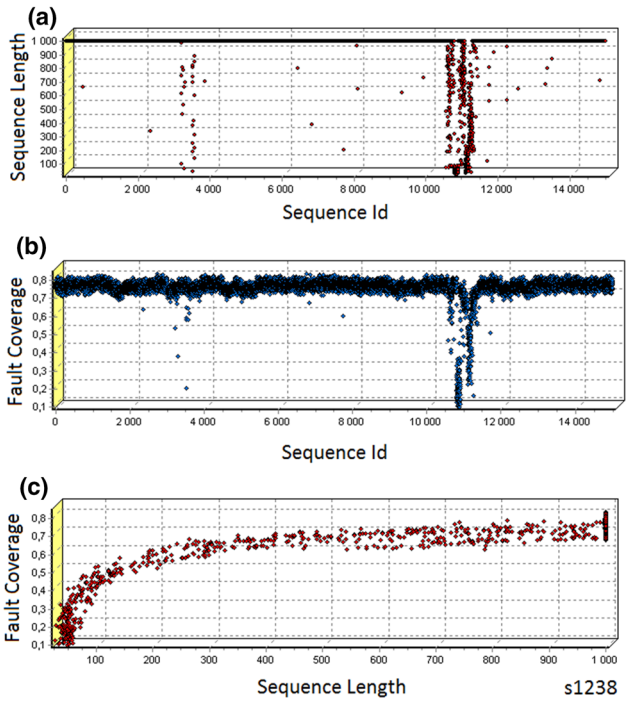

Fig. 17 Simulation graph for s1238 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–15000), CSTP with additional DFF (Id 15001–22500)

in Fig. 17, with slightly less FC from 0.7 to 0.8, are quite similar. In Fig. 18, one can see that the FC depends on the

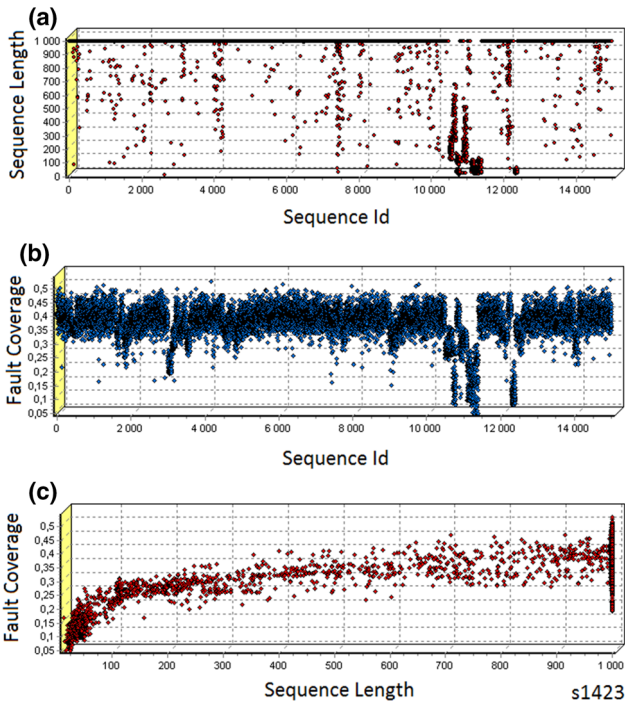

Fig. 18 Simulation graph for s1423 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–15000), CSTP with additional DFF (Id 15001–22500)

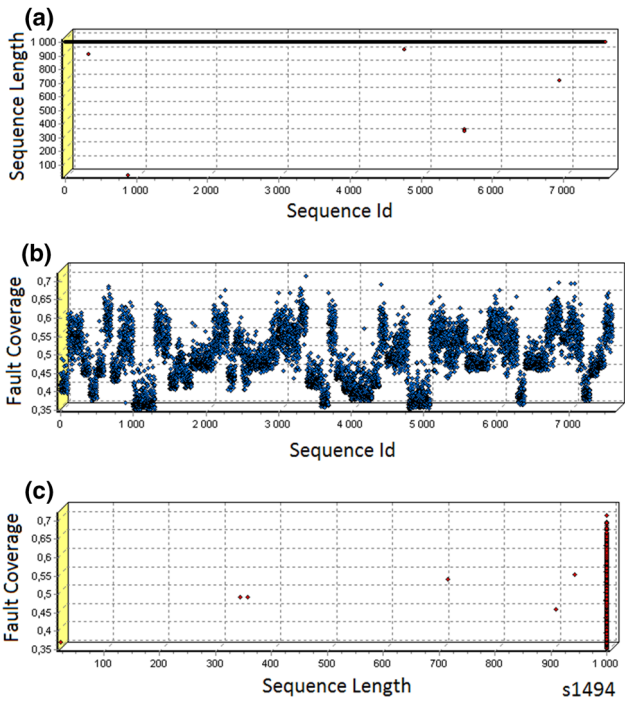

Fig. 19 Simulation graph for s1494 **a** sequence length vs. sequence id, **b** FC vs. sequence id and **c** FC vs. sequence length. STP (Id 1–7500), CSTP (Id 7501–15000), CSTP with additional DFF (Id 15001–22500)

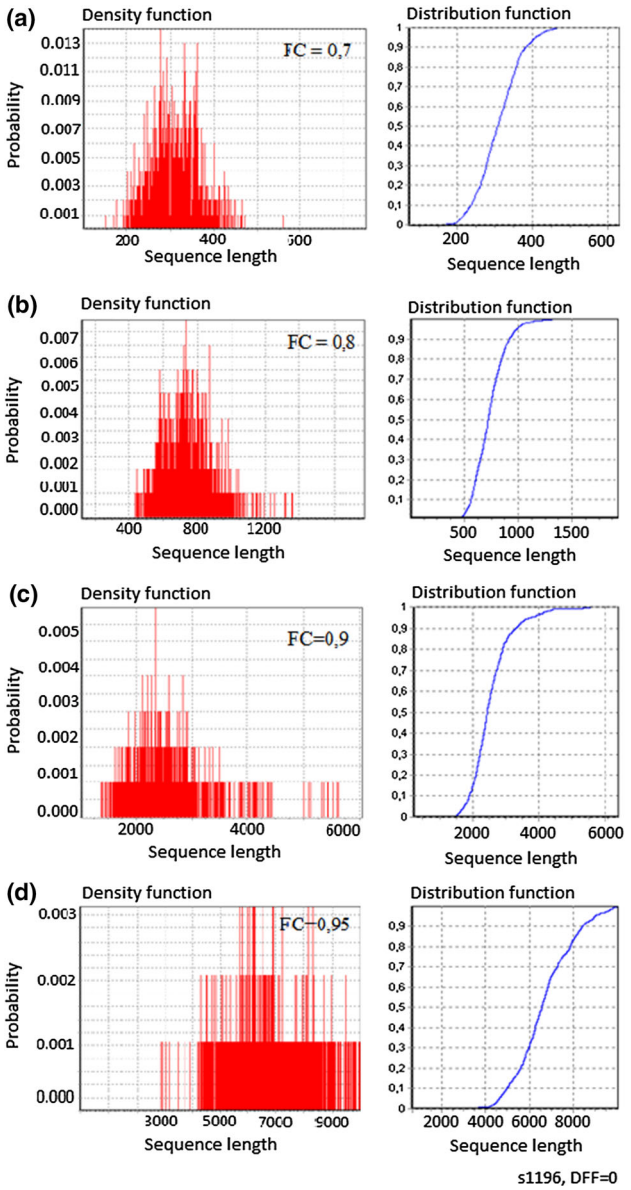

Fig. 20 s1196 density and distribution function that reached **a** FC ≥ 0.7, **b** FC ≥ 0.8, **c** FC ≥ 0.9, **d** FC ≥ 0.95

structure of the ATS oscillates from about 0.35 to 0.5, with a maximum value of 0.530. There is also a singularity in the range of 10500–12000. Compared to the plots Figs. 16 and 18 there are more different sequence lengths, and the FC value is in the lower range. As with the circuit in Fig. 17 and smaller in Figs. 11, 12 and 13, there are ATS configurations that generate short cycles. In Fig. 19, almost all ATS structures have generated sequences longer than 1000 test vectors. Restricting the sequence length to 1000 vectors is too restrictive for medium-sized circuits (Table 1). Despite long sequences, FC values oscillate between 0.35 and 0.7 depending on the ATS structure. There is a very high dependency between ATS, exactly STP and FC (island areas, clusters in plots). Infor-

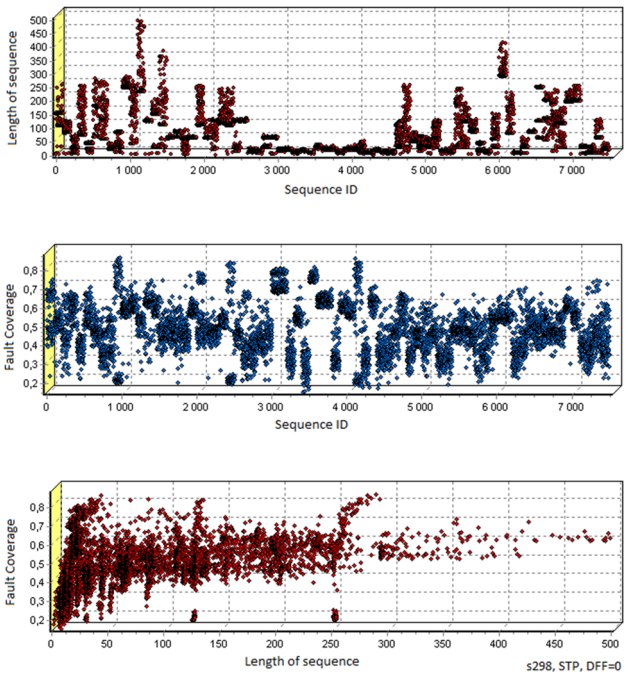

Fig. 21 STP for s298 without use of internal flip-flops

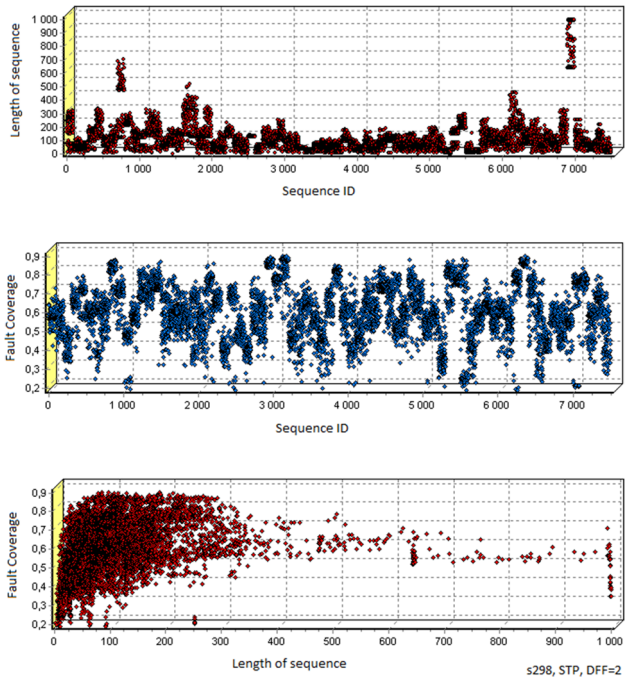

Fig. 23 STP for s298 with use of 2 internal flip-flops

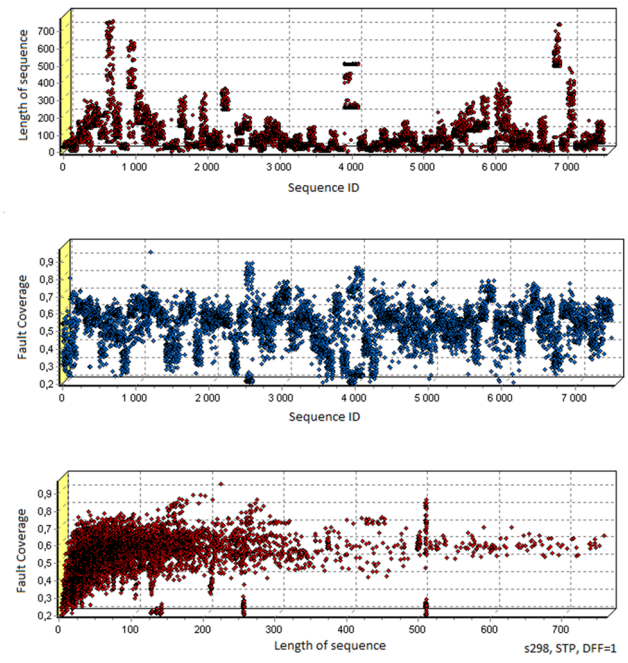

Fig. 22 STP for s298 with use of 1 internal flip-flop

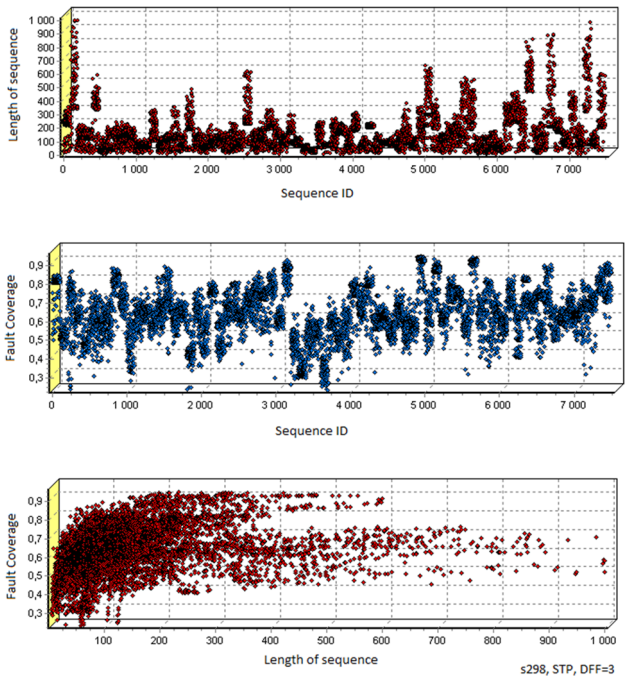

Fig. 24 STP for s298 with use of 3 internal flip-flops

mation about excessive limitation of 1000 sequence length vectors results from the developed stochastic faults model and the need to reduce the computer time-consuming circuit simulation, for example, for s1196 shown in Fig. 20. From Fig. 20, one needs to generate a long test sequence, for example, to get FC = 0.95 (Fig. 20d) probability of close 1 would generate a sequence of 10,000 vectors. This is obviously a necessary condition, but not sufficient due to the variety of ATS structures researched. Figures 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34 and 35 show in detail the effect of incremental including internal memory elements of a sequential circuit into STP structure on diagnostic performance of s298 circuit testing. With the increase

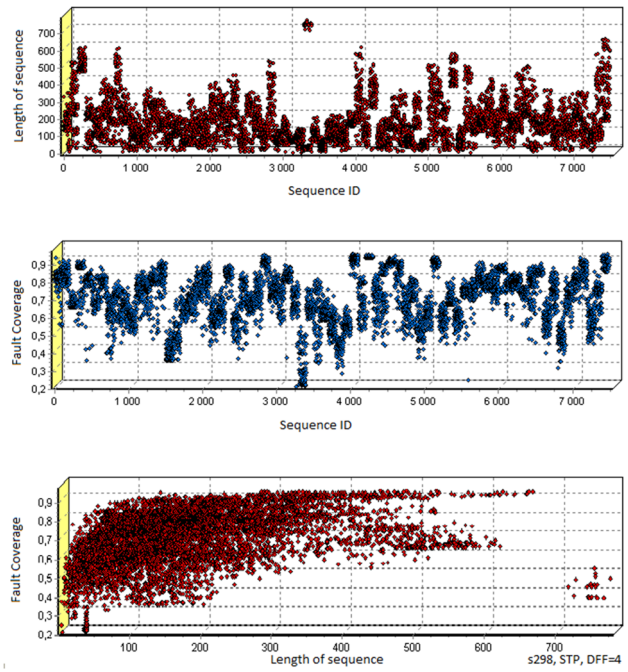

Fig. 25 STP for s298 with use of 4 internal flip-flops

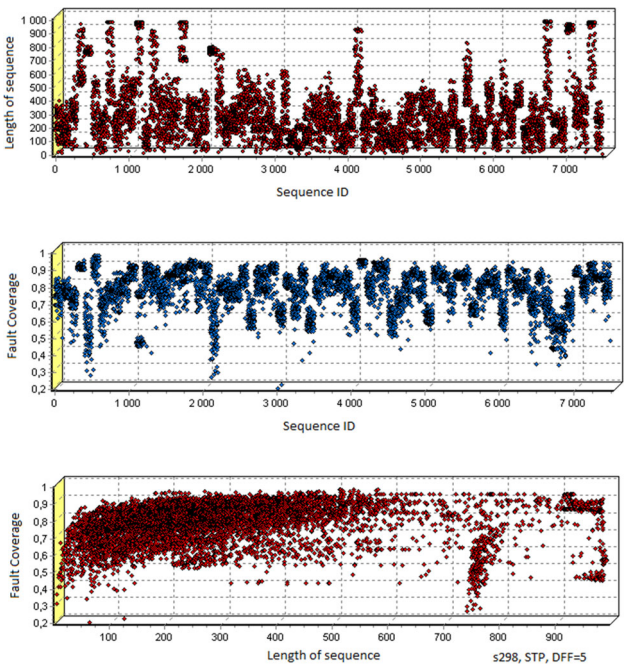

Fig. 26 STP for s298 with use of 5 internal flip-flops

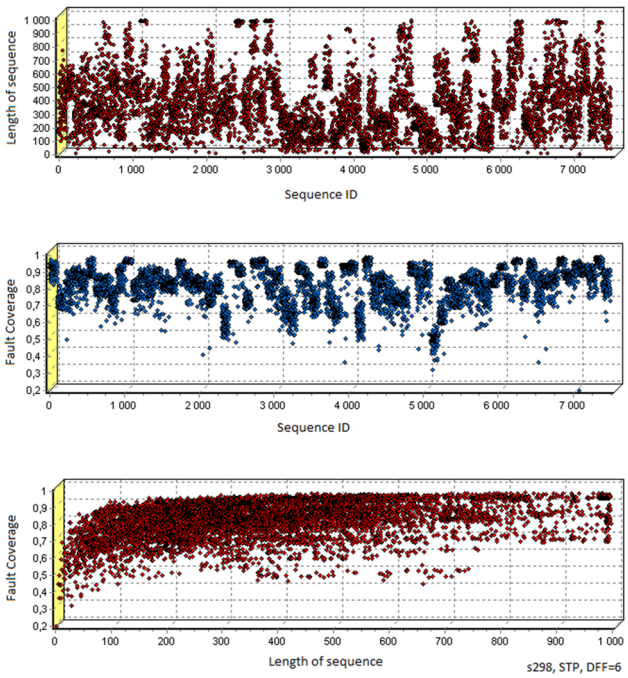

Fig. 27 STP for s298 with use of 6 internal flip-flops

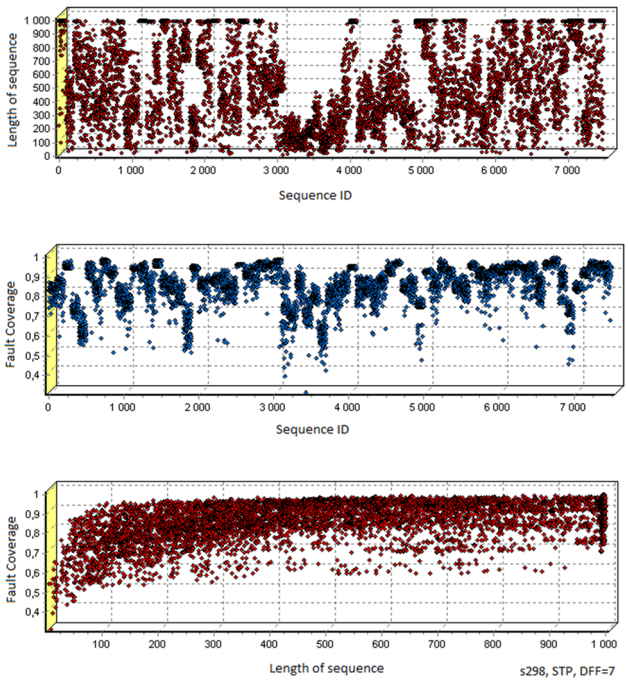

Fig. 28 STP for s298 with use of 7 internal flip-flops

in the number of included memory elements, STP increases the sequence length, but also increases FC. This incorporation of memory elements into STP allows one to convert a sequential circuit into a combinational circuit whose testing process is simplified. In a combinational circuit, the output depends entirely on the input and in the sequential circuit also on the state of the internal memory. The ability to disconnect system memory (MM) results from the use of special, e.g., BILBO multifunctional registers. Disconnecting the internal memory of the sequential system during testing is a standard and widely used approach.

In Table 3, the FC values obtained for a subset of ISCAS'89 are presented. In test simulation procedure, the

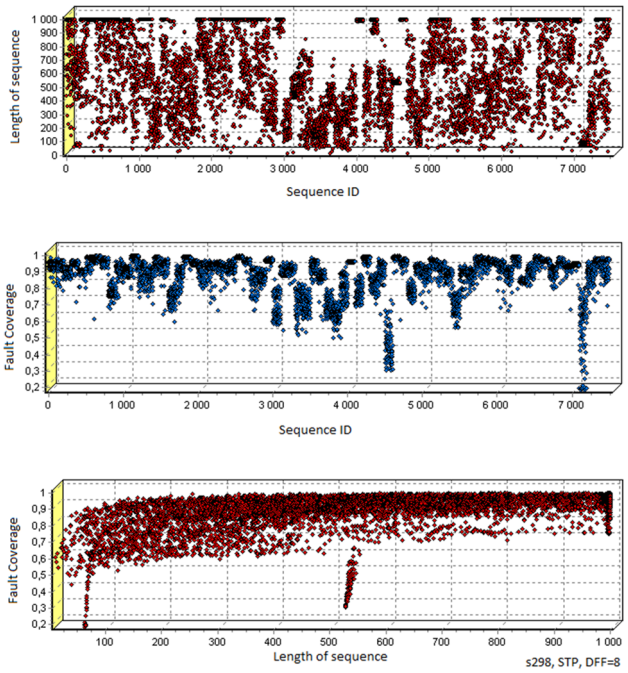

Fig. 29 STP for s298 with use of 8 internal flip-flops

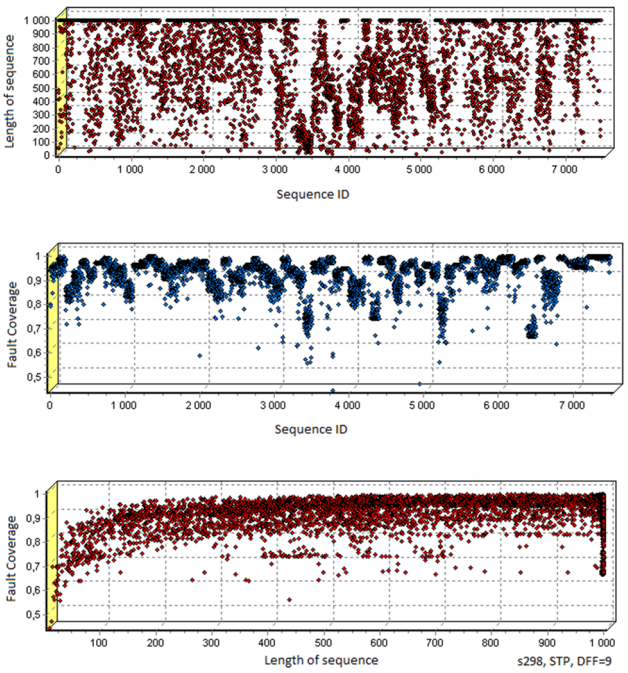

Fig. 30 STP for s298 with use of 9 internal flip-flops

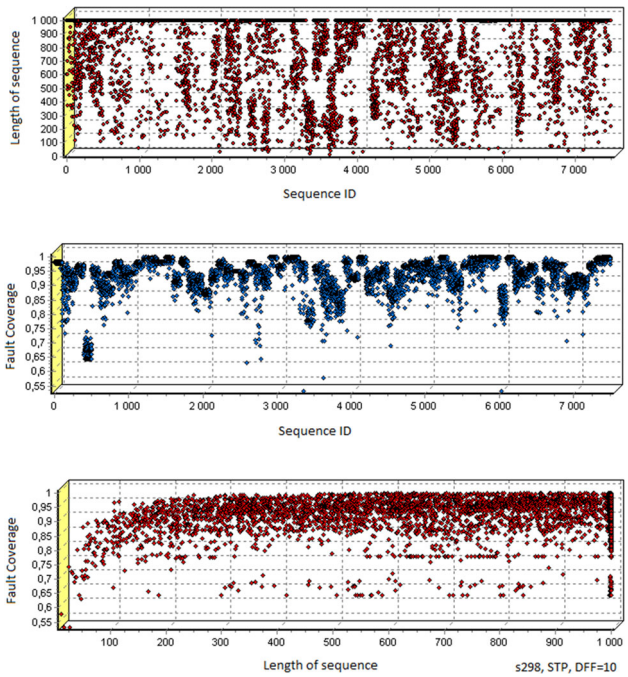

Fig. 31 STP for s298 with use of 10 internal flip-flops

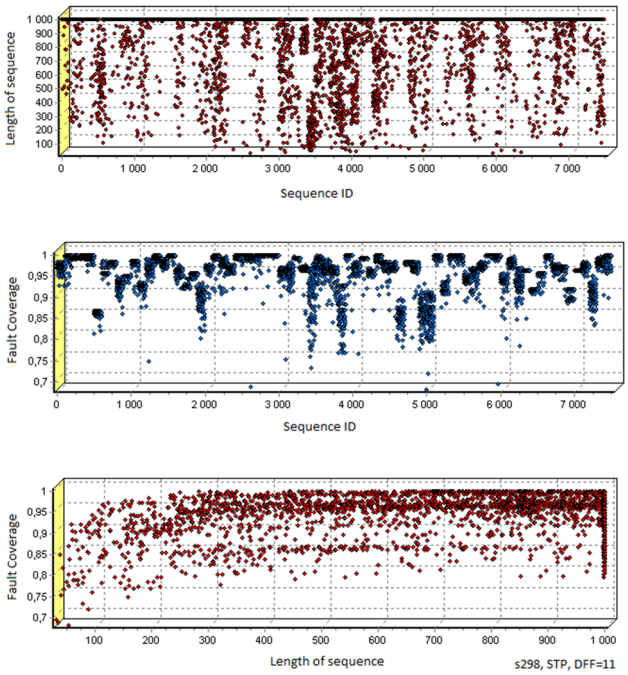

Fig. 32 STP for s298 with use of 11 internal flip-flops

block of sequential circuit memory is not disconnected. In other case, the greatest value of FC = 0.997 was calculated for the structure of the CSTP including all s298 circuit flip-flops (MM) (Fig. 35).

As mentioned earlier, to increase the circuit testability, all or some of circuit (MM) elements should be included into self-testing register (STP or CSTP) (Fig. 36). Disconnecting the (MM) elements, converts the sequential circuit (CUT) to a combinational one, that is easier to test. The effect of disconnecting memory elements from the sequential circuit s298 is shown in Figs. 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34 and 35.

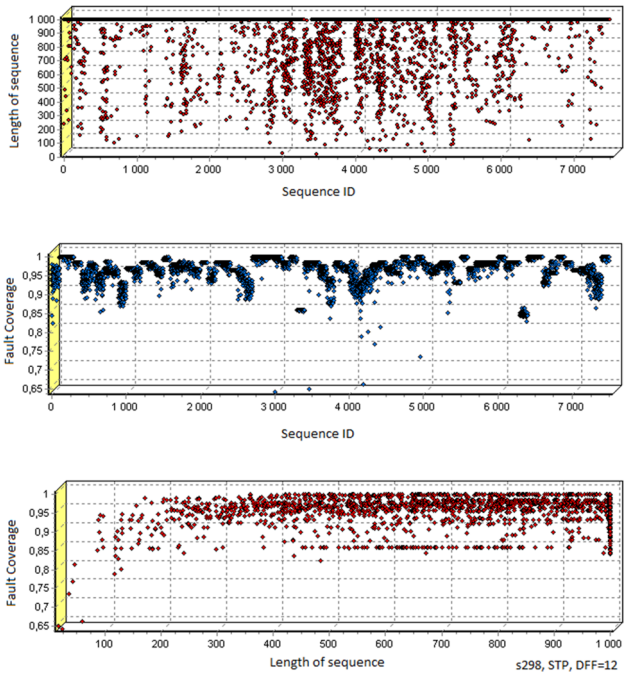

Fig. 33 STP for s298 with use of 12 internal flip-flops

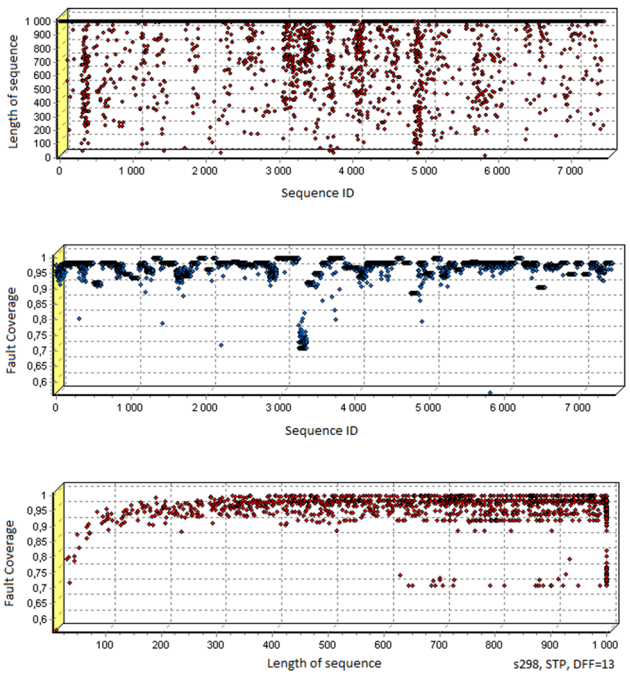

Fig. 34 STP for s298 with use of 13 internal flip-flops

5 Conclusions and future work

Genetic algorithm was able to find appropriate solutions, i.e., the structures of ATS, which were able to generate adequate quality test sequences and the high value of FC was possible to obtain. The conducted experiments show that it is possible to identify ATS structures with a high correlation coefficient between the sequence length and FC. Finding a suitable ATS structure evolutionary with those properties requires the circuit test simulation without faults, and therefore significantly affects the efficiency of the search (exploration) of the solution space. Next, the diagnostic efficacy of ATS structure has to be finally confirmed by simulation circuits with faults, which is much more time complex task. The genetic algorithm used as the method of designing ATS structures was controlled by standard values of crossover and mutation operators. Crossover and mutation processes are specific for encoding connection matrices and exclude unacceptable forms. It should be noted that memory block of circuit operates in accordance with the specification and has not been disconnected, and so the same process of testing simulation was significantly complicated. As demonstrated in the research, the use of even a small subset of the internal memory of the sequential circuit as part of the STP/CSTP register allows for significant increase of fault coverage value. The results of sequential circuits testing even without disconnecting internal memory are comparable to other methods known from the literature. Including circuit memory elements into the STP/CSTP register radically increases the sequence length and FC, but imposes specific system solutions on the sequential circuit memory block, e.g., BILBO register. Research analysis has shown that there is a correlation between FC and the length of the test sequence. The value of the correlation coefficient reaches more than 0.9 for certain ATS structures, rarely reaching small values less than 2. The high correlation within specific ATS structures makes

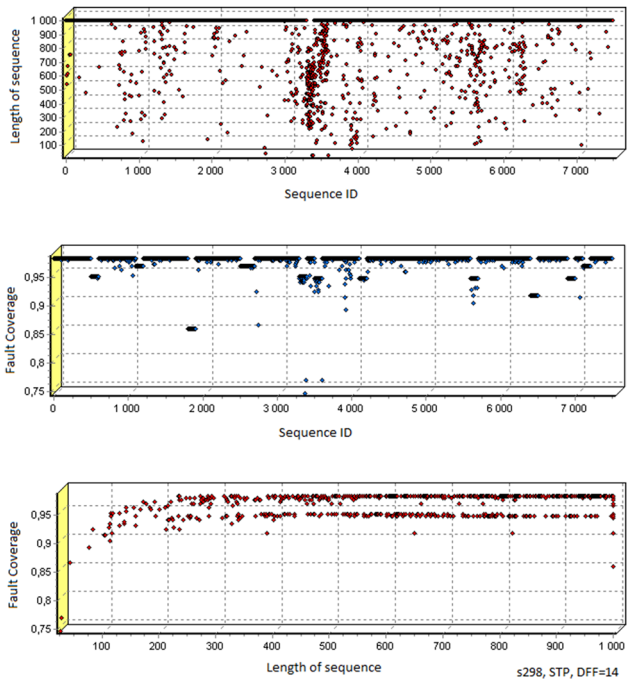

Fig. 35 STP for s298 with use of 14 internal flip-flops

Table 3 ISCAS'89 benchmark fault coverage results

ISCAS'89	s208.1	s298	s349	s382	s444	s641	s713	s820	s953	s1196	s1238	s1423	s1494
GA-ATS	**1.000**	**0.913**	**0.991**	0.891	**0.924**	**0.921**	**0.877**	0.598	0.983	0.894	0.812	0.530	0.714
GATTO	0.679	0.886	NA	0.917	0.890	0.873	0.826	0.918	NA	0.995	0.946	0.963	0.847
CA2	0.673	0.876	0.973	0.877	0.863	0.873	0.826	0.598	0.983	0.832	0.812	0.882	0.877
ATPG	0.677	0.876	0.978	0.949	0.926	0.873	0.826	0.949	0.990	0.997	0.945	0.896	0.964
ATPG-LP	1.000	0.877	0.984	0.927	0.924	0.874	0.877	0.529	0.991	0.995	0.960	0.973	0.972
GATTO+	0.697	0.886	0.978	0.947	0.924	0.873	0.826	0.941	0.991	0.995	0.944	0.967	0.960
CSTP	0.748	0.886	0.833	0.883	0.831	0.834	0.841	NA	NA	0.641	0.622	NA	NA
FSM-ATPG	0.976	0.913	0.954	0.286	0.317	0.887	0.848	0.965	0.995	0.999	0.971	0.445	0.984
CA-GA	1.000	0.893	0.959	0.943	0.924	0.886	0.846	0.528	0.993	0.894	0.954	0.445	0.960
HITEC	NA	0.860	0.954	0.754	0.787	NA	NA	0.956	NA	NA	NA	0.518	NA
HITEC-BDD	NA	0.860	0.957	0.779	0.820	NA	NA	0.956	NA	NA	NA	0.564	NA
CCPS	1.000	0.893	0.968	0.943	0.924	0.886	0.846	0.528	0.993	0.894	0.854	0.866	0.960
CA 90/150	0.948	0.238	0.610	0.165	0.138	0.886	0.847	0.456	0.994	0.942	0.915	0.635	0.559
SELFISH GA	1.000	0.895	0.806	0.942	0.923	0.887	0.847	0.479	0.994	0.953	0.919	0.876	0.929

Highlighted GA-ATS is the approach presented in this paper

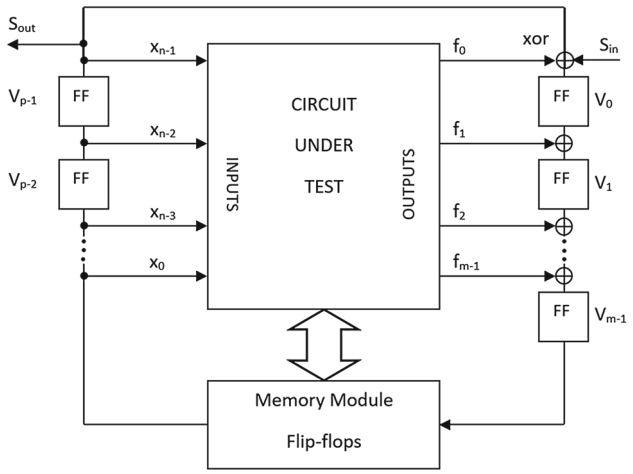

Fig. 36 Memory module under test disconnection

it possible to use deterministic local optimization algorithms or evolutionary exploitation process to select the appropriate connection matrices, additional linear feedback and type of flip-flops used to create STP/CSTP register.

In the future, research will be conducted on the classical testing structures such as independent test pattern generators (LFSRs) and test response compactors (MISRs, SISRs) or cellular automata. It is also very important to enrich used faults model by other types of faults such as open line, bridging and cross-talk faults. Other research related to the selection of other evolutionary algorithms for ATS design will also be made.

References

1. EDCC-2 Companion Workshop on Dependable Computing: Workshop Proceedings. Silesian Technical University, Poland, 15 May 1996. ISBN 83-906582-1-6 (1997)
2. Moorthy, P., Bharathy, S.: An efficient test pattern generator for high fault coverage in built-in-self-test applications. In: IEEE Fourth International Conference on Computing, Communications and Networking Technologies, India (2013)
3. Liu, T., Liu, P., Liu, Y.: An efficient controlled LFSR hybrid BIST scheme. IEICE Electron. Express **15**(8), 1–6 (2018)
4. Hławiczka, A., Badura, D.: Condensed circular self-test path: a low cost circular BIST. In: IEEE European Test Workshop—ETW'96, France (1996)
5. Devika, K.N., Bhakthavatchalu, R.: Design of reconfigurable LFSR for VLSI IC testing in ASIC and FPGA. In: International Conference on Communication and Signal Processing, pp. 928–932, India (2017)
6. Stroud, Ch., Sunwoo, J., Garimella, S., Harris, J.: Built-in self-test for system-on-chip: a case study. In: IEEE ITC International Test Conference (2004)
7. Chodacki, M.: Genetic algorithm for self-test path and circular self-test path design. In: Nguyen, N., Tojo, S., Nguyen, L., Trawiński, B. (eds.) Intelligent Information and Database Systems (ACIIDS). LNCS, vol. 10192. Springer, Tokyo (2017)
8. Corno, F., Prinetto, P., Sonza Reorda, M.: A genetic algorithm for automatic generation of test logic for digital circuits. In: IEEE International Conference on Tools with Artificial Intelligence, Toulouse (1996)
9. Corno, F., Prinetto, P., Rebaudengo, M., Sonza Reorda, M.: A parallel genetic algorithm for automatic generation of test sequences for digital circuits. In: International Conference on High-Performance Computing and Network, Belgium (1996)

10. Corno, F., Prinetto, P., Sonza Reorda, M., Mosca, R.: Advanced techniques for GA-based sequential ATGs. In: IEEE Design & Test Conference, France (1996)
11. Corno, F., Patel, H.J., Rudnicki, E.M., Sonza Reorda, M., Vietti, R.: Enhancing topological ATPG with high-level information and symbolic techniques. In: International Conference on Circuit Design, ICCD98, USA (1998)
12. Corno, F., Sonza Reorda, M., Squillero, G.: Evolving cellular automata for self-testing hardware. In: Third International Conference on Evoluable Systems, ICES2000, UK (2000)